

*МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ*

КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ: ОСВІТА, НАУКА, ВИРОБНИЦТВО

НАУКОВИЙ
ЖУРНАЛ



Відповідальний редактор – професор д.ф-м.н., Пастернак Я.М.

№39 2020

*м. Луцьк
Видавництво Луцького національного технічного університету*

КОМП'ЮТЕРНО-ІНТЕГРОВАНІ
ТЕХНОЛОГІЇ:
ОСВІТА, НАУКА, ВИРОБНИЦТВО

№39 2020р.

РЕДАКЦІЙНА КОЛЕГІЯ:

Головний редактор:	
професор д.ф.-м.н., Пастернак Я.М.	(м. Луцьк)
Заступники головного редактора:	
проф., д.т.н. Андрушак І.Є.	(м. Луцьк)
доц., к.т.н. Пех П.А.	(м. Луцьк)
Відповідальний секретар:	
мол.наук.співробітник Свиридюк К.А.	(м. Луцьк)
Члени редакційної колегії:	
проф, PhD. Milosz Marek	(Польща, м. Люблін)
проф, PhD. Alison McMillan	(Великобританія)
проф, PhD. Дехтяр Ю.Д.	(Литва, м. Рига)
проф., д.т.н. Сайко В.Г.	(м. Київ)
проф., д.т.н. Мороз Б.І.	(м. Дніпро)
проф., д.т.н. Степанов М.М.	(м. Київ)
проф., д.т.н. Тарасенко В.П.	(м. Київ)
проф., д.т.н. Віноградов М.А.	(м. Київ)
проф., д.т.н. Мельник А.О.	(м. Львів)
доц., к.т.н Мельник К.В.	(м. Луцьк)
доц., к.ф.-м.н Мельник В.М.	(м. Луцьк)
доц., к.т.н. Багнюк Н.В.	(м. Луцьк)
доц.,к.т.н. Здолбіцька Н.В.	(м. Луцьк)
доц.,к.т.н. Костючко С.М.	(м. Луцьк)
доц., к.т.н. Драган О.В.	(м. Брест, Білорусія)
доц., к.т.н. Лотиш В.В.	(м.Луцьк)
доц., к.т.н. Гуменюк Л.О.	(м.Луцьк)
доц., к.т.н. Заблоцький В.Ю.	(м.Луцьк)
доц., к.т.н. Решетило О.М.	(м.Луцьк)

Адреса редколегії:

Луцький національний технічний університет,
кафедра комп'ютерної інженерії та кібербезпеки
вул. Львівська 75, ауд.141
м.Луцьк, 43018
тел. (0332) 74-61-15
E-mail: cit@lntu.edu.ua,
сайт журналу: cit-journal.com.ua

Журнал засновано у грудні 2010 р.
Свідоцтво про реєстрацію КВ № 16705–5277 Р.
Засновник: Луцький національний технічний університет

**Рекомендовано до друку Вченою радою
Луцького національного технічного університету
(протокол №9 засідання від 29.05.2020)**

Журнал рішенням МОН України
наказом №515 від 16.05.2016р,
включено в перелік наукових фахових видань.

**Видання індексується у
наукометричних та реферативних базах:**

[Index Copernicus Journal Master List](#)

[Open Academic Journals Index](#)

[Academic Resource Index ResearchBib](#)

[Rootindexing](#)

[Information Matrix for the Analysis of Journals](#)

ISSN 2524-0560 (Online)

ISSN 2524-0552 (Print)

ЗМІСТ

АВТОМАТИКА ТА УПРАВЛІННЯ	
Булатецька Л.В., Булатецький В.В. Особливості вивчення мови запитів SQL в профільному курсі інформатики закладів загальної середньої освіти.	5
Головачук І.П., Величко В.Л., Бурбан О.В., Бобокало С.Ю. Обґрунтування фізичних параметрів та програмна реалізація 3d голографічних проєкцій.	11
Журило А.Г., Сівак Є.М. Побудова перспективи кола, що належить предметній площині.	17
Каганюк О.К., Мельник В.М. Математична модель рухомого об'єкта для розрахунку похибки. <i>[Каганюк А.К., Мельник В.М. Математическая модель расчета параметров регулятора для подвижного объекта.]</i>	22
Касянчук Д.П. Порівняльний аналіз генераторів документації.	30
Клехо О.В., Лехіцький Т.В. Формування фахових компетентностей студентів педагогічного коледжу в умовах професійно-практичної підготовки.	35
Костючко С.М., Кирилюк Л.М., Кались О.В., Сібанда З.Ф., Гаврилюк С.А. Побудова матриці монодромії виконавчого об'єкта нелінійної системи. <i>[Kostiuchko S.M., Kyryliuk L.M., Kalys O.V., Sibanda Z.F., Havryliuk S.A. The monodromy matrix construction for executive object of a nonlinear system.]</i>	40
Кузьміна О.М., Яремко С.А. Підвищення ефективності управління бізнес-процесами на основі імітаційного моделювання.	44
Лепкий М.І., Подоляк В.М., Кошелюк В.А. Апаратне і програмне забезпечення створення та використання 3D-турів.	50
Лишук В. В., Євсюк М. М., Селепина І. Р., Копилець Н. Ю. Математичні моделі пристроїв перетворювальної техніки.	55
Лупенко С.А., Литвиненко Я.В., Стадник Н.Б., Зозуля А.М., Сверстюк А.С. Умовний циклічний випадковий процес дискретного аргументу як узагальнена математична модель циклічних сигналів із подвійною стохастичністю.	60
Мамчич Т.І., Миронюк Л.П., Ройко Л.Л. Досвід використання інформаційно-комунікаційних технологій при викладанні математичних дисциплін в умовах дистанційного навчання.	70
Мартиненко А.А., Мороз Б.І., Гуліна І.Г. Інтелектуальна система підтримки прийняття рішень ідентифікації культурних цінностей. <i>[Martynenko A.A., Moroz B.I., Hulina I.G. An intelligent decision support system for cultural property identification.]</i>	78
Марченко О.О., Марченко О.І. Модель ресурсів неоднорідної розподіленої комп'ютерної системи з локальними зв'язками та її граф.	83
Поплавська Г.В. Аналіз застосування онлайн сервісу Google Classroom для організації дистанційного навчання.	89
Приходько А.О., Приходько О.С. Розробка універсальної установки для механічних випробувань матеріалів.	94
Савицький Т.П., Орлова М. М. Використання штучного інтелекту для системи розумного будинку.	99
Федотов В. В. Математична модель збудження спінового фотоструму у наноструктурах антиферромагнітних діелектриків.	105
Чибіряк Я. І., Коноплянченко Є. В., Марченко А. В. Технологічні закономірності і математичні моделі синтезу раціональної послідовності складання виробів. <i>[Чибиряк Я. И., Коноплянченко Е. В., Марченко А. В. Технологические закономерности и математические модели синтеза рациональной последовательности сборки изделий.]</i>	110

ІНФОРМАТИКА ТА ОБЧИСЛЮВАЛЬНА ТЕХНІКА

Войтенко Є. Д., Орлова М. М. Аналіз вразливості безпеки функціонування BGP та причин складності боротьби з ними.	117
Гринюк С.В., Поліщук М.М. Використання технології шифрування інформації для безпечної передачі в мережі.	122
Козубцова Л.М. Удосконалена методика діагностування кібернетичної захищеності інформаційної системи з урахуванням деструктивних кібернетичних впливів.	127
Костючко С.М., Сахнюк А.А., Мельник К.В. Обхід захисту сайтів за допомогою SQL-ін'єкцій та захист від них.	136
Курдус А.О. Порівняльний аналіз онлайн-компіляторів.	141
Марценюк В.П., Дідманідзе І.Ш., Сверстюк А.С., Андрущак І.Є., Рудь К.І. Автоматизований метод побудови експлойтів в аналізі тестування програмного забезпечення. <i>[Martsenyuk V.P., Didmanidze I.Sh., Sverstiuk A.S., Andrushchak I.Ye., Rud K.I. Automated method of building exploits in analysis software testing.]</i>	146
Мельник К. В., Мельник В.М., Григоришин А. М. Автоматичний збір інформації (парсинг) в мережі.	151
Мельник К.В., Костючко С.М., Мельник Д.С. Оптимізація ведення та аналітики фінансів за допомогою мобільного додатку на основі ОС Android.	157
Мельник В.М., Мельник К.В., Кузьмич О.І., Багнюк Н.В., Кравець О.Р. Дослідження покращення внутрішніх та зовнішніх параметрів швидкодії зв'язку на кластері комунікуючих віртуальних машин.	162
Міскевич О.І., Багнюк Н.В., Христинець Н.А., Марчевська О.Р. Автоматизація виявлення дефектної продукції методами машинного навчання	175
Муляр В.П. Розробка JavaFX-додатків із використанням Scene Builder.	181
Ніколіна І.І., Гулівата І.О. Моделювання кіберзлочинності як загрози цифровізації економіки.	190
Пех П.А., Костюк Ю.Ю., Кравченко М.Б. До питання конструювання класів з конструкторами різного типу.	197
Сіваковська О.М., Ліщина В.О., Ящук А.А, Матвій Ю.Я., Повстяна Ю.С. Аналіз особливостей стандартизації програмних продуктів та розподілених систем керування.	203
Христинець Н.А., Скупейко Д.С. Особливості проектування компонент мікроядра операційної системи засобами GCC, GNU Binary Utilities в композиції з мовами асемблера та С.	208
Христинець Н.А., Михалик А.В., Міскевич О.І Продуктивність технології CrossFire X при навантаженні відеоадаптерів мікропроцесорів AMD.	213

DOI: 10.36910/6775-2524-0560-2020-39-01

УДК: 005

Булатецька Леся Віталіївна, канд. фіз.-мат. наук, доцент

<https://orcid.org/0000-0002-7202-826X>

Булатецький Віталій Вікторович канд. фіз.-мат. наук, доцент

<https://orcid.org/0000-0002-9883-4550>

Східноєвропейський національний університет імені Лесі Українки, м. Луцьк, Україна

ОСОБЛИВОСТІ ВИВЧЕННЯ МОВИ ЗАПИТІВ SQL В ПРОФІЛЬНОМУ КУРСІ ІНФОРМАТИКИ ЗАКЛАДІВ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ

Л.В. Булатецька, В.В. Булатецький. **Особливості вивчення мови запитів SQL в профільному курсі інформатики закладів загальної середньої освіти.** Матеріал статті містить роз'яснення і методичні рекомендації для вивчення теми "мова запитів SQL" в профільному курсі інформатики закладів загальної середньої освіти. В роботі детально описано поняття цілісності даних в реляційній базі даних, проаналізовано ризики порушення цілісності даних при додаванні, оновленні та видаленні записів в багатотабличній базі даних та принципи побудови і порядок виконання запитів SQL на вибірку даних.

Ключові слова: методика вивчення, реляційні бази даних, мова запитів SQL, цілісність бази даних, між-табличні зв'язки.

Л.В. Булатецькая, В.В. Булатецкий. **Особенности изучения языка запросов SQL в профильном курсе информатики заведений общего среднего образования.** Матеріал статті содержит роз'яснення і методическіе рекомендації к изучению темы "язык запросов SQL" в профильном курсе информатики заведений общего среднего образования. В работе подробно описано понятие целостности данных в реляционной базе данных, проанализированы риски нарушения целостности данных при добавлении, обновлении и удалении записей в многотабличной базе данных, принципы построения и порядок выполнения запросов SQL на выборку данных.

Ключевые слова: методика изучения, реляционные базы данных, язык запросов SQL, целостность базы данных, меж-табличные связи.

L.V. Bulatetska, V.V. Bulatetsky. **Features of studying the language of SQL queries in the profile course of informatics of institutions of general secondary education.** The material of the article contains explanations and methodological recommendations for the study of the topic "SQL query language" in the profile course of informatics of institutions of general secondary education. The paper describes in detail the concept of data integrity in a relational database, analyzes the risks of data integrity breach when adding, updating, and deleting records in a multi-table database, and the principles for constructing and performing SQL queries for fetching data.

Keywords: study methodology, relational databases, SQL query language, database integrity, inter-table relationships.

Постановка проблеми та аналіз досліджень. Зміст навчальної програми профільного рівня та рівня стандарту вивчення інформатики для здобувачів 10-11 класів закладів загальної середньої освіти, містить розділ "Бази даних", який вивчається в 11 класі [1,2]. Вивчення даної теми проходить за підручником Завадського І. "Основи баз даних" [3]. В даному підручнику автор основну увагу приділяє проектуванню реляційних баз даних. Знайомство з реляційними базами даних відбувається на прикладі системи керування базами даних (СКБД) Access. У розділах розширеної версії даного підручника пропонується вивчення мови запитів SQL (Structured Query Language). Мову SQL вважають декларативною мовою, на відміну від процедурних мов, на яких пишуться програми. Це означає, що вирази на мові SQL описують, що потрібно зробити, а не яким чином. Після вивчення цієї теми, здобувач повинен знати та розуміти конструкції мови запитів, створювати та виконувати запити на вибірку даних з однієї та кількох зв'язаних таблиць, групувати дані, виконувати запити на введення даних в таблицю, їх зміну і видалення [1]. Автори робіт [3, 4], аналізуючи методичні особливості вивчення розділу "Бази даних" в закладах загальної середньої освіти, виділяють складності у розумінні окремих понять та теоретичних положень. Основні труднощі, які виникають при побудові запитів SQL у здобувачів загальної середньої освіти – це розуміння поняття обмеження цілісності відношень в реляційних базах даних та принципів організації зв'язків між таблицями [4].

Метою роботи є дослідження особливостей методики вивчення мови запитів SQL теми "Бази даних" в профільному курсі інформатики закладів загальної середньої освіти.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Для того, щоб розуміти написання конструкції запитів на вибірку та зміну даних, здобувачі загальної середньої освіти, повинні розуміти окремі поняття теорії реляційних баз даних. Реляційна модель даних була винайдена британським ученим Едгаром Франком Коддом, який вперше використав для моделювання даних математичні принципи теорії множин і математичної логіки. Щоб розуміти суть

завдання створення реляційної бази даних, а також операцій над даними, достатньо розглянути на теоретичному (абстрактному) рівні всього лише кілька основних положень теорії відношень. Таблиці, з яких складається будь-яка реляційна база даних, являють собою деякі відношення, а відношення є не чим іншим, як множинами. У термінології реляційних баз даних рядки таблиці називаються записами (кортежами), стовпці – полями (атрибутами), множину всіх можливих значень в стовпці – доменами. Всі запити до бази даних, спрямовані на вилучення з неї потрібних записів, інтерпретуються як інструкції з виконання тих чи інших операцій, що, в результаті є операціями алгебри множин [5].

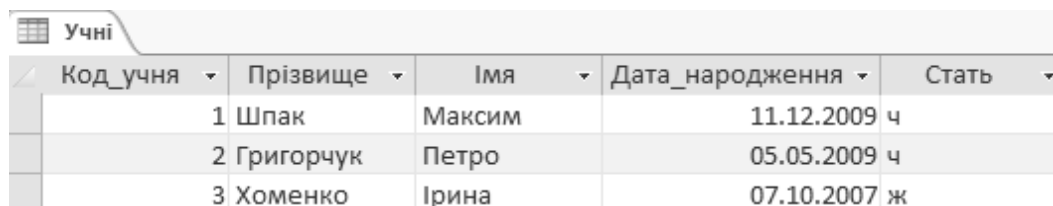
Будь-яке відношення реляційної моделі даних має наступні основні характеристики:

- в кожному рядку таблиці мають міститися дані, які відповідають деякому об'єкту або його частині;
- в кожному стовпці повинні знаходитися дані, що відповідають одному з атрибутів відношення;
- в кожній клітинці таблиці повинне знаходитися тільки єдине значення;
- у кожного стовпця повинно бути унікальне ім'я;
- всі рядки (записи) у таблиці повинні бути різними;
- порядок розташування стовпців і рядків у таблиці не має значення [5].

На практиці доводиться проектувати базу даних як набір декількох таблиць, які спочатку порожні і тільки з часом заповнюються конкретними даними. Однак кожна таблиця окремо, а також сукупність таблиць, зазвичай є не випадковими комбінаціями атрибутів, довільно розподіленими між різними таблицями. Як окремі таблиці, так і вся їх сукупність, що називаються базою даних, володіють деякою цілісністю, яка виражається через різного роду обмеження, що накладаються на значення стовпців і зв'язки між ними.

Здобувачі загальної середньої освіти повинні чітко розуміти поняття цілісності даних в реляційній базі даних. Якщо таблиця в базі даних повністю відповідає деякому об'єкту реального світу, то говорять, що вона володіє семантичною цілісністю і моделює (представляє) цей об'єкт. Наприклад в базі даних "Школа" – таблиця "Учні"(з полями "Код_учня", "Прізвище", "Ім'я", "Дата народження", "Стать") моделює об'єкт реального світу "учень", тобто володіє семантичною цілісністю (Рис.1.). У таблиці, що має семантичну цілісність, повинен бути первинний ключ (PRIMARY KEY), який відіграє важливу роль при створенні таблиць реляційної бази даних. Первинний ключ – це множина з одного або декількох атрибутів, яка однозначно визначає (ідентифікує) весь запис у відношенні. Значення первинного ключа повинні бути унікальними і визначеними.

В таблиці "Учні" в якості первинного ключа взято поле "Код_учня". Первинний ключ задається при створенні таблиці, або після її створення засобами модифікації таблиці. Те, що поле "Код_учня" є первинним ключем, означає, що всі дані в даному полі є визначеними і такими, що не повторюються.



Код_учня	Прізвище	Ім'я	Дата_народження	Стать
1	Шпак	Максим	11.12.2009	ч
2	Григорчук	Петро	05.05.2009	ч
3	Хоменко	Ірина	07.10.2007	ж

Рис. 1. Вміст таблиці "Учні" бази даних "Школа"

При додаванні даних в таку таблицю, потрібно слідкувати, щоб не порушувалось обмеження цілісності первинного ключа. Наприклад, наступні два запити на додавання записів в таблицю "Учні" за допомогою команди INSERT, будуть виконані, оскільки значення стовпця, що визначає первинний ключ є визначеними і відрізняються один від одного.

```
INSERT INTO Учні (Код_учня, Прізвище, Ім'я, Дата_народження, стать) VALUES (4, 'Семенюк', 'Петро', #05.02.200#);
```

```
INSERT INTO Учні (Код_учня, Прізвище, Ім'я, Дата_народження, стать) VALUES (5, 'Семенюк', 'Петро', #05.02.2005#);
```

Якщо ми спробуємо виконати наступний запит на додавання даних:

```
INSERT INTO Учні (Код_учня, Прізвище, Ім'я, Дата_народження, стать) VALUES (3,  
'Семенюк', 'Петро', #05.02.2005#);
```

то цей запит не виконається, так як порушується умова визначення первинного ключа. Всі значення стовпця "Код_учня" повинні бути такими, що відрізняються, а ми пробуємо додати рядок зі значенням поля "Код_учня"=3, який вже існує в нашій таблиці.

Також при оновленні даних, ми не зможемо змінити значення поля "Код_учня" на значення, яке вже існує в базі даних. Тобто наступний запит на оновлення даних не виконається.

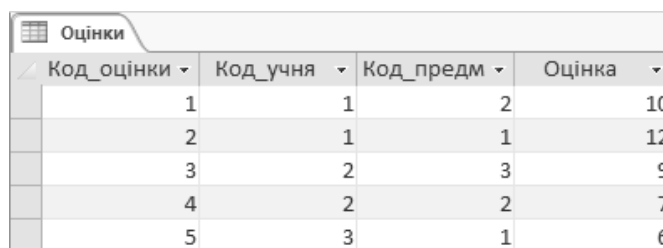
```
UPDATE Учні SET Код_учня = 3 WHERE Код_учня = 2;
```

Для того, щоб не слідкувати при додаванні та зміні даних за значеннями первинного ключа, в базі даних Access, в якості первинного ключа можна створити поле "Лічильник", тип поля вибрати "Автонумерація". Тоді значення поля "Лічильник" створюється автоматично під час кожного збереження запису.

Обмеження на допустимі значення для стовпця таблиці призначені для підтримки доменної цілісності. Ризик порушити доменну цілісність виникає при додаванні і оновленні записів. Обмеження доменної цілісності можна задати при створенні таблиці. Множину допустимих значень стовпця обмежує тип даних. Крім того, можна накласти на стовпець додаткові обмеження. Наприклад, якщо в стовпці "Дата_народження" таблиці "Учень" вказано значення #01.01.1970#, то ми не засумніваємося, що це помилкове значення. У цій же таблиці символічний стовпець "Стать" може приймати тільки два визначені значення, а не довільну комбінацію символів. Тобто, якщо ми спробуємо додати дані, або змінити дані на такі, які не задовольняють обмеженням, що накладаються на дані в стовпці, то такі запити не зможуть виконатися.

У базі даних, яка добре спроектована, міститься декілька таблиць, які пов'язані між собою. Досить складні бази даних проєктуються шляхом розробки множини окремих таблиць з подальшою установкою зв'язків між ними. Зв'язки між таблицями встановлюються за допомогою зовнішніх ключів (FOREIGN KEY). Так стовпець в одній таблиці може посилатися на стовпець іншої таблиці цієї ж бази даних. Подібні посилання являють собою обмеження посилкової цілісності бази даних і відіграють важливу роль при підтримці її загальної цілісності. Зв'язки між таблицями зазвичай несиметричні: одна таблиця залежить від іншої. Припустимо, у базі даних "Школа" є дві таблиці:

- Учні (Код_учня, Прізвище, Ім'я, Дата народження, Стать) – містить список учнів (Рис. 1);
- Оцінки (Код_оцінки, Предмет, Код_учня, Оцінка) – відомості про оцінки, які отримав учень з певного предмету (Рис. 2).



Код_оцінки	Код_учня	Код_предм	Оцінка
1	1	2	10
2	1	1	12
3	2	3	9
4	2	2	7
5	3	1	6

Рис. 2. Вміст таблиці "Оцінки" бази даних "Школа"

У таблиці "Учні" стовпець "Код_учня" є первинним ключем, тобто його значення визначені й унікальні (такі, що не повторюються). У таблиці "Оцінки" стовпець "Код_учня" не зобов'язаний мати унікальні значення, оскільки один і той же учень може мати декілька оцінок. Дані таблиці знаходяться в батьківсько-дочірньому відношенні: таблиця "Учні" - батьківська, "Оцінки" – дочірня. Даний зв'язок між таблицями організується шляхом оголошення стовпця "Код_учня" таблиці "Оцінки" зовнішнім ключем (FOREIGN KEY), який посилається на первинний ключ (PRIMARY KEY) "Код_учня" таблиці "Учні".

Використання зовнішніх ключів забезпечує збереження посилкової цілісності бази даних при зміні і видаленні записів. Разом з тим, наявність посилань породжує так звану проблему аномалій модифікації даних.

Якби таблиці "Учень" і "Оцінки" не були пов'язані, то при видаленні записів з таблиці "Учень" у таблиці "Оцінки" могли залишитися посилання на учня, про якого вже немає відомостей. Цей факт зазвичай розцінюється як аномалія видалення. У випадку визначення в таблиці "Оцінки" зовнішнього ключа "Код_учня" з таблиці "Учень" не вдасться видалити учня, якщо він отримав хоча б одну оцінку.

Наступний запит не виконається:

```
DELETE * FROM Учні WHERE Код_учня = 3;
```

Якщо потрібно видалити з бази даних все, що стосується певного учня, то спочатку видаляються записи в таблиці "Оцінка",

```
DELETE * FROM Оцінка WHERE Код_учня = 3;
```

а потім – з таблиці " Учні ".

```
DELETE * FROM Учні WHERE Код_учня = 3;
```

Аналогічна ситуація може статися і при оновленні або добавленні даних. Наприклад, у таблиці "Учень" спробувати змінити ідентифікатор учня, який має оцінки в таблиці "Оцінка" і, таким чином, зовнішньому ключу тепер немає на що посилатися. Це аномалія зміни. В даному випадку необхідно спочатку додати новий запис в таблицю "Учень", вказавши в ній необхідне значення "Код_учня", потім змінити в таблиці "Оцінка" всі старі значення "Код_учня" на ті, які були введені новому записі таблиці "Учень", а потім видалити з таблиці "Учень" запис зі старим "Код_учня".

Щоб в таблицях, пов'язаних зовнішнім ключем, не робити модифікацію даних в декілька етапів, в базі даних Access при створенні зв'язків, у вікні "Знаряддя бази даних/Зв'язки" потрібно поставити галочку "Каскадне оновлення даних", "Каскадне видалення пов'язаних полів" (Рис. 3.) Тоді при видаленні або оновленні даних з таблиці "Учень" всі пов'язані дані з таблиці "Оцінки" видаляться, або оновляться.

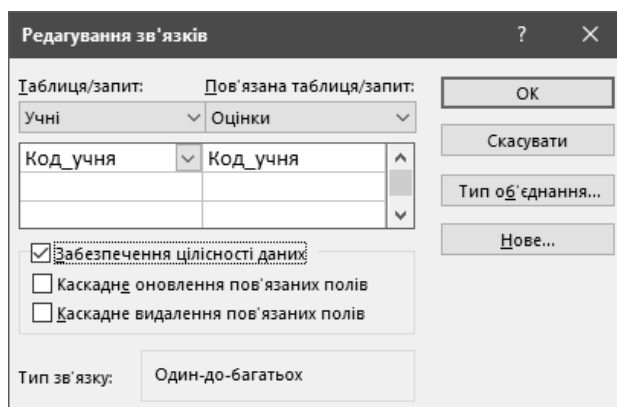


Рис. 3. Вікно редагування зв'язків в Access.

Щоб уникнути труднощів, при виконанні операцій добавлення даних в таблицю, їх зміни і видалення засобами мови запитів SQL здобувачі загальної середньої освіти повинні розуміти поняття первинного ключа, поняття обмеження цілісності відношень в реляційних базах даних та принципи організації зв'язків між таблицями. Створення зв'язків між таблицями часто сприймається здобувачами поверхнево, але воно є досить принциповим для розуміння роботи реляційної бази даних. Не розуміючи механізму зв'язків, забезпечення цілісності даних, здобувачі на практиці не можуть зв'язати одне поле з іншим, не можуть правильно побудувати запит, не розуміють, чому не виконується запит, який правильно структурно написаний. Більш детально про проблеми вивчення зв'язків у базах даних Microsoft Access описано в роботі [4].

Також труднощі виникають при розумінні написання запитів на вибірку даних з декількох таблиць.

Щоб правильно писати запити на вибірку даних з декількох таблиць, здобувачі загальної середньої освіти повинні розуміти принципи побудови та порядок виконання запитів SQL. SQL-вираз для вибірки даних, має вигляд [5]:

```
SELECT списокСтовбців  
FROM списокТаблиць  
WHERE УмовиПошуку  
GROUP BY списокстовбцівГрупування  
HAVING УмовиПошуку  
ORDER BY УмовиСортування;
```

Оператори SELECT (вибрати) і FROM (з) в SQL-виразі, що визначають вибірку даних, є обов'язковими. Для уточнення запиту на вибірку даних існують додаткові оператори:

- WHERE (де) – вказує записи, які повинні увійти в результатну таблицю;
- GROUP BY (групувати по) – групує записи по значеннях певних стовпців;

- HAVING (що мають, за умови) – вказує групи записів, які мають увійти до результатної таблиці;

- ORDER BY (сортувати по) – впорядковує записи.

Ці оператори не є обов'язковими. Їх можна зовсім не використовувати, або використовувати лише деякі з них, або всі одразу. Якщо застосовуються декілька операторів, то в SQL-виразі вони виконуються у певному порядку.

Порядок перерахування операторів в SQL-виразі не співпадає з порядком їх виконання. Однак знання порядку виконання операторів допоможе здобувачам загальної середньої освіти уникнути багатьох непорозумінь при побудові запитів. Отже, перераховані оператори SQL-виразу виконуються в наступному порядку, передаючи один одному результат у вигляді таблиці [5]:

1. FROM – вибирає таблицю з бази даних; якщо зазначено декілька таблиць, то виконується їх декартовий добуток і результуюча таблиця передається для обробки наступному оператору.

2. WHERE – з таблиці вибираються записи, що відповідають умові пошуку.

3. GROUP BY – створюються групи записів, відібраних за допомогою оператора WHERE (якщо він присутній в SQL-виразі); кожна група відповідає якому-небудь значенню стовпця групування.

4. HAVING – обробляє кожен із створених груп записів, залишаючи тільки ті з них, які задовольняють умові пошуку;

5. SELECT – вибирає з таблиці, отриманої в результаті використання перерахованих операторів, тільки вказані стовпці.

6. ORDER BY – сортує записи таблиці.

Якщо нам потрібно вибрати всі відомості про учнів (всі поля таблиці "Учень") і всі відомості про оцінки (всі поля таблиці "Оцінки"), то ми можемо написати запит на вибірку:

SELECT Учні.*, Оцінки.* FROM Учні, Оцінки;

В результаті ми отримуємо декартовий добуток таблиць Учень і Оцінки (табл.3). Декартовим добутком двох реляційних відношень називається відношення яке містить усі можливі з'єднання рядків першого відношення з рядками другого відношення. Тобто, в нашому випадку ми отримуємо таблицю вміст якої зображену на рис.4.

Учні.Код_учня	Прізвище	Імя	Дата_народження	Стать	Код_оцінки	Оцінки.Код_учня	Код_предмета	Оцінка
1	Шпак	Максим	11.12.2009	ч	1	1	2	10
2	Григорчук	Петро	05.05.2009	ч	1	1	2	10
3	Хоменко	Ірина	07.10.2007	ж	1	1	2	10
1	Шпак	Максим	11.12.2009	ч	2	1	1	12
2	Григорчук	Петро	05.05.2009	ч	2	1	1	12
3	Хоменко	Ірина	07.10.2007	ж	2	1	1	12
1	Шпак	Максим	11.12.2009	ч	3	2	3	9
2	Григорчук	Петро	05.05.2009	ч	3	2	3	9
3	Хоменко	Ірина	07.10.2007	ж	3	2	3	9
1	Шпак	Максим	11.12.2009	ч	4	2	2	7
2	Григорчук	Петро	05.05.2009	ч	4	2	2	7
3	Хоменко	Ірина	07.10.2007	ж	4	2	2	7
1	Шпак	Максим	11.12.2009	ч	5	3	1	6
2	Григорчук	Петро	05.05.2009	ч	5	3	1	6
3	Хоменко	Ірина	07.10.2007	ж	5	3	1	6

Рис. 4. Результат виконання декартового добутку таблиць "Учні" і "Оцінки" бази даних "Школа"

Аналізуючи результат виконання запиту, ми бачимо, що в деяких рядках містяться несумісні дані: в одному рядку дані про одного учня і дані про оцінки іншого учня. Для того, щоб отримати коректні дані, в результуючий набір нам потрібно взяти тільки відповідні дані, тобто ті дані де поле "Код_учня" з таблиці "Учень" рівне значенню "Код_учня" таблиці "Оцінка". Для цього ми відфільтруємо отримані дані за допомогою ключового слова WHERE.

SELECT Учні.*, Оцінки.* FROM Учні, Оцінки WHERE Учні.Код_учня=Оцінки.Код_учня;

Результат виконання даного запиту зображено на рис. 5.

Учні.Код_учня	Прізвище	Імя	Дата_народження	Стать	Код_оцінки	Оцінки.Код_учня	Код_предмета	Оцінка
1	Шпак	Максим	11.12.2009	ч	1	1	2	10
1	Шпак	Максим	11.12.2009	ч	2	1	1	12
2	Григорчук	Петро	05.05.2009	ч	3	2	3	9
2	Григорчук	Петро	05.05.2009	ч	4	2	2	7
3	Хоменко	Ірина	07.10.2007	ж	5	3	1	6

Рис. 5. Результат виконання запиту на вибірку інформації про учнів і їх оцінки.

Цей же результат можна отримати, використовуючи операції внутрішнього з'єднання таблиць (INNER JOIN) і ключового слова ON, за яким слідує умова відбору записів.

```
SELECT Учні.*, Оцінки.* FROM Учні INNER JOIN Оцінки ON  
Учні.Код_учня=Оцінки.Код_учня;
```

В даному випадку в результуючий набір з'єднуються тільки ті записи, в яких пов'язані поля обох таблиць співпадають. Внутрішні з'єднання відображають записи в результуючому наборі, як один запис. Якщо не знайдено відповідного значення у полі зв'язку пов'язаної таблиці, дані не відображаються взагалі (рис. 5). Незважаючи на те, що з'єднання з оператором WHERE є коротшим, краще використовувати оператор внутрішнього з'єднання INNER JOIN. Оператори SQL-виразу виконуються в певному порядку, передаючи один одному результат у вигляді таблиці. При використанні для з'єднання оператора WHERE, спочатку виконується оператор "FROM Учні, Оцінки", тобто виконується декартовий добуток двох відношень. Результат декартового добутку передається оператору WHERE, який відфільтровує не відповідні записи. Якщо припустити, що даних в таблицях буде багато і таблиць може теж бути більше ніж дві, то ця проміжна таблиця, яка передаватиметься оператору WHERE буде дуже великою на відміну від таблиці утвореної оператором INNER JOIN, який зразу в результуючу таблицю відбирає тільки відповідні дані. Тобто використання INNER JOIN для з'єднання двох і більше таблиць більш ефективно по відношенню з використанням оператора WHERE.

Висновки та перспективи подальшого дослідження. В роботі розглянуто складності у розумінні окремих понять та теоретичних положень з якими стикаються здобувачі загальної середньої освіти при вивченні теми "мова запитів SQL" розділу "Бази даних" в профільному курсі інформатики закладів загальної середньої освіти. Наведено відомості про поняття цілісності даних в реляційних базах даних і приклади написання запитів до бази даних, які ілюструють аномалії додавання, видалення та оновлення даних, що призводять до порушення цілісності даних. В роботі, також розглянуто порядок виконання операторів в запитах на вибірку даних. Знання порядку виконання операторів допоможе здобувачам загальної середньої освіти уникнути багатьох непорозумінь при побудові запитів. Вивчення поняття цілісності даних та розуміння принципів виконання запитів розвиває логічне мислення, допомагає засвоїти та закріпити навички конструювання запитів, привчає до контролювання можливих помилок при розв'язуванні практичних завдань.

Список бібліографічного опису

1. Інформатика. Профільний рівень // Навчальні програми для 10-11 класів [Електронний ресурс] — Режим доступу : <https://mon.gov.ua/ua/osvita/zagalna-serednya-osvita/navchalni-programi/navchalni-programi-dlya-10-11-klasiv>
2. Інформатика. Рівень стандарту // Навчальні програми для 10-11 класів [Електронний ресурс] — Режим доступу : <https://mon.gov.ua/ua/osvita/zagalna-serednya-osvita/navchalni-programi/navchalni-programi-dlya-10-11-klasiv>
3. Завадський І.О. Основи баз даних : навч. посіб. К. : Видавець І.О. Завадський, 2011. 192 с.
4. Шамшина Н.В. Методичні особливості вивчення зв'язків та типів об'єднання у базах даних Microsoft Access. Фізико-математична освіта. 2018. Випуск 1(15). С. 339–343.
5. Дунаев В. В. Базы данных. Язык SQL. СПб. : БХВ-Петербург, 2006. 288 с.

References

1. Informatyka. Profilnyy riven. Navchalni programi dlya 10-11 klasiv [Elektronnyy resurs]. URL: <https://mon.gov.ua/ua/osvita/zagalna-serednya-osvita/navchalni-programi/navchalni-programi-dlya-10-11-klasiv> (in Ukrainian)
2. Informatyka. Riven standartu. Navchalni programi dlya 10-11 klasiv [Elektronnyy resurs]. URL: <https://mon.gov.ua/ua/osvita/zagalna-serednya-osvita/navchalni-programi/navchalni-programi-dlya-10-11-klasiv> (in Ukrainian)
3. Zavadsky I.O. Fundamentals of the databases: textbook. K. : Publ. I.O. Zavadsky, 2011. 192 p. (in Ukrainian)
4. Shamshina N. Methodical Features Of Studying Relationships And Types Of Joins In Databases Microsoft Access. Physical and Mathematical Education. 2018. Issue 1(15). P. 339–343. (in Ukrainian)
5. Dunaev V.V. Database. SQL language. St. Petersburg: BHV- Petersburg, 2006. 288 p. (in Russian)

Рецензент: Мартинюк Олександр Семенович, доктор педагогічних наук, професор кафедри експериментальної фізики та інформаційно-вимірювальних технологій Східноєвропейського національного університету імені Лесі Українки,.

DOI: 10.36910/6775-2524-0560-2020-39-02

УДК: 004.42

Головачук Ігор Павлович, к.т.н., доцент,

<https://orcid.org/0000-0002-9026-8005>

Луцький національний технічний університет, м. Луцьк, Україна

Величко Володимир Леонідович, викладач-методист,

<https://orcid.org/0000-0001-9588-7246>

Бурбан Олександр Вікторович, к. ф.-м. н.,

<https://orcid.org/0000-0001-6562-3936>

Волинський коледж Національного університету харчових технологій, м. Луцьк, Україна

Бобокало Станіслав Юрійович, студент

Луцький національний технічний університет, м. Луцьк, Україна

ОБҐРУНТУВАННЯ ГЕОМЕТРИЧНИХ ПАРАМЕТРІВ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ 3D ГОЛОГРАФІЧНИХ ПРОЕКЦІЙ

Головачук І.П., Величко В.Л., Бурбан О.В., Бобокало С.Ю. **Обґрунтування фізичних параметрів та програмна реалізація 3d голографічних проєкцій.** У статті проаналізовано передумови для візуалізації інформації. Наведено спроби обґрунтувати геометричні параметри елементів пристрою для демонстрування 3D голографічних проєкцій. Представлено алгоритм та комп'ютерну програму для формування зображень проєкцій.

Ключові слова: голограма, 3D, голографічна проєкція, візуалізація, Python.

Головачук И.П., Величко В.Л., Бурбан А.В., Бобокало С.Ю. **Обоснование физических параметров и программная реализация 3d голографических проєкций.** В статье проанализированы предпосылки для визуализации информации. Приведены попытки обосновать геометрические параметры элементов устройства для демонстрации 3D голографических проєкций. Представлен алгоритм и компьютерную программу для формирования изображений проєкций.

Ключевые слова: голограмма, 3D, голографическая проєкция, визуализация, Python.

Holovachuk I., Velychko V., Burban O., Bobokalo S. **Substantiation of physical parameters and software of 3D-holographic projections.** The article analyzes the prerequisites for information visualization. Attempts to substantiate the geometric parameters of the device elements for demonstrating 3D holographic projections are presented. An algorithm and a computer program for forming projection images are presented.

Keywords: hologram, 3D, holographic projection, visualization, Python.

Постановка наукової проблеми. Завдяки стрімкому розвитку та проникненню інформаційно-комунікаційних технологій в усі галузі діяльності та життя людини, світ зазнав революційних змін. На даний час спостерігається надмірне навантаження споживачів інформаційними потоками, що вимагає застосування нових технологій, способів подання інформації, які були б ефективними у певних умовах або ситуаціях. Технологічний прогрес засобів для відтворення різноманітної візуальної інформації дозволяє використовувати новіші або трансформувати існуючі способи подання інформації у багатьох царинах діяльності людини, від побутової, до науково-промислової, що надає можливість ефективніше застосовувати людську працю як у вузькоспеціалізованих галузях (медицина, моніторинг наукометричних показників, авіація тощо), так і значно ширших (демонстрування творів мистецтва, реклама товарів, послуги, консультації тощо). Такі трансформаційні зміни не оминули науку та освіту.

Одним з ефективних способів подання інформації є візуалізація, а перетворення 2D (плоских) зображень на 3D (об'ємні) моделі надає можливості для природнього і більш зрозумілого людині сприйняття об'єктів. Одним зі способів представлення такого виду контенту є голографічні зображення. Більшість технологій демонстрації голограм є високовартісними та вимагають використання спеціалізованого обладнання, яке не завжди є безпечним для людини (наприклад лазер), тому на увагу заслуговує метод проєціювання динамічних або статичних стереоскопічних зображень на світлопрозорі поверхні, що у поєднанні з особливостями бінокулярного зору людини надає ілюзію об'ємності представлених 3D об'єктів та явищ. Завдяки простоті виготовлення, такі пристрої є доступним засобом, та інструментом для відтворення важливої та складної для досягнення інформації у освіті, науці, медицині, рекламі та інших галузях.

В процесі створення та функціонування подібного виду пристроїв виникає ряд проблем: вибір ефективного джерела проєціювання, підбір та обґрунтування відповідного світлопрозорого матеріалу для призми, створення досконалих 3D моделей, розробка та використання спеціалізованого програмного забезпечення та інш.

Аналіз досліджень. Сучасна наука і, зокрема, оптика перебуває у пошуку нових способів передачі візуальної інформації максимально реалістично. Серед них заслуговує на увагу голографія, яка найбільш повноцінно дозволяє зафіксувати, зберегти і відобразити об'єкти та явища оточуючого світу.

Найповніше дане завдання вирішує відеоголограма – стереозображення певним чином спроекційоване на світлопрозорих поверхнях, що створює ілюзію перебування тривимірних об'єктів у повітрі. Дослідження можливостей та способів використання голографічних проєкцій відображено у вітчизняних та закордонних публікаціях та наукових роботах [5], [6].

Даний вид проєкцій успішно використовується в рекламі, торгівлі, музейній справі та мистецтві, медицині, освіті та багатьох інших галузях людської діяльності. Голограми створюються з метою візуалізації 2D-фрагментів з багат шарових зображень, знятих за допомогою комп'ютерного томографа або магнітно-резонансного томографа з використанням мови програмування MATLAB та безлічі доступних функцій для обробки зображень [5].

Наступне дослідження [4] полягає у розробці методології збору та представлення тривимірних голограм високої чіткості, що створюються за допомогою комп'ютерного бачення та штучного інтелекту, з метою візуалізації 3D-об'єктів.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Ідея розробки пристрою для демонстрування голографічних 3D проєкцій виникла у викладачів кафедри інженерної та комп'ютерної графіки Луцького НТУ у 2018 році. Проєкт вперше було презентовано в рамках конкурсу пітчінг проєктів у Луцькому національному технічному університеті, що відбувся за сприяння British Council та ГО «Молодіжна Платформа». Далі було розроблено установку і представлено на конференції «Сучасна наука та освіта Волині» дослідну установку [1], [2].

Так як дію установки засновано на оптичних ефектах, в першу чергу було досліджено граничну товщину матеріалу для виготовлення стінок призми. Розрахунки здійснено на основі фізіологічних особливостей ока людини та властивостей матеріалу. Згідно з принципом роботи установки [2] (див. рис. 1) у випадку падіння світлового променя на прозору пластину, падаючий промінь частково відбивається від зовнішньої поверхні пластини, а частково від внутрішньої поверхні. Цей факт накладає певні обмеження на товщину прозорих пластин, які використовуються при виготовленні голографічних пірамід. Через особливості будови зорового апарату людини, при певних співвідношеннях товщини пластини до відстані від ока до зображення, око буде сприймати подвійне зображення у піраміді, що є небажаним.

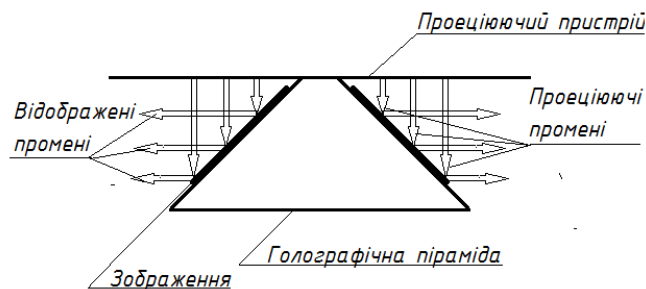


Рис. 1. Схема утворення 3D голографічного зображення

Такі обмеження накладаються роздільною здатністю ока людини. Роздільна здатність – це найменший кут, під яким роздільно видно два близько розташованих предмети. Вона залежить від діаметра вхідної зіниці, освітленості предмета і його форми, а також визначається розміром колбочок на сітківці ока (рис.2)

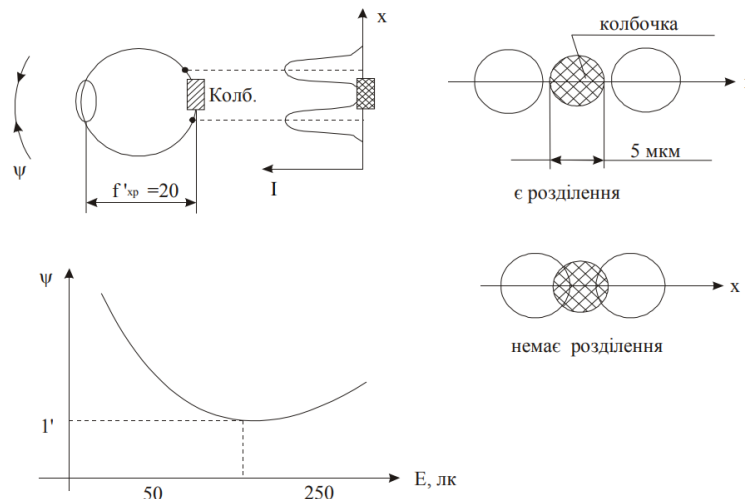


Рис 2. Схема роздільної здатності ока

При зменшенні або збільшенні діаметра зіниці ока від 2 мм роздільна здатність зменшується. Якщо діаметр зіниці ока менше 1 мм, то роздільна здатність зменшується через дифракцію світла, а більше 2 мм - в результаті впливу сферичної і хроматичної аберації і розсіяння світла всередині ока. Для нормального ока при освітленості 50...250 лк і діаметрі зіниці 2 мм гострота зору (γ) складає $60''$ [3].

Тобто, подвійне зображення буде сприйматись людиною у випадку, коли кутова відстань між двома зображеннями однієї і тієї ж точки буде більша $60''$.

Схематичне зображення відбивання світлових променів від прозорої пластини подано на рис. 3.

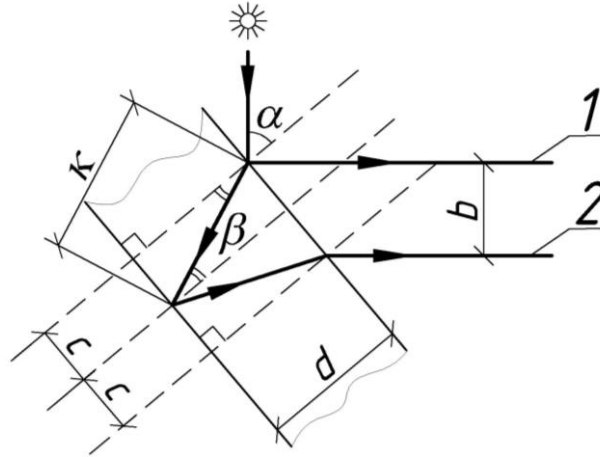


Рис. 3. Схематичне зображення відбивання світлових променів

Критерієм отримання одинарного голографічного зображення є $\Psi \leq 1'$, де Ψ - кутова відстань, або $tg\Psi \leq 3 \cdot 10^{-4}$. Кутова відстань залежить від відстані l між спостерігачем і зображенням та від відстані b між відбитими променями 1 і 2 (див. рис 3).

$$\frac{b}{l} = tg\Psi. \quad (1)$$

Відстань b між відбитими променями можна розрахувати врахувавши заломлення променя при переході через межу поділу матеріал-повітря. З трикутника утвореного заломленим променем, заломлюючою поверхнею та перпендикуляром до поверхні пластини можна знайти половину відстані між точкою відбивання променя 1 та точкою заломлення при виході променя 2 з пластини.

$$c = d \cdot tg\beta, \quad (2)$$

де d – товщина пластини, β – кут заломлення.

Отже, відстань між даними точками буде дорівнювати $2d \cdot tg\beta$. Оскільки кут падіння у нашому випадку завжди фіксований, то кут β буде залежати лише від показника заломлення пластини n .

$$\sin \beta = \frac{\sin \alpha}{n}. \quad (3)$$

Оскільки $\alpha = 45^\circ$, то $\cos \alpha = \sin \alpha$. Відстань b між відбитими променями 1 та 2 рівна:

$$b = \frac{c}{\cos \alpha} = \frac{c}{\sin \alpha}. \quad (4)$$

Підставивши вирази (2) та (3) в (4) отримаємо:

$$b = \frac{d \cdot tg\beta}{\sin \alpha} = \frac{d \cdot \sin \beta}{\sin \alpha \cos \beta} = \frac{d \cdot \sin \alpha}{n \cdot \sin \alpha \cos \beta} = \frac{d}{n \cdot \cos \beta} \quad (5)$$

Використавши тригонометричні формули перетворення перейдемо у правій частині рівності (5) від $\cos\beta$ до $\sin\beta$.

$$b = \frac{d}{n \cdot \sqrt{1 - \sin^2 \beta}} \quad (6)$$

Враховавши (3) отримаємо залежність відстані b від показника заломлення та товщини пластини:

$$b = \frac{d}{n \cdot \sqrt{1 - \frac{\sin^2 \alpha}{n^2}}} \quad (7)$$

Кут падіння променів на пластину завжди рівний 45° і відповідно $\cos \alpha = \cos 45^\circ = \frac{\sqrt{2}}{2}$. Тоді вираз (7) набуде вигляду:

$$b = \frac{d}{n \cdot \sqrt{1 - \frac{1}{2n^2}}} \quad (8)$$

Взявши до уваги вище наведені вирази, отримаємо умову видимості одиничного голографічного зображення:

$$\frac{b}{l} = \frac{d}{l \cdot n \cdot \sqrt{1 - \frac{1}{2n^2}}} \leq 3 \cdot 10^{-4}, \quad (9)$$

або

$$\frac{l}{d} \geq \frac{1}{3 \cdot 10^{-4} \cdot \sqrt{1 - \frac{1}{2n^2}}}. \quad (10)$$

На основі виразу (8) нами було розраховано максимальні товщини пластин, при яких зображення буде одинарним, для матеріалів із різними показниками заломлення та для різних відстаней до зображення. Результати розрахунків представлені у таблиці 1.

Таблиця 1

Максимальна товщина пластини для отримання одиничного зображення d , м

Матеріал	Показник заломлення n	Відстань від спостерігача до піраміди l , м			
		1	3	5	10
Скло звичайне	1,40	$2.543 \cdot 10^{-4}$	$7.63 \cdot 10^{-4}$	$1.272 \cdot 10^{-3}$	$2.543 \cdot 10^{-3}$
Органічне скло	1,53	$2.806 \cdot 10^{-4}$	$8.419 \cdot 10^{-4}$	$1.403 \cdot 10^{-3}$	$2.806 \cdot 10^{-3}$
Скло оптичне	2,04	$2.915 \cdot 10^{-4}$	$8.746 \cdot 10^{-4}$	$1.458 \cdot 10^{-3}$	$2.915 \cdot 10^{-3}$

У ході дослідження було виготовлено ряд моделей прозорих тригранних та чотиригранних пірамід (рис. 4) та створено програмне забезпечення для автоматизованого розрахунку параметрів піраміди (рис.5) і генерування відеоконтейнера із синхронними потоками зображень, що обмежені конкретними розмірами піраміди. Для створення програм було використано мови програмування C# та Python (з бібліотекою `moviepy`), сирцевий код доступний за посиланнями:

1. <https://gist.github.com/gurland/af2f21e11c0935445bc662b7db3c50c4>
2. <https://gist.github.com/gurland/65914ca863d1886ab01faa820f57e089>

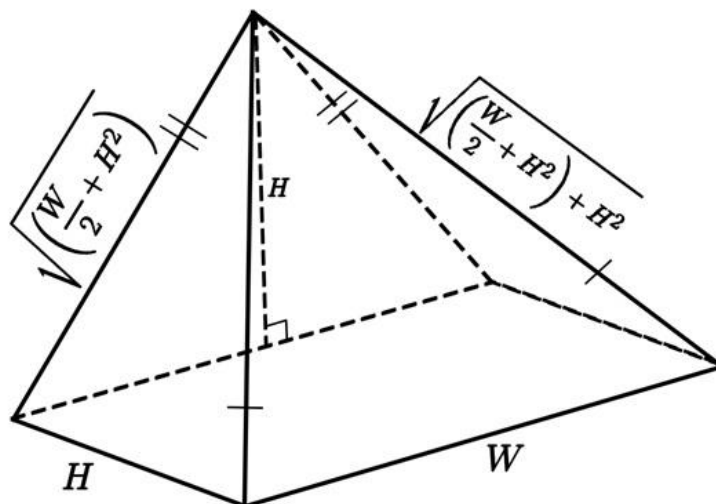


Рис. 4. Геометричні параметри піраміди

В ході досліджень, було встановлено, що основну грань тристоронньої піраміди слід розмістити під кутом 45° до площини проєціюючого монітору. Для покращення ефекту об'ємності зображення, замість відсутньої задньої грані необхідно встановити темний матовий екран, для того щоб всередину піраміди не потрапляло спотворююче псевдоголограму зовнішнє світло. Для збільшення яскравості проєційованого зображення рекомендовано встановити у верхній грані піраміди, під монітором, джерело радіального освітлення.

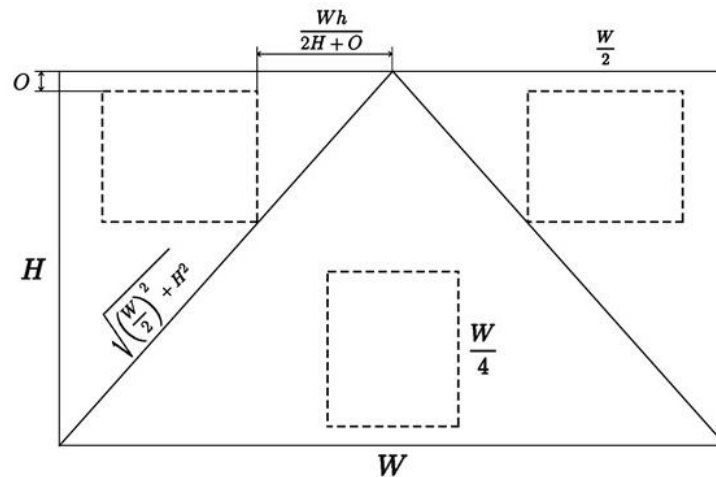


Рис. 5. Розгортка піраміди

Принцип створення псевдоголографічного зображення базується на дзеркальному відбитті світла від рідкокристалічного дисплею гранями зрізаної світлопрозорої піраміди. Зображення на дисплеї розбите на 4 (3) однакових відеокліпи, з динамічними стереозображенням об'єкту на чорному фоні, які розміщені симетрично відносно центру дисплея на певній відстані, де кожен фрагмент повернений на 90° за годинниковою стрілкою відносно сусіднього.

Для рендерингу відеоголограм для різних розширень екрану було написано наступний код, з використання бібліотеки для редагування відео MoviePy:

```
from moviepy.editor import *
def create_hologram(video_name, size=(1280, 720), center_offset=100):
    def resize_clip(clip):
        width, height = clip.size
        if width >= height:
            new_width = int(center_offset * 2)
            new_height = int(height / (width/new_width))
        else:
            new_height = int(center_offset*2)
            new_width = int(width / (height/new_height))
        return clip.resize((new_width, new_height))
    def setpos(clip, center, side, quarter):
        width, height = clip.size
        positions = [(center - center_offset, (side / 2) + center_offset),
                    (center - center_offset - width, (side / 2) - center_offset),
                    (center - center_offset, (side / 2) - center_offset - height),
                    (center + center_offset, (side / 2) - center_offset)]
        return clip.set_position(positions[quarter-1])
    clip = VideoFileClip(video_name)
    background_clip = ColorClip(size, (0, 0, 0), duration=clip.duration)
    side = min(size)
    offset = (max(size) - min(size))/2
    center = offset + (side/2)
    clips = [
        setpos(resize_clip(clip.rotate(180)), center, side, 1),
        setpos(resize_clip(clip.rotate(90)), center, side, 2),
        setpos(resize_clip(clip.rotate(0)), center, side, 3),
        setpos(resize_clip(clip.rotate(-90)), center, side, 4),
    ]
    video = CompositeVideoClip([background_clip, *clips])
    video.write_videofile("output.mp4")
```

Використання даної програми спростило процес моделювання вихідного файлу зображення, автоматизувавши його.

Якщо в якості матеріалу для піраміди використати плівку зворотного проєкціювання, стає можливим досягнення максимального ефекту спостереження об'ємного зображення уявного 3D об'єкта, що зумовлено посиленням антиблікового ефекту, та підвищенні здатності розсіювання променів світла під широким кутом до спостерігача. На даний час, відомо про два типи плівок для зворотного проєкціювання: плівка з мікролінзами та розсіююча матова плівка. Перший варіант є високовартісним. Він ґрунтується на принципі тришарового фільтрування променів світла, завдяки віддзеркаленню

мікролінзами паразитної засвітки, та пропусканні світла через абсорбуючий шар у розсіюючий, завдяки чому досягнути максимально широкого кута поширення світлового потоку. Саме така плівка і дозволяє досягнути найвищої яскравості та контрастності зображення. Другим типом є матова розсіююча плівка, проте вона здатна розсіювати як паразитну засвітку, так і корисне світло, а отже, вимагає виготовлення темної та закритої зверху конструкції піраміди, та використання монітору з найбільшим значенням яскравості (кандел на квадратний метр).

Висновки та перспективи подальшого дослідження. На основі вищевикладеного ми прийшли до висновку, що використання різних матеріалів для виготовлення проєкціуючої призми можна досягнути отримання 3D голографічних зображень високої якості. Дану розробку можна використовувати з освітньою, рекламною, промоційною метою. Заплановано продовжити дослідження властивостей та характеристик даної установки, з метою оптимізації параметрів та мінімізації затрат на виготовлення.

Список бібліографічного опису

1. Бурчак І.Н. Використання 3d голографічних зображень у навчальному процесі / І.Н. Бурчак, В.Л. Величко, І.П. Головачук. // Сучасна наука та освіта Волині : зб. матеріалів наук.-практ. конф. , 22 листоп. 2018 р. , м. Володимир-Волинський. – 2018. – С. 308–310.
2. Головачук І. П. Розробка конструкції установки для демонстрування 3d голографічних зображень / І.П. Головачук, В.Л. Величко. // Науковий журнал "Комп'ютерно-інтегровані технології: освіта, наука, виробництво". – 2018. – №33. – С. 51–56.
3. Міхеєнко Л.А. Геометричні вимірювання. Навчальний посібник з дисципліни "Оптичні вимірювання", частина II: [по напрямку підготовки "Оптико-електронне приладобудування"] /Л.А. Міхеєнко. – Київ. – 2011. – 64 ст.
4. Jaime Moreno. 3D-Holograms in Real Time for Representing Virtual Scenarios / Jaime Moreno, Oswaldo Morales Matamoros, Ricardo Tejeida Padilla, Juan Pablo Francisco Posadas Durán // Polish River Basins and Lakes. – Part II. – 2020. – P. 284-302.
5. Minh H. N. Than, Pyramid Hologram in Projecting Medical Images / Minh H. N. Than, Minh M. N. Pham, Pham Hien. // 7th International Conference on the Development of Biomedical Engineering in Vietnam (BME7). – 2020. – P. 421-426.
6. Takashi Nishitsuji. / Review of Fast Calculation Techniques for Computer-Generated Holograms With the Point-Light-Source-Based Model / Takashi Nishitsuji, Tomoyoshi Shimobaba, Takashi Kakue. // Article in IEEE Transactions on Industrial Informatics PP(99):1-1 · February 2017.

References

1. Burchak Igor. Use of 3d holographic images in the educational process / I. Burchak, V. Velychko, I. Holovachuk. // Modern science and education of Volyn: collection. materials of scientific practice. conf. , 22 November. 2018, Volodymyr-Volynskiy. - 2018. - P. 308–310.
2. Holovachuk Igor. Development of installation design for demonstration of 3d holographic images / I.Holovachuk. V.Velychko. // Scientific journal "Computer-integrated technologies: education, science, production". - 2018. - №33. - P. 51–56.
3. Jaime Moreno. 3D-Holograms in Real Time for Representing Virtual Scenarios / Jaime Moreno, Oswaldo Morales Matamoros, Ricardo Tejeida Padilla, Juan Pablo Francisco Posadas Durán // Polish River Basins and Lakes. – Part II. – 2020. – P. 284-302.
4. Minh H. N. Than, Pyramid Hologram in Projecting Medical Images / Minh H. N. Than, Minh M. N. Pham, Pham Hien. // 7th International Conference on the Development of Biomedical Engineering in Vietnam (BME7). – 2020. – P. 421-426.
5. Takashi Nishitsuji. / Review of Fast Calculation Techniques for Computer-Generated Holograms With the Point-Light-Source-Based Model / Takashi Nishitsuji, Tomoyoshi Shimobaba, Takashi Kakue. // Article in IEEE Transactions on Industrial Informatics PP(99):1-1 · February 2017.

Стаття надійшла 08.05.2020 р.

Рецензенти: Ройко Олександр Юрійович, кандидат технічних наук, голова циклової випускової комісії інформаційних технологій Волинського коледжу НУХТ;

Бурчак Ігор Несторович, кандидат технічних наук, професор кафедри інженерної та комп'ютерної графіки Луцького НТУ.

DOI: 10.36910/6775-2524-0560-2020-39-03

УДК: 514.181.6 + 514.182

Журило Алла Григорівна, к.т.н., доцент,

<https://orcid.org/0000-0003-4084-4622>

Сівак Єлизавета Михайлівна, к.т.н., доцент.

<https://orcid.org/0000-0002-5526-8544>

Національний технічний університет «Харківський політехнічний інститут»

ПОБУДОВА ПЕРСПЕКТИВИ КОЛА, ЩО НАЛЕЖИТЬ ПРЕДМЕТНІЙ ПЛОЩИНІ

Журило А. Г., Сівак Є. М. Побудова перспективи кола, що належить предметній площині. У статті розглянуто основні положення щодо побудови перспективних зображень кіл, які належать предметній площині. Пояснюються складності таких побудов, пропонуються методи спрощення побудов, визначаються необхідні умови для таких побудов. Показано практичне застосування перспективних зображень, встановлена неможливість зображення кіл малих діаметрів в перспективних зображеннях. Тим самим, пояснюються причини можливої заміни еліпсів при виконанні перспективних зображень овалами без втрати наочності. Наведено приклади перспективних зображень кіл при виконанні креслеників.

Ключові слова: перспективні зображення, коло, еліпс, овал, практичне застосування.

Журило А. Г., Сивак Е. М. Построение перспективы окружности, принадлежащей предметной плоскости. В статье рассмотрены основные положения по построению перспективных изображений окружностей, принадлежащих предметной плоскости. Объясняются сложности такого построения, предлагаются пути упрощения построений, определяются необходимые условия для таких построений. Показано практическое применение перспективных изображений, установлена невозможность изображения окружностей малых диаметров в перспективных изображениях. Тем самым, объясняются причины возможной замены эллипсов при выполнении перспективных изображений овалами без потери наглядности. Приведены примеры перспективных изображений окружностей при выполнении чертежей.

Ключевые слова: перспективные изображения, окружность, эллипс, овал, практическое применение.

Zhurilo A. G., Sivak E. M. Building the perspective of a circle that belongs to the subject plane. The article discusses the main provisions for the construction of perspective images of circles belonging to the subject plane. The complexity of such a construction is explained, ways to simplify the construction are suggested, and the necessary conditions for such constructions are determined. The practical application of perspective images is shown. It is established that it is impossible to image circles of small diameters in perspective images. This explains why ellipses can be replaced with ovals when performing perspective images without losing visibility. Examples of perspective images of circles when performing drawings are given.

Key words: perspective images, circle, ellipse, oval, practical use.

Постановка проблеми. Незважаючи на широкий розвиток комп'ютерної техніки та широке застосування її для виконання креслеників, появи вже декількох поколінь програм КОМПАС, AUTOCAD та їхніх аналогів, перспективні проекції широко використовуються у машинобудуванні та архітектурі. Для їх опанування потрібно добре знати їхні властивості та правила їхньої побудови, раціонального розташування, з метою отримання найбільш вигідного зображення, та зменшення кількості різних побудов, тощо. На жаль, перспективні проекції мають відповідні лише їм специфічні властивості, які ускладнюють побудову креслеників та їхніх частин, та, відповідно, використання вказаних проекцій на практиці.

Аналіз останніх досліджень і публікацій. Питання щодо точних графічних побудов має велику історичну давнину, беручи свій початок ще в роботах Архімеда, Евкліда та інших вчених. З вичерпною повнотою і строгою науковою обґрунтованістю теорія точних метричних побудов була розроблена математиком Гаспаром Монжем, який у 1795 – 1799 рр. опублікував результати своєї двадцятирічної роботи під назвою «Нарисна геометрія» [1].

Серед імен, з якими пов'язаний розвиток наукової праці в області перспективних проекцій, можна згадати видатних вітчизняних вчених: Н. М. Бескіна, О. О. Вольберга, Н. О. Глаголева, Є. А. Глазунова, А. І. Добрякова, Д. І. Каргіна, І. І. Котова, М. О. Риніна, С. О. Смирнова, М. Ф. Четверухіна, С. О. Соловійова [2-6].

У даний час теорія перспективи розроблена докладно і висвітлена в численних працях з нарисної геометрії. Питання ж практики побудови перспективних зображень висвітлені в літературі недостатньо. У практиці побудови перспективних зображень часто виникають значні труднощі, обумовлені не тільки недостатньою підготовкою виконавця, але і складністю окремих задач, що вимагають спеціального роз'яснення [6-8].

Положення ускладнюється ще й тому, що за останні 20..30 років практично не публікувалося дослідних робіт із практики побудови перспективних зображень та її основних законів. Ті ж роботи, що були опубліковані раніше, у більшості випадків розглядають найбільш відомі методи

перспективних зображень, та розглядають найбільш нескладні задачі.

Невирішені частини проблеми. Метою статті є визначення причин широкого розповсюдження перспективних зображень, обґрунтування основних властивостей при побудові тіл обертання.

Мета дослідження.

Перспективні зображення є найважливішим розділом побудови об'ємних зображень. Перспектива підпорядковується законам і правилам, за якими можна зображати предмети так, як вони представляються нашому оку в просторі. Відомі побудови перспективних зображень методом архітекторів, перспективної сітки, лінійна, панорамна перспектива та інші її види. Спосіб архітекторів можна вважати різновидом способу сліду променя, коли перспективу не поєднують з фронтальною проекцією, а переносять на вільне поле кресленика, або на інший аркуш, при цьому масштабуючи зображення. Як впливає з назви, спосіб широко застосовується для зображення призм, кубів, паралелепіпедів, як складових будівель і споруд. На практиці широко застосовують методи архітекторів з однією точкою сходу, з двома, з опущеним (піднятим) планом і з бічною стіною. При побудові окремої будівлі площину картини зазвичай проводять під кутом $30...35^\circ$ до фасаду. Для полегшення побудови бажано провести картину через будь-який кут споруди.

Головний промінь повинен бути, перпендикулярним картині і бути бісектрисою кута зору або мати положення близьке до бісектриси. Кут зору приймається в межах $30 \dots 35^\circ$, але не більше 40° . Для об'єктів, що мають розвиток в горизонтальній площині, точка зору обирається по куту в плані, а для об'єктів, що мають розвиток у вертикальній площині – по куту у вертикальній площині. У будь-якому випадку головна точка картини повинна знаходитися в середній третині зображення перспективи. Від цього буде залежати наочність зображення, що наведено на Рис. 1.

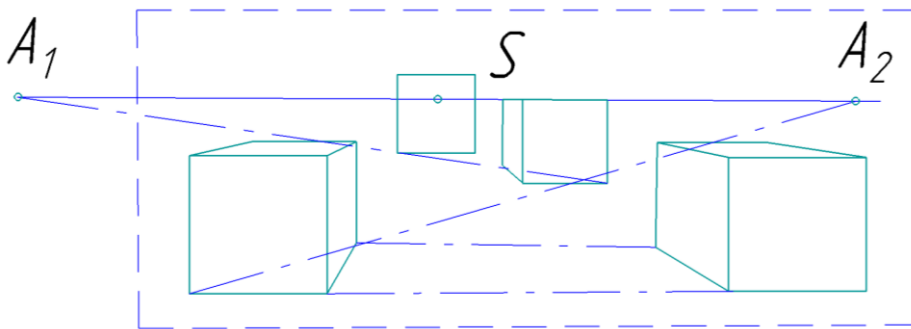


Рис. 1. Наочність зображення кубів на перспективному зображенні в залежності від положення щодо горизонту і точки зору.

При побудові перспективних креслеників досить часто доводиться викреслювати перспективу кола не тільки як перспективу існуючої кривої на даному предметі, але і як допоміжну криву для вирішення деяких геометричних завдань.

Перспективою на площині для кола є еліпс, який зображується допоміжними побудовами ряду точок, які йому належать. Ці точки можливо визначити декількома способами:

- з побудови перспектив ряду точок, обраних на даній окружності;
- з побудови еліпса за двома сполученими діаметрами, з подальшим визначенням декількох точок, що йому належать;
- за визначенням ряду точок зазначеного еліпса.

Що ж стосується ряду точок, які належать еліпсу, то вони повинні бути визначені на такій відстані одна від одної, щоб з'єднуючи кожен суміжну пару таких точок дуги еліпса могли б бути прийняті без значної похибки, за прямолінійні відрізки або ж за дуги кіл яких би то не було радіусів. Так як дуги еліпса в різних частинах кожної його чверті не мають однакової кривизни, то, отже, і відстані між рядом точок, що визначають обрис будь-якої чверті еліпса, будуть не однакові: при вершині, яка визначається великою віссю, вони повинні бути обрані ближче одна до іншої, а в міру наближення до вершини, яка визначається малою віссю, відстані між ними можна послідовно збільшувати. Крім того відстані між рядом точок, що визначають досить точно накреслення чверті еліпса, змінюються в залежності відносини між його півсями. Так як одна і та ж окружність (з незмінним радіусом) може перетворюватися в перспективі в еліпси з різними відносинами між півсями, в залежності від того, наскільки вона буде віддалена як від полюса, так і від центральної площини. Зазначене положення виключає виділення єдиного правила поділу даної окружності на такі дуги, який дозволяли б побудови в перспективі необхідної величини дуги еліпса перспективи

даної окружності. Тому, найбільш раціональний спосіб побудови перспективи кола (еліпса) буде той, який дає можливість визначати на картинній площині точку, що належить еліпсу (перспективі даної окружності) в заданому місці.

Найбільш поширеним способом побудови кола в перспективі є спосіб описаного квадрата. Цей спосіб передбачає, що спочатку будується перспектива квадрата, в який потім вписується еліпс. Як приклад, можна розглянути процес побудови перспективи горизонтально розташованої окружності, яка вписана в квадрат зі сторонами паралельними картинній площині (Рис. 2).

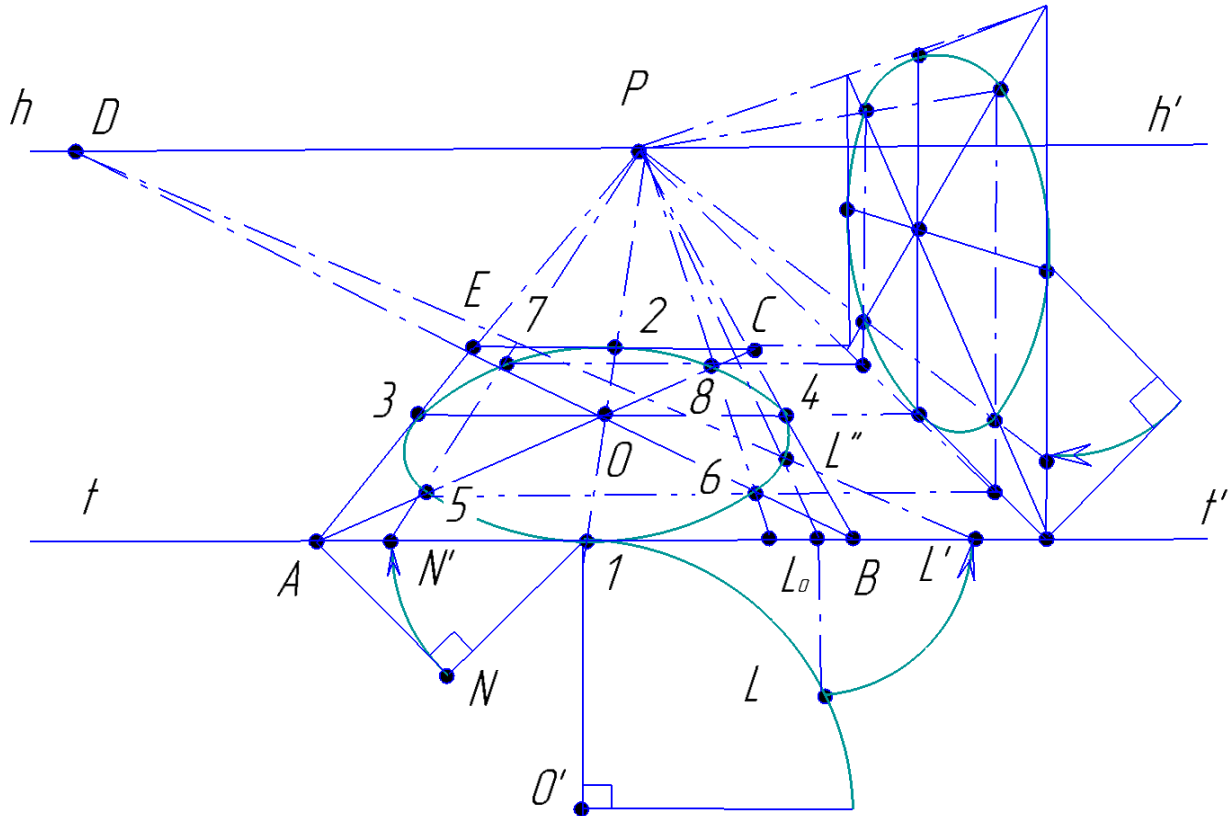


Рис. 2 – Побудова кола способом описаного квадрата у фронтальній перспективі

Побудова квадрата ведеться наступним чином. Намічається найближча до глядача сторона квадрата AB , з вершин якої проводяться прямі в головну точку P . Сторони AB і BC , що йдуть в глибину сторони квадрата будуються за допомогою дистанційної точки D згідно з правилами побудови масштабу глибини. На перетині діагоналей квадрата буде розташовуватися точка O - центр кола. Через точку O проводяться горизонтальна пряма і пряма OP . Ці прямі, будучи осями симетрії квадрата і кола, перетнуть сторони квадрата в точках 1, 2, 3 і 4. Дані точки позначають місця торкання еліпса до сторін квадрата.

На діагоналях квадрата можна побудувати, щонайменш, ще чотири точки еліпса. Так, щоб побудувати точки 5 і 7, викреслюється рівнобедрений прямокутний трикутник $AN1$ і відкладається відрізок $N'1$, що дорівнює меншій стороні трикутника. Далі проводиться пряма $N'P$, що перетинає діагоналі трикутника в шуканих точках 5 і 7. Щоб побудувати ще дві точки еліпса, проводяться горизонтальні прямі через точки 5 і 7. Місця перетину цих прямих з діагоналями квадрата позначають розташування даних точок (вони позначені цифрами 6 і 8). На завершальному етапі побудови точки еліпса від 1 до 8 з'єднуються плавними кривими лініями.

Іноді, для того щоб домогтися більшої точності побудови, виникає необхідність знайти перспективне розташування додаткових точок кола. Для цього в нижній частині зображення можна викреслити фрагмент кола у фронтальній площині з радіусом, рівним половині сторони квадрата $ABCE$. В даному випадку це дуга з центром в точці O' , що стосується боку квадрата в точці 1. На дузі відкладається довільно взята точка L . з цієї точки проводиться перпендикуляр до сторони квадрата, що перетинає її в точці L_0 . Далі на прямій AB відкладається відрізок L_0-1' , що дорівнює відрізку L_0-L . Потім точка $1'$ з'єднується з дистанційною точкою D . у місці перетину прямих L_0-P і $1'-D$ буде розташовуватися точка L'' , яка є однією з точок еліпса. Аналогічним чином можна побудувати ряд інших точок кола в перспективі.

Дуже важно визначити, що подібний спосіб знаходження додаткових точок кола придатний лише при

побудові великих еліпсів. Побудова невеликих кіл в перспективі таким способом не має сенсу, оскільки в цьому випадку неминучі графічні похибки призводять до значного спотворення форми.

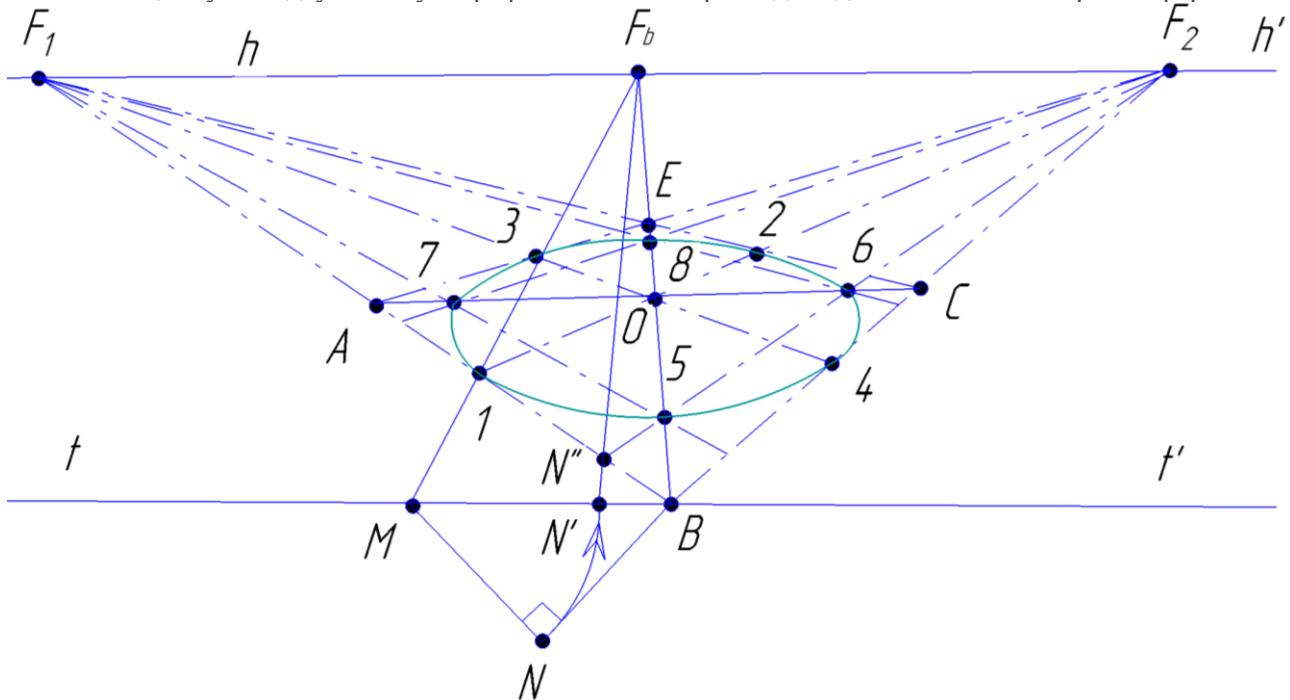


Рис. 3 - Побудова перспективи кола способом описаного квадрата в кутовий перспективі

На Рис. 3 показаний приклад побудови кола, вписаної в квадрат $ABCD$, розташований в кутовий перспективі і стосується підстави картини (квадрат може бути побудований одним з описаних вище способів побудови перспективи). Спочатку проводяться діагоналі квадрата і в місці їх перетину визначається точка O – центр майбутньої окружності, з якої проводяться прямі до точки сходу F_1 і F_2 . Ці прямі перетнуть сторони квадрата в точках 1, 2, 3 і 4. В даних точках окружність належить сторонам квадрата.

Далі будуються точки кола на діагоналях квадрата. Для визначення місця розташування точок 5 і 6 спочатку з точки F_b , що є точкою сходу прямої BE , проводиться пряма через точку 1 і перетинає основу картинної площини в точці M . Потім будується рівнобедрений прямокутний трикутник MBN і на основі картинної площини відкладається відрізок MN' , що дорівнює MN . Далі проводиться пряма $N'F_b$ і в місці перетину даної прямої зі стороною AB відкладається точка N'' . З цієї точки проводиться пряма в точку сходу F_2 , в місці перетину якої з діагоналями квадрата будуть розташовуватися точки 5 і 6. Якщо через дані точки провести прямі в точку сходу F_1 , то місця перетину цих прямих з діагоналями квадрата будуть позначати положення ще двох точок еліпса, позначених цифрами 7 і 8. Утворені точки від 1 до 8 акуратно з'єднуються кривими лініями.

Легко визначити, що отриманий еліпс є практично ідентичним, наведеному у роботі [9]. Причому, при зображенні еліпса у побудові кола в перспективі наявні те ж проблеми точної побудови, використання лекал, чимала кількість допоміжних побудов. Не дуже полегшує побудову еліпса використання комп'ютерної графіки, бо кількість допоміжних побудов залишається необхідною. Тому, при побудові перспективних зображень з метою значного полегшення графічних побудов, варто замість еліпсу будувати овал [10].

Висновки та перспективи подальшого дослідження. Визначено, яким чином коло зображується у перспективних зображеннях, доведено, що еліпс, яким зображується коло у перспективі, має теж властивості, що і еліпс при побудові аксонометричних проєкцій. Доведено, що при побудові еліпсу, хоч у перспективі, хоч у аксонометрії, основною проблемою залишаються питання точної побудови еліпсу, необхідність використання лекал, потреба виконання чималої кількості допоміжних побудов. Тому, при можливості, пропонується при виконання кресленика перспективних зображень замість побудови еліпсу використовувати овал.

Список бібліографічного опису

1. Гордон В. О. Курс начертательной геометрии: учебник / В. О. Гордон, М. А. Семенцов - Огиевский. – М.: Наука, 1976. – 432 с.

2. Ланюк А. В. Аксонометрические проекции: учебник / А. В. Ланюк. — М. : Гос. изд - во лит - ры по строительству и архитектуре, 1956. — 176 с.
3. Порсин Ю. Я. Аксонометрические изображения машиностроительных деталей: учебник / Ю. Я. Порсин. — М.- Л. : Машгиз, 1973. — 188 с.
4. Соловьев С. А. Перспектива / С. А. Соловьев.- М.: Просвещение, 1981. - 144с.
5. Соловьев С. Д. та інші. Черчение и перспектива / С. А. Соловьев. - М.: Высшая школа, 1982.
6. Журило А. Г. Теоретичні та практичні основи аксонометрії [Текст] / А. Г. Журило. Навч. посібник. Х.: НТУ «ХПІ». — 2010. - 196 с.
7. Петерсон В. Е. Перспектива / В. Е. Петерсон. М.:Просвещение, 1970. — 184 с.
8. Ратничин В. М. Перспектива / В. М. Ратничин. Киев, Вища школа, 1982.- 232 с.
9. Журило А. Г. Деякі питання щодо креслення кіл при побудові аксонометричних проєкцій / А. Г. Журило, Є. М. Сівак // Комп'ютерно - інтегровані технології: освіта, наука, виробництво. — 2017. - №26. - С. 93-98. Видавництво Луцького національного технічного університету.
10. Журило А. Г. Построение перспективных изображений при отображении картинной плоскости / А. Г. Журило, Е. М. Сівак, И. Ю. Адашевская // Вестник НТУ «ХПИ». № 17, 2014. С. 73-79.

References

1. Gordon V. O. Kurs nachertatel'noj geometrii: uchebnik / V. O. Gordon, M. A. Semencov - Ogievskij. — М.: Nauka, 1976. — 432 s.
2. Lanyuk A. V. Aksonometricheskie proekcii: uchebnik / A. V. Lanyuk. — М. : Gos. izd - vo lit - ry po stroitel'stvu i arhitekture, 1956. — 176 s.
3. Porsin Yu. Ya. Aksonometricheskie izobrazheniya mashinostroitel'nyh detalej: uchebnik / Yu. Ya. Porsin. — М.- Л. : Mashgiz, 1973. — 188 s.
4. Solov'ev S. A. Perspektiva / S. A. Solov'ev.- М.: Prosveshchenie, 1981. - 144s.
5. Solov'ev S. A. ta insh. Cherchenie i perspektiva / S. A. Solov'ev. - М.: Vysshaya shkola, 1982.
6. Zhurilo A. G. Teoretichni ta praktichni osnovi aksonometrii [Tekst] / A. G. Zhurilo. Navch. posibnik. H.: NTU «HPI». — 2010. - 196 s.
7. Peterson V. E. Perspektiva / V. E. Peterson. М.:Prosveshchenie, 1970. — 184 s.
8. Ratnichin V. M. Perspektiva / V. M. Ratnichin. Kiev, Vishcha shkola, 1982.- 232 s.
9. Zhurilo A. G. Deyaki pitannya shchodo kreslennya kil pri pobudovi aksonometrichnih proekcij / A. G. Zhurilo, E. M. Sivak // Komp'yuterno - integrovani tekhnologii: osvita, nauka, virobnictvo. — 2017. - №26. - S. 93-98. Vidavnictvo Luc'kogo nacional'nogo tekhnichnogo universitetu.
10. Zhurilo A. G. Postroenie perspektivnyh izobrazhenij pri otobrazhenii kartinnoj ploskosti / A. G. Zhurilo, E. M. Sivak, I. Yu. Adashevskaya // Vestnik NTU «HPI». № 17, 2014. S. 73-79.

Рецензенти:

Заступник начальника кафедри
інженерної та аварійно-рятувальної
техніки Університету цивільного
захисту України, д-р техн. наук, професор Л. М. Куценко

Доцент кафедри Геометричного моделювання
та комп'ютерної графіки НТУ «ХПІ»,
канд. техн. наук

О.А. Глібко

DOI: 10.36910/6775-2524-0560-2020-39-04

УДК: 622 232.8

Каганюк Алексей Казимирович, к.т.н, доцент<https://orcid.org/0000-0003-4616-8768>**Мельник Василий Михайлович**, к.ф.-м.н., доцент<https://orcid.org/0000-0001-8282-6639>

Луцкий национальный технический университет

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ РАСЧЕТА ПАРАМЕТРОВ РЕГУЛЯТОРА ДЛЯ ПОДВИЖНОГО ОБЪЕКТА

Каганюк А.К., Мельник В.М. Математическая модель расчета параметров регулятора для подвижного объекта. Дается краткий анализ возмущающих воздействий для обоснования основных видов возмущающих воздействий, которую необходимо учитывать при расчете параметров регулятора при построении системы автоматического управления угледобывающим комбайном в профиле пласта.

Ключевые слова: Математическая модель. Подвижный объект. Погрешность. Система автоматического управления объектом.

Каганюк О.К., Мельник В.М. Математична модель рухомого об'єкта для розрахунку похибки Дається стислий аналіз впливу збуржень для обґрунтування основного виду похибки, яку необхідно враховувати для будови систем автоматичного управління вугільно видобувним комбайном в профільній товщини вугільного пласта.

Ключеві слова: Математична модель. Рухомий об'єкт. Похибка. Система автоматичного управління об'єктом.

Kaganjuk A.K., Melnyk V.M. A Mathematical model for calculating controller parameters for a moving object. A brief analysis of the disturbing influences is given to justify the main types of disturbing influences, which must be taken into account when calculating the parameters of the regulator when constructing the automatic control system for a coal mining combine in the formation profile.

Key words: Mathematical model. Movable object. Error. The system of automatic control of the object.

Постановка проблемы. В настоящее время на мировом рынке наметилась тенденция по развитию добычи энергетических углеводородов. Кто обладает их запасами, тот может себе гарантировать энергонезависимость и обеспечить экономическую безопасность своего государства. Поэтому увеличивается спрос на горно – шахтное оборудование, технику и технологию для добычи угля. Более того данный спрос имеет тенденцию роста ввиду того, что горнодобывающим предприятиям приходится разрабатывать угольные пласты с горно – геологическими нарушениями, повышенной газообильностью угольных пластов, опасных по внезапным выбросам угля и породы. Все труднее обеспечивать достаточный уровень безопасности ведения горных работ.

Для повышения производительности труда рабочих очистного забоя, которым приходится работать в сложных горно – геологических условиях залегания угольных пластов, необходимо внедрять механизированные комплексы, оснащенные системой автоматического управления. Однако, существующие системы автоматического управления, которые разработаны и по мере необходимости внедряются на угольных шахтах, не в состоянии повысить безопасность работ технического персонала, работающих в непосредственной близости с угледобывающим комбайном. Так как они не в состоянии обеспечить безлюдную выемку угля, и проблема повышения безопасности горнорабочих очистного забоя, остается на прежнем уровне. Для этого необходимо создать локальную систему автоматического управления угледобывающим комбайном в профильной плоскости угольного пласта, далее САУ ПП.

Анализ исследований поставленной проблемы. Ряд авторов трудились над созданием таких систем автоматического управления с использованием различного типа датчиков, как в Украине, так и за рубежом [1, 2, 3, 4, 5, 7] однако, данная проблема, так и не решена.

Автоматическое управления угледобывающими комбайнами в профиле пласта, одна из наиболее важных и актуальных задач технического прогресса угольной промышленности. Неотъемлемой частью этой задачи является создание надежных способов и средств автоматического управления, которые могли бы работать в сложных условиях добычных забоев.

Отсутствие отработанных технических решений в этой области в значительной степени тормозит повышение эффективности использования добычной техники и последующий переход к безлюдной выемке угля, из-за сложности визуального наблюдения техническими средствами за положением рабочих органов угледобывающих комбайнов относительно границы раздела «порода-уголь»

При современных скоростях подачи движения угледобывающих комбайнов, трудно избежать присечки породы или потерь угля. Поэтому, исследование данного вопроса важно как с целью

комплексного рішення задач автоматизації угледобуваючих комбайнів, так і підвищення безпеки ведення добувних робіт в складних горно – геологічних умовах.

Рішенням цих питань займалися багато науково-дослідницькі, проектно-конструкторські і навчальні інститути:

ІГД ім. А.А. Скочинського, ДонНІПІ, НПО Автоматгормаш, Гипроуглемаш, Донгипроуглемаш, ДонУГІ, Московський горний інститут, Київський інститут автоматизації, Карагандинський політехнічний інститут і др.

В останні роки в Україні цим напрямком ніхто практично не займається, оскільки дані дослідження вимагають значительних витрат на проведення дослідницьких витрат, що практично не реально в теперішній час.

Раніше в СРСР і за кордоном неодноразово проводилися шахтні випробування автоматизованих комплексів і горнодобуваючих машин, оснащених системами автоматичного управління в профілі пласта з датчиками, порода-уголь. Однак, досвід цих експериментів і випробувань показав, що успішна робота таких машин залежить не тільки від спроможності комплексної системи автоматичного управління добуваючим комбайном, але і окремих її елементів, де найбільш вразливим ланкою є не тільки датчик порода-уголь але і канал передачі інформації від первинного перетворювача, контролюючого межі розділу «порода – уголь», далі ДПУ, а також від конструктивних особливостей угледобуваючої машини, як об'єктів автоматичного управління [6].

Метою нинішніх досліджень є угледобуваючі комбайни, переміщувані по ленті забойного конвеєра, як найбільш поширені з успішної компоновочною схемою розташування режущих органів.

Угледобуваючий комбайн, який може бути оснащений системою автоматичного управління в профілі пласта, в процесі роботи, постійно знаходиться під впливом різних випадкових збурюючих факторів [8].

Всі збурюючі впливи, прикладені до системи автоматичного управління, можна розділити на управляючі і збурюючі. Під управляючими впливами, які можна віднести як входні, розуміються зміни рельєфу кровлі або ґрунту пласта, які система автоматичного управління повинна відслідковувати і формувати управляючий вплив для режущих органів.

Ціль роботи провести аналіз випадкових збурюючих впливів, які виникають в процесі руху рухомого об'єкта. Ранжувати і виявити основні випадкові збурюючі впливи, які необхідно враховувати при розрахунку похибки в процесі управління рухомим об'єктом.

Виклад основного матеріалу при дослідженні даного питання.

Зміни рельєфу пласта носить випадковий характер, т.е. на будь-якому відрізку шляху вони можуть бути різними за знаком, амплітуді і формі [10]. При цьому слід розрізняти зміни, які відбуваються на ділянках довжини, співмірної з розмірами угледобуваючого комбайна, і ті, які відбуваються на ділянках більшої довжини. Перші, в спеціальній літературі називаються мікрорельєфом, другі – макрорельєфом.

Вибірочні статистичні дослідження мікрорельєфу [66] показали, що профілі пластів на ділянках обмеженої довжини можна вважати стаціонарними, нормально розподіленими випадковими функціями з кореляційними залежностями виду:

$$K_h(l) = D_h(l) e^{-\alpha(l)} \cos \beta l, \quad (1)$$

$$K_h(l) = D_h(l) e^{-\alpha(l)} \quad (2)$$

де β і α – коефіцієнти, що визначають затухання кореляційної функції.

Кореляційні функції ряду реалізацій вказують на те, що профілі пластів містять в собі періодичні складові з періодом $\Delta L = \frac{2\pi}{\beta}$,

Величина ΔL коливається в межах від 7.25 м до 4.8 м. При цьому максимальне середнькватратичне відхилення амплітуд B_A профіля для ґрунту пласта становить від 23 мм до 28 мм, а для кровлі пласта від 34 мм до 42 мм. Такі періодичні складові відповідають максимуму на залежності спектра дисперсій [10].

На основі цих даних, для описання управляючого (входного) впливу $h_0(t)$ нами використана залежність виду:

$$h_0(t) = A \sin(\omega t), \quad (3)$$

де A – амплітуда, $A = 2 B_A$;

ω – частота которая будет определяться следующим образом:

$$T = \frac{\Delta L}{V_k}, \quad (4)$$

где V_k – скорость подачи комбайна, тогда

$$\omega = \frac{2\pi}{T} = \frac{2\pi}{\Delta L} V_k \quad (5)$$

Уравнение (3) используется при моделировании входного воздействия.

К возмущающим воздействиям относятся все те, которые прямо или косвенно приводят к отклонению комбайна или его выемочных органов от требуемого положения [10].

Те возмущающие воздействия, которые непосредственно поступают на вход системы управления, являются помехами. К помехам можно относиться, например, воздействия, связанные с физикой работы чувствительных элементов датчика.

Поскольку при проведении исследований, мы будем руководствоваться при выборе только радиоизотопным датчиком, как наиболее перспективным в условия создания систем по безлюдной выемке угля, то и в расчетах по определению погрешности, будем учитывать его характерные особенности.

Для радиоизотопного датчика к помехам относятся изменения плотности и зольности угля и вмещающих пород, флуктуация сигнала, колебания воздушного зазора между датчиком и контролируемой поверхностью и т.д. К помехам, воздействующим на анализируемый сигнал, относится также флуктуация сигнала датчика выдвиги штока, далее ДВШ, которые обусловлены вибрацией РО и самого поворотного редуктора.

В частности, уровень помех обусловлен не только чувствительностью ДПУ воздействием того или иного возмущающего фактора, но и дисперсией его значений.

Нет необходимости приводить вероятностное описание помехи. Необходимо лишь учесть ее действие при анализе работы системы управления в процессе ее математического моделирования.

Возмущающие воздействия, приложенные к объекту управления и влияющие непосредственно на траекторию РО, имеют различную природу и связаны с внутренними и внешними причинами, а именно, с динамикой работы угледобывающей машины [11].

Под внутренней динамикой подразумевается результат совокупного действия сил, обусловленных работой привода, неуравновешенностью вращающихся масс и влиянием процесса резания угля на кинематическую цепочку машины.

Внутренние возмущения порождают высокочастотные колебания (вибрации), результирующие действия которых, зависят от продолжительности их приложения и постоянных времени элементов системы. Для «фильтрации» этих возмущений необходимо, чтобы постоянные времени системы были больше времени их приложения. В этом случае вибрация будет вызывать незначительную погрешность в работе системы управления.

Внешняя динамика машин определяется величиной и направлением равнодействующей всех сил, приложенных к объекту управления в процессе его работы. Внешние возмущающие воздействия можно разделить на непрерывные и импульсные.

Непрерывно действующие возмущения, обуславливаются движением машины по изогнутому в продольном и скрученному в поперечном направлениях ставу конвейера, по неровностям подготовленной в предыдущих циклах машинной дороги, подштыбровкой конвейера или опор машины. Действие этих возмущений передается через опоры машины на ее режущий орган, изменяя его положение в пространстве.

Реакция забоя на режущий орган угледобывающей машины, представляет собой случайный процесс, параметры которого зависят от сопротивляемости угля резанию, типа режущего органа, жесткости опор и т.д. Эти непрерывные возмущения приложены непосредственно к органу машины [12].

Импульсные возмущающие воздействия обусловлены влиянием стыков между секциями конвейера, если последние смещены относительно друг друга. Частичным или полным стопорением режущего органа при большой толщине стружки, что часто вызывает потерю устойчивости угледобывающего комбайна [12].

Для построения системы автоматического управления угледобывающими комбайнами в профиле пласта с последующим контролем толщины угольной пачки, как мы говорили ранее, целесообразно использовать радиоизотопный датчик «порода – уголь». Поскольку, для работы в условиях безлюдной выемки угля и горно – геологических нарушений, целесообразно использовать датчики, которые способны накапливать текущую информацию о контролируемой толщине угольной пачки на определенных участках длины. для выработки управляющего воздействия режущим органам

угледобывающего комбайна перемещающихся со скоростью в диапазоне от 1м/м. до 10 м/м., Под такую категорию датчиков, которые прошли первичную экспериментальную апробацию, в реальных условиях работы угледобывающих комбайнов, можно отнести радиоизотопные датчики «порода – уголь». Данная категория радиоизотопных датчиков, позволяет накапливать информацию о контролируемой толщине предохранительной угольной пачки на участках длиной от 0,2 м до 0,6 м., что позволяет формировать благоприятные площадки, для последующей передвижки забойного конвейера на новую «машинную дорогу». Формирование приемлемых площадок вышеуказанной длины, являются рациональными с точки зрения выполнения маневра для современных угледобывающих комбайнов.

Важнейшим показателем радиоизотопного ДПУ, как первичного преобразователя, является его выходная характеристика [13, 14], т.е. функциональная зависимость между толщиной измеряемой угольной пачки (входной величиной) и значением сигнала на выходе ДПУ, заданная аналитически или в виде графа.

Характерной особенностью радиоизотопных датчиков является то, что они имеют определенную разрешающую способность для контроля толщины угольной пачки, которая ограничивается максимальной чувствительностью в диапазоне от нулевого значения до 50мм. Выходная характеристика ДПУ показана на рис.1.

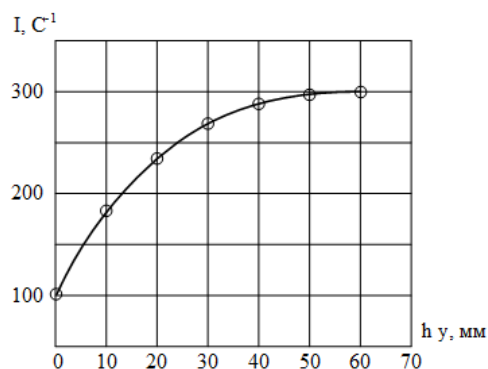


Рис. 1. Выходная характеристика радиоизотопного датчика «порода – уголь»

Выходным сигналом для формирования управляющего воздействия, формируемого регулятором, рекомендованного ранние радиоизотопного ДПУ, является величина регистрируемой скорости счета, зависящая от толщины контролируемой угольной пачки, описываемая уравнением вида [13];

$$I = A_0 + A_1 * h + A_2 * h^2, \quad (6)$$

где A_0, A_1 и A_2 постоянные коэффициенты статической характеристики ДПУ.

В зависимости от типа и соответствующего расположения датчика «порода – уголь», на подвижном объекте, регулятор, который формирует управляющий сигнал исполнительному механизму, могут быть использованы различные алгоритмы по обработке информации, поступающей от ДПУ.

В литературных источниках известны различные методы поиска алгоритмов функционирования регулятора такие, как например, прогнозирование текущего значения сигнала (интерполяция), оценки скорости входного сигнала и т.п. Однако данный метод, учитывающий случайный характер рельефа для входного сигнала с ДПУ, который имеет ограниченные временные параметры анализа и сложность его реализации, не могут конкурировать по надежности и достоверности с методом оценки математического ожидания, путем надежного метода, как интегрирование входного сигнала. Как наиболее простого, с точки зрения усредненного значения контролируемой толщины предохранительной угольной пачки, для формирования управляющего воздействия

Поскольку мы будем ориентироваться на перспективный типоряд угледобывающих комбайнов, перемещающихся по конвейерному ставу и имеющих режущий орган выполненного в виде шнека, то наиболее рациональным, на наш взгляд, является расположить ДПУ непосредственно на режущем органе комбайна. Это позволяет исключить транспортное запаздывание анализируемой толщины предохранительной угольной пачки при передаче информации, поступающей с датчика. А поскольку режущий орган при разрушении горного массива описывает окружность, то и ДПУ, расположенного на этом режущем органе, будет перемещаться по сходной траектории. Измерения контролируемой предохранительной угольной пачки, будут осуществляться в определенные моменты времени, в

ограниченном секторе описываемой окружности. Модель измерения контролируемой толщины угольной пачки, представлена на рис 2.

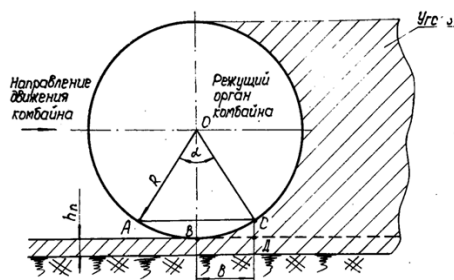


Рис.2. Модель измерения контролируемой толщины угольной пачки, при размещении ДПУ на режущем органе комбайна.

Величина зоны измерения ДПУ это дуга AC, которая определяется параметрами шнека, диаметром шнека, центральным углом α , скорость подачи комбайна и выражается в виде зависимости:

$$t_{\text{изм}} = 2 * \frac{R_{\text{ш}} \sin \alpha}{V_{\text{пр}}}, \quad (7)$$

При скорости вращения шнека ≈ 1 об/с и значение $\alpha=30^\circ$, который ограничивается максимальной чувствительности датчика, имеет минимально возможное время измерения (за один оборот) составляет $\approx 0,1$ с.

Необходимое время, в течение которого ДПУ совершает полный оборот вокруг оси шнека (цикл измерения), обозначим через $t_{\text{ц}}$. Поскольку часть этого времени не используется для получения информации от ДПУ, время $t_{\text{ц}}$ складывается из собственного времени измерения $t_{\text{изм}}$ и времени паузы $t_{\text{п}}$, т.е.

$$t_{\text{ц}} = t_{\text{изм}} + t_{\text{п}}. \quad (8)$$

Очевидно, что за время $t_{\text{ц}}$ датчик, вращаясь и двигаясь поступательно вместе с комбайном, пройдет путь $l_{\text{ц}}$, который также состоит из «пути измерения» $l_{\text{изм}}$ и «пути паузы» $l_{\text{п}}$.

В зависимости от типоразмера режущего органа и скорости подачи комбайна возможны три варианта взаиморасположения зон измерения (рис. 3.):

1. Зоны следуют одна за другой.
2. Зоны отделены друг от друга (по времени и по расстоянию).
3. Зоны частично накладываются друг на друга.

Все три ситуации можно учесть в одном уравнении имеющим вид:

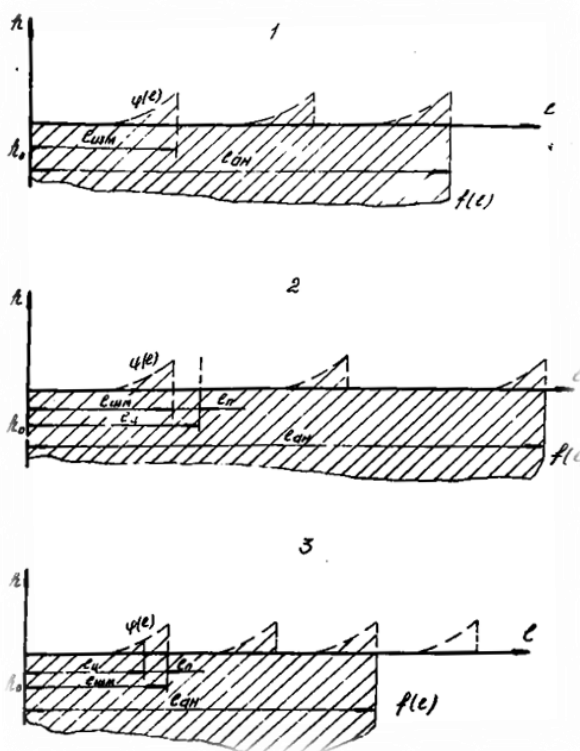


Рис.3. Варианты взаиморасположения зон измерения

$$h_{\text{вых.р}} = \frac{1}{n * t_{\text{изм}}} * \sum_{i=0}^n \int_{t_i}^{t_i+t_{\text{изм}}} h_{\text{вх.р}}(t) dt, \quad (9)$$

где t_i - время начала интегрирования;

$t_{\text{п}}$ - время паузы между окончанием предыдущего и началом последующей зоны измерения;

$h_{\text{вх.р}}$ - входной сигнал регулятора определяемый сигналом с ДПУ;

$h_{\text{вых.р}}$ - математическое ожидание сигнала датчика, оцененное за время t_a ;

n - число циклов интегрирования (анализа).

$$n = \frac{t_a}{t_{\text{изм}}+t_{\text{п}}}, \quad (10)$$

t_a - время анализа сигнала.

Учитывая то обстоятельство, что входной сигнал регулятора определяется сигналом ДПУ, выражение для определения сигнала ДПУ записывается в следующем виде:

$$h_{\text{дпу}} = \frac{1}{n * t_{\text{изм}}} \sum_{i=(k-1)n+1}^{k*n} \int_{t_{\text{п}}(i-1)}^{t_{\text{п}}(i-1)+t_{\text{изм}}} [f(t) + \varphi(t)] dt, \quad (11)$$

Где $f(t)$ - функция, описывающая рельеф пласта;

$\varphi(t)$ - функция, описывающая формируемый исполнительным органом рельеф пласта;

n - число циклов интегрирования;

k - число зон интегрирования сигнала датчика, где $(k=1,2,3...m)$.

Выдача управляющего воздействия производится на основе сравнения $h_{\text{вх.р}}$ с установками регулирования $h_{\text{п}}^{\text{в}}$ и $h_{\text{п}}^{\text{н}}$.

$$\Delta h_{\text{п}} = \begin{cases} \Delta h_{\text{п}}, \text{ при } h_{\text{вх.р}} < h_{\text{п}}^{\text{н}}; \\ 0, \text{ при } h_{\text{п}}^{\text{н}} \leq h_{\text{вх.р}} \leq h_{\text{п}}^{\text{в}}; \\ -\Delta h_{\text{п}}, \text{ при } h_{\text{вх.р}} > h_{\text{п}}^{\text{в}}. \end{cases} \quad (12)$$

Величина управляющего воздействия может быть фиксирована либо пропорциональна величинам $(h_{\text{вх.р}} - h_{\text{п}}^{\text{н}}$ или $(h_{\text{п}}^{\text{н}} - h_{\text{вх.р}})$. В первом случае гидродомкрат включается в нужном направлении на время $t_{\text{в}} = \text{const}$, во втором случае $t_{\text{в}} \sim \Delta h_{\text{п}} (h_{\text{вх.р}} - h_{\text{п}}^{\text{н}})$ или $t_{\text{в}} \sim \Delta h_{\text{п}} (h_{\text{п}}^{\text{н}} - h_{\text{вх.р}})$.

Следует отметить сложность задания пределов интегрирования при наличии под датчиком участков пресекаемой породы. Выходная характеристика ДПУ (рис.2.4) предполагает, что при наличии породы под датчиком его сигнал условно равен «0» (const) в координатах «сигнал – пачка».

С учетом этого, уравнение (2.6) необходимо представить в виде полинома высокой степени, способного описать зависимость сигнала датчика как в положительной, так и в отрицательной области изменения угольной пачки.

Этот путь нерационален, так как связан со значительным усложнением вычислений.

Поэтому нами использованы в расчетах переменные пределы интегрирования на участках, где траектория РО пересекает границу «порода – уголь». Абсцисса точки пересечения, называемая точкой встречи $t_{вс}$, вычисляется по формуле:

$$t_{вс} = \frac{\arcsin \frac{h_{п}}{h_{г}}}{\omega}, \quad (13)$$

В зависимости от знака $h_{п}$ и значения текущей асциссы, значение $t_{вс}$ берется непосредственно в качестве одного из циклов интегрирования, либо складывается (со знаком) с полупериодом

$$\text{огibaющей } T = \frac{2\pi}{\omega}.$$

Отметим, что если ДПУ расположен на корпусе комбайна или погрузочном щитке, измерение толщины угольной пачки происходит непрерывно, а уравнение (2.9) упрощается и принимает вид:

$$h_{ввкп} = \frac{1}{t_a} \int_0^{t_a} h_{вкп}(t) dt. \quad (14)$$

Следующая функция регулятора состоит в установлении заданного времени, анализа сигнала t_a . Здесь источником информации является датчик скорости подачи комбайна (шага квантования) (ДСП). Применяемой в системе «САУК-М» ДСП выдает аналоговый сигнал (токовый), пропорциональный величине скорости подачи комбайна V_k . Требования формирования в процессе регулирования на почве пласта площадок равной длины (соответствующей шагу квантования l_k) означает, что необходимо интегрировать сигнал $i_{V_k}(t)$ в течении времени t_a , определяемого из соотношения:

$$t_a = \frac{l_k}{\int_0^{t_a} i_{V_k}(t) dt}, \quad (15)$$

Где знаменатель соотношения представляет собой среднее значение в скорости подачи V_k .

Регулятор производит также фиксацию момента времени t_0 окончания отработки управляющего воздействия. Здесь возможны два варианта решения – при фиксированной и переменной по величине отработки. В первом варианте отработка фиксируется постоянным временем $t_в$ включение гидродомкрата. В этом случае $t_в$ рассчитывается исходя из характеристики гидродомкрата. Во втором варианте используется петля обратной связи с датчиком выдвигки штока. Осуществляется непрерывное сравнение величины рассогласования значения $h_{ввкп}$ и соответствующей установки $h_{п}^в$ или $h_{п}^н$.

В процессе перемещения ГД сигнал ДВШ сравнивается с рассогласованием, что и дает момент отключения сигнала на отработку.

Эта функция регулятора реализуется операцией сравнения:

$$KO = \begin{cases} 0, \text{ если } h_{двш}(t) < h_{ввкп} - h_{п}^в; \\ 0, \text{ если } h_{двш}(t) < h_{п}^н - h_{ввкп}; \\ 1, \text{ если } h_{двш}(t) = h_{ввкп} - h_{п}^в; \\ 0, \text{ если } h_{двш}(t) = h_{п}^н - h_{ввкп}; \end{cases} \quad (16)$$

Где КО – логический сигнал конца отработки.

Последняя функция – передача приоритета управления углом наклона комбайна. Управление комбайном осуществляется в трех отдельных зонах Z_1 , Z_2 и Z_3 (см. рис.2.7), причем в зоне Z_2 производится плоскопараллельное перемещение Z_0 , а в зонах Z_1 и Z_3 – изменение угла наклона комбайна. Выбор расположения зон управления углом наклона в данной статье рассматриваться не будет. Переход от зоны к зоне фиксируется регулятором на основе информации ДВШ (либо эквивалентных, в данном случае, сигналов конечных выключателей).

Выводы. В результате проведенных исследований и анализа возмущающих воздействий на подвижный объект, была реализована математическая модель для создания регулятора, используемого для построения системы автоматического управления угледобывающими комбайнами в профиле пласта, которая может быть применима для работы в условиях безлюдной выемки угля.

Список библиографических ссылок.

1. Автоматический контроль положения режущего барабана угольных комбайнов. – Mining Journal 1976,286, № 7326, p.59 – 60.
2. Автоматическое управление комбайнами на шахте Уолстентон: Перевод № 1295/71. М.: Гипроуглемаш, 1971г. 33 с.
3. Автоматическое управление режущими органами угледобывающих комбайнов. Черняк З.А. и др. – «Безопасность труда в промышленности», 1982г, №15, с. 32-33.
4. Вебб Р. Автоматическое управление в профиле пласта комбайном Андертон. The Mining Engineer, 1967, V/ 196, №80. P.397.
5. Мелькумов Л.Г., Петунин Н.С. Автоматизация угледобывающих комбайнов: Обзор – М., 1976г. – 50 с.
6. Рудановский А.А. О рациональных способах автоматического управления угледобывающими комплексами в профильной плоскости пласта. – Уголь Украины, 1969г., №7, с. 19 – 21.
7. Ярыгин Б.Э. Автоматическое и дистанционное управление горными машинами, комплексами и агрегатами. Киев: Знание, 1977г.- с.12.
8. Ярыгин Б.Э. Автоматическое и дистанционное управление горными машинами, комплексами и агрегатами. Киев: Знание, 1977г.- с.12.
9. Рудановский А.А. Основы автоматического управления комбайнами в профиле пласта: автореферат док. диссертации. / ИГД им. А.А. Скочинского. М.- 1972. 41 с.
10. Рудановский А.А., Смиттен М.К. Исследование статистических характеристик микрорельефа угольных пластов – М: ИГД им.А. А. Скочинского, 1969. – 28с.
11. Каганюк А.К., славинский В. М., Черняк З. А. К расчету оптимальных параметров к системам автоматического управления комбайнами в профиле пласта. – В кН.: ИГД им. А.А. Скочинского. Науч. сообщ.Вып. 184. Совершенствование и повышение надежности горных машин и средств автоматизации
12. Докукин А.В., Красников Ю.Д., Хургин З.Я. Аналитические основы динамики выемочных машин. М.: Недра, 1966. – 160 с.
13. Нунупаров Г.М. Исследование и разработка бесконтактных датчиков «порода – уголь» для автоматизации шнековых, очистных комбайнов: Дис. На соиск.учен. степ. Канд. Техн. Наук/ МГИ. М., 1976. – 152 с.
14. Агейкин Д.И. Датчики контроля и регулирования: Справочные материалы. – М.: Машиностроение, 1965. – 928 с.

References

1. Automatic control of the position of the cutting drum of coal combines. - Mining Journal 1976.286, No. 7326, p. 59 - 60.
2. Automatic control of combines at the Walstenton mine: Translation No. 1295/71. M.: Giprouglemash, 1971. 33 sec
- 3..Automatic control of cutting organs of coal miners. Chernyak Z.A. and others. - "Labor safety in industry", 1982, No. 15, p. 32-33.
4. Webb R. Automatic control in the reservoir profile by the Underton combine. The Mining Engineer, 1967, V / 196, No. 80. P.397.
5. Melkumov L.G., Petunin N.S. Automation of coal miners: Overview - M., 1976. - 50 p.
6. Rudanovsky A.A. On rational methods for automatic control of coal mining complexes in the profile plane of the reservoir. - Coal of Ukraine, 1969, No. 7, p. 19-21.
7. Yarygin B.E. Automatic and remote control of mining machines, complexes and units. Kiev: Knowledge, 1977 - p. 12.
8. Yarygin B.E. Automatic and remote control of mining machines, complexes and units. Kiev: Knowledge, 1977 - p. 12.
9. Rudanovsky A.A. The basics of automatic control of combines in the reservoir profile: abstract doc. dissertation. / IGD them. A.A. Skochinsky. M. - 1972. 41 p.
10. Rudanovsky A.A., Smihten M.K. The study of the statistical characteristics of the microrelief of coal seams - M: IHD named after A. A. Skochinsky, 1969. -- 28 p.
11. Kaganyuk AK, Slavinsky V. M., Chernyak Z. A. Calculate optimal parameters for automatic control systems for combines in the reservoir profile. - In the book.: IHD them. A. A. Skochinsky. Scientific Messenger 184. Improving and improving the reliability of mining machines and automation
12. Dokukin A.V., Krasnikov Yu.D., Khurgin Z.Ya. Analytical fundamentals of the dynamics of mining machines. M.: Nedra, 1966. -- 160 s.
13. Nunuparov G.M. Research and development of non-contact sensors "rock - coal" for automation of auger, shearers: Dis. On competition. step. Cand. Tech. Science / MGI. M., 1976. - 152 p.
14. Ageykin D.I. Sensors of control and regulation: Reference materials. - M.: Mechanical Engineering, 1965. -- 928 p.

DOI: 10.36910/6775-2524-0560-2020-39-05

УДК: 004.01

Касянчук Дмитро Павлович, студент

Національний технічний університет України "КПІ ім.І.Сікорського"

ПОРІВНЯЛЬНИЙ АНАЛІЗ ГЕНЕРАТОРІВ ДОКУМЕНТАЦІЇ

Касянчук Д.П. Порівняльний аналіз генераторів документації. У статті описані переваги та недоліки різних генераторів. Проведено порівняльний аналіз заявлених генераторів для виявлення найбільш якісних та детальних серед них. Приведені ілюстрації їхнього синтаксису та результатів роботи.

Ключові слова: генератор документації, комерційна розробка, програміст, мови програмування, синтаксис, коментар.

Касянчук Д.П. Сравнительный анализ генераторов документации. В статье описаны преимущества и недостатки различных генераторов. Проведен сравнительный анализ заявленных генераторов для выявления наиболее качественных и детальных среди них. Приведены иллюстрации их синтаксиса и результаты работы.

Ключевые слова: генератор документации, коммерческая разработка, программист, языки программирования, синтаксис, комментарий.

Kasianchuk Dmytro. Comparative analysis of documentation generators. The article describes the advantages and disadvantages of various generators. A comparative analysis of the declared generators was carried out to identify the most high-quality and detailed ones among them. Translated illustrations of their syntax and results.

Keywords: documentation generator, commercial development, programmer, programming languages, syntax, commentary.

Постановка проблеми

Генератор документації — це програма, результатом виконання якої є отримання документації призначеної для програмістів або користувачів системи. Принцип роботи таких програм полягає в аналізі вихідного коду для знаходження синтаксичних конструкцій, які відповідають головним об'єктам програми (тип, метод, функція, процедура і т.д.). Спираючись на всю зібрану інформацію формується документація в одному з форматів — HTML, HTMLHelp, PDF, RTF та інших [3]. Залежно від того, який генератор використовується, є відмінності і в синтаксисі конструкцій, які наявні в документованих коментарях.

Документований коментар — спеціальним чином оформлений коментар, що відповідає певному об'єкту програми [3]. Як правило, вони можуть містити інформацію про автора коду, зміст вхідних і вихідних параметрів для функції і тому подібне. Загалом документація необхідна і дуже важлива для комерційної розробки, адже вона допомагає розібратися з тим, як працює система за менший період часу, а генератори, в свою чергу, значно зменшують кількість часу для створення документації.

Формування мети дослідження

Метою роботи є аналіз певних генераторів документації для порівняння, що дає змогу виявити переваги і недоліки кожного з них. Порівнюватися будуть наступні генератори: Doxygen, JSDoc, Sphinx, JSDuck, Slate.

Порівняльний аналіз генераторів документації

Doxygen (рис. 1) — використовується з такими мовами як C++, C, C#, Fortran, PHP, IDL, Java, VHDL, Objective-C, Python[4]. Даний генератор може сформувати документацію, яка буде містити в собі посилання, діаграми класів, викликів і т.п. в різних форматах: HTML, LaTeX, CHM, RTF, PostScript, PDF, map-сторінки. Doxygen дуже простий і зрозумілий в налаштуванні та установці. Його особливістю є можливість генерації документації не тільки на основі вихідного коду, а й на недокументованому вихідному коді. Також слід зазначити, що в документації, представлений в HTML форматі, реалізована можливість зручного пошуку, а також є посилання на зовнішню документацію. Якщо при роботі з Doxygen користувачу знадобилося задати додаткові параметри для майбутньої документації, то це можна зробити використовуючи конфігураційний файл, який являє собою звичайний текстовий формат. Проте слід зазначити і те, що для роботи з цим файлом існує декілька програм з графічним інтерфейсом, і це суттєво полегшує налаштування конфігураційного файлу. До мінусів можна віднести те, що цей генератор не має справжнього модульного налаштування.

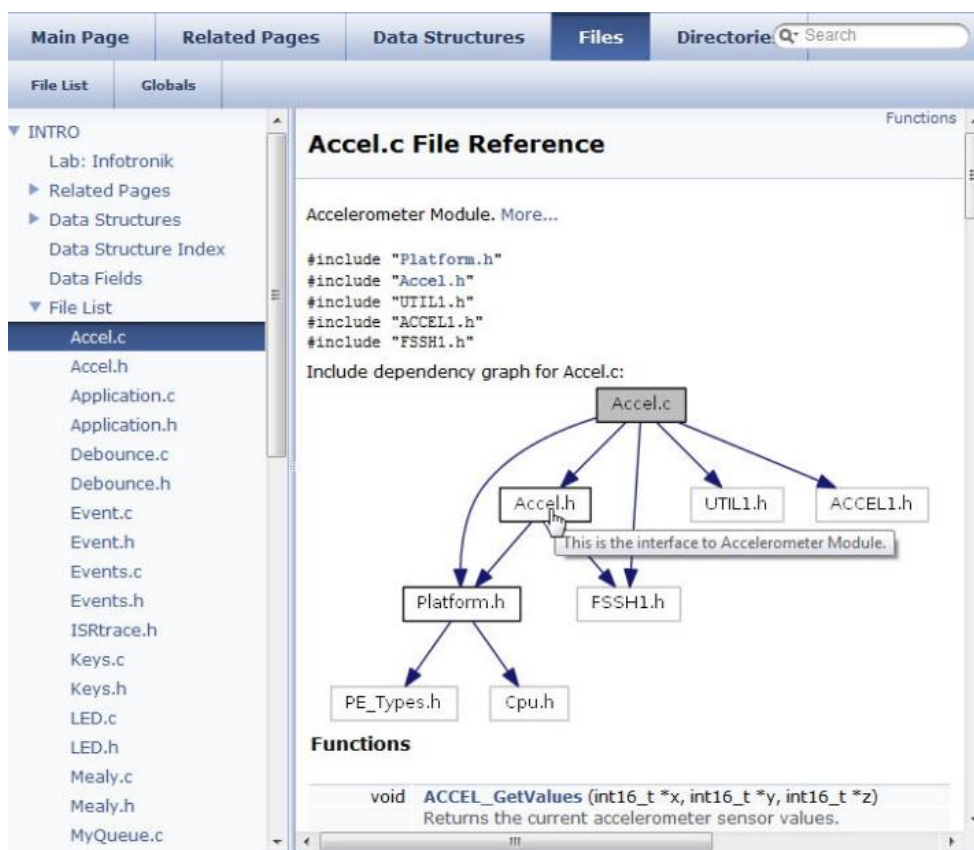


Рис. 1. Результат роботи генератора Doxygen

JSDoc

```
/**
 * Creates a new Circle from a diameter.
 *
 * @param {number} d The desired diameter of the circle.
 * @return {Circle} The new Circle object.
 */
static fromDiameter(d) {
  return new Circle(d / 2)
}

/**
 * Calculates the circumference of the Circle.
 *
 * @deprecated since 1.1.0; use getCircumference instead
 * @return {number} The circumference of the circle.
 */
calculateCircumference() {
  return 2 * Math.PI * this.radius
}
```

Рис. 2. Приклад синтаксису для JSDoc

JSDoc (рис.2) — це генератор документації в HTML форматі з вихідного коду на мові JavaScript. До переваг даного компілятора слід віднести те, що ядро генератора написано на мові JavaScript, а тому для генерації документації треба лише встановлена java-машина і більше ніяких

додаткових файлів. Також плюсом є наявність великої кількості прикладів використання JSDoc, що значно полегшує його освоєння. Також реалізована можливість роботи з коментарями російською мовою. Але існує і певний недолік, а саме, неможливість генерації документації для класів, які знаходяться в анонімній функції. Приклад такого класу:

```
include('class1.js', 'class2.js', function() {  
  /**  
   * Для цього класу генерація неможлива  
   * @class  
   * @constructor  
   */  
  var MyClass = function() {};  
});
```

Sphinx



Рис. 3. Результат роботи генератора Sphinx

Sphinx (рис.3) — генератор, який спочатку створювався лише для мови Python, але потім розробники додали підтримку C та C++ і планується додати підтримку ще більшій кількості популярних мов програмування. Принцип роботи генератора Sphinx в тому, що він перетворює файл в форматі reStructuredText в HTML, PDF, man та інші[5]. reStructuredText — це полегшена мова розмітки. Даний генератор легко встановити і він доступний для всіх операційних систем, які підтримують мову Python. До недоліків можна віднести те, що для використання Sphinx потрібно додатково розібратися з reStructuredText.

JSDuck

```
/** @example  
 * Ext.create('Ext.form.Panel', {  
 *   title: 'Contact Info',  
 *   width: 300,  
 *   bodyPadding: 10,  
 *   renderTo: Ext.getBody(),  
 *   items: [{  
 *     xtype: 'textfield',  
 *     name: 'name',  
 *     fieldLabel: 'Name',  
 *     allowBlank: false  
 *   }, {  
 *     xtype: 'textfield',  
 *     name: 'email',  
 *     fieldLabel: 'Email Address',  
 *     vtype: 'email'  
 *   }]  
 * });  
 */
```


Рис. 4. Приклад синтаксису JSDuck

JSDuck (рис.4) - генератор документації, цікавою особливістю якого є можливість використовувати його для будь-якого коду за умови, що цей код задокументовано відповідно до стандартів [2]. Також JSDuck відзначається легкістю в установці та використанні. Ще однією особливістю можна назвати те, що цей генератор може аналізувати код і генерувати документацію навіть без стандартних блоків. Реалізована можливість додавання спеціальних сторінок, а саме: сторінка привітання, сторінка з категоріями, сторінка з інструкціями, сторінка з прикладами. По підсумкам опитування, яке проводилося серед користувачів JSDuck, було зроблено висновок, що це дуже простий в розумінні і використанні генератор документації, недоліків якого не виявлено.

Slate

The screenshot shows the Slate API documentation interface. On the left is a navigation sidebar with a logo placeholder and a menu containing 'Introduction', 'Authentication', 'Kittens', 'Get All Kittens' (highlighted), 'Get a Specific Kitten', and 'Errors'. The main content area is titled 'Get All Kittens' and includes a description: 'This endpoint retrieves all kittens.' Below this is an 'HTTP REQUEST' section showing 'GET http://example.com/kittens'. A 'QUERY PARAMETERS' table follows:

Parameter	Default	Description
include_cats	false	If set to true, the result will also include cats.
available	true	If set to false, the result will include kittens that have already been adopted.

On the right, there are tabs for 'shell', 'ruby', and 'python'. The 'shell' tab is active, showing a code snippet:

```
require 'kittn'

api =
  Kittn::APIClient.authorize!('meowmeow')
api.kittens.get
```

Below the code, a text box states: 'The above command returns JSON structured like this:' followed by a JSON array:

```
[
  {
    "id": 1,
    "name": "Fluffums",
    "breed": "calico",
    "fluffiness": 6,
    "cuteness": 7
  },
  ...
]
```

Рис. 5. Інтерфейс генератора документації Slate

Slate (рис.5) — це генератор, який має інтуїтивно зрозумілий дизайн, він адаптивний і тому з ним можна працювати як на комп'ютерах так і на планшетах, ноутбуках, телефонах [1]. Slate має посилання на різні пункти документації, що допомагає користувачу швидко знайти потрібний розділ. Також цей генератор використовує Markdown і це спрощує розуміння та редагування. Ще одним плюсом для даного генератора є реалізована можливість підсвічування для 60-ти різних мов програмування і це додає комфорту до роботи з ним.

В таблиці 1 зображена оцінка за 5-ти бальною шкалою всіх засобів за трьома критеріями, які є важливими для генератора документації, а саме: кількість підтримуваних мов, кількість вихідних форматів та інтерфейс користувача. Інтерфейс користувача оцінювався за такими параметрами, як: зручність, відповідність сучасним тенденціям, а також нативність. Варто зазначити, що оцінка інтерфейсу – це суб'єктивна оцінка автора, а кількість підтримуваних мов і вихідних форматів – це значення, яке пропорційне статистичним даним взятим з різних джерел інформації. Сумарна оцінка вираховувалася, як середнє арифметичне цих оцінок.

Таблиця 1. Порівняння генераторів документації за різними критеріями

Інструмент \ Критерій	Кількість підтримуваних мов	Кількість вихідних форматів	Інтерфейс користувача	Сумарна оцінка
Doxygen	4	5	4	4.3
JSDoc	2	4	4	3.3
Sphinx	3	3	4	3.3
Slate	4	4	5	4.3
JSDuck	5	4	5	4.6

Висновок

Генератор документації — це дуже корисний і важливий інструмент в арсеналі програміста, тому він повинен бути зручним, зрозумілим, простим і детальним. З огляду на все вищесказане, можна назвати JSDuck беззаперечним лідером серед представлених генераторів документації. Цей генератор має всі якості, якими повинен бути наділений потужний інструмент для генерації документації. Реалізована підтримка для всіх мов програмування, що робить його універсальним генератором, який підійде для розробника на будь-якій мові програмування. Не поступається JSDuck і генератор Slate, він підтримує 60 мов програмування, а також з ним можна працювати знаходячись будь-де, адже він має адаптивний дизайн і чудово виглядає як на екрані комп'ютера, так і на телефоні. Проте, якщо Ви розробник на мові C, C++, Python, то для Вас будуть цікаві і такі генератори, як Sphinx та Doxygen. Вони зрозумілі, зручні та детальні. Хоча слід пам'ятати і те, що для використання Sphinx буде потрібно розібратися з reStructuredText. Якщо Вам цікаві генератори документації для мови JavaScript, то слід звернути увагу на JSDoc. З використанням цього генератору легко розібратися, адже існує велика кількість прикладів, а також він не потребує додаткових файлів для установки. Проте проблема генератору в тому, що він не генерує документацію для класів, які знаходяться в анонімній функції.

Список бібліографічного опису

1. Slate API Docs Generator, <https://ruprogi.ru/software/slate-api-docs-generator>
2. JSDuck — генератор документації, <https://habr.com/ru/post/214591/>
3. Генератор документації, https://ru.wikipedia.org/wiki/Генератор_документації
4. Doxygen, <http://www.doxygen.nl>
5. Sphinx, <https://www.sphinx-doc.org>

References

1. Slate API Docs Generator, <https://ruprogi.ru/software/slate-api-docs-generator>
2. JSDuck — generator documentacii, <https://habr.com/ru/post/214591/>
3. Generator documentacii, https://ru.wikipedia.org/wiki/Генератор_документації
4. Doxygen, <http://www.doxygen.nl>
5. Sphinx, <https://www.sphinx-doc.org>

DOI: 10.36910/6775-2524-0560-2020-39-06

УДК: 378.147-057.87:005.336.2(477.82)

Клехо Олена Веніамінівна, викладач інформатики

<https://orcid.org/0000-0002-2270-0898>

Лехіцький Тарас Володимирович, викладач інформатики

Луцький педагогічний коледж, місто Луцьк

ФОРМУВАННЯ ФАХОВИХ КОМПЕТЕНТНОСТЕЙ СТУДЕНТІВ ПЕДАГОГІЧНОГО КОЛЕДЖУ В УМОВАХ ПРОФЕСІЙНО-ПРАКТИЧНОЇ ПІДГОТОВКИ

Клехо О.В., Лехіцький Т.В. Формування фахових компетентностей студентів педагогічного коледжу в умовах професійно-практичної підготовки. У статті розглядається проблема формування фахових компетентностей у студентів педагогічного коледжу під час підготовки та проходження практики. З'ясовано сучасні підходи до фахової підготовки майбутніх вчителів інформатики для нової української школи, а також висвітлено особливості змісту та методів професійно-практичної підготовки студентів у коледжі. Дослідження базуються на основі викладання фахових дисциплін та проходження різних видів практики для спеціальності «Початкова освіта» з додатковою кваліфікацією вчитель інформатики в початковій школі Луцького педагогічного коледжу. Авторами продемонстровано навчально-методичні матеріали для теоретично-практичної підготовки майбутніх вчителів, а саме, інноваційні технології (мультимедійні, мережеві), технології проблемного та інтерактивного навчання, робимо акцент на тому, що цікаво сучасному студенту, не відмовляючись від традиційної методики. В результаті дослідження обґрунтували окремі аспекти формування фахової компетентності вчителя початкової школи, дослідили і описали структуру та зміст практичної підготовки майбутніх вчителів інформатики у початковій школі. Переконались, що професійно-практична підготовка студентів коледжу – це пріоритетна система освітнього процесу, спрямована на творче застосування знань у процесі реальної педагогічної ситуації.

Ключові слова: компетентність, фахова компетентність, компетентнісний підхід, вчитель інформатики в початковій школі, методи та форми навчання.

Клехо Е.В., Лехицкий Т.В. Формирование профессиональных компетенций студентов педагогического колледжа в условиях профессионально-практической подготовки. В статье рассматривается проблема формирования профессиональных компетенций у студентов педагогического колледжа при подготовке и прохождении практики. Выяснено современные подходы к профессиональной подготовке будущих учителей информатики для новой украинской школы, а также освещены особенности содержания и методов профессионально-практической подготовки студентов в колледже. Исследования базируются на основе преподавания профессиональных дисциплин и прохождения различных видов практики для специальности «Начальное образование» с дополнительной квалификацией учитель информатики в начальной школе Луцкого педагогического колледжа. Авторами продемонстрировано учебно-методические материалы для теоретико-практической подготовки будущих учителей, а именно, инновационные технологии (мультимедийные, сетевые), технологии проблемного и интерактивного обучения, делаем акцент на том, что интересно современному студенту, не отказываясь от традиционной методики. В результате исследования обосновали отдельные аспекты формирования профессиональной компетентности учителя начальной школы, исследовали и описали структуру и содержание практической подготовки будущих учителей информатики в начальной школе. Убедились, профессионально-практическая подготовка студентов колледжа - это приоритетная система образовательного процесса, направленная на творческое применение знаний в процессе реальной педагогической ситуации.

Ключевые слова: компетентность, профессиональная компетентность, компетентностный подход, учитель информатики в начальной школе, методы и формы обучения.

Klekh O., Lekhitskyi T. Formation of professional competences of students of pedagogical college in the conditions of vocational training. The article deals with the problem of formation of professional competences in the students of the pedagogical college during the preparation and passing of the practice. The modern approaches to the professional training of future computer science teachers for the new Ukrainian school are clarified, as well as the peculiarities of the content and methods of professional and practical training of students in college. The research is based on the teaching of professional disciplines and the passage of various types of practice for the specialty "Elementary education" with additional qualification as a teacher of computer science at the elementary school of Lutsk Pedagogical College. The authors demonstrate educational and methodological materials for theoretical and practical training of future teachers, namely, innovative technologies (multimedia, networking), technologies of problem and interactive learning, we emphasize that it is interesting to the modern student, without abandoning the traditional methodology. As a result of the research, some aspects of the formation of professional competence of elementary school teachers were substantiated, the structure and content of practical training of future elementary school teachers were analyzed and described. They were convinced that vocational training of college students is a priority system of the educational process, aimed at creative application of knowledge in the process of real pedagogical situation.

Keywords: competence, professional competence, competence approach, elementary school computer science teacher, teaching methods and forms.

Постановка наукової проблеми. Головною метою підготовки фахівця у соціально-економічних умовах інформаційного суспільства стає не здобуття ним кваліфікації у вибраній сфері, а набуття та розвиток певних компетентностей, які мають забезпечити йому можливість адаптуватися в умовах динамічного розвитку сучасного світу.

Проблема формування професійної компетентності майбутнього вчителя початкових класів – це підготовка вчителя, здатного глибоко і критично мислити, моделювати освітній процес, самостійно генерувати і втілювати у практику початкової школи нові ідеї та технології навчання і виховання. Професійно-компетентний вчитель має потенціал як для досягнення вагомих результатів у навчальній діяльності, так і для позитивного впливу на формування цілісної особистості учня, бажання постійно працювати над собою, підвищувати самостійно свій фаховий рівень підготовки [11].

Авторські дослідження будуть базуватися на основі викладання фахових дисциплін та проходження різних видів практики для спеціальності «Початкова освіта» з додатковою кваліфікацією вчитель інформатики в початковій школі Луцького педагогічного коледжу, і є продовженням публікації результатів досліджуваної проблеми.

Аналіз останніх публікацій і досліджень. Закон України «Про освіту» [1], Національна доктрина розвитку освіти [2] пропагують особистісний розвиток вчителя як найвищу цінність суспільства. Цьому присвячені наукові дослідження у галузі філософії освіти (В.Андрущенко, Б.Гершунський, І.Зязюн, В. Кремень та ін.); професійної підготовки фахівців у вищій школі (А. Алексюк, С. Гамараш, С. Гончаренко, А. Гуржій, А. Капська, В. Козаков, М. Лазарев, П.Підкасистий, В.Семиченко, С.Сисоєва, А.В.Сохань [8] та ін.); теорії компетентнісного підходу до навчання (В. Байденко, Є. Зеєр, І. Зимня, І. Зязюн, Н. Кузьміна) [5].

Постановка завдання. Метою статті є продовження теоретичного-практичного обґрунтування окремих аспектів формування фахової компетентності майбутнього вчителя початкової школи та вчителя інформатики в початковій школі, а також висвітлення структури та змісту практичної підготовки майбутніх вчителів інформатики у початковій школі, які успішно реалізують суть компетентнісного підходу на заняттях з методики навчання інформатики у коледжі. А також розробка і демонстрація навчально-методичних матеріалів для підготовки професійно компетентного випускника, педагога.

Виклад основного матеріалу й обґрунтування отриманих результатів досліджень. Давня мудрість говорить: хто стоїть на місці, той відстає, а хто не хоче відставати, мусить рухатися вперед, і не зупинятися, досягнувши вершини, і підійматися вище. У цьому й полягає основна місія сучасного учителя. Вже не достатньо бути на уроці та поза ним актором, режисером, діловодом, диригентом, дипломатом, психологом, новатором і компетентним фахівцем. Модель сучасного вчителя передбачає готовність до застосування нових освітянських ідей, здатність постійно навчатися, бути у постійному творчому пошуку. Ці якості не видаються з додатком до диплома про педагогічну освіту, а формуються у щоденній учительській праці.



Рис.1. Результати опитування студентів.

Досліджуючи питання професійної компетентності, ми провели опитування студентів, і отримали результати, які змусили нас попрацювати над питанням формування фахової компетентності.

Освітній простір XXI століття визначає компетентнісна освіта. У зв'язку з цим сьогодні головним виміром якості функціонування освітньої системи має бути здатність молодого покоління повноцінно жити і активно діяти в новому світі, постійно самовдосконалюватись, адекватно реагувати на зміни, особливо у періоди технологічних та цивілізаційних змін.

Тобто, щоб прийняти історичний виклад XXI століття, освіта має носити випереджувальний характер, бути націленою в майбутнє, на розв'язання проблем нового століття, розвиток

компетентності учнів, формування у них проєктивної культури, нових способів мислення і діяльності. На сучасному етапі в Україні компетентнісний підхід визнано одним із напрямків модернізації освіти. Готовність українських педагогів запровадити компетентнісний підхід у системі освіти задекларовано в основних освітніх документах. Так, у Концепції загальної середньої освіти формування основних компетентностей особистості – життєвої, соціальної, загальнонаукової, загальнокультурної, технологічної, комунікативної і комп'ютерної – визначається як одне з головних завдань сучасної школи України.

Відповідно до вимог нового Державного стандарту дошкільної, базової та повної загальної середньої освіти сучасна парадигма національної освіти в Україні ґрунтується на засадах особистісно зорієнтованого, компетентнісного і діяльнісного підходів, що реалізовані в освітніх галузях і відображені в результативних складових змісту дошкільної, базової і повної загальної середньої освіти. При цьому особистісно зорієнтований підхід до навчання забезпечує розвиток академічних, соціокультурних, соціально-психологічних та інших здібностей дошкільників та учнів, а компетентнісний підхід сприяє формуванню ключових і предметних компетентностей. Отже, в умовах модернізації змісту національної освіти в Україні, пошуку нових освітніх парадигм, орієнтованих на розвиток всебічно освіченої, творчої особистості, особливого значення набуває реалізація в педагогічній практиці сучасної школи саме компетентнісного підходу.

Таким чином, сучасна освіта, зорієнтована на компетентісну модель, збільшує її результативний компонент, визначає переміщення акцентів з накопичення обсягу знань на цілеспрямований розвиток ключових і предметних компетентностей як інтегрованої якості особистості.

Саме це завдання, постійно навчатися, бути творчим, інноваційним, працюючим над собою педагогом, ставлять викладачі перед собою та студентами.

З цією метою викладачами ведеться ґрунтовна робота щодо підготовки матеріально-технічної та методичної бази для студентів ОКР «Молодший спеціаліст», «Бакалавр» з фахових дисциплін, для підготовки та проведення уроків з інформатики.

Аналізуючи зміст своєї роботи ми можемо сказати, що для теоретично-практичної підготовки майбутніх вчителів намагаємось використовувати інноваційні технології (мультимедійні, мережеві), технології проблемного та інтерактивного навчання, робимо акцент на тому, що цікаво сучасному студенту, не відмовляючись від традиційної методики.

Не менш важливу роль у формуванні професійних компетентностей під час проходження практики, відіграють вдало підібрані практичні завдання, ситуативні задачі, ділові ігри, які використовуються під час вивчення фахових дисциплін, розв'язування яких вимагає від майбутніх вчителів активної практично-пошукової діяльності, самопідготовки та творчості.

Дослідивши структуру та зміст навчально-методичних, мультимедійних, програмно-технічних матеріалів педагогічної практики, пропонуємо найцікавіші приклади:

- навчально-методичний посібник для студентів-практикантів з додаткової кваліфікації «Вчитель інформатики в початковій школі», який розповсюджений в друкованому вигляді та на сайті електронної бібліотеки коледжу;

- електронні посібники, авторські сайти із зразками уроків для вчителя інформатики в початковій школі, матеріали для проведення виховних годин;

- систематизовано дидактичні матеріали для підготовки уроків інформатики на хмарному сервісі Padlet;

- підготовка і проведення засідань круглих столів, за участю студентів, вчителів шкіл області, викладачів, науковців, на яких обговорюються актуальні питання організації освітнього процесу, з метою набуття здатності ефективно діяти, розв'язуючи стандартні та проблемні методичні задачі;

- налагоджено спільне створення та використання конспектів, презентацій до уроків на сервісі Google Docs, інших хмарних сервісів;

- систематизовано та розповсюджено у вигляді веб-ресурсів на безкоштовному хостингу матеріали (технологічні картки, розробки уроків, пакети тестових завдань, фізкультхвилинки, відеозавдання ін.) для підготовки та проведення уроків інформатики.

Наскрізне застосування інформаційно-комунікаційних технологій в освітньому процесі стає пріоритетним інструментом забезпечення успіху нової української школи і перехід від одноразових проєктів у системний процес, який охоплює всі види діяльності.

Інтегроване навчання легко і на сучасному етапі реалізується на уроках інформатики під час роботи над проєктами, при поясненні нового матеріалу та на етапі закріплення. Студенти за

допомогою сервісів, що вивчали під час методики навчання інформатики у коледжі, розробляють завдання з інших навчальних дисциплін, які учні виконують, наприклад, як завдання для розвитку навичок володіння клавіатурою, мишею.

Реалізація STEM-освіти може бути в якості предметів за вибором: наприклад, Робототехніка, «Цікава математика з Lego». Проте більшість фахівців схиляється до думки про STEM-освіта не як окремий центр при школі, а напрями, впроваджені у весь навчальний процес закладу, інтеграцією предметів. Адже конструювання можна використовувати не тільки для загального розвитку, а й у математиці, вивченні світу, розвиткові мови.

Використання конструкторів Lego, які дозволяють програмувати, зокрема комплекту WeDo 2.0 дозволяє проводити захоплюючу проектну діяльність з учнями 1-4 класів, використовуючи їх навички проектування, конструювання та програмування. Проекти, що входять до комплекту, наочно демонструють учням принципи роботи різних механізмів, фізичних законів або природних явищ.

На нашу думку, важливим для успішного формування професійних компетентностей в умовах професійно-практичної підготовки є залучення до створення професійно значущих продуктів. Тому і при вивченні інформативних дисциплін, і методики навчання інформатики ми ставимо акцент не на опануванні навичками користувача, а на підготовку до використання ІТ у професійній діяльності:

– розробка дидактичних матеріалів з використанням різних сервісів: Padlet, LearningApps, PowToon (створення навчальних відео) та Ted-Ed (створення відеороликів);

– формування вмінь розміщення навчальних матеріалів для своєї професійної діяльності на власному ресурсі (ведення власних блогів, створення та підтримка сайтів);

– використання смартфонів в якості цінних інструментів з навчальною метою, які підтримують знання, а не відволікають від навчального процесу;

– вміння будувати позитивну поведінку онлайн (як правильно використовувати соціальні медіа, як виробляти і публікувати цінний контент);

– вміння бути інноваційним (передбачає використання не тільки традиційних підручників, а й соціальних медіа, TED-роликів) та прагнути до саморозвитку (використовувати нові можливості, що дають сервіси дистанційного навчання, Prometheus, Coursera)

На заняттях з дисципліни «Методика навчання інформативної освітньої галузі» студенти знайомляться з новітнім освітнім програмним забезпеченням, принципами роботи інтерактивної панелі у лабораторії НУШ та смарткейсами для вчителя початкової школи та методикою її використання. У лабораторії НУШ студенти освоюють програму MozaBook.

Висновки і перспективи подальших досліджень: Таким чином, використавши методи наукових досліджень, проаналізувавши ряд літературних та інтернет-джерел, акумулювавши власний досвід та матеріали інших авторів, ми отримали наступні результати:

- обґрунтували окремі аспекти формування фахової компетентності вчителя початкової школи та вчителя інформатики в початковій школі;

- дослідили і описали структуру та зміст практичної підготовки майбутніх вчителів інформатики у початковій школі;



Рис. 2. Результати опитування студентів та випускників, з метою планування подальших досліджень

- переконались, що професійно-практична підготовка студентів коледжу – це пріоритетна система методів та засобів, спрямованих не тільки на набуття компетенцій шляхом отримання знань,

їхнє запам'ятовування, а й на відтворення та творче застосування у процесі реальної педагогічної ситуації.

Надалі наші дослідження будуть спрямовані на вивчення окремих складових професійної компетентності майбутнього вчителя початкової школи, зокрема інформаційної компетентності з подальшою розробкою та використанням демонстраційних прикладів, інструктивних карток та інших навчально-методичних матеріалів.

Список бібліографічного опису

1. Закон України «Про освіту».
2. Національна доктрина розвитку освіти.
3. Онищенко І., Модель формування фахової компетентності майбутнього вчителя початкових класів [Електронний ресурс] / Ірина Онищенко – Режим доступу: <http://archive.nbuv.gov.ua/portal>
4. Пометун О. [Текст] Навчасмо по-іншому [посібник для викладачів професійно-технічних училищ] / О.Пометун та ін. К.,2008. – с. 14-15.
5. Психологія і педагогіка життєтворчості / Навч.-метод. посібник за ред. Л.В.Сохань, І.Г.Єрмакова, В.О.Тихоновича. – К., 1996. – 792 с.
6. Симуляції або імітаційні ігри [Електронний ресурс] – Режим доступу: <http://www.rusnauka.com>
7. Ярошинська О. Формування фахової компетентності майбутніх учителів початкової школи в умовах освітнього середовища професійної підготовки [Електронний ресурс] / Олена Ярошинська – Режим доступу: <http://irbis-nbuv.gov.ua>
8. Ящук І. П. Формування життєвокомпетентної особистості : науково-методичний посібник / І. П. Ящук. – Хмельницький : Видавництво Хмельницького гуманітарно-педагогічного інституту, 1999. – 54 с.
9. Технології опрацювання дискусійних питань, [Електронний ресурс] – Режим доступу: http://studopedia.com.ua/1_278032_tehnologii-opratsyuvannya-diskusiynih-pitan.html

References

1. The Law of Ukraine «On Education».
2. National doctrine of educational development.
3. Onishchenko, Model of formation of professional competence of the future elementary school teacher [Electronic resource] / Irina Onishchenko – Access mode: <http://archive.nbuv.gov.ua/portal>
4. Pometun O. [Text] We teach differently [a guide for teachers of vocational schools] / O.Pometun et al. – K., 2008. - with. 14-15.
5. Psychology and pedagogy of life-creation / Educational method. manual ed. LVSokhan, IGYermakova, VO Tikhonovich. – K., 1996. - 792 p.
6. Simulations or Simulation Games [Online Resource] – Access Mode: <http://www.rusnauka.com>
7. Yaroshynska O. Formation of professional competence of future primary school teachers in the conditions of vocational training environment [Electronic resource] / Elena Yaroshynska – Access mode: <http://irbis-nbuv.gov.ua>
8. Yashchuk IP Formation of a life-competent personality: a scientific and methodological manual / IP Yashchuk. – Khmelnytsky: Publishing House of Khmelnytsky Humanitarian and Pedagogical Institute, 1999. - 54 p.
9. Technologies for working out discussion questions, [Electronic resource] – Access mode: http://studopedia.com.ua/1_278032_tehnologii-opratsyuvannya-diskusiynih-pitan.html

Рецензенти: **Муляр Вадим Петрович**, кандидат педагогічних наук, доцент кафедри експериментальної фізики та інформаційно-вимірювальних технологій Східноєвропейського національного університету ім.Лесі Українки, **Лук'янчук Мар'яна Василівна**, кандидат педагогічних наук, викладач іноземних мов Луцького педагогічного коледжу

Стаття надійшла 03.04.2020

DOI: 10.36910/6775-2524-0560-2020-39-07

УДК: 621.313.33

Kostiuchko Serhii Mykolaiovych, Ph.D.

<https://orcid.org/0000-0002-1262-6268>

Kyryliuk Liudmyla Mykolaivna, master

Kalys Oleh Viacheslavovych, student

Sibanda Zeti Fortune, student

Havryliuk Stanislav Andriiovych, student

Lutsk National Technical University

THE MONODROMY MATRIX CONSTRUCTION FOR EXECUTIVE OBJECT OF A NONLINEAR SYSTEM.

Костючко С.М., Кирилюк Л.М., Кались О.В., Сібанда З.Ф., Гаврилюк С.А. Побудова матриці монодромії виконавчого об'єкта нелінійної системи. У статті розкрито один із методів побудови матриці монодромії, розкрито суть найпростішої побудови та обчислення такої матриці. За цим методом будується математична модель виконавчого пристрою, яка може бути використана для дослідження перехідних та усталених станів.

Ключові слова: матриця монодромії, математична модель, асинхронний мотор, рівняння стану.

Костючко С.Н., Кирилюк Л.Н., Кались А.В., Сибанда З.Ф., Гаврилюк С.А. Построение матрицы монодромии исполнительного объекта нелинейной системы. В статье раскрыт один из методов построения матрицы монодромии, раскрыта суть самой простой построения и вычисления такой матрицы. По этому методу строится математическая модель исполнительного устройства, которая может быть использована для исследования переходных и установившихся состояний.

Ключевые слова: матрица монодромии, математическая модель, асинхронный двигатель, уравнение состояния.

Kostiuchko S.M., Kyryliuk L.M., Kalys O.V., Sibanda Z.F., Havryliuk S.A. The monodromy matrix construction for executive object of a nonlinear system. The article reveals one of a monodromy matrix constructing methods, reveals the essence of the simplest construction and calculation such matrix. This method is used to build a actuator mathematical model, which can be used to study transients and steady-state processes.

Keywords: monodromy matrix, mathematical model, asynchronous motor, state equation.

Problem formulation

The monodromy matrix of any physical device is used in the steady-state processes, static stability and parametric sensitivity analysis. We will show how to find it in the case of electrical devices using mathematical models on the example of an induction motor with a rotor rectangular groove.

Recent research analysis

To solve this problem, it was necessary to solve several important theoretical problems, in particular, to develop: a general theory of the electric skin effect, the principle of constructing circuit-field electrical devices mathematical models, and the theory of parametric sensitivity auxiliary model. This became the basis for constructing matrices of monodromy systems.

Mathematical model

The machines rotor winding by the number of turns is considered to be reduced to 125 stator windings. Thus, it is flowed not by physical but by reduced currents. At the same time, these currents appear in the boundary conditions of the electromagnetic field. Therefore, the equation of the electromagnetic circuit also must be reduced together with the currents. For which it is enough to give the environment parameters. The rotor is made in the squirrel wheel form. This is a multi-loop system, which is usually equivalent to two circuits. However, such equivalence should not affect the geometry of the groove space. All groove dimensions must be intact. In the transformed coordinates, the rotor winding currents and voltages also are reduced in frequency to the currents and voltages of the stator winding. The electromagnetic field equation also must be given. The motor's electromagnetic state equation is written in the form

$$\frac{d\mathbf{i}}{dt} = \mathbf{A}(\mathbf{u} - \mathbf{\Omega}^T \mathbf{\Psi} - \mathbf{R}\mathbf{i}) \quad (1)$$

where

$$\begin{bmatrix} \lambda_S \\ \lambda_R \end{bmatrix}, \lambda = \mathbf{u}, \Psi, \mathbf{i}; \mathbf{A} = \begin{bmatrix} A_S & A_{SR} \\ A_{RS} & A_R \end{bmatrix}; \mathbf{\Omega}' = \begin{bmatrix} & \\ & \Omega \end{bmatrix}; \mathbf{R} = \begin{bmatrix} \mathbf{R}_S & \\ & \mathbf{R}_R \end{bmatrix}. \quad (2)$$

Here $\mathbf{i}_k = (i_{kA}, i_{kB})_t$, $k = S, R$ are columns of stator winding phase currents and the rotor winding converted currents; $\mathbf{u}_k = (u_{kA}, u_{kB})_t$, $k = S, R$ are the stator winding phase voltages columns; A_S, A_{SR}, A_{RS}, A_R are matrixes

$$A_S = \alpha_S(1 - \alpha_S \mathbf{G}); A_{SR} = A_{RS} = -\alpha_S \alpha_R \mathbf{G}; A_R = \alpha_R(1 - \alpha_R \mathbf{G}), \quad (3)$$

where $\mathbf{G}, \mathbf{\Omega}$ are matrixes

$$\mathbf{G} = \begin{bmatrix} T + b_A i_A & b_B \\ b_A i_B & T + b_B i_B \end{bmatrix}, \mathbf{\Omega} = \frac{\omega}{\sqrt{3}} \begin{bmatrix} -1 & -2 \\ 2 & 1 \end{bmatrix}, \quad (4)$$

moreover

$$b_A = b(2i_A + i_B); b_B = b(i_A + 2i_B); b = \frac{2}{3} \frac{R - T}{i_m^2}; \quad (5)$$

$$R = \frac{1}{\alpha_S + \alpha_S + \rho}; T = \frac{1}{\alpha_S + \alpha_S + \tau}.$$

Here τ, ρ are inverse static and differential inductances, they are found by the characteristic of magnetization (idling) of the machine as:

$$\tau = \left[\frac{\Psi_m(i_m)}{i_m} \right]^{-1}; \rho = \left[\frac{d\Psi_m(i_m)}{di_m} \right]^{-1}, \quad (6)$$

where i_m is the module of spatial vector of magnetization currents

$$i_m = 2\sqrt{(i_A^2 + i_{A'B}^2 + i_B^2) / 3}; i_A = i_{SA} + i_{RA}; i_B = i_{SB} + i_{RB}. \quad (7)$$

In the absence of saturation, the magnetization characteristic degenerates directly $i_m = \alpha_m \Psi_m$, where α_m is the main inverted inductance, and the matrix (4) according to (6) degenerates into a diagonal

$$\mathbf{G} = \frac{1}{\alpha_S + \alpha_R + \alpha_m} \begin{bmatrix} 1 & \\ & 1 \end{bmatrix}, \quad (8)$$

which greatly simplifies the equation (1). In this case, we obtain the simplest of all known the induction motor mathematical models. $\mathbf{R}_S, \mathbf{R}_R$ are resistance matrix

$$\mathbf{R}_S = \begin{bmatrix} r_S & \\ & r_S \end{bmatrix}; \mathbf{R}_R = \begin{bmatrix} r_R & \\ & r_R \end{bmatrix}, \quad (9)$$

moreover α_S, α_R are inverse inductances of dissipation of stator and rotor windings; r_S is stator phase resistance; r_R is the resistance of the rotor winding; $\mathbf{\Omega}$ are angular velocity matrix ω .

The column's components of complete flux couplings of the stator and rotor windings are found as follows

$$\Psi_{kj} = \frac{1}{\tau} (i_{Sj} + i_{Rj}) + \frac{1}{\alpha_k} i_{kj}, \quad j = A, B; k = S, R. \quad (10)$$

Column's elements of a stator and rotor voltages

$$u_S = (U_m \sin(\omega_0 t), U_m \sin(\omega_0 t - 120^\circ))_t; u_R = 0, \quad (11)$$

where U_m, ω_0 are amplitude and circular frequency of network voltage.

The equation of mechanical state has the form

$$\frac{d\omega}{dt} = J(M_E - M) / p_0, \quad (12)$$

moreover

$$M_E = \sqrt{3} p_0 (\Psi_{SA} i_{SB} - \Psi_{SB} i_{SA}), \quad (13)$$

where $M(\omega)$ is mechanical moment; p_0 is the magnetic poles number of pairs; J is the moment of rotor inertia; M_E is electromagnetic moment.

System of differential equations (1), (12) are mathematical A-model of the actuator (asynchronous motor). It is intended for the transient and steady-state processes analysis. For its practical use it is necessary to know the following object parameters: stator and rotor windings dissipation supports and inverse inductances; idle characteristic, and without taking into account the saturation of the main magnetic circuit - the machine's inverse main inductance, the magnetic poles number of pairs and the rotor inertia moment. The input signals are: phase supply voltages and mechanical torque on the shaft.

Monodromy matrix construction

When constructing an auxiliary sensitivity model, we form a unknowns column \mathbf{y}

$$\mathbf{y}(t) = (\Psi, \omega)_t, \quad (14)$$

The monodromy matrix is written in the form (everywhere later the matrix $\mathbf{S} = \mathbf{z}$)

$$\Phi = (\mathbf{A}\mathbf{z}, \mathbf{w})_t, \quad (15)$$

Thus, the executive object's monodromy matrix construction to requires the integration the first variation's equations $\frac{d\mathbf{z}}{dt}$ and $\frac{d\mathbf{w}}{dt}$:

$$\frac{d\mathbf{z}}{dt} = -(\mathbf{\Omega} + \mathbf{R}\mathbf{A})\mathbf{z} - \frac{\partial \mathbf{\Omega}}{\partial \omega} w \Psi, \quad (16)$$

where

$$\mathbf{z} = \frac{\partial \Psi}{\partial \mathbf{x}(0)}; \quad \mathbf{w} = \frac{\partial \omega}{\partial \mathbf{x}(0)}. \quad (17)$$

Matrix $\partial \mathbf{\Omega} / \partial \omega = \text{const}$. The matrix structural form can be determined using the equation $\mathbf{\Omega} = \text{diag}(1, \mathbf{\Omega}_R)$ and formula

$$\mathbf{\Omega} = \frac{\omega}{\sqrt{3}} \begin{bmatrix} 1 & 2 \\ -2 & -1 \end{bmatrix}. \quad (18)$$

$$\frac{d\mathbf{w}}{dt} = \frac{p_0}{J} \left(\sqrt{3} p_0 \left(\frac{\partial \Psi_s}{\partial \mathbf{x}(0)} \times \mathbf{i}_s + \Psi_s \times \frac{\partial \mathbf{i}_s}{\partial \mathbf{x}(0)} \right) - \frac{\partial M(\omega)}{\partial \omega} w \right). \quad (19)$$

On the s-th iteration of Newton's formula

$$\mathbf{x}(0)^{(s+1)} = \mathbf{x}(0)^{(s)} - f'(\mathbf{x}(0)^{(s)})^{-1} f(\mathbf{x}(0)^{(s)}), \quad (20)$$

linear variational equations (16), (19) are subject to compatible integration with nonlinear (1), (12) on the time interval $[0, T]$.

The proposed analysis method has received a comprehensive test in complex problems of electromechanics. And it turned out to be very effective.

Conclusions

Determination of the electrical devices monodromy matrix is most simply carried out on the basis of integrating the first variation equations of the differential equations of device's state. Only on the basis of the monodromy matrix there is a practical possibility to build general algorithms for the analysis of physical devices in full, using the general theory similar mathematical apparatus of nonlinear differential equations. This applies to the analysis of transients and steady-state processes, determining the static stability of the found steady-state processes and, finally, to find parametric sensitivities matrices in transient and steady-state processes.

References

1. G. Nalecz. Application of Sensitivity Methods to Analysis and Synthesis of Vehicle Dynamic Systems. // Vehicle System Dynamics, 18 (1989), 0042-3114/89/1801-0001, Swets & Zeitlinger, pp. 1-44.
2. Tchaban V. Auxiliary model of parametric sensitivity / Vasil Tchaban, Serhiy Kostiucho, Zorana Tchaban // Computational Problems of Electrical Engineering. – Lviv, 2012. – V. 2, № 2. – P. 105-111.
3. Lyashko S.I. Generalized Optimal Control of Linear Systems with Distributed Parameters / Dordrecht, Boston, London. – Kluwer Academic Publishers. – 2002. – 455 p.
4. Aprile T.I. A computer algorithm to determine the steadystate response of non-linear oscillators / Aprile T.I., Trick T.N. // IEEE, Trans. Circuit Theory. – 1972. – vol. 19. – P. 354-360.
5. Korn G. Mathematical handbook for scientists and engineers / Korn G., Korn T. Toronto, London: Mc. Graw-Hill book comp. – 1996. – 720 p.
6. Tomovic, R., Sensitivity Analysis of Dynamic Systems, McGraw-Hill Book Co., New York, 1963.
7. Brayton, R. K., Spence, R., Sensitivity and Optimization, Elsevier, New York, 1980.
8. Frank, P. M., Introduction to System Sensitivity Theory, Academic Press, New York, 1978.
9. Kelley, H. J., "Method of Gradients", Optimization Techniques with Applications to Aerospace Systems, edited by George Leitmann, Academic Press, Orlando FL, 1962.
10. Schmit, L. A., Jr., "Structural Synthesis - Its Genesis and Development", AIAA Journal, v. 19, 1981, pp. 1249-1263.
11. McKeown, J. J., "Parametric Sensitivity Analysis of Nonlinear Programming Problems", Nonlinear Optimization - Theory and Algorithms, edited by L. C. W. Dixon, E. S. Spedicato and G. P. Szego, Birkhauser, Boston, 1980.
12. McKeown, J. J., "An Approach to Sensitivity Analysis", Numerical Optimization of Dynamic Systems, North Holland, edited by L. C. W. Dixon and G. P. Szego, 1980, pp. 349-362.
13. Vanderplaats, G. N., Yoshida, H., "Efficient Calculation of Optimum Design Sensitivity", AIAA Paper 84-0855, 1984.
14. Youngbok Kim. A study on the control system design for ship mooring winch system. - Journal of Mechanical Science and Technology 28 (3) (2014) 1065~1072 www.springerlink.com/content/1738-494x, DOI 10.1007/s12206-013-1181-9.
15. O. M. Amao and T. I. Fossen, Finite element modeling of mooring lines, Mathematics and Computers in Simulation, 53 (2004) 415-422.
16. P. I. Barth Berntsen, O. M. Amao and B. J. Leira, Ensuring mooring line integrity by dynamic positioning: Controller design and experimental tests, Automatica, 45 (2009) 1285-1290.
17. D. Lee, J. Kim and H. Kim, Design of an automatic winch system for small fishing vessel, Bulletin of Korean Society of Fisheries Technology, 36 (3) (2000) 157-166.
18. D. Lee, Dynamic characteristics of a hydraulic fishing winch simulator, Korean Journal of Fisheries and Aquatic Sciences, 37 (4) (2004) 330 - 336.
19. D. Lee and S. Chun, Characteristics of ship winches and their hydrostatic drives, Journal of the Korea Society of Fluid Power & Construction Equipment, 4 (2) (2007) 27 - 34.
20. Y. An, Driving characteristics improvement of SRM winch system using torque sharing function, Journal Of the Korean Society of Marine Engineering, 31 (4) (2007) 433-440.
21. J. Ha, D. Han and G. Han, The study for improve the braking power of mooring winch brake, Proceedings of 2008 Autumn Conference of KSME, (2008) 512-517.
22. S. Kostiucho and V. Tchaban, "Variational Method of Auxiliary Equations in Nonlinear Systems Analysis and Synthesis Problems," 2019 IEEE 20th International Conference on Computational Problems of Electrical Engineering (CPEE), Lviv-Slavske, Ukraine, 2019, pp. 1-5, doi: 10.1109/CPEE47179.2019.8949123.
23. S. Kostiucho, O. Kuzmych, A. Aitouche, S. Grinyuk and O. Mekush, "Application of Parametric Sensitivity Method to Analysis of Automatic Mooring Winch with Electric Drive System," 2019 4th Conference on Control and Fault Tolerant Systems (SysTol), Casablanca, Morocco, 2019, pp. 294-299, doi: 10.1109/SYSTOL.2019.8864751.

DOI: 10.36910/6775-2524-0560-2020-39-08

УДК 303.094.7:005.511 (083.92)

Кузьміна Олена Михайлівна, к.т.н., доцент

<https://orcid.org/0000-0002-0061-9933>

Яремко Світлана Анатоліївна, к.т.н., доцент

<http://orcid.org/0000-0002-0605-9324>

Вінницький торговельно-економічний інститут Київського національного торговельно-економічного університету, м.Вінниця, Україна

ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ УПРАВЛІННЯ БІЗНЕС-ПРОЦЕСАМИ НА ОСНОВІ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ

Кузьміна О.М., Яремко С.А. Підвищення ефективності управління бізнес-процесами на основі імітаційного моделювання. Стаття присвячена актуальним питанням впровадження імітаційного моделювання у діяльність компаній для підвищення ефективності бізнесу через ухвалення оптимальних управлінських рішень, уникнення ризиків та оцінки бізнесу. У сучасних умовах економіки та високої конкурентної боротьби управління бізнесом стає все більш аналітичним, яке не можливе без застосування сучасних інформаційних технологій. Ефективним інструментальним засобом підтримки керівника стають реінжиніринг бізнесу, імітаційне моделювання та засоби бізнес-аналітики.

Ключові слова: інформаційна технологія, інформаційна система, бізнес-інжиніринг, бізнес-процес, імітаційне моделювання, імітаційна модель.

Кузьмина Е.М., Яремко С.А. Повышение эффективности управления бизнес-процессами на основе имитационного моделирования. Статья посвящена актуальным вопросам внедрения имитационного моделирования в деятельность компаний для повышения эффективности бизнеса посредством принятия оптимальных управленческих решений, избежание рисков и оценки бизнеса. В современных условиях экономики и высокой конкурентной борьбы управления бизнесом становится все более аналитическим, которое невозможно без применения современных информационных технологий. Эффективным инструментальным средством поддержки руководителя становятся реинжиниринг бизнеса, имитационное моделирование и средства бизнес-аналитики.

Ключевые слова: информационная технология, информационная система, бизнес – инжиниринг, бизнес – процесс, имитационное моделирование, имитационная модель

Kuzmina O.M., Yaremko S.A. Improved business process management based on simulation modeling. The article deals with topical issues of implementation of simulation modeling in the activity of companies for improving business efficiency by making optimal management decisions, avoiding risks and evaluating business. In today's economic environment and highly competitive business management, it is becoming increasingly analytical, which is impossible without the use of modern information technologies. Business reengineering, simulation modeling and business intelligence tools are effective tools for supporting a leader.

Key words: information technology, information system, business engineering, business process, simulation, simulation model

Постановка наукової проблеми. В теперішній час найбільший інтерес до імітаційних моделей, в першу чергу, проявляють компанії та підприємства, що працюють в умовах високої конкуренції та мають чітку групу споживачів. Планування у компанії завжди пов'язане з майбутнім, а модель є уявленням очікуваної реальності. Таким чином, уявлення можливих майбутніх стратегій може розглядатися як моделювання майбутнього. Розвиток моделювання бізнес – процесів відбувається завдяки створенню моделей, здатних адекватно описувати реальність. Бурхливий розвиток інформаційних технологій та обчислювальної техніки надає фахівцям широкі можливості для створення все більш ефективних моделей економічної діяльності. Діяльність компанії розглядається як процес, який може бути спроектований, змодельований, і якщо необхідно, то перепроектований відповідно до інженерних принципів і обліку поставлених цілей. За допомогою інформаційних технологій імітаційного моделювання створюють фахівці, економісти, менеджерів найбільш сприятливі умови для вибору, аналізу та впровадження раціонального за фінансовими показниками варіанту. Управління компанією стає все більш аналітичним, а керівник компанії все більшою мірою стає аналітиком, працює з інформацією, і на її основі приймає управлінське рішення, тому застосування імітаційних моделей стає ефективним інструментальним засобом підтримки бізнесу. Ці технології дозволяють підвищити ефективність управлінських процесів, які позитивно впливають на фінансовий результат та конкурентоспроможність.

Аналіз досліджень. Проведений аналіз досліджень в галузі інформаційних технологій, що пов'язані зі імітаційним моделюванням показав, що цю проблематику вже розглядають з різних аспектів. Ситник В.Ф., Титаренко Г.А., Гужва В.М. досліджували теоретичні та методологічні загальні засади створення та використання інформаційних технологій на основі імітаційного моделювання. Інша група науковців Войнаренко М.П., Деміденко М.А., Орлов А.І., Шаров С.В., більш зосередилась на імітаційних методах та моделях. Процеси бізнес-аналітики та прогнозування

досліджували Дюк В. А., Норвіг П., Рассел С., Ханк Д. Не зважаючи на те, що інформаційні технології на основі імітаційного моделювання використовуються достатньо давно, питання їх впровадження в оптимізацію менеджменту і бізнес-процесів компаній та підприємств вимагають подальшого дослідження і є досить актуальними.

Формулювання мети дослідження. Метою статті є висвітлення особливостей застосування інформаційних технологій на основі імітаційних моделей для підвищення ефективності прийняття управлінських рішень в бізнесі, уникнення ризиків та оцінки бізнесу.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. При впровадженні нової інформаційної технології в компанії необхідно оцінити ризик відставання від конкурентів у результаті її неминучого старіння, тому що інформаційні продукти, як ніякі інші види матеріальних товарів, мають надзвичайно високу швидкість змінюваності новими видами або версіями. Періоди змінюваності коливаються від декількох місяців до одного року. Якщо в процесі впровадження нової інформаційної технології цьому фактору не приділяти належної уваги, можливо, що до моменту завершення переходу фірми на нову інформаційну технологію вона вже застаріє і прийдеться вживати заходів щодо її модернізації. Такі невдачі з впровадженням інформаційних технологій звичайно пов'язані з недосконалістю технічних засобів, тоді як основною причиною невдач є відсутність або слабка спрацьованість методології використання інформаційної технології.

Одне з головних завдань керівництва організації замовника і розробника - активне навчання майбутніх користувачів, підвищення рівня їх кваліфікації як користувачів, але насамперед як постановників. Користувач повинен бути заздалегідь ознайомлений з методикою проведення обстеження об'єкта, порядком узагальнення результатів, що допоможе йому визначити і виділити завдання та функції, що підлягають автоматизованій обробці і кваліфіковано зробити постановку завдання. Постановка завдання - опис завдання за певними правилами, яке дає вичерпне уявлення про сутність та логіку перетворення інформації для отримання результату. Користувач - фахівець у своїй галузі, він знає, чого він хоче. Але крім професійних знань в предметній області, користувач повинен мати знання інформаційних технологій для правильної постановки завдання. Це справедливо як для розробки інформаційної системи, так і для використання готових рішень.

План постановки завдання замовником інформаційної системи [1, 4]:

- організаційно - економічна сутність задачі - найменування, місце рішення, мета рішення, споживачі рішення і спосіб його доставки, періодичність рішення, джерела інформації, зв'язок з іншими завданнями;

- опис вхідної інформації - перелік вхідної інформації, форми подання, приклади документів, частота надходження інформації, форми контролю інформації і т.д.;

- опис вихідної інформації - перелік результатної інформації, форми подання, періодичність та строки подання, перелік користувачів результатної інформації, перелік запитної інформації, способи контролю результатної інформації і т.д.;

- опис алгоритму розв'язання задачі - опис способів формування результатної інформації, опис послідовності дій із змінною і умовно-постійною інформацією і т.п.;

- опис умовно-постійної інформації - перелік класифікаторів, довідників, таблиць, опис форми їх подання, способів використання умовно-постійної інформації тощо.

Одним з раціональних шляхів проектування інформаційної системи та інформаційної технології є використання типових проектних рішень, реалізованих у стандартних проектах, в пакетах прикладних програм. Можливість такого підходу пов'язана з наявністю у будь-якій організації загальних і унікальних рис. Використання спільності рис і завдань дозволяє прив'язати готові рішення (моделі і програми) до умов конкретного користувача і його завданням. Наприклад, більшість компаній вирішує типові завдання в бухгалтерському обліку, фінансах, організації управлінської праці, автоматизації документообігу, створенні інформаційно - довідкових систем, управлінні кадрами і т.п. У рамках таких завдань використання типових рішень буде виправданим і ефективним. Для створення інформаційної системи рекомендуються максимально використовувати стандартні пакети програм автоматизації бізнесу:

- інформаційні технології «клієнт-сервер» в корпоративному документообігу і ділових операціях;

- управління електронними документами;

- проектування, моделювання та аналіз складних інформаційних систем;

- фінансово-економічний аналіз діяльності;

- розробка систем підтримки прийняття рішень.

Можна виділити п'ять типових рівнів рішень, запропонованих на ринку комп'ютерних технологій:

1. Придбання окремих модулів програмно-апаратних засобів у вже сформованих каналах розповсюдження комп'ютерної техніки і самостійне побудова конфігурації необхідної інформаційної системи .

2. Звернення до підприємств - системним інтеграторам, що додає вартість рішень за надання кваліфікованих послуг. Придбання окремих модулів програмно - апаратних засобів та самостійна побудова інформаційної системи необхідної конфігурації.

3. Звернення до консалтингових компаній, які при створенні великих комплексних проектів, здійснюваних кількома виконавцями - системними інтеграторами - консультують виконання закінченого проекту, придбання і освоєння програмно-апаратних засобів, а також побудова інформаційної системи необхідної конфігурації. Відповідальність за проект несе компанія - системний інтегратор.

4. Компанія - системний інтегратор - не тільки створює систему, а й супроводжує її експлуатацію протягом узгодженого часу.

5. Виконання проектів системи і послуг з обслуговування апаратно-програмних засобів, подальшу модернізацію системи бере на себе спеціалізована компанія. Можливе знаходження апаратно-програмних засобів у власності спеціалізованої компанії, при цьому підприємство користується тільки інформацією.

Вимоги до розробника інформаційної системи. Велике значення має рівень і якість обслуговування, що надається розробником. Найкраще, коли замовник отримує від постачальника весь спектр послуг:

- постановка системи управління компанією (обстеження компанії з питань постановки обліку та документообігу, консалтингові послуги тощо);
- постачання і впровадження системи;
- «довічний» супровід системи (гарантійне та післягарантійне обслуговування, проведення тематичних семінарів як з проблем методології та організації обліку, так і з питань використання інформаційної системи).

Вибір фірми-розробника. Основні критерії вибору:

- час роботи на ринку фінансово-економічного програмного забезпечення;
- ліцензійна чистота програмного продукту;
- ліцензійна чистота засобів розробки;
- рівень реалізованих проектів;
- позиції фірми в рейтингах.

Досвід створення ІС і ІТ в управлінні економічними об'єктами показав, що ефективність функціонування компанії залежить не стільки від рівня автоматизації інформаційних процесів, скільки від аналітичної цілеспрямованості процедур самої управлінської діяльності та в обґрунтованості прийнятих фахівцями рішень. В основі цього лежить системно-технічний, інженерний підхід, який отримав назву бізнес-інжинірингу [1, 4].

Під бізнес-інжинірингом розуміється виконання комплексу, проектувальних робіт по розробці методів і процедур управління бізнесом, коли без зміни прийнятої структури управління в компанії (підприємстві, фірмі) досягається поліпшення її фінансового положення. Метою бізнес-інжинірингу є забезпечення фахівцеві найбільш сприятливих умов роботи при підвищенні прибутковості компанії (за рахунок досягнення ефективності виробництва), зниженні собівартості проєктованих робіт, скороченні внутрішніх витрат, підвищенні професійної підготовки, відповідальності, продуктивності праці персоналу, збільшенні об'єму продажів, надання широкого спектру послуг на ринку. При створенні ІС і ІТ бізнес-інжиніринг застосовується для вдосконалення управління виробничими і господарськими процесами. Бізнес-інжиніринг використовує системно-технічний і структурно-функціональний підходи в проєктуванні, які дозволяють результативніше використовувати переваги нових ІТ та людських ресурсів. Це веде до досягнення успіху і уникнення ризиків від управлінської діяльності.

Інжиніринг бізнесу ввів обов'язковий набір прийомів і методик, які компанія повинна використовувати для проєктування бізнесу відповідно до своїх цілей. Діяльність компанії розглядається як процес, який може бути спроектований, змодельований, і якщо необхідно, то перепроєктований відповідно до інженерних принципів і обліку поставлених цілей.

Реалізація інжинірингу будується на комп'ютеризації робочих місць фахівців і застосуванні ними для проєктування бізнесу ряду методик:

- 1) виділення покрокових процедур проектного процесу;
- 2) впровадження систем позначень, що описують процедури;
- 3) використання евристичних і прагматичних рішень, що дозволяють описувати ступінь відповідності спроектованого варіанту бізнесу заданим цілям.

Впровадження інжинірингу відкрило можливість об'єднувати в єдиний процес проектування впорядкування управлінських процедур, розробку нових інформаційних технологій і системи підтримки прийняття рішень.

ІС управління і ІТ за допомогою комп'ютерного моделювання створюють фахівцям, економістам, менеджерів найбільш сприятливі умови для вибору, аналізу та впровадження раціонального за фінансовими показниками варіанту. Тим самим досягається значне поліпшення організації управлінської діяльності і інформаційного обслуговування працівників управління, керівників відповідних підрозділів організації, прийняття ними обґрунтованих рішень.

Розвиток ринкових відносин, як за кордоном, так і в нашій країні примушує постійно переглядати технологію виконання виробничих і фінансових процесів, використовувати нові ІТ для підвищення якості продукції(послуг), шукати резерви підвищення ефективності, як правило, нетривіальними методами. Реальним стало створення і впровадження в повсякденну практику управлінської діяльності методики формування бізнес-процесів [2]. Бізнес-процес - це цілісний опис основних видів діяльності організації і їх проекція на організаційні структури економічного об'єкту, з урахуванням взаємодії між ними. Не тільки проектування, але і функціонування бізнес-процесів повністю залежать від використання фахівцями (економістами, менеджерами, керівниками середнього і верхнього рівнів управління організацією) досягнень в області нових ІТ (використання корпоративних ІС, Інтернет - технологій для активізації бізнесу та інші). Робота фахівців в середовищі автоматизованих ІТ дає великий економічний ефект. Створення такого складного організаційно-технологічного комплексу методичних рішень, направлено на кардинальне поліпшення управління бізнесом отримало назву реінжиніринг бізнес-процесів. Реінжиніринг бізнес-процесів – це розвиток методів інжинірингу, зокрема, системно-технічного і інформаційного підходів до проектування бізнес-процесів. Об'єктами вивчення і проектування реінжинірингу є бізнес-процеси, що протікають в організації. Основне завдання реінжинірингу — перепроєктування системи управління що діє, і створення на базі інтегрованої інформаційної системи нової технології управління бізнесом. Завдяки цьому повинні бути реалізовані поставлені цілі та отримані результати, що мають цінність для споживача, а для організації досягнутий бажаний економічний ефект — корінне поліпшення таких показників діяльності організації, як вартість, якість, послуги, темпи розвитку. Досягається це тим, що реінжиніринг передбачає заміну ієрархічного (структурного), строго функціонального принципу управління усередині організації на процесний (міжфункціональний), який повинен забезпечувати підвищення якості, вироблюваної продукції (послуг) за рахунок формування потоку завдань, що переходять від одного виконавця до іншого або від одного підрозділу до іншого. Під процесним підходом розуміється формалізація діяльності економічного об'єкту у вигляді бізнес-процесів. Системне управління підприємством спрямовано як на кожний бізнес-процес окремо, так і на взаємодію процесів між собою, забезпечуючи при цьому якість технологій виконання бізнес-процесів в рамках існуючої або переглянутої організаційної структури підприємства. Проект по реінжинірингу бізнесу, як правило, включає етапи: розробку зразка майбутньої організації; аналіз існуючого бізнесу; розробку нового бізнесу і впровадження нового бізнесу.

Таким чином, при проектуванні ІС нові ІТ повинні служити технологічною платформою реального реінжинірингу в організації.

При моделюванні бізнес-процесів, проектування потоків робіт ґрунтується на системному і інформаційному підходах. Побудова процесної моделі підприємства передбачає виділення на верхньому рівні наступних видів бізнес-процесів з подальшою їх декомпозицією:

- 1) основні бізнес-процеси (виробництво продукції, послуг);
- 2) допоміжні бізнес-процеси (забезпечення матеріальними, фінансовими, технічними, інформаційно-технологічними і іншими ресурсами);
- 3) бізнес-процеси управління підприємством (стратегічне, тактичне, оперативне управління).

Використовуючи декомпозицію досліджуваного процесу аналізуються і уточнюються чинники, що визначають якість виконання управління, формуються фундаментальні цілі функціонування організації, виявляються ключові чинники успіху, які необхідні і достатні для досягнення бажаних результатів. Для аналізу і проектування нової ІТ управління організацією застосовуються об'єктний і функціонально-технологічний методи, що дозволяють створити

процесно-орієнтований підхід, який відповідає вимогам досягнення поставлених перед організацією проблем та забезпечує реальні можливості інформаційного супроводу управлінських процесів [2].

Планування на підприємстві завжди пов'язане з майбутнім, а модель є уявленням очікуваної реальності. Таким чином, уявлення можливих майбутніх стратегій може розглядатися як моделювання майбутнього. Розвиток моделювання бізнес – процесів відбувається завдяки створенню моделей, здатних адекватно описувати реальність. Бурхливий розвиток інформаційних технологій та обчислювальної техніки надає фахівцям широкі можливості для створення все більш ефективних моделей економічної діяльності.

Необхідність враховувати вплив безлічі динамічно змінних у часі чинників обмежує застосування статичних методів, які можуть бути рекомендовані тільки для проведення грубих, попередніх розрахунків, з метою орієнтовної оцінки ефективності моделі. Моделі, де всі початкові дані та всі зв'язки точно відомі називають детермінованими. Часто буває так, що маючи статистичну інформацію, наприклад результати бухгалтерського обліку, ставиться завдання зробити оцінювання майбутнього за допомогою цих моделей. Майбутнє є процесом ризикованим з бізнесового погляду. При прийнятті ділових рішень ці ризики мають бути оцінені. Тому, виникає проблема прийняття рішень в умовах ризику, коли параметри і змінні задачі є випадковими величинами (наприклад, собівартість продукції, частку ринку, загальний збут у майбутніх періодах не можна визначити точно). Якщо це так, то чи буде ризикований захід прибутковим? У імітаційному моделюванні є три шляхи дослідження аспектів ризикованості [3]:

- перший шлях полягає у використанні сценарного підходу «Що..., Якщо...?». Це дає змогу дослідити альтернативні ситуації за допомогою модифікування моделі й визначити впливи змін. Такий підхід придатний для дослідження впливів змін однієї чи двох змінних або отримання специфічної відповіді, що базується на припущеннях вищого керівництва. Він не є найефективнішими для аналізу ризику;

- другий шлях — зробити оцінювання найкращого і найгіршого випадків. За цього підходу оцінки створюються з урахуванням сприятливіших і найнесприятливіших умов, які кожна вхідна змінна могла б мати. Для найкращого випадку встановлюють оптимістичні значення, а для найгіршого — песимістичні. Дослідження критичних ситуацій дуже корисне, але цей підхід також не виводить на сукупність ситуацій, які можуть реально очікуватися.

- третій шлях — використовувати імітаційне моделювання. Динамічні методи, засновані на імітаційному моделюванні найбільш ефективними. Серед яких найбільш застосованим є метод Монте-Карло.

Моделювання і імітація відрізняються тому що:

- моделювання — це аналітичне дослідження, яке включає формування ситуації у вигляді математичних відношень;

- імітація — експериментальне дослідження, оскільки воно включає виконання вибіркового машинних експериментів на моделі ситуації.

У вибіркового експерименті створюється великий ряд випробувань. Через наявність невпевненості наслідок кожного випробування може відрізнитися від наслідків інших випробувань. За імітації вибіркового експерименту виконуються на комп'ютерній моделі, даючи змогу в такий спосіб зробити багато випробувань з незначними матеріальними витратами (на відміну від проведення натурних експериментів).

Процедура імітації в короткому викладі складається з таких кроків:

1. Створення імітаційної моделі, яка визначає невпевненість і ризику.
2. Виконання вибіркового експериментів на комп'ютері, повторюючи розв'язання за моделлю велику кількість разів. Кожного разу отримуючи один можливий сценарій.
3. Проведення статистичних аналізів наслідків експериментів.
4. Використання одержаних статистичних результати для прийняття рішення.

Після дослідження результатів можна провести додаткові розв'язування задачі за моделлю, щоб краще підсилити статистичну точність результатів, або зупинитися на певному досягнутому результаті, прийнявши потрібні рішення, ґрунтуючись на отриманих результатах.

Імітаційні моделі діяльності економічного об'єкту, що побудовані за допомогою відповідних автоматизованих інформаційних систем, забезпечують генерацію стандартних бухгалтерських процедур і звітних фінансових документів та бізнес-операцій, що реалізуються в часі [1, 2, 4].

Модель бізнес-процесу — це опис загального виробничого процесу в термінах конкретної інформаційної системи.

Модель бізнес-функції являє собою функціональну ієрархічну декомпозицію бізнес-функцій. Відношення між такими бізнес-функціями утворюють бізнес-процеси. Бізнес-функції використовуються для досягнення конкретних цілей.

Модель бізнес-організації - це опис організаційної структури та структури персоналу об'єкту.

Під бізнес-операціями розуміють конкретні дії, що здійснюються в організації (на підприємстві) у процесі економічної діяльності, результатом яких є зміни в обсягах і напрямках руху грошових потоків. Ці моделі відображають реальну діяльність організації (підприємства) через опис грошових потоків (надходжень і виплат) як подій, що відбуваються в різні періоди часу. Під час розрахунків використовуються такі важко прогнозовані чинники, як показники інфляції, плановані обсяги збуту та інші, тому для розробки стратегічного плану й аналізу ефективності моделі застосовується сценарний підхід. Сценарний підхід передбачає здійснення альтернативних розрахунків на основі даних, що відповідають різним варіантам розвитку компанії.

Використання імітаційних фінансових моделей у процесі планування й аналізу ефективності економічної діяльності компанії дозволяє «програти» різні варіанти стратегій і в результаті прийняти обґрунтоване управлінське рішення, спрямоване на досягнення поставлених цілей.

Висновки та перспективи подальшого дослідження. Імітаційне моделювання відноситься до систем штучного інтелекту, який стає основним вектором розвитку технологій, розробок та інновацій у світі. Прогрес в обчислювальних потужностях, можливість збирати та обробляти величезні масиви інформації, а також миттєвий доступ до новітніх алгоритмів обробки даних є основою для еволюції інтелектуальних технологій. Розумні машини і додатки стали повсякденним явищем, допомагаючи нам приймати більш швидкі і точні рішення. Експерти прогнозують, що з кожним роком роль інтелектуальних систем буде стрімко рости, і вже у найближчі 5 років такі технології стануть ключовим гравцем на ринку людських асистентів. Згідно з аналітичними даними, тільки у 2019 році близько 35% організацій у всьому цивілізованому світі витратили понад 17% свого бюджету на впровадження інтелектуальних розробок. Це говорить про те наскільки необхідними вони є для створення конкурентного бізнесу та ефективних умов розвитку будь-якого напрямку діяльності в цілому. Деякі приклади застосування інтелектуальних систем у бізнес сфері:

- імітаційне моделювання може допомогти банкам, страховикам і інвесторам приймати більш розумні рішення, орієнтовані на основні потреби і запити клієнтів;
- імітаційне моделювання, аналізуючи активність користувачів, допоможе відстежувати їх поведінку і формувати моделі витрат, пропонуючи індивідуальні фінансові консультації.
- імітаційне моделювання забезпечує своєчасне реагування на ринкові тенденції, здійснюючи аналіз ринку.
- імітаційне моделювання здатне прораховувати ризики за допомогою аналізу кредитних балів, схем витрат, локальних і загальних фінансових даних, адаптуючи рекомендації як безпосередньо під сам сектор послуг, так і під кожного клієнта.
- імітаційне моделювання допомагає забезпечувати й утримувати конкурентоспроможність, дозволяючи компанії за допомогою правильних алгоритмів машинного навчання швидко реагувати на бізнес-аналітику, підвищувати продуктивність і відкривати нові потоки прибутків.

Список бібліографічного опису

1. Войнаренко М.П., Кузьміна О.М., Янчук Т.В. Інформаційні системи та технології в управлінні організацією: Навчальний посібник. Вінниця: «Едельвейс&Ко», 2015. 495с.
2. Гужва В.М. Інформаційні системи і технології на підприємствах: Навчальний посібник. Київ: КНЕУ, 2001. 400с.
3. Ситнік В.Ф. Системи підтримки прийняття рішень: Навчальний посібник. Київ: КНЕУ, 2004. 614с.
4. Титоренко Г.А. Информационные системы в экономике: Учебник для студентов вузов. Москва: ЮНИТИ-ДАНА, 2008. 463 с.

References

1. Voinarenko M.P., Kuzmina O.M., Yanchuk T.V. Informatsiini systemy ta tekhnologiyi v upravlinni organizatsieiu: navchalui posibnyk. Vinnytsia: «Edelweis&Ko», 2015.495s.
2. Huzhva V.M. Informatsiini systemy ta tekhnologiyi na pidpriemstvakh: navchalui posibnyk. Kyiv: KNEU, 2001. 400 s.
3. Sytnik V.F. Systemy pidtrymky pryinyattya rishen: navchalui posibnyk. Kyiv: KNEU, 2004. 614 s.
4. Tytorenko H.A. Informatsiini systemy v ekonomitsi: Posibnyk dlya studentiv vuziv. Moskva: YUNITI-DANA, 2008.463 s.

Рецензенти:

1. Корж Н.В., професор кафедри менеджменту та адміністрування Вінницького торговельно-економічного інституту КНТЕУ, доктор економічних наук, професор.
2. Войтко В.В., доцент кафедри програмного забезпечення Вінницького національного технічного університету, кандидат технічних наук, доцент.

DOI: 10.36910/6775-2524-0560-2020-39-09

УДК: 004:338.48

Лепкий Михайло Іванович, к.т.н., доцент

<https://orcid.org/0000-0003-2470-6780>

Подоляк Володимир Миколайович, к.т.н., доцент

<https://orcid.org/0000-0002-8521-1258>

Кошелюк Віктор Андрійович, к.т.н., асистент.

<https://orcid.org/0000-0002-4136-5087>

Луцький національний технічний університет

АПАРАТНЕ І ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СТВОРЕННЯ ТА ВИКОРИСТАННЯ 3D-ТУРІВ

Лепкий М.І., Подоляк В.М., Кошелюк В.А. Апаратне і програмне забезпечення створення та використання 3D-турів. Розглянуто особливості, механізми та інструменти віртуального 3D-туру, що виник завдяки сучасним інформаційно-комунікаційним технологіям та можливостям Інтернету. Висвітлено особливості апаратного і програмного забезпечення створення та використання 3D- турів. Наведено приклади розроблених 3D- турів для віртуальних подорожей та дано їх характеристику.

Ключові слова: віртуальний туризм, віртуальна подорож, віртуальний тур, 3D-панорама, 3D-тур.

Лепкий М.И., Подоляк В.М., Кошелюк В.А. Апаратное и программное обеспечение создания и использования 3D-туров. Рассмотрены особенности, механизмы и инструменты виртуального 3D-тура, Возникший благодаря современным информационно-коммуникационным технологиям и возможностям Интернета. Освещены особенности аппаратного и программного обеспечения создания и использования 3D- турів. Приведены примеры разработанных 3D- турів для виртуальных путешествий и дано их характеристику.

Ключевые слова: виртуальный туризм, виртуальное путешествие, виртуальный тур, 3D-панорама, 3D-тур.

Lepky M.I, Podoliak V.M, Koshelyuk V.A. Hardware and software for creating and using 3D blueprints Features, mechanisms and tools of the virtual 3D-tup, which arose due to modern information and communication technologies and Internet capabilities, are considered. Features of hardware and software for creating and using 3D-tupiv are highlighted. Examples of developed 3D-stages for virtual travels are given and their characteristic is given.

Keywords: virtual tourism, virtual trip, virtual tour, 3D-panorama, 3D-tour.

Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями. На теперішній час віртуальне середовище стає важливою ознакою сучасної реальності, так як надає можливість виходити за рамки нашого матеріального світу, а також сприяє подоланню просторових обмежень. Сучасний рівень інформаційно-комунікаційних технологій дозволив розширити можливості нових продуктів туристичної галузі, через розміщення їх в мережі Internet, додаючи до них можливість впровадження інтерактивних ефектів. Дані нововведення дозволили створювати інформаційні системи всередині продукту, які включають в себе різноманітний відеоматеріал, анімацію, звук, інформаційні вікна та меню, а також різноманітні спеціальні ефекти, наприклад відображення туристичного об'єкта залежно від падіння світла, що спостерігається. В свою чергу «віртуальний тур – спосіб реалістичного відображення тривимірного багатоелементного простору на плоскому екрані. Елементами віртуального туру, зазвичай, є сферичні панорами, з'єднані між собою інтерактивними посиланнями-переходами. Він створює в глядача «ефект присутності» – яскраві зорові образи, що запам'ятовуються» [3]. Все це надає унікальну можливість створювати програмні продукти по відомих містах, музеях, туристичних об'єктах з повним зануренням у віртуальну реальність.

Аналіз останніх досліджень і публікацій, в яких започатковано розв'язання даної проблеми. Питання дослідження використання новітніх інформаційних технологій в діяльності підприємств туристичної сфери і туризму в цілому проводили С. Арімов, В. Балута, П. Бурцев, О. Виноградова, А. Галиновский, О. Губанова, В. Гуляев, А. Демаш, М. Єфремова, І. Зорін, І. Калашников, В. Квартальнов, Д.Купінський, Ю. Миронов, М. Морозов, Н. Морозова, Т. Новгородцев, Н. Плотнікова, М. Огороков, М. Робсон, М. Скопень, Т. Ткаченко, О. Тоффлер Ф. Уллах, М. Хайдеггер, А. Шлевков, Л. Шульгіна, С. Чупров та інші. Проте дана тематика вимагає подальшого дослідження у зв'язку з високими темпами розвитку Інтернет-технологій та необхідністю їх застосування в туристичній індустрії.

Виділення невирішених раніше частин загальної проблеми, котрим присвячується означена стаття. Незважаючи на різноманітні наукові дослідження з даної тематики, питання застосування апаратного та програмного забезпечення сучасних ІКТ та можливостей Інтернету для туристичної галузь потребують подальшого вивчення.

Формулювання мети дослідження. Метою статті є розгляд апаратного та програмного забезпечення створення та використання 3D-турів, які виникли завдяки сучасним комп'ютерним технологіям, електронним засобам комунікацій та можливостям Інтернету.

Виклад основного матеріалу. На сьогоднішній день 3D-панорами і технології їх виготовлення починають тільки розвиватися, хоча панорамний рисунок існував вже довгий час. Це можна пояснити тим, що до теперішнього часу процес створення таких рисунків був достатньо важким. Коли виникли цифрові фотографії, розвиток 3D-панорам отримує новий поштовх для свого розвитку, а обробка одержаного матеріалу стала менш трудомісткою і простішою. Хоча істинна причина такого повсякденного використання є ріст і нові можливості сучасних комп'ютерів, які дозволяють створювати нові туристичні продукти в реальному часі.

Віртуальні тури – це он-лайнні або оф-лайнні презентації, які дозволяють потенційним клієнтам оглядати будь-який об'єкт. Залежно від типу такого туру це можуть бути сферичні або циліндричні панорами об'єктів будь-якого розміру (експонати музеїв та картинних галерей, кімнати та інші приміщення готелів, вулиці та будівлі міст, алеї парків, види з висоти пташиного польоту тощо), які рухаються довільно [10].

Віртуальний тур – це реалістичне тривимірне зображення, що складається з циліндричних або сферичних панорам, які зібрані з фотографій, тривимірних об'єктів і активних посилань-переходів (хотспотів). Більшість віртуальних турів дають можливість «рухатися» за допомогою інтерактивних навігаційних клавіш та вказівників їх переходу [2].

На сьогоднішній час 3D-тури являються рекламними продуктами, що дають можливість більш краще показувати туристу об'єкт або туристичну послугу, на противагу перегляду фотографій, відеороликів, читання описів, за рахунок чого створюється відносно більший ефект присутності. Основними ознаками 3D-турів є інтерактивність, показовість, інформативність та комунікативність.

Коли турист використовує 3D-тури, то він краще сприймає невідомий об'єкт, який стає більш реальнішим. Відмінною його ознакою є яскраві образи і незабутні враження від побаченого [12].

Панорама дозволяє представити людині простір, який подібний до особистої присутності людини – вона може обертатися довкола точки знімання в будь-яку сторону і розглядати будь-яку ділянку панорами, збільшуючи її [7].

Розглянемо основні способи подання панорами у вигляді 2D-зображення, що зберігається на комп'ютері: еквідистантна і кубічна проекції.

Побудова еквідистантної проекції являється складним процесом. Для цього потрібно розкласти сферу на окремі площини, при цьому розтягуючи зеніт і надир (тобто верх і низ) до отримання прямокутного зображення. Якщо ж помістити сферу з зображенням всередину куба зі стороною, що дорівнює діаметру сфери, спроектувати сферу на кожен бік куба і розрізати отриманий куб, то отримаємо шість проекцій: фронтальна, тилова, права, ліва, зеніт і надир сфери.

Після отримання проекцій 3D панорами її необхідно програмним чином перетворити назад для подання у вигляді сфери. Для цього існують спеціальні програми – панорамні плеєри. Саме на рівні візуалізації в панорамних плеєрах відбувається впровадження різноманітних інтерактивних ефектів.

Таким чином, весь процес створення 3D-туру зводиться до наступних дій:

- а) здійснення фотозйомки об'єктів, які увійдуть в віртуальний тур;
- б) обробка отриманих результатів, створення з ряду окремих фотознімків набору 3D-панорами;
- в) програмування ефектів інтерактивності і виготовлення віртуального туру [6].

Наведемо труднощі, які виникають на кожному з етапів виготовлення 3D-туру.

а) підготовка фото матеріалу (фотозйомка). Існують різні технічні обмеження на обладнання, використовуючи яке проводиться зйомка. Жоден з об'єктів не в змозі передати весь простір повністю, за один кадр. Кожен кадр може вмістити тільки частину простору. Для фотозйомки найкраще підходять ширококутні об'єктиви, які дозволяють відзняти сферу за мінімальну кількість кадрів. Однак у таких об'єктивів присутні сильні геометричні спотворення – дисторсія. В результаті спотворень подальша обробка матеріалу значно ускладнюється, а якість майбутньої панорами помітно знижується. Таким чином, доводиться шукати певний компроміс між простотою виготовлення і якістю результату [4].

Окрім проблеми з вибором фотоапарату і об'єктиву виникає також проблема динаміки зйомки: фотоапарат слід зафіксувати в процесі створення фотографій і обертати максимально близько до нодальної точки об'єктива.

Нодальна точка – це точка в об'єктиві камери, де перетинаються промені світла, що йдуть до плівки або матриці. Це дозволяє значно знизити негативний вплив паралакса між віднятими

кадрами. Паралакс – це зміщення об'єктів переднього плану щодо об'єктів заднього плану при повороті камери. Для обертання навколо нодальної точки зазвичай використовуються спеціальні панорамні головки [5].

Крім того, в зв'язку з тим, що з однієї точки робиться відразу велика кількість знімків, підвищуються вимоги до штатива, який повинен бути досить стійкий, мати гнучкі параметри налаштування зміни ракурсів фотоапарату без переміщення самого штатива.

б) Обробка фотоматеріалу. Оскільки після етапу підготовки фотоматеріалу ми маємо деяке число кадрів (кількість яких може варіюватися від 6 до 100 кадрів, в залежності від умов сюжетної програми), виникає проблема – яким чином скласти з них ціле, єдине зображення. При цьому необхідно врахувати спотворення простору на кожній з фотографій, що виникають при використуванні конкретних моделей об'єктивів. Після об'єднання ряду знімків в єдину фотографію потрібно обробити кольорову гамму і виправити дефекти, які неминуче виникають в процесі зйомки [6];

в) Додавання інтерактивності. Панорама відтворюється панорамними плеєрами, створеними на основі Java, QuickTime або Adobe Flash (ActionScript) технологій, дозволяючи використати всі їхні можливості для реалізації різних інтерактивних ефектів. Java і QuickTime технології вимагають завантаження спеціальних надбудов для браузера, які можуть досягати десятків мегабайт. Технологія Adobe Flash є стандартом і встановлена на комп'ютерах більшості користувачів, підтримує апаратне прискорення, тому саме вона отримує значну перевагу при реалізації інтерактивних 3D-турів;

г) Вибір програмного забезпечення для створення 3D-туру. Програми для побудови турів відрізняються дружнім, інтуїтивно зрозумілим інтерфейсом і зручністю роботи, а також забезпечують досягнення вражаючого результату за порівняно короткий проміжок часу (правда, останнє можливе лише за умови наявності ідеальних знімків, зшиваються в панораму). У підсумку на розробку програмного продукту витрачається мінімум часу, тоді як при використанні інших технологій для отримання того ж самого результату потрібен був би тиждень роботи цілої команди розробників [6].

Кафедрами «Професійної освіти та комп'ютерних технологій», «Комп'ютерних наук», «Туризму та готельно-ресторанної справи» Луцького НТУ розроблені різноманітні 3 D-тури - 3D-тур рекреаційний комплекс «Рест-парк», 3D-тур «Віртуальний Луцьк», 3D-тур «Центр традиційної культури «Медова хата» та інші.

Віртуальна екскурсія «Віртуальний Луцьк» розроблена за допомогою програм: PTGui, Microsoft Image Composite Editor, Tourweaver 5 Professional Edition. Для розробки віртуальних турів «Рекреаційний комплекс «РЕСТПАРК» та «Центр традиційної культури «Медова хата» використовувався комплекс програмних засобів, а саме: Kolor Autopano Giga, PanotourPro і Adobe Photoshop. Дані програмні продукти мають дружній, інтуїтивно зрозумілий інтерфейс, а також забезпечують досягнення вражаючого результату за порівняно короткий проміжок часу (правда, останнє можливо лише за умови наявності ідеальних знімків, які зшиваються в панораму). І як підсумок, на розробку програмного продукту витрачається мінімум часу, тоді як при використанні інших технологій для отримання того ж самого результату потрібно було б тиждень роботи цілої команди розробників.

Для редагування панорам використовувався програмний продукт Pano2VR – програма для створення сферичних і циліндричних 3D панорам, а також для створення на їх основі унікальних віртуальних турів зі всілякими ефектами і переходами. Вона дозволяє зберігати створені панорами в форматах Adobe Flash 10 (SWF), HTML5 або QuickTime.

Підтримується робота з графічними форматами JPEG, PNG, TIFF, PSD (Photoshop), Radiance HDR і файлами QuickTime VR з JPEG кодуванням. Є можливість створювати власні шаблони для майбутніх панорам, додавати в них кнопки, анімацію, звук і ефекти переходів [13].

Для створення панорамних фотографій застосовувався програмний продукт Kolor Autopano Giga 2.6.3 – програма для створення панорамних зображень, яку без перебільшення можна вважати грандіозним проривом в цій області протягом останніх років. Весь процес по створенню панорами повністю автоматизований: сама піджене фрагменти, відкоригує і збалансує колір і яскравість, може навіть сама знайти підходящі для склеювання фото у зазначеній папці. Підтримується велика кількість форматів (включаючи RAW).

Kolor Autopano Giga дозволяє об'єднувати декілька фотознімків з частковим перекриттям у великий панорамний знімок. На відміну від багатьох популярних програм, що застосовуються для склеювання панорам, тут склейка відбувається в повністю автоматичному режимі з використанням власних алгоритмів SIFT і RANSAC, що виробляють склеювання набагато швидше і точніше, повна підтримка HDR (High Dynamic Range). Робота з програмою дуже проста – достатньо лише вказати теку, що містить одну або декілька серій знімків для об'єднання, і програма сама знайде і об'єднає ці

знімки в панорами. При цьому, можливості практично необмежені – можна створювати панорами з полем зору 360 – 180 градусів, що містять необмежену кількість окремих фотографій, при цьому розмір панорами обмежений лише розмірами вашого вінчестера. Області застосування програми обширні: кінематографія, архітектура, продаж нерухомості, картографія, астрономія і т.д.

У доповненні до звичайних функцій Autorpano Pro, Giga-версія програми дозволяє створювати панорами в 360 градусів, експортувати їх в Flash формат в якості віртуального туру і створювати з сотень або тисяч фотографій гігапиксельні зображення [9].

Для створення 3D віртуального туру об'єкта використовувалась програма Kolor Panotour Pro 2.0 – програма, призначена для створення віртуальних турів з ваших фотографій. У кінцевому вигляді цей тур виглядає на екрані так, як ніби користувач знаходиться в центрі кімнати і якщо перемістити мишку вправо або вліво, то виходить як-ніби він сам крутиться по сторонах, якщо натиснути на двері, то користувач переміщується в наступну кімнату і т.д. У кінцевому підсумку виходить, що людина через комп'ютер зможе оглянути приміщення чи інший об'єкт. Таку технологію можна побачити на сайтах різних курортних готелів, кафе, офісів, ресторанів і т.д. Все що вимагається від розробника – це фотографії. Також в свій проект можна додавати наявні в програмі різноманітні ефекти [11].

Для редагування готових панорамних фотографій та усування різного роду дефектів, присутніх на панорамних зображеннях, використовувався графічний редактор Adobe Photoshop CS3. Adobe Photoshop – графічний редактор, розроблений і поширюваний фірмою Adobe Systems. Цей продукт є лідером ринку в області комерційних засобів редагування растрових зображень, і найвідомішим продуктом фірми Adobe.

Adobe Photoshop тісно пов'язаний з іншими програмами для обробки медіафайлів, анімації та іншої творчості. Спільно з такими програмами, як Adobe ImageReady, Adobe Illustrator, Adobe Premiere, Adobe After Effects і Adobe Encore DVD, він може використовуватися для створення професійних DVD, забезпечує засоби нелінійного монтажу і створення таких спецефектів, як фони, текстури і т. д. для телебачення, кінематографу і всесвітньої павутини. Основний формат Photoshop, PSD, може бути експортований і імпортований всіма програмними продуктами, переліченими вище.

Через високу популярність Adobe Photoshop підтримка його формату файлів, PSD, була реалізована в його основних конкурентів, таких, як Macromedia Fireworks, Corel, Pixel image editor, WinImages і т.д. Adobe Photoshop підтримує обробку зображень, як з традиційною глибиною кольору (8 біт, 256 градацій яскравості на канал), так і з підвищеною (16 біт, 65536 відтінків в кожному каналі) [14].

Для використання розроблених віртуальних турів підійде практично будь-який комп'ютер середньої потужності, що є непоганою перевагою, оскільки на даний момент деяка частина обладнання не є повністю оновлена, це стосується будь-якої організації, підприємства чи іншого об'єкту. Для успішного запуску і використання віртуальних 3D-турів також необхідно, щоб на комп'ютері був встановлений флеш-плеєр та будь-який браузер (бажано firefox, opera, chrome).

Запускаються віртуальні тури «Рекреаційний комплекс «РЕСТПАРК» та «Центр традиційної культури «Медова хата» через файл формату swf, що підтримується майже на всіх комп'ютерах.

На головній сторінці розміщуються активні точки переходу, що дозволяють переміститись по віртуальному туру до наступної панорами. Також на зручній позиції знаходиться панель управління віртуальним туrom.

Панель управління містить часто використовувані іконки віддалення/зближення, вперед/назад, вліво/вправо, вихідна позиція парами, повноекранний режим перегляду, кнопка запуску автоматичного обертання навколо осі.

Для управління віртуальним 3D-туром в меню було реалізовано кнопку допомоги «Help», яка розміщується першою з правої сторони. В будь-який момент можна відкрити вікно допомоги.

Процес розробки віртуальних турів включає декілька етапів:

- вибір кількості точок (панорам) у віртуальному турі;
- розробка індивідуального меню та додаткових інтерактивних функцій;
- фотозйомка об'єкту;
- монтаж 3D-панорам і віртуального туру;
- розміщення на сайті і тестування [1].

Висновки з даного дослідження. Віртуальне представлення об'єкта дозволяє максимально ознайомити потенційного клієнта із запропонованим туром, або його окремими складовими, викликає позитивні емоції та впливає на його вибір. Отже, для стимулювання покупок турів, пробудження бажання потенційних клієнтів відвідати певні місця, туристична фірма, музей або готельно-ресторанний комплекс обов'язково повинні використовувати віртуальний туризм у своїй

діяльності, формувати віртуальні тури, розміщувати панорамні зображення визначних місць та культурних пам'яток для залучення більшої кількості клієнтів.

Список бібліографічного опису.

1. Гадецька З.М. Сучасні мультимедійні засоби просування готельно-ресторанних та туристичних послуг. *Науковий журнал «Молодий вчений»*. Харків, 2014. №2 (17). С.45-51
2. Ветрова А.А., Макаренко А.Ю. Создание конструктора для разработки виртуальных 2D и 3D экскурсий [Электронный ресурс]. *Прикаспийский журнал : управление и высокие технологии*. 2008. № (2). Режим доступа до ресурсу : [http://hi-tech.asu.edu.ru/files/2\(2\)/91-95.pdf](http://hi-tech.asu.edu.ru/files/2(2)/91-95.pdf).
3. Виртуальный тур [Электронный ресурс]. *Art Of Web: сайт*. Режим доступа: <http://artofweb.ru/solutions/aow-businesscenter/>.
4. Дорофеев С.Ю., Тюгаев Д.Н. Создание аппаратно-программного комплекса для изготовления виртуальных туров на основе интерактивных 3D-панорам. *Инновационные технологии кафедры КСУП : Научно-практическая конференция*. Томск, 2008.
5. Дорофеев С.Ю., Тюгаев Д.Н. Интерактивные виртуальные 3D-Туры. *Научная сессия ТУСУР-2009: Матер. докладов Всеросс. научно-техн. конф. студентов, аспирантов и молодых учёных*. Томск : В-Спектр, 2009. С. 338-341.
6. Зайцева М.А., Лысак А.П., Дорофеев С.Ю. Технология создания виртуальных интерактивных туров RUBIUS 3DTourKit. *Известия Томского политехнического университета*, 2010. Т. 317. № 5. С. 97–102.
7. Кабак В.В., Подоляк В.М. Теоретичні засади використання інформаційних технологій у сфері туризму. Перспективи розвитку туризму в Україні та світі: управління, технології, моделі: колективна монографія. Видання п'яте : за наук. ред. проф. Матвійчук Л.Ю. Луцьк : ІВВ Луцького НТУ, 2019. С. 66-83.
8. Манько А.В., Орлик О.В. Інструменти та механізми віртуального туризму. *Інформаційні технології в економіці і управлінні. Збірник наукових студентських праць*. Одеса, 2019. С.119-125.
9. Несминова Н.Н. 3D панорамы: создание и преимущества. М., 2012. 125 с.
10. Орлик О.В. Інформаційні системи в сфері управління організацією. *Вісник соціально-економічних досліджень : зб. наук. пр.* Одеса : Вид-во ОДЕУ, 2002. Вип. 12. С. 188–191.
11. Панорамы и виртуальные туры [Электронный ресурс]. Режим доступа: <http://noinfo.ru/page.php?164-3d-panoramy-virtualnye-tury-kakoy-soft-vybrat>
12. Потапюк Л.М., Масовець О.А. Застосування технологій віртуальної реальності в освіті. *Сучасна наука та освіта Волині : зб. матеріалів науково-практ. конф., 22 листоп. 2018 р.*, м. Володимир-Волинський. Луцьк : Волиньполіграф, 2018. С. 338-343.
13. Создание интерактивных панорам и виртуальных туров [Электронный ресурс]. Режим доступа: <http://3dnews.ru/646669>
14. Создание 3D-панорамы [Электронный ресурс]. Режим доступа: <http://1panorama.ru/kak-sozdat-3d-panoramu-redaktirovanie-proekcij-sozdanie-3d-panoramy-11p>.

References.

1. Hadetska Z.M. Suchasni multymediini zasoby prosuvannia hotelno-restorannykh ta turystychnykh posluh. *Naukovyi zhurnal «Molodyi vchenyi»*. Kharkiv, 2014. №2 (17). S.45-51
2. Vetrova A.A., Makarenko A.Yu. Sozdanie konstruktora dlya razrabotki virtualnykh 2D i 3D ekskursiy [Elektronnyi resurs]. *Prikaspiyskiy zhurnal : upravlenie i vyisokie tehnologii*. 2008. # (2). Rezhim dostupu do resursu : [http://hi-tech.asu.edu.ru/files/2\(2\)/91-95.pdf](http://hi-tech.asu.edu.ru/files/2(2)/91-95.pdf).
3. Virtualnyi tur [Elektronnyi resurs]. *Art Of Web: sayt*. Rezhim dostupa: <http://artofweb.ru/solutions/aow-businesscenter/>.
4. Dorofeev S.Yu., Tyugaev D.N. Sozdanie aparatno-programmnogo kompleksa dlya izgotovleniya virtualnykh turov na osnove interaktivnykh 3D-panoram. *Innovatsionnye tehnologii kafedry KSUP : Nauchno-prakticheskaya konferentsiya*. Tomsk, 2008.
5. Dorofeev S.Yu., Tyugaev D.N. Interaktivnyie virtualnye 3D-Turyi. *Nauchnaya sessiya TUSUR-2009: Mater. докладов Vseross. nauchno-tehn. конф. studentov, aspirantov i molodykh uchYonyih*. Tomsk : V-Spekt, 2009. S. 338-341.
6. Zaytseva M.A., Lyisak A.P., Dorofeev S.Yu. Tehnologiya sozdaniya virtualnykh interaktivnykh turov RUBIUS 3DTourKit. *Izvestiya Tomskogo politehnicheskogo universiteta*, 2010. T. 317. # 5. S. 97–102.
7. Kabak V.V., Podoliak V.M. Teopetychni zasady vykopystannia infopmatsiinykh tekhnolohii u sfepi tupyzmu. Pepspektyvy pozvytku tupyzmu v Ukraini ta sviti: uppavlinnia, tekhnolohii, modeli: kolektyvna monohpafiia. Vydannia piate : za nauk. ped. pprof. Matviichuk L.Iu. Luts'k : IVV Lutskoho NTU, 2019. S. 66-83.
8. Manko A.V., Orlyk O.V. Instrumenty ta mekhanizmy virtualnoho turyzmu. *Informatsiini tekhnolohii v ekonomitsi i upravlinni. Zbirnyk naukovykh studentskykh prats*. Odesa, 2019. S.119-125.
9. Nesminova N.N. 3D panoramy: sozdanie i preimuschestva. M., 2012. 125 s.
10. Orlyk O.V. Informatsiini systemy v sferi upravlinnia orhanizatsiieiu. *Visnyk sotsialno-ekonomichnykh doslidzhen : zb. nauk. pr.* Odesa : Vyd-vo ODEU, 2002. Vyp. 12. S. 188–191.
11. Panoramy i virtualnye turyi [Elektronnyi resurs]. Rezhim dostupu: <http://noinfo.ru/page.php?164-3d-panoramy-virtualnye-tury-kakoy-soft-vybrat>
12. Potapiuk L.M., Masovets O.A. Zastosuvannia tekhnolohii virtualnoi realnosti v osviti. *Suchasna nauka ta osvita Volyni : zb. materialiv naukovoprakt. конф., 22 lystop. 2018 r.*, m. Volodymyr-Volynskiy. Luts'k : Volynpolihraf, 2018. S. 338-343.
13. Sozdanie interaktivnykh panoram i virtualnykh turov [Elektronnyi resurs]. Rezhim dostupu: <http://3dnews.ru/646669>
14. Sozdanie 3D-panoramy [Elektronnyi resurs]. Rezhim dostupu: <http://1panorama.ru/kak-sozdat-3d-panoramu-redaktirovanie-proekcij-sozdanie-3d-panoramy-11p>.

DOI: 10.36910/6775-2524-0560-2020-39-10

УДК 519.876.5; 621.31.33

Віктор Васильович Лишук, доцент

<https://orcid.org/0000-0003-4049-8467>

Микола Миколайович Євсюк, доцент

<https://orcid.org/0000-0002-3768-8959>

Йосип Романович Селепина, доцент

<https://orcid.org/0000-0002-2421-1844>

Назар Юрійович Копилець, студент

Луцький національний технічний університет

МАТЕМАТИЧНІ МОДЕЛІ ПРИСТРОЇВ ПЕРЕТВОРЮВАЛЬНОЇ ТЕХНІКИ

В.В. Лишук, М.М. Євсюк, Й.Р. Селепина, Н.Ю. Копилець. Математичні моделі пристроїв перетворювальної техніки. У статті запропоновано математичні моделі однофазних мостових випрямляч та інвертора, диференціальні рівняння яких записані в нормальній формі Коші. Показано, що такі моделі є найбільш ефективними для аналізу різноманітних режимів роботи. Розв'язок диференціальних рівнянь реалізується із застосуванням числових методів з допомогою комп'ютерної техніки.

Ключові слова: математична модель, випрямляч, однофазні випрямляч та інвертор, ключ, алгебраїчні та диференціальні рівняння.

В.В. Лышук, Н.Н. Евсюк, И.Р. Селепина, Н.Ю. Копилец. Математические модели устройств преобразовательной техники. В статье предложены математические модели однофазных мостовых выпрямителя и инвертора, дифференциальные уравнения которых записаны в нормальной форме Коши. Показано, что такие модели являются наиболее эффективными для анализа различных режимов работы. Решение дифференциальных уравнений реализуется с применением численных методов с помощью компьютерной техники.

Ключевые слова: математическая модель, однофазные выпрямитель и инвертор, ключ, алгебраические и дифференциальные уравнения.

V.V. Lyshuk, M.M. Yevsiuk, Y.R. Selepyna, N.Yu. Kopylets'. Mathematical models of converting equipment devices. The article proposes mathematical models of single-phase bridge rectifiers and inverters whose differential equations are written in the normal Cauchy form. Such models have been shown to be most effective for analyzing different operating modes. The solution of differential equations is implemented using numerical methods using computer technology.

Keywords: mathematical model, rectifier, single-phase rectifier and inverter, key, algebraic and differential equations.

Постановка наукової проблеми. На сьогодні задача випрямлення синусоїдного струму дуже часто зустрічається на практиці. Адже значна частина електроприймачів за своїм принципом роботи вимагає постійного струму. Це і електротранспорт, електротермічні установки, живлення автономного електроінструменту тощо. З іншого боку важливим завданням є також і показники якості (пульсації, коливання) вихідних параметрів, зокрема струмів та напруг.

Аналогічно зворотня задача інвертування електроенергії використовується так само широко, як і випрямлення. Якість інвертованої напруги визначається в основному двома видами модуляцій: амплітудно-імпульсній (АІМ) та широтно-імпульсній (ШІМ). З іншого боку впровадження інверторів пов'язане з розвитком відновлюваних джерел електроенергії, де виникає задача інвертування генерованої постійної напруги, що накопичується в акумуляторах, в змінну синусоїдну для живлення промислових приймачів, і приєднання паралельно до енергетичної системи [1, 2].

Тому створення математичних моделей таких пристроїв, які б максимально відображали фізичні процеси перетворення електроенергії з найменшими похибками і затратами комп'ютерного часу щодо симуляції набуває особливої актуальності.

Аналіз досліджень. Математичне моделювання є потужним інструментом при дослідженні та аналізі електротехнічних та електронних пристроїв. Диференціальні рівняння, які є основою побудови таких моделей, розв'язуються явними чи неявними числовими методами з допомогою персональних комп'ютерів. Це так звана комп'ютерна симуляція типових режимів їх роботи [3, 4].

Використовуючи загальну теорію диференціальних рівнянь в області електротехніки розроблені методи побудови математичних моделей електротехнічних пристроїв, які дадуть змогу, використовуючи один і той же математичний апарат, знайти розв'язки у часовій області різноманітних електротехнічних перехідних процесів [4].

Чисельні методи є потужним інструментом розв'язання диференціальних рівнянь і знаходять все ширше застосування при аналізі електродинамічних процесів. Інженери-електротехніки, які мають справу з різними електротехнічними пристроями, повинні вміти складати рівняння їх динаміки. Математичний апарат полягає в поданні диференціальних рівнянь електромагнітного стану

в нормальній формі Коші, що дасть змогу мінімізувати затрати комп'ютерного часу та підвищити точність результатів [3].

Аналізуючи розв'язок диференціальних рівнянь, а саме часові графічні залежності можна уникнути багатьох аварійних режимів. З іншого боку натурний експеримент може бути нереалізованим із-за дороговизни обладнання.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження.

Розглянемо методику побудови моделі на прикладі однофазного мостового інвертора, а саме формування диференціальних рівнянь. Залежно від параметрів електричної схеми можна отримати різні форми вихідного сигналу, зокрема й близькі до форми синусоїди. Подання роботи напівпровідникових вентилів за схемою ідеального ключа значно спрощує алгоритм аналізу та зменшує кількість обчислювальних операцій [2].

Однофазний випрямляч складається з однофазного понижувального трансформатора з насиченою магнітною системою, напівпровідникових діодів, С-фільтра та індуктивно-активного навантаження. Принципова схема такого пристрою зображена на рис.1.

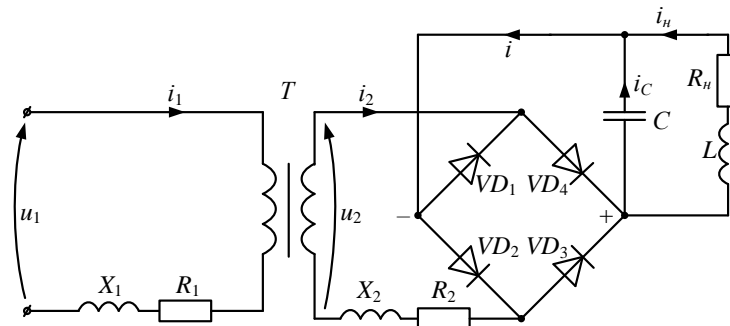


Рис.1. Електрична схема випрямляча.

Рівняння обмоток трансформатора запишемо за II законом Кірхгофа

$$e_k = \frac{d\psi_k}{dt} = u_k - i_k \cdot R_k, \tag{1}$$

$$i_k = \alpha_k (\psi_k - w_k \cdot \Phi), \quad k = 1, 2. \tag{2}$$

Тут $\alpha_k = \frac{1}{L_k} = \frac{\omega}{X_k}$ – обернена індуктивність розсіяння обмоток, Гн^{-1} , ψ_k , $w_k \cdot \Phi$ – основні та

робочі потокозчеплення обмоток відповідно, Вб.

При наявності згладжуючого конденсатора можливий режим, коли всі діоди закриті. Тому рівняння стану магнітопроводу запишемо так

$$w_1 i_1 + \chi w_2 i_2 = \rho' \Phi, \tag{3}$$

де χ – логічна змінна, яка приймає значення 0 або 1 (на проміжку відкритих діодів $\chi = 1$, на проміжку закритих діодів $\chi = 0$), ρ' – статичний магнітний опір осердя трансформатора, що визначається з кривої намагнічення сталі $B = f(H)$ або $V = f(\Phi)$.

Продиференціювавши (2), (3) за часом і використавши (1) отримаємо рівняння трансформатора в остаточному вигляді

$$\begin{aligned} \frac{d\Phi}{dt} &= \sum_{k=1}^2 d_k (u_k - i_k \cdot R_k); \\ \frac{di_2}{dt} &= \sum_{k=1}^2 a_k (u_k - i_k \cdot R_k). \end{aligned} \tag{4}$$

Тут введено позначення таких коефіцієнтів

$$d_1 = w_1 \cdot \alpha_1 \cdot \lambda; \quad d_2 = \chi \cdot w_2 \cdot \alpha_2 \cdot \lambda; \quad \lambda = \frac{1}{(\rho' + w_1^2 \alpha_1 + w_2^2 \alpha_2)}; \quad a_1 = -w_2 \cdot \alpha_2 \cdot d_1; \quad a_2 = \alpha_2 (1 - w_2 \cdot \alpha_2 \cdot d_2). \tag{5}$$

Інтегруючи (4), знаходимо магнітний потік і струм вторинної обмотки трансформатора. Струм первинної обмотки знаходимо згідно рівняння (3)

$$i_1 = \frac{(\rho' \Phi(\Phi) - \chi W_2 i_2)}{w_1}. \quad (6)$$

Напругу первинної обмотки трансформатора u_1 вважаємо заданою синусоїдною, а напругу u_2 , що є такою ж як на конденсаторі u_C або ЕРС вторинної обмотки на проміжку часу закритих діодів визначимо з рівняння

$$u_2 = u_C = e = w_2 \frac{d\Phi}{dt}. \quad (7)$$

На проміжку відкритих вентилів $\chi = 1$. Початок цього режиму триває доти, поки не виконається умова

$$u_C \pm w_2 \frac{d\Phi}{dt} \geq 0. \quad (8)$$

Знак «+» відповідає додатній півхвилі попереднього струму i_2 , «-» – від'ємній півхвилі. Рівняння конденсатора та навантаження мають вигляд

$$\frac{du_C}{dt} = \frac{i - i_n}{C}; \quad \frac{di_n}{dt} = \frac{u_C - iR_n}{L_n}. \quad (9)$$

Аналіз перехідних процесів однофазного випрямляча пов'язаний з інтегруванням диференціальних рівнянь (4), (9).

Розглянемо алгоритм розрахунку перехідних процесів у змодельованому однофазному мостовому випрямлячі.

1. Маючи поточне значення (на першому кроці початкове) значення потоку Φ обчислюємо статичний магнітний опір магнітопроводу трансформатора за характеристикою намагнічування відомої марки сталі.

2. Маючи поточні значення (на першому кроці початкові) значення напруги u_C , $i_n(t)$ $i_2(t)$ перевіряємо умову $i_2 = 0$. Якщо вона задовольняється, то присвоюємо $\chi = 0$, якщо ні, то $\chi = 1$.

3. За даними пунктів 1, 2 та за рівнянням (3) визначаємо струм i обчислюємо відповідно похідні в рівняннях (4), (9).

4. За даними пункту 3 перевіряємо умову відкриття діодів (7). Якщо вона задовольняється, то $\chi = 1$, в іншому випадку – присвоюємо попереднє значення.

5. Інтегруємо рівняння (4), (9) на наступному кроці і отримуємо приріст невідомих.

6. Інтегрування продовжуємо до тих пір, поки $t < t_{end}$.

Кількість змінних стану моделі в процесі інтегрування змінюється від комутації до комутації від чотирьох до трьох (Φ , U_C , i_n , i_2 при відкритих діодах та Φ , U_C , i_n при закритих діодах).

Перейдемо до формування математичної моделі інвертора. По суті процеси перетворення електроенергії тут є зворотніми, але замість некерованих діодів використано керовані від генератора імпульсів тиристри.

Однофазний інвертор складається з джерела живлення з дроселем, однофазного трансформатора з насиченою магнітною системою, напівпровідникових тиристорів, що керуються генератором імпульсів, комутуючого конденсатора C_k та індуктивно-активного навантаження.

Генератор імпульсів та комутуюча ємність задають частоту напруги на навантаженні. Принципова схема інвертора, що відповідатиме нашій моделі зображена на рис.2.

Рівняння обмоток трансформатора запишемо за II законом Кірхгофа [3]

$$\frac{d\psi_1}{dt} = u_C - i_1 \cdot R_1, \quad (10)$$

$$\frac{d\psi_2}{dt} = (-R_2 + R_n) i_2, \quad (11)$$

Струми обмоток знаходимо з рівняння (2).

Рівняння електромагнітного стану (намагнічувальних сил) запишемо подібно до (3)

$$w_1 i_1 + w_2 i_2 = F = \rho' \Phi. \quad (12)$$

Тиристор змодельовано ідеальним ключем [2], що змінюватиме структуру електричного кола. При цьому для кожної окремої комбінації відкритих і закритих тиристорів необхідно записати свою систему алгебро-диференціальних рівнянь. Це дещо створює складності при аналізі роботи таких пристроїв, але завжди можна звести множину систем таких алгебро-диференціальних рівнянь до однієї системи, вивши додаткові логічні змінні.

Робота такого інвертора визначається коливними процесами заряду і розряду комутуючого конденсатора. Відкриваючи тиристри T2, T4 струм протікає через конденсатор і первинну обмотку трансформатора, що в свою чергу створить струм на навантаженні.

Коли напруга на конденсаторі досягне максимального значення, тиристри T2, T4 закриються. Для того, щоб форма кривої струму навантаження була якомога ближчою до синусоїдальної, в цей момент необхідно відкрити тиристри T1, T3.

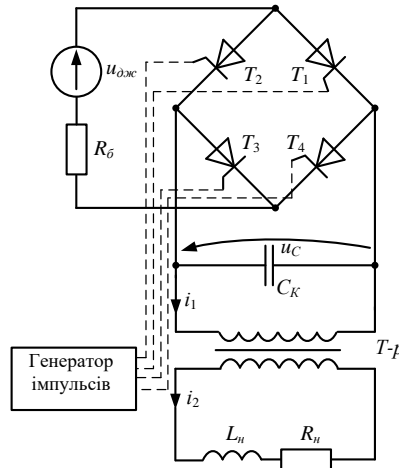


Рис.2. Електрична схема інвертора.

Рівняння напруги комутуючого конденсатора має вигляд

$$\frac{du_C}{dt} = \frac{\chi u - u_C - i_1}{C_K}, \quad (13)$$

де $u_{дж}$ – напруга джерела, χ – логічна константа, що приймає значення $\chi = \pm 1$, залежно від того, яка пара тиристорів відкрита.

Рівняння (5) дає змогу моделювати попарну роботу тиристорів T1, T3 в перший півперіод ($\chi = 1$) і T2, T4 в другий півперіод ($\chi = -1$).

Така модель можлива завдяки тому, що заряджений протягом півперіоду конденсатор при подачі від генератора імпульсів керуючого імпульсу напруги на тиристри T2, T4 закриває тиристри T1, T3.

Така робота мостової схеми еквівалентна миттєвій зміні полярності джерела напруги на початку наступного півперіоду відносно клем комутуючого конденсатора.

Отже, для змінної χ можемо записати умову

$$\chi = \begin{cases} 1, & \text{якщо } nT \leq t < nT + 0,5T, \\ -1, & \text{якщо } nT + 0,5T \leq t < (n+1)T, \end{cases} \quad (14)$$

де $n = 1, 2, 3, \dots$; T – період.

Диференціюючи за часом (3), (12) і розв'язуючи отриману систему рівнянь відносно невідомих згідно (10) отримуємо остаточно

$$\begin{aligned} \frac{d\Phi}{dt} &= \alpha_1 (u_C - i_1 \cdot R_1) + \alpha_{12} (R_n + R_1) i_2; \\ \frac{di_1}{dt} &= \alpha_{21} (u_C - i_1 \cdot R_1) + \alpha_{22} (R_n + R_1) i_2, \end{aligned} \quad (15)$$

де

$$a_1 = w_1 \alpha_1 \lambda; \quad a_{12} = -w_2 \alpha_2 \lambda; \quad \lambda = \frac{1}{(\rho' + w_1^2 \alpha_1 + w_2^2 \alpha_2)}; \quad a_{21} = \alpha_1 (1 - w_1 a_1); \quad a_{22} = -\alpha_1 w_1 a_{12}. \quad (16)$$

Тут $w_1, w_2, \alpha_1, \alpha_2, \lambda$ – відповідно числа витків, обернені індуктивності обмоток, причому α_2 включає в себе й індуктивність навантаження, λ – сумарна магнітна провідність.

Інтегруючи (15), знаходимо магнітний потік і струм вторинної обмотки трансформатора. Струм вторинної обмотки знаходимо згідно рівняння (3)

$$i_2 = \frac{(\rho' \Phi - w_1 i_1)}{w_2}. \quad (17)$$

Рівняння конденсатора та навантаження мають вигляд

$$i_C = \frac{(\chi u_{дж} - u_C)}{R_\sigma} - i_1. \quad (18)$$

Аналіз перехідних процесів однофазного інвертора пов'язаний з інтегруванням диференціальних рівнянь (13), (15).

І насамкінець, розглянемо алгоритм розрахунку перехідних процесів у інверторі.

1. Маючи поточне значення (на першому кроці початкове) значення потоку Φ обчислюємо статичний магнітний опір магнітопроводу трансформатора за характеристикою намагнічування відомої марки сталі $V = f(\Phi)$.

2. Маючи поточні значення (на першому кроці початкові) значення напруги $u_C, i_2(t)$ перевіряємо умову $i_2 = 0$. Якщо вона задовольняється, то присвоюємо $\chi = -1$, якщо ні, то $\chi = 1$.

3. За даними пунктів 1, 2 та за рівнянням (3) визначаємо струм і обчислюємо відповідно похідні в рівняннях (13), (15).

4. За даними пункту 3 перевіряємо умову відкриття тиристорів (14). Якщо вона задовольняється, то $\chi = 1$, в іншому випадку – присвоюємо попереднє значення.

5. Інтегруємо рівняння (13), (15) на наступному кроці і отримуємо приріст невідомих.

6. Інтегрування продовжуємо до тих пір, поки $t < t_{end}$.

Висновки та перспективи подальшого дослідження. Основною перевагою математичної моделі мостового інвертора є те, що диференціальні рівняння стану записані в нормальній формі Коші. Це максимально спрощує найважливіший етап дослідження – обчислювальний процес в процесі комп'ютерної симуляції. Саме завдяки цьому одержано практичну можливість аналізу перехідних процесів у електротехнічних пристроях, зокрема у однофазних мостових випрямлячах та інверторах. Інтегрування системи диференціальних рівнянь можна здійснити з допомогою комп'ютерних програм відомими явними методами, зокрема методами Ейлера або Рунге-Кутта.

Список бібліографічного опису

1. Волков И.В. Математические модели многоканальных инверторов с улучшенным спектральным составом выходного напряжения // И.В. Волков, Д.П. Каршенов // Техническая электродинамика. – 2015. – № 1. – С. 14-20.
2. Дзелендзяк У. Режими роботи однофазного тиристорного інвертора в системах керування / У. Дзелендзяк, В. Самогий, А. Павелчак // Вісник Національного університету "Львівська політехніка". – 2007. – № 598: Комп'ютерні науки та інформаційні технології. – С. 26-30.
3. Чабан В. Математичне моделювання електромеханічних процесів / В. Чабан. – Львів : Видавництво Державного університету „Львівська політехніка”, 1997. – 342 с.
4. Чабан В. Чисельні методи / В. Чабан. Львів: Видавництво Національного університету „Львівська політехніка”, 2001. – 186 с.

References

1. Volkov I.V. Mathematical models of multichannel inverters with improved spectral composition of output voltage // I.V. Volkov, D.P. Karshenov // Technical electrodynamics. – 2015. - № 1. - P. 14-20.
2. Dzelendziak U. Modes of operation of a single-phase thyristor inverter in control systems / U. Dzelendziak, V. Samothy, A. Pavelchak // Bulletin of the National university "Lviv Polytechnic". – 2007. – № 598: Computer sciences and information technologies. – С. 26-30.
3. Tchaban V. Mathematical modeling of electromechanical processes / V. Tchaban. – Lviv: Publisher of the State university "Lviv Polytechnic", 1997. – 342 p.
4. Tchaban V. Numerical methods / V. Tchaban. – Lviv: Publisher of the National university "Lviv Polytechnic". 2001. – 186 p.

DOI: 10.36910/6775-2524-0560-2020-39-11

УДК: 004.41: 616.12-073.96/97

Лупенко Сергій Анатолійович¹, д.т.н., професор<https://orcid.org/0000-0002-6559-0721>Литвиненко Ярослав Володимирович¹, д.т.н., доцент<https://orcid.org/0000-0001-7311-4103>Стадник Наталія Богданівна¹, асистент<https://orcid.org/0000-0002-7781-7663>Зозуля Андрій Миколайович¹, аспірант<https://orcid.org/0000-0003-1582-3088>¹ Тернопільський національний технічний університет імені Івана Пулюя, УкраїнаСверстюк Андрій Степанович², к.т.н., доцент<https://orcid.org/0000-0001-8644-0776>² Тернопільський національний медичний університет імені І.Я. Горбачевського.

УМОВНИЙ ЦИКЛІЧНИЙ ВИПАДКОВИЙ ПРОЦЕС ДИСКРЕТНОГО АРГУМЕНТУ ЯК УЗАГАЛЬНЕНА МАТЕМАТИЧНА МОДЕЛЬ ЦИКЛІЧНИХ СИГНАЛІВ ІЗ ПОДВІЙНОЮ СТОХАСТИЧНІСТЮ

Лупенко С.А., Литвиненко Я.В., Стадник Н.Б., Зозуля А.М., Сверстюк А.С. Умовний циклічний випадковий процес дискретного аргументу як узагальнена математична модель циклічних сигналів із подвійною стохастичністю. Робота присвячена розвитку математичного моделювання циклічних сигналів із подвійною стохастичністю, а саме, побудові їх математичної моделі у вигляді умовного циклічного випадкового процесу дискретного аргументу. Ця модель дає змогу несуперечливо врахувати стохастичність циклічних сигналів як при їх морфологічному статистичному аналізі, так і при статистичному аналізі їх ритму.

Ключові слова: циклічний сигнал, умовний циклічний випадковий процес, морфоаналіз, аналіз ритму.

Лупенко С.А., Литвиненко Я.В., Стадник Н.Б., Зозуля А.М., Сверстюк А.С. Условный циклический случайный процесс дискретного аргумента как обобщенная математическая модель циклических сигналов с двойной стохастичностью. Работа посвящена развитию математического моделирования циклических сигналов с двойной стохастичностью, а именно, построению их математической модели в виде условного циклического случайного процесса дискретного аргумента. Эта модель позволяет непротиворечиво учесть стохастичность циклических сигналов как при их морфологическом статистическом анализе, так и при статистическом анализе их ритма.

Ключевые слова: циклический сигнал, условный циклический случайный процесс, морфоанализ, анализ ритма.

Lupenko S.A., Lytvynenko Ia.V., Stadnyk N.B., Zozulia A.M., Sverstiuk A.S. Conditional cyclic random process of a discrete argument as a generalized mathematic model of cyclic signals with double stochasticity. The work is devoted to the development of mathematical modeling of cyclic signals with double stochasticity, namely, the construction of their mathematical model in the form of a conditional cyclic random process of a discrete argument. This model allows to take into account stochasticity of cyclic signals both in their morphological statistical analysis and in statistical analysis of their rhythm.

Keywords: cyclic signal, conditional cyclic random process, morphoanalysis, rhythm analysis.

Постановка наукової проблеми. Циклічні процеси та явища значно поширені у різних сферах дійсності, зокрема, вони мають місце в економіці, біології, медицині, техніці. Дослідження циклічних процесів із залученням сучасних інформаційних систем потребує попередньої розробки адекватних їх математичних моделей. Зважаючи на це, виникає необхідність розробки математичних моделей циклічних процесів, зокрема у вигляді умовного циклічного випадкового процесу дискретного аргументу. Дана модель дає змогу несуперечливо врахувати стохастичність циклічних сигналів, яка має засоби врахування подвійної стохастичності у його морфологічній та ритмічній структурах.

Аналіз досліджень. На сьогодні відомо багато різних математичних моделей циклічних процесів, серед яких гармонічна, періодична та майжеперіодична детерміновані функції, періодично корельований та періодично розподілений випадковий процес, лінійний періодичний випадковий процес, майже періодично корельований випадковий процес, циклічний випадковий процес [1-7]. Всі ці моделі циклічних сигналів не дають змогу одночасно врахувати подвійну стохастичність у їх структурі, а саме, стохастичності у морфологічній структурі сигналу та стохастичності у ритмічній структурі сигналу. Прикладами циклічних процесів та сигналів із подвійною стохастичністю можуть бути кардіосигнали електричної, магнітної та механічної природи, економічні циклічні процеси, процеси появи плям на Сонці, процеси розтріскування нанопокриття на поверхні матеріалів.

У роботі [8] розглянуто підходи до математичного моделювання та методів опрацювання сигналів серця на базі циклічних випадкових процесів та векторів з їх програмною реалізацією [9, 10]. Статистичний аналіз та обґрунтування математичної моделі досліджуваних сигналів проведений

в роботах [11-13]. На основі отриманих результатів у роботах [14-16] запропоновано інформативні ознаки циклічних біосигналів для побудови кардіодіагностичних систем.

У роботах [17, 18] започатковано розробку математичної моделі циклічного сигналу, яка має засоби врахування подвійної стохастичності у його структурі, а саме – стохастичності у морфологічній структурі сигналу та стохастичності у ритмічній структурі сигналу. Дану модель названо умовним циклічним випадковим процесом. Умовний циклічний випадковий процес враховує циклічність, стохастичність морфологічної та ритмічної структур досліджуваних процесів (сигналів).

Не зважаючи на отримані у роботах [8-16] результати, не вирішеними залишилося ряд завдань, зокрема, необхідно уточнити означення класу умовних циклічних випадкових процесів та означення випадкової функції ритму умовного циклічного випадкового процесу, а також уточнити суть підходу до проведення статистичного аналізу та аналізу ритму циклічних сигналів у рамках їх математичної моделі у вигляді умовного циклічного випадкового процесу. Також необхідно відзначити, що відсутнє означення умовного циклічного випадкового процесу дискретного аргументу, що унеможливило розробку адекватних статистичних методів аналізу цифрових сигналів.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Побудуємо умовний циклічний випадковий процес дискретного аргументу (умовного циклічного дискретного випадкового процесу). Для цього, згідно із роботами [17, 18], коротко розглянемо поняття циклічного випадкового процесу дискретного аргументу та введемо поняття класу ізоморфних циклічних випадкових процесів дискретного аргументу.

Означення 1. Областю визначення дискретного циклічного випадкового процесу є впорядкована дискретна $\mathbf{D} = \left\{ t_{ml} \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$ множина дійсних чисел, для елементів якої має місце такий тип лінійного упорядкування: $t_{m_1 l_1} < t_{m_2 l_2}$, якщо $m_2 < m_1$, або якщо $m_2 = m_1$, а $l_2 < l_1$, в інших випадках $t_{m_1 l_1} > t_{m_2 l_2}$ ($m_2, m_1 \in \mathbf{Z}, l_2, l_1 = \overline{1, L}, 0 < t_{m, l+1} - t_{m, l} < \infty$).

Означення 2. Дискретний випадковий процес $\xi(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D} \subset \mathbf{R}$, називається циклічним дискретним випадковим процесом, якщо існує така дискретна функція $T(t_{ml}, n)$, яка задовольняє умовам функції ритму, що скінченновимірні вектори $(\xi(\omega, t_{m_1 l_1}), \xi(\omega, t_{m_2 l_2}), \dots, \xi(\omega, t_{m_k l_k}))$ та $(\xi(\omega, t_{m_1 l_1} + T(t_{m_1 l_1}, n)), \xi(\omega, t_{m_2 l_2} + T(t_{m_2 l_2}, n)), \dots, \xi(\omega, t_{m_k l_k} + T(t_{m_k l_k}, n)))$, $n \in \mathbf{Z}$, при всіх цілих $k \in \mathbf{N}$ є стохастично еквівалентними у широкому розумінні.

Для дискретного циклічного випадкового процесу сімейство його функцій розподілу задовольняє наступним рівностям:

$$\begin{aligned} F_{k\xi}(x_1, \dots, x_k, t_{m_1 l_1}, \dots, t_{m_k l_k}) &= F_{k\xi}(x_1, \dots, x_k, t_{m_1 l_1} + T(t_{m_1 l_1}, n), \dots, t_{m_k l_k} + T(t_{m_k l_k}, n)) = \\ &= F_{k\xi}(x_1, \dots, x_k, y(t_{m_1 l_1}, n), \dots, y(t_{m_k l_k}, n)), x_1, \dots, x_k \in \mathbf{R}, t_{m_1 l_1}, \dots, t_{m_k l_k} \in \mathbf{D}, n \in \mathbf{Z}, k \in \mathbf{N}. \end{aligned} \quad (1)$$

Дамо означення ізоморфізму відносно порядку та значень циклічних випадкових процесів дискретного аргументу. Нехай маємо циклічні випадкові процеси $\xi_1(\omega, t), \omega \in \Omega, t_{ml} \in \mathbf{D}$ із функцією ритму $T_1(t_{ml}, n)$ та $\xi_2(\omega, t'), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ із функцією ритму $T_2(t'_{ml}, n)$. Области визначення $\mathbf{D} = \left\{ t_{ml} \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$ та $\mathbf{D}' = \left\{ t'_{ml} \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$ цих випадкових процесів у загальному випадку є різними ($\mathbf{D} \neq \mathbf{D}'$).

Означення 3. Циклічний випадковий процес $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ із функцією ритму $T_1(t_{ml}, n)$ та циклічний випадковий процес $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ із функцією ритму $T_2(t'_{ml}, n)$, будемо називати ізоморфними відносно порядку та значень або казати, що між цими циклічними випадковими процесами існує ізоморфізм відносно порядку та значень, якщо мають місце такі властивості:

1. Ізоморфізм відносно порядку між областями визначення циклічних випадкових процесів (ізоморфізм між упорядкованими числовими множинами \mathbf{D} та \mathbf{D}'), а саме:

1.а) має місце бієкція між \mathbf{D} та \mathbf{D}' (позначається так: $\mathbf{D} \leftrightarrow \mathbf{D}'$), тобто, будь-якому $t_{ml} \in \mathbf{D}$, відповідає лише одне $t'_{ml} \in \mathbf{D}'$ ($t_{ml} \rightarrow t'_{ml}$), а будь-якому $t'_{ml} \in \mathbf{D}'$ ставиться у відповідність лише одне $t_{ml} \in \mathbf{D}$ ($t'_{ml} \rightarrow t_{ml}$), причому для будь-яких різних $t_{m_1 l_1}, t_{m_2 l_2} \in \mathbf{D}$ їх образи $t'_{m_1 l_1}, t'_{m_2 l_2} \in \mathbf{D}'$ є різними, і навпаки (відповідні елементи $t_{ml} \in \mathbf{D}$ та $t'_{ml} \in \mathbf{D}'$ будемо називати бієктивно пов'язаними і позначати це так: $t_{ml} \leftrightarrow t'_{ml}$);

1.б) зберігається тип лінійного упорядкування множин \mathbf{D} та \mathbf{D}' , тобто, $\forall t_{m_1 l_1}, t_{m_2 l_2} \in \mathbf{D}$, $\exists t'_{m_1 l_1}, t'_{m_2 l_2} \in \mathbf{D}'$, що $t'_{m_1 l_1} \leftrightarrow t_{m_1 l_1}$, $t'_{m_2 l_2} \leftrightarrow t_{m_2 l_2}$ та має місце відношення порядку $t'_{m_2 l_2} > t'_{m_1 l_1}$, якщо $t_{m_2 l_2} > t_{m_1 l_1}$, і навпаки.

2. Ізоморфізм відносно порядку циклічних випадкових процесів $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$, а саме:

2.а) має місце бієкція між випадковими процесами $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ (позначається так: $\xi_1(\omega, t_{ml}) \leftrightarrow \xi_2(\omega, t'_{ml})$), тобто будь-якій парі $(t_{ml}, \xi_1(\omega, t_{ml}))$ із випадкового процесу $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$, відповідає лише одна пара $(t'_{ml}, \xi_2(\omega, t'_{ml}))$ $((t_{ml}, \xi_1(\omega, t_{ml})) \rightarrow (t'_{ml}, \xi_2(\omega, t'_{ml})))$ із випадкового процесу $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$, а будь-якій парі $(t'_{ml}, \xi_2(\omega, t'_{ml}))$ із випадкового процесу $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$, відповідає лише одна пара $(t_{ml}, \xi_1(\omega, t_{ml}))$ $((t'_{ml}, \xi_2(\omega, t'_{ml})) \rightarrow (t_{ml}, \xi_1(\omega, t_{ml})))$ із випадкового процесу $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$, причому для будь-яких різних $t_{m_1 l_1}, t_{m_2 l_2} \in \mathbf{D}$ їх образи $t'_{m_1 l_1}, t'_{m_2 l_2} \in \mathbf{D}'$ є різними, і навпаки (відповідні пари $(t_{ml}, \xi_1(\omega, t_{ml}))$ та $(t'_{ml}, \xi_2(\omega, t'_{ml}))$ будемо називати бієктивно пов'язаними і позначати це так: $(t_{ml}, \xi_1(\omega, t_{ml})) \leftrightarrow (t'_{ml}, \xi_2(\omega, t'_{ml}))$);

2.б) циклічні випадкові процеси $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ є упорядкованими за своїми областями визначення, причому порядкові типи випадкових процесів співпадають із порядковими типами їх областей визначення \mathbf{D} та \mathbf{D}' . Тобто, множина пар $\{(t_{ml}, \xi_1(\omega, t_{ml})), t_{ml} \in \mathbf{D}\}$, що формує (репрезентує) циклічний випадковий процес $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ є упорядкованою за параметром t_{ml} і має однаковий порядковий тип із числовою множиною \mathbf{D} , оскільки завжди існує бієктивне відображення області визначення \mathbf{D} на сам випадковий процес $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$, а саме елементу $t_{ml} \in \mathbf{D}$ ставиться у відповідність лише одна пара $(t_{ml}, \xi_1(\omega, t_{ml}))$, і навпаки, причому для двох різних $t_{m_1 l_1}, t_{m_2 l_2} \in \mathbf{D}$ відповідні їм пари $(t_{m_1 l_1}, \xi_1(\omega, t_{m_1 l_1}))$ та $(t_{m_2 l_2}, \xi_1(\omega, t_{m_2 l_2}))$ також різні. Те ж саме має місце і для випадкового процесу $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$, тобто лінійний порядок із області визначення \mathbf{D}' індукується у сам випадковий процес $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$;

2. в) має місце один і той же тип упорядкування циклічних випадкових процесів $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$, а саме: для будь-яких різних пар $(t_{m_1 l_1}, \xi_1(\omega, t_{m_1 l_1}))$ та $(t_{m_2 l_2}, \xi_1(\omega, t_{m_2 l_2}))$, що перебувають у бієктивній пов'язаності із парами $(t'_{m_1 l_1}, \xi_2(\omega, t'_{m_1 l_1}))$ та $(t'_{m_2 l_2}, \xi_2(\omega, t'_{m_2 l_2}))$ $((t_{m_1 l_1}, \xi_1(\omega, t_{m_1 l_1})) \leftrightarrow (t'_{m_1 l_1}, \xi_2(\omega, t'_{m_1 l_1})), (t_{m_2 l_2}, \xi_1(\omega, t_{m_2 l_2})) \leftrightarrow (t'_{m_2 l_2}, \xi_2(\omega, t'_{m_2 l_2})))$, мають місце відношення порядку $(t'_{m_2 l_2}, \xi_2(\omega, t'_{m_2 l_2})) > (t'_{m_1 l_1}, \xi_2(\omega, t'_{m_1 l_1}))$ та $(t_{m_2 l_2}, \xi_1(\omega, t_{m_2 l_2})) > (t_{m_1 l_1}, \xi_1(\omega, t_{m_1 l_1}))$, якщо $t'_{m_2 l_2} > t'_{m_1 l_1}$ та $t_{m_2 l_2} > t_{m_1 l_1}$ $(t'_{m_1 l_1} \leftrightarrow t_{m_1 l_1}, t'_{m_2 l_2} \leftrightarrow t_{m_2 l_2})$;

3. Має місце рівність значень циклічних випадкових процесів $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$, коли їх відповідні аргументи $t_{ml} \in \mathbf{D}$ та $t'_{ml} \in \mathbf{D}'$ перебувають у бієктивній пов'язаності $(t_{ml} \leftrightarrow t'_{ml})$, а саме, враховуючи властивість циклічності цих випадкових процесів, мають місце такі рівності:

$$\mathbf{P}\{\xi_1(\omega, t_{ml} + T_1(t_{ml}, n)) = \xi_2(\omega, t'_{ml} + T_2(t'_{ml}, n))\} = 1, \\ t_{ml} + T_1(t_{ml}, n) \leftrightarrow t'_{ml} + T_2(t'_{ml}, n), t_{ml} \in \mathbf{D}, t'_{ml} \in \mathbf{D}', n \in \mathbf{Z}. \quad (2)$$

Для ізоморфних відносно порядку та значень циклічних випадкових процесів $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ має місце рівність їх (k -вимірних) функцій розподілу $F_{k_{\xi_1}}(x_1, \dots, x_k, t_{m_1 l_1}, \dots, t_{m_k l_k})$ та $F_{k_{\xi_2}}(x_1, \dots, x_k, t'_{m_1 l_1}, \dots, t'_{m_k l_k})$, коли відповідні набори їх аргументів $t_i + T_1(t_i, n)$ та $t'_i + T_2(t'_i, n), i = \overline{1, k}, n \in \mathbf{Z}$ перебувають у бієктивній пов'язаності, а саме мають місце такі співвідношення:

$$F_{k_{\xi_1}}(x_1, \dots, x_k, t_{m_1 l_1} + T_1(t_{m_1 l_1}, n), \dots, t_{m_k l_k} + T_1(t_{m_k l_k}, n)) = \\ = F_{k_{\xi_2}}(x_1, \dots, x_k, t'_{m_1 l_1} + T_2(t'_{m_1 l_1}, n), \dots, t'_{m_k l_k} + T_2(t'_{m_k l_k}, n)), \\ x_i \in \mathbf{R}, t_{m_i l_i} \in \mathbf{D}, t'_{m_i l_i} \in \mathbf{D}', t_{m_i l_i} + T_1(t_{m_i l_i}, n) \leftrightarrow t'_{m_i l_i} + T_2(t'_{m_i l_i}, n), i = \overline{1, k}, k \in \mathbf{N}, n \in \mathbf{Z}. \quad (3)$$

Нехай маємо клас Θ усіх можливих циклічних випадкових процесів дискретного аргументу, що задані на ймовірносному просторі $(\Omega, \mathbf{F}, \mathbf{P})$ та на одній із множин класу $\mathbf{SetOf_D}$ усіх можливих

множин типу $\mathbf{D} = \left\{ t_{ml} \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$. На множині (класі) Θ задамо відношення еквівалентності $\varphi \subset \Theta^2$, а саме, бінарне відношення $\varphi(\xi_1(\omega, t_{ml}), \xi_2(\omega, t'_{ml}))$, де $\xi_1(\omega, t_{ml}), \xi_2(\omega, t'_{ml})$ - довільні дискретні циклічні випадкові процеси із Θ . Відношення еквівалентності $\varphi \subset \Theta^2$ може бути задане різним чином, зокрема, і шляхом постулювання ізоморфізму відносно порядку та значень між циклічними випадковими процесами із Θ .

Задання відношення еквівалентності $\varphi \subset \Theta^2$ на класі Θ породжує розбиття $\mathbf{D}_\Theta = \{\Theta_\beta, \beta \in \mathbf{B}\}$ класу Θ на підкласи ізоморфних відносно порядку та значень циклічних випадкових процесів дискретного аргументу, тобто для елементів розбиття $\mathbf{D}_\Theta = \{\Theta_\beta, \beta \in \mathbf{B}\}$, мають місце такі співвідношення:

$$\bigcup_{\beta \in \mathbf{B}} \Theta_\beta = \Theta, \Theta_\beta \neq \emptyset, \Theta_{\beta_1} \cap \Theta_{\beta_2} = \emptyset, \beta_1 \neq \beta_2, \beta_1, \beta_2 \in \mathbf{B}. \quad (4)$$

Виберемо один довільний елемент із розбиття $\mathbf{D}_\Theta = \{\Theta_\beta, \beta \in \mathbf{B}\}$, а саме, деякий клас еквівалентності (ізоморфності) $\Theta_\xi \in \mathbf{D}_\Theta$ циклічних випадкових процесів дискретного аргументу. До складу цього класу входять ізоморфні відносно порядку та значень дискретні циклічні випадкові процеси, які відрізняються один від одного лише своїми функціями ритму, а перехід від одного процесу до іншого можна забезпечити через дію відповідного оператора перетворення шкали. А саме, два довільних циклічних випадкових процесів $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ із Θ_ξ пов'язані через прямий $\mathbf{G}_{y_{12}}[\cdot]$ та обернений $\mathbf{G}_{y_{21}}[\cdot]$ оператори перетворення шкали, а саме:

$$\xi_2(\omega, t'_{ml}) = \mathbf{G}_{y_{12}}[\xi_1(\omega, t_{ml})], \omega \in \Omega, t_{ml} \in \mathbf{D}, t'_{ml} \in \mathbf{D}'. \quad (5)$$

$$\xi_1(\omega, t_{ml}) = \mathbf{G}_{y_{21}}[\xi_2(\omega, t'_{ml})], \omega \in \Omega, t_{ml} \in \mathbf{D}, t'_{ml} \in \mathbf{D}'. \quad (6)$$

Згідно робіт [17, 18], дія операторів перетворення шкали $\mathbf{G}_{y_{12}}[\cdot]$ та $\mathbf{G}_{y_{21}}[\cdot]$ на циклічні випадкові процеси $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ повністю визначається (задається) своїми функціями перетворення шкали $y_{12}(t_{ml}), t_{ml} \in \mathbf{D}$ та $y_{12}(t'_{ml}), t'_{ml} \in \mathbf{D}'$, які є зростаючими функціями, а саме так:

$$\xi_2(\omega, t'_{ml}) = \xi_2(\omega, y_{12}(t_{ml})) = \xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}, t'_{ml} \in \mathbf{D}'. \quad (7)$$

$$\xi_1(\omega, t_{ml}) = \xi_2(\omega, y_{21}(t'_{ml})) = \xi_2(\omega, t'_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}, t'_{ml} \in \mathbf{D}'. \quad (8)$$

Згідно робіт [17, 18], між функціями ритму $T_1(t_{ml}, n)$ та $T_2(t'_{ml}, n)$ циклічних випадкових процесів $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ із класу еквівалентності Θ_ξ , які пов'язані через оператори перетворення шкали $\mathbf{G}_{y_{12}}[\cdot]$ та $\mathbf{G}_{y_{21}}[\cdot]$ мають місце такі залежності:

$$T_2(y_{12}(t_{ml}), n) = y_{12}(t_{ml} + T_1(t_{ml}, n)) - y_{12}(t_{ml}), t_{ml} \in \mathbf{D}, n \in \mathbf{Z}. \quad (9)$$

$$T_1(y_{21}(t'_{ml}), n) = y_{21}(t'_{ml} + T_2(t'_{ml}, n)) - y_{21}(t'_{ml}), t'_{ml} \in \mathbf{D}', n \in \mathbf{Z}. \quad (10)$$

Статистичний аналіз будь-яких двох циклічних випадкових процесів $\xi_1(\omega, t_{ml}), \omega \in \Omega, t_{ml} \in \mathbf{D}$ та $\xi_2(\omega, t'_{ml}), \omega \in \Omega, t'_{ml} \in \mathbf{D}'$ із класу еквівалентності $\Theta_\xi \in \mathbf{D}_\Theta$ за їх будь-якими двома багаточисловими реалізаціями $\xi_{1\omega}(t_{ml})$ та $\xi_{2\omega}(t'_{ml})$ повинен давати близькі результати, а саме, подібні статистичні оцінки ймовірнісних характеристик циклічних сигналів. Даний факт лежить в основі забезпечення емпіричної узгодженості результатів статистичного аналізу циклічного сигналу (електрокардіосигналу, магнітокардіосигналу, фонокардіосигналу) за його різними реєстрограмами.

Для формальної ідентифікації конкретного дискретного циклічного випадкового процесу, клас Θ_ξ подамо як множину мічених параметром λ ізоморфних відносно порядку та значень циклічних випадкових процесів $\Theta_\xi = \left\{ \xi_\lambda(\omega, t_{ml}^\lambda), \omega \in \Omega, t_{ml}^\lambda \in \mathbf{D}_\lambda, \lambda \in \Lambda \right\}$, що задані на одному і тому ж ймовірнісному просторі $(\Omega, \mathbf{F}, \mathbf{P})$. Параметр λ набирає значень із деякої множини $\Lambda (\lambda \in \Lambda)$, потужність якої дорівнює потужності класу Θ_ξ . Будь-які, елементи із Θ_ξ відрізняються між собою лише дискретною областю їх визначення, і як наслідок, їх дискретними функціями ритму. Власне область визначення $\mathbf{D}_\lambda = \left\{ t_{ml}^\lambda \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$ будь-якого дискретного циклічного випадкового процесу

$\xi_\lambda(\omega, t_{ml}^\lambda)$, $\omega \in \Omega$, $t_{ml}^\lambda \in \mathbf{D}_\lambda$ із Θ_ξ однозначно його ідентифікує (вирізняє, маркує) з поміж інших випадкових процесів із Θ_ξ . Також, область визначення $\mathbf{D}_\lambda = \left\{ t_{ml}^\lambda \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$ конкретного λ -процесу $\xi_\lambda(\omega, t_{ml}^\lambda)$, $\omega \in \Omega$, $t_{ml}^\lambda \in \mathbf{D}_\lambda$ із Θ_ξ повністю визначає його функцію ритму $T_\lambda(t_{ml}^\lambda, n)$, а саме:

$$T_\lambda(t_{ml}^\lambda, n) = t_{m+n, l}^\lambda - t_{ml}^\lambda, m, n \in \mathbf{Z}, l = \overline{1, L}, L \geq 2, t_{ml}^\lambda \in \mathbf{D}_\lambda. \quad (11)$$

Таким чином, множину **SetOf_D** усіх можливих областей визначення ізоморфних відносно порядку та значень циклічних випадкових процесів дискретного аргументу із класу Θ_ξ також можна подати як параметричну множину, а саме:

$$\mathbf{SetOf_D} = \{ \mathbf{D}_\lambda, \lambda \in \Lambda \} = \left\{ \left\{ t_{ml}^\lambda \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}, \lambda \in \Lambda \right\}. \quad (12)$$

Враховуючи наведені вище міркування, має місце бієкція між класом еквівалентності $\Theta_\xi = \{ \xi_\lambda(\omega, t_{ml}^\lambda), \omega \in \Omega, t_{ml}^\lambda \in \mathbf{D}_\lambda, \lambda \in \Lambda \}$, множиною областей визначення **SetOf_D** та множиною функцій ритму $\mathbf{SetOf_T} = \{ T_\lambda(t_{ml}^\lambda, n), t \in \mathbf{R}, n \in \mathbf{Z}, \lambda \in \Lambda \}$ ізоморфних відносно порядку та значень дискретних циклічних випадкових процесів із Θ_ξ , а саме, $\Theta_\xi \Leftrightarrow \mathbf{SetOf_D} \Leftrightarrow \mathbf{SetOf_T}$. За таких взаємооднозначних відображень між класами функцій, бієктивно пов'язаними елементами, будуть ті елементи множин Θ_ξ , **SetOf_D** та **SetOf_T**, які мають однаковий параметр λ , а саме: $\xi_\lambda(\omega, t_{ml}^\lambda) \leftrightarrow \mathbf{D}_\lambda \leftrightarrow T_\lambda(t_{ml}^\lambda, n)$.

Наведені вище математичні об'єкти формально враховують той факт, що моделлю циклічного сигналу, наприклад, електрокардіосигналу є циклічний випадковий процес, а також відображають той експериментальний факт, що оцінки функцій ритму електрокардіосигналу за різними його реєстрограмами суттєво відрізняються, однак статистичні характеристик самого сигналу за різними його реєстрограмами є близькими. Проте такий математичний опис необхідно доповнити новим ймовірнісним простором, що уможливить опис ритму електрокардіосигналу як випадкової функції ритму в рамках теорії випадкових процесів. Такий опис дасть змогу усунути наявну суперечність між описами ритму та морфологічної структури в існуючих моделях циклічних сигналів та процесів.

Для цього розглянемо стохастичний експеримент, який описується деяким ймовірнісним простором $(\Omega', \mathbf{F}', \mathbf{P}')$, що є стохастично незалежним із $(\Omega, \mathbf{F}, \mathbf{P})$. Введемо випадковий об'єкт $\lambda(\omega') = \lambda$, $\omega' \in \Omega', \lambda \in \Lambda$ як вимірну функцію з областю визначення Ω' та областю значень Λ . У такому разі ω' -реалізацією випадкового об'єкту $\lambda(\omega')$ є параметр λ , який визначає відповідний йому циклічний випадковий процес $\xi_\lambda(\omega, t_{ml}^\lambda)$, $\omega \in \Omega$, $t_{ml}^\lambda \in \mathbf{D}_\lambda$ із областю визначення \mathbf{D}_λ та функцією ритму $T_\lambda(t_{ml}^\lambda, n)$. Тобто, можна ввести такі три випадкові об'єкти, які задані на ймовірнісному просторі $(\Omega', \mathbf{F}', \mathbf{P}')$, а саме: умовний циклічний випадковий процес дискретного аргументу $\xi(\omega, t_{ml}(\omega'))$, $\omega' \in \Omega', \omega \in \Omega, t_{ml}(\omega') \in \mathbf{D}(\omega')$; випадкову дискретну область його визначення $\mathbf{D}(\omega') = \left\{ t_{ml}(\omega') \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$, що задана на ймовірнісному просторі $(\Omega', \mathbf{F}', \mathbf{P}')$ та приймає значення із множини **SetOf_D**, а також випадкову функцію ритму $T(t_{ml}(\omega'), n)$ умовного циклічний випадковий процес дискретного аргументу $\xi(\omega, t_{ml}(\omega'))$, яка приймає свої значення (детерміновані функції ритму) із множини **SetOf_T**.

Дамо означення цим трьом ймовірнісним об'єктам. Спочатко дамо означення умовного циклічного випадкового процесу дискретного аргументу.

Означення 4. Випадковий об'єкт $\{ \xi(\omega, t_{ml}(\omega')) \in \Theta_\xi, \omega' \in \Omega', \omega \in \Omega, t_{ml}(\omega') \in \mathbf{D}(\omega') \}$, який задано на стохастично незалежних ймовірнісних просторах $(\Omega, \mathbf{F}, \mathbf{P})$ та $(\Omega', \mathbf{F}', \mathbf{P}')$ називається **умовним циклічним випадковим процесом дискретного аргументу**, якщо для кожної ω' , відповідна його ω' -реалізація $\{ \xi_{\omega'}(\omega, t_{ml}^{\omega'}) \}, \omega \in \Omega, t_{ml}^{\omega'} \in \mathbf{D}_{\omega'} \}$ належить класу Θ_ξ ізоморфних відносно порядку та значень циклічних випадкових процесів дискретного аргументу.

Означення 5. Випадкова функція $T(t_{ml}(\omega'), n)$, $\omega' \in \Omega', t_{ml}(\omega') \in \mathbf{R}, n \in \mathbf{Z}$, яка задана на ймовірнісному просторі $(\Omega', \mathbf{F}', \mathbf{P}')$ називається **випадковою функцією ритму умовного циклічного випадкового**

процесу дискретного аргументу, якщо для кожної ω' , відповідна її ω' -реалізація $T_{\omega'}(t_{ml}^{\omega'}, n)$, $t_{ml}^{\omega'} \in \mathbf{D}_{\omega'}$ належить класу **SetOf_T**, кожен елемент якого задовольняє умовам функції ритму, а саме: 1) групі умов: 1.a) $T_{\omega'}(t_{ml}^{\omega'}, n) > 0$, якщо $n > 0$ ($T_{\omega'}(t_{ml}^{\omega'}, 1) < \infty$); 1.b) $T_{\omega'}(t_{ml}^{\omega'}, n) = 0$, якщо $n = 0$; 1.c) $T_{\omega'}(t_{ml}^{\omega'}, n) < 0$, якщо $n < 0$, $t_{ml}^{\omega'} \in \mathbf{D}_{\omega'} \subset \mathbf{R}$; 2) для будь-яких $t_{m_1 l_1}^{\omega'} \in \mathbf{D}_{\omega'}$ та $t_{m_2 l_2}^{\omega'} \in \mathbf{D}_{\omega'}$, для яких $t_{m_1 l_1}^{\omega'} < t_{m_2 l_2}^{\omega'}$, для функції $T_{\omega'}(t_{ml}^{\omega'}, n)$ виконується строга нерівність $T_{\omega'}(t_{m_1 l_1}^{\omega'}, n) + t_{m_1 l_1}^{\omega'} < T_{\omega'}(t_{m_2 l_2}^{\omega'}, n) + t_{m_2 l_2}^{\omega'}$, $\forall n \in \mathbf{Z}$; 3) функція $T_{\omega'}(t_{ml}^{\omega'}, n) \in$ найменшою за модулем ($|T_{\omega'}(t_{ml}^{\omega'}, n)| \leq |T_{\omega'}^{\gamma}(t_{ml}^{\omega'}, n)|$) серед усіх таких функцій $\{T_{\omega'}^{\gamma}(t_{ml}^{\omega'}, n), \gamma \in \Gamma\}$, які задовольняють вище наведеним умовам 1 та 2.

Означення 6. Випадковий об'єкт $\mathbf{D}(\omega') = \left\{ t_{ml}(\omega') \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$, який задано на ймовірнісному просторі $(\Omega', \mathbf{F}', \mathbf{P}')$ називається випадковою областю визначення **умовного циклічного випадкового процесу дискретного аргументу**, якщо для кожної ω' , відповідна його ω' -реалізація $\mathbf{D}_{\omega'} = \left\{ t_{ml}^{\omega'} \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$ є дискретною підмножиною дійсних чисел, елементи якої задовольняють такі умови: $t_{m_1 l_1}^{\omega'} < t_{m_2 l_2}^{\omega'}$, якщо $m_2 < m_1$, або якщо $m_2 = m_1$, а $l_2 < l_1$, в інших випадках $t_{m_1 l_1}^{\omega'} > t_{m_2 l_2}^{\omega'}$ ($m_2, m_1 \in \mathbf{Z}, l_2, l_1 = \overline{1, L}, 0 < t_{m, l+1}^{\omega'} - t_{ml}^{\omega'} < \infty$).

Умовний циклічний випадковий процес $\{\xi(\omega, t_{ml}(\omega')), \omega' \in \Omega', \omega \in \Omega, t_{ml}(\omega') \in \mathbf{D}(\omega')\}$ дискретного аргументу дає змогу одночасного врахування як стохастичності сигналів, що враховується при статистичному морфологічному їх аналізі, так і стохастичності ритмічної структури досліджуваного сигналу, що враховується при проведенні аналізу ритму.

Морфологічний статистичний аналіз зводиться до статистичного аналізу будь-якої його ω' -реалізації $\{\xi_{\omega'}(\omega, t_{ml}^{\omega'}), \omega \in \Omega, t_{ml}^{\omega'} \in \mathbf{D}_{\omega'}\}$ як циклічного випадкового процесу із детермінованою функцією ритму $T_{\omega'}(t_{ml}^{\omega'}, n)$, $t_{ml}^{\omega'} \in \mathbf{D}_{\omega'}$ згідно із відомими методами статистичного опрацювання циклічних випадкових процесів [1-4]. Зокрема, такий стистичний аналіз зводиться до статистичного оцінювання вкладених у циклічний випадковий процес L стаціонарних та стаціонарно-пов'язаних випадкових процесів $\left\{ \varphi_l(\omega, t_{ml}^{\omega'}), \omega \in \Omega, t_{ml}^{\omega'} \in \mathbf{D}_{\omega'}, m \in \mathbf{Z}, l = \overline{1, L} \right\}$. Кожний такий випадковий стаціонарний процес заданий на дискретній множині $\mathbf{D}_{\omega'}^l = \left\{ t_{ml}^{\omega'} \in \mathbf{R}, m \in \mathbf{Z}, l = const \right\}$, яка є вкладеною в $\mathbf{D}_{\omega'}$ і описує (моделює) l -ту фазу досліджуваного електрокардіосигналу. Значення стаціонарного дискретного стаціонарного випадкового процесу $\varphi_l(\omega, t_{ml}^{\omega'}), \omega \in \Omega, t_{ml}^{\omega'} \in \mathbf{D}_{\omega'}, m \in \mathbf{Z}, l = const$ визначаються так:

$$\varphi_l(\omega, t_{ml}^{\omega'}) = \xi_{\omega'}(\omega, t_{ml}^{\omega'}), \omega \in \Omega, t_{ml}^{\omega'} \in \mathbf{D}_{\omega'}^l, m \in \mathbf{Z}, l = const. \quad (13)$$

Аналіз серцевого ритму зводиться до статистичного аналізу елементів випадкової області визначення $\mathbf{D}(\omega') = \left\{ t_{ml}(\omega') \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2 \right\}$ умовного циклічного випадкового процесу дискретного аргументу $\{\xi(\omega, t_{ml}(\omega')), \omega' \in \Omega', \omega \in \Omega, t_{ml}(\omega') \in \mathbf{D}(\omega')\}$, або до статистичного аналізу його випадкової функції ритму $T(t_{ml}(\omega'), n), \omega' \in \Omega', t_{ml}(\omega') \in \mathbf{R}, n \in \mathbf{Z}$. Випадкова функція ритму $T(t_{ml}(\omega'), n)$ повністю визначається елементами випадкової області $\mathbf{D}(\omega')$ згідно із формулою:

$$T(t_{ml}(\omega'), n) = t_{m+n, l}(\omega') - t_{m, l}(\omega'), m, n \in \mathbf{Z}, l = \overline{1, L}, t_{m, l}(\omega') \in \mathbf{D}(\omega'). \quad (14)$$

Зокрема, при $n = 1$, функція ритму $T(t_{ml}(\omega'), 1)$ обчислюється так:

$$T(t_{ml}(\omega'), 1) = t_{m+1, l}(\omega') - t_{m, l}(\omega'), m \in \mathbf{Z}, l = \overline{1, L}, t_{m, l}(\omega') \in \mathbf{D}(\omega'). \quad (15)$$

Якщо в аналізі серцевого ритму брати за основу випадкову функцію ритму $T(t_{ml}(\omega'), 1)$ умовного циклічного випадкового процесу $\xi(\omega, t_{ml}(\omega'))$, зберігаючи чітку прив'язку до фази серцевого циклу та номеру серцевого циклу, то математичну модель ритмокардіосигналу подамо як вектор випадкових послідовностей:

$$\mathbf{V}_L(\omega', m) = \left\{ \Delta T_l(\omega', m), \omega' \in \Omega', l = \overline{1, L}, m \in \mathbf{Z} \right\}, \quad (16)$$

де кожна l -та компонента вектора є випадковою послідовністю $\Delta T_l(\omega', m)$, значення якої дорівнює значенню випадкової функції ритму $T(t_{ml}(\omega'), 1)$ у моменти часу $t_{ml}(\omega')$ із дискретної множини $\mathbf{D}_l(\omega') = \{t_{ml}(\omega') \in \mathbf{R}, m \in \mathbf{Z}, l = \text{const}\}$, яка є вкладеною в $\mathbf{D}(\omega')$ і описує часові відстані між однотипними l -ми фазами досліджуваного електрокардіосигналу у двох його сусідніх циклах, а саме:

$$\Delta T_l(\omega', m) = T(t_{ml}(\omega'), 1) - t_{m,l}(\omega'), \quad (17)$$

$$m \in \mathbf{Z}, l = \overline{1, L}, t_{m,l}(\omega') \in \mathbf{D}(\omega').$$

У разі використання випадкової області $\mathbf{D}(\omega') = \{t_{ml}(\omega') \in \mathbf{R}, m \in \mathbf{Z}, l = \overline{1, L}, L \geq 2\}$ для аналізу серцевого ритму, перейдемо до такої матриці випадкових послідовностей:

$$\mathbf{M}(\omega', m) = \left\{ \Delta T_{l_1 l_2}(\omega', m), l_1 l_2 = \overline{1, L} \right\}, \quad (18)$$

де кожний елемент матриці $\mathbf{M}(\omega', m)$ є випадковою послідовністю $\Delta T_{l_1 l_2}(\omega', m)$, значення якої визначаються так:

$$\Delta T_{l_1 l_2}(\omega', m) = t_{m, l_2}(\omega') - t_{m, l_1}(\omega'), m \in \mathbf{Z}, l_1 l_2 = \overline{1, L}. \quad (19)$$

Елементи $\Delta T_{l_1 l_2}(\omega', m)$ матриці описують часові відстані (інтервали) між, загалом різнотипними, фазами l_1 та l_2 досліджуваного електрокардіосигналу у всіх його циклах, а саме, задають різниці між моментами часу $t_{m, l_1}(\omega')$ та $t_{m, l_2}(\omega')$ із дискретних множин $\mathbf{D}_{l_1}(\omega') = \{t_{ml_1}(\omega') \in \mathbf{R}, m \in \mathbf{Z}, l_1 = \text{const}\}$ та $\mathbf{D}_{l_2}(\omega') = \{t_{ml_2}(\omega') \in \mathbf{R}, m \in \mathbf{Z}, l_2 = \text{const}\}$, які є вкладеними в $\mathbf{D}(\omega')$ і стосується l_1 -ї та l_2 -ї фаз досліджуваного електрокардіосигналу.

Проаналізуємо детальніше властивості матриці $\mathbf{M}(\omega', m)$, а також її взаємозв'язок із вектором $\mathbf{V}_L(\omega', m)$. Перш за все, відзначимо, що множина $\left\{ \Delta T_{ll}(\omega', m), l = \overline{1, L} \right\}$ діагональних елементів матриці $\mathbf{M}(\omega', m)$ є множиною детермінованих послідовностей із нульовими значеннями, а саме:

$$\Delta T_{ll}(\omega', m) = t_{m, l}(\omega') - t_{m, l}(\omega') = 0, \forall m \in \mathbf{Z}, \forall l = \overline{1, L}, \forall \omega' \in \Omega'. \quad (20)$$

Зважаючи на це, діагональ матриці не несе жодної інформації про серцевий ритм. Також, легко бачити, що елементи $\Delta T_{l_1 l_2}(\omega', m)$ та $\Delta T_{l_2 l_1}(\omega', m)$ є протилежними за значеннями, а саме:

$$\Delta T_{l_1 l_2}(\omega', m) = -\Delta T_{l_2 l_1}(\omega', m), \forall m \in \mathbf{Z}, \forall l = \overline{1, L}, \forall \omega' \in \Omega', \quad (21)$$

причому, якщо $l_2 > l_1$, то $\Delta T_{l_1 l_2}(\omega', m) > 0$, а $\Delta T_{l_2 l_1}(\omega', m) < 0$, і навпаки, якщо $l_2 < l_1$, то $\Delta T_{l_1 l_2}(\omega', m) < 0$, а $\Delta T_{l_2 l_1}(\omega', m) > 0$. Зважаючи на такі залежності для елементів матриці $\mathbf{M}(\omega', m)$, для аналізу серцевого ритму слушно використовувати лише ті її елементи, які лежать над її головною діагоналлю, оскільки, лише вони є додатними та мають фізичну інтерпретацію часової тривалості між різними фазами електрокардіосигналу у різних його циклах. А всі елементи матриці $\mathbf{M}(\omega', m)$, що лежать під її діагоналлю, є від'ємними та за модулем дорівнюють відповідним елементам, що знаходяться над діагоналлю цієї матриці. Тобто, вся інформація про серцевий ритм сконцентрована у множині елементів над головною діагоналлю матриці $\mathbf{M}(\omega', m)$, а саме:

$$\left\{ \Delta T_{l_1 l_2}(\omega', m), l_1 = \overline{1, L}, l_2 = \overline{l_1 + 1, L} \right\}. \quad (22)$$

Оскільки як матриця $\mathbf{M}(\omega', m)$, так і вектор $\mathbf{V}_L(\omega', m)$ повністю визначаються випадковою дискретною множиною $\mathbf{D}(\omega')$, то між ними існує взаємооднозначний взаємозв'язок, а саме, будь-яка послідовність $\Delta T_{l_1 l_2}(\omega', m)$ із $\mathbf{M}(\omega', m)$, може бути визначена через елементи $\Delta T_l(\omega', m)$ із $\mathbf{V}_L(\omega', m)$. Тому математичною моделлю ритмокардіосигналу із підвищеною роздільною здатністю у практичних завданнях аналізу серцевого ритму може бути як модель структура $\left\{ \Delta T_{l_1 l_2}(\omega', m), l_1 = \overline{1, L}, l_2 = \overline{l_1 + 1, L} \right\}$, так і вектор $\mathbf{V}_L(\omega', m)$.

У медичній діагностичній практиці може виникнути ситуація, коли необхідно застосовувати гібридний підхід, а саме, при обґрунтованому виборі моделі ритмокардіосигналу із підвищеною роздільною здатністю слушно описувати ритм серця як за допомогою деякої підмножини елементів випадкової матриці $\mathbf{M}(\omega', m)$ (коли необхідно досліджувати тривалості між різнотипними фазами електрокардіосигналу у рамках кожного його циклу), так і за допомогою підмножини елементів випадкового вектора $\mathbf{V}_L(\omega', m)$ (коли необхідно досліджувати часові відстані між однотипними фазами електрокардіосигналу у двох його сусідніх циклах). Тому, у подальшому викладі матеріалу для математичного опису ритмокардіосигналу із підвищеною роздільною здатністю будемо використовувати вектор $\Xi_L(\omega', m) = \left\{ T_l(\omega', m), \omega' \in \Omega, l = \overline{1, L}, m \in \mathbf{Z} \right\}$, елементи якого є можуть бути як елементами із матриці $\mathbf{M}(\omega', m)$, так і елементи із вектора $\mathbf{V}_L(\omega', m)$. Розмірність (кількість компонент) L вектора $\Xi_L(\omega', m)$ визначає роздільну здатність ритмокардіосигналу та дорівнює кількості досліджуваних часових інтервалів між наперед виділеними фазами в електрокардіосигналі.

Модель ритмокардіосигналу у вигляді вектора $\Xi_L(\omega', m)$ є надто абстрактною, щоб на її основі можливо було розробляти конкретні методи статистичного аналізу ритмокардіосигналу із підвищеною роздільною здатністю. Така надлишкова абстрактність зумовлена невизначеністю ймовірнісної структури цього випадкового вектора, що є наслідком абстрактності ймовірнісного простору $(\Omega, \mathbf{F}, \mathbf{P})$, на якому він заданий, оскільки ймовірнісна міра \mathbf{P} може бути довільною. Уточнення ймовірнісної міри \mathbf{P} потребують окремих досліджень. Тому необхідно суттєво уточнити, конкретизувати цю модель з метою розробки статистичних методів аналізу ритмокардіограми із підвищеною роздільною здатністю на її основі.

Висновки та перспективи подальших досліджень. У даній роботі розвинуто у вигляді деталізації процедуру побудови класу умовних циклічних випадкових процесів. У роботі розроблено процедуру побудови математичної моделі циклічних цифрових сигналів у вигляді умовного циклічного випадкового процесу дискретного аргументу. Дано означення класу ізоморфних відносно порядку та значень циклічних випадкових процесів дискретного аргументу. Ця модель дає змогу несуперечливо врахувати стохастичність циклічних сигналів як при їх морфологічному статистичному аналізі, так і при статистичному аналізі їх ритму. Дано означення випадкової функції ритму умовного циклічного випадкового процесу дискретного аргументу. Розроблена математична модель циклічного сигналу у вигляді умовного циклічного випадкового процесу дискретного аргументу несуперечливо відображає його подвійну (морфологічну та ритмічну) стохастичність, та дає змогу проводити статистичний морфологічний аналіз та аналіз ритму в сучасних інформаційних системах цифрового опрацювання даних.

В подальших дослідженнях, на основі результатів робіт [19-21], необхідно розширити функціональні можливості застосування математичної моделі циклічних цифрових сигналів у вигляді умовного циклічного випадкового процесу дискретного аргументу, використовуючи Web-технології, забезпечивши при цьому проведення імітаційного моделювання досліджуваних сигналів із використанням їх нової моделі.

Список бібліографічного опису

1. Li J., Deng H. Vibration suppression of rotating long flexible mechanical arms based on harmonic input signals. *Journal of Sound and Vibration*. Vol. 436. 2018. P. 253-261.
2. Voss H.U. Hypersampling of pseudo-periodic signals by analytic phase projection. *Computers in Biology and Medicine*. Vol. 98. 2018. P. 159-167.
3. Medvedev A., Proskurnikov A., Zhusubaliyev Z. Mathematical modeling of endocrine regulation subject to circadian rhythm. *Annual Reviews in Control*. Vol. 46. 2018. P. 148-164.
4. McLachlan N.M., Grayden D.B. Enhancement of speech perception in noise by periodicity processing: A neurobiological model and signal processing algorithm. *Speech Communication*. Vol. 57. 2014. P. 114-125.
5. Javorskyj I., Kravets I., Matsko I., Yuzefovych R. Periodically correlated random processes: Application in early diagnostics of mechanical systems. *Mechanical Systems and Signal Processing*. Vol. 83. 2017. P. 406-438.
6. Javorskyj I., Yuzefovych R., Matsko I., Zakrzewski Z., Majewski J. Coherent covariance analysis of periodically correlated random processes for unknown non-stationarity period. *Digital Signal Processing*. Vol. 65. 2017. P. 27-51.
7. Naseri H., Homaeinezhad M., Pourkhajeh H. Noise/spike detection in phonocardiogram signal as a cyclic random process with non-stationary period interval. *Computers in Biology and Medicine*. Vol. 43(9). 2013. P. 1205-1213.
8. Лупенко С., Зозуля А., Сверстюк А., Стадник Н. Математичне моделювання та методи опрацювання сигналів серця на базі циклічних випадкових процесів та векторів. *Sciences and Education a New Dimension. Natural and Technical Sciences*. Vol. VI(20). ISSUE 172. 2018. Budapest. P. 47-54.
9. Лупенко С.А., Луцук Н.С., Литвиненко Я.В., Зозуля А.М. Програмний комплекс для морфологічного аналізу та аналізу серцевого ритму з підвищеною інформативністю. *Інформаційні технології та комп'ютерна інженерія*. № 1. ВНТУ. 2016. С. 13-22.

10. Литвиненко Я.В., Лупенко С.А., Сверстюк А.С. Програмний комплекс для обробки та моделювання синхронно зареєстрованих кардіосигналів з використанням моделей та методів теорії циклічних функціональних відношень. Вісник Хмельницького національного університету. 2009. №5. С.80-87.
11. Lupenko S., Lutsyk N., Yasnii O., Sobaszek Ł. Statistical analysis of human heart with increased informativeness. Acta mechanica et automatica. Vol. 12. 2018. P. 311–315.
12. Лупенко С.А., Литвиненко Я.В., Сверстюк А.С. Статистичний сумісний аналіз кардіосигналів на основі вектора циклічних ритмічно пов'язаних випадкових процесів. Електроніка та системи управління. Національний авіаційний університет. 2008. № 4 (18). С. 22-29.
13. Сверстюк А. С. Обґрунтування та верифікація математичної моделі синхронно зареєстрованих кардіосигналів з використанням вектора циклічних ритмічно пов'язаних випадкових процесів. Вимірювальна та обчислювальна техніка в технологічних процесах. 2009. № 1. С. 143-147.
14. Lupenko S., Lutsyk N., Yasnii O., Zozulia A. The Modeling and diagnostic features in the computer systems of the heart rhythm analysis with the increased informativeness. 9th International Conference on Advanced Computer Information Technologies (ACIT) June 5-7 in Ceske Budejovice, Czech Republic. 2019. P. 121-124.
15. Lupenko S., Lutsyk N., Yasnii O., Zozulia A. The Modeling and Diagnostic Features in the Computer Systems of the Heart Rhythm Analysis with the Increased Informativeness. 2019 9th International Conference on Advanced Computer Information Technologies (ACIT). IEEE, 2019. P. 121-124.
16. Martsenyuk V. P., Sverstiuk A. S., Klos-Witkowska A., Horkunenko A. B., Rajba S. Vector of Diagnostic Features in the Form of Decomposition Coefficients of Statistical Estimates Using a Cyclic Random Process Model of Cardiosignals. The 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 18-21 September. Metz. 2019. Vol. 1. P. 298-303.
17. Лупенко С.А., Луцук Н.С., Стадник Н.Б. Модель із подвійною стохастичністю у задачах математичного моделювання та аналізу циклічних процесів та сигналів. Матеріали IV науково-технічної конференції „Інформаційні моделі, системи та технології“, 15-16 травня 2014 року, Тернопіль. С. 10.
18. Лупенко С., Сверстюк А., Луцук Н., Стадник Н., Зозуля А. Умовний циклічний випадковий процес як математична модель коливних сигналів та процесів із подвійною стохастичністю. Поліграфія і видавнича справа. No.1 (71). 2016. С. 147-159.
19. Марценюк В.П., Семенець А.В., Сверстюк А.С. Концептуальні підходи до інтегрованого середовища проведення наукових медико-біологічних досліджень. Штучний інтелект. 2003. №2. С. 35-44.
20. Марценюк В.П., Кравець Н.О., Сверстюк А.С. Інформаційна система медико-біологічних досліджень: проект на основі Web-технологій. Український журнал телемедицини та медичної телематики. 2003. Т.1, № 1. С. 57-60.
21. Лупенко С.А., Дем'ячук Н.Р., Сверстюк А.С. Концептуально-методологічні основи імітаційного моделювання циклічних сигналів на ЕОМ із використанням їх моделі у вигляді циклічного функціонального відношення. Вимірювальна та обчислювальна техніка в технологічних процесах. 2008. № 4. С. 101–111.

References

1. Li J., Deng H. Vibration suppression of rotating long flexible mechanical arms based on harmonic input signals. Journal of Sound and Vibration. Vol. 436. 2018. P. 253-261.
2. Voss H.U. Hypersampling of pseudo-periodic signals by analytic phase projection. Computers in Biology and Medicine. Vol. 98. 2018. P. 159-167.
3. Medvedev A., Proskurnikov A., Zhusubaliyev Z. Mathematical modeling of endocrine regulation subject to circadian rhythm. Annual Reviews in Control. Vol. 46. 2018. P. 148-164.
4. McLachlan N.M., Grayden D.B. Enhancement of speech perception in noise by periodicity processing: A neurobiological model and signal processing algorithm. Speech Communication. Vol. 57. 2014. P. 114-125.
5. Javorskyj I., Kravets I., Matsko I., Yuzefovych R. Periodically correlated random processes: Application in early diagnostics of mechanical systems. Mechanical Systems and Signal Processing. Vol. 83. 2017. P. 406-438.
6. Javorskyj I., Yuzefovych R., Matsko I., Zakrzewski Z., Majewski J. Coherent covariance analysis of periodically correlated random processes for unknown non-stationarity period. Digital Signal Processing. Vol. 65. 2017. P. 27-51.
7. Naseri H., Homaeinezhad M., Pourkhajeh H. Noise/spike detection in phonocardiogram signal as a cyclic random process with non-stationary period interval. Computers in Biology and Medicine. Vol. 43(9). 2013. P. 1205-1213.
8. Lupenko S., Zozulia A., Sverstiuk A., Stadyk N. Matematychnе modelivannia ta metody opratsiuvannia syhnaliv sertsia na bazi tsyklichnykh vypadkovykh protsesiv ta vektoriv. Sciences and Education a New Dimension. Natural and Technical Sciences. Vol. VI(20). ISSUE 172. 2018. Budapest. P. 47-54.
9. Lupenko S.A., Lutsyk N.S., Lytvynenko Ya.V., Zozulia A.M. Prohramnyi kompleks dlia morfolohichnoho analizu ta analizu sertsevoho rytmu z pidvyshchenoiu informatyvniuiu. Informatsiini tekhnolohii ta kompiuterna inzheneriia. № 1. VNTU. 2016. S. 13-22.
10. Lytvynenko Ya.V., Lupenko S.A., Sverstiuk A.S. Prohramnyi kompleks dlia obrobky ta modelivannia synkhronno zareiestrovanykh kardiosyhnaliv z vykorystanniam modelei ta metodiv teorii tsyklichnykh funktsionalnykh vidnoshen. Visnyk Khmelnytskoho natsionalnoho universytetu. 2009. №5. S.80-87.
11. Lupenko S., Lutsyk N., Yasnii O., Sobaszek Ł. Statistical analysis of human heart with increased informativeness. Acta mechanica et automatica. Vol. 12. 2018. P. 311–315.
12. Lupenko S.A., Lytvynenko Ya.V., Sverstiuk A.S. Statystychnyi sumisnyi analiz kardiosyhnaliv na osnovi vektora tsyklichnykh rytmichno poviazanykh vypadkovykh protsesiv. Elektronika ta systemy upravlinnia. Natsionalnyi aviatsiynyi universytet. 2008. № 4 (18). S. 22-29.
13. Sverstiuk A. S. Obhruntuvannia ta verifyfikatsiia matematychnoi modeli synkhronno zareiestrovanykh kardiosyhnaliv z vykorystanniam vektora tsyklichnykh rytmichno poviazanykh vypadkovykh protsesiv. Vymiriuvalna ta obchysliuvalna tekhnika v tekhnolohichnykh protsesakh. 2009. № 1. S. 143-147.
14. Lupenko S., Lutsyk N., Yasnii O., Zozulia A. The Modeling and diagnostic features in the computer systems of the heart rhythm analysis with the increased informativeness. 9th International Conference on Advanced Computer Information Technologies (ACIT) June 5-7 in Ceske Budejovice, Czech Republic. 2019. P. 121-124.

15. Lupenko S., Lutsyk N., Yasnii O., Zozulia A. The Modeling and Diagnostic Features in the Computer Systems of the Heart Rhythm Analysis with the Increased Informativeness. 2019 9th International Conference on Advanced Computer Information Technologies (ACIT). IEEE, 2019. P. 121-124.
16. Martsenyuk V. P., Sverstiuk A. S., Klos-Witkowska A., Horkunenko A. B., Rajba S. Vector of Diagnostic Features in the Form of Decomposition Coefficients of Statistical Estimates Using a Cyclic Random Process Model of Cardiosignals. The 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 18-21 September. Metz. 2019. Vol. 1. P. 298-303.
17. Lupenko S.A., Lutsyk N.S., Stadnyk N.B. Model iz podviinoiu stokhastychnistiu u zadachakh matematychnoho modeliuвання ta analizu tsyklichnykh protsesiv ta syhnaliv. Materialy IV naukovo-tekhnichnoi konferentsii „Informatsiini modeli, systemy ta tekhnolohii“, 15-16 travnia 2014 roku, Ternopil. S. 10.
18. Lupenko S., Sverstiuk A., Lutsyk N., Stadnyk N., Zozulia A. Umovnyi tsyklichnyi vypadkovyi protses yak matematychna model kolyvnykh syhnaliv ta protsesiv iz podviinoiu stokhastychnistiu. Polihrafiia i vydavnycha sprava. No.1 (71). 2016. S. 147-159.
19. Martseniuk V.P., Semenets A.V., Sverstiuk A.S. Kontseptualni pidkhody do intehrovanoho seredovyscha provedennia naukovykh medyko-biolohichnykh doslidzhen. Shtuchnyi intelekt. 2003. №2. S. 35-44.
20. Martseniuk V.P., Kravets N.O., Sverstiuk A.S. Informatsiina systema medyko-biolohichnykh doslidzhen: proekt na osnovi Web-tekhnolohii. Ukrainskyi zhurnal telemedytsyny ta medychnoi telematyky. 2003. T.1, № 1. S. 57-60.
21. Lupenko S.A., Demianchuk N. R., Sverstiuk A. S. Kontseptualno-metodolohichni osnovy imitatsiinoho modeliuвання tsyklichnykh syhnaliv na EOM iz vykorystanniam yikh modeli u vyhladi tsyklichnoho funktsionalnoho vidnoshennia. Vymiriuvalna ta obchysliuvalna tekhnika v tekhnolohichnykh protsesakh. 2008. № 4. S. 101–111.

DOI: 10.36910/6775-2524-0560-2020-39-12

УДК: 004.9:378.016

Мамчич Тетяна Іванівна, к.ф.-м.н., доцент

<https://orcid.org/0000-0002-1934-1484>

Миронюк Лілія Павлівна, к.ф.-м.н., доцент

<https://orcid.org/0000-0002-4822-659X>

Ройко Лариса Леонідівна, к.пед. н., доцент

<https://orcid.org/0000-0002-7318-0925>

Східноєвропейський національний університет імені Лесі Українки, м. Луцьк, Україна

ДОСВІД ВИКОРИСТАННЯ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ ПРИ ВИКЛАДАННІ МАТЕМАТИЧНИХ ДИСЦИПЛІН В УМОВАХ ДИСТАНЦІЙНОГО НАВЧАННЯ

Мамчич Т. І., Миронюк Л. П., Ройко Л. Л. Досвід використання інформаційно-комунікаційних технологій при викладанні математичних дисциплін в умовах дистанційного навчання. У статті проаналізовано ідеї використання ресурсів Office 365, Google Classroom та Zoom при викладанні математичних дисциплін для студентів нематематичних спеціальностей в умовах дистанційного навчання.

Ключові слова: дистанційне навчання, інформаційно-комунікаційні технології, вища математика, Office 365, Google Classroom, Zoom.

Мамчич Т. И., Миронюк Л. П., Ройко Л. Л. Опыт использования информационно-коммуникационных технологий при преподавании математических дисциплин в условиях дистанционного обучения. В статье проанализированы идеи использования ресурсов Office 365, Google Classroom и Zoom при преподавании математических дисциплин для студентов нематематических специальностей в условиях дистанционного обучения.

Ключевые слова: дистанционное обучение, информационно-коммуникационные технологии, высшая математика, Office 365, Google Classroom, Zoom.

Mamchych T. I., Myroniuk L. P., Royko L. L. Information and communication technologies usage experience at teaching of mathematical disciplines in the terms of the distance learning. The ideas of implementing Office 365, Google Classroom and Zoom resources in teaching mathematics for the students of non-mathematical specialities in the distance learning were analyzed.

Keywords: distance learning, information and communication technologies, higher mathematics, Office 365, Google Classroom, Zoom.

Постановка наукової проблеми. Відповідно до постанови Кабінету Міністрів України [7], з 12 березня усі навчальні заклади України припинили роботу у звичайному режимі і перейшли на карантин. Міністерство освіти і науки зобов'язало педагогів забезпечити організацію дистанційного навчання для студентів. Ця ситуація, у свою чергу, поставила перед викладацьким складом задачу по швидкому переведенню існуючих курсів у дистанційну форму навчання.

Насправді, дистанційне навчання стало викликом не тільки для студентів, а й для їх наставників. Як показав досвід, успішність навчання в онлайн-режимі залежить саме від ефективної його організації, від керівництва самим процесом і майстерності викладача, що приймає участь у ньому. Вимог або якихось обмежень стосовно дистанційного навчання в умовах карантину ні навчальні заклади, ні викладачі не отримали. Таким чином, кожен почав працювати на власний розсуд, спираючись на досвід і специфіку навчальної дисципліни.

Дистанційне навчання – це не нова форма освітнього процесу. Вона не вимагає чогось захмарного, достатньо сісти за комп'ютер і налаштуватись на освітній процес. Для організації такої форми навчання можна використати, наприклад, такі ресурси, як Moodle, Office 365 (Teams, Class Notebook та ін.), Skype, Google Classroom та Hangouts, Zoom, Viber, Telegram, Facebook Messenger та ін.

В Україні дистанційне навчання регулюється «Положенням Про дистанційне навчання», затвердженим у квітні 2013 року Наказом Міністерства освіти і науки України № 466. Згідно з цим документом, дистанційне навчання – це «індивідуалізований процес набуття знань, умінь, навичок і способів пізнавальної діяльності, який відбувається в основному за опосередкованої взаємодії віддалених один від одного учасників навчального процесу у спеціалізованому середовищі, яке функціонує на базі сучасних психолого-педагогічних та інформаційно-комунікаційних технологій» [5].

Аналіз досліджень. Проблему розроблення технологій дистанційного навчання та впровадження їх в освітній простір розкрили В. Ю. Биков, Н. В. Буркіна, А. П. Веремчук, С. С. Вітвицька, І. В. Герасименко, В. В. Кабак, В. О. Красношарпа, В. М. Кухаренко, Н. В. Рашевська, Ю. В. Триус [9].

Питанням наукового забезпечення дистанційного навчання, напрямам досліджень цієї галузі присвячені роботи В. Ю. Бикова, Л. В. Бойко, Ю. О. Дорошенка, Ю. В. Іванова, М. І. Жалдака, М. М. Карпенка та ін. [6].

Психолого-педагогічні аспекти та технології створення дистанційних курсів, контроль знань та їх оцінювання висвітлено у працях О. А. Григорової, В. В. Дайнеко, В. М. Кухаренко, Н. Г. Сиротенко тощо.

Організаційно-педагогічні основи дистанційної освіти в Україні та за її межами, підходи до їх реалізації досліджували Т. Андерсон, Р. С. Гуревич, В. В. Олійник, Ю. А. Пасічник, М. Рагус, Дж. Тракслер, М. Шарплз О. В. Собаєва, П. М. Таланчук.

Перспективи дистанційного навчання у навчальних закладах України і за кордоном розглядали Д. Аттевель, Н. В. Буркіна, К. В. Корсак, Г. О. Козлакова, П. В. Стефаненко.

Проте досліджень, присвячених проблемі навчання різних математичних дисциплін у дистанційній формі, обмаль. Варто назвати роботи В. Р. Бурачека [1], Л. П. Вороновської [2], Л. П. Гусак [3], В. Є. Пузирьової [8].

Отже, питання ефективної реалізації технологій дистанційного навчання під час викладання математичних дисциплін для студентів університету є вкрай актуальним.

Метою статті є представлення досвіду використання інформаційно-комунікаційних технологій при викладанні математичних дисциплін для студентів нематематичних спеціальностей в умовах дистанційного навчання викладачами кафедри вищої математики та інформатики Східноєвропейського національного університету імені Лесі Українки.

Відповідно до мети авторами статті було поставлене завдання: розглянути можливість використання ресурсів Office 365 (Outlook, Class Notebook, Teams), Google Classroom, а також Zoom для забезпечення проведення лекційних та практичних занять, модульних контрольних робіт, самостійної та індивідуальної робіт, прийняття іспитів в умовах дистанційного навчання.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Викладання математичних дисциплін дистанційно має свої особливості, які зумовлені використанням формул та графічних зображень. Комп'ютерні програми, які будують формули та графічні зображення є, але проблема, власне, не в них, а в самій технології підготовки матеріалів до викладання: проведення віддалених занять потребує швидкої реакції під час занять, як з боку викладача, так і студентів, а також потреби в організації навчального процесу при існуючих технічних можливостях студентів та викладачів [4].

Передумови дистанційного навчання в карантинний період наступні:

– матеріали для викладання курсів у викладачів є, але вони не обов'язково у формі, готовій до застосування онлайн. Підготовка цих матеріалів потребує багато часу, а навчальний період обмежено затвердженням розкладом;

– під час занять бажано мати аудіо-візуальний контакт, але не всі викладачі та студенти забезпечені відповідною технікою;

– під час занять з математики виникає потреба швидкого обміну повідомленнями у вигляді формули, або графічного зображення між учасниками процесу, коли використання інших програм (редактор формул у Microsoft Word, або TEX) не є доцільним, адже потребує часу;

– окремою проблемою є проведення опитування (контрольні, модульні, екзамен). Частково це можуть бути тести, але у математиці вони не перевіряють знання у повному обсязі, хотілось би побачити також хід розв'язування;

– під час письмових екзаменів існує ризик невідповідної технології проведення, яка провокуватиме академічну недобросовісність, тобто, списування, якщо завдання для цілої групи однакові. Якщо ж завдання індивідуальні, то виникає організаційна проблема розробки індивідуальних наборів завдань, їх одночасного повідомлення, отримання відповідей, перевірки великої кількості варіантів.

Викладання навчальної дисципліни «Моделювання знань» для студентів 4 курсу спеціальності «Соціологія» забезпечувалося сервісом для проведення відео-конференцій та онлайн-зустрічей Zoom.

Наш досвід використання цієї програми є ще досить коротким, проте вже можна зробити певні висновки та поширити отримані методичні напрацювання.

Сервіс дозволяє досить комфортно читати лекції. Студенти можуть бачити та чути викладача, а також усе, що викладач демонструє на своєму комп'ютері. Це може бути презентація, яка супроводжується поясненням, або ж демонстрація текстових файлів з відповідної теми (рис. 1). Можна сказати, що при такому способі студенти ще краще бачать всі формули, ніж би це було вживу, коли записи роблять на дошці. Розмір шрифту, як відомо, регулюється.

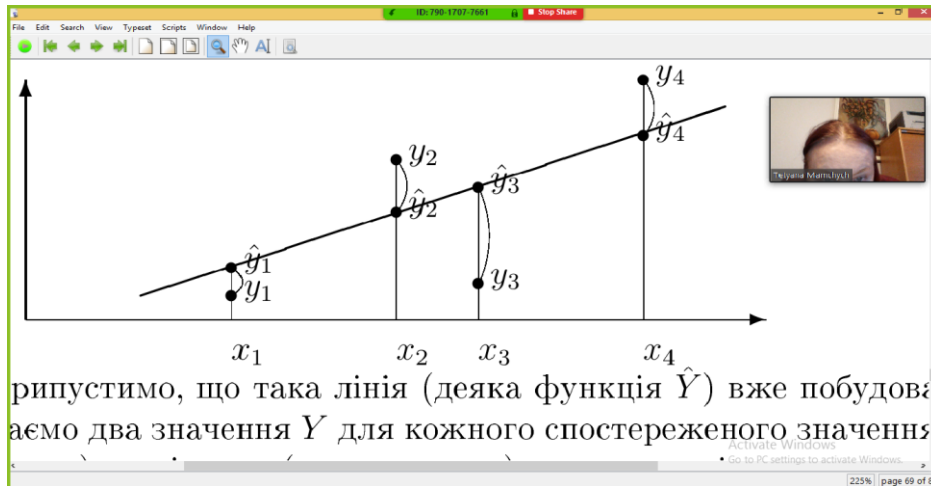


Рис. 1. Фрагмент зображення програми Zoom: демонстрація текстового файлу

Для негайного обміну повідомленнями у вигляді формули, чи графічного зображення можна написати його на папері і продемонструвати на камеру (рис. 2).

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(\sum (x_i - \bar{x})^2)(\sum (y_i - \bar{y})^2)}}$$

$$-1 \leq r \leq 1$$

Рис. 2. Фрагмент зображення програми Zoom: демонстрація тексту, написаного від руки

Для опитувань, коли відповідями є числа, або текст без формул можна використати чат. Є можливість обміну повідомленнями між усіма учасниками, або лише з викладачем.

Для практичного заняття потрібно пересилати завдання та отримувати відповіді. В програмі організовано швидкий обмін файлами.

Дана технологія цілком придатна для читання лекцій, проведення практичних занять і для екзаменування, а методологія застосування потребує подальшого вивчення та розвитку.

Викладання навчальної дисципліни «Математика для економістів та економічне моделювання» для студентів 1 та 2 курсів факультету міжнародних відносин було організоване через Google Classroom та Zoom. Зупинимось на характеристиці використання засобів Google Classroom у проведенні практичних занять.

Google Classroom є додатком Google, який розроблений саме для освітніх потреб. Ця платформа дозволяє викладачу використовувати усі інструменти, наприклад, для створення і обміну завданнями (Google Drive), для написання завдань (Docs), для розкладу (Calendar) і для спілкування (Gmail). Також допомагає створювати і впорядковувати завдання або матеріали, давати роз'яснення (коментувати) і організовувати ефективне спілкування з студентами у режимі реального часу.

У Classroom були створені класи «Математика для економістів 1 курс» та «Математика для економістів 2 курс» (рис. 3).

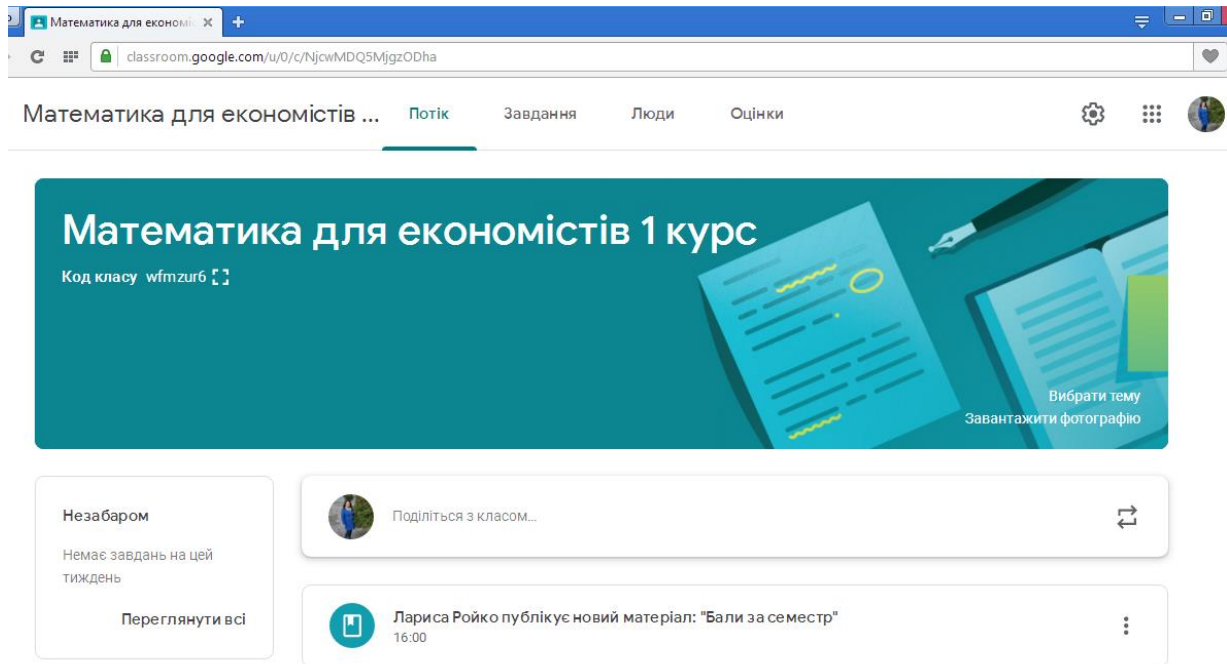


Рис. 3. Фрагмент зображення Classroom студентів спеціальностей МЕВ та МІБ

Студенти мали можливість приєднатись до свого класу через спецкоди, які їм повідомив викладач. При використанні Google Classroom папка «Classroom» у викладача автоматично створюється на його робочому Google Диску. Аналогічно вона створюється і у студентів із вкладеними папками даного навчального курсу.

Згідно розкладу навчальних занять студенти міжнародники отримували завдання, які викладач мав можливість створити у вкладці «Завдання». Це могли бути звичайні індивідуальні завдання по темі, завдання у вигляді тестів, завдання подані у формі матеріалу (наприклад, пояснення до розв'язування типових завдань). Всі ці завдання були підготовлені раніше і зберігались в електронному вигляді. Зараз вони прикріплювались файлом (рис. 4). Є можливість прикріпити відео.

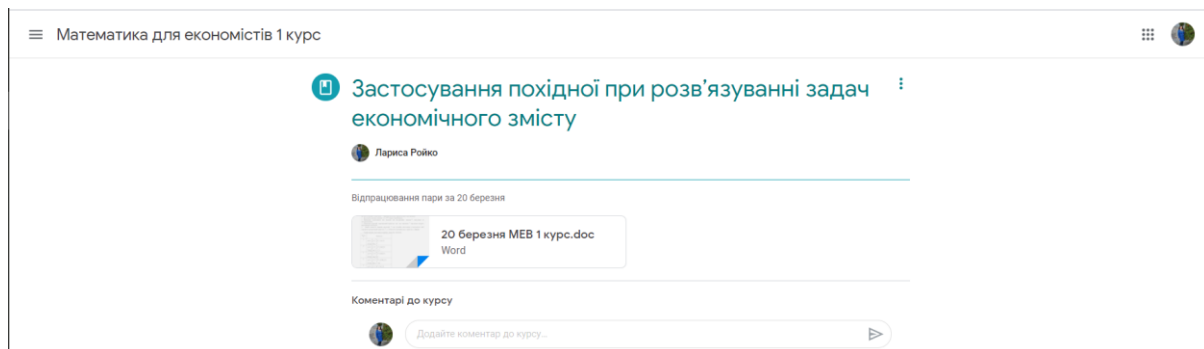


Рис. 4. Фрагмент зображення завдань по темі «Застосування похідної»

І викладач, і студенти бачили усі завдання на головному екрані Google Classroom. Це дозволяло контролювати роботу навіть одразу у двох класах. При створенні кожного завдання вказувався термін виконання роботи. Коли студент здавав завдання до початку терміну, на його документі з'являвся статус «Перегляд». Завдання підбирались індивідуально і були обмежені у часі, тому це унеможливлювало списування один в одного.

Система оцінювання у Classroom може бути адаптована під будь-яку кількість балів. Після того, як викладач оцінив роботу, він мав можливість написати коментарі до її виконання і за допомогою кнопки «Повернути» відправити отримані бали на пошту кожного студента (за необхідності оцінку можна змінити). Також є можливість всі бали, отримані у класі, представити у вигляді таблиці.

Підтримування зв'язку між викладачем та студентами відбувається за допомогою сервісу «Оголошення», також тут можна слідкувати за станом виконання або перевірки заданих завдань.

За допомогою цієї платформи були проведені практичні заняття, на яких студенти факультету міжнародних відносин виконували індивідуальні завдання по темах, самостійні та модульні контрольні роботи. Дана платформа цілком придатна для таких видів робіт.

Викладання курсу «Вища математика» для студентів 1 курсу географічного факультету спеціальності «Геодезія та землеустрій» здійснювалося засобами Office 365, а також Zoom.

Після проходження студентами відповідної реєстрації в Office 365, кожен із них отримав лист, що його додано до групи «surveyors-2019» в Outlook (рис. 5):

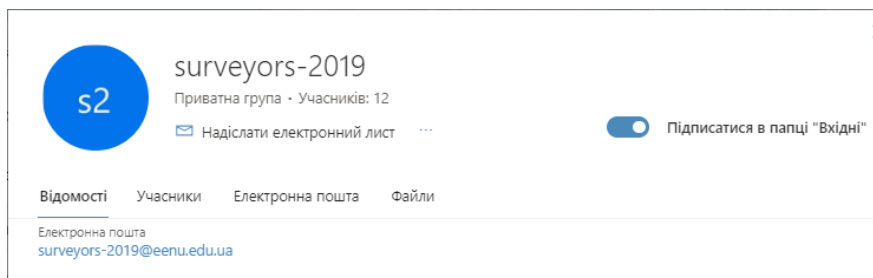


Рис. 5. Фрагмент зображення в Outlook групи «surveyors-2019»

Наявність групи дала можливість надсилати певні повідомлення чи вказівки усім студентам одночасно, що значно економило час.

Студентів було проінформовано, що їх додано до блокноту для класу «Вища математика», створеного засобами надбудови Class Notebook для OneNote. Як відомо, блокноти OneNote для класу містять бібліотеку вмісту для супровідних матеріалів, простір для співпраці, а також особисту робочу область для кожного студента. В нашому випадку остання, в свою чергу, включала такі розділи, як «Домашні завдання», «Індивідуальні домашні завдання», а також «Модульні контрольні роботи».

Студенти були попереджені, що перша модульна контрольна робота буде проводитись дистанційно. В день її написання, кожен студент отримав відповідне посилання на роботу, створену засобами Forms (рис. 6):

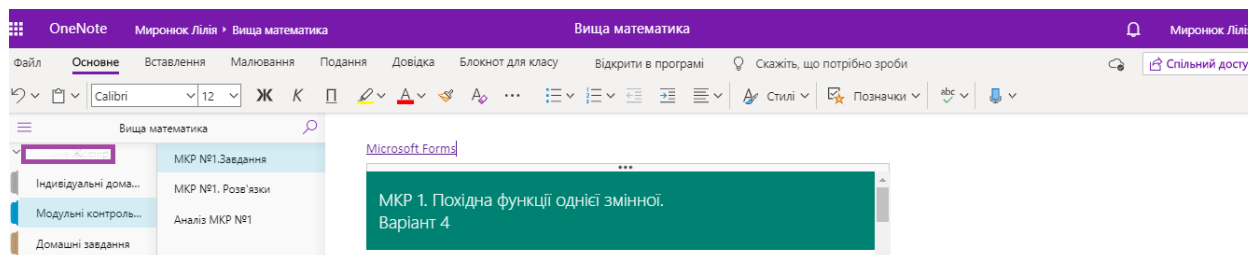


Рис. 6. Фрагмент зображення Class Notebook з посиланням на модульну контрольну роботу

Модульна контрольна робота містила теоретичні запитання у вигляді тестів, а також практичні завдання. Варто відзначити, що можливості створення тестів в Forms саме з математичних дисциплін значно розширилися порівняно з попередніми роками. Якщо раніше необхідні формули можна було вставити лише як зображення, то тепер спеціальна опція «Математика» дозволяє вводити формули «вручну» (рис. 7):

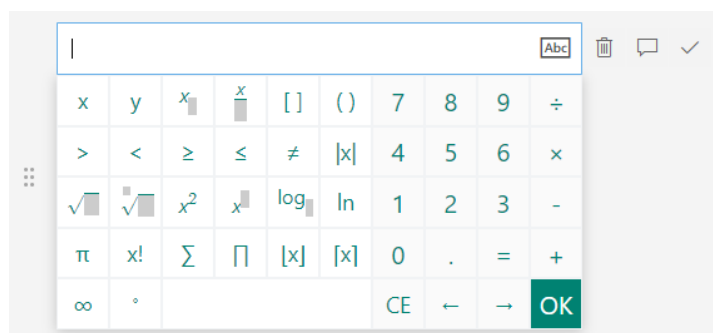


Рис. 7. Фрагмент зображення у Forms опції «Математика»

Зауважимо також, що користуватися можна не лише наявними шаблонами, але й додавати інші спеціальні символи. Так, для прикладу, якщо є необхідність набору грецьких літер, то достатньо знати їх правильне написання в Forms, яке повністю співпадає з відповідним написанням у Word версії 2007 року та наступних років. Зокрема, символ α буде вставлено, якщо у відповідному рядку формул ввести наступне: $\backslash\alpha$. Степені змінних чи чисел, а також індекси змінних можна вводити з використанням шаблону або символів \wedge та $_$ відповідно, наприклад, ввівши x^2 , отримаємо: x^2 , а ввівши x_2 , отримаємо: x_2 . Дроби можна вводити з використанням шаблону або символу \backslash . Після введення $a\backslash b$, одержимо: $\frac{a}{b}$.

Опитування, проведене в Forms, після написання контрольної роботи, показало, що для завантаження зображення вимагалось більше часу, тоді як з введеними вручну формулами таких проблем не було.

По закінченню написання контрольної роботи студенти мали можливість ознайомитись з отриманими балами за тести по теоретичному матеріалу. Розв'язані ж практичні завдання вони прикріплювали в своїй особистій робочій області в блокноті для класу, вкладинці «МКР№1.Розв'язки», яка, як відомо, доступна лише їм та викладачеві.

Досить зручною є можливість перегляду результатів у Excel, де вказано час початку та завершення тесту Forms, адже час написання контрольної роботи був обмежений – 1 год, 20 хв.

Кожному студенту було надіслано аналіз контрольної роботи, вказано на наявні помилки.

У простір для співпраці студенти додавали законспектовані самостійно приклади завдань, що мають практичне застосування, різні рисунки тощо. Зокрема, для кращого засвоєння нового матеріалу, що стосувався функцій двох змінних, а також повторення та поглиблення знань з теми «Поверхні другого порядку», вивченої минулого семестру, студенти отримали завдання засобами web-орієнтованої системи Wolframalpha побудувати поверхні, задані як в явному, так і неявному вигляді. Очевидно, що даний досвід сприяв реалізації міжпредметних зв'язків, наприклад, з дисциплінами «Геодезія», «Вища геодезія». Після виконання завдання студенти зауважили, що, як їм стало відомо, насправді Земля не є ідеальною сферою, тож в геодезії та космонавтиці, зазвичай, для опису фігури Землі вибирають еліпсоїд обертання (перша поверхня на рис. 8) або геоїд:

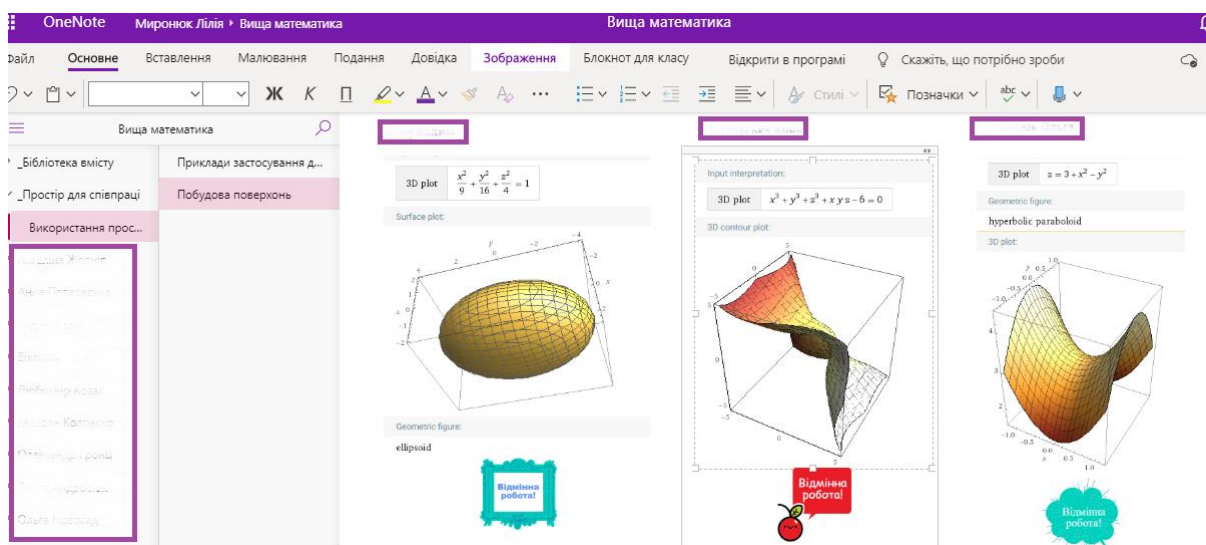


Рис. 8. Фрагмент зображення простору для співпраці у Class Notebook

Лекційні та практичні заняття проводились з використанням Zoom.

Студенти, що з певних причин не мали можливості бути присутні на заняттях, могли ознайомитись з теоретичним матеріалом в блокноті для класу, вкладинці «Бібліотека вмісту».

Фотографії виконаних домашніх завдань, однакових для усіх, а також індивідуальних домашніх завдань, студенти прикріплювали у відповідних розділах особистої робочої області в блокноті для класу.

Викладання курсу «Математика для економістів та економічне моделювання» для студентів 2 курсу факультету міжнародних відносин англomовної групи здійснювалося повністю аналогічно засобами Office 365, а також Zoom.

Висновки та перспективи подальшого дослідження. Дистанційне навчання математичних дисциплін в останні роки є сферою посиленних наукових досліджень. Незважаючи на отриманий досвід його організації через вимушені обставини, а також потенціал комп'ютерних технологій у

можливості зробити навчальний процес доступним у будь-який час і у будь-якому місці, автори статті залишаються прихильниками змішаного навчання.

Варто зазначити, що опитування, проведене в Forms зі студентами 1 курсу спеціальності «Геодезія та землеустрій», показало, що студенти краще сприймають матеріал з математичних дисциплін на стаціонарних заняттях, аніж дистанційно, проте, в цілому, вони були задоволені опануванням нових технологій. Зауважимо, що деякі зі студентів пропонували навіть відомі їм програми, які можна використовувати для організації навчального процесу в умовах дистанційного навчання. Серед інших варто назвати Discord – безкоштовний месенджер з підтримкою VoIP та відеоконференцій, що спочатку був орієнтований на користувачів комп'ютерних ігор. На даний час нам відомо, що викладач математики Тарас Павлов пропонує використовувати цю програму для дистанційного навчання математики учнями.

Досвід використання ресурсів Office 365 (Outlook, Teams, Class Notebook, Forms, Whiteboard та ін.), а також Zoom дає підстави стверджувати, що вони цілком придатні для проведення лекцій, практичних занять та іспитів. Для самостійної та індивідуальної роботи зручне використання Google Classroom. У цілому, методологія застосування даних ресурсів потребує подальшого вивчення та розвитку, адже дистанційне навчання з математики повинно мати певну концепцію, налагоджену систему існування та психолого-педагогічні основи.

Список бібліографічного опису

1. Бурачек В. Р. Забезпечення глибини засвоєння матеріалу при дистанційному вивченні дисциплін математичного циклу / В. Р. Бурачек // Збірник тез доповідей на Всеукраїнському науково-методичному семінарі з елементами вебінару. – Харків : РВВ ХТЕІ КНТЕУ, 2016. – 76 с. [Електронний ресурс]. Режим доступу: www.vtei.com.ua/images/VN/31_03.pdf.
2. Вороновська Л. П. Дистанційний курс у системі викладання вищої математики студентам, які навчаються в галузі знань «Будівництво та архітектура» / Л. П. Вороновська // Збірник наукових праць. Педагогічні науки. – Херсон. – 2018. – Вип. (84), т. 2. – С. 83–87.
3. Гусак Л. П. Проблеми дистанційного навчання вищої математики в економічному ВНЗ. / Л. П. Гусак // Матеріали міжвузівського вебінару / відп. ред. Л. Б. Ліщинська. – Вінниця : ВТЕІ КНТЕУ, 2017. – 102 с. [Електронний ресурс]. Режим доступу: www.vtei.com.ua/images/VN/31_03.pdf.
4. Мамчич Т. І., Мамчич І. Я., Ройко Л. Л., Ройко О. О. Навчання методам прикладної математики за підтримки програми R / Т. І. Мамчич, І. Я. Мамчич, Л. Л. Ройко, О. О. Ройко // Науковий журнал «Комп'ютерно-інтегровані технології: освіта, наука, виробництво» – №35, ЛНТУ, 2019. – С. 37–41.
5. Наказ Міністерства освіти і науки України «Про затвердження Положення про дистанційне навчання» [Електронний ресурс]. – Режим доступу : <http://zakon4.rada.gov.ua/laws/show/z0703-13>.
6. Пахолок З., Миронюк Л. Методика опрацювання хмарних сервісів Google Slides та Sites для створення презентацій про рідну країну і розміщення їх на сайті в курсі «Українська мова як іноземна» / З. Пахолок, Л. Миронюк // Науковий журнал «Комп'ютерно-інтегровані технології: освіта, наука, виробництво» – №38, ЛНТУ, 2020. – С. 51–58.
7. Постанова Кабінету Міністрів України «Про запобігання поширенню на території України гострої респіраторної хвороби COVID-19, спричиненої коронавірусом SARS-CoV-2» № 211 від 11.03 2020 р. [Електронний ресурс]. – Режим доступу : <https://zakon.rada.gov.ua/laws/show/211-2020-п>
8. Пузирьов В. С. Дистанційне навчання вищої математики: досвід Донецького національного університету / В. С. Пузирьов // Педагогічні науки: теорія, історія, інноваційні технології, № 6 (50), 2015. – С. 283–290.
9. Триус Ю. В. Комп'ютерно-орієнтовані методичні системи навчання математичних дисциплін у ВНЗ: проблеми, стан і перспективи / Ю. В. Триус // Науковий часопис НПУ імені М. П. Драгоманова. Серія 2 : Комп'ютерно-орієнтовані системи навчання. – 2010. – №. 9. – С. 16–29 // [Електронний ресурс]. Режим доступу: http://nbuv.gov.ua/UJRN/Nchnpu_2_2010_9_5.

References

1. Burachek V. R. Zabezpechennia hlybiny zasvoiennia materialu pry dystantsiinomu vyvchenni dystsyplin matematychnoho tsyklu / V. R. Burachek // Zbirnyk tez dopovidei na vseukrainskomu naukovo-metodychnomu seminarі z elementamy vebinaru. – Kharkiv : RVV XTEI KNTEU, 2016. – 76 s. [Elektronnyi resurs]. Rezhym dostupu: www.vtei.com.ua/images/VN/31_03.pdf.
2. Voronovska L. P. Dystantsiinyi kurs u systemi vykladannia vyshchoi matematyky studentam, yaki navchaiutsia v haluzi znan «Budivnytstvo ta arkhitektura» / L. P. Voronovska // Zbirnyk naukovykh prats. Pedahohichni nauky. – Kherson. – 2018. – Vyp. (84), t. 2. – S. 83–87.
3. Husak L. P. Problemy dystantsiinoho navchannia vyshchoi matematyky v ekonomichnomu VNZ. / L. P. Husak // Materialy mizhvuzivskoho vebinaru / vidp. red. L. B. Lishchynska. – Vinnytsia : VTEI KNTEU, 2017. – 102 s. [Elektronnyi resurs]. Rezhym dostupu: www.vtei.com.ua/images/VN/31_03.pdf.
4. Mamchych T. I., Mamchych I. Ia., Roiko L. L., Roiko O. O. Navchannia metodam prykladnoi matematyky za pidtrymky prohramy R / T. I. Mamchych, I. Ia. Mamchych, L. L. Roiko, O. O. Roiko // Naukovyi zhurnal «Kompiuterno-intehrovani tekhnolohii: osvita, nauka, vyrobnytstvo» – №35, LNTU, 2019. – S. 37–41.
5. Nakaz Ministerstva osvity i nauky Ukrainy «Pro zatverdzhennia Polozhennia pro dystantsiine navchannia» [Elektronnyi resurs]. – Rezhym dostupu : <http://zakon4.rada.gov.ua/laws/show/z0703-13>.
6. Pakholok Z., Myroniuk L. Metodyka opratsiuvannia khmarnykh servisiv Google Slides ta Sites dlia stvorennia prezentatsii pro ridnu krainu i rozmishchennia yikh na сайti v kursі «Ukrainska mova yak inozemna» / Z. Pakholok, L. Myroniuk // Naukovyi zhurnal «Kompiuterno-intehrovani tekhnolohii: osvita, nauka, vyrobnytstvo» – №38, LNTU, 2020. – S. 51–58.

7. Постанова Кабінету Міністрів України «Про запобігання поширенню на території України гострої респіраторної хвороби COVID-19, спричиненої коронавірусом SARS-CoV-2» № 211 від 11.03 2020 р. [Elektronnyi resurs]. – Rezhym dostupu : <https://zakon.rada.gov.ua/laws/show/211-2020-p>.
8. Puzyrov V. Іe. Dystantsiine navchannia vyshchoi matematyky: dosvid Donetskoho natsionalnoho universytetu / V. Іe. Puzyrov // Pedagogichni nauky: teoriia, istoriia, innovatsiini tekhnolohii, № 6 (50), 2015. – S. 283–290.
9. Tryus Yu. V. Kompiuterno-oriientovani metodychni systemy navchannia matematychnykh dystsyplin u VNZ: problemy, stan i perspektyvy / Yu. V. Tryus // Naukovyi chasopys NPU imeni M. P. Drahomanova. Seriia 2 : Kompiuterno-oriientovani systemy navchannia. – 2010. – №. 9. – S. 16–29 // [Elektronnyi resurs]. Rezhym dostupu: http://nbuv.gov.ua/UJRN/Nchnpu_2_2010_9_5.

DOI: 10.36910/6775-2524-0560-2020-39-13

УДК 004.048

Мартиненко Андрій Анатолійович, старший викладач, аспірант.

<https://orcid.org/0000-0002-5033-4696>

Мороз Борис Іванович, д.т.н., професор.

<https://orcid.org/0000-0002-5625-0864>

Гуліна Ірина Григорівна, к.т.н., доцент.

<https://orcid.org/0000-0003-2565-5006>

Національний технічний університет «Дніпровська політехніка»

AN INTELLIGENT DECISION SUPPORT SYSTEM FOR CULTURAL PROPERTY IDENTIFICATION

Мартиненко А.А., Мороз Б.І., Гуліна І.Г. Интеллектуальна система підтримки прийняття рішень ідентифікації культурних цінностей. В статті розглядається проблема ідентифікації культурних цінностей, питання розробки методів і моделей організації та обробки даних і знань в інтелектуальній системі підтримки прийняття рішень ідентифікації культурних цінностей. Також зазначається складність та комплексність підходу вирішення поставленої задачі, визначені перспективи та шляхи подальшого дослідження даної предметної області.

Ключові слова: експертна система, системи підтримки прийняття рішень, інтелектуальний аналіз даних, ідентифікація культурних цінностей.

Мартыненко А.А., Мороз Б.И., Гулина И.Г. Интеллектуальная система поддержки принятия решений идентификации культурных ценностей. В статье рассматривается проблема идентификации культурных ценностей, вопросы разработки методов и моделей организации и обработки данных и знаний в интеллектуальной системе поддержки принятия решений идентификации культурных ценностей. Также отмечается сложность и комплексность подхода решения поставленной задачи, определены перспективы и пути дальнейшего исследования данной предметной области.

Ключевые слова: экспертная система, системы поддержки принятия решений, интеллектуальный анализ данных, идентификация культурных ценностей.

Martynenko A.A., Moroz B.I., Hulina I.G. An intelligent decision support system for cultural property identification.

The article considers the problem of identifying cultural property, the development of methods and models for organizing and processing data and knowledge in an intelligent decision support system for cultural property identification. The complexity of the approach for solving the stated problem, prospects and ways of further research of this subject area are noted and identified.

Key words: expert system, decision support system, data mining, cultural property identification.

Introduction. Problem statement. The problem of research and development of methods and models of processing data and knowledge using intelligent decision support systems (IDSS) for the identification of cultural property (CP) is an urgent and promising task [1]. Despite the rapid development of information and intelligent technologies, there is currently no one expert system (ES) or decision support system (DSS) to systematically address the identification and analysis of cultural property. As stated by the authors in their work [1], systems of this type can additionally solve a wider range of tasks: finding and storing information about the CP, and can also be used to popularize this issue in society. It should also be noted that most of the tasks and problems of identifying cultural property are either in the process of being partially resolved or are being formulated.

Literature review. The rapid development of this area of focus has been observed in recent years. Classical works on the design and development of the principles of creating DSS are the works of Pospelova, Popova, Zade, Waterman, Bakaeva, Ulyanovskaya and others. The ideas of these works has been reflected in a number of studies which examine the problems of organizing knowledge bases, models and methods of working with fuzzy or incomplete data.

The conducted analysis of the works in this area leads to the conclusion that, despite the great development and attempts of designing the ES and the DSS, there is no definitive and comprehensive solution to the problem of the identification of cultural property with the help of implementation of the DSS.

It should also be noted that there were attempts to solve problems of this type, but they were narrowly targeted and partially solved. For example, some of the problems is solved using fuzzy set theory which cannot fully achieve the desired quality due to a number of constraints.

Previously unsolved parts. At present, there is no universal model for development of IDSS. The introduction and use of modern tools and approaches in creating systems of this type, the use of a systematic approach, according to the authors, is more effective.

Objectives of the research. To improve the quality and efficiency of solving the practical problem of identification of CP through the implementation and use of IDSS created on the basis of modern methods, models, technologies and tools.

Main part. Recently, scientists and researchers have proposed more modern and effective methods and means of object identification and the construction of modern IDSS. It should also be noted that modern solutions help to avoid or reduce a number of disadvantages that existed in previous solutions.

At the same time, there are still models of different ES and DSS that are developed under uncertainty using probability theory and fuzzy logic [9], expert systems with uncertain knowledge, using subjective probability theory and Bayes theorem as a basis for uncertainty management (Bayesian trust networks as one of the directions of construction and organization of work of expert systems, representation of knowledge using Bayesian trust network and conditional event independence), systems based on Demster-Schaeffer theory [7, 8], etc.

Also, in recent years, the use of such a powerful tool as neural networks (NN) has become very popular [10, 11]. In this case, the use of NN can be applied to the recognition, identification of CP objects and the construction of the IDSS.

Unlike the traditional use of NN for solving only the problems of pattern recognition and formation, the DSS can solve the following tasks:

- pattern recognition and formation;
- receiving and storing of knowledge (empirically found regular relationships of images and influences on the object of control);
- assessment of the quality characteristics of the images;
- decision making (choice of influences).

Consider the scheme proposed by the authors in their work (fig. 1.) [2].

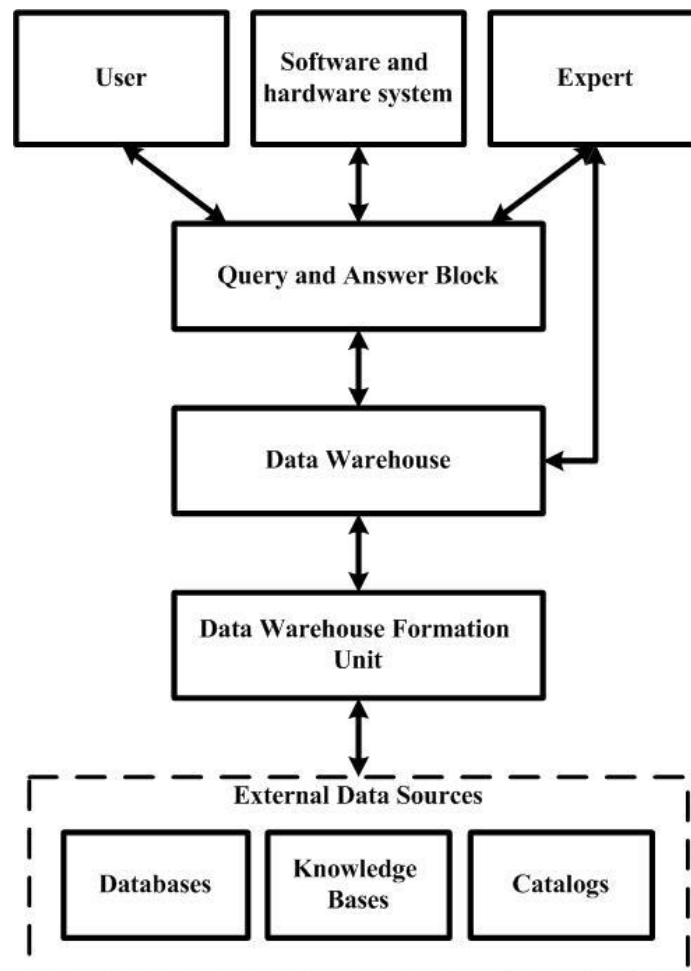


Fig. 1. The IDSS scheme of CP identification

If to look at the proposed scheme in more detail, the following points can be noted:

"User". The following categories may act as a user:

- experts in the field of cultural values and art objects;
- museums;
- educational institutions;
- organizations for the creation and filling the catalogs of cultural values and art objects;
- state bodies for control over cultural values and objects of art;
- customs;
- average users of the system (guests of the resource);
- art critics;
- historians;
- auctions of cultural property and art objects;
- other systems and information resources.

"Software and hardware system". In this case, this part of the system can be a separate, functionally logical unit or module (optional for this type of system). The system can be an automated scanning/photographing system or a system that performs measurements of various parameters of cultural objects and transmits for processing and analysis to the "Query and Answer Block" through the appropriate interfaces (software and hardware).

"Expert". In the given scheme the expert can be both in the role of the user-consultant in the process of the development of the system itself, and in the role of the person who forms/fills the "Data Warehouse" according to the established rules and data formats. An expert group may also act as an expert, since the effectiveness of the system (identification and conceptualization stages) is largely determined by the successful formation of a reputable expert group and obtaining the qualitative knowledge from the group that forms the basis of any IDSS.

The essence of the process of knowledge discovery lies in conducting by experts an intuitive-logical analysis of the problem area with a quantitative assessment of the judgments they formulate. In this scheme, experts can determine the objects and concepts of the subject area (goals, decisions, alternative situations, etc.), measure the characteristics (probability of event completion, coefficients of significance of goals, decisions preferences, etc.) [4].

"Query and Answer Block". This software or hardware is a subsystem with a user interface by means of which a query is generated to the data warehouse for analysis. This part can be made in the form of a dialog where the user select/set the criteria of the object. After the request is made in the installed view, the subsystem access the data warehouse exactly where information about the object being analyzed is collected. But the main task of this unit is to analyze the data received and generate a response to the user's request.

"Data Warehouse" and "Data Warehouse Formation Unit". In their work [2], the authors considered the work of these subsystem and noted that special attention should be paid to the issues of justifying the criteria for data warehousing and data consolidation. The issue of creating and filling data warehouses is considered in the work [3]. Consideration should also be given to developing methods and models for evaluating the quality of third-party data when filling the data warehouse.

"External Data Sources" ("Databases (DB)", "Knowledge Bases (KB)" and "Catalogs"). In this scheme these blocks are conditionally united into one whole and represent the already existing CP object bases, for example these could be:

- museum catalogs or databases;
- descriptions or databases of Ukrainian and foreign organizations for the control of cultural values and objects of art;
- catalogs or databases of the customs services and the Ministry of Internal Affairs regarding the description of cultural values and objects of art;
- catalogs of auctions of cultural property and art objects;
- works of art historians and historians;
- DBs or DBs of other systems and information resources.

Conclusions and prospects for further research. The authors analyzed the problems associated with the development of an intelligent expert system for the identification and analysis of cultural value objects, as well as the solutions used. The obtained results allow us to draw the following conclusions:

- The problem area is relatively new and not well understood field of research,

- The process of identification and analysis of cultural objects, and the objects themselves are not sufficiently described, since cultural values are various objects of different shapes, structures and materials with specific characteristics [1, 5]. It should also be noted that cultural values are constantly created, which makes the process of data generation difficult.
- Creating universal models of knowledge bases for intelligent decision support systems remains an open and relevant issue.
- Existing models of presenting knowledge and creating databases and knowledge in this field have disadvantages.

Creating a universal intelligent decision support system for cultural values identification is a big complex and extremely time-consuming task that, according to the authors, can be divided into separated tasks that can be solved within the framework of individual scientific research and practical implementation and use in the problems of similar class.

For example, if to refer to the scheme presented in Fig. 1, then the formation and work organization of each block is a separated task.

And if in the case of the block "User", the task can be solved quite simply and quickly, since the principles of user work with the system already exist and are sufficiently described and implemented in systems of this type, then others present more complex tasks.

For example, the block "Hardware and Software System", which is not obligatory for this type of systems, but during the rapid development of modern information technologies and variety of software and hardware, this system can greatly facilitate and make the process of identification more effective. For example, performing in an automatic mode high precision measurements of certain parameters/characteristics of objects (size, weight, barcode scanning, high resolution scanning, etc.), which in turn can reduce the time and minimize the effect of human factors. Thus, the development of software and hardware systems of this type that can be included in the IDSS is difficult and more practical engineering work, so the authors of the work have not considered it in detail but it can be considered as a further perspective development and improvement of the system as a whole.

There are various approaches in the organization of work and filling of "Data Warehouse", the development of algorithms and the organization of principles of data warehouse formation, which represent a separate scientific problem of data analysis and processing of large data sets [2]. But it can be noted that for certain systems, storage can be created in different forms and formats, depending on the complexity of the system and requirements for its functionality. For example, it may look like a simple greenhouse, relational database or OLAP cube.

The performed analysis also allowed the authors to specify the task and to determine a possible plan for its solution.

The principles of organizing the work of the "Block of query and answer formation" in the author's paper are of great scientific interest. Since the main purpose of this block is to analyze the obtained data and to formulate a response to the user's request, then in the opinion of the authors, special attention should be given to such tasks as:

1. Analysis of knowledge representation models and development of a new model for the given problem area if necessary.
2. Development of a form and a method of querying the data warehouse in accordance with the tasks to be solved and in the context of the specific characteristics of the objects.
3. Development of an algorithm for searching and sampling data in the data warehouse.
4. Development of a model for processing and analysis of data obtained from search and sampling (item 3).
4. Development of criteria for evaluation of the reliability of the results and formation of the answers that the system produces as a result of the analysis.
5. Approval of the developed methods and models of organization and processing of data and knowledge in the intelligent decision support system for cultural values identification on the example of some certain type of objects of cultural values.

Список бібліографічних посилань

1. А.А. Мартиненко, Б.І. Мороз, І.Г. Гуліна «Інтелектуальна система підтримки прийняття рішень ідентифікації культурних цінностей». IV Всеукраїнська науково-практична конференція «Перспективні напрямки сучасної електроніки, інформаційних і комп'ютерних систем (MEICS-2019)». Дніпро, 27–29 листопада 2019 р.

2. А.А. Мартиненко, Б.І. Мороз, І.Г. Гуліна «Сховища даних системи підтримки прийняття рішень ідентифікації культурних цінностей» XV міжнародна конференція з проблем використання інформаційних технологій в освіті, науці та промисловості. Дніпро, 5–6 грудня 2019 р.
3. Орешков В.И. Паклин Н.Б. «Консолидация данных — ключевые понятия» режим доступу <https://www.cfin.ru/itm/olap/cons.shtml>
4. Хабаров С.П. «Интеллектуальные информационные системы», режим доступу http://www.habarov.spb.ru/new_es/index.htm
5. Закон України "Про вивезення, ввезення та повернення культурних цінностей", режим доступу <https://zakon.rada.gov.ua/laws/show/ru/1068-14/>
6. Ульяновська Ю.В. «Моделі та методи обробки даних в єдиній автоматизованій інформаційній системі митної служби», автореферат дисертації на здобуття наукового ступеня кандидата технічних наук, 05.13.06 – автоматизовані системи управління та прогресивні інформаційні технології, Харків – 2005
7. Е.А. Ганцева, В.А. Каладзе, А.М. Поляков. «Формирование экспертного вывода на основе теории свидетельств» режим доступу <https://cyberleninka.ru/article/n/formirovanie-ekspertnogo-vyvoda-na-osnove-teorii-svidetelstv/viewer>
8. А.С. Луцаевский, Т.С. Чайникова «Параметрическая идентификация правила Инагаки с помощью генетических алгоритмов». Збірник наукових праць Харківського університету Повітряних Сил ім. І. Кожедуба, випуск 3(15), 2007
9. Джарратано Джозеф, Райли Гари. «Экспертные системы: принципы разработки и программирование», 4-е издание.: Пер. с англ. — М.: ООО "И.Д. Вильямс", 2007. — 1152 с.
10. Кононюк А.Ю. «Нейронні мережі і генетичні алгоритми» – К.: «Корнійчук». 2008. – 446 с
11. Демиденко М.А. «Системы підтримки прийняття рішень»: навч. посіб. / М.А. Демиденко; Нац. гірн. ун-т. — Електрон. текст. дані. – Д.: 2016. – 104 с.

References

1. A.A. Martynenko, BI Moroz, I.G. Hulina "An Intelligent Decision Support System for Identifying Cultural Property". IV All-Ukrainian Scientific-Practical Conference "Promising directions of modern electronics, information and computer systems (MEICS-2019)". Dnipro, November 27-29, 2019
2. A.A. Martynenko, BI Moroz, I.G. Hulina "Data Warehouse of Decision Support System for Identification of Cultural Property" XV International Conference on Information Technology Use in Education, Science and Industry. Dnipro, December 5-6, 2019
3. Oreshkov V.I. Paklin N.B. "Data Consolidation - Key Concepts" access mode <https://www.cfin.ru/itm/olap/cons.shtml>
4. Khabarov S.P. "Intelligent Information Systems", access mode http://www.habarov.spb.ru/new_es/index.htm
5. Law of Ukraine "On Exports, Imports and Returns of Cultural Property", access regime <https://zakon.rada.gov.ua/laws/show/en/1068-14/>
6. Ulyanovskaya Yu.V. «Models and methods of data processing in a single automated information system of the customs service», dissertation author's abstract on obtaining a scientific degree of Candidate of Technical Sciences, 05.13.06 - automated control systems and progressive information technologies, Kharkiv – 2005
7. E.A. Gantseva, V.A. Koladze, A.M. Poles. "Formation of expert opinion based on the theory of evidence" access mode <https://cyberleninka.ru/article/n/formirovanie-ekspertnogo-vyvoda-na-osnove-teorii-svidetelstv/viewer>
8. A.S. Lutseevsky, TS Chaynikov "Parametric identification of the Inagaki rule using genetic algorithms". Collection of scientific works of Kharkov University of the Air Force. I. Kozheduba, Issue 3 (15), 2007
9. Jarratano Joseph, Riley Gary. "Expert Systems: Principles of Development and Programming", 4th Edition.: Transl. with English. - M.: LLC "ID. Williams", 2007. - 1152 p.
10. Kononyuk A.Yu. Neural Networks and Genetic Algorithms - K.: Korniychuk. 2008. - 446 p
11. Demidenko MA "Decision Support Systems": Tutorial. tool. / M.A. Demidenko; Nat. mines Univ. — Electron. text. data. - D.: 2016. - 104 p.

DOI: 10.36910/6775-2524-0560-2020-39-14

УДК: 004.8

Марченко Олександр Іванович, к.т.н., доцент,

<https://orcid.org/0000-0002-4537-3420>

Марченко Олексій Олександрович, асистент

<https://orcid.org/0000-0002-5080-4811>

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

МОДЕЛЬ РЕСУРСІВ НЕОДНОРІДНОЇ РОЗПОДІЛЕНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ З ЛОКАЛЬНИМИ ЗВ'ЯЗКАМИ ТА ЇЇ ГРАФ

Марченко О.О., Марченко О.І. Модель ресурсів неоднорідної розподіленої комп'ютерної системи з локальними зв'язками та її граф. На основі аналізу методу MCTS та підходів до його паралелізації у статті запропоновані нова модель ресурсів неоднорідної розподіленої комп'ютерної системи з локальними зв'язками (НРКСЛЗ) та її граф. Показано відмінності цієї моделі та її графу порівняно з відомим узагальненим графом системи ресурсів. Передбачається, що запропоновані модель та її граф дозволять розробити нові способи планування ресурсів НРКСЛЗ при реалізації різних видів паралелізації MCTS.

Ключові слова: паралельні та розподілені комп'ютерні системи, модель ресурсів комп'ютерної системи, задачі штучного інтелекту, дерева ігор, пошук в дереві методом Монте-Карло, MCTS, паралелізація MCTS.

Марченко А.А., Марченко А.И. Модель ресурсов неоднородной распределенной компьютерной системы с локальными связями и ее граф. На основе анализа метода MCTS и подходов к его параллелизации в статье предложены новая модель ресурсов неоднородной распределенной компьютерной системы с локальными связями (НРКСЛС) и ее граф. Показано отличия этой модели и ее графа по сравнению с известным обобщенным графом системы ресурсов. Предполагается, что предложенные модель и ее граф позволят разработать новые способы планирования ресурсов НРКСЛС при реализации различных видов параллелизации MCTS.

Ключевые слова: параллельные и распределенные компьютерные системы, модель ресурсов компьютерной системы, задачи искусственного интеллекта, деревья игр, поиск в дереве методом Монте-Карло, MCTS, параллелизация MCTS.

Marchenko O.O., Marchenko O.I. Resource model for heterogenous distributed computer system with local connections and its graph. Basing on the MCTS method analysis and approaches to its parallelization the paper proposes a new resource model of a heterogeneous distributed computer system with local connections (HDCSLC) and its graph. Differences of the model and its graph from the known generalized graph of a resource system are shown. It is supposed that the proposed model and graph would allow to create new resource planning techniques for HDCSLC with the aim of implementation of various kinds of MCTS parallelization.

Keywords: Parallel and distributed computer systems, resource model for computer system, artificial intelligence tasks, game trees, Monte-Carlo tree search method, MCTS, MCTS parallelization.

Рис. 1. Лім.14.

Постановка наукової проблеми.

Прискорення обчислення задач штучного інтелекту, в яких використовуються надзвичайно великі обсяги даних і час рішення яких зараз є занадто великим, було і залишається актуальною проблемою. Апробація нових методів, способів, алгоритмів, структур даних в цій галузі виконується, як правило, на складних інтелектуальних іграх (шахи, Go), а формами подання інформації в задачах штучного інтелекту зазвичай є форми у вигляді дерев послідовних рішень. Одним з методів штучного інтелекту, що був запропонований порівняно нещодавно і призначений для виконання швидкого пошуку правильних рішень у дереві інформації, є метод пошуку в дереві з використанням методу Монте-Карло (Monte Carlo Tree Search – MCTS) [1]. Вдосконалення методу MCTS з метою прискорення процесу пошуку кращих рішень може бути виконано як на алгоритмічному рівні, так і на рівні його розпаралелення. В той час, як ефективна паралелізація інших способів пошуку по дереву, що застосовуються в тих же задачах (наприклад, мінімакний пошук), є проблематичною [2], то, на противагу цим способам, MCTS дозволяє зробити розпаралелення достатньо ефективно, оскільки виконує пошук по дереву напіввипадковим чином і здійснює множину моделювань подальшого розвитку подій від заданого стану, які є незалежними. На сьогодні вже були запропоновані декілька методів паралелізації методу MCTS, які дозволяють отримати непогані результати, але це, очевидно, далеко ще не всі можливості розпаралелення MCTS, особливо враховуючи велику різноманітність сучасних багатокomp'ютерних, багато-процесорних та багатоядерних обчислювальних систем.

Особливий інтерес представляє пошук підходів до ефективної паралелізації MCTS для неоднорідних розподілених комп'ютерних систем з локальними зв'язками (НРКСЛЗ), оскільки такі системи можуть бути зібрані на основі комп'ютерного парку навіть невеликих організацій і тому є найбільш доступними для більшості дослідників.

Аналіз досліджень.

Для нашого дослідження важливі два різних напрямки попередніх досліджень. Одним з них є дослідження з підвищення ефективності пошуку MCTS на основі його базової теорії, а другим – підвищення ефективності цього пошуку на основі розпаралелення і використання сучасних комп'ютерних систем.

Найбільш фундаментальною узагальнюючою роботою з методу MCTS першого напрямку є огляд 2012 року, підготовлений десятима провідними вченими під керівництвом Камерона Броуне, що досліджують цей метод [1]. В наступні роки з'явилися не тільки нові модифікації та покращення існуючих способів, але й принципово нові варіанти реалізації MCTS. Ще один погляд на структуру та взаємозв'язки між ідеями існуючих способів покращення MCTS був запропонований авторами у вигляді класифікації [3, 4]. В подальшому в [5, 6, 7, 8] авторами були запропоновані прості та логарифмічні критерії типу «глибина-ширина» DWCs (Depth-Width Criteria) для контролю форми дерева пошуку при використанні методу Монте-Карло, а також спосіб покращення ефективності роботи MCTS, названий пошуком по дереву методом Монте-Карло з контролем форми дерева (Monte-Carlo Tree Search with Tree Shape Control – MCTS-TSC) [6]. Гілмар Фіннссон та Інгві Бьорнссон [9] досліджували ситуацію, чи віддає MCTS перевагу деревам з певною формою, більш глибоким чи більш широким, чи дотримує якогось певного балансу між глибиною і шириною. Крім того, використовуючи критерії типу «глибина-ширина», можна більш ефективно, з точки зору пошуку MCTS, виконувати паралелізацію роботи цього алгоритму. Тому, другим важливим напрямком попередніх досліджень є паралелізація роботи MCTS. Як зазначалося у [3, 4], за критерієм орієнтації способу паралелізації на конкретні апаратні ресурси, чи відсутності такої орієнтації, існуючі способи паралелізації пошуку в дереві методом MCTS можуть бути поділені на апаратно-незалежні способи та апаратно-орієнтовані способи. Для розпаралелення на апаратно-незалежному рівні були запропоновані три загальних підходи: листкова паралелізація, коренева паралелізація і деревна паралелізація [1, 10, 11].

Як відомо, схема роботи методу MCTS складається з наступних чотирьох послідовних етапів [1].

Етап 1: вибір.

Етап 2: розширення.

Етап 3: моделювання.

Етап 4: переобчислення.

Листкова паралелізація реалізується на етапі моделювання (рисунок 1) і, за визначенням даним в [10], представляє собою виконання множини одночасних моделювань від вершини-листа, яка була додана після закінчення процесу пошуку актуальної вершини, використовуючи політику дерева. Така паралелізація може бути ефективно реалізована на графічних процесорах GPU.

Кореневу паралелізацію [10, 11, 12] іноді називають багато-деревним MCTS, оскільки декілька дерев пошуку MCTS будуються одночасно, тобто розпаралелюються на поточному корені дерева. Такий вид паралелізації MCTS є більш високорівневим і доцільним для реалізації на окремих центральних процесорах CPU або комп'ютерах.

При деревній паралелізації виконується декілька одночасних ітерацій пошуку наступного ходу на одному й тому ж дереві [10, 13]. В цьому випадку виникає класична проблема паралельного програмування, що полягає у недопущенні одночасного доступу до дерева з різних потоків. Для цього потрібно забезпечити ексклюзивний доступ до певної частини дерева MCTS кожному потоку, що також більш ефективно можна реалізувати на більш високому рівні комп'ютерів, процесорів CPU та ядер.

Метою даної роботи є побудова моделі ресурсів неоднорідної розподіленої комп'ютерної системи з локальними зв'язками та її графу, яка є розвитком моделі архітектури grid-системи (GAM), запропонованої авторами раніше в рамках моделі динамічної паралелізації [8], з метою подальшого узагальнення процесу розпаралелення пошуку по дереву методом Монте-Карло на основі способу покращення MCTS з контролем форми дерева пошуку, базуючись на критеріях типу «глибина-ширина»

[5, 6, 7], що забезпечить більш ефективний розподіл наявних апаратних ресурсів для розпаралелення пошуку методом MCTS на різних етапах його роботи.

Модель ресурсів неоднорідної розподіленої комп'ютерної системи з локальними зв'язками та її граф

Для однозначності розуміння викладеного далі матеріалу сформулюємо визначення НРКСЛЗ.

Неоднорідна розподілена комп'ютерна система з локальними зв'язками (НРКСЛЗ) – це неоднорідна розподілена комп'ютерна система (НРКС) у вигляді локальної комп'ютерної мережі (LAN), комп'ютери якої об'єднані у LAN стандартними мережевими засобами.

Модель ресурсів (RM – Resource Model) НРКСЛЗ визначимо як

$$RM(t) = (NN, Node[1...NN], Link[1...NN, 1...NN], t),$$

де

NN(Nodes Member) – загальна кількість вузлів мережі НРКСЛЗ;

Node[1...NN] – масив дескрипторів вузлів мережі НРКСЛЗ, в якому кожен вузол цієї мережі визначається як

Node[i] = (Free(t), RAM, NCC, CF, NGPU, GPU[1...NGPU]), де

Free(Flag "free/busy") - прапорець «вільний/зайнятий» в момент часу t;

RAM - розмір оперативної пам'яті комп'ютера i-го вузла;

NCC(Number of CPU Cores) – кількість ядер центрального процесора i-го вузла;

CF (Cores Frequency) – частота ядер центрального процесора i-го вузла;

NGPU(Number of GPU) – кількість GPU плат i-го вузла;

GPU[1...NGPU] – масив дескрипторів i-го вузла, якщо він має GPU тип;

(тобто NGPU>0), де

GPU[ii] = (Free(t), M, NGC, GCF...?)*, де

Free(Flag "free/busy") – прапорець «вільний/зайнятий» в момент часу t;

M(Memory) – розмір пам'яті GPU;

NGC(Number of GPU Cores) – кількість ядер GPU;

GCF(GPU Core Frequency) - частота ядер GPU;

* – при необхідності кількість параметрів GPU у дескрипторі GPU[ii] може бути збільшена.

Link[1...NN, 1...NN] – масив дескрипторів ліній зв'язку між вузлами мережі НРКСЛЗ, в якому кожна лінія зв'язку (тип кабеля, мережна карта, протокол передачі даних) між j-м (j=1...NN) та k-м (k=1...NN) вузлами якої визначається як,

Link[j, k] = (B)**, де

B(Bandwidth) – пропускна здатність лінії зв'язку.

** – в даній моделі обмежимося тільки одним параметром в дескрипторі, але при необхідності кількість параметрів у дескрипторі може бути розширена.

Визначену модель ресурсів (RM) НРКСЛЗ можна також представити у вигляді нижченаведеного графа моделі ресурсів GRM (Graph of Resource Model) (рис.1).

Зазначимо, що при визначенні цього графа оперувати тільки абстрактними ваговими коефіцієнтами, як зроблено для узагальненого графа в [14] буде недостатньо, оскільки в даному випадку дуже важливими є також якісні характеристики ресурсів НРКСЛЗ.

Розглянемо граф GRM детально.

0-й рівень – це рівень головного комп'ютера (Host), на якому починається пошук MCTS, виконується основний його цикл та визначається результат пошуку.

Зауваження: в якості головного комп'ютера використовувати комп'ютер з потужним графічним процесором недоцільно.

1-й рівень – це рівень вузлових комп'ютерів (Nodes) мережі НРКСЛЗ.

2-й рівень – це рівень ресурсів вузлових комп'ютерів (GPUs, Cores) мережі НРКСЛЗ.

3-й рівень – це рівень ресурсів графічних процесорів, які можуть бути встановлені в певних вузлових комп'ютерах НРКСЛЗ.

Відзначимо, що граф моделі ресурсів (GRM) НРКСЛЗ відповідає узагальненому графу системи ресурсів, визначеному в [14]. Граф GRM, так само, як і в [14] визначається четвіркою категорій:

$$GR = (VR, ER, WVR, WER)$$

з наступними відповідностями, які потрібні для більш детального і спеціалізованого погляду на ресурси:

- єдиній множині однотипних вершин VR в графі GRM відповідають вершини чотирьох категорій (Node, Core, GPU, GCore), але в даній відповідності їх також можна розглядати як однотипні, оскільки категорія вершини визначається її дескриптором (вагою), а не власне самою вершиною.

- єдиній множині однотипних ребер ER в графі GRM, в принципі, відповідають ребра чотирьох категорій, але їх так само, як і вершини, можна також розглядати як однотипні, оскільки категорія ребра також фактично визначається його дескриптором (вагою).

- єдиній множині однотипних ваг вершин WVR в графі GRM відповідають ваги чотирьох категорій. Вага вершини категорії Node визначається її дескриптором Node[i]. Вага вершини категорії Core визначається значенням параметра частоти цього ядра CF в дескрипторі Node[i]. Вага вершини категорії GPU визначається її дескриптором GPU[ii]. Вага вершини категорії GCore визначається значенням параметра частоти графічного процесора GCF в дескрипторі GPU[ii].

- єдиній множині однотипних ваг ребер WER в графі GRM в загальному випадку можуть відповідати ваги чотирьох категорій. Вага ребра, що з'єднує дві вершини j та k категорії Node, визначається дескриптором Link[j, k]. Вагу інших ребер в графі GRM можна не враховувати, оскільки призначення завдань на ці ресурси при розпаралеленні MCTS буде залежати тільки від категорій відповідних вершин (Core, GPU, GCore).

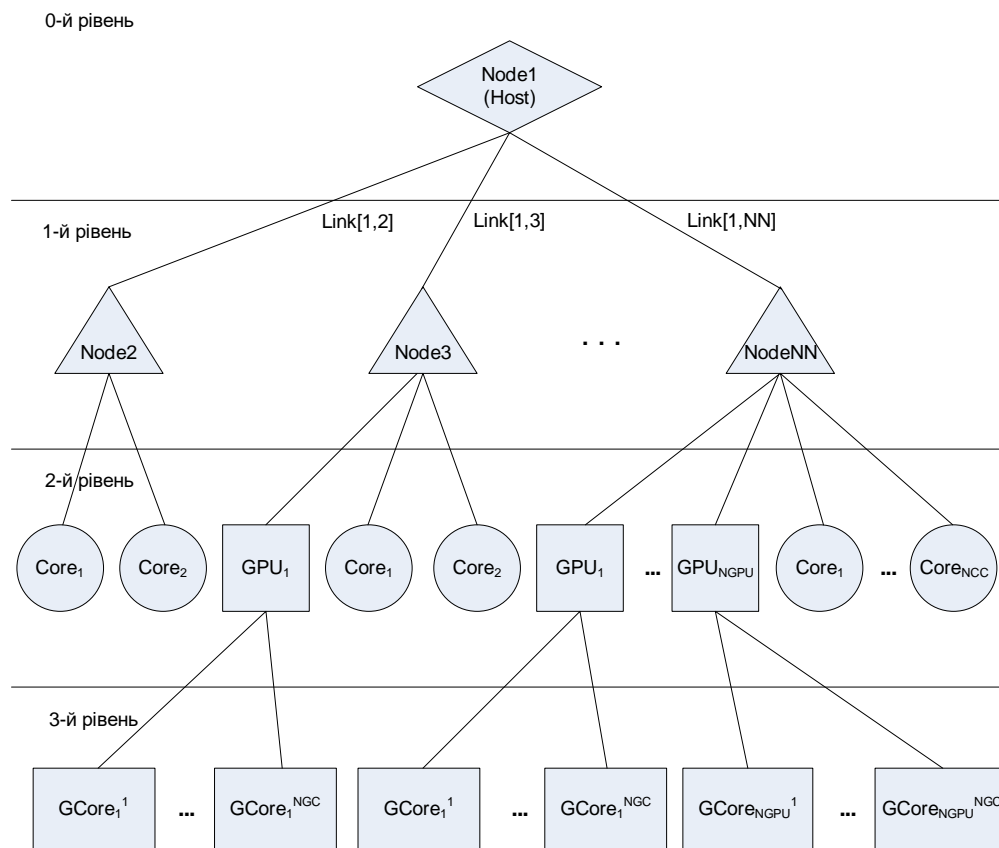


Рис.1. Граф моделі ресурсів GRM

Висновки

Паралелізація пошуку в дереві з використанням методу MCTS може бути виконана згідно декількох різних підходів. На основі аналізу методу MCTS та підходів до його паралелізації у статті запропонована нова модель ресурсів неоднорідної розподіленої комп'ютерної системи з локальними зв'язками

(НРКСЛЗ) та її граф. Показано відмінності цієї моделі та її графу порівняно з узагальненим графом системи ресурсів, визначеному в [14].

Подальшим дослідженням в даному напрямку може бути розробка різних способів планування ресурсів НРКСЛЗ при реалізації різних видів паралелізації МСТS (кореневої, деревної, листкової) на основі запропонованих моделі ресурсів НРКСЛЗ та її графу.

Список бібліографічних посилань

1. Cameron Browne. A Survey of Monte Carlo Tree Search Methods / Cameron Browne, Edward Powley, Daniel Whitehouse, and others // IEEE Trans. on Computational Intelligence and AI in Games. – vol. 4. – no. 1. – March 2012. – P. 1-49.
2. Schaeer, J. The APHID Parallel algorithm / Schaeer, J., Brockington M. G // Proceedings of the 8th IEEE Symposium on Parallel and Distributed Processing. – 1996. – P. 428-432.
3. Марченко О. І. Структура та критерії класифікації способів реалізації та покращення пошуку по дереву методом Монте-Карло. / Марченко О.І., Марченко О.О., Орлова М.М. // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – 2015. – № 21. – С. 51–57.
4. Марченко О.І. Класифікація способів реалізації та покращення пошуку по дереву методом Монте-Карло / Марченко О.І., Марченко О.О., Орлова М.М. // Штучний інтелект. – 2016. – №2(72). – С. 59-69.
5. Марченко О.О. Критерій «глибина-ширина» для контролю форми дерева пошуку при використанні методу Монте-Карло. / Марченко О.І., Марченко О.О. // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – 2016. – № 24-25. – С.42-47.
6. Oleksandr I. Marchenko. Monte-Carlo Tree Search with Tree Shape Control. / Oleksandr I. Marchenko, Oleksii O. Marchenko // 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON). Conference Proceedings. May 29 – June 2, 2017., Kyiv, Ukraine. – 2017. – P. 812-8173.
7. Марченко О.О. Логарифмічний критерій контролю форми дерева для покращення пошуку по дереву методом Монте-Карло. / Марченко О.І., Марченко О.О. // Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – 2017. – № 27. – с.37-43.
8. Марченко О.О. Спосіб динамічного розпаралелення пошуку в дереві методом Монте-Карло в grid-системах. / Марченко О.І., Марченко О.О. // Вимірювальна та обчислювальна техніка в технологічних процессах. – 2017. – №3(59). – с.194-200.
9. Hilmar Finnsson. Game-Tree Properties and MCTS Performance. / Hilmar Finnsson and Yngvi Björnsson // GIGA 2011: Proceedings of the 2nd International General Game Playing Workshop, 2011, pp.23-30.
10. G. M. J.-B. Chaslot. Parallel Monte-Carlo Tree Search / G. M. J.-B. Chaslot, M. H. M. Winands, and H.J. van den Herik // Proc. Comput. And Games, LNCS 5131, Beijing, China. – 2008. P.60–71.
11. T. Cazenave. On the Parallelization of UCT / T. Cazenave and N. Jouandeu // Proc. Comput. Games Workshop, Amsterdam, Netherlands. – 2007. – P. 93–101.
12. Y. Soejima. Evaluating Root Parallelization in Go / Y. Soejima, A. Kishimoto, and O. Watanabe // IEEE Trans. Comp. Intell. AI Games. – vol. 2. – no. 4. – 2010. – P. 278–287.
13. A. Bourki. Scalability and Parallelization of Monte-Carlo Tree Search / A. Bourki, G. M. J.-B. Chaslot, M. Coulm, V. Danjean, H. Doghmen, J.-B. Hoock, T. Herrault, A. Rimmel, F. Teytaud, O. Teytaud, P. Vayssie`re, and Z. Yu // Proc. Int. Conf. Comput. and Games, LNCS 6515, Kanazawa, Japan. – 2010. – P. 48–58.
14. Симоненко В.П. Организация вычислительных процессов в ЭВМ, комплексах, сетях и системах. // Киев, ТОО «ВЕК+». – 1997. – 302 с.

References

1. Cameron Browne. A Survey of Monte Carlo Tree Search Methods / Cameron Browne, Edward Powley, Daniel Whitehouse, and others // IEEE Trans. on Computational Intelligence and AI in Games. – vol. 4. – no. 1. – March 2012. – P. 1-49.
2. Schaeer, J. The APHID Parallel algorithm / Schaeer, J., Brockington M. G // Proceedings of the 8th IEEE Symposium on Parallel and Distributed Processing. – 1996. – P. 428-432.
3. O.I.Marchenko. Structure and criteria for classification of techniques for implementation and improvement of Monte-Carlo tree search./ O.I.Marchenko, O.O.Marchenko // Komp'uterno-integrovani tekhnologii: osvita, nauka, vyrobnytstvo, No.21, 2015, pp.51-57 (in Ukrainian).
4. O.I.Marchenko. Classification of Monte-Carlo Tree Search Enhancement Techniques Oriented to Specifics of the Method./ O.I.Marchenko, O.O.Marchenko, M.M.Orlova // Artificial Intelligence, No.2(72), 2016, pp.59-69 (in Ukrainian).
5. O.O.Marchenko. "Depth-Width" Criterion for Control of the Search Tree Shape Using Monte-Carlo Method./ O.O.Marchenko, O.I.Marchenko // Komp'uterno-integrovani tekhnologii: osvita, nauka, vyrobnytstvo, No.24-25, 2016, pp.42-47 (in Ukrainian).
6. Oleksandr I. Marchenko. Monte-Carlo Tree Search with Tree Shape Control. / Oleksandr I. Marchenko, Oleksii O. Marchenko // 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON). Conference Proceedings. May 29 – June 2, 2017., Kyiv, Ukraine. – 2017. – P. 812-8173.
7. O.O.Marchenko. Logarithmic criterion for tree shape control for improvement of the Monte-Carlo tree search method./ O.O.Marchenko, O.I.Marchenko // Komp'uterno-integrovani tekhnologii: osvita, nauka, vyrobnytstvo, No.27, 2017, pp.37-43 (in Ukrainian).
8. O.O.Marchenko. Technique for MCTS dynamic parallelization for grid-systems./ O.O.Marchenko, O.I.Marchenko // Vymiriuvalna ta obchysliuvalna tekhnika v tekhnolohichnykh procesakh, No.3(59), 2017, pp.37-43 (in Ukrainian).
9. Hilmar Finnsson. Game-Tree Properties and MCTS Performance. / Hilmar Finnsson and Yngvi Björnsson // GIGA 2011: Proceedings of the 2nd International General Game Playing Workshop, 2011, pp.23-30.

10. G. M. J.-B. Chaslot. Parallel Monte-Carlo Tree Search / G. M. J.-B. Chaslot, M. H. M. Winands, and H.J. van den Herik // Proc. Comput. And Games, LNCS 5131, Beijing, China. – 2008. P.60–71.
11. T. Cazenave. On the Parallelization of UCT / T. Cazenave and N. Jouandeau // Proc. Comput. Games Workshop, Amsterdam, Netherlands. – 2007. – P. 93–101.
12. Y. Soejima. Evaluating Root Parallelization in Go / Y. Soejima, A. Kishimoto, and O. Watanabe // IEEE Trans. Comp. Intell. AI Games. – vol. 2. – no. 4. – 2010. – P. 278–287.
13. A. Bourki. Scalability and Parallelization of Monte-Carlo Tree Search / A. Bourki, G. M. J.-B. Chaslot, M. Coulm, V. Danjean, H. Doghmen, J.-B. Hoock, T. Herrault, A. Rimmel, F. Teytaud, O. Teytaud, P. Vayssi`re, and Z. Yu // Proc. Int. Conf. Comput. and Games, LNCS 6515, Kanazawa, Japan. – 2010. – P. 48–58.
14. Simonenko V.P. Organization of computing processes in computers, complexes, networks, and systems. // Kiev, TOO "VEK+". – 1997. – 302 p.

DOI: 10.36910/6775-2524-0560-2020-39-15

УДК: 004.05.

Поплавська Ганна Володимирівна, викладач

Комунальний заклад вищої освіти Луцький педагогічний коледж Волинської обласної ради

АНАЛІЗ ЗАСТОСУВАННЯ ОНЛАЙН СЕРВІСУ GOOGLE CLASSROOM ДЛЯ ОРГАНІЗАЦІЇ ДИСТАНЦІЙНОГО НАВЧАННЯ

Поплавська Г.В. Аналіз застосування онлайн сервісу Google Classroom для організації дистанційного навчання. У статті проаналізовано основні можливості застосування сервісу Google Classroom для організації дистанційного навчання та подальшого його використання.

Ключові слова: організація навчального процесу, Google Classroom, Moodle, освітня платформа, дистанційне навчання, інтернет-ресурс.

Поплавская А. В. Анализ применения онлайн сервиса Google Classroom для организации дистанционного обучения. В статье проанализированы основные возможности применения сервиса Google Classroom для организации дистанционного обучения и дальнейшем его использовании

Ключевые слова: организация учебного процесса, Google Classroom, Moodle, образовательная платформа, дистанционное обучение, интернет-ресурс.

Poplavskaya G.V. Analysis of the use of the Google Classroom online service for distance learning. The article examines the main features of using Google Classroom to organize and continue using distance learning.

Keywords: educational process organization, Google Classroom, Moodle, educational platform, distance learning, online resource.

Постановка наукової проблеми.

В сучасному світі цифрових технологій та різноманітних гаджетів та дивайсів, традиційні методи організації навчального процесу втрачають свою актуальність і потребують активного інтегрування у віртуальне освітнє середовище. Використання інтернет-ресурсів у закладах освіти різного рівня та профілю вже досить давно стало загальною нормою й стандартною міжнародною практикою, як, наприклад, дистанційна освіта. Така трансформація дозволяє урізноманітнити навчання, перенести частину занять в онлайн простір, що не потребує постійної фізичної присутності студента та викладача в одному місці у певний проміжок часу.

Масштаби впровадження інтернет-технологій в освітній процес визначаються розумним прагненням його учасників зробити навчання ефективнішим [1] та цікавішим, з одного боку, а також менш трудомістким та часозатратним, з іншого.

Сприяє цьому процесу постійне розроблення різноманітних віртуальних платформ та сервісів, які виробники програмних продуктів вільно розповсюджують в академічному середовищі. Чи потрібно викладачу постійно освоювати нові «пропозиції електронного викладання» та як їх застосовувати? Наскільки цікавим та простим буде для студента певний веб-сервіс? Такі питання особливо гостро постали у сьогоднішні.

Аналіз досліджень і публікацій. Останнім часом проблемі дистанційного навчання приділяється велика увага не лише у засобах масової інформації, а й в науковій літературі. Дистанційна система навчання знаходиться у центрі уваги наукових кіл у сьогоднішні. Сучасні тенденції свідчать про подальшу активізацію досліджень у цій сфері. Зокрема, теоретичними, методологічними та методичними проблемами дистанційного навчання займалися такі вітчизняні науковці, як В. Кухаренко, С. Вітвицька, В. Ясулайтіс, Е. Полат, А. Петров, О. Тищенко, В. Биков, О. Овчарук, О. Пінчук та інші. Проблемами організації дистанційної освіти займалися й зарубіжні науковці, серед них Деллінг Р. (Німеччина), Рамбле Г., Кіган Д., Сімонсон М., Мур М., Кларк А., Томпсон М. (США) та інші.

Докладний аналіз проектів Росії, Німеччини, Чехії, Австралії, Китаї, Ізраїлі, Африці, Сінгапурі, Бразилії, Колумбії, Азербайджану та США показали, що хмарні сервіси використовуються вчителями й студентами зарубіжні країни для: удосконалення навчального процесу, організації доступу до навчальних матеріалів, ведення графіків роботи, зберігання навчальних програм, активізації діяльності учнів, організації навчання, забезпечення учнів домашніми завданнями, організації дистанційного та змішаного навчання [2].

Тим не менш, залишається відкритим питання організації дистанційного навчання на базі різноманітних Інтернет – сервісів.

Формулювання цілей статті. Мета статті - аналіз функціональних можливостей та особливостей сервісу Google Classroom із позиції ефективності організації дистанційно навчального

процесу, узагальнення власного досвіду використання Google Classroom у викладанні дисциплін інформативного спрямування.

Основні результати досліджень. Зараз все частіше студенти, учні, працівники майже всіх галузей переносять свою активну діяльність в глобальну павутину. Особливо це стало актуальним у сьогоденні.

Та й бурхливий розвиток інформаційних технологій вплинув на всі сфери діяльності людини також, адже це виклик часу. Не залишився осторонь і освітній процес - було сформовано «цифровий освітній простір». Гостро постало питання на використання дистанційного навчання. Частина освітян впоралась із викликом дистанційної освіти, але частина – взяли лише «висоту» Viber. Традиційні способи здобуття вищої освіти набули нових рис, поєднуючи лекційно-практичну систему навчання з можливостями дистанційного навчання. Виникла потреба в організації середовища, в якому обидві сторони освітнього процесу (здобувач освіти та викладач) могли б комфортно співпрацювати.

Розвитку дистанційного навчання сприяє постійне розроблення нових освітніх платформ та сервісів, які дають широкі можливості. Для ефективного впровадження онлайн освіти важливо правильно обрати платформу, на базі якої вона буде здійснюватися.

Найбільш популярна [3] система організації дистанційного навчання - платформа Moodle (Modular Object-Oriented Dynamic Learning Environment). Moodle - це модульне об'єктно-орієнтоване динамічне навчальне середовище, яке називають також системою управління навчанням (LMS), системою управління курсами (CMS), віртуальним навчальним середовищем (VLE) або просто платформою для навчання, яка надає викладачам та здобувачам освіти розвинутий набір інструментів для комп'ютеризованого навчання.

На даний момент Moodle вже має 129 мільйонів користувачів в усьому світі й продовжує розвиватися. Система безкоштовна, але потребує досить продуктивного власного веб-сервера, що не завжди доступно навіть для освітніх установ.

Серед різноманіття подібних систем особливої уваги заслуговує сервіс, котрий у 2014 році запустила компанія Google, а саме Google Classroom (Google Клас) [4], який надає зручний інтерфейс для створення й керування навчальними курсами. Google Classroom – це інструмент, що зв'яже Google Docs, Google Drive і Gmail. Завдяки платформі Classroom можна організувати проектну роботу, перевернуте навчання, дистанційне навчання.

Це відкриває нові можливості у викладанні: допомагає створювати і впорядковувати завдання, перевіряти, оцінювати завдання та виставляти оцінки, коментувати і організовувати ефективне спілкування із здобувачами освіти в режимі реального часу або в режимі дистанційного навчання, реалізувати індивідуальний підхід у навчанні, проводити онлайн обговорення, консультації, коментувати.

Серед різноманітних віртуальних освітніх платформ сервіс Google Classroom вирізняється, перш за все, простотою у використанні. Він не потребує спеціального навчання для створення курсу, а також надає користувачам зручний інтерфейс для керування навчальним процесом. Корпорація Google Inc. заявила: «Ми [4] розробили цей сервіс разом із викладачами, щоб допомогти їм заощаджувати час, легко й швидко організовувати заняття й ефективно спілкуватися з учнями».

Головними особливостями Google Classroom можна вважати:

- просте налаштування. Викладач може організовувати декілька курсів одночасно, запрошувати студентів та інших викладачів, планувати навчальний процес;
- економія часу та паперу. Роздавати, контролювати виконання, перевіряти завдання можна в одному сервісі;
- зручність. Студенти можуть переглядати завдання, планувати строки їх виконання відповідно до календарного плану курсу, вчасно надсилати виконані роботи для перевірки;
- доступність і безпека. Клас доступний безкоштовно. У цьому сервісі відсутня реклама, а матеріали та особисті дані користувачів не потрапляють до інших сервісів.[5]

Для створення свого курсу на платформі Google Classroom викладачу потрібно увійти з особистого акаунту Google до відповідного додатка головного меню в браузері (рис. 1).

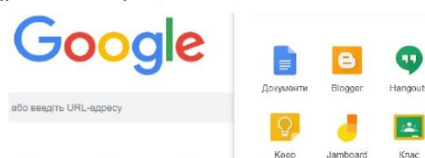


Рис.1 Перший етап створення курсу у сервісі Google Classroom

Після обрання статусу викладача Ви можете почати створення та налаштування власного курсу (рис. 2).

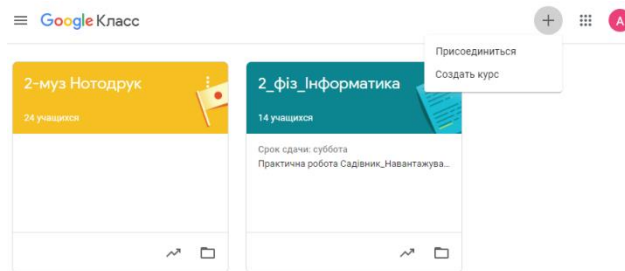


Рис 2. Приклад створення курсу у сервісі Google Classroom

Сервіс Google Classroom надає різноманітні шаблони та налаштування: від кольорового оформлення теми курсу до створення його наповнення (рис. 3).

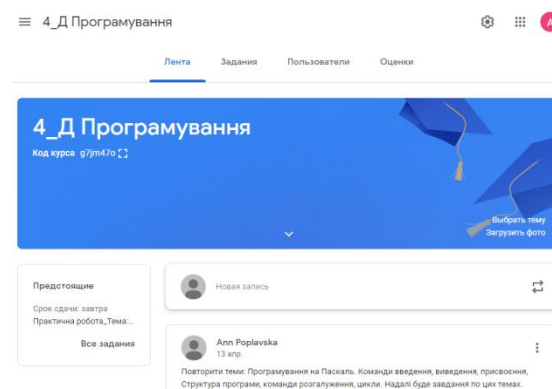


Рис.3 Приклад оформлення курсу у сервісі Google Classroom

За допомогою меню викладач може структурувати курс за розділами, визначивши їх тематику. Далі по кожній темі створити календарний план, в якому вказати терміни вивчення відповідного матеріалу, а також виконання обов'язкових завдань (рис. 4).

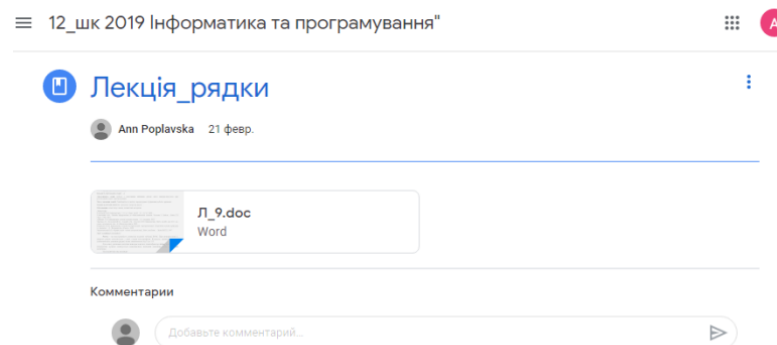


Рис.4 Приклад розробки курсу Інформатика та програмування у сервісі Google Classroom

Можливості платформи Google Classroom в наповненні курсу досить різноманітні. Це й аутентичні лекції в паперовому або відео-форматі, і відповідні розділи підручників, а також записи з YouTube. Крім того, сервіс надає можливість спільного викладання. Тобто, для кращого засвоєння певного матеріалу викладач може запросити до 20 інших освітян.

Важлива перевага Google Classroom - це інтегрованість з іншими сервісами Google, завдяки чому викладач має можливість завантажувати в Клас інформацію у будь-якому форматі. В цьому сенсі Google Classroom потребує від викладача добре продуманої програми курсу. Тільки за такої умови сервіс надасть нові можливості в організації навчального процесу.

Щоб створити завдання, потрібно обрати відповідний пункт меню. Далі в діалоговому вікні вказати тематику завдання, дати пояснення щодо оформлення та терміни виконання, прикріпити саме

завдання. Існує можливість створювати індивідуальні завдання для кожного студента, а також установлювати шкалу оцінювання. Сервіс Google Classroom запропонував нову форму завдання - тести. Можна запланувати публікацію будь-якого матеріалу. Кожна публікація (об'ява, матеріал, завдання) з'являється на сторінці «Лента», а студенти отримують сповіщення на свою електронну адресу. Для цього вони можуть використовувати навіть свої гаджети з доступом до інтернет-ресурсів.

Сторінка «Пользователи» надає інформацію про склад курсу, а також можливість електронного листування. Щоб приєднати студента до курсу, викладач має запросити його або надати код доступу. Запрошення автоматично надходить на електронну пошту.

Щоб освоїти Google Classroom, не потрібно багато часу. В інтернеті пропонують масу уроків по створенню Класу.

Особливо хотілося б зазначити, що студенти без проблем залучились до віртуального навчання.

Важлива перевага платформи Google Classroom - продуктивна комунікація. У Клас викладачі можуть публікувати завдання, розсилати оголошення й починати обговорення, а здобувачі освіти - обмінюватися матеріалами, додавати коментарі й спілкуватися через електронну пошту.

Інформація про здані роботи постійно оновлюється, що дозволяє викладачам оперативно перевіряти завдання, ставити оцінки й додавати коментарі. Мої особисті враження від досвіду роботи з Google Classroom дуже позитивні. Сервіс не є альтернативою живого спілкування зі студентами. Він модернізує взаємодію учасників навчального процесу. Немає потреби зберігати купу завдань протягом навчального року та пам'ятати терміни їх виконання. Існує можливість оперативно домовитись про консультацію або додаткове заняття.

Використання сервісу можливе на будь-якому гаджеті, що має вихід в Інтернет. Це спрощує роботу викладача та підвищує зацікавленість студента. Адже він усвідомлює свою невідривність від навчальної дисципліни і бачить зв'язок з життям.

Основна перевага Google Класу, порівняно з іншими сервісами аналогами такими, як Edmodo (освітній сайт, який являє собою усічену соціальну мережу за типом Facebook або ВКонтакте, яка дозволяє спілкуватися викладачам та студентам, об'єднавшись навколо процесу навчання) [6], Moodle (платформа для навчання, яка надає викладачам, учням, студентам та адміністраторам розвинутий набір інструментів для комп'ютеризованого навчання, в тому числі дистанційного) [7] та ін., полягає в простоті використання та інтуїтивно зрозумілого інтерфейсу.

Висновки та перспективи подальших досліджень. Наприкінці ще раз зазначу – навчання онлайн, зокрема використання сервісу Google Classroom має ряд переваг:

- навчання будь-де та будь-коли. Онлайн це зручно – все що потрібно, щоб розпочати – бажання та доступ до інтернету.

- доступність навчальних матеріалів. Здобувач освіти отримує усі необхідні матеріали, та має змогу повернутися до них у будь-який момент. Тож проблема нестачі чи відсутності підручників та методичок зникає.

- зворотній зв'язок. Здобувач освіти завжди має онлайн зв'язок з викладачем або репетитором, які контролюють виконання завдань та допомагають у їх розв'язанні.

- зручність для викладача. Вчителі або репетитори, що займаються педагогічною діяльністю дистанційно, можуть приділяти увагу більшій кількості студентів і працювати більш ефективно, витрачаючи менше часу та зусиль.

- на радість батькам. Завдяки гейміфікації, інтерактивним урокам та контролю зі сторони вчителя - дитина поринає в процес вивчення матеріалу дуже швидко та з величезним інтересом.

Переваг онлайн-навчання можна привести багато: зростання інтересу студентів та учнів до нового цікавого методу навчання, економія ресурсів всіх учасників навчального процесу, автоматизація контролю виконання завдань тощо.

Науковий прогрес рухається із шаленою швидкістю, прогресивні країни світу поступово діджиталізуються, тому нам так само потрібно оновлюватися та рухатися вперед.

Діти сьогодні зростають у цифровому світі і для них цілком природним є цифрове середовище, яке вони використовують для отримання та обміну інформацією.

Отже, для них не є проблемою навчатися онлайн дистанційно, навпаки – це нова цікава пригода для дитини!

Тож, не вагайтесь – використовуйте дистанційне навчання для своєї дитини чи класу вже зараз.

Список бібліографічного опису

1. Варенко Т. К. Гибридная система организации учебного-педагогического процесса с использованием веб-сервиса Google «Класс». Вісник Харківського національного університету імені В. Н. Каразіна : зб. наук. праць, Харків: ХНУ ім. В. Н. Каразіна, 2014. (№ 1125). вип. 79. С. 86–93.
2. Литвинова С. Г. Проектування хмаро-орієнтованих навчальних середовищ загальноосвітніх навчальних закладів. Зарубіжний досвід [Електронний ресурс] Інформаційні технології і засоби навчання: електронне наукове фахове видання. 2014. Том 41. №3 С. 10-27. Режим доступу: <http://journal.iitta.gov.ua/index.php/itlt/article/view/1052>
3. Moodle. Вікіпедія. Вільна енциклопедія. [Електронний ресурс] Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Moodle>.
4. Google Classroom - онлайн класс для эффективного обучения [Електронний ресурс] Вебдванольные заметки. Блог об интернет сервисах и проектах. Режим доступа: <http://www.web2me.ru/google/classroom/>.
5. Google Класс. Справка-Класс. [Електронний ресурс] 2015. Режим доступу до ресурсу: https://support.google.com/edu/classroom/answer/6020279?hl=ru&ref_topic=6020277.
6. Osvita.ua. Сучасні освітні інструменти для вчителів: [Електронний ресурс]. Режим доступу до ресурсу: <http://osvita.ua/school/46016/>
7. Moodle.org. Що таке Moodle: [Електронний ресурс]. Режим доступу до ресурсу: <https://moodle.org/mod/page/view.php?id=8174>
8. Пліш І.В. Використання інформаційно-комунікаційних технологій управління якістю освіти в школах приватної форми власності. [Електронний ресурс] Інформаційні технології і засоби навчання. 2012. №1 (27). Режим доступу до ресурсу: <http://journal.iitta.gov.ua>.

References

1. Varenko T. K. Hybrid system of organization of educational and pedagogical process using the Google web-service "Class". Bulletin of the Kharkiv National University named after VN Karazin: Coll. of sciences. , Kharkov: KhNU them. V. N. Karazin, 2014. (№1125). vol. 79. P. 86–93.
2. Litvinova S.G. Design of cloud-oriented educational environments of general educational institutions. Foreign experience [Electronic resource] Information technologies and teaching aids: an electronic scientific professional publication. 2014. Vol 41. №3 S. 10-27. Access mode: <http://journal.iitta.gov.ua/index.php/itlt/article/view/1052>
3. Moodle. Wikipedia. Free Encyclopedia. [Electronic resource] Mode of access to the resource: <https://en.wikipedia.org/wiki/Moodle>.
4. Google Classroom - An Online Classroom for Effective Learning [Online Resource] Web Demo. Blog about internet services and projects. Access mode: <http://www.web2me.ru/google/classroom/>.
5. Google Classroom. Class Help. [Online Resource] 2015. Resource Access Mode: https://support.google.com/edu/classroom/answer/6020279?hl=en&ref_topic=6020277.
6. Osvita.ua. Modern Educational Tools for Teachers: [Electronic resource]. Resource access mode: <http://osvita.ua/school/46016/>
7. Moodle.org. What is Moodle: [Electronic resource]. Resource Access Mode: <https://moodle.org/mod/page/view.php?id=8174>
8. Plish I.V. Use of information and communication technologies for quality management of education in private property schools. [Electronic resource] Information technology and learning tools. 2012. №1 (27). Resource access mode: <http://journal.iitta.gov.ua>.

Рецензенти: Марчук Сергій Степанович, кандидат педагогічних наук, викладач-методист, завідувач кафедри педагогіки, психології та окремих методик Луцького педагогічного коледжу;

Муляр Вадим Петрович, кандидат педагогічних наук, доцент кафедри експериментальної фізики та інформаційно-вимірювальних технологій Східноєвропейського національного університету імені Лесі Українки.

DOI: 10.36910/6775-2524-0560-2020-39-16

УДК: 681.2.084

Приходько А.О.,

Приходько Олексій Сергійович, к.т.н.

<https://orcid.org/0000-0001-8496-1295>

Луцький національний технічний університет

РОЗРОБКА УНІВЕРСАЛЬНОЇ УСТАНОВКИ ДЛЯ МЕХАНІЧНИХ ВИПРОБУВАНЬ МАТЕРІАЛІВ

Приходько А.О., Приходько О.С. Розробка універсальної установки для механічних випробувань матеріалів.

На основі використання контролера Arduino Nano розроблено компактну та недорогу випробувальну установку, яка дозволяє проводити статичні випробування матеріалів з максимальним навантаженням 200 Н та може використовуватись з навчально-демонстраційною метою під час викладання уроків фізики в школі, курсів механіки, опору матеріалів, теорії пружності тощо у ЗВО.

Ключові слова: випробувальні установки, Arduino Nano, тензодатчик, кроковий двигун

Приходько А.О., Приходько О.С. Разработка универсальной установки для механических испытаний материалов. На основе использования контролера Arduino Nano разработана компактная и недорогая испытательная установка, позволяющая проводить статические испытания материалов с максимальной нагрузкой 200 Н и которая может использоваться в учебно-демонстрационных целях во время преподавания уроков физики в школе, курсов механики, сопротивления материалов, теории упругости и т.д. в высших учебных учреждениях.

Ключевые слова: испытательные установки, Arduino Nano, тензодатчик, шаговый мотор

Prykhodko A.O., Prykhodko O.S. Development of a universal device for mechanical testing of materials. Based on the use of the Arduino Nano controller, a compact and inexpensive universal testing machine has been developed. It allows static testing of materials with a maximum load of 200 N and can be used for educational and demonstration purposes in the teaching of physics lessons at school, courses in mechanics, resistance of materials, theory of elasticity, etc. in higher education.

Keywords: universal testing machines, Arduino Nano, strain gauge, stepper motor

Постановка проблеми. На сучасному етапі розвитку технологій кожного року створюються нові матеріали з новими можливостями та параметрами. Для визначення характеристик цих матеріалів, а також для контролю якості використовуються універсальні випробувальні машини. Вони дозволяють визначити якісні та кількісні механічні характеристики матеріалу зразків, якість покриттів, критичні навантаження елементів конструкцій та приладів. Наразі є досить багато різновидів випробувального устаткування: від багатотонних великогабаритних розривних машин та пресів до настільних установок з навантаженням до 100 Н. Випробувальні машини використовуються також і в навчальному процесі для демонстрації закономірностей та діаграм деформування матеріалів. На жаль, в лабораторіях університетів часто можна зустріти досить старе устаткування (60-70-ті роки минулого століття), що не мають інтерфейсу під'єднання до ПК та не дають можливості швидко будувати графіки та візуалізувати результати досліджень у реальному часі. Останнє є дуже важливим для навчального процесу, де часто необхідно демонструвати основні принципи випробувань і проводити досліди (однак без високих вимог до точності). У зв'язку з цим було вирішено виготовити бюджетну настільну випробувальну установку з максимальним навантаженням до 200Н.

Аналіз досліджень. На даний час існує низка готових настільних випробувальних установок з невеликим максимальним навантаженням. Наприклад, випробувальне устаткування від фірми A&D [2], модельний ряд МСТ з максимальним навантаженням від 100 до 2500Н та широким спектром захватів, дозволяє проводити низку досліджень на розтяг, стиск, згин, зминання тощо. Однак через відносно високу вартість, впровадження їх в навчальний (та науковий) процес є проблематичним.

Виклад основного матеріалу та обґрунтування отриманих результатів дослідження. Для вибору схеми навантаження у випробувальній установці було проаналізовано переваги та недоліки існуючих конструкцій та розглянуто вимоги до проведення випробувань [2,5,6,8]. Для двоколонної (рис. 1, а) схеми перевагами являється висока навантажувальна здатність, симетричність навантаження, та хороша жорсткість конструкції. Однак, для забезпечення симетричності навантаження необхідна чітка синхронізація переміщень по двох колонах. Це можна реалізувати, наприклад, ремінною передачею, як показано на рисунку. Консольна конструкція (рис. 1, б) передбачає лише один різьбовий вал, простіша в реалізації, компактніша та забезпечує дещо кращий доступ до випробувальних зразків. Проте, внаслідок несиметричного навантаження рухомої частини із захватом, дає суттєвий згинальний момент на вал та лінійні підшипники. Для нашої конструкції був вибраний проміжний варіант (рис. 1, в). Для його реалізації використано один різьбовий вал та дві лінійних направляючих. Враховуючи невелике максимальне навантаження (200 Н), яке має

забезпечувати установка, простоту обробки та виготовлення фанерної конструкції, а також значно нижчу вартість в порівнянні з алюмінієвим профілем, матеріалом конструкції було вибрано фанеру 10 мм. Розрахунок абсолютної деформації показав величину ~18 мкм при максимальному навантаженні. Для розрахунку приймали модуль пружності $E = 9000$ МПа (семишарова фанера впоперек волокон[7]). Відносна деформація рівна 0,0072%, чого більш ніж достатньо для забезпечення жорсткості конструкції.

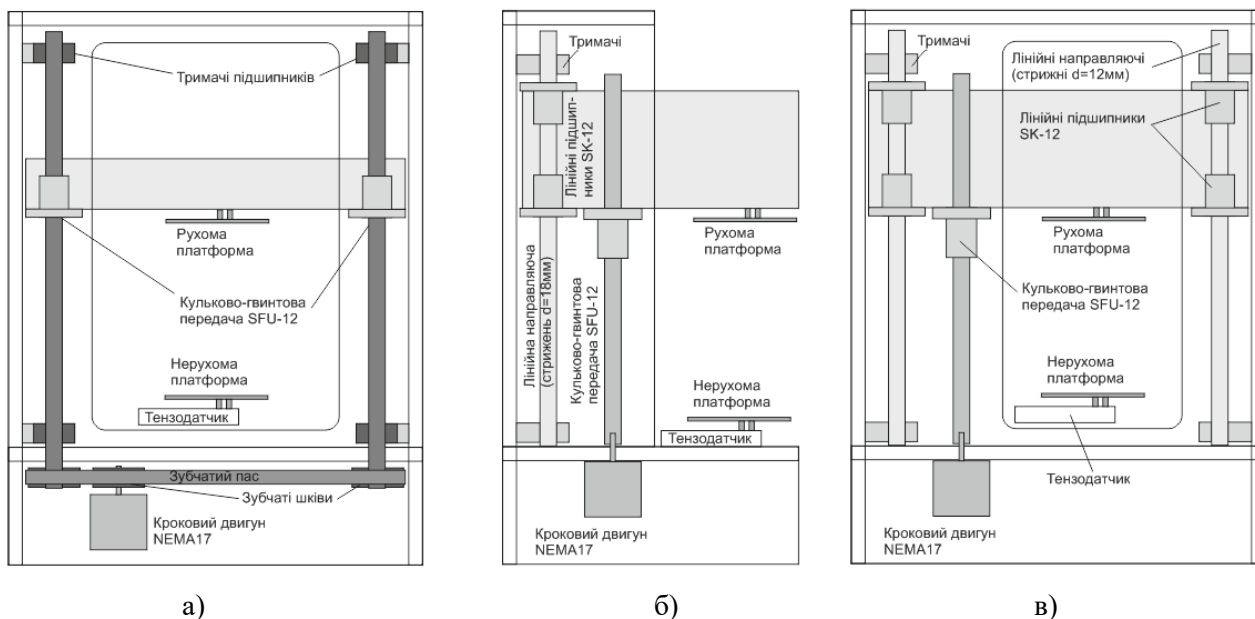


Рис. 1. Схеми навантажень випробувальних установок (а – двоколонна, б – консольна, в – змішана)

Вимірювання навантажень проводилось за допомогою тензодатчика з максимальним навантаженням 200 Н. Для спрощення принципової схеми використано стандартний модуль із спеціалізованою мікросхемою HX711, призначеною для роботи з тензодатчиками, яка містить в собі необхідний набір диференційних операційних підсилювачів, контролер для обробки даних та інтерфейс шини I2C [1].

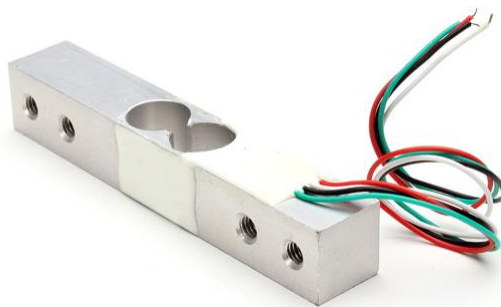


Рис. 2. Вимірювальний елемент з тензодатчиком [4]

Вимірювання переміщень проводилось непрямим способом за допомогою підрахунку імпульсів, що надходять на драйвер крокового двигуна.

Крок різьбового валу SFU12[3] – 4 мм, кількість кроків двигуна на оберт – 200, режим ділення кроку – 1/16. Отже переміщення на 1 імпульс:

$$\frac{4000 \text{ мкм}}{200 \text{ кроків} \cdot 16} \approx 1.455 \text{ мкм}$$

Це максимально теоретична точність позиціонування рухомого захвату для нашого двигуна та різьбового гвинта. Реальна точність все буде меншою, через деформації самої конструкції, деформації тензодатчика, а також інших факторів.

Підбір крокового двигуна за потужністю здійснювався з міркувань забезпечення крутного моменту, необхідного для створення поздовжньої сили 200 Н на різьбовому валу. За формулою знаходимо кут нахилу різьбової лінії:

$$\varphi = \arctg \frac{Pz}{\pi d} = \arctg \frac{4 \cdot 3}{3.14 \cdot 12} \approx 17.7^\circ,$$

де $P = 4$ мм – крок різьби; $z = 3$ – кількість заходів різьби; $d = 12$ мм – діаметр різьбового валу. Необхідний крутний момент розраховується за формулою

$$M = 0.5dF_{max}tg\varphi = 0.5 \cdot 0.12 \cdot 200 \cdot tg17.7^\circ \approx 38 \text{ Н}\cdot\text{см}.$$

Згідно отриманого крутного моменту підбираємо кроковий двигун Nema23 23HS0601 з крутним моментом на валу 40 Н·см.

Принципова електронна схема приладу (рис. 3) складається з кількох модулів. Основою є плата Arduino Nano. Для обробки сигналів з тензодатчика необхідний модуль HX711. Для індикації процесу та результатів експерименту використано рідкокристалічний символний дисплей 1602 з адаптером шини I2C.

Для керування кроковим двигуном вибрано драйвер A4988 з максимальним струмом на фазу 2А за умови встановлення радіатора на мікросхему.

Слід врахувати, що в драйверах A4988 заборонено відключати кабель двигуна при наявності живлення. Мікросхема драйвера згорає практично в 95% випадках.

Дисплей і модуль HX711 підключені на шину I2C (виводи A4 і A5 Arduino). Для керування випробувальною установкою використано 3 кнопки (вгору, вниз, стоп). Крім того, в установці є два вимикачі, які спрацьовують в крайніх положеннях рухомої частини з захватом. Для регулювання швидкості переміщення рухомого захвату використано змінний резистор, підключений до аналогового входу A0. Керування машиною також можливе і через USB за допомогою розробленої програми.

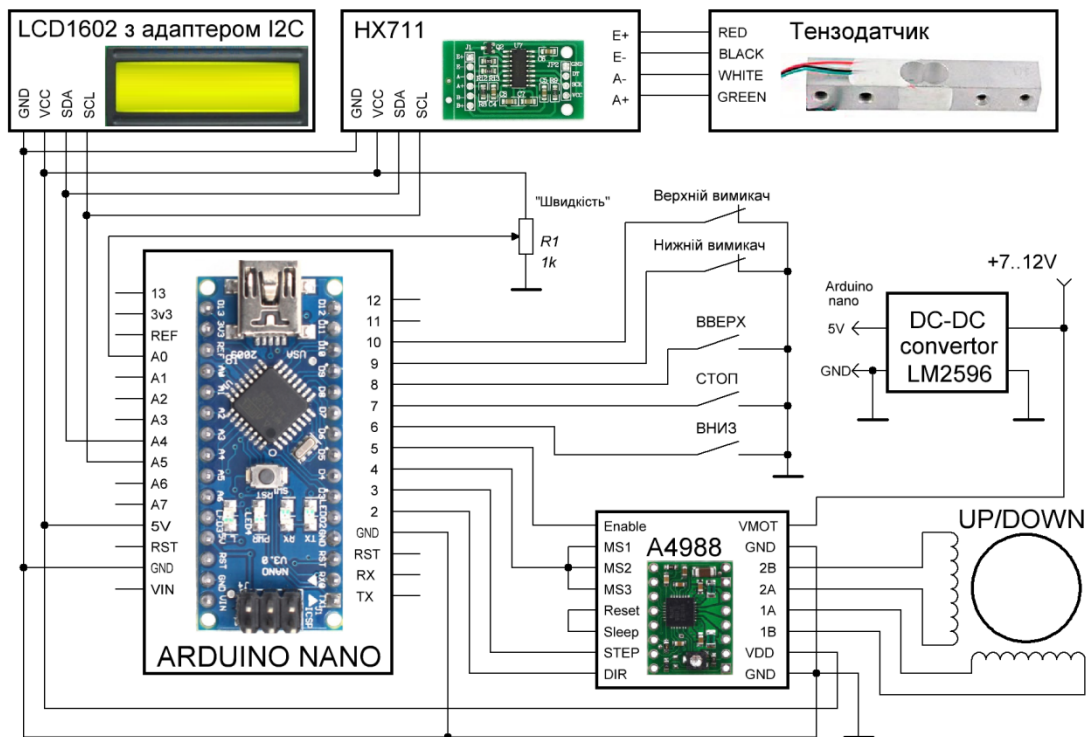


Рис. 3. Принципова схема

Програмне забезпечення для Arduino Nano написано в середовищі Arduino IDE та містить модулі коду для рідкокристалічного дисплея 1602, модуля HX711, драйвера крокового двигуна та обробки команд з кнопок та кінцевих вимикачів.

Керування драйвером двигуна відбувається за допомогою переривання Timer1 по співпаданню з регістром

```
ISR(TIMER1_COMPA_vect)
{
    if (MotorStarted)
    {
        digitalWrite(3,!digitalRead(3));
        if (FF) len=len+0.00125;
        if (BF) len=len-0.00125;
    }
}
```


}

Коефіцієнт 0.00125 – це переміщення рухомої частини в міліметрах на один імпульс, який надходить на драйвер двигуна.

Такий підхід дозволяє розвантажити основну функцію loop() та забезпечити стабільну швидкість зміни навантаження. Крім того, змінюючи регістр OCR1A, можна змінювати швидкість переміщення. Для цього на аналоговий вхід A7 підключено потенціометр, напруга на якому і визначає регістр OCR1A.

```
Speed=analogRead(A7)/10+1;
Vspeed=10.0/Speed;
cli(); // вимкнути глобальні переривання
OCR1A = Speed;
TCNT1=0;
sei(); // ввімкнути глобальні переривання
```

Змінна Vspeed містить значення швидкості в мм/с і призначена для виводу на дисплей та на ПК.

Варто відмітити, що найвищий пріоритет при роботі випробувальної установки мають кінцеві вимикачі, потім кнопки на передній панелі, і лише потім команди з ПК. Так, при спрацьовуванні верхнього кінцевого вимикача контролер зупинить рух вгору і не буде відкликатись на натискання кнопки «Вгору» та відповідну команду з ПК.

```
if ((digitalRead(8)==0)||((Buf=='u')) if (digitalRead(10)) //переключаємо мотор на рух вгору
якщо натиснена кнопка та рама не в крайньому верхньому положенні
```

Єдиний виняток – команда СТОП, яка зупинить машину як з програми, так і натисненням кнопки на передній панелі.

Установка має режим автокалібрування тензодатчика. Для цього необхідно виставити мінімально можливу швидкість (регулятор швидкості до упору проти годинникової стрілки) та затиснути кнопку «СТОП» на кілька секунд. При цьому значення з тензодатчика запам'ятується і буде вважатись як нульове.

```
//функція калібрування. По довгому натисненню кнопки "Стоп" при мінімальній швидкості
if ((digitalRead(7)==0)&&(Speed>100))
{
    offset_count++;
    delay(10);
}
else offset_count=0;
if (offset_count>10) Calibration(); //виклик функції калібрування
// кінець виклику калібрування
```

При натисканні кнопки "Стоп" установка повинна утримувати зразок при заданому навантаженні. Проте, якщо це не потрібно, то затиснувши кнопку на 1 секунду, можна вимкнути драйвер двигуна. На дисплеї в нижньому правому куті замість символів "M1" з'являться символи "M0".

```
//функція вимкнення драйверу (вимикаємо подачу напруги на мотор)
if ((digitalRead(7)==0) && (MotorStoped=1))
{
    motor_stop_count++;
    delay(10);
}
else motor_stop_count=0;
if (motor_stop_count>5) digitalWrite(5,HIGH); //Motor disable
```

На рис. 4 наведено вигляд дисплею установки для різних режимів роботи.



Рис. 4. Дисплей випробувальної установки при різних режимах роботи:
 (зліва направо: калібрування тензодатчика, результат калібрування, стиск зразка)

Програмне забезпечення для ПК написано в середовищі Lazarus і містить інтуїтивний інтерфейс для керування випробувальною машиною та має інструменти побудови графічних залежностей зміни характеристик навантажування, переміщення, деформування тощо. Доступні також функції автоматичної обробки графіків, задання програми циклічних навантажень, наявна база даних характеристик найбільш поширених матеріалів, збереження даних в Excel та багато інших. Скріншот програми представлено на рисунку 5.

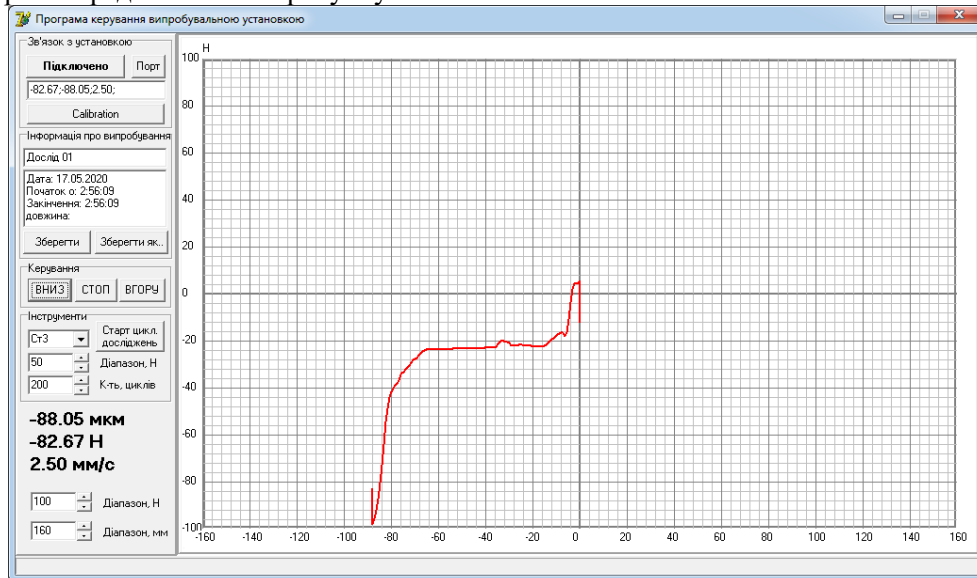


Рис. 5. Діаграма стиску тестового зразка (кубик пінополістиролу)

Висновки та практичне застосування. Розроблено компактну та недорогу випробувальну установку. Апробація випробувальної установки показала, що вона дозволяє проводити статичні випробування матеріалів та може використовуватись з навчально-демонстраційною метою під час викладання уроків фізики в школі, курсів механіки, опору матеріалів, теорії пружності, тощо у ЗВО. Собівартість (без врахування витрат часу на проектування та виготовлення дослідного екземпляру) склала 1200-1300 грн., що значно менше промислових аналогів з подібною функціональністю (до кількох десятків тисяч грн.).

Список бібліографічних посилань.

1. Документація на модуль HX711 [Електронний ресурс]. – Режим доступу: <https://arduino.ua/docs/hx711.pdf>
2. Tabletop Material Testing Instruments [Електронний ресурс]. – Режим доступу: https://www.aandd.jp/products/test_measuring/stb/stb.html
3. SFU1204 Ball ScrewKit. Dataspecs. URL: <https://www.handsontec.com/dataspecs/linearmotion/SFU1204-BallScrew-Kit.pdf> (дата звернення: 15.05.2020)
4. Сайт arduino.ua, сторінка тензодатчика 20кг [Електронний ресурс]. – Режим доступу: <https://arduino.ua/prod1977-datchik-vesa-tenzodatchik-20-kg>
5. DSTU 2824-94 Розрахунки та випробування на міцність. Види і методи механічних випробувань. Терміни та визначення
6. DSTU EN 10002-1:2006 Металеві матеріали. Випробування на розтяг. Частина 1. Метод випробування за кімнатної температури (EN 10002-1:2001, IDT)
7. DSTU EN 636:2014 Фанера. Технічні умови (EN 636:2003, IDT) [Чинний від 21.11.2014]. Український державний науково-дослідний інститут "Ресурс" (УкрНДІ "Ресурс"), 2014. 16 с.
8. Czichos, Horst Springer Handbook of Materials Measurement Methods. Berlin: Springer. 2006, pp. 1206 p.. - ISBN 9783540207856.

References

1. HX711 Datasheet [URL]: <https://arduino.ua/docs/hx711.pdf> (date of application: 15.05.2020)
 2. Tabletop Material Testing Instruments [URL]: https://www.aandd.jp/products/test_measuring/stb/stb.html (date of application: 15.05.2020)
 3. SFU1204 Ball ScrewKit. Dataspecs. [URL]: <https://www.handsontec.com/dataspecs/linearmotion/SFU1204-BallScrew-Kit.pdf> (date of application: 15.05.2020) (date of application: 15.05.2020)
 4. Arduino.ua Site, loadcells 200N page [URL]: <https://arduino.ua/prod1977-datchik-vesa-tenzodatchik-20-kg>
 5. DSTU 2824-94 Calculations and strength tests. Types and methods of mechanical tests. Terms and definitions
 6. DSTU EN 10002-1: 2006 Metallic materials. Tensile tests. Part 1. Test method at room temperature (EN 10002-1: 2001, IDT)
- DSTU EN 636: 2014 Plywood. Technical conditions (EN 636: 2003, IDT) [Effective from 21.11.2014]. Ukrainian State Research Institute "Resource" (UkrNDI "Resource"), 2014. 16 p. Czichos, Horst Springer Handbook of Materials Measurement Methods. Berlin: Springer. 2006, pp. 1206 p.. - ISBN 9783540207856

DOI: 10.36910/6775-2524-0560-2020-39-17

УДК: 004.771

Савицький Тимофій Павлович, магістрант,

Орлова Марія Миколаївна, к.т.н., доцент.

<https://orcid.org/0000-0002-6617-4631>

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

ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ СИСТЕМИ РОЗУМНОГО БУДИНКУ

Савицький Т.П., Орлова М. М. Використання штучного інтелекту для системи розумного будинку. Технології розумного дому та штучного інтелекту розвиваються швидко та в різних галузях науки. В статті розглянуто і проаналізовано способи та особливості побудови систем розумного будинку з використанням штучного інтелекту, детально розглянуто основні функції, які на сьогодні використовуються в системі розумного будинку та штучного інтелекту. Аналіз літератури та огляд продуктів даються для визначення функцій та ролей штучного інтелекту у системах розумного будинку. В роботі розглянуто моделі взаємодії користувачів, систем штучного інтелекту і модулів розумного будинку та аналізуються особливості їх використання. Визначається статус використання штучного інтелекту при розробці систем розумного дому і основні способи та напрямки його використання.

Ключові слова: штучний інтелект; функції штучного інтелекту; інтелектуальні взаємодії; розумний дім; ресурси розумного дому.

Савицький Т.П., Орлова М. Н. Использование искусственного интеллекта для системы умного дома. Технологии умного дома и искусственного интеллекта развиваются быстро и в разных областях науки. В статье рассмотрены и проанализированы способы и особенности построения систем умного дома с использованием искусственного интеллекта, подробно рассмотрены основные функции, которые сегодня используются в системе умного дома и искусственного интеллекта. Анализ литературы и обзор продуктов даются для определения функций и ролей искусственного интеллекта в системах умного дома. В работе рассмотрены модели взаимодействия пользователей, систем искусственного интеллекта и модулей умного дома, а также анализируются особенности их использования. Определяется статус использования искусственного интеллекта при разработке систем умного дома, основные способы и пути его использования.

Ключевые слова: искусственный интеллект; функции искусственного интеллекта; интеллектуальные взаимодействия; умный дом; ресурсы умного дома.

Savytskyi T.P., Orlova M. M. Using Artificial Intelligence for a Smart Home System. Smart home and artificial intelligence technologies are evolving rapidly and in various fields of science. The article discusses and analyses the methods and features of building smart home systems using artificial intelligence. Literature analysis and product review are provided to determine the functions and roles of AI in smart home systems. Models of the interaction of users, artificial intelligence systems and smart home modules are considered in the work and the peculiarities of their use are analyzed. The status of the use of artificial intelligence in the development of smart home systems and the main ways and directions of its use are determined.

Keywords: artificial intelligence; functions of artificial intelligence; intellectual interactions; smart home; resources of a smart home.

Вступ. В останні роки розвиток технології розумного дому сприяв переходу будинку від традиційного до інтелектуального за допомогою підключеного Інтернету. Розумний дім - це помешкання, оснащене технологіями, які включають датчики контролю навколишнього середовища, вхідними та вихідними даними і під'єднані до єдиного середовища [1]. Оснащені високотехнологічними автоматичними системами розумні будинки для зручності можуть відстежувати та контролювати домашні дії, забезпечувати мешканцям кращий комфорт та, можливо, заощаджувати споживання енергії. Технологія розумного дому збирає та аналізує дані з модулів, які входять до домашнього середовища. Він також передає інформацію користувачам і розширює потенціал управління різними системами [2].

Постановка наукової проблеми. Взаємодія штучного інтелекту з ресурсами розумного будинку стає все більш важливою у світі. В майбутньому розумним будинкам будуть приділяти все більше уваги взаємодії між людьми та навколишнім середовищем, щоб зробити будівлі більш стійкими та персоналізованими. Одним із важливих напрямків застосування штучного інтелекту для розумних будинків є створення єдиного середовища застосування. Технологія штучного інтелекту використовується в управлінні пристроїв, управлінні енергією, в охороні здоров'я, в інтелектуальній взаємодії, безпеці, розважальних системах та персональних роботах, використовуючи розпізнавання діяльності, обробку даних, прийняття рішень, розпізнавання зображень, прогнозування та розпізнавання голосу. Але технологія штучного інтелекту, розумні будинки та користувачі мають різні моделі взаємодії. В основному, існує дві типи моделей взаємодії. Перший - користувачі безпосередньо дають команди кожному розумному домашньому пристрою, а штучний інтелект,

вбудований у кожен пристрій, приносить користь саме конкретному пристрою. Другий - де користувачі дають інструкції штучному інтелекту, а штучний інтелект контролює кожен пристрій. У зв'язку з цим, буде розглянуто дві моделі застосування штучного інтелекту з розумним будинком.

Мета роботи полягає в аналізі існуючих рішень підвищення ефективності функціонування розумного будинку за рахунок використання штучного інтелекту, класифікації областей їх використання та розробці моделі ефективного використання енергії в будинку та спрощення системи управління ресурсами розумного дому.

Аналіз досліджень. Після аналізу літератури [4], [5], сформулюємо п'ять основних функцій розумного будинку, а саме: управління пристроями, управління енергією, охорона здоров'я, інтелектуальна взаємодія та безпека. Розділимо функції ШІ у розумних будинках на шість узагальнених

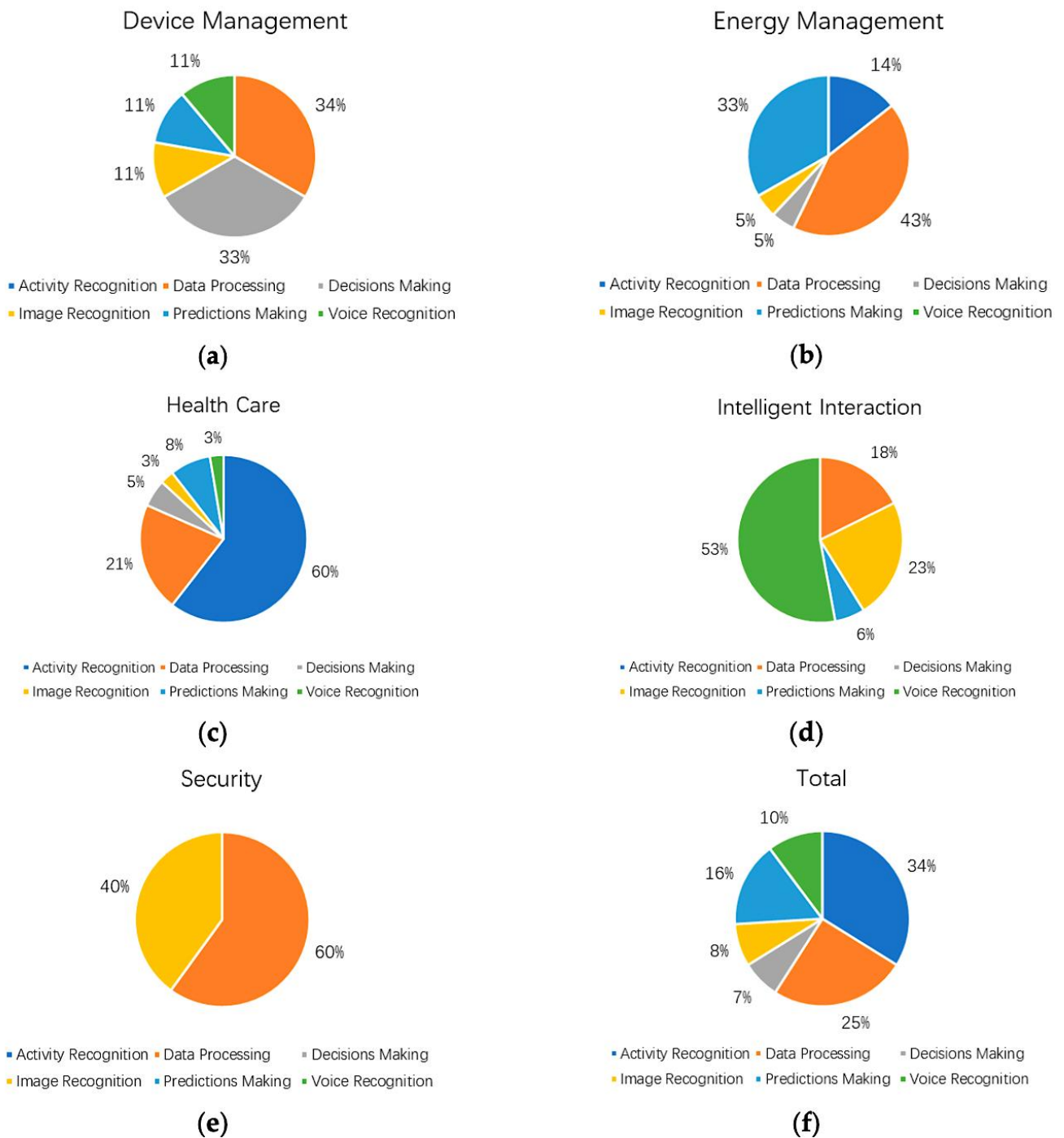


Рис. 2. Кількість публікацій у різних галузях. а) Функції штучного інтелекту (ФШІ) в управлінні пристроями; б) ФШІ у сфері енергетики; в) ФШІ у сфері охорони здоров'я; г) ФШІ для інтелектуальної взаємодії; е) ФШІ у сфері безпеки; ф) ФШІ для всіх функцій розумного будинку [6] кластерів діяльності людини: розпізнавання діяльності, обробка даних, прийняття рішень, розпізнавання зображень, прогнозування та розпізнавання голосу. У статті робота з даними включає обробку самих даних, семантичний аналіз та технології на основі правил управління розумного дому [6].

Як показано на рисунку 2а, управління розумним пристроєм домашнього пристрою підтримується п'ятьма функціями ШІ. На жаль, застосування ШІ в управлінні розумними пристроями не знайшли широкого висвітлення в описі літератури, оскільки є достатньо новими. Управління енергією розумного будинку підтримується п'ятьма функціями ШІ: розпізнаванням діяльності, обробкою даних, прийняттям рішень, розпізнаванням зображень та прогнозуванням. ШІ для розпізнавання активності більш широко досліджується ніж попередні галузі. Інтелектуальна взаємодія розумного будинку підтримується чотирма функціями ШІ: обробкою даних, розпізнаванням зображень, прогнозуванням та розпізнаванням голосу. З 2012 року в цій галузі було багато досліджень, але більшість досліджень є дуже новими. Безпека розумного будинку підтримується двома функціями ШІ: обробкою даних та розпізнаванням зображень. Решта досліджень, в основному, стосуються базових досліджень із застосування технології ШІ в розумному будинку. На рисунку 2f показана обробка даних та розпізнавання активності, які широко використовуються у всіх програмах розумного дому.

Виклад основного матеріалу та обґрунтування отриманих результатів. Технологія штучного інтелекту (ШІ) використовується в продуктах розумного дому. Для структуризації найбільш поширених функцій, які на сьогодні використовуються, визначимо шість основних кластерів функцій ШІ у розумних будинках, а саме: розпізнавання діяльності користувачів, обробка даних, розпізнавання голосу, розпізнавання зображень, прийняття рішень та формулювання прогнозу подальших дій. В аспекті розпізнавання активності пристроїв розумного будинку можуть розпізнавати діяльність людини за допомогою ШІ [3], який аналізує дані сенсорів, щоб виявити дії людей і формує сигнал тривоги, якщо є аномальна активність. В аспекті обробки даних ШІ заснований на зчитуванні інформації з різних модулів, методах аналізу даних та виявленні внутрішніх взаємозв'язків між ними. В системі розумного будинку на основі ШІ використовується модуль розпізнавання голосу для управління різними ресурсами. Наприклад, розпізнавання голосу використовується в Apple HomePod, Amazon Alexa, Athom Homey, Google Home та ін [3]. В аспекті розпізнавання зображень ШІ використовується для розпізнавання обличчя, емоцій, біометрики та аналізу навколишнього середовища. Він може вимірювати та аналізувати поведінку людини, а також фізичні структури та форми тіла. ШІ може вирішити, які дії слід вжити у відповідь на вхідні дані [4]. Наприклад, у інтелектуальній системі безпеки, якщо камера виявить незнайомця, який проникає в будинок, вона спрацює гучним сповіщенням та подає попередження на смартфон користувача або він може зателефонувати 102. Ці системи повинні бути досить швидкими та ефективними. Дані датчиків збираються комп'ютерною мережею і зберігаються у базі даних для оброблення інтелектуальним агентом, що генерує корисні знання, такі як закономірності, прогнози та тенденції. На основі цієї інформації розумний дім із ШІ може вибрати та автоматизувати дії для досягнення цілей програми розумного дому [5].

Аналіз інтересу до розумного будинку та штучного інтелекту показує, що вся взаємодія між пристроями та застосуванням ШІ виконується на двох рівнях. На сьогодні доступно багато рішень та розумних домашніх ресурсів, які мають низький рівень взаємодії між різними пристроями, серед них пристрої, які визначають параметри навколишнього середовища та управління будинком. Були спроби створення застосування високого рівня взаємодії продуктів розумного дому.

На рисунку 1 представлено тенденцію розвитку технології продуктів розумного дому та ШІ.

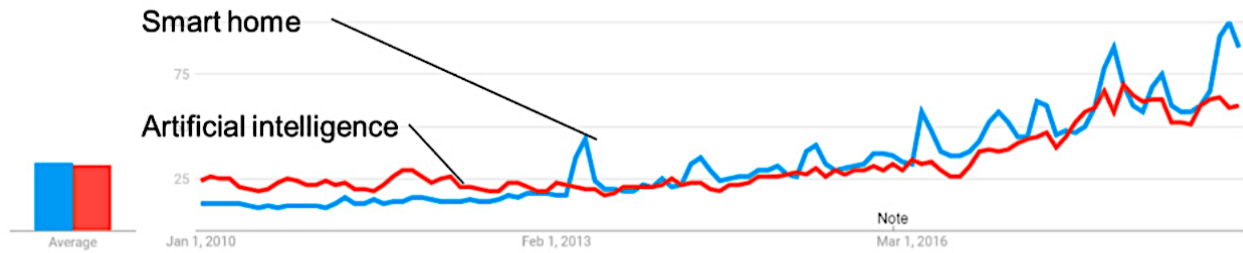


Рис. 1. Інтерес до розумного будинку та штучного інтелекту

Розподіл розумних будинків в області застосування ШІ показано на рисунку 3. У сукупності ці результати показують, що з часом обговорювалося все більше і більше областей застосування, і ці результати показують, що все більше обговорюються прикладні області використання ШІ. З 2015 року активно почався розвиток ШІ у сфері охорони здоров'я. Дослідження управління енергетикою зменшувалися з кожним роком, тоді як дослідження інтелектуальної взаємодії з кожним роком збільшувалися. Розглянемо результати п'яти областей застосування ШІ до розумного дому, а саме: управління пристроями, управління енергією, охорона здоров'я, інтелектуальна взаємодія та безпека.

По-перше, що стосується управління розумними пристроями дому, з прогресом технологій кількість електричних приладів у будинку зростає, а етапи роботи стають все складнішими. На жаль, ШІ не може допомогти користувачам автоматично керувати пристроями. Деякі дослідники впроваджували ШІ в системах розумного будинку для контролю та управління ресурсами в будинку, автоматично

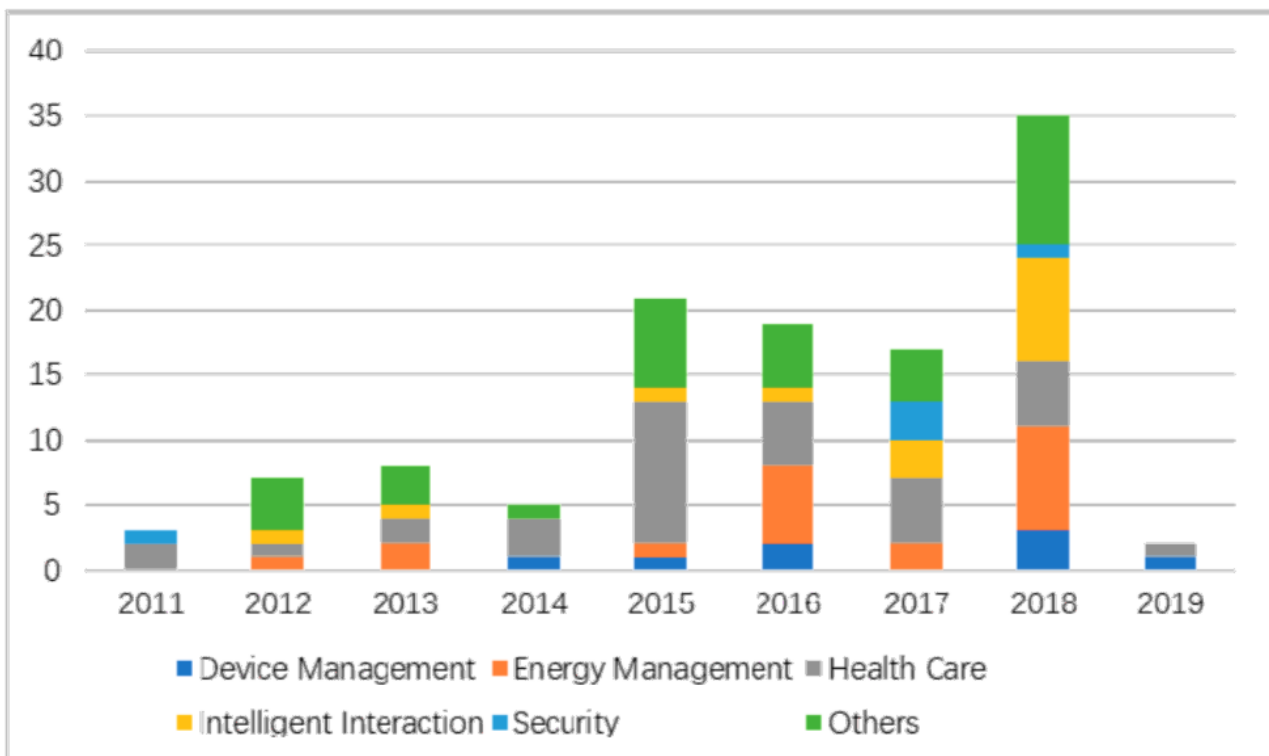


Рис. 3. Поля застосування розумних будинків з ШІ

контролюючи світло і температурні умови [7]. Інтелектуальне управління в розумному будинку також може бути реалізовано шляхом аналізу даних сенсорної мережі, вивчення попередньої поведінки користувача [7] або моделей користувачів, застосовуючи алгоритм логістичної класифікації, заснований на TensorFlow [8]. Централізоване управління може приймати електронні рішення, такі як моніторинг, підвищення комфорту, зручності, контроль навколишніх умов та надання необхідної інформації [8].

По-друге, управління енергією розумного будинку стає все більш важливим та нагальним для суспільства. Люди з різних галузей наполегливо працюють над зменшенням споживання енергії та підвищенням енергоефективності. Координація споживання енергії розумних приладів у розумних будинках може досягти більшої ефективності споживання [9]. Шаблони споживання енергії та їх взаємозв'язок із чинниками навколишнього середовища можуть бути проаналізовані ШІ для прогнозування щоденної потреби в електроенергії [9]. ШІ може допомогти середовищу розумного будинку у визначенні поведінки споживача енергії, щоб підтримати автоматизацію будинку та зменшити споживання енергії [9]. Розпізнавання діяльності ШІ також може допомогти зв'язати діяльність та існуючу побутову техніку, а потім дати рекомендації користувачам кожного разу, коли він виявить енергетичні покращення [9].

По-третє, що стосується інтелектуального догляду за домом, то з поступовим збільшенням тривалості життя охорона здоров'я стає все більш важливою. Використовуючи методи машинного навчання та штучного інтелекту за сенсорними даними, можна відстежувати та виявляти зміни у поведінці та способі життя людей [10]. Використовуючи алгоритм кластеризації без нагляду, модель періодичної нейронної мережі та генетичний алгоритм, системи ШІ можуть постійно контролювати людей похилого віку в розумних будинках та надсилати попередження спостерігачу, якщо виникають нетипові дії [10].

По-четверте, відносно інтелектуальної взаємодії модулів розумного дому, враховуючи, що кількість розумних домашніх пристроїв постійно збільшується, інтелектуальна взаємодія може змусити користувачів відчувати себе більш комфортно. Більше не потрібно підходити до кожного пристрою, щоб керувати ним вручну. Більшість дослідників використовували штучні нейронні мережі для класифікації входів користувачів для створення природного діалогу, надаючи користувачам можливість керувати приладами голосовими чи текстовими командами [11]. Розпізнавання голосу на основі ШІ використовує технологію взаємодії на основі аудіо, яка дозволяє користувачам мати повний контроль над домашнім середовищем [11]. Розпізнавання зображень також допомагає ШІ зрозуміти жести людей. Взаємодія людини та комп'ютера на основі жестів є природною та інтуїтивно зрозумілою. Люди з порушеннями мови можуть спілкуватися з розумними домашніми пристроями за допомогою динамічних жестів [11].

І останнє, з точки зору безпеки розумного будинку, для захисту власності та особистої безпеки необхідно забезпечити будинок від несподіваних подій та аварій. Штучний інтелект розпізнавання зображень може розпізнати зловмисника та попередити власника будинку [12]. Небезпека буває різною: пожежі, CO₂ тощо. ШІ може бути використаний для аналізу даних з датчиків та створення звуків тривоги [12].

Технології ШІ, розумні будинки та користувачі мають різні моделі взаємодії. В основному, існує дві типи моделей взаємодії. Перший показаний на рисунку 4а, де користувачі безпосередньо дають команди кожному розумному домашньому пристрою, а ШІ, вбудований у кожен пристрій, приносить користь саме конкретному пристрою. Розумне управління будинком, охорона здоров'я та безпека надають перевагу цій схемі. Друга модель зображена на рисунку 4б, де користувачі дають інструкції ШІ, а ШІ контролює кожен пристрій. Розумне управління домашнім пристроєм та інтелектуальна взаємодія працюють за цією схемою.

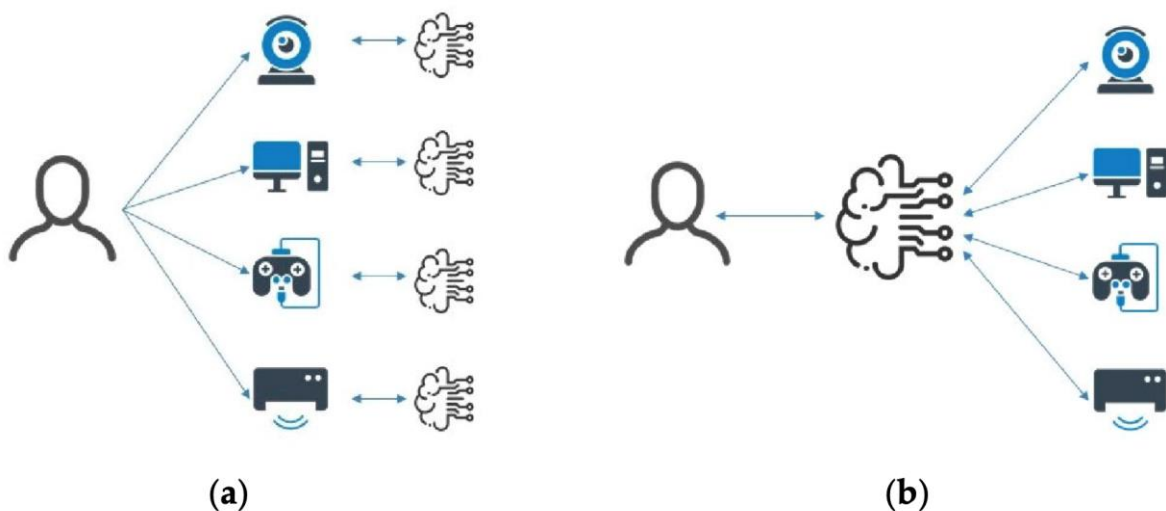


Рис. 4. а) Перша модель користувачів, ШІ та розумних будинків; б) другий зразок користувачів, ШІ та розумних будинків.

Отже, якщо використовувати другу модель взаємодії між користувачами, розумним будинком та ШІ, то підвищується ефективність взаємодії за рахунок створення єдиної екосистеми. Користувачі формують і передають команди в екосистему, а система штучного інтелекту аналізує ці команди, стан пристроїв розумного будинку і взаємодіє зі всіма ресурсами розумного будинку одночасно.

Висновки. В роботі розглянуто взаємозв'язок способів штучного інтелекту та їх застосуванням при створенні систем розумного будинку. Проаналізовано різні області використання штучного інтелекту, функціональні особливості систем зі ШІ, що розробляються деякими компаніями. Доведено, що технологія штучного інтелекту допомагає розумним будинкам в управлінні пристроями, управлінні енергією, охороною здоров'я, інтелектуальній взаємодії, безпеці, розважальних системах та персональних роботах, використовуючи розпізнавання діяльності, обробку даних, прийняття рішень, розпізнавання зображень, прогнозування та розпізнавання голосу. Показано, що на сьогодні системах розумного будинку зазвичай використовуються доволі прості методи штучного інтелекту, такі як розпізнавання зображень і голосу.

Проведене в даній роботі дослідження доводить необхідність використання способів штучного інтелекту при створенні розумного будинку для управління всіма його ресурсами, що дозволить створити єдину екосистему розумного будинку та розробити єдиний архітектурний дизайн управління.

Reference

1. Trivodaliev, K.V.; Risteska Stojkoska, B.L. A Review of internet of things for smart home: Challenges and solutions. *J. Clean. Prod.* 2016, 127, 454–464.
2. Norvig, P.; Russell, S. *Artificial Intelligence: A Modern Approach*, 2nd ed.; Prentice Hall Press: Upper Saddle River, NJ, USA, 2007.
3. Min, G.; Rho, S.; Chen, W. Advanced issues in artificial intelligence and pattern recognition for intelligent surveillance system in smart home environment. *Eng. Appl. Artif. Intell.* 2011, 23, 1287–1290.
4. Fritz, R.; Dermody, G. A Conceptual framework for clinicians working with artificial intelligence and health-assistive smart homes. *Nurs. Inq.* 2019, 25, 2–9.
5. Kumar, S.; Abdul Qadeer, M. Application of AI in home automation. *Int. J. Eng. Technol.* 2013, 4, 913–917
6. Mouftah, H.T.; Qela, B.; Observe, learn, and adapt (OLA)—An algorithm for energy management in smart homes using wireless sensors and artificial intelligence. *IEEE Trans. Smart Grid* 2014, 3, 1788–1802
7. Wardana, I.N.K.; Crisnapati, P.N.; Aryanto, I.K.A.A. Rudas: Energy and sensor devices management system in home automation. In *Proceedings of the 2016 IEEE Region 10 Symposium (TENSYP)*, Bali, Indonesia, 9–11 May 2016; pp. 184–187.
8. Park, M.H.; Ju, Y.W.; Jang, Y.H.; Park, S.C. Design of tensorflow-based proactive smart home managers. *Lect. Notes Electr. Eng.* 2019, 503, 94–105.
9. Jithish, J.; Sankaran, S. A Hybrid adaptive rule based system for smart home energy prediction. *CEUR Workshop Proc.* 2018, 1720.
10. Liouane, Z.; Roose, P.; Lemlouma, T.; Weis, F.; Messaoud, H. A Genetic Neural Network Approach for Unusual Behavior Prediction in Smart Home; Abraham, A., Franke, K., Köppen, M., Eds.; Springer: Berlin/Heidelberg, Germany 2018; Volume 2, pp. 942–969.
11. Li, Y.; Su, H.; Liu, L. *Gesture Recognition Based on Accelerometer and Gyroscope and Its Application in Medical and Smart Homes*; Springer International Publishing: New York, NY, USA, 2019; Volume 9983, pp. 61–71.
12. Wicaksono, W.R.; Surantha, N. Design of smart home security system using object recognition and PIR sensor. *Procedia Comput. Sci.* 2019, 89, 320–322.

DOI: 10.36910/6775-2524-0560-2020-39-18

УДК: 539.1.074+539.2

Федотов Вячеслав Віталійович, ст.викладач

<https://orcid.org/0000-0001-7821-0989>

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», м. Київ, Україна

МАТЕМАТИЧНА МОДЕЛЬ ЗБУДЖЕННЯ СПІНОВОГО ФОТОСТРУМУ У НАНОСТРУКТУРАХ АНТИФЕРОМАГНІТНИХ ДІЕЛЕКТРИКІВ

Федотов В. В. Математична модель збудження спінового фотоструму у наноструктурах антиферомагнітних діелектриків. Розглянуто фізичні принципи, що лежать у основі фотогальванічного ефекту, який виникає внаслідок проходження циркулярно поляризованого світла у середовищі металу і виражається у генеруванні прямого фотоструму, напрям якого залежить від поляризації світла. Визначено особливості виникнення даного ефекту у структурах антиферомагнітних діелектриків, зазначено його нелінійність та запропоновано подальше використання у опто-спінтронних запам'ятовуючих пристроях і нанодетекторах електромагнітного поля. У рамках побудови математичного апарату розгляд спінового імпульсу було перенесено з фотонів на магнони. Таким чином, було вказано, що циркулярно поляризоване електромагнітне поле в рамках моделі генерує прямий спіновий фотострум магнона, причому напрям струму визначається спіральністю світла. Показано, що явище резонансу на базі загальних принципів нанофотоники може бути представлено як взаємодія другого порядку поляризованого світла з речовиною. Розроблена математична модель була вдосконалена через врахування внеску геометричної організації складних ґратчастих структур антиферомагнітних діелектриків і явище антисиметричного обміну.

Ключові слова: наноструктури антиферомагнітних діелектриків, круговий фотогальванічний ефект, магنونний спіновий фотострум, циркулярно поляризоване світло, взаємодія другого порядку світла з речовиною, антисиметричний обмін, складні ґратчасті структури.

Федотов В. В. Математическая модель возбуждения спинового фототока в наноструктурах антиферромагнитных диэлектриков. Рассмотрены физические принципы, лежащие в основе фотогальванического эффекта, который возникает в результате прохождения циркулярно поляризованного света в среде металла и выражается в генерировании прямого фототока, направление которого зависит от поляризации света. Определены особенности возникновения данного эффекта в структурах антиферромагнитных диэлектриков, указано его нелинейность и предложено дальнейшее использование в опто-спинтронных запоминающих устройствах и нанодетекторах электромагнитного поля. В рамках построения математического аппарата рассмотрение спинового импульса было перенесено из фотонов на магнеты. Таким образом, было указано, что циркулярно поляризованное электромагнитное поле в рамках модели генерирует прямой спиновый фототок магнетона, причем направление тока определяется спиральностью света. Показано, что явление резонанса на базе общих принципов нанофотоники может быть представлено как взаимодействие второго порядка поляризованного света с веществом. Разработанная математическая модель была усовершенствована через учет геометрической организации сложных решетчатых структур антиферромагнитных диэлектриков и явление антисимметричного обмена.

Ключевые слова: наноструктуры антиферромагнитных диэлектриков, круговой фотогальванический эффект, магنونный спиновый фототок, циркулярно поляризованный свет, взаимодействие второго порядка света с веществом, антисимметричный обмен, сложные решетчатые структуры.

Fedotov V. V. Mathematical model of spin photocurrent excitation in of antiferromagnetic isolators nanostructures

Fundamental principles of circular photogalvanic effect as a result of circularly polarized light probing of metal environment and generation of a direct photocurrent which direction depends on the polarization of light are considered. The peculiarities of the occurrence of the mentioned effect in the structures of antiferromagnetic dielectrics are determined, nonlinearity of the effect is indicated, and further use in opto-spintronics' data storage and nanodetectors of the electromagnetic field is proposed. Due to necessity of the mathematical apparatus development, the consideration of the spin pulse was transferred from photons to magnons. Thus, it was stated that the circularly polarized electromagnetic field within the model generates a direct magnon spin photocurrent and the direction of the current is determined by the helicity of light. It is shown that the phenomenon of resonance based on general principles of nanophotonics can be represented as the interaction of the second order polarized light-matter interaction. The mathematical model was further adopted in order to take into account the contribution of the geometric organization of antiferromagnetic isolator complex lattice structures and the phenomenon of antisymmetric exchange.

Keywords: antiferromagnetic isolators nanostructures, circular photogalvanic effect, magnon spin photocurrents, circularly polarized light, second order light-matter interaction, antisymmetric exchange, complex lattice structures.

Постановка наукової проблеми. На сьогоднішній день структури на основі антиферомагнітних діелектриків розглядаються як один з найбільш перспективних матеріалів спінтроніки [1, 2]. Математичні моделі, що визначають процеси генерації та циркуляції спінових струмів електронів описуються через такі квазічастинки як магнони [10, 11]. Типові властивості антиферомагнітних магнетонів вказують на широку сферу застосувань наноструктур антиферомагнітних діелектриків, зокрема пропонується використовувати даного підходу при побудові спінтронних запам'ятовуючих пристроїв, що характеризуються високою інформаційною ємністю та високоефективних детекторів електромагнітного поля, що визначає рівень **актуальності даного дослідження.**

Аналіз сучасних досліджень та публікацій у зазначеній галузі вказує на перспективність застосування механізму ультрашвидкого оптичного збудження магніонів у структурах на основі антиферомагнітних діелектриків [1, 2], що дозволяє виділити окрему галузь антиферомагнітної опто-спінтроники. З метою побудови математичної моделі відповідного процесу було досліджено такі явища нанофотоники як хіральність і суперхіральність фотонів [3-5, 7, 8], принципи взаємодії світла з речовиною [6, 9] та особливості збудження магніонного спінового фотоструму у антиферомагнетиках [10, 11, 20, 21]. Також для уточнення математичного апарату, що описує механізми збудження спінових струмів у феромагнетиках та антиферомагнетиках було використано роботи по моделюванню у галузі квантової теорії магнетизму [12-19], зокрема математичні моделі, що враховують внесок геометричної організації складних ґратчастих структур феромагнетиків та антиферомагнетиків [21-23] та явище антисиметричного обміну [24].

Зазначені матеріали наукових публікацій, тим не менш, не надають цілісної методології математичного моделювання та розробки опто-спінтронних пристроїв на базі антиферомагнітних наноструктур, що може бути виділено як **невирішену частину загального дослідження**. Таким чином, за **мету даного дослідження** може бути поставлено задачу побудови методологічної бази для аналізу процесів генерації та циркуляції спінових магніонних струмів у відповідності до структурних особливостей середовища антиферомагнітних діелектриків.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження.

1. Збудження фотоструму у середовищі провідника та діелектрику

Моделювання механізму генерації спінового струму у середовищі металу або антиферомагнітного діелектрику базується на врахування пучка поляризованого світла яким опромінюється зразок та фотогальванічному ефекті(рис. 1, 2).

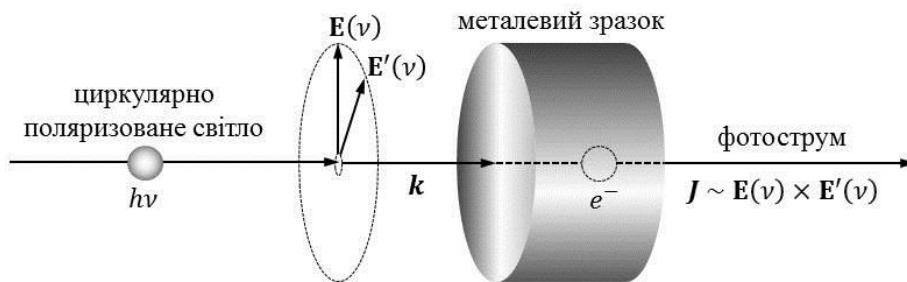


Рис. 1. Базова модель генерації фотоструму циркулярно поляризованим світлом у металевому зразку

У середовищі металу круговий фотогальванічний ефект (circular photogalvanic effect, CPGE) слід розглядати як нелінійний процес, що відображає взаємодію зразка і електромагнітного поля циркулярно поляризованої хвилі, напруженість якого описується функцією від частоти $\mathbf{E}(\nu)$. Якщо хвильовий вектор \mathbf{k} напрямлено перпендикулярно поверхні зразка, то вектори напруженості електромагнітного поля циркулярно поляризованої хвилі $\mathbf{E}(\nu)$ та $\mathbf{E}'(\nu)$ будуть, у свою чергу паралельні поверхні зразка, а їх векторний добуток є пропорційним по відношенню до значення густини фотоструму \mathbf{J} , що виникає у зразку, а також визначає напрям відповідного вектору (рис. 1).

Модель генерації магніонного спінового фотоструму у зразку антиферомагнітного діелектрику може бути побудована аналогічним чином. У даному випадку математичний апарат має включати векторний добуток векторів магнітної індукції $\mathbf{B} \times \mathbf{B}'$ (рис. 2).

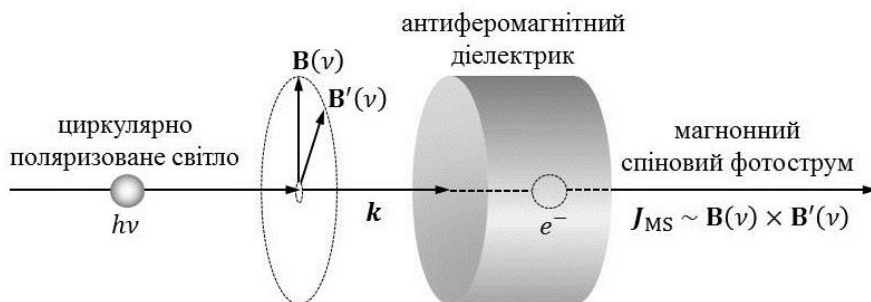


Рис. 2. Модель генерації магніонного спінового фотоструму циркулярно поляризованим світлом у зразку антиферомагнітного діелектрику

На основі векторного добутку можна визначити електромагнітну хіральність зразка [13-15], як показник, що характеризує асиметрію у взаємодії світла з речовиною [16-18]. Характерно, що зміна величини магнітної складової оптичної хіральності може бути визначена через залежність $\chi^M(k) \sim ik \cdot B(k) \times B'(k)$. Нерівномірний розподіл сукупності магнітонів можна розглядати як передачу спінового кутового моменту від світла до антиферомагнітних структур, що призводить до генерації спінового фотоструму, причому для такої системи можна визначити умови резонансу взаємодії.

2. Моделювання процесу генерації магнітного спінового фотоструму

Моделювання оптичної генерації спінових струмів може бути проведено на базі класичної електродинамічної моделі. Так, наприклад, для антиферомагнетика з кубічної кристалічної структурою можна виділити дві підрешітки M_1 і M_2 , на основі чого можна визначити густину енергії через введення коефіцієнтів обміну між підрешітками c_1 і c_2 та коефіцієнт легко осьової анізотропії c_3 вздовж вісі z :

$$W = c_1 \cdot M_1 \cdot M_2 + c_2 \cdot \nabla M_1 \cdot \nabla M_2 - c_3 \cdot ((M_1^z)^2 + (M_2^z)^2). \quad (1)$$

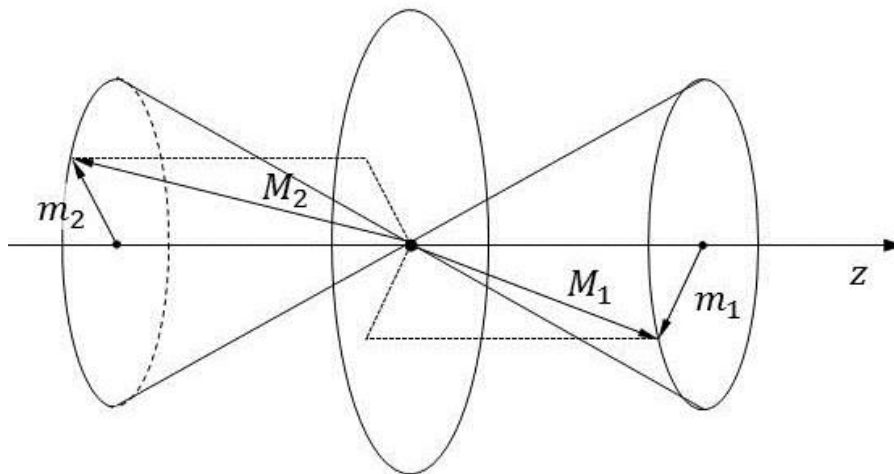


Рис. 3. Схема прецесійного руху векторів магнетизації підрешіток антиферомагнетика з кубічної кристалічною структурою

На базі лінеаризованого рівняння Ландау-Лівшица можна записати наступну систему рівнянь для флуктуаційних компонент підрешітки:

$$\begin{cases} \partial_t p_k = -\gamma \cdot M_s \cdot (2c_3 - c_2 \cdot k^2) \cdot z_u \times q_k \\ \partial_t q_k = -\gamma \cdot M_s \cdot (2c_1 + c_2 \cdot k^2 + 2c_3) \cdot z_u \times p_k \end{cases}, \quad (3)$$

де

$$\begin{cases} p_k = m_{1k} + m_{2k} \\ q_k = m_{1k} - m_{2k} \end{cases}, \quad (4)$$

На основі зазначеної системи рівнянь можна вивести наступні складові математичного апарату моделі генерації магнітного спінового фотоструму у середовищі антиферомагнітного діелектрику:

- магнетизація насичення M_s ;
- компоненти Фур'є-розкладу функції збудження магнетизації m_{1k} відносно впорядкованого стану, що може бути описано ;
- одиничний вектор вздовж вісі z ;
- гіромагнітне відношення γ .

При цьому впорядкований стан системи описується через наступну рівність:

$$M_i = m_i + (-1)^{i+1} \cdot M_s \cdot z_u. \quad (5)$$

Враховуючи те, що обертальна симетрія зберігається при взаємодії між електромагнітним полем і системою спінів, вираз для спінового струму як функції від аргументів p_k і q_k можна знайти з рівняння безперервності для z -складової щільності намагніченості:

$$m^z = \frac{m_2^2 - m_1^2}{2 \cdot M_s}. \quad (6)$$

На основі проведеного аналізу розраховується величина щільності магнетонного спінового фотоструму:

$$J_s = -i \cdot \frac{\sum_k \left(\frac{\partial(2c_3 - c_2 k^2)}{\partial k} (p_k^* \times p_k)_z + \frac{\partial(2c_1 + c_2 k^2 + 2c_3)}{\partial k} (q_k^* \times q_k)_z \right)}{4 \cdot M_s} \quad (7)$$

Таким чином, динаміка зміни намагніченості, збуджена електромагнітною хвилею світла, котрим опромінюється зразок, пов'язана з напрямом вісі z і, відповідно, значення p_k і q_k пропорційні $B(k)$.

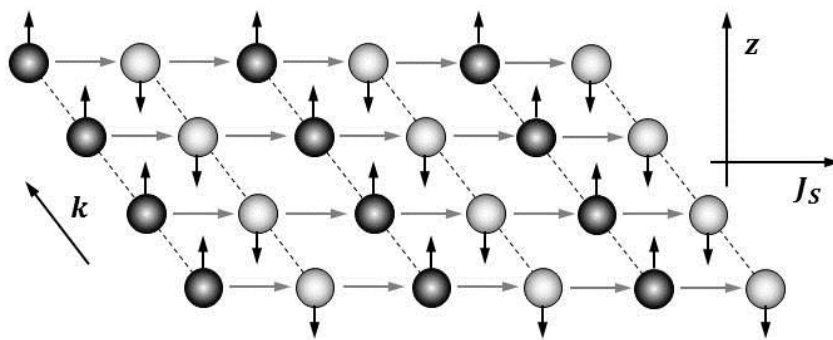


Рис. 4. Схема збудження спінового фотоструму у середовищі антиферомагнетика з кубічною кристалічною структурою

Запропонована математична модель відображає збудження спінового фотоструму у середовищі антиферомагнетика з кубічною кристалічною структурою, що представлена на рис. 4, де чорні стрілки відображають спіни, а сірі — антисиметричний обмін. Але математичний апарат також може бути адаптовано для відображення процесу збудження спінового фотоструму у середовищі антиферомагнетика з гексагональною кристалічною структурою, що є актуальним для багатьох наноструктурних матеріалів на базі антиферомагнітних діелектриків.

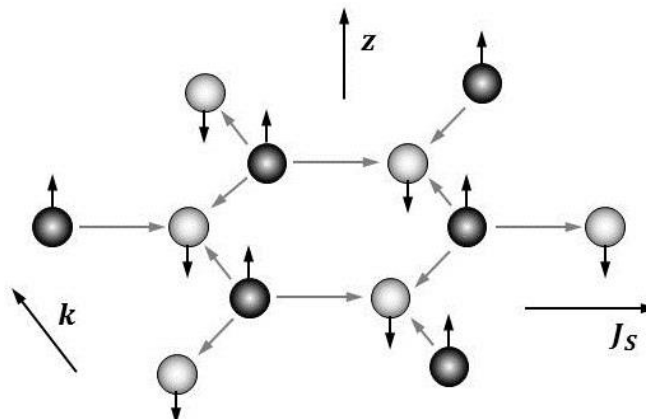


Рис. 5. Схема збудження спінового фотоструму у середовищі антиферомагнетика з гексагональною кристалічною структурою

Відповідно, розроблена методологічна база може бути надалі використана для аналізу процесів генерації та циркуляції спінових магنونних струмів у відповідності до структурних особливостей середовища антиферомагнетних діелектриків.

Висновки. В результаті аналізу базових фізичних принципів, що лежать у основі кругового фотогальванічного ефекту визначено особливості виникнення магنونного спінового фотоструму у структурах антиферомагнетних діелектриків, зазначено його нелінійність та запропоновано області його використання у пристроях опто-спінтроники. При побудові математичної моделі розгляд спінового імпульсу було перенесено з фотонів на магнони. Показано, що циркулярно поляризоване електромагнетне поле в рамках моделі генерує прямий спіновий фотострум магнона, причому напрям струму визначається спіральністю світла. Розроблена математична модель була вдосконалена через врахування внеску геометричної організації складних ґратчастих структур антиферомагнетних діелектриків і явище антисиметричного обміну.

References

1. Satoh, T., Iida, R., Higuchi, T., Fujii, Y., Koreeda, A., Ueda, H., ... & Ivanov, B. A. (2017). Excitation of coupled spin-orbit dynamics in cobalt oxide by femtosecond laser pulses. *Nature communications*, 8(1), 1-6.
2. Němec, P., Fiebig, M., Kampfrath, T., & Kimel, A. V. (2018). Antiferromagnetic opto-spintronics. *Nature Physics*, 14(3), 229-241.
3. Valev, V. K., Govorov, A. O., & Pendry, J. (2017). Chirality and Nanophotonics: Chirality and Nanophotonics (Advanced Optical Materials 16/2017). *Advanced Optical Materials*, 5(16).
4. Tang, Y., & Cohen, A. E. (2011). Enhanced enantioselectivity in excitation of chiral molecules by superchiral light. *Science*, 332(6027), 333-336.
5. Lovesey, S. W., Collins, J. T., & Collins, S. P. (2019). Superchiral photons unveil magnetic circular dichroism. *Physical Review B*, 99(5), 054428.
6. Bliokh, K. Y., Kivshar, Y. S., & Nori, F. (2014). Magnetoelectric effects in local light-matter interactions. *Physical review letters*, 113(3), 033601.
7. Canaguier-Durand, A., Hutchison, J. A., Genet, C., & Ebbesen, T. W. (2013). Mechanical separation of chiral dipoles by chiral light. *New Journal of Physics*, 15(12), 123037.
8. Starosta, K., Koike, T., Chiara, C. J., Fossan, D. B., & Vaman, C. (2002, April). Chirality and angular momentum coupling in odd-odd nuclei. In *AIP Conference Proceedings* (Vol. 610, No. 1, pp. 815-819). American Institute of Physics.
9. Lüth, H., & Ibach, H. (2003). *Solid-state physics: an introduction to principles of materials science*. Springer-Verlag Berlin Heidelberg.
10. Proskurin, I., Ovchinnikov, A. S., Kishine, J. I., & Stamps, R. L. (2018). Excitation of magnon spin photocurrents in antiferromagnetic insulators. *Physical Review B*, 98(13), 134422.
11. Sharma, S., Blanter, Y. M., & Bauer, G. E. (2017). Light scattering by magnons in whispering gallery mode cavities. *Physical Review B*, 96(9), 094412.
12. Johansen, Ø., & Brataas, A. (2018). Nonlocal coupling between antiferromagnets and ferromagnets in cavities. *Physical review letters*, 121(8), 087204.
13. Nolting, W., & Ramakanth, A. (2009). *Quantum theory of magnetism*. Springer Science & Business Media.
14. Lin, W., Chen, K., Zhang, S., & Chien, C. L. (2016). Enhancement of thermally injected spin current through an antiferromagnetic insulator. *Physical review letters*, 116(18), 186601.
15. Takei, S., Moriyama, T., Ono, T., & Tserkovnyak, Y. (2015). Antiferromagnet-mediated spin transfer between a metal and a ferromagnet. *Physical Review B*, 92(2), 020409.
16. Khymyn, R., Lisenkov, I., Tiberkevich, V. S., Slavin, A. N., & Ivanov, B. A. (2016). Transformation of spin current by antiferromagnetic insulators. *Physical Review B*, 93(22), 224421.
17. Hoffmann, M. C., Khanna, V., & Cavalleri, A. (2010). Noncollinear Broadband Terahertz-pump—Terahertz-probe spectroscopy of semiconductors. International Conference on Ultrafast Phenomena. doi: 10.1364/up.2010.me42.
18. Hirori, H., Doi, A., Blanchard, F., & Tanaka, K. (2011). Single-cycle terahertz pulses with amplitudes exceeding 1 MV/cm generated by optical rectification in LiNbO₃. *Applied Physics Letters*, 98(9), 091106.
19. Kawano, M., & Hotta, C. (2019). Thermal Hall effect and topological edge states in a square-lattice antiferromagnet. *Physical Review B*, 99(5), 054422.
20. Okuma, N. (2018). Quantum Theory of Antiferromagnetic Opto-spintronics: Reciprocal and Nonreciprocal Magnons Coupled with Polarized Photons. *arXiv preprint arXiv:1805.08226*.
21. Owerre, S. A., Mellado, P., & Baskaran, G. (2019). Photoinduced Floquet topological magnons in Kitaev magnets. *EPL (Europhysics Letters)*, 126(2), 27002.
22. Matusiak, M., Babij, M., Pomjakushina, E., & Conder, K. (2016). Wiedemann–Franz law in iron-based superconductor Fe_{1+x}Te_{1-x}Se_x. *physica status solidi (b)*, 253(8), 1607-1611.
23. Kimura, T., Hamrle, J., Yang, T., & Otani, Y. (2005, April). Magnetization switching due to non-local spin injection into small ferromagnetic particle. In *2005 IEEE International Magnetism Conference (INTERMAG)* (pp. 133-134). IEEE.
24. Okubo, S., Ueda, T., Ohta, H., Zhang, W., Sakurai, T., Onishi, N., ... & Sakai, T. (2012). Dzyaloshinsky-Moriya interaction and field-induced magnetic order in an antiferromagnetic honeycomb lattice compound Bi₃Mn₄O₁₂ (NO₃) studied by high-field electron spin resonance. *Physical Review B*, 86(14), 140401.

DOI: 10.36910/6775-2524-0560-2020-39-19

УДК: 62-503.56:621.717

Чибирик Яна Ивановна, к.т.н., доцент

<https://orcid.org/0000-0002-0634-7609>

Сумской государственной университет, г. Сумы, Украина

Коноплянченко Евгений Владиславович, к.т.н., доцент

<http://orcid.org/0000-0003-4814-1796>

Сумской национальный аграрный университет, г. Сумы, Украина

Марченко Анна Викторовна, к.т.н., доцент

<https://orcid.org/0000-0003-2003-3531>

Сумской государственной университет, г. Сумы, Украина

ТЕХНОЛОГИЧЕСКИЕ ЗАКОНОМЕРНОСТИ И МАТЕМАТИЧЕСКИЕ МОДЕЛИ СИНТЕЗА РАЦИОНАЛЬНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ СБОРКИ ИЗДЕЛИЙ

Чибирик Я. И., Коноплянченко Е. В., Марченко А. В. Технологические закономерности и математические модели синтеза рациональной последовательности сборки изделий. В данной статье на основе анализа иерархического строения изделия и технологии выполнения сборочных работ установлено, что деление изделий на сборочные единицы приводит к сокращению длительности цикла изготовления за счет параллельного выполнения сборочных операций. Установлено, что в условиях конкретного производства на этапе конструкторской подготовки целесообразно выявлять скрытые резервы по сокращению длительности сборочных процессов. В работе выведены зависимости, позволяющие оценить количество возможных вариантов сборки для технологических схем различных структурных видов: последовательного, минимально-параллельного и максимально-параллельного. Предложенные математические модели в виде матрицы базирования, доступа, размерных связей содержат информацию о конструкторских особенностях изделия и в неявном виде задают последовательность выполнения сборочных операций. Использование разработанного алгоритма решения задачи, с учетом матричного описания исходных данных, позволяет ограничить перебор возможных вариантов сборки изделия, повысить качество технологического проектирования за счет снижения вычислительных затрат и сокращения времени поиска последовательности сборки, минимальной по длительности производственного цикла.

Ключевые слова: длительность цикла сборки, матрица базирования, матрица доступа, структура изделия, последовательность сборки.

Чибирик Я. І., Коноплянченко Є. В., Марченко А. В. Технологічні закономірності і математичні моделі синтезу раціональної послідовності складання виробів. У даній статті на основі аналізу ієрархічної будови виробу і технології виконання складальних робіт встановлено, що поділ виробів на складальні одиниці призводить до скорочення тривалості циклу виготовлення за рахунок паралельного виконання складальних операцій. Встановлено, що в умовах конкретного виробництва на етапі конструкторської підготовки доцільно виявляти приховані резерви по скороченню тривалості складальних процесів. У роботі виведені залежності, що дозволяють оцінити кількість можливих варіантів складання для технологічних схем різних структурних видів: послідовного, мінімально-паралельного і максимально-паралельного. Запропоновані математичні моделі у вигляді матриці базування, доступу, розмірних зв'язків містять інформацію про конструкторські особливості виробу і в неявному вигляді задають послідовність виконання складальних операцій. Використання розробленого алгоритму розв'язання задачі, з урахуванням матричного опису вхідних даних, дозволяє обмежити перебір можливих варіантів складання виробу, підвищити якість технологічного проектування за рахунок зниження обчислювальних витрат і скорочення часу пошуку послідовності складання, мінімальної по тривалості виробничого циклу.

Ключові слова: тривалість циклу складання, матриця базування, матриця доступу, структура виробу, послідовність складання.

Chybirak Ya. I., Konoplianchenko Ie. V., Marchenko A. V. Technological patterns and mathematical models of the synthesis of a rational sequence of product assembly. Based on the analysis of the product hierarchical structure and the assembling technology, it was defined that dividing the products into assembly units leads to a reduction in the length of manufacturing cycle through parallel execution of assembly operations. It was determined that in the conditions of a specific production it is advisable to identify hidden reserves aimed at reduction of assembly duration at the stage of constructional design. This article presents dependencies, which allow estimating the number of possible assembly options for technological schemes of various structural types: sequential, minimum-parallel, and maximum-parallel. Suggested mathematical models in the form of locating matrix, access matrix, and matrix of dimensional relationships contain information about the product design and implicitly specify the assembly sequence. Taking into account matrix description of the source data, developed solution algorithm allows limiting possible assembly options for the product, improve the quality of technological design through reduction of computational costs and seek time needed for the assembly sequence (with the minimum duration of the production cycle).

Keywords: assembly cycle duration, locating matrix, access matrix, product structure, assembly sequence.

Постановка научной проблемы. Переход к современным формам организации производств и необходимость выпуска конкурентоспособной продукции требует поиска новых закономерностей выполнения технологий, учитывающих как конструкторско-технологические свойства изделия, так и свойства технологического оснащения. Заключительным этапом производственного процесса в машиностроении является сборка, поэтому качество готовых изделий во многом зависит от ее технологии. Основой для развития технологических процессов сборки являются технологические схемы сборки (ТСС). ТСС являются связующим звеном между структурой производственной системы и

изделием, так как с одной стороны они отражают вид сборки изделия по последовательности, а с другой – структуру сборочной системы. ТСС для одного и того же изделия можно составлять в нескольких вариантах. Но во всех случаях при сборке необходимо стремиться к минимизации трудоемкости, себестоимости, длительности производственного цикла. В статье решается задача синтеза рациональной по длительности производственного цикла последовательности сборки изделий.

Зависимости, приведенные в работе [1, 2] позволяют установить, что с уменьшением длительности производства продукции возрастает производительность работы, уменьшается себестоимость готовых изделий, ускоряется оборот средств, вложенных в предприятие. Следовательно, проектирование такой последовательности сборки, которая приводит к сокращению производственного цикла по времени, является актуальной задачей. Построение рациональной последовательности сборки является конструкторско-технологической задачей. Эффективность ее решения зависит от использования компьютерных технологий и требует формализованного математического описания для разработки соответствующего алгоритмического и программного обеспечения.

Анализ исследований. Рассмотрим некоторые примеры совершенствования сборочных процессов. В статье [3] предложены алгебраические модели, которые можно использовать для выбора рациональных проектных решений на подготовительном этапе сборочного производства. Структура изделия описана в виде гиперграфа, который используется для генерации различных вариантов сборки изделия. Однако разработанный метод не учитывает точностные характеристики изготовления деталей, кроме того, многовариантность сборки для многокомпонентных деталей обуславливает сложность анализа сетевой модели графа. В работе [4] сборочный процесс описан в виде целевой задачи оптимизации и предложен алгоритм ее решения, разработанный с использованием эволюционного подхода на основе метода нечеткой оценки Парето. Данный метод построения порядка сборки не учитывает размеры и массу деталей, осуществляет перебор всех возможных вариантов, требуя для своей реализации больших объем компьютерной памяти и вычислительных мощностей. Общая идея метода, представленного в работе [5] состоит в использовании средств имитационного моделирования для подготовки опорных точек локальной траектории движения деталей в САПР и последующей передаче их в систему управления сборочного роботизированного комплекса. В статье [6] представлены возможности интеллектуального процесса сборки, с использованием специальных инструментов дополненной реальности, в сочетании с логическими процедурами автоматизированного проектирования и планирования. Сборочный процесс реализован в виртуальной среде, где инженеры и конструкторы могут отслеживать точное положение, ориентацию отдельного сборочного элемента, видеть процесс перемещения элементов сборки в соответствии с его траекторией. Это позволяет предотвращать возможные ошибки в ходе технологического процесса.

Исследования показывают, что в настоящее время накоплен определенный опыт в решении задач оптимизации сборочных процессов. При этом не достаточно внимания уделяется сокращению длительности производственного цикла и определению резервов по его сокращению, исходя из возможностей конструкторско-технологических характеристик изделия. Поэтому требуется дальнейшая работа по выявлению закономерностей между длительностью цикла сборки и структурой изделия.

Изложение основного материала и обоснование полученных результатов исследований.

Структуру изделия можно представить графически, выполнив деление на отдельные сборочные единицы (СЕ) и детали. После 1-го этапа деления получим множество СЕ и деталей 1-го порядка. Продолжая деление, получаем сборочные элементы 2-го, 3-го, ..., n-го порядка. На n-м уровне иерархии находятся неделимые элементы – детали. Таким образом, граф, представленный на рис. 1 отображает структуру изделия. Корень графа обозначает собранное изделие, вершины – отдельные сборочные элементы, количество которых возрастет, а состав упрощается с увеличением уровня иерархии.

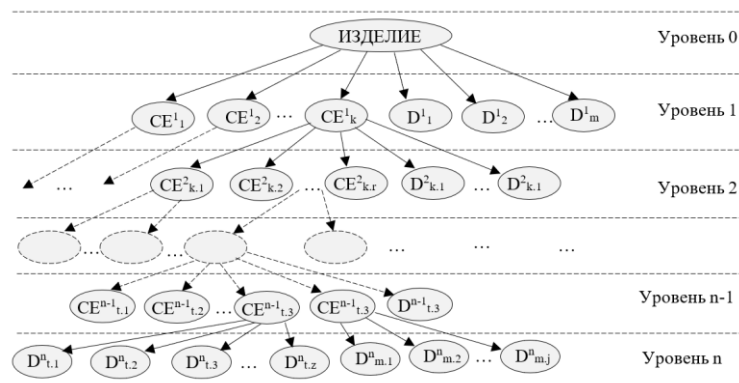


Рис. 1. Иерархическая структура изделия

Анализ полученной иерархической структуры изделия и технологии сборочных работ позволяет установить следующее:

- СЕ-цы, находящиеся на одном уровне иерархии не связаны между собой наличием общих поверхностей базирования, поэтому процесс их сборки может быть реализован в любой последовательности;
- деление изделий на отдельные, независимые СЕ-цы дает возможность среди множества сборочных операций определить те, которые могут выполняться одновременно;
- последовательность сборки СЕ-ц, которые находятся на различных уровнях иерархии и связаны наличием общих базовых поверхностей, определяется порядком вхождения этих СЕ-ц в конструкцию изделия;
- СЕ-ца является составной частью изделия и обладает всеми свойствами его конструкции, поэтому технологические закономерности построения рациональной сборки, выявленные для отдельной СЕ-цы, могут быть применены для изделия в целом.

Последовательность сборки изделия отображается его технологической схемой (ТС). Для одного изделия можно построить несколько вариантов ТС, при этом изменению подлежат такие важные показатели как длительность производственного цикла, масса и габаритные размеры СЕ-ц, точность выполнения сборочных операций, надежность изделия.

Существует три вида структур ТС [1]: последовательный, минимально-параллельный и максимально-параллельный.

Рисунок 1 отображает существующие структурные виды ТСС для изделия, состоящего из 8 деталей. Для представленных схем наблюдается различные значения длительности сборки:

- $T_{\text{пос.}} = 7$ усл. ед. времени, для последовательного структурного вида;
- $T_{\text{мин.-п.}} = 4$ усл. ед. времени, для минимально-параллельного вида;
- $T_{\text{макс.-п.}} = 3$ усл. ед. времени, для максимально-параллельного вида.

Таким образом, $T_{\text{пос.}} > T_{\text{мин.-п.}} > T_{\text{макс.-п.}}$.

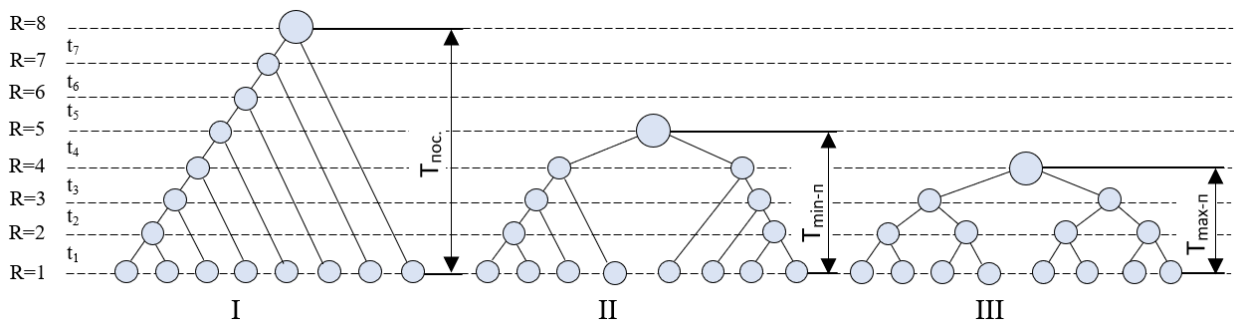


Рис. 2. Графы ТС сборки для изделия из 8 деталей: I – последовательный вид; II – минимально-параллельный вид; III – максимально-параллельный вид

Последовательность и длительность сборки, а также количество выполняемых сборочных операций зависит от структуры изделия. Учет этой зависимости позволяет выявлять скрытые резервы по сокращению длительности сборки. Поэтому, в условиях конкретного производства, на этапе конструкторской подготовки, целесообразно изменять конструкцию изделия с целью использования выявленного резерва.

Определим количество возможных вариантов сборки для различных структурных видов ТС: последовательного, минимально-параллельного и максимально-параллельного. С использованием численного моделирования получены зависимости, отражающие количество возможных вариантов сборки для изделия, состоящего из N деталей.

В таблице 1 показана зависимость возможных вариантов сборки изделия от количества деталей N для различных уровней графа последовательной структуры (рис. 1).

Обобщая результаты, приведенные в таблице 1, получена общая зависимость, связывающая количество деталей в изделии и количество вариантов сборки для технологической схемы последовательного структурного вида:

$$V_{\text{т.п.}} = \frac{1}{2^{(N-1)}} N(N-1)^2(N-2)^2(N-3)^2 \cdot \dots \cdot 2^2 \cdot 1 = \frac{1}{2^{(N-1)}} N!(N-1)! \quad (1)$$

Граф мінімально-паралельного структурного виду (рис. 1) має більш складну структуру: на кожному наступному рівні R графа, кількість вершин зменшується на два порівняно з попереднім (R-1)-м рівнем.

Таблиця 1. Кількість варіантів збірки изделия для ТС послідовального структурного виду

Уровень графа	Кількість варіантів збірки
$R = N - 1$	$C_{N-(R-1)}^2 = \frac{1}{2}(N - (R - 1))(N - R) = \frac{1}{2}(2 \cdot 1)$
$R = N - 2$	$C_{N-(R-1)}^2 = \frac{1}{2}(N - (R - 1))(N - R) = \frac{1}{2}(3 \cdot 2)$
...	...
$R = 3$	$C_{N-2}^2 = \frac{1}{2}(N - 2)(N - 3)$
$R = 2$	$C_{N-1}^2 = \frac{1}{2}(N - 1)(N - 2)$
$R = 1$	$C_N^2 = \frac{1}{2}N(N - 1)$

Учитывая это, в таблицах 2,3 показаны зависимости возможных вариантов сборки для ТС минимально-параллельной структуры.

Таблиця 2. Кількість варіантів збірки изделия для ТС мінімально-паралельного структурного виду

Уровень графа	Кількість варіантів збірки	Уровень графа	Кількість варіантів збірки
$N - \text{парное, } N \geq 4$		$(N - \text{не парное, } N \geq 5)$	
$R = \frac{N}{2} - 1$	$\left(\frac{N}{2} - (R - 2)\right) \left(C_{\left(\frac{N}{2} - (R - 1)\right)}^2\right)^2$	$R = \frac{(N - 1)}{2}$	$C_{\left(\frac{N+1}{2} - (R - 1)\right)}^2 \cdot C_{\left(\frac{N-1}{2} - (R - 1)\right)}^2$
$R = \frac{N}{2} - 2$	$\left(\frac{N}{2} - (R - 2)\right) \left(C_{\left(\frac{N}{2} - (R - 1)\right)}^2\right)^2$	$R = \frac{(N - 1)}{2} - 1$	$C_{\left(\frac{N+1}{2} - (R - 1)\right)}^2 \cdot C_{\left(\frac{N-1}{2} - (R - 1)\right)}^2$
...
$R = 3$	$\left(\frac{N}{2} - 1\right) \left(C_{\left(\frac{N}{2} - 2\right)}^2\right)^2$	$R = 3$	$C_{\left(\frac{N+1}{2} - 2\right)}^2 \cdot C_{\left(\frac{N-1}{2} - 2\right)}^2$
$R = 2$	$\left(\frac{N}{2}\right) \left(C_{\left(\frac{N}{2} - 1\right)}^2\right)^2$	$R = 2$	$C_{\left(\frac{N+1}{2} - 1\right)}^2 \cdot C_{\left(\frac{N-1}{2} - 1\right)}^2$
$R = 1$	$\left(1 + \frac{N}{2}\right) \left(C_{\frac{N}{2}}^2\right)^2$	$R = 1$	$C_{\frac{N+1}{2}}^2 \cdot C_{\frac{N-1}{2}}^2$

Применяя к данным, приведенным в таблице 2 основные правила комбинаторики установлено, что для ТС минимально-параллельного структурного вида количество возможных вариантов сборки изделия выражается зависимостью:

$$V_{\text{т. мин-п.}} = \begin{cases} \frac{1}{4^{\left(\frac{N-1}{2}\right)}} \left(1 + \frac{N}{2}\right)! \left(\frac{N}{2}\right)^2 \left[\left(\frac{N}{2} - 1\right)!\right]^4, & \text{если } N \text{ парное } (N \geq 4) \\ \frac{(N+1)}{4^{\left(\frac{N-1}{2}\right)}} \left[\left(\frac{N-3}{2}\right)!\right] \left[\left(\frac{N-1}{2}\right)!\right]^3, & \text{если } N \text{ не парное } (N \geq 5) \end{cases} \quad (2)$$

Учитывая тот факт, что для графа ТС максимально-параллельной структуры (рис. 1) на каждом следующем уровне R по сравнению с предыдущим (R-1)-м уровнем, количество сборочных

компонентов уменьшается в 2 раза, были полученные зависимости количества возможных вариантов сборки для различных уровней графа, которые представлены в табл. 3.

Таблица 3. Количество вариантов сборки изделия для ТС максимально-параллельного структурного вида

Уровень графа	Количество вариантов сборки
$R = \lceil \log_2 N \rceil - 1$	$\left(\left\lceil \frac{N}{2^{R-1}} \right\rceil - 1 \right) \left(\left\lceil \frac{N}{2^{R-1}} \right\rceil - 3 \right) = 3$
...	...
$R = 4$	$\left(\left\lceil \frac{N}{2^3} \right\rceil - 1 \right) \left(\left\lceil \frac{N}{2^3} \right\rceil - 3 \right)$
$R = 3$	$\left(\left\lceil \frac{N}{2^2} \right\rceil - 1 \right) \left(\left\lceil \frac{N}{2^2} \right\rceil - 3 \right)$
$R = 2$	$\left(\left\lceil \frac{N}{2} \right\rceil - 1 \right) \left(\left\lceil \frac{N}{2} \right\rceil - 3 \right)$
$R = 1$	$(N-1)(N-3)$

Общая формула теоретически возможных вариантов последовательности сборки для технологической схемы максимально-параллельного структурного вида выражается произведением:

$$V_{\text{т. макс.-п.}} = \prod_{k=1}^{\lceil \log_2 N \rceil - 1} \left(\left\lceil \frac{N}{2^{k-1}} \right\rceil - 1 \right) \left(\left\lceil \frac{N}{2^{k-1}} \right\rceil - 3 \right) \quad (3)$$

Большую значимость представляет поиск практически реализуемых в производственных условиях вариантов сборки $V_{\text{пр.}}$, среди которых можно выбрать оптимальный вариант по длительности производственного цикла. Количеством теоретически возможных $V_{\text{т.}}$ вариантов сборки является частью множества или подмножеством практически реализуемых $V_{\text{пр.}}$ на производстве вариантов сборки: $V_{\text{пр.}} \subseteq V_{\text{т.}}$

Полученные соотношения (1), (2), (3) показывают, что при увеличении количества деталей N возрастает количество возможных вариантов сборки и, как следствие значительно увеличивается время проектирования технологического процесса, поэтому построение всех возможных вариантов сборки $V_{\text{т.}}$ не является целесообразным. С целью исключения из рассмотрения не нужных вариантов, в процессе разработки рациональной последовательности сборки изделий следует учитывать конструкторско-технологические ограничения. В качестве ограничивающих факторов используют конструкторско-технологические признаки СЭ:

- наличие общих базовых поверхностей между деталями и сборочными единицами;
- масса и габаритные размеры деталей;
- иерархичность строения изделий;
- точность изготовления отдельных сборочных элементов;
- ограничения доступа на перемещение одних деталей другими к месту установки.

Таким образом, количество вариантов сборки во многом определяется особенностями конструкции изделия. Для их учета разработаем математические модели.

Выполнение сборочными элементами (СЭ), входящими в состав изделия, определенных функций обеспечивается наличием между соответствующими поверхностями СЭ связей сопряжения и положения. Если детали в изделии сопрягаются по общим контурам, они имеют общие базовые поверхности. Эти связи базирования разделяются на основные и вспомогательные. Основные связи определяются наличием типов соединений между деталями, вспомогательные – реализуются в конструкции как следствие основных. Для учета связей базирования введем матрицу $B_{N \times N}$ (N – количество деталей в изделии), элементы которой имеют вид:

$$B_{i,j} = \begin{cases} 0, & \text{если между } i - \text{ми } j - \text{м СЭ} - \text{ми отсутствуют общие связи базирования;} \\ 1, & \text{если между } i - \text{ми } j - \text{м СЭ} - \text{ми существует сборочное соединение.} \end{cases}$$

Матрица $B_{N \times N}$ описывает взаимное расположение СЭ в изделии. Исходной информацией для ее построения является чертеж изделия.

Кроме непосредственного контакта между поверхностями, относительная ориентация деталей в конструкции также задается размерными связями: линейными, угловыми, радиальными, диаметральными. Размеры, входящие в сборочную размерную цепь (СРЦ), реализуются в соответствии с последовательностью сборки элементов, входящих в ее состав. В случае наличия общих звеньев для нескольких размерных цепей, последовательность их сборки определяется правилами ранжирования СРЦ [1]. Для описания размерных связей в изделии определим математическую модель в виде матрицы $Z_{n \times n}$ (n - количество деталей, входящих в СРЦ):

$$Z_{i,j} = \begin{cases} 1, \text{ если } i - \text{й и } j - \text{й СЭ} - \text{ты связаны замыкающим звеном размерной цепи (РЦ);} \\ 2, \text{ если точность замыкающего звена РЦ достигается методом полной взаимозаменяемости;} \\ 3, \text{ точность достигается методом неполной взаимозаменяемости;} \\ 4, \text{ точность достигается методом групповой взаимозаменяемости;} \\ 5, \text{ точность достигается методом пригонки;} \\ 6, \text{ точность достигается методом регулирования;} \\ 7, \text{ для достижения точности применяются компенсирующие материалы} \end{cases}$$

Матрица $Z_{n \times n}$ позволяет находить замыкающие звенья размерной цепи, а также учитывать метод достижения точности сборки.

Те звенья размерной цепи, к которым предъявляются высокие требования по точности сопряжения, должны иметь также высокую точность изготовления. Чтобы избежать возрастания суммарных погрешностей размеров в процессе сборки, в первую очередь должны собираться те детали, к обеспечению точности которых предъявляются более высокие требования. Точность деталей опишем вектором T_N , значение элемента которого $T_j = k$, где k – качество точности изготовления детали.

На порядок сборки СЭ влияет также их масса и размерные характеристики. В первую очередь в сборке должны участвовать детали, имеющие меньшие размеры и массу. Это объясняется тем, что СЭ больших габаритов и массы тяжело перемещать и сопрягать с требуемой точностью. Для учета этих факторов введем вектор M_N с элементами ($M_j = m$, где m – масса деталей) и вектор G_N с элементами ($G_j = g$, где g – наибольший габаритный размер детали).

На последовательность сборки также оказывает влияние ограничение доступа одних деталей другими к месту установки, которые представим матрицей доступа $D_{N \times N}$, элементы которой:

$$D_{i,j} = \begin{cases} 0, \text{ если } i - \text{е сборочное соединение реализуется после } j - \text{го;} \\ 1, \text{ в противном случае.} \end{cases}$$

Матрицы базирования и доступа определяют порядок вхождения деталей в изделие, относительно базовых СЭ, т. е. задают последовательность выполнения соединений.

Разработанные математические модели в виде одномерных и многомерных массивов в неявном виде содержат информацию о конструкторско-технологических характеристиках изделия, удобны для хранения данных в памяти компьютера и позволяют разработать общий алгоритм синтеза последовательности сборки, рациональной по длительности производственного цикла (рис. 3).

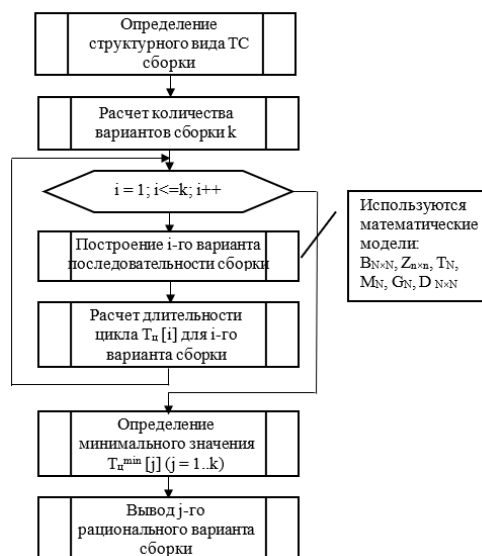


Рис. 3. Алгоритм построения рациональной последовательности сборки изделий

Выводы и перспективы дальнейших исследований. В данной статье, в результате выполненных исследований, выявлены критерии рациональной последовательности сборки. С использованием численного моделирования получены аналитические зависимости между структурным видом технологической схемы сборки и количеством возможных вариантов сборки, которые показали значительное увеличение вариативности с усложнением конструкции изделий. Это обусловило необходимость учета ограничений, позволяющих среди множества вариантов сборки выявлять те, которые могут быть реализованы на практике. Для их учета были разработаны математические модели, которые описывают основные характеристики изделия: габаритные размеры сборочных элементов, их массу, точность изготовления, ограничения доступа на перемещения одних деталей другими к месту установки, наличие общих баз деталей и сборочных единиц, виды сборочных соединений. Матричное представление математических моделей, позволяет сохранять исходную информацию в памяти компьютера и использовать ее в автоматическом режиме. Предложенный алгоритм построения рациональной последовательности сборки изделий избавляет от необходимости перебора всех допустимых решений и сокращает количество вычислительных итераций.

К ограничениям разработанного алгоритма следует отнести ручной ввод исходных данных, что может привести к субъективной оценке и к значительным временным затратам при обработке большого объема вводимой информации, которая значительно возрастает с усложнением конструкции собираемых изделий. Поэтому дальнейшие исследования будут направлены на возможность автоматической генерации исходных данных с использованием двумерных чертежей в цифровом виде, созданных с помощью программных инструментов CAD/CAE/CAM-систем.

Список библиографического описания

1. Захаров, Н. (1993) Построение рациональных автоматизированных сборочных технологических систем. 48 с.
2. Bojko, A., Ivahnenko, A., Chernyanskii, A., Solncev, A., & Tovkach, P. (2012) Rational decision-making in design of assembly diagrams. *Science and Education of the Bauman MSTU*, 12(10), P. 33-36. <https://doi.org/10.7463/1012.0475116>
3. Bojko, A. (2016) Algebraic Models of Product Assembly Process. *Science and Education of the Bauman MSTU*, 16(12), P.216-232. <https://doi.org/10.7463/1216.0852565>
4. Yong-fa, Q., & Zhi-gang, X. (2007) Assembly Process Planning Using a Multi-objective Optimization Method. 2007 International Conference on Mechatronics and Automation, P. 593-598. <https://doi.org/10.1109/ICMA.2007.4303610>
5. Revina, I. V., & Trifonova, E. N. (2020). Simulation modeling of the assembly process. *Journal of Physics: Conference Series*. <https://doi.org/10.1088/1742-6596/1441/1/012110>
6. Novak-Marcincin, J., Barna, J., Janak, M., Novakova-Marcincinova, L., & Torok, J. (2012). Visualization of intelligent assembling process by augmented reality tools application. 2012 4th IEEE International Symposium on Logistics and Industrial Informatics, P. 33-36. <https://doi.org/10.1109/LINDI.2012.6319505>

References

1. Zakharov, N. (1993) Building rational automated assembly process systems. 48 p.
2. Bojko, A., Ivahnenko, A., Chernyanskii, A., Solncev, A., & Tovkach, P. (2012) Rational decision-making in design of assembly diagrams. *Science and Education of the Bauman MSTU*, 12(10), P. 33-36. <https://doi.org/10.7463/1012.0475116>
3. Bojko, A. (2016) Algebraic Models of Product Assembly Process. *Science and Education of the Bauman MSTU*, 16(12), P.216-232. <https://doi.org/10.7463/1216.0852565>
4. Yong-fa, Q., & Zhi-gang, X. (2007) Assembly Process Planning Using a Multi-objective Optimization Method. 2007 International Conference on Mechatronics and Automation, P. 593-598. <https://doi.org/10.1109/ICMA.2007.4303610>
5. Revina, I. V., & Trifonova, E. N. (2020). Simulation modeling of the assembly process. *Journal of Physics: Conference Series*. <https://doi.org/10.1088/1742-6596/1441/1/012110>
6. Novak-Marcincin, J., Barna, J., Janak, M., Novakova-Marcincinova, L., & Torok, J. (2012). Visualization of intelligent assembling process by augmented reality tools application. 2012 4th IEEE International Symposium on Logistics and Industrial Informatics, P. 33-36. <https://doi.org/10.1109/LINDI.2012.6319505>

DOI: 10.36910/6775-2524-0560-2020-39-20

УДК: 004.7

Войтенко Єлизавета Дмитрівна, магістрант.

Орлова Марія Миколаївна, к.т.н., доцент.

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», м. Київ, Україна.

АНАЛІЗ ВРАЗЛИВОСТІ БЕЗПЕКИ ФУНКЦІОНУВАННЯ BGP ТА ПРИЧИН СКЛАДНОСТІ БОРОТЬБИ З НИМИ

Войтенко Є. Д., Орлова М. М. Аналіз вразливості безпеки функціонування BGP та причин складності боротьби з ними. У цій статті проведено дослідження та аналіз вразливостей роботи BGP зі сторони його алгоритмічної поведінки та зі сторони механіки налагоджування зв'язку між автономними системами за вимогами даного протоколу. Також наведений системний аналіз проблематики введення схвалених IETF розширень до цього протоколу, проблематики тестування нових рішень, а також причин повної або часткової відмови від остаточного їх загальнодоступного використання. На основі виконаного аналізу описані можливі варіанти подальших напрямків дій для боротьби з описаними вразливостями.

Ключові слова: Border Gateway Protocol, BGPSEC, безпека маршрутизації, безпека Інтернету, міждоменна маршрутизація.

Войтенко Е. Д., Орлова М. Н. Анализ уязвимости безопасности функционирования BGP и причин сложности борьбы с ними. В этой статье проведено исследование и анализ уязвимостей работы BGP со стороны его алгоритмической поведінки и со стороны механики налаживания связи между автономными системами согласно требованиям данного протокола. Также приведен системный анализ проблематики введения одобренных IETF расширений к этому протоколу, проблематика тестирования новых решений, а также причин полного или частичного отказа от окончательного их общедоступного использования. На основе выполненного анализа описаны возможные варианты дальнейших направлений действий для борьбы с описанными уязвимостями.

Ключевые слова: Border Gateway Protocol, BGPSEC, безопасность маршрутизации, безопасность Интернета, междоменная маршрутизация.

Voitenko I., Orlova M. Analysis of the security vulnerability of BGP functioning and the reasons for complexity of dealing with them. This article investigates and analyzes the vulnerabilities of BGP from its algorithmic behavior and from the mechanics of debugging autonomous systems according to the requirements of this protocol. It also provides a systematic analysis of the issue of the introduction of approved IETF extensions to this protocol, the issues of testing new solutions, as well as the reasons for the total or partial rejection of their final public use. Based on the analysis, possible options for further action to address the described vulnerabilities are described.

Keywords: Border Gateway Protocol, BGPSEC, routing security, Internet security, cross-domain routing.

Постановка наукової проблеми. У світі сучасних комунікацій чималу роль відіграє протокол граничного шлюзу BGP (Border Gateway Protocol). Саме цей протокол є засобом міждоменної маршрутизації, що об'єднує Інтернет, забезпечуючи маршрут передачі IP-пакетів для обміну між мережами, керованими різними провайдерами, по всьому світу.

І хоча перша версія протоколу побачила світ у далекому 1989 році, а значить і всі недоліки у внутрішніх механізмах його безпеки відомі теж вже досить немалий час, протокол BGP досі залишається вразливим для безлічі атак, які можуть викликати великомасштабні збої в роботі, стати причиною витоку маршрутів, а також можуть бути використані для інших шкідливих цілей в глобальній мережі, таких як розсилка спаму або відстеження трафіку. Наприклад, в 2014 році викрадення маршрутів BGP було вдало використано для крадіжки понад 83 тисяч доларів в криптовалютному еквіваленті [1, с. 387]. Таким чином, забезпечення безпеки BGP є ключем до підвищення загальної безпеки екосистеми Інтернету.

Знаючи все це здається дивним відсутність постійного активного вдосконалення політики безпеки маршрутизації в BGP, адже забезпечення надійності комутації в комп'ютерних мережах постійно актуальний та досить популярний напрямок для роботи серед спеціалістів та науковців. Насправді вся ця бездіяльність лише зовнішня. Інженерна рада Інтернету IETF (Internet Engineering Task Force) – відкрите міжнародне співтовариство проектувальників, учених, мережевих операторів і провайдерів, постійно знаходиться в пошуку рішень, але лише 17 з усіх, що коли-небудь були надіслані на розгляд, були затверджені та мали спробу виведення реалізації для широкого доступу [3]. Отже, такий малий об'єм впровадження має під собою досить серйозне підґрунтя у вигляді складності впровадження змін до BGP через обмеженість рівня втручання зі сторони багатьох аспектів.

Аналіз останніх досліджень і публікацій. Пропозиції з приводу забезпечення безпеки BGP надходять не тільки від IETF [1-4], але і від промислових діячів, а також від академічної спільноти [5]. Попередні опубліковані дослідження дослідницької спільноти з усього світу були сфокусовані саме на оцінці продуктивності та ефективності пропозицій щодо безпеки [5, 6], їх обмежень та переваг щодо їх гарантій безпеки [7], методів, які використовуються для захисту BGP [7], динаміки їх архітектури [8] і аналіз їх додаткових переваг безпеки при впровадженні та повноцінне функціонування [8].

Постановка завдання. Метою статті є дослідження вразливостей безпеки в BGP, їх класифікація, системний аналіз причин відсутності повноцінної працюючих рішень, причин відмов від пропозицій з вирішення даної проблеми та складностей використання навіть затверджених рішень.

Виклад основного матеріалу дослідження. З накопиченням практичного досвіду роботи з BGP-4 зросла обізнаність про його недоліки та слабкі місця. Основним аспектом безпеки, обговорюваним і найбільш виділеним, підкресленим в Заявці на обговорення RFC (Request for Comments) BGP, інформаційному документі, що містить технічні специфікації та стандарти Інтернету в цілому та BGP вчасності, є доступність. Існує розуміння того, що BGP, як єдиний інструмент маршрутизації між автономними системами, повинен забезпечувати якомога більшу доступність, в тому числі і в разі атаки або ж ненавмисного збою.

Отже, за таких вимог для підвищення доступності більшість розширень протоколу BGP повинні бути, та власне вже в основному і є, націлені на зниження складності управління та ручного налаштування оголошувачів маршрутних таблиць BGP, які найбільш схильні до ненавмисних збоїв, а також на зменшення нестабільності маршрутизації, з огляду на зростаючу складність топологій мереж та політик розсилки.

Іншими ж аспектами безпеки, тема яких піднімається в RFC, можна назвати цілісність та правильність інформації щодо маршрутизації. Вони обговорюються в документах спеціально присвячених саме вразливостям безпеки конкретно BGP або протоколів маршрутизації загалом. Докладний аналіз уразливості протоколу BGP був опублікований в 2006 році [2]. Хоча в документі, в основному, розглядаються вразливості, пов'язані з механізмами якраз сеансів BGP та обміну повідомленнями, також в ньому згадується, що вразливості, пов'язані саме з алгоритмічною поведінкою BGP. Конкретні причини в тому документі ще не були повністю сформовані. Однак зараз можна сказати, що існує три наступні групи «відсутностей», які впливають на роботу протоколу.

1. Відсутність аутентифікації повідомлень BGP, що не забезпечують цілісність, аутентифікацію та динамічну перевірку даних BGP в повідомленнях BGP: повідомлення BGP можуть бути підроблені, змінені, видалені або ретрансльовані випадково вузлом чи навіть навмисно сторонньою особою.

2. Відсутність перевірки повноважень мережевих (префіксних) оголошень: будь-який BGP-оголошувач може оголосити маршрут до будь-якого префіксу.

3. Відсутність перевірки аутентичності шляху автономної системи і атрибутів шляху: сам шлях і його окремі атрибути можуть бути змінені під час стрибків оголошення.

Дійсно, коли BGP-приймач отримує BGP-повідомлення, він без якихось додаткових підтверджень вважає, що повідомлення приходить від легітимного партнера, якщо ідентифікаційна інформація в повідомленні просто збігається з інформацією відомого партнера. Крім того, якщо оголошувач BGP обробляє інформацію про маршрутизації, він вважає, що префікси маршрутів в повідомленні були оголошені законно, і що всі автономні системи на маршруті оголошення правильно додали свій власний унікальний номер до маршруту та змінили атрибути шляху без внесення помилкових даних. Як зафіксовано в RFC 4360, «оператор (мережа), який покладається на інформацію, передану в BGP, повинен мати транзитивні довірчі відносини на шляху назад до джерела інформації» [2]. Втім, як правило, це не так, і оскільки все більше мереж підключається до Інтернету, це стає все важче.

Протокол BGP не містить всередині себе жодного механізму для перевірки коректності відправки інформації щодо керівних принципів та норм Інтернет-спільноти. Інтернет складається з взаємозв'язаних мереж. З іншої ж сторони, існує досить чітка ієрархія організацій, які займаються розподілом та делегуванням IP-адрес (номерних ресурсів Інтернету). Управління з присвоєння номерів в Інтернеті IANA (Internet Assigned Numbers Authority) знаходиться на вершині процесу розподілу адрес та розподіляє великі частини простору IP-адрес для IPv4 й IPv6 серед п'яти регіональних реєстрів Інтернету RIR (Regional Internet Registries), що охоплюють різні географічні регіони: ARIN для Північної Америки, LACNIC для Південної Америки та Карибського басейну, RIPE для Європи та Близького Сходу, APNIC

для Азіатсько-Тихоокеанського регіону та AFRINIC для Африки. Ці регіональні реєстри потім виділяють менші пули IP-адрес локальним інтернет-реєстрам LIR (Local Internet Registries) або безпосередньо постачальникам інтернет-послуг ISP (Internet Service Providers). LIR та ISP, в свою чергу, делегують IP-адреси іншим (меншим) інтернет-провайдерам та мережам. Тобто, більшість законних префіксів IP-мережі, які були законно виділені, дотримувалися певного шляху розподілу в ієрархії.

Крім вразливостей, описаних вище, існують вектори атак, пов'язані з механікою зв'язку BGP. Їх можна розділити на три наступні основні категорії.

1. Уразливості, пов'язані з повідомленнями BGP та кінцевим автоматом: існує безліч способів, якими повідомлення BGP може в кінцевому підсумку закрити сеанс BGP, ініціюючи видалення інформації про маршрутизацію, отриманої вузлами BGP.

2. Уразливості, пов'язані з безпекою транспортного рівня: враховуючи, що протокол BGP працює поверх TCP, він успадковує уразливості TCP, оскільки механізм безпеки транспортного рівня не використовується.

3. Уразливості, пов'язані з інфраструктурою, в якій працює протокол BGP: конфігурація, експлуатація та управління інфраструктурою, що підтримує BGP, можуть вплинути на правильну роботу BGP.

Основні мотиви, якими керуються пропозиції щодо безпеки BGP, які були затверджені IETF, суттєво відрізняються між собою, хоча здається, що перелік вразливостей можна звести до чітко розмежованих груп, але при їх вирішенні все знову змішується разом в єдине ціле. Мотиви дуже різняться, від необхідності захисту зв'язку BGP до розвитку фактично з нуля захищених моніторів для перевірки правильності роботи динаміків BGP, основні причини, через які розробники розширень вважали ці рішення хорошим, дуже різноманітні. Це, мабуть, впливало з того, що між цими пропозиціями не існує згоди щодо того, що потрібно забезпечити, яким чином воно має бути захищеним чи які є прийнятні компроміси. Деякі рішення, такі як Listen and Whisper [15], та використання списків вихідних даних стосуються лише суперечливих маршрутів, в той час, коли інші рішення стосуються всіх оголошень маршруту [14].

Крім того, є чіткий вплив перших пропозицій на останні рішення. Багато хто бере елементи першої пропозиції, вдосконалює деякі аспекти або чітко виступає проти якогось принципу та базуючись на цьому пропонує альтернативу. Наприклад, S-BGP [6] запропонував ієрархічну інфраструктуру PKI для перевірки розподілу префіксу IP, до якої пізніше також були включені інші рішення. Однак soBGP [6] запропонував більш децентралізовану структуру PKI з невеликою групою довірених організацій вгорі. А вже потім ESM [6] запропонував ще більш децентралізовану структуру PKI, де кожна автономна система видаватиме власні сертифікати, і лише у випадку конфлікту потрібно буде отримати цей сертифікат.

Крім того, оскільки зазвичай мотивація дизайнерів керується однією особливістю пропозиції рішення та охоплює більше аспектів безпеки BGP, розробник використовує елементи попередніх пропозицій чи іншого вже стандартизованого протоколу. Наприклад, розробники SPV [8] зосередилися на розробці більш ефективного механізму перевірки AS Path, ніж у S-BGP, але використовують централізовану інфраструктуру PKI на зразок S-BGP для перевірки походження префікса.

Нарешті, жодна з цих пропозицій не була повністю реалізована або використана, і лише невелика частина з них досі обговорюється в роботах, пов'язаних з безпекою BGP. Однак багато з цих робіт вплинули на розвиток безпечної та коректної маршрутизації між доменами SIDR (Secure Inter-Domain Routing), а деякі потенційно можуть вплинути на поточні послуги з моніторингу BGP, або Інтернет-провайдерами, або іншими організаціями.

З вивчення життєвого циклу розширень BGP видно, що жодна пропозиція щодо безпеки не була швидко реалізована та розгорнута, незалежно від її мотивації та розробки. Рішення, орієнтовані на безпеку на транспортному рівні, є чудовим прикладом: застарілий TCP-MD5 [12] досі використовується, а TCP-AO та GTSM [13] мають обмежене використання, хоча вони були стандартизовані відповідно 9 та 15 років тому. Навіть для TCP-MD5 пройшло майже десятиліття, щоб вийти за рамки обмеженого тестового використання.

Крім того, основна мотивація та розвиток розробок свідчать про відсутність згоди щодо того, що потрібно забезпечити та що забезпечити в контексті BGP. Деякі пропозиції стосуються забезпечення безпеки в окремих випадках, наприклад, коли існує конфлікт між двома маршрутами, тоді як інші охоплюють усі можливі випадки. Крім того, деякі пропозиції визначали перевірку походження префікса як таку, що стосується сертифіката та атестації розподілу блоку IP-адрес та авторизації транзиту, тоді як

інші визначали це щодо власного оголошення ресурсів або консенсусу Інтернет-провайдерів серед того, що Інтернет-провайдери вважають правильними. Це пов'язано з делегуванням довіри, що має на увазі різні пропозиції та їх залишкові вразливості.

Насправді проблематика визначеності меж в даному питанні дуже значна. Немає згоди за жодним питанням. Від впливу вразливостей одна на одну до того, що вважати вразливостями, а що прийнятними збоями, від того яким способом розподіляти номери автономних систем до того яку частину інфраструктури робити відкритою і так далі.

З іншого боку, якщо відкинути невизначеність головної мети, все одно залишається питання часу на тестування, яке насправді дуже значне і важливе. Спочатку кожне рішення не один раз тестується на тестових імітаційних моделях загального використання та специфічно розроблених. Після цього йде період обмеженого використання, коли розширення випробовують на робочій, проте замкнутій або ж обмеженій, мережі. І, як згадувалося раніше, такий випробувальний термін може тривати роками і в кінці кінців тестове розширення може втратити свою актуальність ще до повноцінного схвалення. На жаль, зменшити випробування за часом і випустити в загальний доступ не ідеально перевірений варіант призводить до зовнішньої небезпеки у випадку протоколу маршрутизації BGP і даний аспект складності поки що не вдалося зменшити.

А останньою причиною дуже рідкого впровадження пропозицій, щодо покращення безпеки BGP, можна назвати їх вартість. Всі рішення передбачають певний рівень модифікації конфігурації оголошувача маршрутів BGP або використовують пристрої, які діють як проксі, що потребують розширеної конфігурації, збільшуючи ризик потенційних помилок конфігурації, що впливають на доступність. І всі ці витрати лягають на плечі операторів мережі. Закономірним буде питання, а чи зобов'язані оператори мережі та Інтернет-провайдери забезпечувати безпеку маршрутизації? Адже є й інші способи підвищення безпеки, наприклад, моніторинг інформації про маршрутизацію BGP, який можна забезпечити самостійно або ж через спеціалізовані аутсорсингові компанії, такі як ThousandEyes.

Як варіант тимчасового виходу для користувачів, яким безпека потрібна зараз, а не коли знайдуть ідеальне рішення, можна запропонувати захищену маршрутизацію в якості додаткової послуги від провайдерів та мережевих операторів за окремі додаткові витрати. Клієнти, які хочуть покращити гарантії безпеки для трафіку до мережі та з неї, можуть домовитися з провайдером про додатковий рівень безпеки для маршрутизації їх повідомлень та потоків. Провайдери провайдерів та мережеві оператори передають це, дотримуючись усіх кращих практик безпеки оперативної безпеки BGP [14] та конкретно відстежуючи маршрути для цих клієнтів. У цьому випадку зусилля щодо захисту ISP були б досить вузькими, обмежуючи помилки неправильної конфігурації. Крім того, додаткові витрати, понесені провайдером, безпосередньо переносилися б на клієнтів.

Якщо безпечна маршрутизація не є обов'язковою функцією мережевих операторів, клієнтам доведеться платити за забезпечення маршрутизації, чого вони, можливо, не бажають робити. Положення може стимулювати клієнтів, що надають критичні послуги кінцевому користувачу, який працює з публічним Інтернетом, наприклад, Інтернет-банкінг, для забезпечення маршрутизації їх інформаційних потоків шляхом моніторингу, кращих практик та оновлення інформації про маршрутизацію в Регістрі Інтернет маршрутизації IRR (Internet Routing Registry), в ідеалі включаючи дійсну Авторизацію походження маршруту ROA (Route Origin Authorisation).

Висновки та перспективи подальших досліджень. В роботі показано, що при використанні в мережі протоколу BGP забезпечення безпеки передачі інформаційних потоків виходить на новий рівень важливості та цінності. В результаті проведеного аналізу були виділені 3 групи вразливостей для безпеки, які пов'язані з повідомленнями, безпекою на транспортному рівні та з інфраструктурою. Показано, що на сьогоднішній день, незважаючи на це, немає загальних вимог щодо безпеки, головних цілей покращення чи критеріїв кінцевого рішення.

Всі рішення, що були створені на момент написання статті і розглянуті в даній роботі, дуже різноманітні, деякі наслідують попередні пропозиції, деякі зовсім ні на що не схожі. Показано, що через відсутність чітко сформульованих вимог багато часу йде на вивчення запропонованих рішень, порівняння з раніше відомими та вирішення щодо їх практичного використання. Крім того описані етапи та проблеми тестування нових рішень від початкової теоретичної оцінки до можливості вводу в експлуатацію або ж відмови від подальшої розробки. Наведено обґрунтування великих часових витрат саме на процес тестування. Також представлені деякі з причин випадків відмови від представлення користувачам рішень, що успішно пройшли тестування. Згадується й економічний аспект, який відіграє

не останню роль у впровадженні нових рішень в загальну експлуатацію і лягає повністю на плечі провайдерів та операторів мережі, які, в свою чергу виражають супротив такому положенню справ.

Все це стало причинами того, що проблема вразливості мережі Інтернет не просто є актуальною, а й наближається до статусу такої, «що не вирішується» в найкоротший час для багатьох користувачів.

В роботі доведено, що на даний час найкращим способом забезпечення маршрутизації BGP є використання індивідуальної безпеки через застосування таких практик безпеки, як розробка фільтрів маршрутів, використання та оновлення інформації про маршрутизацію, наявну в IRR, та моніторинг. Проте, можливо, зараз ще не досягнутий той рівень розвитку технології BGP, при якому можливе використання безпечної маршрутизації за замовчуванням. Це може бути додатковою послугою, яка пропонується провайдерами або зовнішніми компаніями. Показано, що вже сьогодні є компанії, які надають послуги з моніторингу безпеки BGP на основі власної розгорнутої інфраструктури. Компанії, що надають контент-мережу (CDN), можуть також надавати послуги безпеки BGP своїм клієнтам на основі їх мережевої інфраструктури. Доведено, що користувачам, для яких безпека передачі інформаційних потоків в мережі Інтернет є критичним питанням вже зараз і які не можуть чекати розробки та впровадження розширення, яке б вирішило усі проблеми безпеки функціонування BGP, такий підхід був би необхідним та оптимальним вже зараз.

Список бібліографічних джерел

1. Apostolaki, M., Zohar, A. & Vanbever, L. (2017). Hijacking Bitcoin: Routing Attacks on Cryptocurrencies. *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 375–392.
2. Rekhter, Y., Sangli, S. R. & Tappan, S. (2006). RFC 4360: BGP Extended Communities Attribute.
3. Lepinski, M. & Sriram, K. (2017). RFC 8205: BGPsec Protocol Specification.
4. White, R. (2003). Securing BGP Through Secure Origin BGP. *The Internet Protocol Journal*, 2003, vol. 6, № 3.
5. McPherson, D. & Scudder, J. G. (2007). RFC 5065: Autonomous System Confederations for BGP.
6. Chandra, R., Chen, E. & Bates, T. (2000). RFC 2796: BGP Route Reflection – An Alternative to Full Mesh IBGP.
7. Snijders, J., Bagdonas, I., Patel, K., Heitz, J. & Hilliard, N. (2017). RFC 8092: BGP Large Communities Attribute.
8. Chen, E. (2000) RFC 2918: Route Refresh Capability for BGP-4.
9. Chandra, R. & Scudder, J. G. (2000). RFC 2842: Capabilities Advertisement with BGP-4.
10. Fernando, R., Sangli, S. R., Rekhter, Y., Chen, E. & Scudder, J. G. (2007). RFC 4724: Graceful Restart Mechanism for BGP.
11. Villamizar, C., Govindan, R. & Chandra, R. (1998). RFC 2439: BGP Route Flap Damping.
12. Heffernan, A. (1998). RFC 2385: Protection of BGP Sessions via the TCP MD5 Signature Option.
13. Touch, J., Mankin, A. & Bonica, R. P. (2010). RFC 5925: The TCP Authentication Option.
14. Savola, P., Gill, V., Pignataro, C., Meyer, D. & Heasley, J. (2007). RFC 5082: The Generalized TTL Security Mechanism (GTSM).
15. Subramanian, L., Roth, V., Stoica, I., Shenker, S. & Katz, R. H. (2004). Listen and Whisper: Security Mechanisms for BGP. *1st Symposium Networked System Design and Implementation*, pp. 14-16.

References

1. Apostolaki, M., Zohar, A. & Vanbever, L. (2017). Hijacking Bitcoin: Routing Attacks on Cryptocurrencies. *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 375–392.
2. Rekhter, Y., Sangli, S. R. & Tappan, S. (2006). RFC 4360: BGP Extended Communities Attribute.
3. Lepinski, M. & Sriram, K. (2017). RFC 8205: BGPsec Protocol Specification.
4. White, R. (2003). Securing BGP Through Secure Origin BGP. *The Internet Protocol Journal*, 2003, vol. 6, № 3.
5. McPherson, D. & Scudder, J. G. (2007). RFC 5065: Autonomous System Confederations for BGP.
6. Chandra, R., Chen, E. & Bates, T. (2000). RFC 2796: BGP Route Reflection – An Alternative to Full Mesh IBGP.
7. Snijders, J., Bagdonas, I., Patel, K., Heitz, J. & Hilliard, N. (2017). RFC 8092: BGP Large Communities Attribute.
8. Chen, E. (2000) RFC 2918: Route Refresh Capability for BGP-4.
9. Chandra, R. & Scudder, J. G. (2000). RFC 2842: Capabilities Advertisement with BGP-4.
10. Fernando, R., Sangli, S. R., Rekhter, Y., Chen, E. & Scudder, J. G. (2007). RFC 4724: Graceful Restart Mechanism for BGP.
11. Villamizar, C., Govindan, R. & Chandra, R. (1998). RFC 2439: BGP Route Flap Damping.
12. Heffernan, A. (1998). RFC 2385: Protection of BGP Sessions via the TCP MD5 Signature Option.
13. Touch, J., Mankin, A. & Bonica, R. P. (2010). RFC 5925: The TCP Authentication Option.
14. Savola, P., Gill, V., Pignataro, C., Meyer, D. & Heasley, J. (2007). RFC 5082: The Generalized TTL Security Mechanism (GTSM).
15. Subramanian, L., Roth, V., Stoica, I., Shenker, S. & Katz, R. H. (2004). Listen and Whisper: Security Mechanisms for BGP. *1st Symposium Networked System Design and Implementation*, pp. 14-16.

Рецензію надав к.т.н., доцент кафедри інформаційних технологій Київського національного університету будівництва і архітектури Щербина Олександр Андрійович

DOI: 10.36910/6775-2524-0560-2020-39-21

УДК: 681.3

Гринюк Сергій Васильович, асистент,

<https://orcid.org/0000-0002-0080-3167>

Поліщук Микола Миколайович, к.т.н., ст. викладач

<https://orcid.org/0000-0002-1218-5925>

Луцький національний технічний університет

ВИКОРИСТАННЯ ТЕХНОЛОГІЙ ШИФРУВАННЯ ІНФОРМАЦІЇ ДЛЯ БЕЗПЕЧНОЇ ПЕРЕДАЧІ В МЕРЕЖІ

Гринюк С.В., Поліщук М.М. Використання технологій шифрування інформації для безпечної передачі в мережі.

Технологія шифрування даних не тільки може шифрувати та дешифрувати дані, але й може реалізувати цифровий підпис, аутентифікацію та автентифікацію та інші функції, забезпечуючи таким чином конфіденційність, цілісність і підтвердження передачі даних по мережі. З метою підвищення безпеки даних у мережевому зв'язку в роботі розглядається гібридна система шифрування, яка використовує для шифрування та дешифрування потрійний алгоритм DES з високою безпекою, і два ключі шифруються за допомогою алгоритму RSA, забезпечуючи таким чином безпеку потрійного ключа DES та вирішуючи проблему управління ключами.

Ключові слова: безпека мережевого зв'язку, шифрування, алгоритм DES, алгоритм RSA, цифровий підпис.

Гринюк С.В., Полищук Н.Н. Использование технологии шифрования информации для безопасной передачи в сети. Технология шифрования данных не только может шифровать и дешифровать данные, но и может реализовать цифровой подписи, аутентификацию и проверку подлинности и другие функции, обеспечивая таким образом конфиденциальность, целостность и подтверждения передачи данных по сети. С целью повышения безопасности данных в сетевой связи в работе рассматривается гибридная система шифрования, которая использует для шифрования и дешифрования тройной алгоритм DES с высокой безопасностью, и два ключа шифруются с помощью алгоритма RSA, обеспечивая таким образом безопасность тройного ключа DES и решая проблему управления ключами.

Ключевые слова: безопасность сетевой связи, шифрования, алгоритм DES, алгоритм RSA, цифровая подпись.

Hryniuk S.V, Polishchuk M.M. Use information encryption technology for secure network transmission. Data encryption technology can not only encrypt and decrypt data, but can also implement digital signature, authentication and authentication and other functions, thus ensuring the confidentiality, integrity and authentication of data transmission over the network. In order to increase data security in network communication, the paper considers a hybrid encryption system that uses a triple DES algorithm with high security for encryption and decryption, and the two keys are encrypted using the RSA algorithm, thus ensuring the security of the DES triple key and solving the key management problem.

Keywords: network security, encryption, DES algorithm, RSA algorithm, digital signature.

Вступ. Завдяки розвитку комп'ютерних технологій для роботи та життя людей з'явилися великі можливості, а саме спілкування на відстані, обмін інформацією через мережу тощо. Але при цьому виникли проблеми інформаційної безпеки – це стан захищеності інформаційної системи, включаючи інформацію і інфраструктуру самої системи. Інформаційна система знаходиться в стані захищеності, якщо забезпечені її конфіденційність, доступність і цілісність. Конфіденційність (confidentiality) – гарантія того, що секретні дані доступні тільки користувачам, яким доступ дозволений (легальним або авторизованим користувачам). Доступність (availability) – гарантія того, що авторизовані користувачі завжди отримують доступ до даних. Цілісність (integrity) – гарантія збереження даними правильних значень, яка забезпечується заборонаю неавторизованим користувачам змінювати, модифікувати, видаляти або створювати дані. При цьому загроза – дія, яка направлена на порушення інформаційної безпеки системи, атака – реалізована загроза, а ризик – імовірнісна оцінка величини можливого збитку, який несе власник інформаційного ресурсу в результаті успішно проведеної атаки [3].

Економічні втрати, спричинені вразливістю безпеки інформаційної системи, з кожним роком зростають, а проблема безпеки стає все більш серйозною. Ризик інформаційної безпеки обмежує ефективне використання інформації, що в свою чергу впливає на економіку, національну оборону та навіть несе загрози національній безпеці. Іншими словами, інформаційна безпека має важливий вплив на розвиток сучасного суспільства, захист національної безпеки та соціальної стабільності, та має вирішальний вплив на успіх чи провал інформаційної революції.

Криптовалюта є ядром технологій інформаційної безпеки. Це ключ до реалізації конфіденційності та цілісності. [4]

Постановка проблеми. Популяризація комп'ютерних мережно-комунікаційних технологій є проявом інформатизації сучасного суспільства. Це дає змогу різним сферам діяльності змінити традиційний спосіб передачі та швидкість розповсюдження інформації, яка сприяє покращенню

ефективності роботи. Технологія шифрування даних є головною складовою при передачі інформації через комп'ютерні мережі та спрямована на підвищення її безпеки [5]. В даний час існує багато проблем, пов'язаних із безпекою мережі:

По-перше, різноманітні мережеві хакери використовують власну технологію та інші переваги для комп'ютерної мережі як носія якогось прямого порушення нормального робочого процесу, крадіжка інформації про користувачів та інші дії;

По-друге, деякі злочинці використовують комп'ютерні вразливості для мережевого вірусу, який не тільки впливає на використання комп'ютерних функцій, але й через руйнівну силу самого вірусу, внаслідок чого користувачі комп'ютерної мережі паралізовані, що робить інформацію про користувача або файли повністю відкритою;

По-третє, завдяки швидкому розвитку комп'ютерних технологій з'явилися різноманітні веб-ресурси, що привертають велику кількість користувачів, які будуть з різних причин публікувати деякі помилкові негативні новини та дискредитувати інших і при цьому впливати на життя та роботу інших людей. Щоб цього уникнути почали використовувати технологію шифрування даних при передачі через мережу [6].

Виклад основного матеріалу. Шифрування – практичний засіб забезпечення секретності інформації. Сучасні методи шифрування є математичними перетвореннями (алгоритми), в яких повідомлення розглядаються як числа або алгебраїчні елементи в деякому просторі. Ці алгоритми відображають область «змістовних повідомлень» і область «беззмістовних повідомлень». Повідомлення, з числа «змістовних», також є вихідними даними алгоритму шифрування та називаються відкритим текстом (cleartext), беззмістовні повідомлення, які є результатом роботи алгоритму шифрування, називаються зашифрованим текстом (ciphertext). Якщо знехтувати сенсом повідомлення, то вхідні дані алгоритму шифрування зручно називати вихідним текстом (plaintext), який не зобов'язаний бути осмисленим [2].

Існує два класи криптосистем – симетричні і асиметричні. У симетричних схемах шифрування (класична криптографія) секретний ключ шифрування збігається з секретним ключем дешифрування (рис. 1).

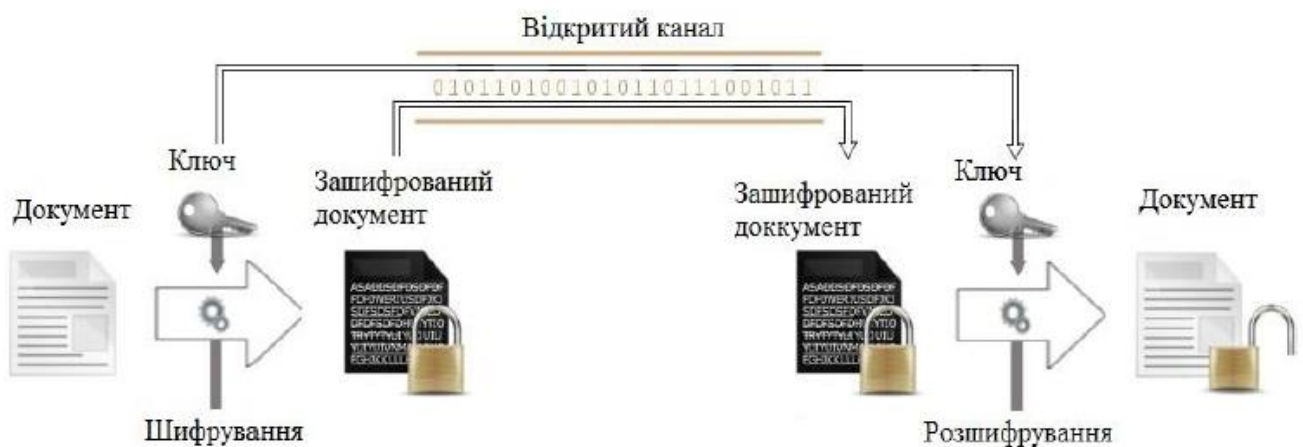


Рис. 1. Схема роботи симетричних алгоритмів

Популярні стандартні симетричні алгоритми шифрування даних є DES (Data Encryption Standard) та AES (Advanced Encryption Standard). AES забезпечує кращий захист, використовує 128-бітові ключі (працює з 192- і 256- бітними ключами) і має високу швидкість роботи, кодуючи за один цикл 128-бітний блок на відміну від 64-бітного блоку DES. В даний час AES є найбільш поширеним симетричним алгоритмом шифрування [1].

У симетричних алгоритмах проблема у ключі. По-перше, криптостійкість багатьох симетричних алгоритмів залежить від якості ключа, це висуває підвищені вимоги до служби генерації ключів. По-друге, важливою є надійність каналу передачі ключа другому учаснику секретних переговорів. Проблема з n ключами виникає навіть у системі з двома абонентами, а в системі з абонентами, охочим обмінюватися секретними даними за принципом «кожен з кожним», потрібно $n * (i - 1)2$ ключів, які

повинні бути згенеровані і розподілені надійним чином. Тобто кількість ключів пропорційна квадрату кількості абонентів, що при великій кількості абонентів робить задачу 13 надзвичайно складною. Несиметричні алгоритми, які засновані на використанні відкритих ключів, знімають цю проблему [3]. В асиметричних схемах шифрування (криптографія з відкритим ключем) відкритий ключ шифрування не збігається з секретним ключем дешифрування (рис. 2).

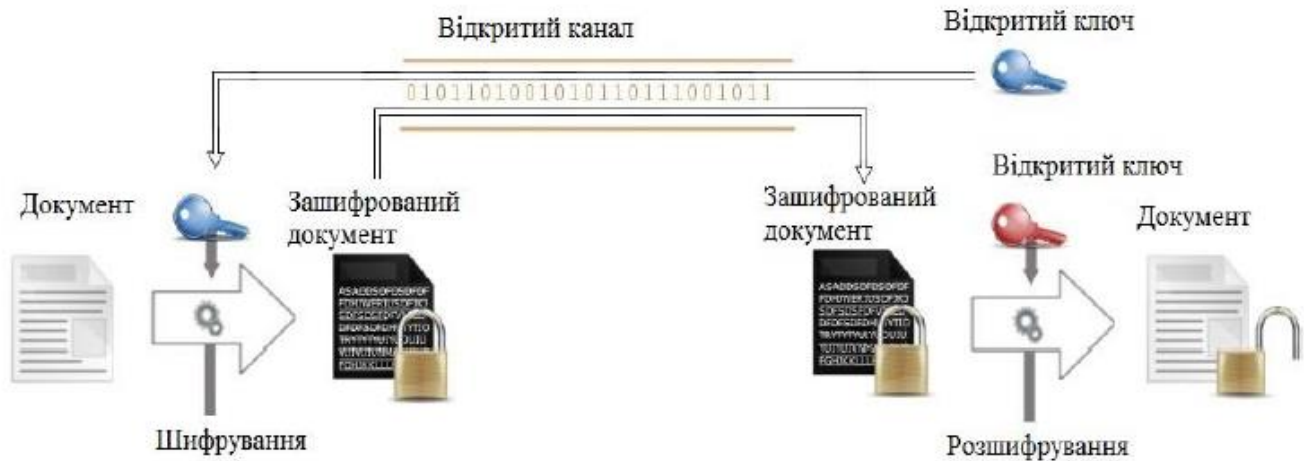


Рис. 2. Схема роботи асиметричних алгоритмів

Особливість шифрування з відкритим ключем в тому, що одночасно генерується унікальна пара ключів. Текст, зашифрований одним ключем, розшифровується тільки з використанням другого. У моделі кріптосхеми з відкритим ключем три учасники: відправник, отримувач і зломисник. Завдання відправника в тому, щоб по відкритому каналу зв'язку передати повідомлення в захищеному вигляді. Одержувач генерує на своєму боці два ключа: відкритий E і закритий D . Закритий ключ D (особистий) абонент зберігає в захищеному місці, а відкритий ключ E він може передати всім, з ким хоче підтримувати захищені відносини. Для шифрування тексту служить відкритий ключ, але розшифрувати цей текст можна тільки за допомогою закритого ключа. Тому відкритий ключ передається відправнику в незахищеному вигляді. Відправник, використовуючи відкритий ключ одержувача, шифрує повідомлення X і передає його одержувачу. Одержувач розшифровує повідомлення своїм закритим ключем D . Очевидно, що числа, одне з яких служить для шифрування тексту, а інше - для дешифрування, не можуть бути незалежними один від одного, а значить, є теоретична можливість обчислення закритого ключа з відкритого. Однак це пов'язано з величезним обсягом обчислень, які вимагають відповідно величезного часу [3].

Для того щоб в мережі усі n абонентів мали можливість не тільки приймати зашифровані повідомлення, але і самі посилати такі, кожен абонент повинен мати власну пару ключів E і D . В мережі буде $2n$ ключів: n відкритих ключів для шифрування і n секретних ключів для дешифрування. Так вирішується проблема масштабованості – квадратична залежність кількості ключів від числа абонентів в симетричних алгоритмах замінюється лінійною залежністю в несиметричних.

Хоча інформація про відкритий ключ не є секретною, її потрібно захищати від підробок, щоб зломисник під ім'ям легального користувача не нав'язав свій відкритий ключ, після чого за допомогою свого закритого ключа він зможе розшифровувати всі повідомлення, що посилаються легальному користувачеві, і відправляти свої повідомлення від його імені.

З огляду на особливості технології шифрування симетричного ключа та відкритого ключа, у практичних додатках їх буде два види технології шифрування в поєднанні з гібридною системою шифрування, тобто поєднанням DES та RSA, для передачі мережі шифрування даних за допомогою DES та шифрування за допомогою RSA

Цей спосіб не тільки забезпечує безпеку даних та підвищує швидкість шифрування та дешифрування. Комбінація DES та RSA показана на рис. 3.

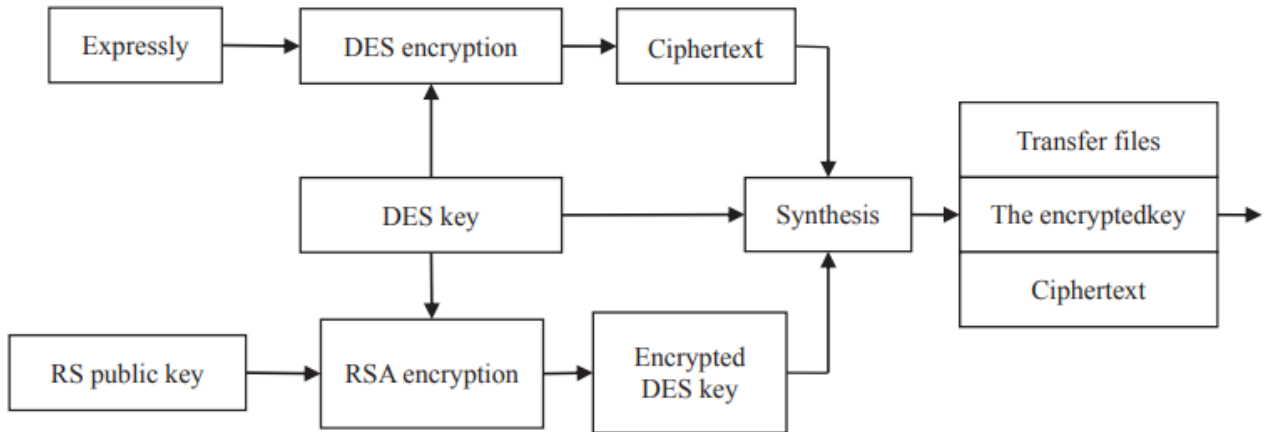


Рис. 3. DES та RSA поєднані за діаграмою технології шифрування

По-перше, відправник використовує алгоритм DES для шифрування інформації plaintext симетричним ключем для отримання знака ciphertext, а потім шифрує симетричний ключ, використовуючи відкритий ключ RSA приймача для отримання зашифрованого DEA або ключ IDEA, зашифровує ciphertext та зашифрований ключ разом із мережею до приймача. Після отримання інформацію про ciphertext приймач розшифровує ключ власним ключем, щоб отримати ключ DES, а потім розшифровує ciphertext за допомогою ключа і, нарешті, отримує інформацію про plaintext, відіграючи, таким чином, роль захисту конфіденційної інформації.

Хоча DES має широкий спектр застосувань і його можна легко отримати з різних джерел. Ключова довжина DES - це лише 56 біт, тому легко атакувати, надійність шифрування не може відповідати сучасним потребам безпеки. Тому використання двох 56-бітових змішувань ключа шифрування та дешифрування, довжина ключа 112, надійність шифрування значно зросла, що є потрійним шифрування DES (3DES). Принципова схема алгоритму 3DES показана на hbc/ 4, де ключ K1, K2 генерується випадковим чином.

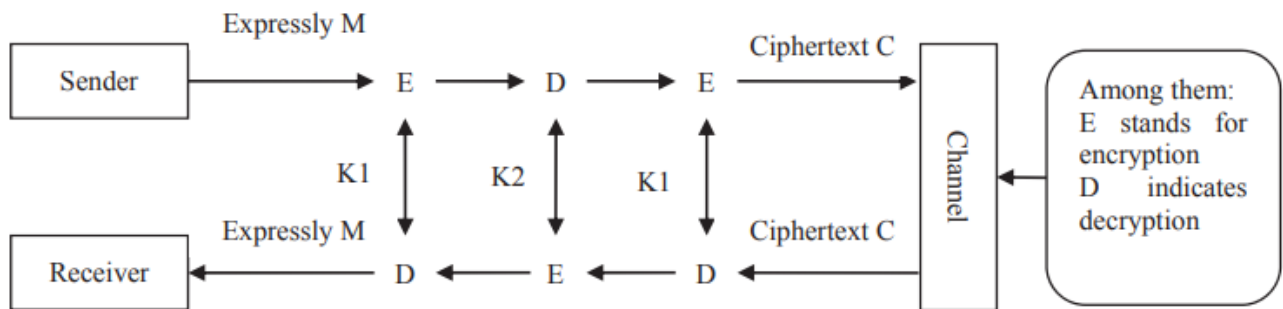


Рис.4. Схема алгоритму 3DES

Потрійна реалізація алгоритму DES включає два класи: DES клас та ТРИ клас. Основні функції класу DES:

```
public static byte[] encrypt(byte[] ourceword64, int[] [] iSubKeys)//encryption
public static byte[] decrypt(byte[] bCipher-text, int[] [] SubKeys)//
decrypt
THREE main categories of functions:
public static byte[] encrypt(byte[] ourceword, int[] [] SubKeys1, int[] []
SubKeyS2)// decryption
public static byte[] decrypt(byte[] sword,int[] [] SubKeys1, int[] []
SubKeyS2)//encryption
return plaintext
```

Криптосистема відкритого ключа може використовуватися в технології цифрового підпису. В системі цифрового підпису RSA є однією з найпоширеніших технологій цифрового підпису. Вимоги до технології цифрового підпису повинні бути: жодна інша особа не може підробити підпис, підпис може бути перевірений, і підписувач не може заперечити власний підпис. Цифровий підпис RSA використовує модель аутентифікації в криптосистемі відкритого ключа, як показано на рис. 5. Схема підпису RSA та алгоритм RSA дуже схожа на шифр, не однакова з приватним ключем для підписання відкритий ключ для перевірки, таким чином гарантуючи, що інші люди не можуть підробити підпис, всі власники підписів відкритого ключа можуть бути перевірені, і підписувач не може заперечувати підпис згодом. Для створення можна використовувати: класи, визначені в Java, публічні та приватні ключі RSA, клас підпису в Javaх. Пакет безпеки, крім того, що використовується для підписання, може також використовуватися для перевірки цифрових підписів. Метод `initverfy()` об'єкта підпису передається у відкритий ключ, виконує метод підтвердження `()` та перевіряє вихідні дані з інформацією про підпис у своєму підписі.

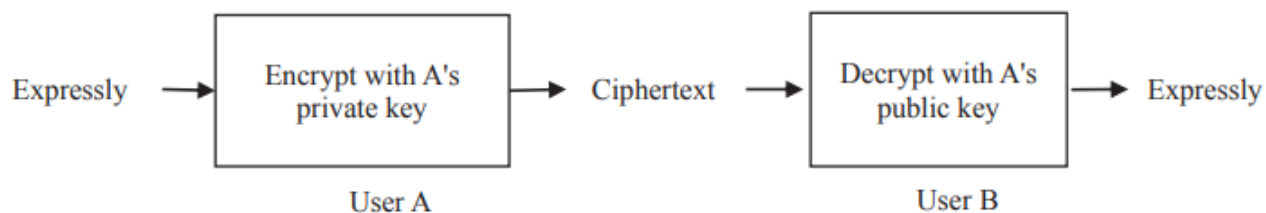


Рис. 5. Модель аутентифікації, що використовується для цифрових підписів

Висновки. Технологія комп'ютерних інформаційних мереж у світі швидко розвивається, що робить інформаційну мережу популярною, люди активно користуються послугами інтернету. Питання безпеки також є серйозною загрозою. Проблема безпеки комп'ютерної мережі - це довгий шлях, який потрібно пройти для вирішення проблеми Завдання та технологія шифрування даних для вирішення цієї проблеми є основним засобом підтримки безпеки комп'ютерної мережі. З розвитком інформаційної ери, комп'ютерне шифрування даних та технології також зробили більш поглиблене дослідження, дослідники повинні продовжувати здійснювати технологічні інновації, а технологія шифрування даних стала ефективною гарантією підтримки безпеки комп'ютерної мережі Система шифрування даних, розроблена в цій статті повністю використовує технологію шифрування даних і цифрову технологію підпису, яка не тільки вирішує ключову проблему управління, але й забезпечує цілісність і достовірність даних.

Список бібліографічних посилань

1. Баричев С.Г. Основи сучасної криптографії / С.Г. Баричев, Р.Е. Серов. // Учебный посібник. – М.: Горячая линия, Телеком, 2002. – 152с.:іл.
2. Мао В. Сучасна криптографія. Теорія і практика / В. Мао. – М.: Вільямс, 2005. – 768 с.
3. Олифер В. Г. Комп'ютерні мережі. Принципи, технології, протоколи / В. Г. Олифер, Н. А.Олифер. // Підручник для вузів. – 5-е вид. – СПб.: Питер, 2016. – 922с.: іл.

References

4. S.H. Yin. Computer network security in the data encryption technology [J]. Electronic Technology and Software Engineering, 2015, 18: 214
5. W. Tang. Cryptography and network security technology foundation [M]. Beijing: Mechanical Industry Press. 2004.
6. David A. Solomon, Mark E. Russinovieh. Inside Microsoft Windows 2000. Microsoft Press. 2000. 8.

DOI: 10.36910/6775-2524-0560-2020-39-22

УДК: [004.02/.032/.421] + 621.391 +004.031.42+007.2

Козубцова Леся Михайлівна

<https://orcid.org/0000-0002-7866-8575>

Військовий інститут телекомунікацій та інформатизації ім. Героїв Крут, м. Київ, Україна

УДОСКОНАЛЕНА МЕТОДИКА ДІАГНОСТУВАННЯ КІБЕРНЕТИЧНОЇ ЗАХИЩЕНОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ З УРАХУВАННЯМ ДЕСТРУКТИВНИХ КІБЕРНЕТИЧНИХ ВПЛИВІВ

Козубцова Л.М. Удосконалена методика діагностування кібернетичної захищеності інформаційної системи з урахуванням деструктивних кібернетичних впливів. В статті проаналізовано відомі спроби рішень наукової задачі з розрахунку кібернетичної захищеності інформаційної системи спеціального призначення. Встановлено, що на даний час існуючі рішення не враховують при розрахунку деструктивний інформаційний вплив, а тому результат носить статичний характер. Запропонований математичний апарат методики забезпечує розрахунок кібернетичної захищеності інформаційної системи спеціального призначення для моделі найгіршого варіанту настання події загрози нульового дня.

Ключові слова: методика, оцінка, кібернетична захищеність, інформаційна система спеціального призначення, деструктивний кібернетичний вплив.

Козубцова Л.М. Усовершенствованная методика диагностирования кибернетической защищенности информационной системы с учетом деструктивных кибернетических воздействий. В статье проанализированы известные попытки решения научной задачи по расчету кибернетической защищенности информационной системы специального назначения. Установлено, что в настоящее время существующие решения не учитывают при расчете деструктивное информационное влияние, а потому результат носит статический характер. Предложенный математический аппарат методики обеспечивает расчет кибернетической защищенности информационной системы специального назначения для модели наихудшего варианта наступления события угрозы нулевого дня.

Ключевые слова: методика, оценка, кибернетическая защищенность, информационная система специального назначения, деструктивный кибернетический влияние.

Kozubtsova L. M. Improved method of diagnostics of cybernetic protection of the information system taking into account destructive cybernetic influences. The article analyzes well-known attempts to solve the scientific problem of calculating the cybernetic security of a special-purpose information system. It is established that currently existing solutions do not take into account destructive information influence in the calculation, and the result is static. The proposed mathematical apparatus of the method provides the calculation of cybernetic security of a special-purpose information system for the model of the worst-case scenario of a zero-day threat event.

Keywords: methodology, assessment, cybernetic security, special-purpose information system, destructive cybernetic influence.

Постановка завдання і зв'язок її з важливими науковими завданнями. Відповідно до мети, об'єкта, предмета та визначеного наукового завдання дисертаційного дослідження необхідно: визначити в умовах реальних деструктивних кібернетичних впливах числових значень кібернетичної захищеності кожного компонента (K_j) засобу (Z_i) інформаційної системи спеціального призначення (ІС СП) ($P_{K3(K_j Z_i)}$) та кожного засобу (Z_i) ІС СП ($P_{K3(Z_i)}$), а також ІС СП в цілому ($P_{K3(S)}$), див рис. 1 [1].

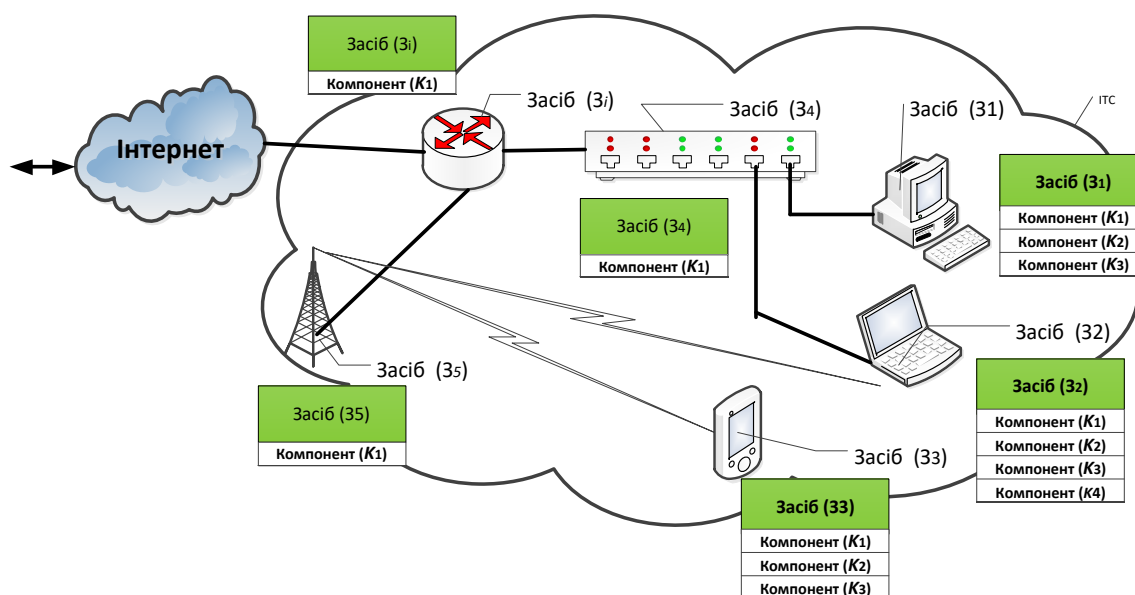


Рис. 1 Фрагмент умовної ІС СП

Аналіз останніх досліджень і публікацій. Аналіз останніх досліджень і публікацій за обраним напрямком досліджень представлено в наступних публікаціях.

В роботі [2] запропоновано, що посадовим особам, які відповідальні за кібернетичну безпеку ІТС (адміністративному персоналу) надається формальний апарат для кількісної оцінки поточного стану кібернетичної безпеки у будь-який момент на заданому часовому інтервалі, визначених кроків функціонування ІТС асоційованого з тривалістю періоду оновлення формалізованої бази шаблонів (правил, сигнатур) кібернетичних атак. Поряд з відносною простотою, наочністю, наявні особливості, а саме, що чисельні величини показників у складі приведених виразів мають сенс для визначення тільки у ході практичного моделювання, натурального експерименту чи практичної повсякденної роботи, як результат роботи відповідних автоматизованих функцій у складі програмного забезпечення діяльності адміністративного персоналу ІТС.

Запропонована методика [3], ґрунтується на методі анкетування, надає можливість одержати числову характеристику комплексного показника оцінки рівня кібербезпеки держави, значення якого дозволяє визначати необхідність розробки належних заходів щодо підвищення результативності власних систем кібернетичної безпеки. Однак не наводиться з допомогою якого математичного апарату досягається заданий результат.

Виходячи з суспільно важливої наукової задачі авторами у роботі [4] започатковано і обговорено єдиний підхід до побудови методики оцінки кібернетичної захищеності ІТС організації. Подальші наукові пошуки дозволили вибудувати методика оцінки кібернетичної захищеності системи зв'язку організації [5]. Вона також ґрунтувалася на методі експертного анкетування системних адміністраторів. Подальші вдосконалення [6] дозволили розширити межі застосування методика, а саме для оцінки кібернетичної захищеності інформаційно-телекомунікаційного вузла зв'язку Збройних Сил України.

Слід зазначити що всі розглянуті методики [2 – 6] дозволяють здійснити розрахункову оцінку кібернетичної захищеності інформаційно-телекомунікаційного вузла зв'язку Збройних Сил України на деякий момент часу t_0 , за якого активний деструктивний інформаційний вплив на ІС СП відсутній $F_{ДІВ} = 0$. Іншими словами методики дозволяють визначити рівень досягнення виконання політики безпеки на ІС СП, яка б гарантувала захист її від кібернетичних впливів з деякою ймовірністю, яка прямувала б до 1 за умови відсутності загрози нульового дня.

Таким чином, поза уваги вище перелічених методик лишається можливість обчислення (прогнозування) оцінки кібернетичної захищеності ІС СП на деякий момент часу $t_{ДІВ}$, за якого на ІС СП здійснюється активний кібернетичний ДІВ $F_{ДІВ} = 1$ в результаті чого втрачено деякий актив (Ак), або функціональність засобів (сервісів) W_{FZi} . Саме через відсутність в обігу понять актив організації, втрати цього активу та відсутності коефіцієнта функціональності засобів (сервісів) W_{FZi} , не можливо зрозуміти реальний зміст методики оцінки кібернетичної захищеності ІС СП, вибудувати математичний апарат її розрахунку, моделювати найгірші варіанти стану кібернетичної захищеності ІС СП в результаті настання загрози нульового дня.

Мета статті. Апробувати результат дисертаційного дослідження щодо удосконаленої методики діагностування кібернетичної захищеності ІС СП з урахуванням деструктивних кібернетичних впливів.

Виклад основного матеріалу.

Вихідні дані методики: $I_{чкз}$ – часткові індикатори кібернетичної захищеності політики безпеки; Z_i – кількість засобів ІС СП; K_j – кількість компонентів у складі засобу (Z_i) ІС СП; W_{Zi} – ваговий коефіцієнт кожного засобу (Z_i) зі складу ІС СП; W_{Kji} – ваговий коефіцієнт компонента (K_j) зі складу засобу (Z_i) ІС СП; W_{FZi} – коефіцієнти функціональності засобів (сервісів) (Z_i); W_{Fkj} – коефіцієнти функціональності компонентів (сервісів) засобу (K_j); W_{FS} – коефіцієнт функціональної працездатності всієї системи S , яка має виконувати цільову функцію за умов ДІВ; час ($t_0, t_{ДІВ1}, \dots, t_{ДІВn}$) контрольні часові точки вимірювань.

Необхідно знайти: $P_{Kz(Kji)}$ – показник (ймовірність) кібернетичної захищеності компонента (K_j) зі складу засобу (Z_i) ІС СП; $P_{Kz(Zi)}$ – показники (ймовірність) кібернетичної захищеності засобу (Z_i) ІС СП; $P_{Kz(S)}$ – показник (ймовірність) кібернетичної захищеності всієї ІС СП.

Прийняті обмеження:

Введемо наступні обмеження та припущення, що час t_0 , - момент часу коли деструктивний інформаційний вплив відсутній ДІВ=0; час $t_{ДІВ1}, \dots, t_{ДІВn}$ – момент часу фіксації дії деструктивного інформаційного впливу ДІВ $\neq 0$.

В даній удосконаленій методиці використовується наступні методи випробувань: експериментально-розрахунковий, порівняльний.

Етап 1. Реалізація заходів з категоріювання та розкладання ІС СП на компоненти та елементи щодо вразливості кібернетичного впливу

- 1) складну ІС СП потрібно розкласти на: засоби (Z_i), а засоби на її компоненти (K_j);
- 2) експертна група уповноважених фахівців кібернетичної безпеки призначає наступні вагові коефіцієнти: для кожного засобу (Z_i) ІС СП складної системи (W_{Z_i}); кожного компонента (K_j) кожного засобу (Z_i) ІС СП – $W_{K_jZ_i}$; коефіцієнти функціональності засобів (сервісів) W_{FZ_i} . Коефіцієнт функціональності засобів (сервісів) W_{FZ_i} приймає значення [1, 0] (табл. 1).

Таблиця 1. Приклад підходу до формування коефіцієнта функціональності засобу

№ п/п	Індикатор реакції системи	Контрольний засіб фіксації функціональної роботи	Обсяг інформації	W_{FZ_i}
1.	відмова пристрою	Акт звірки конфігурації системи до і після кібернетичного впливу	вдала спроба (без компрометації)	0
			вдала спроба (з компрометацією)	0,5
			не вдала спроба (компрометація)	1

Перед початком розрахунку кібернетичної захищеності ІС СП визначається ваговий коефіцієнт кожного компонента (K_j) у кожному засобі (Z_i) ІС СП – $W_{K_jZ_i}$.

Ваговий коефіцієнт – $W_{K_jZ_i}$ призначений врегулювати питання логічної важливості в порядку спадання компонентів у засобу (Z_i) ІС СП.

Вагові коефіцієнти розраховується згідно якісної шкали Сааті. Для цього необхідно здійснити експертну оцінку попарних порівнянь вагових значень компонентів (K_j).

Отримання коефіцієнтів важливості компонент $W_{K_jZ_i}$ здійснюється із використанням методу рангових оцінок [7; 8].

Чим більше ранг, тим більша вага компоненти $W_{K_jZ_i}$, а сума всіх значень коефіцієнтів одного засобу (Z_i) ІС СП рівна 1, як наведено у виразі (4).

Для визначення коефіцієнтів експертами створюється матриця елементів $W_{K_jZ_i}$, як відносна оцінка елемента за якісною шкалою Сааті [2].

Отримана матриця $W_{K_jZ_i}$ діагональна, симетрична, що дозволяє визначити елементи всіх рядків зі одним відомим. Важливість $W_{K_jZ_i}$ кожної компоненти оцінюється з позиції впливу на працездатність, кібернетичної захищеність та безпечний стан ІС СП.

Очевидно, що на абонентському терміналі, смартфоні найвищий $W_{K_jZ_i}$ присвоюється операційній системі (ОС) і найнижчий Bluetooth. Проте з точки функціональної важливості передача текстових або голосових повідомлень неможлива без працездатного радіотерміналу GSM, CDMA, Wi-Fi, Bluetooth. В той же час втрачається сенс при виході з ладу планшету або пошкодженні ОС (див. рис. 1).

Для врахування наслідків впливу активних кібернетичних впливів в розрахункові формули кібернетичної захищеності [5; 6] кожному засобу (Z_i) ІС СП в обов'язковому порядку вводиться коефіцієнт функціональності компонентів (сервісів) засобу W_{FK_j} .

Етап 2. Розрахунок показників $P_{K3(K_jZ_i)}$ – кібернетичної захищеності кожного компонента (K_j) зі складу засобу (Z_i) ІС СП.

1) визначаються найбільш критичні часткові індикатори кібернетичної захищеності ($I_{чкз}$) з політики безпеки. Уточнений перелік часткових індикаторів ($I_{чкз}$) для кожної кожного компонента (K_j) засобу (Z_i) ІС СП визначає експертна група уповноважених фахівців кібернетичної безпеки.

2) розподіляються часткові індикатори ($I_{чкзи}$) за кожним компонентом (K_j) кожного засобу (Z_i) ІС СП;

3) перевіряються засоби (Z_i) ІС СП на відповідність налаштувань параметрів кібернетичної захищеності за частковими індикаторами.

Звірка здійснюється шляхом перевірки налаштувань кожного компонента (K_j) засобу (Z_i) ІС СП у відповідності з рекомендаціями об'єкта прийняття рішення (експертна група уповноважених фахівців кібернетичної безпеки) з переліком часткових індикаторів.

Частковий індикатор ($I_{чкз}$) приймає вихідне значення "1" або "0" за наступних умов, якщо: налаштування компонента (K_j) засобу (Z_i) ІС СП відповідають політиці безпеці, тоді ($I_{чкз}$) = "1", а в іншому випадку ($I_{чкз}$) = "0" - налаштування компонента (K_j) засобу (Z_i) ІС СП не відповідають політиці безпеці. Кількість ($I_{чкз}$) для різних компонентів засобів має різну кількість. Результати обчислювань заносяться до табл. 2.

Таблиця 2. Матриця показників часткових індикаторів $I_{чкзi}$ для компонент (K_j) кожного засобу (Z_i) ІС СП

Компонента засобу	Значення показників часткових індикаторів ($I_{чкз}$)					g_k
	1	2	3	g_k	
Компонент №1 засобу №1	0	1	1	0	g_1+g_0
Компонент №2 засобу №1	1	1	1		1	g_1+g_0
Компонент №3 засобу №1	1	0	0		0	g_1+g_0
Компонент №1 засобу №2	0	0	1		0	g_1+g_0
.....	g_1+g_0
Компонент №1 засобу №N	1	0	1	0	g_1+g_0

Показник $P_{кз(KjZi)}$, обчислюємо за формулою (1):

$$P_{кз(KjZi)} = \frac{\sum_{i=1}^k g_i}{\sum_{i=1}^k (g_0 + g_i)} = \frac{\sum_{i=1}^k g_i}{g_k} \quad (1)$$

де g_k – кількість питання, що відповідають відповідній компоненті K_j ; g_l – кількість індикаторів ($I_{чкз}$), які приймає значення “1” для відповідної компоненти K_j ; g_0 – кількість індикаторів ($I_{чкз}$), які приймає значення “0” для відповідної компоненти K_j .

Етап 3. Обчислення показника $P_{кз(Zi)}$ – кожного засобу (Z_i) зі складу ІС СП.

Показник кібернетичної захищеності $P_{кз(Zi)}$ обчислюється за формулою (2), як зважену та нормовану оцінку індикаторів стану кібернетичної захищеності всіх компонентів (j) кожного засобу (Z_i) ІС СП.

$$P_{кз(Zi)} = \frac{\sum_{j=1}^m (P_{кз(KjZi)} \times W_{Fkj} \times W_{KjZi})}{\sum_{j=1}^m (W_{Fkj} \times W_{KjZi})}, \quad (2)$$

де m – кількість компонентів (K_j) у кожному засобі (Z_i) ІС СП; W_{Fkj} – коефіцієнт функціональності компоненту (сервісу) (K_j) засобу (Z_i) ІС СП.

Результати обчислювань заносяться до табл. 3.

Таблиця 3. Результати обчислювань

№	Компоненти	K_1				K_j			$P_{кз(Zi)}$
		$P_{кз(K1Zi)}$	W_{FK1}	W_{K1Zi}	...	$P_{кз(K1Zi)}$	W_{FK1}	W_{K1Zi}	
1	Компонент №1 засобу №1				...				
2	Компонент №2 засобу №1								
..
K	Компонент №1 засобу №N								

Етап 4. Розрахунок $P_{кз(S)}$ – кібернетичної захищеності ІС СП в цілому.

Кількісним показником для оцінки кібернетичної захищеності складної системи є $P_{кз(S)}$ – ймовірність того, що в складній системі (ІС СП), всі її засоби та їх компоненти будуть захищені від кібернетичного втручання ДІВ та функціонуватимуть в штатному режимі.

Показник кібернетичної захищеності $P_{кз(S)}$ ІС СП загалом розраховується за формулою (3), як зважена та нормована оцінка показників стану кібернетичної захищеності всіх засобів складної системи:

$$P_{кз(S)} = \sum_{i=1}^L (P_{кз(Zi)} \times W_{Fzi} \times W_{Zi}), \quad (3)$$

де L – кількість засобів (Z_i) у складі ІС СП;

Тоді, W_{Fzi} коефіцієнт функціональності засобу (Z_i) ІС СП в цілому обчислюється за формулою (4):

$$W_{Fzi} = \sum_{i=1}^m (W_{FKj}) \leq 1, \quad (4)$$

де W_{zi} – ваговий коефіцієнт засобу (Z_i) в складній системі ІС СП.

Результати обчислювань $P_{кз(S)}$ заносяться до табл. 4.

Таблиця 4. Результат розрахунку показників $P_{K3(S)}$ кібернетичної захищеності ІС СП в цілому

Засоби Z_i	$P_{K3(Z_i)}$	W_{FZ_i}	W_{Z_i}	$P_{K3(S)}$
Засіб №1				Формула (3)
Засіб №2				
Засіб №3				
.....	
Засіб №N				

Етап 5. Обробка, аналіз та оцінка результатів випробувань.

Методи контролю за дослідним зразком складної системи:

- зовнішній огляд за системою оповіщення про інциденти (при її наявності);
- проведення вимірювань швидкості передачі файлів (відео, графічних, текстових матеріалів);
- проведення розбірливості голосових повідомлень;
- зовнішній огляд стану зразку у випадку здійснення на нього кібернетичного впливу.

Критерії, при виконанні яких фрагмент випробувальної ІС СП вважається таким, що витримав випробування. Оцінка здійснюється на всіх етапах випробувань. ІС СП вважається такою, що пройшла перевірку випробування на кібернетичну захищеність, якщо за результатами розрахунку кібернетичної захищеності станом на час ($t_0, t_{ДІВ1}, t_{ДІВ2}$) задовольнило критерії табл. 5.

Таблиця 5. Критерії кібернетичної захищеності

№ п/п	$P_{K3(S)}$	Рівень кібернетичної захищеності	Кольорове трактування
1	$0,75 \leq P_{K3(S)} \leq 1$	високий	зелена
2	$0,4 \leq P_{K3(S)} \leq 0,75$	середній	жовта
3	$0 \leq P_{K3(S)} \leq 0,4$	низький	червона

Опис постановки експериментальної частини реалізації методики діагностування кібернетичної захищеності інформаційної системи з урахуванням деструктивних кібернетичних впливів

Етап 1. Реалізація заходів з категоріювання та розкладання ІС СП на компоненти та елементи щодо вразливості кібернетичного впливу.

1) Керівник комісії з перевірки кібернетичної захищеності формує наступні групи фахівців за напрямками та відповідальністю:

- група №1 фіксації зміни стану функціонування ІС СП під час діагностування;
- група №2 створення ДІКВ – відпрацювання умовних дій в ролі «хакера»;
- група №3 розрахунку – розраховують кібернетичну захищеність ІС СП на всіх етапах діагностування;

група №4 умовні користувачі (АРМ ІС СП), здійснюють фіксацію передачі голосових, текстових, графічних даних, відео потоку. Здійснюють інструментальне діагностування, вимірювання, оцінювання кількісних та якісних показників.

Порядок взаємодії учасників випробувань за даною методикою:

- фахівці контролю та фіксації безпосередньо з групою розрахунку;
- керівник випробувань через команду здійснення ДІКВ з – умовним хакером.

Забороняється особам, що здійснюють кібернетичний вплив (№2) повідомляти початок настання події ДІКВ групі №1.

2) розкласти складну систему (ІС СП), яка визначена на тестування на наступні складові:

- на засоби (Z_i) ІС СП;
- на компоненти (K_j) засобу (Z_i) ІС СП;

3) розподілити і закріпити перелік часткових індикаторів ($I_{чк3}$) за кожною компонентою (K_j) кожного засобу (Z_i) ІС СП;

- 4) призначити вагові коефіцієнти:
 - кожному засобу (Z_i) ІС СП (W_{Z_i});
 - кожній компоненті (K_j) кожного засобу (Z_i) ІС СП – W_{KjZ_i} ;
 - функціональності W_{FZ_i}, W_{Fkj} ;

5) розрахувати станом на момент часу (t_0) «відсутні деструктивного інформаційного впливу» числове значення кібернетичної захищеності:

- кожного компонента (K_j) зі складу засобу (Z_i) ІС СП – $P_{K3(KjZi)}$;
- кожного засобу (Z_i) ІС СП $P_{K3(Zi)}$;

всієї складної ІС СП – $P_{K3(S)}$;

б) результат розрахунку комісія заноситься до таблиці порівнянь стану кібернетичної захищеності протоколу випробувань та здійснити оцінку відповідності стану кібернетичної захищеності ІС СП станом на момент часу (t_0).

7) оперативна нарада за результатами підготовчого етапу методики. Постановка завдання на відпрацювання першого етапу методики випробувань.

Етап 2. Розрахунок $P_{K3(S)}$ – кібернетичної захищеності ІС СП в цілому станом на (t_0) за етапами 2-5.

Етап 3. Розрахунок числового значення кібернетичної захищеності ІС СП станом на момент часу ($t_{ДВ1}$) за результатами «активної фази ДІВ»:

1) за командою керівника випробувань група №2 здійснює «активну фазу ДІВ» на складну систему із зазначенням часу ($t_{ДВ1}$).

2) група №1 переходить в посилений режим очікування;

3) група №3 працює в штатному режимі. З надходженням вихідних результатів від групи № 1 про зміни стану системи або реакції, приступають до обчислювань кібернетичної захищеності;

4) з настанням часу ($t_{ДВ1}$), групи №2 за допомогою набору СПЗ здійснює:

рекогносцировку мережевого обладнання;

отримання IP-адресу для роботи в мережі;

аналіз вразливостей;

злом шляхом складання словників злому для підбору імен, облікових записів і паролів до них;

злом та проникнення до системи;

проводить Ddos-атаку.

За результатами сканування мережевого обладнання та отриманих вихідних даних здійснює санкціоновані (з позиції дозволених) кібернетичні втручання (впливи) для досягнення мети.

5) перевіряється (відслідковується):

групою №1 за допомогою функціонування штатної системи фіксації, контролю та оповіщення про зміни стану компоненти засобу(ів), зміни функціонування засобу(ів) або всієї складної ІС СП. Результати контролю надаються групі №3 у формі вихідних даних (час настання події ($t_{кв1ф}$), засіб, компоненти засобу, загроза чи інше) та рекомендовані значення коефіцієнта функціональності – W_{Fzi} , W_{Fkj} ;

групою №4 за допомогою інструментальних засобів оцінюють функціональні зміни стану компоненти засобу(ів), зміни функціонування засобу(ів) та надають пропозиції щодо W_{Fkj} відповідної компоненти засобу(ів), зміни функціонування засобу(ів) в залежності від реакції на кібернетичне втручання. Відбувається фіксація фактичного часу настання події зміни функціонування компоненти засобу(ів) або зміни функціонування всього засобу(ів) ($t_{ДВ1фзф}$);

б) група №3 здійснює:

Розрахунок за методикою числового значення кібернетичної захищеності всієї ІС СП $P_{K3(S)}$ станом на час ($t_{ДВ1ф}$);

результати обчислювань заносяться до таблиці порівнянь стану кібернетичної захищеності в протоколі випробувань із зазначенням часу ($t_{ДВ1ф}$).

7) Здійснити оцінку:

відповідності стану кібернетичної захищеності ІС СП станом на момент ($t_{ДВ1}$)

порівняти час реакції системи оповіщення про настання кібернетичних інцидентів ($t_{ДВ1}$) з ($t_{ДВ1ф}$), та ($t_{ДВ1фзф}$). Він має бути ($t_{ДВ1} \approx t_{ДВ1ф} \approx t_{ДВ1фзф}$) в межах прийнятної норми (згідно формуляру, техпаспорту);

часу затраченого на проведення проникнення і злому.

На всіх етапах перевірки групи №1-4 здійснюють фіксацію змін технічного стану та відхилення від нормального функціонування, як окремих складових (компонентів засобів) так і засобів в цілому та системи в цілому.

Таким чином: в запропонованій методиці на відміну від методик [2 – 6] можливо здійснити розрахункову оцінку кібернетичної захищеності ІТС на деякий момент часу $t_{ДВ}$, за якого здійснено активний кібернетичний вплив на цю систему $F_{ДВ} = 1$ в результаті чого, організація може втратити деякий актив (Ак); математичний апарат методики забезпечує розрахунок кібернетичної захищеності складної системи (ІС СП) для моделі найгіршого варіанту настання події загрози нульового дня.

Складання переліку часткових індикаторів діагностування кібернетичної захищеності компонентів ІС СП

Одним з відповідальних завдань, що покладається на експертну групу уповноважених фахівців кібернетичної безпеки є складання актуального та адекватного переліку часткових індикаторів.

Кількість ($I_{чкз}$) для різних компонентів засобів (Z_i) має різну кількість.

В дисертаційній роботі пропонується скласти перелік часткових індикаторів кібернетичної захищеності компонентів для трьох рівнів захищеності, а в сукупності за допомогою всіх трьох констатувати максимальний рівень захищеності.

В роботі при розробці індикаторів діагностування кібернетичної захищеності інформаційної системи використано найкращі практики Nist, DoD та у відповідності до вимог Інструкції Міністерства оборони США "Cybersecurity Activities Support to DoD Information Network Operations" від 07.03.2016 № 8530.01, Cyber incident handling program. Chairman of the joint chiefs of staff manual CJCSM 6510.01B. 10 July 2012.

Загальні відомості про перелік групових індикаторів діагностування різного рівня стану кібернетичного захисту ІС СП подано у зведеній табл. 6.

Зазначені переліки часткових індикаторів діагностування кібернетичної захищеності компонентів розроблено і практично апробовано в рамках розробки програми та методики визначальних відомчих випробувань засобів широкосмугового мультисервісного радіодоступу мережі McWiLL на замовленні МО України.

Таблиця 6. Загальні відомості про перелік групових індикаторів діагностування різного рівня стану кібернетичного захисту ІС СП

Перелік питань / Часткові індикатори кібернетичної захищеності	Рівень кібернетичної захищеності		
	Низький	Середній	Високий
Перелік групових індикаторів діагностування початкового рівня стану кібернетичного захисту інформаційної	+	-	-
група питань №1 Вимоги з організаційного захисту інформаційної системи	+	-	-
група питань №2 Виконання вимог з технічного захисту інформаційної системи від витоків по акустичному каналу інформації про керування прав до доступу (адміністрування)	+	-	-
група питань №3 Виконання вимог з технічного захисту інформаційної системи від витоків візуальної інформації про керування прав до доступу (адміністрування)	+	-	-
група питань №4 Виконання вимог з технічного захисту від загроз викрадення носіїв матеріальних інформації	+	-	-
група питань №5 Виконання вимог з організації та спроможності забезпечення захищеності інформаційної системи шляхом розмежування прав доступу	+	-	-
група питань №6 Виконання вимог з захищеності інформаційної системи від заходів розвідки інфраструктури	+	-	-
група питань №7 Виконання вимог з захищеності інформаційної системи від заходів кібернетичного впливу на функціонування інфраструктури	+	-	-
група питань №8 Виконання вимог з захищеності інформаційної системи від заходів кібернетичного впливу на функціонування інфраструктури шляхом застосування системи захисту інфраструктуру	+	-	-
група питань №9 Виконання вимог з виконання вимог з програмного захисту	+	-	-
група питань №10 Діагностування моделі внутрішнього і зовнішнього порушника.	+	-	-
Перелік групових індикаторів діагностування середнього рівня стану кібернетичного захисту інформаційної системи	-	+	-
група питань №1. Індикатор оцінки та аналізу вразливостей	-	+	-
група питань №2. Індикатор керування вразливостями	-	+	-
група питань №3. Індикатор захищеності від шкідливого програмного забезпечення	-	+	-

група питань №4. Індикатор моніторингу безпеки інформації	-	+	-
група питань №5. Індикатор обробки кіберінцидентів	-	+	-
група питань №6. Індикатор моніторингу інсайдерської діяльності користувачів	-	+	-
група питань №7. Індикатор попередження мережевої розвідки	-	+	
Перелік групових індикаторів діагностування високого рівня стану кібернетичного захисту інформаційної системи	-	-	+
група питань №1. «Авторизація неавторизованих та авторизованих мережевих пристроїв»	-	-	+
група питань №2. «Авторизація неавторизованого та авторизованого програмного забезпечення»	-	-	+
група питань №3. «Безпечна конфігурація для апаратного та програмного забезпечення»	-	-	+
група питань №4. «Безперервна оцінка вразливостей та їх виправлення»	-	-	+
група питань №5. «Контроль використання адміністративних привілеїв»	-	-	+
група питань №6. «Контроль документування подій в ІТС»	-	-	+
група питань №7. «Захист електронної пошти та веб-браузерів»	-	-	+
група питань №8. «Захист від шкідливого програмного забезпечення»	-	-	+
група питань №9. «Обмеження та контроль мережевих портів, протоколів та сервісів»	-	-	+
група питань №10. «Можливість відновлення даних»	-	-	+
група питань №11. «Безпечні конфігурації для мережевого обладнання»	-	-	+
група питань №12. «Захист мережевого периметру»	-	-	+
група питань №13. «Захист даних»	-	-	+
група питань №14. «Контрольований доступ»	-	-	+
група питань №15. «Контроль бездротового доступу»	-	-	+
група питань №16. «Моніторинг та управління обліковими записами»	-	-	+
група питань №17. «Відпрацювання навичок з інформаційної безпеки та проведення тренінгів для усунення недоліків»	-	-	+
група питань №18. «Захист прикладного програмного забезпечення»	-	-	+
група питань №19. «Настанова з реагування на інциденти»	-	-	+
група питань №20. «Тести на проникнення»	-	-	+

Загальні відомості щодо практичної реалізації методики розрахунок кібернетичної захищеності складної системи у формі спеціалізованого програмного забезпечення

Практична реалізація у вигляді спеціалізованого програмного забезпечення (СПЗ) методики забезпечує розрахунок кібернетичної захищеності складної системи, а саме для інформаційно-телекомунікаційної системи було виконано в контексті виконання доручення Начальника військ зв'язку Збройних Сил України – начальника Головного управління Зв'язку та інформаційних систем Генерального штабу Збройних Сил України Військового інституту телекомунікацій та інформатизації на виконання оперативного завдання на проведення дослідження на тему «Комплексна методика оцінки ефективності функціонування системи зв'язку Збройних Сил України за основними характеристиками» в частині що стосується розробки «Методика оцінки показника кіберзахищеності, як складової стійкості системи зв'язку Збройних Сил України» від 25.11.2016. З метою покращення якості впровадження результатів теоретичних досліджень у практику та застосування військ (сил), зниження впливу негативних умов і факторів на організацію проведення досліджень на заходах оперативної та бойової підготовки, а також розвиток методичної бази, що застосовується під час досліджень на заходах оперативної та бойової підготовки Збройних Сил України тому методика експериментально перевірялася на дослідження на військових навчаннях під час стратегічного командно-штабному

навчанні з органами військового управління, військами (силами) Збройних Сил України "Непохитна стійкість – 2017" в період з 11.09.2017 по 26.09.2017 р. офіцерами-дослідниками Наукового центру зв'язку та інформатизації Військового інституту телекомунікацій та інформатизації та підтвердили задовільний характер.

Висновки. Найважливішими науковими й практичними результатами дослідження стали:

1. Удосконалено методику діагностування кібернетичної захищеності інформаційної системи спеціального призначення. В запропонованій методиці на відміну від відомих забезпечено здійснити розрахункову оцінку кібернетичної захищеності інформаційної системи спеціального призначення на деякий момент часу $t_{див}$, за якого здійснено активний деструктивний інформаційно-кібернетичний вплив на цю систему $F_{див} = 1$ з метою прогнозування і запобігання втратити деяких актив (Ак). Математичний апарат методики забезпечує розрахунок кібернетичної захищеності інформаційної системи спеціального призначення для моделі найгіршого варіанту настання події загрози нульового дня.

2. Надано практичні рекомендації щодо підходу з визначення коефіцієнтів функціональності компонентів складної інформаційної системи спеціального призначення які вразливі деструктивним інформаційним та кібернетичним впливом.

Перспективи подальших досліджень доцільно зорієнтувати на обґрунтуванні методики діагностування кібернетичної стійкості функціонування інформаційної системи спеціального призначення в кібернетичному просторі.

Список бібліографічного опису

1. Козубцов І.М., Козубцова Л.М. Постановка завдання на розробку методики оцінки кібернетичної захищеності інформаційно-телекомунікаційної системи. *Міжнародна науково-практична конференція «Спільні дії військових формувань і правоохоронних органів держави: проблеми та перспективи»* (м. Одеса 12-13 вересня 2019 р.) Військова академія, 2019. С. 228 – 229.
2. Хусаїнов П.В. Показник кібернетичної безпеки автоматизованої системи у часі. *Збірник наукових праць ВІТІ*. Київ, 2015. Вип. № 1. С. 101 – 111.
3. Кудінов В.А. Методика оцінки рівня кібербезпеки в Україні. *Матеріали всеукраїнської науково-практичної конференції „Кібербезпека в Україні: правові та організаційні питання”* (Одеса, 21 жовтня 2016). Одеса. С. 151 – 152.
4. Черноног О.О., Козубцов І.М., Кутаєв В.В., Козубцова Л.М., Терещенко Т.П. Обговорення єдиного підходу до побудови методики оцінки кібернетичної захищеності ІТС організації. *Міжнародна науково-практична конференція “Застосування інформаційних технологій у підготовці та діяльності сил охорони правопорядку”* (14-15 березня 2018 року). – Харків. Національна академія Національної гвардії України, 2018. С. 15 – 16.
5. Козубцов І.М., Козубцова Л.М., Кутаєв В.В., Терещенко Т.П. Методика оцінки кібернетичної захищеності системи зв'язку організації. *Сучасні інформаційні технології у сфері безпеки та оборони*. 2018. №1 (31). С. 43 – 46.
6. Кутаєв В.В., Радченко М.М., Козубцова Л.М., Терещенко Т.П. Методика оцінки кібернетичної захищеності інформаційно-телекомунікаційного вузла зв'язку. *Збірник наукових праць ВІТІ*. К.: ВІТІ, 2018. № 2. С. 67 – 76.
7. Ротштейн А.П. Интеллектуальные технологии идентификации: нечеткие множества, генетический алгоритм, нейронные сети. *Винница: УНИВЕРСУМ*, 1999. 320 с.
8. Хусаїнов П.В. Методика визначення раціональної послідовності надання інформаційних повідомлень оператору системи захисту. *Збірник наукових праць ВІТІ НТУУ „КПІ”*. Київ, 2006. Вип. № 3. С. 148 – 155.

References

1. Kozubov I. M., Kozubova L. M. (2019) Statement of the problem to develop methods for the assessment of cyber security in information and telecommunication systems. International scientific and practical conference "Joint actions of military formations and law enforcement agencies of the state: problems and prospects" (Odessa, September 12-13, 2019) Military Academy, Pp. 228 – 229.
2. Khusainov P. V. (2015) Indicator of cybernetic security of an automated system in time. Collection of scientific works of VITI. Kiev. Vol. # 1. Pp. 101 – 111.
3. Kudinov V. A. (2016) Methodology for assessing the level of cybersecurity in Ukraine. Materials of the all-Ukrainian scientific and practical conference "Cybersecurity in Ukraine: legal and organizational issues"(Odessa, October 21, 2016). Odessa. Pp. 151 – 152.
4. Chernonog O. O., Kozubtsov I. M., Kutsaev V. V., Kozubtsova L. M., Tereshchenko T. P. (2018) Discussion of a unified approach to the construction of a methodology for evaluating the cybernetic security of an organization's its. International scientific and practical conference " application of information technologies in the training and activities of law enforcement forces" (March 14-15, 2018). – Kharkov. National Academy of the National guard of Ukraine. Pp. 15 – 16.
5. Kozubtsov I. M., Kozubtsova L. M., Kutsaev V. V., Tereshchenko T. P. (2018) Methods for evaluating the cybernetic security of the organization's communication system. Modern information technologies in the field of security and defense. No. 1 (31). Pp. 43 – 46.
6. Kutsaev V. V., Radchenko M. M., Kozubtsova L. M., Tereshchenko T. P. (2018) Methods for evaluating cybernetic security of an information and telecommunications node. Collection of scientific works of VITI. K: VITE. Vol. # 2. Pp. 67 – 76.
7. Rotshtein A. p. Intelligent identification technologies: fuzzy sets, genetic algorithm, neural networks. Vinnytsia: UNIVERSUM, 1999. 320 p.
8. Khusainov P. V. (2006) Method of determining the rational sequence of providing information messages to the operator of the protection system. Collection of scientific works of VITI NTUU "KPI". Kiev. Vol. # 3. Pp. 148 – 155.

DOI: 10.36910/6775-2524-0560-2020-39-23

УДК: 004.42(07)

Костючко Сергій Миколайович, к.т.н.

<https://orcid.org/0000-0002-1262-6268>

Сахнюк Андрій Анатолійович, магістр

Мельник Катерина Вікторівна, к.т.н., доцент

<https://orcid.org/0000-0002-9991-582X>

Луцький національний технічний університет

ОБХІД ЗАХИСТУ САЙТІВ ЗА ДОПОМОГОЮ SQL-ІН'ЄКЦІЙ ТА ЗАХИСТ ВІД НИХ

Костючко С.М., Сахнюк А.А., Мельник К.В. Обхід захисту сайтів за допомогою SQL-ін'єкцій та захист від них. У статті розкрито суть несанкціонованого доступу до бази даних сайту; можливості та основи використання SQL-ін'єкцій; способи захисту від несанкціонованого вторгнення; переваги та недоліки POD.

Ключові слова: SQL-ін'єкція, база даних, сайт, PDO.

Костючко С.Н., Сахнюк А.А., Мельник К.В. Обход защиты сайтов с помощью SQL-инъекций и защита от них. В статье раскрыта суть несанкционированного доступа к базе данных сайта; возможности и основы использования SQL-инъекций; способы защиты от несанкционированного вторжения; преимущества и недостатки POD.

Ключевые слова: SQL-инъекция, база данных, сайт, PDO.

Kostiuchko S.M., Sahnjuk A.A., Melnyk K.V. Bypass site protection by means SQL injections and protection against them. The article reveals the essence of unauthorized access to the web-site database; possibilities of using SQL-injections; ways to protect against unauthorized intrusion; POD advantages and disadvantages.

Keywords: SQL injection, database, site, PDO.

Постановка проблеми.

Через різні вразливості веб-серверів і нескладні процедури, атаки на веб-сервери зросли, в основному за рахунок впровадження сценаріїв ASP або PHP. Атаки з використанням SQL ін'єкцій стали основним напрямком, в той час як процес компіляції веб-сервера переважає в сценарії, ігноруючи явище тестування безпеки програмного коду, що призводить до великої кількості надання інтерактивних лазівок в роботі на веб-сервері. Зловмисники можуть використовувати сервер, конфігурувати базу даних дефектів і розробляти структуру незаконних запитів за допомогою програм або сценаріїв вторгнення на сервер, отримувати дозволи адміністратора веб-сайту і отримувати відповідний контент баз даних.

Виклад основного матеріалу й обґрунтування отриманих результатів.

Ін'єкція SQL - це вразливість веб-безпеки, яка дозволяє зловмиснику втручатися у запити, які подає програма до своєї бази даних. Як правило, зловмисник дозволяє переглядати дані, які вони зазвичай не можуть отримати. Це може включати дані, що належать іншим користувачам, або будь-які інші дані, до яких сама програма може отримати доступ. У багатьох випадках зловмисник може змінювати або видаляти ці дані, викликаючи постійні зміни у змісті чи поведінці програми.

У деяких ситуаціях існує можливість застосувати атаку ін'єкцій SQL, щоб поставити під загрозу базовий сервер чи іншу інфраструктуру.

Успішна атака ін'єкцій SQL може призвести до несанкціонованого доступу до конфіденційних даних, таких як паролі, дані кредитної картки або особиста інформація користувача.

В дослідженні було перевірено роботу підриву логіки програми за допомогою ін'єкцій SQL, де є можливість змінити запит, задля перешкоджання логіки програми, але також існує велика різноманітність і інших вразливих ситуацій, атак та методів ін'єкцій SQL, які виникають у різних ситуаціях. Деякі поширені приклади введення SQL включають:

- Отримання прихованих даних, де можна змінити SQL-запит, щоб повернути додаткові результати.

- UNION-атаки, де можливо отримати дані з різних таблиць баз даних.

- Вивчення бази даних, де можливо отримати інформацію про версію та структуру бази даних.

- Сліпа ін'єкція SQL, коли результати запиту, яким керує зловмисник, не відповідають результатам програми.

Підрив логіки програми

Розглянемо програму, яка дозволяє користувачам входити з іменем користувача та паролем. Якщо користувач подає ім'я користувача Andrew та пароль 123456, програма перевіряє облікові дані, виконуючи наступний SQL-запит:

```
SELECT * FROM accounts WHERE username = 'andrew' AND password = '123456'
```

Якщо запит повертає реквізити користувача, то реєстрація успішна. В іншому випадку реєстрація не відбудеться.

Якщо, наприклад, в поле паролю Ввести знак " ' " може виникнути помилка, яка надасть нам потрібну інформацію про базу даних чи про файли сайту. Наприклад, як зображено на рисунку нижче.

Error: Failure is always an option and this situation proves it	
Line	49
Code	0
File	/var/www/mutillidae/process-login-attempt.php
Message	Error executing query: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '''' at line 1
Trace	#0 /var/www/mutillidae/index.php(96): include() #1 {main}
Diagnostic Information	SELECT * FROM accounts WHERE username='andrew' AND password='''

Рис. 1 – Помилка при відмові в реєстрації користувача

Для перевірки можливості використання SQL коду в полі реєстрації можна в полі password ввести 123456' and 1=1#. Після вводу в разі незахищеності системи реєстрація пройде успішно. Вид запиту після вводу пароля:

```
SELECT * FROM accounts WHERE username = 'andrew' AND password = '123456' and 1=1#
```

Щоб перейти на відповідний акаунт наприклад andrew, але пароль нам не відомий можна використати or замість and після чого запит буде мати вигляд:

```
SELECT * FROM accounts WHERE username = 'andrew' OR password = '123456' or 1=1#
```

Також існує можливість ввійти, як будь-який користувач, без пароля просто за допомогою поля username. Наприклад, введення імені користувача admin# та порожнього пароля призводить до наступного запиту:

```
SELECT * FROM accounts WHERE username = 'admin#' AND password = 'aaaaaa'
```

Цей запит повертає користувача, ім'я користувача якого є, admin і успішно входить в систему. Так як після 'admin' стоїть знак коментаря #, то все, що йде після нього, буде ігноруватись. Тому запит в дійсності буде таким:

```
SELECT * FROM accounts WHERE username = 'admin' #
```

Підрив логіки програми з більш складнішим захистом.

Візьмем більш складну систему яка блокує різні символи крім букв. Тоді при вводі будь-якого знаку буде видаватись така помилка:

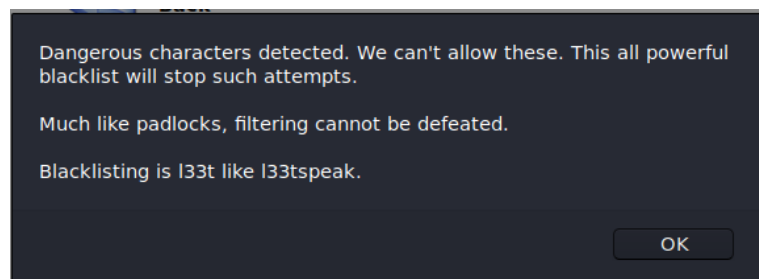


Рис. 2 – Помилка при вводі на стороні клієнта

Дана помилка відбувається тільки на стороні клієнта, і до серверу запит не відправляється, тому даним способом доступ до бази не буде отримано. Але, якщо будуть введені коректні символи, то зв'язок з базою відбудеться хоча б, щоб перевірити параметри авторизації. На даному етапі можна скористатися програмою Burp Suite та переглянути які пакети надсилаються:

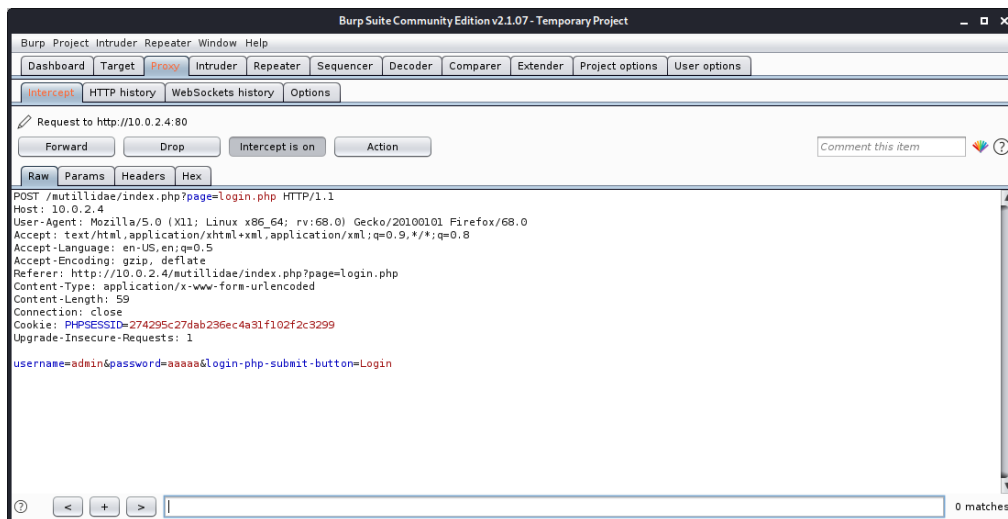


Рис. 3 – Інтерфейс програми Burp Suite

Дана програма може "притримати" пакети, які вже вийшли з клієнта, а попередня помилка не буде завадою для отримання доступу до бази. В цій програмі також можливо дещо підправити відправлені дані, а саме переправити пароль на вже відомий код '123456' or 1=1#.

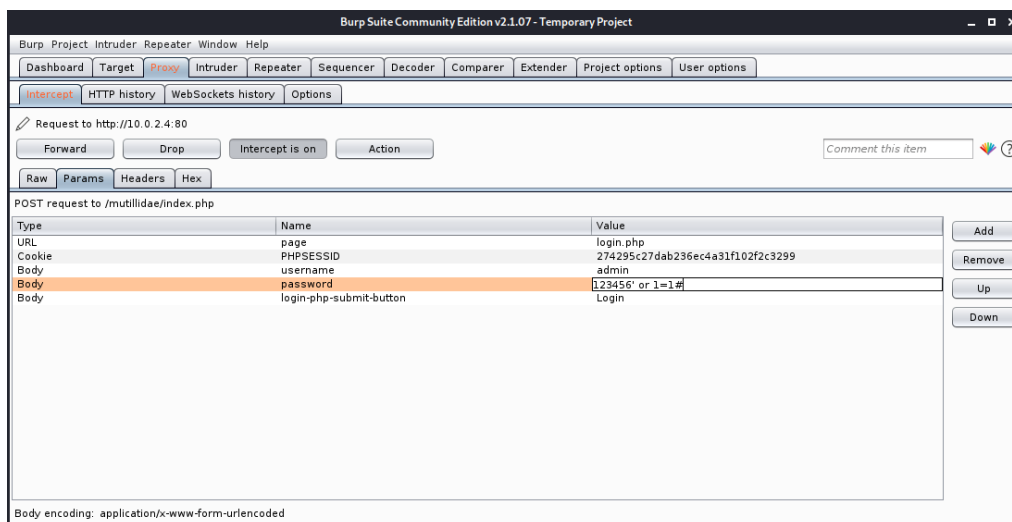


Рис. 4 – Деякі можливості програмного продукту Burp Suite

Виявлення вразливості ін'єкцій SQL

Більшість уразливостей ін'єкцій SQL можна швидко та надійно знайти за допомогою веб-сканера вразливості Burp Suite.

Ін'єкцію SQL можна виявити вручну за допомогою систематичного набору тестів проти кожної точки входу в програму. Зазвичай це стосується:

- Подання символу єдиної цитати та пошуку помилок чи інших аномалій.
- Подання деякого специфічного для SQL синтаксису, який оцінює базове (вихідне) значення точки введення та інше значення та шукає систематичні відмінності в отриманих відповідях додатків.
- Посилають логічні умови, такі, як OR 1=1 і OR 1=2, і шукаємо відмінності у відповідях додатка чи сайту.
- Подання корисних навантажень, призначених для запуску затримок у часі при виконанні в межах SQL-запиту, та пошуку відмінностей у часі, необхідному для відповіді.
- Подання корисних навантажень OAST, призначених для запуску позадіапазонної мережевої взаємодії при виконанні в рамках SQL-запиту, а також для моніторингу будь-яких результуючих взаємодій.

Введення SQL в різні частини запиту

Більшість уразливостей для введення SQL виникає в WHERE пункті SELECT запиту. Цей тип ін'єкції SQL, як правило, добре зрозумілий досвідченим тестерам.

Але вразливості введення SQL можуть в принципі виникати в будь-якому місці в межах запиту та в різних типах запитів. Найбільш поширені інші місця, де виникає ін'єкція SQL, це:

- У UPDATE висловлюваннях, в межах оновлених значень або WHERE пункту.
- У INSERT висловлюваннях у межах вставлених значень.
- У SELECT висловлюваннях, в межах назви таблиці чи стовпця.
- У SELECT заявах, у межах ORDER BY.

Захист сайту від SQL-ін'єкцій

Так як ми вже визначили, що SQL-ін'єкції дуже небезпечні тому потрібно мати надійний захист від даних атак. Тому одним з виходів використовувати PHP Data Objects або PDO. PDO (PHP Data Objects) - рівень абстракції для запитів вашої бази даних і є приголомшливою альтернативою MySQLi, оскільки він підтримує 12 різних драйверів баз даних. MySQL - це найпопулярніша база даних. Вона також дуже добре поєднується з PHP, саме тому ця пара технологій добре підтримується у світі PHP.

Якщо фактично знаєте, що єдиною базою даних SQL, якими ви користуєтесь, є або MySQL, або MariaDB, слід вибрати PDO.

Існує думка, що головна перевага PDO полягає в тому, що він переноситься з бази даних в базу даних але насправді це не є так. Насправді, дуже рідко приймається рішення щодо переключення баз даних на конкретний проект, а саме для компаній середнього рівня дана перевага є недоречною. Незважаючи на це, зазвичай, як правило, вважають за краще дотримуватися поточної технології, яка використовується, якщо тільки немає обґрунтованих причин втрачати значну кількість часу та грошей для переносу.

Справжня перевага PDO полягає в тому, що використовується практично подібний API для будь-якої з безлічі баз даних, які він підтримує, тому вам не потрібно вивчати нову для кожної. Названі параметри також, безсумнівно, є величезною перевагою для PDO, оскільки ви можете повторно використовувати однакові значення в різних місцях запитів. На жаль, ви не можете використовувати одні і ті ж названі параметри не один раз із вимкненим режимом емуляції.

Робота підготовлених заявок PDO

Для простішого пояснення, підготовлені заявки PDO працюють так:

1. Підготуйте SQL-запит із порожніми значеннями, як заповнювачі або знаком питання, або ім'ям змінної з двокрапкою, що передує йому, для кожного значення.
2. Прив'яжіть значення або змінні до заповнювачів
3. Виконувати запит одночасно.

Недоліки PDO

Однак при використанні визначених виразів спільно з PDO необхідно знати деякі нюанси, щоб уникнути неприємних ситуацій. Наприклад, в MySQL клієнта деякі запити, складені за допомогою визначених виразів, тому не можуть бути виконані, а так само вони не використовують кеш, що може уповільнити роботу вашого web-додатка.

Гарантована безпека при використанні визначених виразів звучить добре, але розробники не повинні приймати PDO і інші види абстракції, зумовлені вираженням, як абсолютний захист від злому. Будь-які вхідні дані повинні перевірятися, PDO - додаткова лінія оборони. Це розширення не закриває все безліч вразливостей, за допомогою яких може бути завдано шкоди вашій інформації, але в той же час, PDO непогано справляється з питанням запобігання SQL ін'єкцій.

Висновки та перспективи подальшого дослідження.

Отже, SQL-ін'єкції дуже небезпечні для незахищеної системи, так це є найпопулярніший спосіб взлому, і не є дуже складним для розуміння і використання. Тому навіть недосвідчений зловмисник може отримати доступ до бази даних. Байдуже ставлення до захисту від даних атак може призвести до катастрофічних наслідків.

Для надійного захисту використовуйте різні технології захисту, як приклад, технологію PDO, яка захистить від SQL ін'єкцій і дозволить використовувати чистий шар абстракції з бази даних, що забезпечить вам майбутню гнучкість у разі зміни баз даних.

Список бібліографічного опису

1. Addison Berry, Angela Byron, Bruno De Bondt. Using Drupal (2nd Edition). – O'Reilly. 2012. – 500p.
2. Al-Darwish, N.: Page Gen: An Effective Scheme for Ddynamic Generation of Web Pages. Information and Software Technology 45(10), 15 July 2003, Pages 651-662
3. Cynthia McCourt. Drupal: The Guide to Plainning and Building Websites. – Wrox. 2011. – 504 p.
4. Douglas Vernon Denny. Drupal 7 Webform Cookbook. – Packt Publishing. 2012. – 276 p.
5. FabienPotencier. TemplatingEnginesin PHP (переклад), Templatingenginesin PHP – Follow-Up (переклад)
6. Jennifer Hodgdon. A Programmer's Guide to Drupal. – O'Reilly. 2012. – 114p.
7. Ric Shreves, Brice Dunwoodie. Drupal 7 Bible. – Wiley. 2011. – 768 p.
8. Smarty 3.1.29 Released – 2015.
9. Trevor James. Migrating to Drupal 7. – Packt Publishing. 2012. – 158p.
10. <http://anton.shevchuk.name/php/php-template-engin/>
11. <https://drudesk.com.ua/blog/funktsionalni-mozhlyvosti-twig>
12. ZhangZhuo, The SQL injection attack technology and preventive measures research, 2-4 2007.01.
13. ZhouWenYu, Based on preventing SQL injection network security technology analysis and application, 43-50 2010.06.
14. Xiaozhu The SQL;1; injection into holes of ASP too mysterious full contact, 2005.01.

References

15. Addison Berry, Angela Byron, Bruno De Bondt. Using Drupal (2nd Edition). – O'Reilly. 2012. – 500p.
16. Al-Darwish, N.: Page Gen: An Effective Scheme for Ddynamic Generation of Web Pages. Information and Software Technology 45(10), 15 July 2003, Pages 651-662
17. Cynthia McCourt. Drupal: The Guide to Plainning and Building Websites. – Wrox. 2011. – 504 p.
18. Douglas Vernon Denny. Drupal 7 Webform Cookbook. – Packt Publishing. 2012. – 276 p.
19. FabienPotencier. TemplatingEnginesin PHP (переклад), Templatingenginesin PHP – Follow-Up (переклад)
20. Jennifer Hodgdon. A Programmer's Guide to Drupal. – O'Reilly. 2012. – 114p.
21. Ric Shreves, Brice Dunwoodie. Drupal 7 Bible. – Wiley. 2011. – 768 p.
22. Smarty 3.1.29 Released – 2015.
23. Trevor James. Migrating to Drupal 7. – Packt Publishing. 2012. – 158p.
24. <http://anton.shevchuk.name/php/php-template-engin/>
25. <https://drudesk.com.ua/blog/funktsionalni-mozhlyvosti-twig>
26. ZhangZhuo, The SQL injection attack technology and preventive measures research, 2-4 2007.01.
27. ZhouWenYu, Based on preventing SQL injection network security technology analysis and application, 43-50 2010.06.
28. Xiaozhu The SQL;1; injection into holes of ASP too mysterious full contact, 2005.01.

DOI: 10.36910/6775-2524-0560-2020-39-24

УДК: 004.4`4

Курдус Анастасія Олександрівна, студентка 4-го курсу
Національний технічний університет України "КПІ ім.І.Сікорського"

ПОРІВНЯЛЬНИЙ АНАЛІЗ ОНЛАЙН-КОМПІЛЯТОРІВ

Курдус А.О. Порівняльний аналіз онлайн-компіляторів. У статті описані різні онлайн-компілятори, приведені переваги та недоліки кожного з них. Проведений порівняльний аналіз і, згідно з результатами, онлайн-компілятори розподілені по категоріям. Прикріплені рисунки інтерфейсів онлайн-компіляторів.

Ключові слова: онлайн-компілятор, інтерфейс, комерційна розробка, програміст, мови програмування, користувач, редагування.

Курдус А.О. Сравнительный анализ онлайн-компиляторов. В статье описаны различные онлайн-компиляторы, приведены преимущества и недостатки каждого из них. Проведен сравнительный анализ и, согласно результатам, онлайн-компиляторы распределены по категориям. Прикреплены рисунки интерфейсов онлайн-компиляторов.

Ключевые слова: онлайн-компилятор, интерфейс, коммерческая разработка, программист, языки программирования, пользователь, редактирование.

Kurdus Anastasiia. Comparative analysis of online compilers. The article describes various online compilers, shows the advantages and disadvantages of each among them. A comparative analysis is carried out, and, according to the results, online compilers are categorized. Attached are the drawings of the interfaces of online compilers.

Keywords: online compiler, interface, commercial development, programmer, programming languages, user, editing.

Постановка проблеми. Онлайн-компілятори часто використовуються як в навчальній діяльності, так і в комерційній розробці, адже майже всі розробники рано чи пізно стикаються з необхідністю запустити або швидко перевірити якийсь код і розуміло, що для такого простого завдання не хочеться запускати важкі десктопні IDE або прикладні компілятори. Проте трапляється так, що в інтерфейсі користувача онлайн-компілятора досить важко розібратися або бракує потрібних функцій.

Формування мети дослідження. Метою роботи є аналіз певних онлайн-компіляторів і порівняння їхніх інтерфейсів, що дає можливість знайти прості в користуванні і багатофункціональні онлайн-компілятори.

Порівняльний аналіз онлайн-компіляторів. Аналізуватися будуть наступні онлайн-компілятори: CodePad.io, C++shell, IdeOne, CodingGround, JDoodle, TryCode. CodePad.io — цей онлайн-компілятор (рис. 1.) підтримує велику кількість мов програмування (C, C++, D, Haskell, Lua, OCaml, PHP, Perl, Plain Text, Python, Ruby, Tcl) [2]. Проте слід зауважити, що хоч вибір мов програмування і великий, але немає можливості обрати потрібну версію інтерпретатора або компілятора. Інтерфейс онлайн-компілятора досить простий і зрозумілий у використанні, потрібно просто вставити текст коду в редактор CodePad і вже можна скомпілювати і виконати його. Головна перевага цього сервісу в тому, що він дуже легкий, тобто сайт буде швидко працювати і при повільному інтернеті. Серед наявних функцій, CodePad дає можливість ділитися своїм кодом з колегами електронною поштою або в чаті. Серед мінусів можна назвати відсутність підсвічування коду, що вводиться, і це може створити певний дискомфорт для розробника.



Рис. 1. Онлайн-компілятор CodePad

C++shell — онлайн-компілятор (рис. 2.), що підтримує лише мови C та C++, тому підійде тільки для розробників на цих мовах[4]. Проте така зосередженість на певних мовах програмування дозволяє реалізувати всі можливості оригінального C та C++. Крім того, існує можливість задати велику кількість налаштувань (версію C, версію компілятора, додаткові функції і т.д.). Щодо інтерфейсу онлайн-компілятора, то він трохи застарілий.

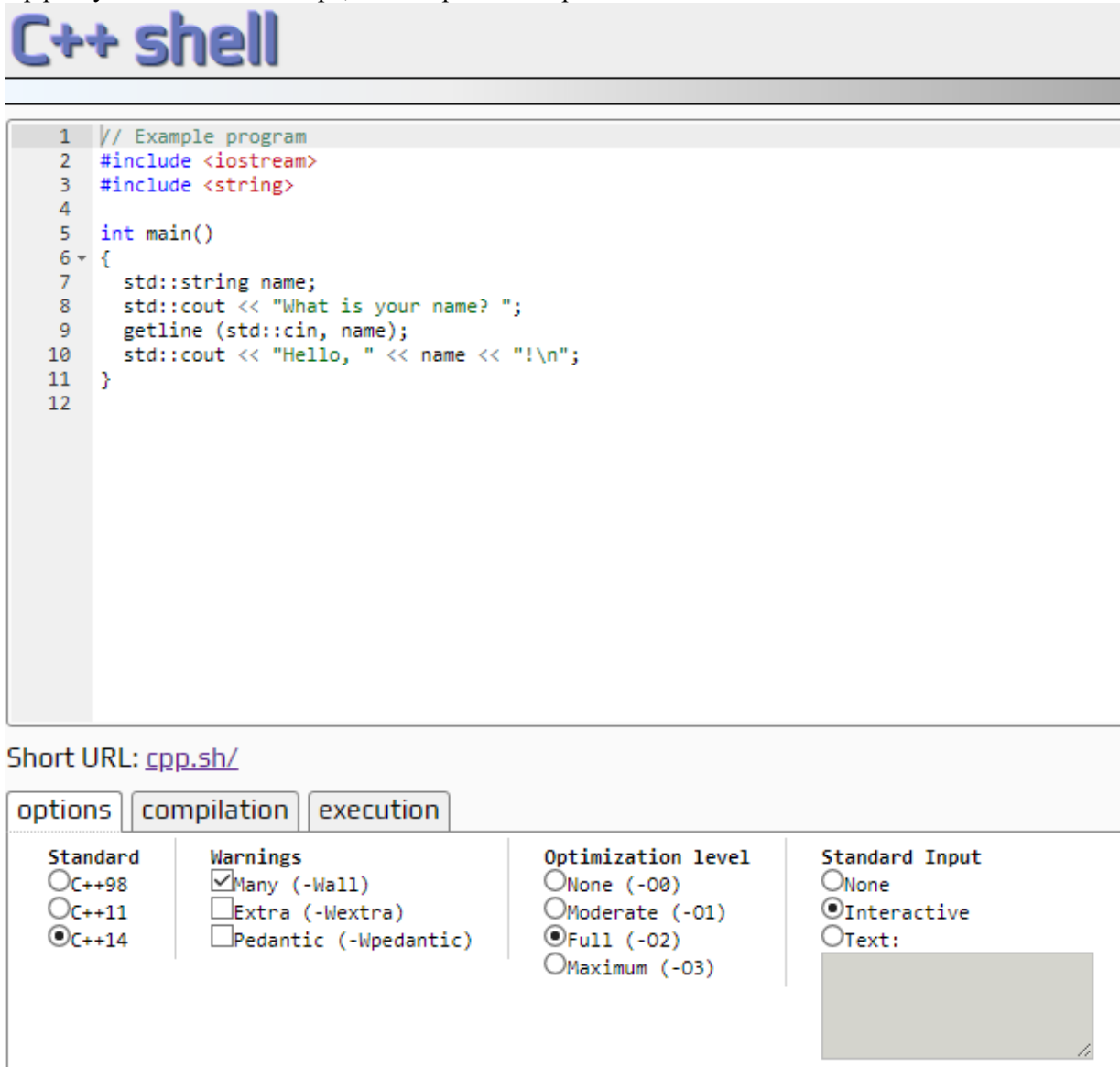


Рис. 2. Онлайн-компілятор C++shell

IdeOne — онлайн-компілятор (рис. 3.), який підтримує 60 мов програмування[2]. В мови, які підтримує цей компілятор, входять C++, D, Javascript, Coffee Script, R, Rust, Scala, Node.js, Ruby, Python, Perl, Go і багато інших. Інтерфейс дуже легкий в освоєнні та інтуїтивно зрозумілий, що є досить важливо, адже розробнику не потрібно витрачати час для того, щоб зрозуміти, як працювати з сервісом. Також надана можливість налаштувати ліміт часу, стандарт потоків вводу, а також можна додавати коментарі та мітки до коду [1]. Ще однією перевагою даного онлайн-компілятора є можливість ділитися своїм кодом з колегами, а отже працювати над одним кодом навіть на відстані, що є досить важливим і потрібним функціоналом для комерційної розробки. Та найбільшою особливістю IdeOne є те, що всі члени групи, яка працює з кодом, можуть зробити так званий fork файлу. Fork - відгалуження частини коду, з можливістю продовжити роботу над проектом, використовуючи сторонню мову програмування. При цьому вихідний код зберігається [1].



Рис. 3. Онлайн-компілятор IdeOne

JDoodle — це онлайн-компілятор (рис. 4.), який підтримує 72 мови програмування, а також 2 бази даних[3]. Досить корисною функцією онлайн-компілятора є можливість спільної роботи декількох програмістів над одним проектом. Також є підсвічування коду, що можна віднести до корисного функціоналу. Слід зазначити, що інтерфейс JDoodle простий, зрозумілий та сучасний. Проте даний онлайн-компілятор має і недолік, а саме: підтримку всього одного файлу. Тобто його не можна використовувати при вивченні концепції пакетів, інкапсуляції та подібного.

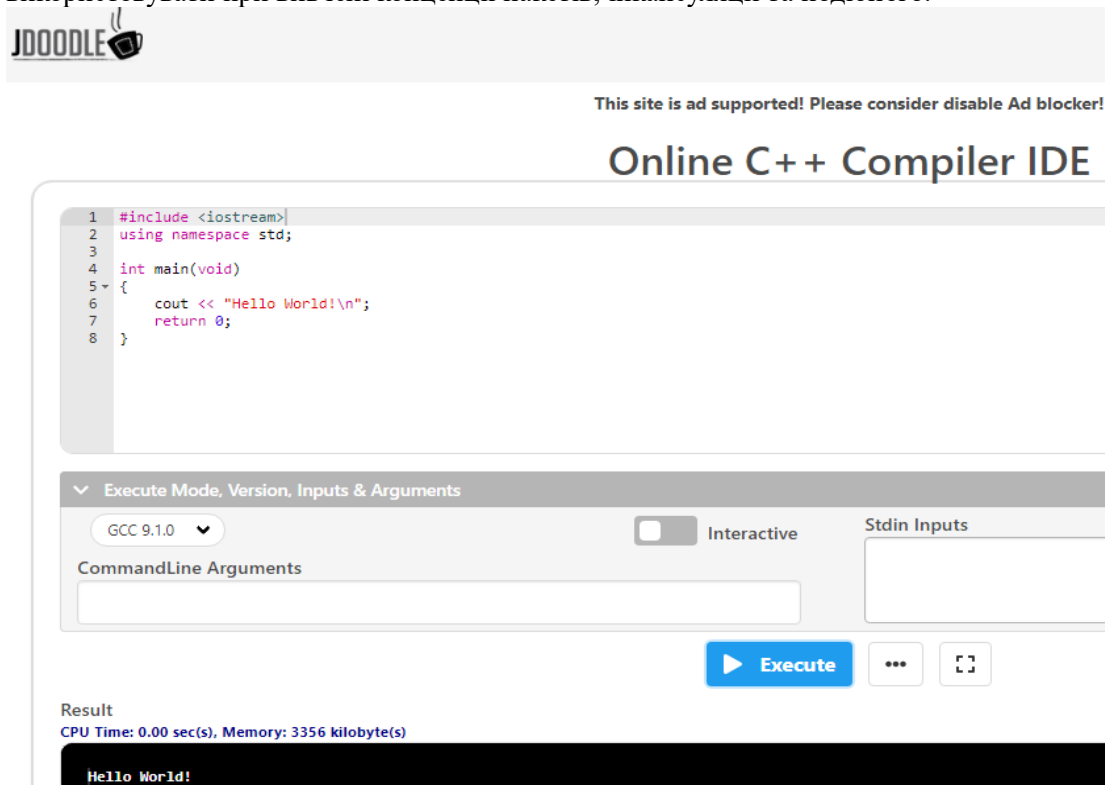


Рис. 4. Онлайн-компілятор JDoodle

CodingGround — це онлайн-компілятор (рис. 5.), який підтримує понад 75 мов програмування[5]. Інтерфейс онлайн-компілятора приємний у користуванні та досить мінімалістичний, що є перевагою, адже не змушує розробника розбиратися з великою кількістю кнопок і панелей. Також представлений функціонал для онлайн програмування та можливість ділитися кодом з колегами.

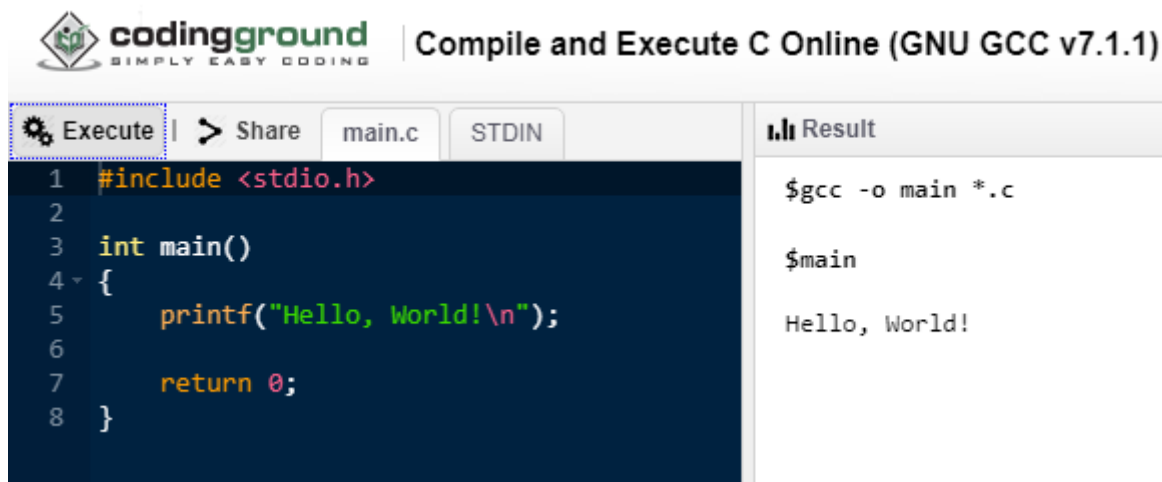


Рис. 5. Онлайн-компілятор CodingGround

TryCode — онлайн-компілятор (рис. 6.), який підійде в першу чергу для спільної роботи декількох розробників. Вбудована підтримка розробки в режимі реального часу, що є корисною функцією, як для комерційної діяльності, так і для навчання майбутніх програмістів. Проте, так як розробники TryCode більше зосередилися на ідеї колективного доступу до одного коду, то функціонал і налаштування дещо обмежені. Що стосується інтерфейсу, то він нативний і не потребує багато часу для розуміння того, як з ним працювати. Варто зазначити, що TryCode не є онлайн-компілятором для багатьох мов програмування, лише для JavaScript, а також для HTML та CSS.

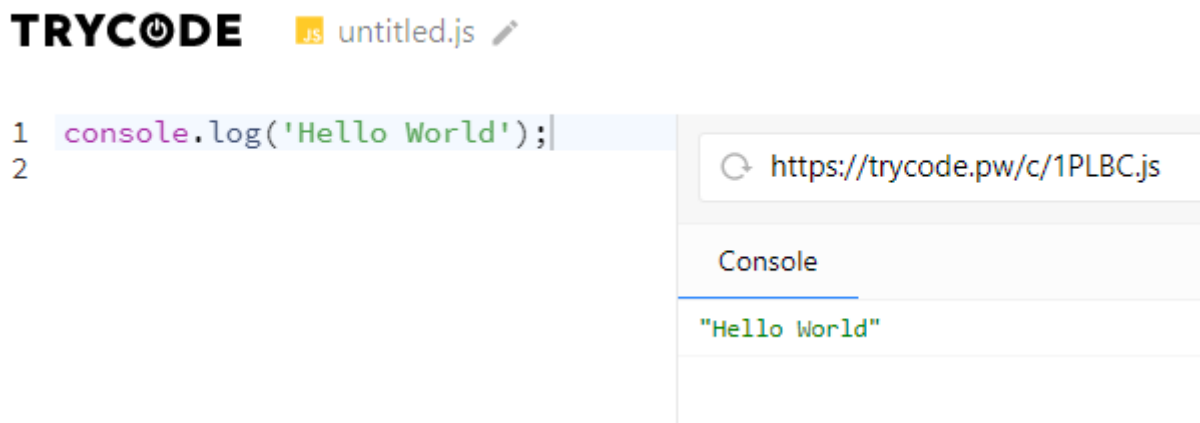


Рис. 6. Онлайн-компілятор TryCode

Загалом, якщо структурувати все вище сказане, то отримаємо Таблицю 1. В ній зазначені певні критерії за якими порівнюються онлайн-компілятори, а саме: інтерфейс та різноманітність функціоналу, які суб'єктивно оцінюються за п'ятибальною шкалою, спираючись на опис кожного онлайн-компілятора приведеного вище, а також кількість мов програмування, що підтримуються онлайн-компілятором, яка була взята з джерел опису представлених онлайн-компіляторів. Одним з пунктів оцінки онлайн-компіляторів є швидкість їхньої роботи, яка була отримана експериментальним шляхом, методом вимірювання, за допомогою спеціальних функцій, часу компіляції однієї і тієї ж програми. Такий пункт, як популярність серед користувачів – це пропорційно виведена оцінка, яка базується на статистичних даних з різних джерел інформації. Останнім пунктом таблиці є середній бал, він показує середнє значення по всім показникам для кожного онлайн-

компілятора, причому всі кількісні показники (швидкість роботи та к-ть мов програмування) були пропорційно оцінені за п'ятибальною шкалою.

Таблиця 1. Суб'єктивна оцінка онлайн-компіляторів за певними критеріями

Критерій \ Компілятори	CodePad	C++shell	IdeOne	CodingGround	JDoodle	TryCode
Інтерфейс	3	3	5	4	3	4
К-ть мов, що підтримуються	15	2	60	75+	72	2
Різноманітність функціоналу	3	4	5	5	4	3
Швидкість роботи	37	18	13	31	20	7
Популярність	3	2	4	5	4	3
<i>Середній бал</i>	3	3	4.6	4.2	4	3.2

Висновок

Якщо розглядати онлайн-компілятори по кількості мов, які підтримуються, то беззаперечним лідером є CodingGround, адже кількість мов більше ніж 75. Проте онлайн-компілятор JDoodle, який підтримує 72 мови програмування також можна назвати лідером, адже кількість мов велика і майже кожен користувач може знайти потрібну йому мову. До того ж, цей онлайн-компілятор має багато інших переваг, а саме: нативний інтерфейс, опцію, яка дозволяє ділитися кодом, велику кількість іншого функціоналу та налаштувань. Також до трійки лідерів можна віднести і IdeOne, який підтримує 60 мов програмування. Інший онлайн-компілятор, на який варто звернути увагу тим, хто є розробником на JavaScript і для кого важливим аспектом є колективна робота в режимі реально часу - це TryCode. Його розробники зосередилися саме на комфортності колективної роботи і тому TryCode дуже швидкий, але він має мало функцій та налаштувань. На одному рівні знаходяться такі онлайн-компілятори, як: CodePad.io, C++shell. В кожного є свої переваги і недоліки, але ці онлайн-компілятори складно назвати універсальними. Наприклад, C++shell чудово підійде тільки розробникам на C та C++, адже він підтримує тільки ці мови програмування, хоча і реалізує всі можливості оригінального C. CodePad має важливу перевагу - легкість, це означає, що при поганому з'єднанні з інтернетом, він буде працювати стабільно. Проте, у цього онлайн-компілятора є і недоліки, а саме: відсутність підсвічування коду, що робить розробку незручною, а також неможливість вибору потрібної версії інтерпретатора або компілятора.

Список бібліографічного опису

1. Подборка онлайн-компиляторов, <https://gitjournal.tech/podborka-onlajn-kompiljatorov-cto-jeto-kak-oni-rabotajut-i-kakoj-vybrat/#i>
2. Подборка онлайн-компиляторов: тестируем код прямо в браузере, <https://tproger.ru/digest/compile-code-online/>
3. Онлайн-средства для разработчика, <https://javarush.ru/groups/posts/1922-online-sredstva-dlja-razrabotchika>
4. C++ онлайн-компилятори, <https://arne-mertz.de/2017/05/online-compilers/>
5. CodingGround, <https://www.tutorialspoint.com/codingground.htm>

References

1. Podborca onlain-compiljatorov, <https://gitjournal.tech/podborka-onlajn-kompiljatorov-cto-jeto-kak-oni-rabotajut-i-kakoj-vybrat/#i>
2. Podborca onlain-compiljatorov, <https://tproger.ru/digest/compile-code-online/>
3. Online-sredstva dlja razrabotchika, <https://javarush.ru/groups/posts/1922-online-sredstva-dlja-razrabotchika>
4. C++ online compilers, <https://arne-mertz.de/2017/05/online-compilers/>
5. CodingGround, <https://www.tutorialspoint.com/codingground.htm>

DOI: 10.36910/6775-2524-0560-2020-39-25

УДК: 004.05(075.8)

Марценюк Василь Петрович, д.т.н., професор,

<https://orcid.org/0000-0001-5622-1038>

Університет Бельсько-Бяли, Польща¹⁾

Дідманідзе Ібраїм Шотаєвич, д.фіз.-мат.н., професор,

<https://orcid.org/0000-0001-6695-4980>

Батумський державний університет імені Шота Руставелі, Грузія²⁾

Сверстюк Андрій Степанович, к.т.н., доцент,

<https://orcid.org/0000-0001-8644-0776>

Тернопільський національний медичний університет імені І.Я. Горбачевського, Україна³⁾

Андрущак Ігор Євгенович, д.т.н., професор,

<https://orcid.org/0000-0002-8751-4420>

Рудь Катерина Іванівна, аспірант.

<https://orcid.org/0000-0002-1148-6080>

Луцький національний технічний університет, Україна⁴⁾

AUTOMATED METHOD OF BUILDING EXPLOITES IN ANALYSIS SOFTWARE TESTING.

Марценюк В.П., Дідманідзе І.Ш., Сверстюк А.С., Андрущак І.Є., Рудь К.І. Автоматизований метод побудови експлойтів в аналізі тестування програмного забезпечення. У статті розглядається проблема методу автоматизованої побудови експлойтів для уразливості переповнення буфера на стеку і його застосування до задачі оцінки критичності помилок у програмному забезпеченні і побудові захисту програмних засобів. Проаналізовано сучасні практичні і теоретичні методи вторгнення в різні операційні системи.

Ключові слова: безпека цільової системи, експлойт, інформаційна безпека, переповнення системного буфера, несанкціонований доступ, вразливість в програмному забезпеченні, класифікація помилок.

Марценюк В.П., Дідманідзе І.Ш., Сверстюк А.С., Андрущак І.Є., Рудь К.І. Автоматизированный метод построения эксплойтов в анализе тестирования программного обеспечения. В статье рассматривается проблема метода автоматизированного построения эксплойтов для уязвимости переполнения буфера на стеке и его применение к задаче оценки критичности ошибок в программном обеспечении и построении защиты программных средств. Проанализированы современные практические и теоретические методы вторжения в различные операционные системы.

Ключевые слова: безопасность целевой системы, эксплойт, информационная безопасность, переполнение системного буфера, несанкционированный доступ, уязвимость в программном обеспечении, классификация ошибок.

Martsenyuk V.P., Didmanidze I.Sh., Sverstiuk A.S., Andrushchak I.Ye., Rud K.I. Automated method of building exploits in analysis software testing. The article discusses the problem of the method of automated construction of exploits for the buffer overflow vulnerability on the stack and its application to the task of evaluating the criticality of errors in software and building software protection. The modern practical and theoretical methods of invading various operating systems are analyzed.

Keywords: target system security, exploit, information security, system buffer overflow, unauthorized access, software vulnerability, error classification.

Formulation of the problem. In modern distributed information systems, various types and nature of threats prevail associated with unauthorized access and data leakage. There are threats that are aimed at harming the personal data of the user through their damage or copying for their personal benefit for the purposes of using directly against the user himself. The standard definition describes an "exploit" as a program or code that exploits security flaws in a specific application to infect a device. Users may mistakenly consider this to be separate malware. However, in fact, this is a piece of program code that allows you to penetrate the computer system and affect its operation. Using a specific vulnerability, this tool provides attackers with the necessary permissions to launch malicious components and infect the system.

Security solution developers often mention exploits in their publications as one of the most serious data and system security issues, although it is not always clear what the difference is between exploits and malware in general.

At the development stage, all programs and networks incorporate protection mechanisms against hackers, such as locks, preventing unauthorized attacks from outside. The vulnerability is similar to an open window, to get through which is not difficult for an attacker. In the case of a computer or network, cybercriminals can install malicious software by exploiting a vulnerability in order to gain control or infect the system for their own personal gain with corresponding consequences. Most often, all this happens without the knowledge of the user.

Analysis of research. Attackers are constantly improving their tools and finding new ways to infect a large number of devices. One of the common methods for malware to infect victims' computers has been the use of exploits that ensure the rapid spread of threats. They also allow you to access programs and

subsequently infect the user's device through vulnerability in the security system. In recent years, the most active have been threats that exploit vulnerabilities in Java products, in Adobe software, and in the Windows operating system.

Recently, exploits have been used in many well-known cyber attacks. An example is the massive attack of the WannaCryptor virus (or WannaCry), which has become the largest digital threat in the world in recent years. It is worth noting that during this attack, the EternalBlue exploit was used, which was allegedly stolen by a group of cybercriminals at the National Security Agency (NSA). EternalBlue was aimed at the vulnerability of the implementation of the SMB protocol in an irrelevant version of Microsoft. In addition, EternalBlue was also a tool during the famous Diskcoder.C attack (Petya, NotPetya and ExPetya).

Browsers along with Flash, Java, and Microsoft Office are among the most attacked software categories. Because of their ubiquity, both security experts and hackers are actively exploring them, and browser developers are forced to regularly issue patches to fix vulnerabilities. It is best to install these patches at once, but, unfortunately, this does not always happen - after all, you will have to close all the tabs.

A special problem, of course, is the exploits of unknown vulnerabilities discovered and used by criminals - the so-called zero-day vulnerabilities. It may take a long time before manufacturers find out about the problem and fix it.

Presentation of the main material and the justification of the results. Depending on the method of gaining access to the vulnerable software, exploits are divided into remote and local. Remote exploit works through the network and exploits a security vulnerability without any prior access to the vulnerable system; a local exploit runs directly on the vulnerable system, requiring prior access to it. Usually used to gain administrator / superuser privileges.

Exploits are caused by errors in the software development process, which result in vulnerabilities in the program protection system that are successfully used by cybercriminals to gain unlimited access to the program itself, and through it further to the entire computer. Exploits are classified according to the type of vulnerability used by the hacker: zero day, DoS, spoofing, or XXS. Of course, program developers will soon release security updates in order to eliminate the defects found, but until this moment the program is still vulnerable to attackers.

Exploits are a subtype of malware. The term is associated with the English verb "to exploit", meaning "exploit, use in their own interests." However, an exploit is not necessarily a separate application (executable file): it can take the form of a small piece of malicious code or a set of commands executed in a certain order. Using the vulnerability in any system or application program, the exploit performs an unauthorized action on the victim's device. As a rule, it allows you to increase privileges on the target system or execute arbitrary code.

The objectives of the exploit are varied: downloading and installing malware, increasing access rights, stopping the system, and disclosing confidential data. It should be noted that the exploits themselves do not perform destructive actions directly: their task is to take advantage of the vulnerability to ensure that the code embedded in them is launched. All subsequent operations are carried out by the malicious load, and what the attacker will achieve depends on its content [1].

Since exploits exploit flaws in program security mechanisms, the average user has virtually no chance of determining their presence. That is why it is extremely important to keep installed programs up-to-date, especially in a timely manner, to install security updates issued by program developers. If the software developer releases a security update to eliminate a known vulnerability in his software, but the user does not install it, then, unfortunately, the program will not receive the latest virus definitions.

Exploit - a computer program, a piece of program code or a sequence of commands that exploit vulnerabilities in software and are used to carry out an attack on a computer system. The purpose of the attack can be both the seizure of control over the system (privilege escalation) and the disruption of its functioning (DoS attack). Exploits are actually designed to perform third-party actions on a vulnerable system and can be distributed in the form of source code, executable files, or a verbal description of the vulnerability exploitation.

After the vulnerability has been closed by the manufacturer, the chance of a successful exploit is rapidly decreasing. Therefore, the so-called 0day exploits that use recently appeared vulnerabilities that have not yet become generally known are especially popular among hackers [2].

The next part is quite technical, so feel free to skip, unless you are really interested in how it works. Keep in mind that cybercriminals often prefer exploits to other methods of infection, because, unlike social engineering, in which everything is done at random, exploiting vulnerabilities invariably gives the desired result.

There are two ways to feed exploits to users. Firstly, when they visit a site containing malicious exploit code. Secondly, when a user opens a file with a harmless-looking file with hidden malicious code. As you might guess, in the second case, spam or phishing emails are usually used to deliver an exploit.

Exploits take advantage of software vulnerabilities. Vulnerability is a hole in your software that can be used by malware to migrate to your device. Malicious programs exploit these vulnerabilities to prevent security threats from infecting your computer.

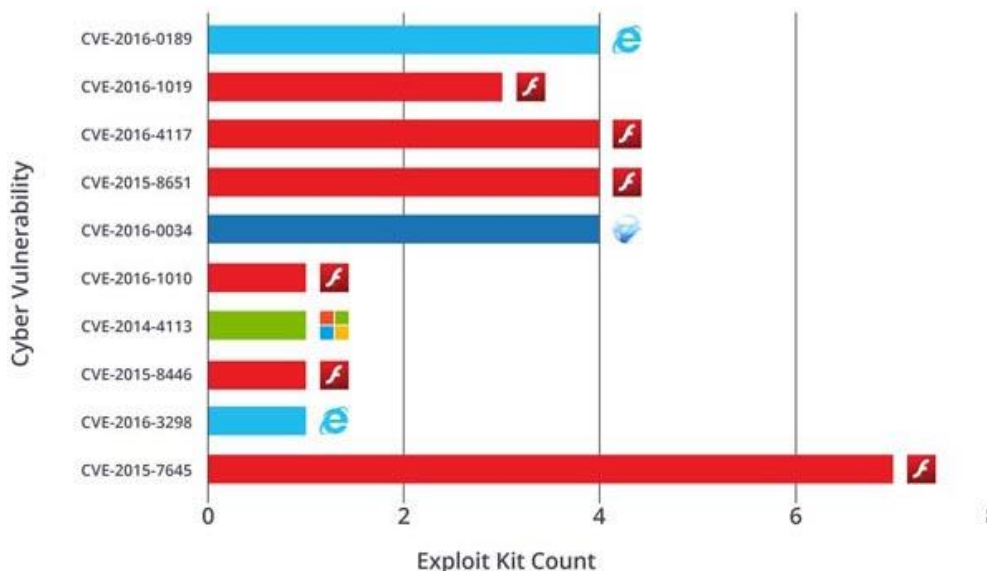
Most often, the use of exploits is the first part of a wider attack. Hackers scan outdated systems that contain important vulnerabilities that they then use to deploy targeted malware. Frequently used exploits include what is called shellcode. This is small useful malware data that is used to download additional malware from networks controlled by cybercriminals. This allows hackers to infect devices and infiltrate organizations.

Exploit packages are more complete tools that contain a set of exploits. These kits are designed to scan devices for vulnerabilities of various types of software and, if detected, deploy additional malware to further infect the device. Kits can use exploits designed for various software products, including Adobe Flash Player, Adobe Reader, Internet Explorer, Oracle Java and Sun Java.

The most common way to distribute exploits and exploit kits is through web pages, but exploits can also be emailed. Some websites are unknown and inadvertently contain malicious code and exploits in their advertising materials. The infographic below shows how a set of exploits can try to use the device when visiting a compromised web page.

Exploits are designed to attack specific versions of software containing vulnerabilities. Thus, if the user has the right version of the software when opening a malicious object, or if the website uses this software to work, the exploit is launched. Once it is accessed through a vulnerability, the exploit downloads additional malware from the server of criminals that carry out subversive activities, such as identity theft, using a computer as a botnet element to send spam, or perform DDoS attacks, and so on. Exploits pose a threat even to cautious and conscientious users who regularly update their software. The reason lies in the temporary gap between the discovery of the vulnerability and the release of the patch to fix it [3].

In this interval, exploits are free to operate and threaten the security of almost all Internet users in the absence of automatic means to prevent exploit attacks installed in the system. Again, let's not forget about the open tabs syndrome - timely updating of programs often requires some sacrifices from the user, which not everyone is ready to go right at the time the patch is released (Pic 1).



Pic 1. Vulnerability adoption by Exploit Kit

Depending on the method of gaining access to the vulnerable software, exploits are divided into remote (local) and local (local):

- remote exploit works through the network and exploits a security vulnerability without any prior access to the vulnerable system;
- a local exploit runs directly on the vulnerable system, requiring prior access to it. Usually used to gain root privileges by an attacker.

An exploit attack can target various components of a computing system - server applications, client applications, or operating system modules. To exploit the server vulnerability, it's enough to form an exploit and send a request to the server containing malicious code. Using a client's vulnerability is a bit more complicated - you need to convince the user of the need to connect to a fake server (click the link if the vulnerable client is a browser) [4].

Exploits are actually designed to perform third-party actions on a vulnerable system and can be divided among themselves as follows:

- exploits for operating systems
- exploits for application software (music players, office suites)
- exploits for browsers (Internet Explorer, Mozilla Firefox, Opera)
- exploits for online products (IPB, WordPress, VBulletin, phpBB)
- exploits for internet sites (facebook.com, hi5.com, livejournal.com)
- other exploits

The exploit can be distributed in the form of source codes, executable modules, or a verbal description of the exploitation of the vulnerability. It can be written in any programming language (most commonly used: C / C ++, Perl, Python, PHP, HTML + JavaScript)

Exploits can also be classified by the type of vulnerability they use, such as: buffer overflows, SQL code injection, crossite scripting, crossite request forgery

Information obtained as a result of vulnerability detection can be used both to write an exploit and to eliminate the vulnerability. Therefore, both parties are equally interested in it - both the cracker and the hacked software manufacturer. The nature of the distribution of this information determines the time it takes the developer to release the patch. Bundles of exploits are a package of exploits for several programs (versions) at once and / or for different vulnerabilities in them. In the latest versions of bundles, an exploit is selected specifically for a specific user program. In most cases, exploit kits are used for attacks that use browser vulnerabilities or their add-ons (common targets, for example, are Java, Flash and PDF). There are also sets of local exploits for elevating privileges in an attacked system. In fact, such sets are also bundles, but in a hacker environment they are not considered and are not called.

Exploits are created by highly qualified cybercriminals who sell them on the black market to other attackers. As elements of cyber weapons, they are developed and used by special services. Also, exploits may be the result of the work of information security specialists who want to show how the vulnerability they discovered can be exploited. In this case, software vendors are notified of vulnerabilities before the exploit is published in the public domain [5].

Exploits are often packaged together - so as to test the target system for a wide range of vulnerabilities. Once one or more is discovered, the appropriate exploits come into play. Exploit kits also make extensive use of special code obfuscation techniques (experts call it the clever word "obfuscation") to avoid finding and replacing Internet addresses to prevent researchers from calculating them. Exploits can be applied to any elements of a computer system. The target of an attack can be operating system modules, application programs, and even hardware components. For a successful attack, it is necessary to force the victim to follow the link, download and open the file so that the exploit is able to exploit the desired vulnerability.

We list a few of the most famous exploit kits, or, as they are also called, exploit kits [6-7].

Angler is one of the most sophisticated kits on the black market. This set of exploits, by its appearance, changed the rules of the game after it began to detect antiviruses and virtual machines (often used by security experts as decoys) and use encrypted files to make research difficult. This is one of those exploit kits that most quickly includes the newly discovered zero-day vulnerabilities in its arsenal, and its malicious programs work in memory, without writing to the victims' hard drives.

Nuclear Pack - Affects victims with Java and Adobe PDF exploits, and also spoils Caphaw, the notorious banking trojan.

Neutrino is a suite from Russian-language developers containing several Java exploits. Neutrino became famous last year due to the fact that the owner put it up for sale at a very modest price - \$ 34 thousand. Most likely, this was done after the arrest of a certain Paunch, the creator of the next set, which we want to talk about.

The Blackhole Kit is the most common web threat in 2012, targeting vulnerabilities in older versions of Firefox, Chrome, Internet Explorer, and Safari, as well as many popular plugins such as Adobe Flash, Adobe Acrobat, and Java. After the victim has been lured or redirected to the landing page, the confusing JavaScript detects the contents of the victim's machine and loads the exploits for which this computer is vulnerable [8].

Despite the efforts of specialists, there are vulnerabilities in almost all programs, which means that there is always a loophole for attackers to prepare an exploit. By the time developers release a patch (a small program to fix vulnerable files), the malware can do massive damage. All users are at risk, including the most cautious and attentive. The consequences of using an exploit can be very different. It depends on the task that malware posed: from disrupting the system to losing large sums of money, secret information. You can secure your device from exploit if you use anti-virus programs from well-known companies that are constantly improving their products. Regular updating of the operating system and application programs, refusal to click on suspicious links, ignoring spam messages, and careful attention to financial transactions will reduce the risk of infection [9].

Conclusion and prospects for further research

As stated above, exploits are a subtype of malware, but they are not detected by all security programs. Successful detection requires a defensive solution to use behavioral analysis - this is the only reliable method to combat exploits. Malicious programs can be numerous and varied, but most of them have similar behaviors.

Due to the fact that exploits are the result of perfect flaws, their elimination is the direct responsibility of the developers, so it is the authors who will have to prepare and send the bug fix. Nevertheless, the obligation to keep the installed programs updated and to install update packages in a timely manner so as not to give hackers a chance to take advantage of vulnerabilities lies entirely with the user of the program. One of the possible ways not to miss the latest updates is to use an application manager that will ensure that all installed programs are updated, or - even better - use the automatic update search and installation tool.

Rely on common sense and follow the basic rules for safe browsing on the Internet. Hackers can only exploit the vulnerability if they can gain access to your PC. Do not open attachments in suspicious messages or download files from unknown sources. Keep installed programs up-to-date and install security updates in a timely manner. If you want to simplify this task as much as possible, download Avast antivirus, which will not only provide reliable protection against all types of malware, but also help with installing the latest updates for third-party programs.

References

1. Andrianov V.I., Andronov A.V. Intelligent means of ensuring information security of automated systems in conditions of uncertainty // Journal of scientific publications of graduate students and doctoral students. 2010.No 8 (50). S. 120-121.
2. Vakhrushev I. A. et al. Method for searching for format string vulnerability // Transactions of the Institute for System Programming of the Russian Academy of Sciences, vol. 27, no. 4, 2015, pp. 23-38. DOI: 10.15514 / ISPRAS-2015-27 (4) -2.
3. Buynevich M.V., Izrailov K.E. Utility for searching for vulnerabilities in software of telecommunication devices using machine code algorithmization. Part 1. Functional architecture // Information Technologies and Communications. 2016.V. 4.No 1.P. 115-130.
4. Krasov A.V., Shterenberg S.I., Fakhrutdinov R. M., Ryzhakov D. V., Pestov I. E. Analysis of enterprise information security based on user data collection from open resources and monitoring of information resources using a machine training // T-comm: Telecommunications and transport. 2018.V. 12.No 10.P. 36-40.
5. Padaryan V.A., Kaushan V.V., Fedotov A.N. Automated method for constructing exploits for the buffer overflow vulnerability on the stack. Proceedings of ISP RAS, vol. 26, no. 3, pp. 127-144. DOI: 10.15514 / ISPRAS-2014-26 (3) -7.
6. Heelan S. Automatic generation of control flow hijacking exploits for software vulnerabilities. Master's thesis, University of Oxford, 2009.
7. Huang S.K. et al. Crax: Software crash analysis for automatic exploit generation by modeling attacks as symbolic continuations Software Security and Reliability (SERE), 2012 IEEE Sixth International Conference on. IEEE, 2012, pp. 78-87.
8. Fedotov A.N. A method for evaluating the exploitability of software defects. Proceedings of ISP RAS, vol. 28, no. 4, 2016, pp. 137-148. DOI: 10.15514 / ISPRAS- 2016-28 (4) -8.
9. Shterenberg S.I., Andrianov V.I. Investigation of adaptive attack techniques based on hidden attachment in executable files // Collection of articles of the International scientific and technical conference "Science, Technology, Innovations" (Bryansk, March 25-27, 2014) Bryansk: Reliable cars, 2014. S. 287-294.

DOI: 10.36910/6775-2524-0560-2020-39-26

УДК: 004.624

Мельник Катерина Вікторівна, к.т.н., доцент,

<https://orcid.org/0000-0002-9991-582X>

Мельник Василь Михайлович, к.фіз.-мат.н., доцент,

<https://orcid.org/0000-0001-8282-6639>

Григоришин Андрій Миколайович, студент.

Луцький національний технічний університет, м. Луцьк, Україна.

АВТОМАТИЧНИЙ ЗБІР ІНФОРМАЦІЇ (ПАРСИНГ) В МЕРЕЖІ

Мельник К. В., Мельник В.М., Григоришин А. М. Автоматичний збір інформації (парсинг) в мережі. В даній роботі проаналізовано основні підходи до автоматизованого збору інформації в мережі інтернет, приведено характеристику контексту при зборі інформації. Розроблена система автоматичного збору інформації (парсингу) в мережі на мові програмування Golang.

Ключові слова: Парсинг, пошук інформації, Golang, автоматичний збір даних.

Мельник Е.В., Мельник В.М., Григоришин А. Н. Автоматический сбор информации (парсинг) в сети. В данной работе проанализированы основные подходы к автоматизированному сбору информации в сети интернет, приведена характеристика контекста при сборе информации. Разработана система автоматического сбора информации (парсинга) в сети на языке программирования Golang.

Ключевые слова: Парсинг, поиск информации, Golang, автоматический сбор данных.

Melnyk K., Melnyk V.M., Hryhorushyn A. Automatic collection of information (parsing) in the network. This paper analyzes the main approaches to automated information collection on the Internet, and provides a description of the context for collecting information. A system for automatically collecting information (parsing) on the network in the Golang programming language.

Keywords: Parsing, information search, Golang, automatic data collection.

1. Постановка проблеми. Зміст поняття «пошук інформації» в мережі Інтернет в сучасній науковій літературі, в основному, зводиться до вивчення алгоритмів роботи пошукових систем. В сучасних умовах, всі дані, що представлені в глобальній мережі Інтернет, можна назвати неструктурованими. В основному, такі дані – це HTML сторінки, в яких зберігається текстова інформація, а також посилання на певні файли, що зберігаються на сервері. Через швидке зростання кількості інформації в мережі інтернет, виникла нагальна потреба розвивати технології, які дозволять збирати і обробляти інформацію швидко, ціленаправлено і головне – автоматично. Розробка системи автоматичного збору інформації (парсингу) в мережі інтернет дозволить значно прискорити цей процес.[2, 3, 9]

Отже, парсер контенту – це не що інше, як скрипт, здатний сортувати інформацію, виділяючи найважливішу і обробляючи її згідно з алгоритмом, створеному для вирішення того чи іншого завдання. [1]

2. Аналіз останніх досліджень і публікацій. Пошук та аналіз текстової інформації досліджували видатні вітчизняні та зарубіжні вчені: Ньюелл А., Люгер Д. Ф., Фостер Д. М., Анісімов О. В., Поспелов Д. О., Попов Є. В., Широков В. О.

Крім того, проблемі розробки системи автоматичного збору інформації (парсингу) в мережі приділяли увагу вітчизняні та зарубіжні вчені, зокрема: Цхошвілі Д. З., Іванова Н. А., Суханов, А. А., Маратканов, А. С., Менщиков А. А., Гатчин Ю. А., Каргін Б. Б., Логутова Т. Г., Єгорченкова Н. Ю., Єгорченков О. В., Чорна Н. О.

Парсинг сайтів є ефективним рішенням для автоматизації збору і зміни інформації. У порівнянні з людиною, комп'ютерна програма-парсер: швидко обійде тисячі веб-сторінок; акуратно відокремить технічну інформацію від «людської»; безпомилково відбере потрібне і відкине зайве; ефективно упакує кінцеві дані в необхідному вигляді.

3. Виділення невирішених раніше частин загальної проблеми. Виходячи з вищесказаного актуальним завданням є розробка методів персоналізації результатів роботи пошукових систем шляхом надання користувачу інструментів управління пошуковою видачею, а також використання нових моделей ранжування, заснованих на суб'єктивних для кожного користувача параметрах. Розробка програмного забезпечення (ПЗ), яке вирішує проблеми пошуку та автоматизованої обробки інформації, нині дуже потрібна й актуальна. Великі обсяги інформації змушують користувачів таких сайтів витратити все більше і більше часу на її вивчення.[4, 6, 7]

4. Формулювання мети дослідження. Мета дослідження полягає проведенні аналізу існуючих методів автоматичного збору інформації та розробка власної системи автоматичного збору інформації (парсингу) в мережі.

5. Виклад основного матеріалу дослідження. Для збору інформації в мережі інтернет використовують спеціальні програми, які ще називаються веб-роботами або парсерами (краулерами). Їх можна умовно розділити на дві категорії: роботи, використовувані для законних цілей (аналіз контенту сайту, індексування для покращення роботи пошукових систем або створення «дзеркал» веб-сайтів) і програми, що використовуються зловмисниками для незаконних дій з порушенням авторських прав. Веб-роботи можуть не тільки збирати і обробляти інформацію, але і виконувати активні дії на веб-ресурсі, такі як купівля товарів і послуг, написання рекламних текстів, розсилка спаму. Крім того, робота веб-роботів призводить до збільшення навантаження на сервер і зменшення пропускної здатності, а також проблем доступу до ресурсу у звичайних користувачів.[9, 10]

В даний час в мережі інтернет доступні два типи парсерів: послуга парсингу ресурсів, що цікавляться такими програмами та використання універсальних програм-парсерів. У першому випадку замовник звертається із запитом на збір необхідних даних. В результаті він отримує шукану інформацію в необхідному вигляді і форматі [8]. Перевагами цього способу є: можливість отримати до 10000 записів; парсинг може проводитися будь-якого веб-ресурсу; немає необхідності проходити реєстрацію на сайті і використовувати VPN; надання інформації в зручному форматі. Недоліками даного способу є: висока вартість наданої послуги парсингу сайтів (ціна може варіюватися від 100\$ за 10000 записів); необхідно тривалий час на збір даних (від 1 до 3 днів); додаткові витрати за необхідність доповнення або оновлення даних (від 50\$).

Отже, парсер контенту – це не що інше, як скрипт, що здатний сортувати інформацію, виділяючи найважливішу і обробляючи її згідно з алгоритмом, створеному для вирішення того чи іншого завдання. Проаналізуємо, для чого можна використовувати парсер контенту.

По-перше, як відомо, найкращі сайти – це ті Інтернет-ресурси, на яких є цікава актуальна інформація. Нікому не потрібні вчорашні новини і сенсації, наприклад. Також, коли мова йде про сайти-обмінники валют, де необхідно змінювати інформацію про курс валют часом по кілька разів на день, створення парсера вкрай необхідно. Скрипт в таких випадках буде виконувати всю роботу сам, цілодобово відстежуючи зміни курсу валют в Нацбанку.

По-друге, парсер необхідний для автоматичного оновлення вашого Інтернет-ресурсу. Користувачі, які зайшли на сторінку один раз і виявили там застарілу інформацію, більше ніколи не повернуться на неї. Саме тому для збереження постійних користувачів, а також для залучення нових необхідне регулярне оновлення інформації на вашій інтернет-сторінці.

По-третє, парсер контенту – ідеальний інструмент для миттєвого наповнення сайту корисними даними. Так, коли в мережі величезна кількість сайтів різної спрямованості, лише мала їх частина виявляється в кінцевому підсумку корисними користувачам. Саме тому, важливо, щоб інформація на вашій веб-сторінці була не тільки актуальною, але ще і корисною.

По-четверте, – централізація даних. Відомо, що інформації в мережі предостатньо, при цьому самої різної. Вся проблема в тому, що вся ця інформація розкидана по безлічі Інтернет-ресурсах і зібрати її не так просто. Використовуючи парсер контенту, можна об'єднати вичерпну кількість корисної інформації на сайті, тим самим залучаючи все більше відвідувачів. Такий хід – відмінний варіант для далекоглядних розробників, які піклуються про те, щоб навіть випадкові відвідувачі, ставали постійними.

В процесі парсинга застосовуються скриптові мови програмування: PHP, Perl, Ruby, Python, JavaScript і багато інших. Лідерами ринку професійних інформаційно-пошукових систем є: система інтернет-моніторингу та конкурентної розвідки *Avalanche*, система пошуку інформації в Інтернеті *FileForFiles & SiteSputnik* і система автоматичного моніторингу веб-сайтів *WebSite Watcher*, система інтелектуального аналізу тексту «аналітичний кур'єр», систему аналізу текстової інформації *RCO Fact Extractor SDK*, систему аналізу текстових і структурованих даних *PolyAnalyst*.

Головні етапи пошуку інформації в мережі інтернет містять:

1. Аналіз процесу вилучення адрес. Алгоритм вилучення перебирає кілька варіантів і оцінює найкращий на основі найпростіших вагових характеристик і за допомогою експерта. Аналізуючи текст, програма має певний набір шаблонів, що виділяють певні блоки тексту, що містять якірні слова. Такий текст витягується із загального контексту з запасом в кілька слів спочатку і після кінця виразу.

2. Етап очищення тексту включає видаляються з тексту нерелевантних спеціальних символів, дужок, елементів вирівнювання або HTML розмітки. Також відбувається модифікація тексту з

урахуванням морфології, далі фраза розбивається на окремі слова і вирази та слова переводиться в початкову форму, яка придатна для подальшого пошуку в словнику.

3. Етап розбиття на слова. Текст після попереднього очищення розбивається на слова (токени) відповідно до регулярного виразу [! ,? ; / S] +. Додатково фільтруються зайві токени з чорного списку. Це допомагає максимізувати швидкість роботи системи за рахунок перебору тільки найбільш релевантних варіантів. Кожному токену зіставляється тип (числовий, текстовий, стрічковий).

В нашому дослідженні ми створимо та охарактеризуємо парсер на мові Golang.

Golang, або Go – мова програмування, початок якого був покладений в 2007 році співробітниками компанії Google: Робертом Гризмером, Робом Пайком і Кеном Томпсоном. Це швидка, статично типізована компільована мова, при використанні якої створюється враження використання динамічно типізованої та інтерпретуємої мови. Розмір програм на Go розпочинається з 3 рядків і може досягати декількох мільйонів, записаних в один або декілька файлів з розширенням .go. Сучасні текстові редактори, наприклад, vim, підтримують його синтаксис. [5]

Go підтримує типобезпеку та можливість динамічного Введення даних, а також містить багату стандартну бібліотеку функцій і вбудовані типи даних на зразок масивів з динамічним розміром і асоціативних масивів. За допомогою механізмів багатопоточності Go спрощує розподіл обчислень і мережевої взаємодії, а сучасні типи даних відкривають програмісту світ гнучкого і модульного коду. Програма швидко компілюється, при цьому є збирач сміття і підтримується рефлексія.

Опишемо порядок налаштування оточення. Спершу викачаємо 64-бітову версію Go з офіційного сайту. Залежно від використовуваної операційної системи виконуємо наступні дії.

UNIX / Linux / MacOS X / FreeBSD

Витягаємо викачаний архів в теку /usr/local/go. Наприклад:

```
tar -C /usr/local -xzf go1.8.3.linux-amd64.tar.gz
```

Додаємо теку /usr/local/go/bin в змінну оточення PATH:

```
export PATH=$PATH:/usr/local/go/bin
```

Кожна програма на мові Go складається з пакетів (packages). Пакет main – головний, з нього розпочинається виконання програми.

Розглянемо головні етапи парсингу на Go:

1. Отримання початкового коду веб-сторінки – скачати програмний код тієї сторінки сайту, з якої необхідно витягти інформацію. У різних мовах для цього передбачені різні способи. В PHP найчастіше використовують бібліотеку cURL або ж вбудовану функцію file_get_contents. В Golang для цього підійде бібліотека net/http та функція get. При перегляді коду елемента на потрібній сторінці ми побачимо наступне (рис. 1).

```
<!doctype html>
<html class="client-js ve-available" lang="uk" dir="ltr">
  <head>...</head>
  <body class="mediawiki ltr sitedir-ltr mw-hide-empty-elt">
    <div id="mw-page-base" class="noprint"></div>
    <div id="mw-head-base" class="noprint"></div>
    <div id="content" class="mw-body" role="main">
      <a id="top"></a>
      <div id="siteNotice" class="mw-body-content">...</div>
      <div class="mw-indicators mw-body-content">
      </div>
      <h1 id="firstHeading" class="firstHeading" lang="uk">
        "Кобзар"
      <span class="mw-editsection">...</span>
    </h1>
```

Рис. 1. Скріншот коду елемента Інтернет-сторінки

2. Витяг з html-коду необхідних даних. Отримавши сторінку, необхідно обробити її – виокремити із сторінки саме ту інформацію, заради якої і відбувається весь цей процес. Можна, звичайно ж, використовувати для цього регулярні вирази, проте є більш зручний шлях –

спеціалізовані бібліотеки. В Golang з цим справиться бібліотека gokogiri. Розглянемо дану бібліотеку на прикладі роботи розробленого парсера:

```
package main
import (
    "net/http"
    "io/ioutil"
    "github.com/moovweb/gokogiri"
)
func main() {
    resp, _ := http.Get("http://www.google.com")
    page, _ := ioutil.ReadAll(resp.Body)
    doc, _ := gokogiri.ParseHtml(page)
    doc.Free()
}
```

Приклад реалізації парсера, можна знайти за посиланням: <https://ahryhory-parser-go-example.online>.

Скористатися цим сервісом досить легко. У верхній частині вікна є поле для введення тексту, в яке можна скопіювати посилання на сторінку з потрібною нам статтею (рис. 2).

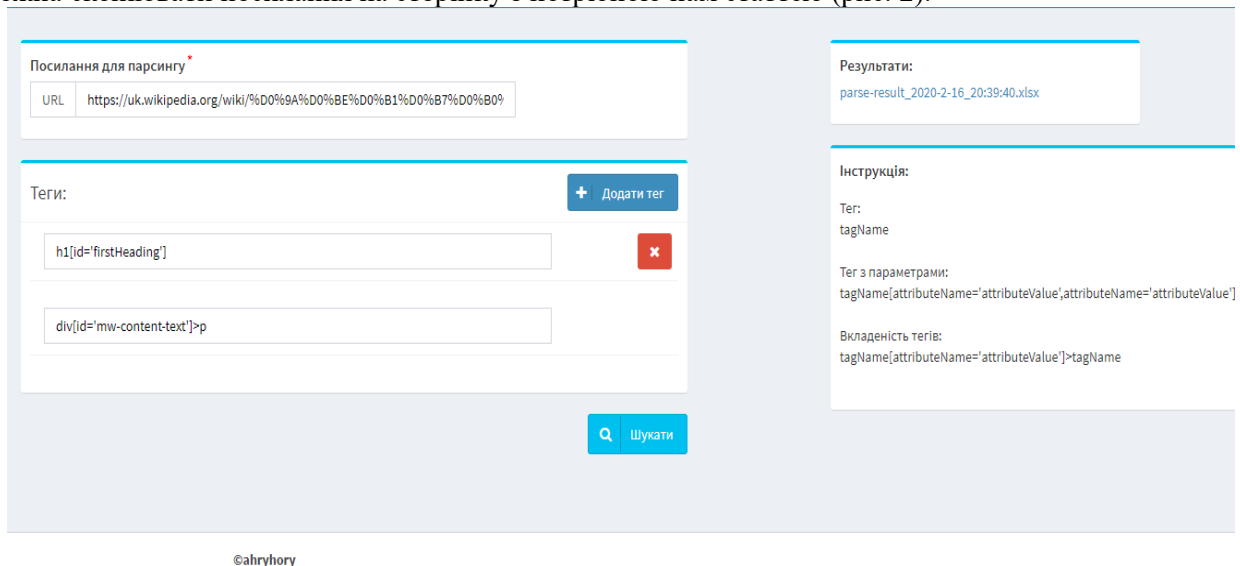


Рис. 2. Інтерфейс прикладу реалізації розробленого парсера

У блоці «Теги» (рис. 2) потрібно вписати вирази за якими буде здійснено пошук за заданим посиланням. Проведемо аналіз основних моментів цих виразів на основі прикладу, `tagName[attributeName='attributeValue']>tagName`:

- 1) «tagName» – назва потрібного тегу;
- 2) «attributeName» – назва атрибута в шуканому теґові;
- 3) «attributeValue» – значення атрибута в шуканому теґові;
- 4) «>» – означає вкладеність, тобто наступний тег має бути всередині попередньо зазначеного;

3. Фіксація результату. Успішно оброблені дані на сторінці потрібно зберегти в необхідному вигляді для подальшої обробки. Спарсені дані зазвичай заносяться в базу даних, однак є й інші варіанти. Іноді потрібно записати в файл або будувати ієрархічні JSON-структури, іноді конвертувати в excel-таблицю або згенерувати динамічний rss-потік. Якщо пошукові параметри парсера (<https://ahryhory-parser-go-example.online>) були задані правильно, то у правій частині вікна (рис. 2) з'явиться доступний до скачування файл (.xlsx) з результатами (рис. 3).

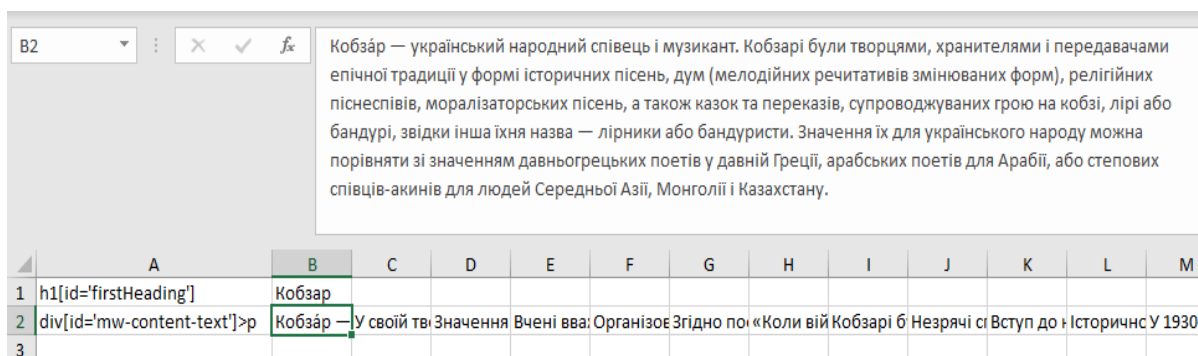


Рис. 3. Результати роботи розробленого парсера

6. Висновки. Нами проаналізовано основні підходи до автоматизованого збору інформації в мережі інтернет. Приведено характеристику контексту при зборі інформації і сказано, що отримання нової інформації безпосередньо з первинних джерел відбувається набагато рідше, тільки у випадках, коли пошук у відкритих джерелах виявляється марний або є непридатним від самого початку.

Перелічено основні етапи пошуку інформації в мережі інтернет та дано їх коротку характеристику. Окремо проаналізовано методи автоматизованого пошуку інформації. Для кожної програми наведено переваги та недоліки.

Наукова новизна дослідження полягає в розробці системи автоматичного збору інформації (парсингу) в мережі. Наголошено на основних інструментах та етапах для написання парсера та дана коротка характеристика мови програмування Golang. Перелічено існуючі проблеми при проведенні пошуку інформації сучасних умовах та перелічено основні методи пошуку.

Перспективи подальших досліджень. В перспективі подальших досліджень є розробка графічного інтерфейсу для сервера. Розроблена система надає користувачу ефективні, зручні та прості у використанні інструменти обробки веб-даних на всіх етапах взаємодії з семантичною павутиною від створення веб-ресурсів до пошуку інформації. Створені алгоритми парсинга можуть використовуватися програмістами у створенні власних програмних продуктів.

Список бібліографічного опису

1. Абрамова Т. А. Розробка парсинг-системи для отримання прихованих посилань зі сторінок соціальних мереж *Вісник ПензГУ*. 2016. №3 (15). С. 22–31.
2. Вдовін І. В. Застосування технології Web Mining до вилучення наукових даних в мережі інтернет *Інформаційні технології в науці і виробництві: матеріали всерос. молодеж. науч. – техн. конф. Омськ: Вид-во ОмГТУ, 2015*. С. 10–14.
3. Вдовін І. В. Актуальні питання автоматичного вилучення даних з веб-сторінок *перспективи розвитку інформаційних технологій*. 30 січня 2015 *Новосибірськ, 2015*. С. 11–16.
4. Комарова А. В., Менщиков А. А. Метод автоматизованого вилучення адрес з неструктурованих текстів *International Journal of Open Information Technologies*. 2017. №11. С. 43–55.
5. Менщиков А. А., Гатчин Ю. А. Побудова системи виявлення автоматизованого збору інформації з веб-ресурсів *інженерні кадри – Майбутнє інноваційної економіки: в 8 ч. 2015. Т. 4*. С. 58–61.
6. Менщиков А. А. Вивчення поведінки засобів автоматизованого збору інформації з веб-ресурсів *Питання кібербезпеки*. 2017. №3 (21). С. 49–56.
7. Мосалев П. М. Огляд методів нечіткого пошуку текстової інформації *Вісник Московського державного університету друку*. 2016. № 2. С. 87–91.
8. Суханов А. А., Маратканов, А. С. аналіз способів збору соціальних даних з мережі Інтернет *International Scientific Review*. 2017. Bun. 1. С. 22–25.
9. Чиркин Е. С. Деякі проблеми автоматизованого вилучення даних з веб-сторінок *Інтернет і сучасне суспільство. Санкт-Петербург, 2015*. С. 291–294.
10. Madaan, A. Hadoop: Solution to Unstructured Data Handling *Big Data Analytics. Advances in Intelligent Systems and Computing*. 2017. № 1. P. 47–54.
11. Skobelev, V. V. Analysis of the structure of attributed transition systems without hidden transitions *Cybernetics and Systems Analysis*. 2017. № 2. P. 165–170.

References

1. Abramova T. A. Rozrobka parsynh-systemy dlja otrymannja prykhovanyx posylan' zi storinok social'nyx merezh *Visnyk PenzHU*. 2016. #3 (15). S. 22–31.
2. Vdovin I. V. Zastosuvannja tehnolohiji Web Mining do vyluchennja naukovyx danyx v merezi internet *Informacijni tehnolohiji v nauki i vyrobnyctvi: materialy vseros. molodez. nauč. – techn. konf. Oms'k: Vyd-vo OmHTU, 2015*. S. 10–14.

3. Vdovin I. V. Aktual'ni pytannja avtomatyčnogo vylučennja danyx z veb-storinok *perspektyvy rozvytku informacijnyx tehnolohij*. 30 sičnja 2015 Novosybirsk, 2015. S. 11–16.
4. Komarova A. V., Menščykov A. A. Metod avtomatyzovanoho vylučennja adres z nestrukturovanых tekstiv *International Journal of Open Information Technologies*. 2017. #11. S. 43–55.
5. Menščykov A. A., Hatčyn Ju. A. Pobudova systemy vyjavlennja avtomatyzovanoho zboru informaciji z veb-resursiv *inženerni kadry – Majbutnje innovacijnoji ekonomiky: v 8 č. 2015. T. 4*. S. 58–61.
6. Menščykov A. A. Vyvčennja povedinky zasobiv avtomatyzovanoho zboru informaciji z veb-resursiv *Pytannja kiberbezpeky*. 2017. #3 (21). S. 49–56.
7. Mosalev P. M. Ohljad metodiv nečitkocho pošuku tekstovoji informaciji *Visnyk Moskovs'koho deržavnoho universytetu druku*. 2016. #2. S. 87–91.
8. Suxanov A. A., Maratkanov, A. S. analiz sposobiv zboru social'nyx danyx z mereži Internet *International Scientific Review*. 2017. Vyp. 1. S. 22–25.
9. Čyrkyn E. S. Dejaki problemy avtomatyzovanoho vylučennja danyx z veb-storinok *Internet i sučasne suspil'stvo. Sankt-Peterburh*, 2015. S. 291–294.
10. Madaan, A. Hadoop: Solution to Unstructured Data Handling *Big Data Analytics. Advances in Intelligent Systems and Computing*. 2017. № 1. P. 47–54.
11. Skobelev, V. V. Analysis of the structure of attributed transition systems without hidden transitions *Cybernetics and Systems Analysis*. 2017. № 2. P. 165–170.

DOI: 10.36910/6775-2524-0560-2020-39-27

УДК: 004.4

Мельник Катерина Вікторівна, к.т.н., доцент

<https://orcid.org/0000-0002-9991-582X>

Костючко Сергій Миколайович, к.т.н., доцент

<https://orcid.org/0000-0002-1262-6268>

Мельник Дмитро Сергійович, студент

Луцький національний технічний університет

ОПТИМІЗАЦІЯ ВЕДЕННЯ ТА АНАЛІТИКИ ФІНАНСІВ ЗА ДОПОМОГОЮ МОБІЛЬНОГО ДОДАТКУ НА ОСНОВІ ОС ANDROID

Мельник К.В., Костючко С.М., Мельник Д.С. Оптимізація ведення та аналітики фінансів за допомогою мобільного додатку на основі ОС Android. Фінансова грамотність один з найнеобхідніших навиків для сучасної людини. Зростання матеріальних статків у значній мірі залежить від того, на скільки розумно людина використовує ресурси та розпоряджається грошима. Ведення та аналітика доходів і витрат перший крок до успіху у керуванні фінансами. У даній статті розглядається розробка додатку в IDE Android Studio для ведення та аналітики фінансів.

Ключові слова: смартфон, OS Android, фінансовий облік, фінансова аналітика.

Мельник К.В., Костючко С.Н., Мельник Д.С. Оптимизация учета и аналитики финансов с помощью мобильного приложения на основе операционной системы Android. Финансовая грамотность один из самых необходимых навыков в жизни современного человека. Увеличение материального состояния зависит от того, насколько разумно человек использует ресурсы и распоряжается деньгами. В данной статье рассматривается разработка приложения в IDE Android Studio для учета и аналитики финансов.

Ключевые слова: смартфон, OS Android, финансовый учет, финансовая аналитика.

Melnyk K.V., Kostiuchko S.M., Melnyk D.S. Optimization of accounting and financial analytics using a mobile application based on the Android operating system. Financial literacy is one of the most necessary skills in the life of a modern person. The increase in material condition depends on how rationally a person uses resources and manages money. This article discusses application development in the Android Studio IDE for accounting and financial analytics.

Keywords: smartphone, OS Android, financial accounting, financial analytic.

1. Постановка проблеми. Фінансова грамотність один з найнеобхідніших навиків для сучасної людини. Зростання матеріальних статків у значній мірі залежить від того, на скільки розумно людина використовує ресурси та розпоряджається грошима. Ведення та аналітика доходів і витрат перший крок до успіху у керуванні фінансами.[1]

Найзручнішим засобом для обліку грошей є мобільний додаток. Оскільки виходячи з дому людина завжди бере з собою смартфон, а тому може моментально записати будь-які свої фінансові операції. До того ж такий додаток може проводити аналіз даних, що також необхідно для фінансово грамотної людини. Сучасні гаджети дозволяють робити це в автоматичному режимі, позбавляючи необхідності механічно віднімати та додавати суми. Додатки для контролю фінансів пропонують лише вводити дані про свої доходи та витрати і робити висновки про свою економічну ситуацію на основі доступних та зрозумілих графіків.[2]

Отже, мобільний додаток найзручніший та найкращий спосіб для ведення та аналітики фінансів.

2. Аналіз останніх досліджень і публікацій.

Аналізуючи Play Market можна зробити висновок, що подібні додатки мають більше 100 тисяч завантажень, що підтверджує те, що все більше людей користується мобільними додатками для обліку фінансів, тому і виникає необхідність в зручних інструментах для полегшення процесу запису та аналізу. Існують як безкоштовні так і платні програми. Також є додатки з обмеженим функціоналом, щоб отримати повний доступ до всіх можливостей програми необхідно оплатити підписку. Найчастіше якраз такі найбільш необхідний функціонал є платним.

Топ-п'ятірку бестселерів у категорії «Фінанси» Google Play визначив так: CoinKeeper, Wallet, Monefy, «Дзен-мані» та «Домашня бухгалтерія». При цьому всі додатки в топі виявилися з безкоштовним основним функціоналом.[3]

Порівнюючи ці додатки можна зробити такий висновок – інтерфейс програм складний та перевантажений. Процес запису фінансової операції займає близько 5 кліків, що погіршує досвід користування додатком. Також не у всіх додатках є можливість експорту/імпорту даних. Аналітика пропонує стандартний набір інструментів та не завжди надає достатньо інформації для аналізу.

3. Виділення невирішених раніше частин загальної проблеми. Отже, актуальним завданням є розробка мобільного додатку для обліку фінансів та аналітики даних. З простим та зручним інтерфейсом для швидкого запису фінансових операцій, інструментами, які допоможуть аналізувати, експортувати та імпортувати дані. Та привчити користувача до запису своїх витрат та доходів.

4. Формулювання мети дослідження. Мета дослідження полягає в проведенні аналізу існуючих аналогів для ведення та аналітики грошових потоків та розробка власного додатку обліку фінансів з включенням платних функцій.

5. Виклад основного матеріалу дослідження. Створюючи проект розробник може обрати найновішу версію Android, використовувати усі найновіші функції. Але у даному випадку у користувачів, які мають нижчі версії Android можуть бути проблеми у використанні додатку, або взагалі неможливість його використання у зв'язку з не підтримкою певних функцій. Даний вибір є не рентабельним з комерційної точки зору, оскільки зазвичай найновіші версії Android встановлено на 1-10% смартфонів. До прикладу Android 7.0 (Nougat), згідно з інформацією наданою в Android Studio встановлено 8.1% девайсів, старішу версію Android 6.0 (Marshmallow) - 39.3%. В той же час обравши нижчу версію ОС розробник не має можливості використовувати всі переваги нових версій, але підтримувати нормальну роботу додатку буде більша кількість смартфонів.

Даний проект створений для Android 5.0, оскільки 71% девайсів підтримуються з цієї версії Android, незважаючи на те, що дана версія не є найновішою, але ще доволі популярна у використанні.

Середовищем розробки для цього було обрано Android Studio, на мові Kotlin, використовуючи MVP (Model-View-Presenter) шаблон проектування, для чистоти коду та розділення відповідальності.

Android Studio є офіційним середовищем розробки від компанії Google, з'явилася ще в травні 2013. Це заміна інструментів розробки для Android Eclipse (ADT), як основного середовища розробки для власних розробників додатків Android. Пакет інсталяції цього середовища можна абсолютно безкоштовно завантажити з сайту <http://developer.android.com/sdk/index.html>.

В MVP шаблоні View компонент має містити посилання на Presenter, для цього застосуємо ще один паттерн Dependency Injection (DI), для цього використаємо бібліотеку Dagger 2, яка вирішить проблему впровадження залежностей. Передавати залежності клієнту замість дозволити йому створювати сервіс є фундаментальною вимогою до MVP шаблону проектування. Для уникнення проблем з життєвим циклом, шаблонним кодом та простішої реалізації шаблону MVP використаємо бібліотеку Моху.

Якщо додаток неможливо зробити цілковито синхронним, то наявність єдиного асинхронного ресурсу повністю позбавляє всіх переваг традиційного імперативного стилю програмування. Якби написаний код виконувався синхронно та в одному потоці, то він би виконувався послідовно та без будь-яких проблем, проте коли є необхідність в асинхронності, наприклад при виконанні запиту до бази даних.

Всі данні про фінансові операції зберігаються локально, тобто в пам'яті пристрою, тому потрібно створити БД (базу даних) та організувати роботу по збереженню та читанню даних. Для цього використаємо Room.

Room – це високорівневий інтерфейс для низькорівневих прив'язок SQLite, вистроєних в Android. Даний інтерфейс виконує більшу частину своєї роботи під час компіляції, створюючи API-інтерфейс надбудову над SQLite API. Це спрощує роботу програміста та виключає необхідність використання Cursor та ContentResolver. В даному додатку використовується три таблиці бази даних:

- Доходи.
- Витрати.
- Категорії.

Ось так виглядає інтерфейс таблиці витрат (рис. 1). В ньому описані методи для збереження та читання даних таблиці.

```
@Dao
interface CostsDao {

    @Query("SELECT * FROM ${DatabaseConfiguration.Costs.TABLE_NAME}")
    fun getAll(): Single<List<CostsEntity>>

    @Query("SELECT * FROM ${DatabaseConfiguration.Costs.TABLE_NAME} WHERE ${DatabaseConfiguration.Costs.Columns.DATE} BETWEEN :from AND :to")
    fun getBetweenDates(from: Date, to: Date): Single<List<CostsEntity>>

    @Query("SELECT * FROM ${DatabaseConfiguration.Costs.TABLE_NAME} WHERE ${DatabaseConfiguration.Costs.Columns.CATEGORY_ID} = :categoryId")
    fun getByCategory(categoryId: Int): Single<List<CostsEntity>>

    @Insert(onConflict = OnConflictStrategy.REPLACE)
    fun insert(costsEntity: CostsEntity): Long

    @Query("DELETE FROM ${DatabaseConfiguration.Costs.TABLE_NAME}")
    fun clear()
}
```

Рис. 1 Інтерфейс таблиці витрат

На головному екрані додатку відображається навігаційне меню та круговий графік з місячним фінансовим оборот у відсотковому співвідношенні (рис. 2). На задньому фоні розміщено view елемент з плаваючими хвилями, це надає користувачу відчуття спокою та врівноваженості, згідно з

дослідженнями, це сприяє кращому враженню від користування додатком, збільшує ймовірність повторного користування та зменшує ймовірність видалення.

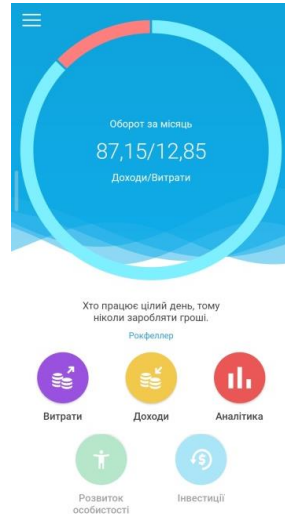


Рис. 2. Головний екран

Користувачу доступний стандартний набір функцій. Для відкриття додаткового функціоналу необхідно оплатити підписку. Доступні підписки на місяць та рік. Механізм покупки та перевірки активних підписок реалізовано за допомогою стандартного сервісу від Google - In-app Billing Subscriptions. Даний інструмент дозволяє автоматизувати списання коштів з рахунку користувача при наявності активної підписки та дозволяє перевіряти статус підписки.[4]

До платних функцій додатку входять:

- Створення кастомних категорій.
- Фільтрація по датах.
- Імпорт/експорт даних для відновлення.
- Експорт даних у файл Excel.

Однією з головних функцій додатку є аналітика. В аналітиці відображаються графіки доходів і витрат (рис. 3) та графік різниці доходів/витрат. Графік різниці у фінансово грамотних користувачів повинен бути вище 0.

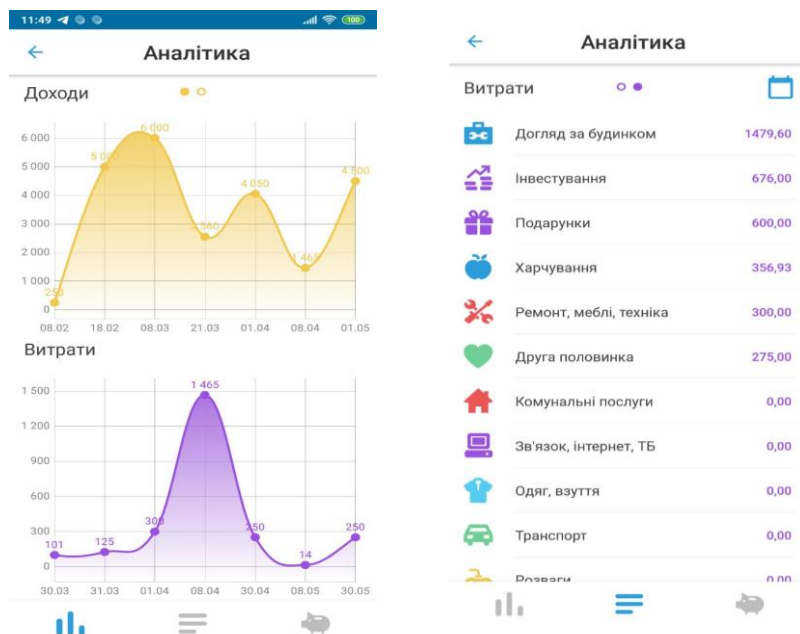


Рис. 3 Графіки доходів і витрат та список витрат по категоріям

Особливою функцією в аналітиці є розрахунок вартості години роботи, яка розраховується з даних про витрати за попередні місяці. І показує необхідний мінімальний дохід за годину для покриття витрат. Це дозволяє оцінити та порівняти свій рівень доходу.

Додаток дозволяє розрахувати дату фінансової незалежності, тобто коли пасивні доходу приносять прибутку більше витрат. Для цього є спеціальна категорія Скарбничка, яку користувач поповнює раз в місяць. Для розрахунку дати фінансової незалежності необхідно ввести відсоток доходу, який користувач отримує з грошей накопичених в скарбничці. Після чого відбуваються певні обчислення і коли дохід з відсотків перевищує місячні витрати – користувач вважається фінансово незалежним.

Вище перераховані інструменти аналітики є унікальними і не мають аналогів у інших додатках.

Також в аналітиці відображаються дані витрат та доходів по категоріям з можливістю сортувати дані по даті (рис. 3). Даний інструмент присутній не у всіх додатках, але за допомогою нього можна з легкістю проаналізувати на що витрачається найбільше коштів. Це дає можливість користувачеві оптимізувати свої витрати у майбутньому. Дрібні та необдумані покупки часто залишаються непоміченими для людини, але займають значну частину витрат, а постійний облік фінансів та дані аналітичні інструменти дозволять контролювати свої фінанси та зменшити витрати в майбутньому.

Для того, щоб користувач не забував записувати свої фінансові операції у додатку є функція сповіщень, увімкнути та вискнути які можна в налаштуваннях. Сповіщення приходять два рази на день з нагадуванням про необхідність записати свої доходи/витрати. Оскільки додаток працює без взаємодії з сервером, тому і сповіщення налаштовуються локально. Реалізовано це з допомогою AlarmManager, в якому задається таймер до наступного пробудження та задається PendingIntent. PendingIntent – це свого роду контейнер, в який можна запакувати дані. При пробудженні додатку у фоні від AlarmManager, відкривається цей контейнер з даними, після чого запускається сервіс, який вказали у Intent, а об'єкт класу Intent записали у PendingIntent. Приклад реалізацій планувальника сповіщень (рис. 4).

```
private fun scheduleNotification() {
    val notificationIntent = Intent( packageContext this, NotificationPublisher::class.java)

    val pendingIntent :PendingIntent! = PendingIntent.getBroadcast( context this, requestCode 0, notificationIntent, PendingIntent.FLAG_UPDATE_CURRENT)

    val alarmManager :AlarmManager = getSystemService(Context.ALARM_SERVICE) as AlarmManager
    alarmManager.setRepeating(AlarmManager.RTC_WAKEUP, System.currentTimeMillis(), intervalMillis: 60 * 60 * 1000, pendingIntent)
}
```

Рис. 4 Метод для планування сповіщень

Купуючи підписку користувач отримує можливість імпортувати та експортувати дані для збереження, а також зберігати дані у файлі Excel. Ця функція також є унікальною, та немає аналогів у інших додатках. Вона буде корисною для просунутих користувачів, у програмі Excel за допомогою різних інструментів та формул можна проводити будь-який аналіз, проводити вибірки, будувати графіки та діаграми, а також просто переглянути таблицю з даними, оскільки не зважаючи на зручність та популярність смартфонів, переглядати великі таблиці даних зручніше на великому екрані. Завдяки даній функції користувач отримує таку можливість. Для реалізації створення *.xlsx файлів використовується Apache POI API. Дане API має потужний функціонал, та дозволяє створювати листи заповнювати клітинки, стилізувати їх, записувати формули, а також читати ці файли. Це свого роду конструктор *.xlsx файлів. Ось так виглядає заповнення даних у клітинки таблиці:

```
for(int i=0; i<yourArrayList.size(); i++){
    Row row = sheet.createRow(i);
    row.createCell(CELL_INDEX_0).setCellValue(VALUE_YOU_WANT_TO_KEEP_ON_1ST_COLUMN);
    row.createCell(CELL_INDEX_1).setCellValue(VALUE_YOU_WANT_TO_KEEP_ON_2ND_COLUMN);
}[5]
```

Створюючи резервну копію даних, створюється файл формат *.json. Це простий, зручний та популярний формат передачі масивів інформації. Конвертація даних у json формат не займає багато часу та зусиль. Для цього необхідно створити об'єкт класу Gson, після чого викликати метод цього класу toJson(), параметром цього методу може бути об'єкт будь-якого типу, що є доволі зручним та не вимагає додаткових конвертацій.

```
val gson = GsonBuilder().setPrettyPrinting().create()
```



```
jsonSubCategoryExportList = gson.toJson(subCategoryList)
```

6. Висновки. Мною проаналізовано аналогічні додатки на ринку та виявлено переваги та недоліки. Розроблено та реалізовано архітектуру та концепт роботи і дизайн додатку. Створено унікальні аналітичні інструменти та експорт даних у файл *.xlsx.

Перспективи подальших досліджень. В перспективі подальших досліджень є розробка взаємодії з сервером для кращої синхронізації даних між пристроями. Створення онлайн сервісу для аналізу та редагування даних. Підключення взаємодії з банкінгом для автоматичного збору інформації про фінансові операції та створення iOS версії.

Список бібліографічного опису

1. Ареф'єва І. Фінансова грамотність – мистецтво багатіти [Електронний ресурс] / Ірина Ареф'єва. – 30.04.2020. – Режим доступу до ресурсу: <https://financer.com/ua/finansova-gramotnist/>.
2. Гроші люблять облік: топ-5 додатків для контролю фінансів [Електронний ресурс] / Розробник Expensify, – 2019. – Режим доступу до ресурсу: <https://cutt.ly/rybPu7u>.
3. Гордієнко Т. Як завести здорові фінансові звички? П'ять безкоштовних додатків для обліку грошей [Електронний ресурс] / Тетяна Гордієнко // 2018 – Режим доступу до ресурсу: <https://cutt.ly/lybAcOH>.
4. In-app Billing Subscriptions со стороны сервера [Електронний ресурс] // 2013 – Режим доступу до ресурсу: <https://habr.com/ru/post/160911/>.
5. Shaki S. H. Create an excel file programmatically in Android [Електронний ресурс] / Shahadat Hossain Shaki // Mar 21, 2019 – Режим доступу до ресурсу: <https://cutt.ly/rybAYut>.

References

1. Arefieva I. Financial literacy - the art of getting rich [Electronic resource] / Irina Arefieva. – 30.04.2020 - Mode of access to the resource: <https://financer.com/ua/finansova-gramotnist/>.
2. Money loves accounting: top 5 applications for financial control [Electronic resource] / Developer Expensify, - 2019. - Resource access mode: <https://cutt.ly/rybPu7u>.
3. Gordienko T. How to start healthy financial habits? Five free applications for accounting money [Electronic resource] / Tatiana Gordienko // 2018 - Resource access mode: <https://cutt.ly/lybAcOH>.
4. In-app Billing Subscriptions from the server [Electronic resource] // 2013 - Resource access mode: <https://habr.com/ru/post/160911/>.
5. Shaki S. H. Create an excel file programmatically in Android [Electronic resource] / Shahadat Hossain Shaki // Mar 21, 2019 - Resource access mode: <https://cutt.ly/rybAYut>.

DOI: 10.36910/6775-2524-0560-2020-39-28

УДК 004.451

Мельник Василь Михайлович, к.ф.-м.н., доцент,

<http://orcid.org/0000-0001-8282-6639>

Мельник Катерина Вікторівна, к.т.н., доцент,

<http://orcid.org/0000-0002-9991-582X>

Кузьмич Олена Іванівна, к.т.н., доцент,

<http://orcid.org/0000-0002-8717-4497>

Багнюк Наталія Володимирівна, к.т.н., доцент,

<https://orcid.org/0000-0002-7120-5455>

Кравець Олег Русланович, магістрант

Луцький національний технічний університет

ДОСЛІДЖЕННЯ ПОКРАЩЕННЯ ВНУТРІШНІХ ТА ЗОВНІШНІХ ПАРАМЕТРІВ ШВИДКОДІЇ ЗВ'ЯЗКУ НА КЛАСТЕРІ КОМУНІКУЮЧИХ ВІРТУАЛЬНИХ МАШИН

В.М. Мельник, К.В. Мельник, О.І. Кузьмич, Н.В. Багнюк, О.Р. Кравець. Дослідження покращення внутрішніх та зовнішніх параметрів швидкодії зв'язку на кластері комунікуючих віртуальних машин. На кластері віртуальних машин для високопродуктивної обробки даних, який є бінарно сумісним для додатків зі стандартним сокетним інтерфейсом, виявлено покращення внутрішніх та зовнішніх параметрів мережевої швидкодії за допомогою введеного в систему механізму спрощеної комунікації, реалізованого на базі Xen 3.2 та ядра Linux, який демонструє взаємодію віртуальних машин, подібно до організації зв'язку на основі UNIX DOMAIN сокетів. Встановлено, що пропускна здатність між комунікуючими машинами з використанням спрощеного зв'язку зростає приблизно на 2,11 %, ніж для традиційного TCP/IP протоколу, а швидкодія передачі повідомлень – на 7,8–7,9 %.

Ключові слова: кластер віртуальних машин, механізм спрощеної комунікації, високопродуктивна обробка, параметри швидкодії.

В.М. Мельник, К.В. Мельник, О.И. Кузьмич, Н.В. Багнюк, А.Р. Кравец. Исследование улучшения внутренних и внешних параметров быстродействия связи на кластере коммуницирующих виртуальных машин. На кластере виртуальных машин для высокопроизводительной обработки данных, который является бинарно совместимым для приложений со стандартным сокетным интерфейсом, выявлено улучшение внутренних и внешних параметров сетевой производительности с помощью введенного в систему механизма упрощенной коммуникации, реализованного на базе Xen 3.2 и ядра Linux, который демонстрирует взаимодействие виртуальных машин, подобно организации связи с помощью UNIX DOMAIN сокетов. Встановлено, что пропускная способность между коммуницирующими машинами с использованием упрощенной связи возрастает примерно на 2,11%, чем для традиционного TCP/IP протокола, а быстродействие передачи сообщений – на 7,8-7,9%.

Ключевые слова: кластер виртуальных машин, механизм упрощенной коммуникации, высокопроизводительная обработка, параметры быстродействия.

V. Melnyk, K. Melnyk, O. Kuzmich, N. Bahniuk, O. Kravets. Internal and external communication performance improving research on a cluster of communicated virtual machines. Improvements to the internal and external parameters of network performance are revealed on the virtual machine cluster for high-performance data computing that is binary compatible for applications using the standard socket interface with the simplified communication mechanism introduction implemented on the basis of Xen 3.2 and Linux kernel, which demonstrates the interoperability between machines with similar to UNIX DOMAIN sockets communication. It is revealed that bandwidth between communication machines with simplified communication is increases by about 2.11%, than for traditional TCP/IP protocol, and the speed of message transmission are increased on 7.8-7.9%.

Keywords: virtual machines cluster, simplified communication mechanism, high-performance processing, performance parameters.

1. Постановка проблеми

Для вирішення завдань високої складності в наш час використання високопродуктивних обчислень набуло високого розмаху. Подібні обчислення здійснюються не тільки в суто науковій діяльності природничих наук, як це спостерігалось ще недавно. Вони актуальні і набули сьогодні великого розмаху в різних сферах людської діяльності [1-4]. Однак, з аналізу пророблених нами літературних даних [5–8], їх продуктивність ще може зазнавати вагомих втрат від різноманітних недоліків технічного походження. Наприклад, спад продуктивності обчислень може проявлятися в залежності від низької пропускної здатності мережі та її продуктивності. Така залежність може спостерігатися як збоку зовнішньої швидкодії передачі даних, так і всередині кластера високопродуктивних обчислень. Такий спад може бути обумовлений і ростом часу затримки передачі повідомлень.

Для того, щоб вирішити вище описані недоліки, багатьма авторами наукових статей було здійснено спробу розробки та реалізації багатьох технічних та програмних механізмів. Так, з метою покращення міждомієної комунікації в роботі [5] було запропоновано використання ком-сокетів, які повинні б значною мірою обумовити зростання таких мережевих параметрів, як пропускної здатності

та зниження часу затримки отримання повідомлення. Авторами даної роботи був змодельований і практично втілений спеціальний підхід для спрощення механізму комунікації і покращення продуктивності обміну. З метою вирішення задачі швидкодії обміну даними між хмарами у наукових медичних цілях в роботі [6] було залучено практичний підхід з використанням java-сокетів та формування єдиної копії протоколу під час повторного використання ниток виконання. В даній роботі використовувалися бібліотеки спеціально розроблені для підвищення швидкості обміну повідомленнями по мережі, які, відповідно, сприяли покращенню і інших мережевих характеристик впливу на процес обміну даними та зростання його продуктивності.

В інших роботах [16] увага приділялася питанням вивчення впливу інших типів сокетів на продуктивність обробки даних на базі різноманітних комп'ютерних систем та механізми їх захисту. В деяких із них для вивчення параметрів продуктивності під час взаємодії java-додатків з мережею впроваджувались спеціальні механізми-аналізatori. У решти зазначених робіт для високопродуктивних обчислень та дослідження з'єднання зосереджувалася увага на вивченні мережевих параметрів затрат, використанні займаної пам'яті та надійності передавання даних по мережі. Проте домінуючим із найбільш актуальних в роботах вирішувалося питання підвищення параметрів продуктивності чи пропускну здатності різних типів мережі на різних ділянках її функціонування.

Досить вагомо в наші дні зростає інтерес і до високопродуктивних хмарних обчислень, а також побудови швидкісних обчислювальних систем які також активно впроваджуються в різні види громіздких обчислень для вирішення важких завдань у різних сферах людської діяльності [7,8]. В наведених наукових працях також вирішується питання підвищення продуктивності мережі з метою обміну необхідною для обробки інформацією між додатками, які з'єднуються з хмарами, що містять різноманітні медичні дані, встановлені в результаті діагностики та аналізів у різних клініках чи діагностичних центрах, а також встановлених заключень та історій захворювань під час перебування на лікуванні в стаціонарних лікарнях. Використовуючи java-сокети, було здійснено спробу вирішення задачі підвищення швидкої обробки вищезгаданих даних та роботоздатності додатків з хмарами в цілому. Для вирішення подібних проблем обробки різнобічної інформації все частіше спеціалістам з різних галузей рекомендуються програмні системи обробки у вигляді додатків, які призначені для хмарних обчислень, для яких є необхідним високошвидкісний інструмент управління доступом до даних та їх передачею по мережі. В зазначених працях заодно вирішується також питання ефективного використання задіяних ресурсів для проведення високопродуктивної обробки даних.

2. Аналіз досліджень

На основі проведених досліджень та отриманих результатів в роботах [9–11] було конкретно відзначено, що гостьовий Хеп-домен в Linux працює зі значно нижчою продуктивністю в мережі комунікації з хмарою, ніж власний домен Linux. Подібна проблема вирішується також в роботі [12], тільки в умовах, коли додаток, встановлений на віртуальній машині, комунікує з іншою віртуальною машиною, яка встановлена на іншій фізичній машині. У результаті пророблених досліджень було виявлено зниження продуктивності системи з досить помітним коефіцієнтом 2–3 по відношенню до вхідного її навантаження. Ще більше зниження продуктивності з коефіцієнтом 5 було виявлене і для вихідного навантаження на мережу з'єднання. В даних роботах констатується також, що TCP/IP сімейство протоколів, розроблене для комплексної комунікації в середовищі мережі Інтернет досить не підходить для продуктивної роботи в середовищі кластера віртуальних машин.

В роботі [16] наведено модель взаємодії двох типів зразків з'єднання між віртуальними машинами в кластері. Згідно даних запропонованої в роботі моделі для реалізації внутрідоменного зв'язку бралися до уваги віртуальні машини 1 і 2, які були встановлені на одній і тій же фізичній машині. Позадоменний зв'язок згідно моделі було втілено між віртуальними машинами 3 і 4, які були встановлені на інших окремих фізичних машинах. Для вирішення зазначеної проблеми підвищення продуктивності в роботі запропоновано також розроблений механізм спрощеної комунікації для мережевої комунікації між віртуальними машинами, які входили в склад віртуального кластера. Заодно, в експерименті ставилася задача і перевірки працюючих на цих машинах додатків на повну бінарну сумісність на основі використання стандартних сокетів та їх інтерфейсу і роботи.

Для втілення поставленої задачі продуктивності мережі та зростання швидкодії передачі повідомлень був впроваджений протокол спрощеного зв'язку, який володів низьким часом затримки передавання повідомлень. Перевага його становила і те, що розробникам зовсім не потрібно було переписувати працюючі додатки, які брали участь в комунікації з хмарою. Скорочена модель взаємодії віртуальних машин всередині кластера та позакластерна взаємодія машин зображені на рисунку 1. Запропонований в роботі протокол використовувався виключно для удосконалення

внутрішнього доменного зв'язку, а не позадоменного, що також був локалізований для з'єднання з віртуальним кластером. Іншими словами, для будь-якої віртуальної машини, що входить у склад віртуального кластера і підтримує робочий зв'язок з фізичною або віртуальною машиною поза його межами, необхідними є традиційні TCP/IP протоколи.

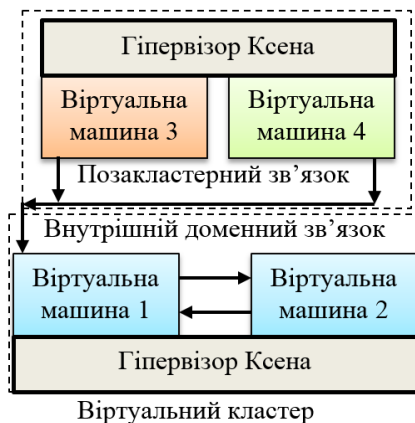


Рис. 1 – Модель комунікації віртуальних машин у кластері та позакластерний зв'язок

Для поліпшення продуктивності міждоменної мережі також потрібно було інтегрувати петлю Ксена згідно моделі роботи [13] в машині зі спрощеною комунікацією зв'язку. Для отримання кращої продуктивності в з'єднаннях міждоменної мережі, петля Ксена повинна здійснювати перехоплення вихідних пакетів повідомлень ще перед мережевим рівнем згідно моделі TCP/IP. Далі її робота полягає у виявленні тих повідомлень, які призначені для обчислювальних робіт на машинах віртуального кластера. Слід додати ще, що ці віртуальні машини об'єднані високошвидкісним каналом розділювальної пам'яті, який здійснює обхід традиційного інтерфейсу віртуалізованої мережі.

Під час аналізу літературних даних були окремі роботи, дослідження в яких базувалися на зменшенні накладних затрат, що відносяться до мережевої віртуалізації вводу/виводу та впливали на роботу і продуктивність обробки даних хмарних додатків. Зменшення мережевих затрат обумовлюється застосуванням високошвидкісного каналу зв'язку між окремими робочими доменами. В роботі [14] підвищення мережевої продуктивності обумовлене обходом операцій з залученням TCP/IP-стеків, під час яких мають прояв затрати на гортання сторінок. Обходячи їх забезпечується пряме прискорення каналу з'єднання між віртуальними доменами, розіщеними на одній і тій же фізичній машині. Використання високопродуктивного міждоменного механізму зв'язку XWay здійснює перехоплення зв'язаних викликів ще між рівнями INET та TCP. Механізм XWay застосовує спеціальний модуль комутації для визначення місця знаходження домена призначення. Якщо домен знаходиться на одній і тій же фізичній машині, то комутуючий перемикач намагається відкрити високошвидкісний XWay-канал зв'язку, поєднавши його з XWay-сокетом. Таким чином, з рівня комутатора запит INET пересилається на TCP-рівень, причому це зовсім не впливає на зовнішній зв'язок в мережі, тому що для неї XWay-канал не існує. Треба сказати також, що XWay підтримує в роботі повну бінарну сумісність додатків, що запрограмовані для сокетів зі традиційним інтерфейсом. Та все ж цей механізм призначений виключно для організації зв'язку всередині домена. Поряд з тим, багато мережевих додатків для інтенсивної обробки даних у кластерах віртуальних машин використовують не позадоменний зв'язок, а беруть за основу модель гібридної внутрідоменної комунікації [15]. В такому разі за законом Амдаля здійснення оптимізації з'єднання тільки виключно між доменами не приведе до значного зростання продуктивності такої системи [11].

Відповідно до закону GPL [13] механізм XenLoop представляється у вигляді звичайного програмного проекту з доступним вихідним кодом. Так само, як і XWay, механізм XenLoop напрямлений на підтримку підвищення продуктивності мережі всередині домена в середовищі об'єднаних віртуальних машин. На рівні користувача він сильно акцентує на питанні доступності до віртуальних машин і домагається порівняно вищої продуктивності між гостьовими машинами-співрезидентами кластера за рахунок покращення з'єднання. Його дія в основному полягає в аналізі та перевірці мережевих пакетів, які надходять на мережевий рівень, і відслідковує їх, які із них призначені для отримання і якими віртуальними машинами, враховуючи розміщення їх на одній фізичній машині. Така перевірка виконується через використання високошвидкісного каналу

комунікації, організованого за допомогою спеціально створеної розділювальної пам'яті між взаємодіючими віртуальними машинами, зовсім обминаючи традиційний мережевий інтерфейс віртуалізації. Таким чином, віртуальна машина-гість за допомогою введеного механізму XenLoop може вільно мігрувати між об'єднаними машинами, зовсім не торкаючись діючої мережевої комунікації та зовсім доступно виконувати і підтримувати з'єднання між діючим каналом XenLoop та мережевим розгалуженням комунікації кластера.

Механізм XenLoop базується на побудові програмних модулів для Linux-ядра в домені Xen з індексом 0 та інших взаємодіючих гостьових доменах. Будь-яка віртуальна машина у ньому в робочому режимі має можливість динамічно завантажувати модуль XenLoop. З проведеного аналізу та оцінки застосування послідовності немодифікуючих тестів підтверджено [13], що дія XenLoop намагається знижувати значення часу затримки обходу повідомлення при проходженні ним повного кругового маршруту між взаємодіючими віртуальними машинами приблизно в 5 разів і, відповідно, підвищувати пропускну здатність на ділянці задіяної ділянки мережі приблизно в 6 разів. Проте, як і у випадку використання XWay, за допомогою наведеного XenLoop механізму можна оптимізувати продуктивність комунікації тільки всередині домена, що розцінюється як недолік. Оскільки на мережевому рівні XenLoop здійснює перехоплення вхідних мережевих пакетів, то в роботу обов'язково задіюються додаткові ресурси CPU, які і обумовляють зайві затрати на TCP/IP-обробку. Архітектура використання механізму XenLoop в домені користувача описана в роботі [16].

Існує інший спосіб, застосований в роботі [17], який також спрямований до оптимізації мережі з підвищенням її продуктивності. Він полягає в залученні моніторингу до відтворення треку проходження пакетів пересилання між взаємодіючими віртуальними машинами, чи хоча би між привілейованими віртуальними машинами. Його використання дає можливість вагомо покращувати продуктивність зв'язку та безпечно чи відокремлено відтворювати операції вводу-виводу між віртуальними машинами. Архітектуру моніторингу проходження між робочими віртуальними машинами зображено на рисунку 2. Однак, сам моніторинг проходження між ними можна застосовувати тільки в такому віртуальному середовищі, в якому мережеве обладнання надає підтримку проходженню діючої операційної системи. То ж лише прикладні програми, які написані з урахуванням застосування інтерфейсу проходження операційної системи, мають можливість забезпечення продуктивністю загаданого моніторингу обходу між діючими віртуальними машинами.

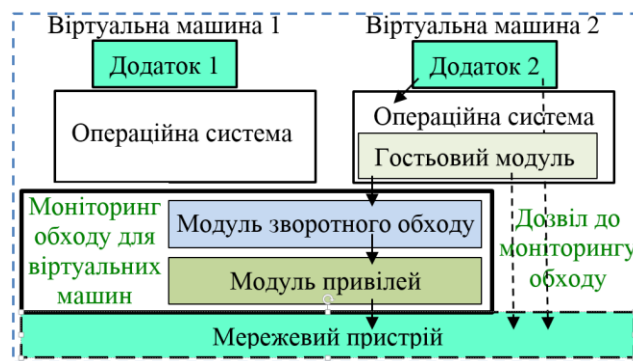


Рис. 2 – Архітектурна схема моніторингу обходу між робочими віртуальними машинами

Протокол віддаленого доступу до пам'яті розроблений у вигляді спеціального надійного датаграмного протоколу віконного типу для реалізації спрощеного зв'язку. Він цілеспрямовано здійснює віддалений доступ до сегмента розділювальної пам'яті, яка належить одній підмережі. Він може бути поданий у вигляді програмного модуля [18, 19], який здійснює координацію використання ресурсів пам'яті в широких кластерних системах з метою підтримки великого її обсягу та втілення інтенсивних навантажень для операцій вводу-виводу. Він слугує абсолютно прозорою розподіленою системою для віртуальних машин в середовищі Xen. В процесі ефективної передачі, наприклад, однієї сторінки пам'яті протокол віддаленого доступу видаляє традиційні TCP-функції, до яких відноситься функціональна абстракція, яка відповідає за потоково-байтовий порядок доставки повідомлень, здійснення контролю перевантажень і функціональності IP-маршрутизації в межах локальної мережі. Таким чином протокол віддаленого доступу до пам'яті виконує обхід традиційного стеку TCP/IP протоколів і напряду з'єднується з драйвером мережевого пристрою. Він умовно налаштовується на передачу сторінкової пам'яті таким способом, щоб інші програми, які здійснюють власну взаємодію через сокети, не отримували покращення продуктивності під час його роботи.

Дослідження було виконано на базі системи Xen 3.2 з ядром Linux 2.6.18. Надалі здійснимо короткий огляд системи з Xen та архітектури виконання операцій вводу-виводу для Xen-мережі, зображеної на рисунку 3, продемонструвавши її особливості побудови. Xen в нашому випадку – це не що інше, як монітор віртуальної машини x86, що дозволяє операційним системам багаторазового спрямування розподіляти поміж собою діюче апаратне обладнання. Використовуючи Xen є можливість запуску декількох операційних систем для одночасної роботи на одній фізичній машині, де кожна операційна система виступає гостьовим доменом, між якими повинен бути один домен 0 з привілейованим хостингом для здійснення управління діючого програмного забезпечення на рівні додатків користувача.

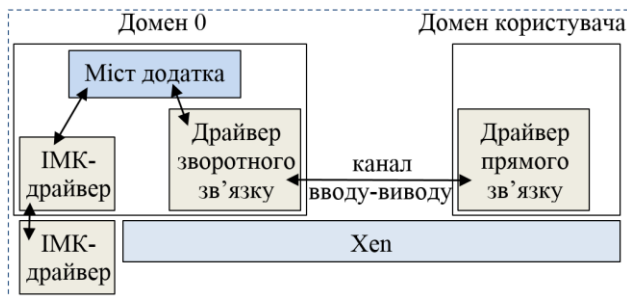


Рис. 3 – Схема архітектури для операцій вводу-виводу в мережі Xen

Дана схема представлення Xen-системи подає два віртуалізаційні підходи. Перший підхід реалізований на основі представленої абстракції діючого апаратного обладнання представляє *пара-віртуалізацію*, яка подібна, однак не ідентична до схем роботи апаратних комплексів, сконструйованих для функціонування операційних систем для гостьових користувачів. А тому деякі з діючих операційних систем, зокрема сімейства Windows, не спроможні працювати в даному режимі роботи. Повна абстракція підлеглого апаратного забезпечення реалізує *повну Xen-віртуалізацію*, проте вона потребує робочої підтримки з процесорної сторони.

Діючі в наш час мережеві комунікації, спроектовані у відповідності до Xen здійснюють свою роботу через використання мережевого інтерфейсу віртуалізації. На рисунку 3 схематично наведений домен користувача, який здійснює з'єднання з головним доменом через власний фронт-драйвер, або *мережевий вхідний драйвер*. В домені 0 для організації комунікації повинен функціонувати драйвер зворотного зв'язку, або *мережевий вихідний драйвер*. в кожному гостьовому домені мережевий інтерфейс входу повинен бути підключений до відповідного інтерфейсу в домені 0 через власний зворотній або вихідний мережевий кінець. Діючі драйвери кінця *виходу* повинні фізично з'єднуватися через програмний міст з мережевою платою. Мережеві додатки поєднані з роботою гостьових операційних систем надсилають свої повідомлення на власний фронт-драйвер, який, в свою чергу, продовжує надсилання повідомлень до зворотного чи вихідного драйвера, відтворюючи операції їх копіювання чи сторінкового гортання.

Проаналізувавши наведені в літературі наукові результати можна зробити відповідний висновок, що питання зменшення додаткових мережевих затрат які мають місце у функціонуванні кластера віртуальних машин та дослідження параметрів швидкодії передачі даних з метою покращення внутрішніх та зовнішніх її параметрів частково вирішувалося тільки деякими науковими роботами. Проте загальне питання дослідження параметрів швидкодії мережі комунікації на різних рівнях саме з метою покращення продуктивності роботи мережевих додатків для високопродуктивних обчислень на віртуальних машинах в хмарі в науковій літературі розглядалося значно менше і на сьогодні становить актуальність подібних наукових досліджень. Також в наш час велика увага приділяється і науковим завданням введення нових ефективних механізмів зв'язку та дослідження їх параметрів швидкодії з метою покращення продуктивності роботи мережевих додатків, враховуючи не тільки внутрікластерну взаємодію, а розглядаючи її в інтеграції з позакластерною та її мережевими ділянками. Додатковим питанням також впливає і бінарна сумісність роботи мережевих додатків, починаючи вже від етапу їх розробки.

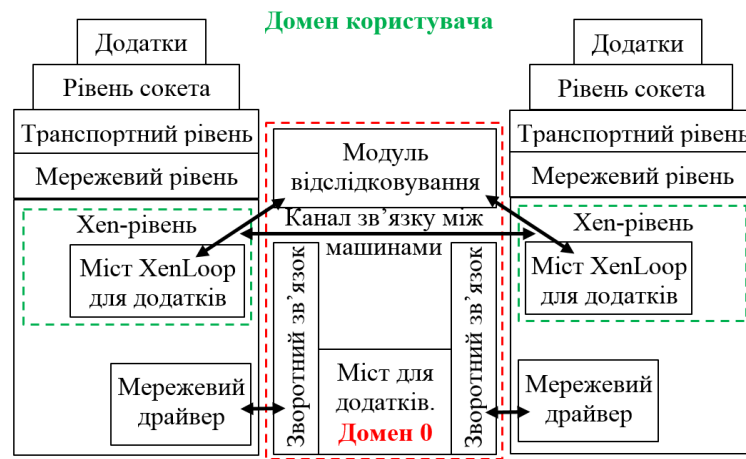


Рис. 4 – Архітектурна схема механізму спрощеної комунікації з Xen

3. Мета і завдання дослідження

Метою даної роботи є дослідження покращення внутрішніх та зовнішніх параметрів швидкодії зв'язку на кластері комунікуючих віртуальних машин на основі впровадженого механізму спрощеного зв'язку в даний хмарний кластер, призначений для виконання високопродуктивних обчислень. Виходячи з даної мети механізм повинен підтримувати бінарну сумісність додатків в процесі їх роботи, не зважаючи на те, що вони створені на основі використання стандартного інтерфейсу для сокетів віртуальних машин.

В дослідженні було намічено наступні завдання для досягнення поставленої мети:

- використавши діючий механізм спрощеного зв'язку, довести кількісне покращення продуктивності мережі у кластері віртуальних машин для виконання хмарних високопродуктивних обчислень. В системі, взятій для проведення дослідження повинен бути використаний розроблений протокол спрощеної комунікації, який би підтримував згадану в меті дослідження бінарну сумісність мережесих додатків у ході їх роботи.

- для покращення продуктивності внутрідоменної комунікації між віртуальними машинами в спроектованій системі впровадити XenLoop-петлю з метою здійснення перехоплення надісланих пакетів повідомлень ще на рівні, нижче мережевого. В процесі дослідження необхідного нам параметра пропускної здатності між двома віртуальними машинами з урахуванням механізму спрощеної комунікації та спеціально розробленого протоколу нам необхідно застосувати еталонний тест з характерним набором NAS-міток і переконатися про проходження наміченого пакета через створений канал FIFO, (див. рис. 4) обумовлений введенням XenLoop-петлі [25].

- в ході проведення дослідження з залученням протоколу спрощеного зв'язку та протоколу віддаленого доступу до пам'яті експериментально довести зростання пропускної здатності в комунікації між доменами, що представлені відповідними віртуальними машинами, яка залежить від міждоменної мережевої пропускної здатності та часу затримки.

4. Виклад основного матеріалу й обґрунтування отриманих результатів дослідження.

Весь механізм спрощеної комунікації об'єднує в собі два модулі. Перший модуль втілює протокол спрощеного зв'язку, що здійснює управління позадоменним мережесим з'єднанням до кластера віртуальних машин. Другий модуль реалізує рівень XenLoop, через який протікає високошвидкісний комунікаційний канал для можливості покращення з'єднання всередині домена між віртуальними машинами у віртуальному кластері [17].

Серед всіх модулів основним є модуль, який відтворює протокол спрощеної комунікації та його роботу. Він містить в собі три об'єднані модулі, перший з яких представляє ділянку ядра. Для перехоплення повідомлень, які надходять зі сторони додатків вищого рівня зі збереженням стабільними програмованих інтерфейсів [20, 21], необхідно в гостьових операційних системах провести незначні зміни, тобто в традиційному ядрі Linux. Інший підлеглий модуль реалізує розроблений протокол спрощеного зв'язку і втілений в змодельовану систему. З літератури відомо, що сімейство TCP/IP протоколів інтегрує в собі адресацію звернень, мультиплексію повідомлень, управління з'єднаннями між кінцевими машинами, контроль шляху проходження, регламентування передачі по лінії зв'язку з метою уникнення перевантажень, фрагментації пакетів з можливістю їх перебудови і інші дії. Він досить добре повинен поєднувати свою роботу в багатоконпонентному мережевому Інтернет-середовищі. Проте, робота протоколу TCP/IP в середовищі віртуального

кластера для високопродуктивних обчислень та роботи з хмарою, в якій інфраструктура мережі володіє набагато вищою продуктивністю порівняно з мережею Інтернет, стає в значній мірі затрудненою для машинних процесорів. На основі цих обставин появилася ідея, що привела до створення протоколу спрощеної комунікації, призначення і робота якого описані в [16]. На діаграмі, зображеній на рисунку 5 зображена логічна послідовність операцій під час встановлення з'єднання з використанням протоколу спрощеного зв'язку.

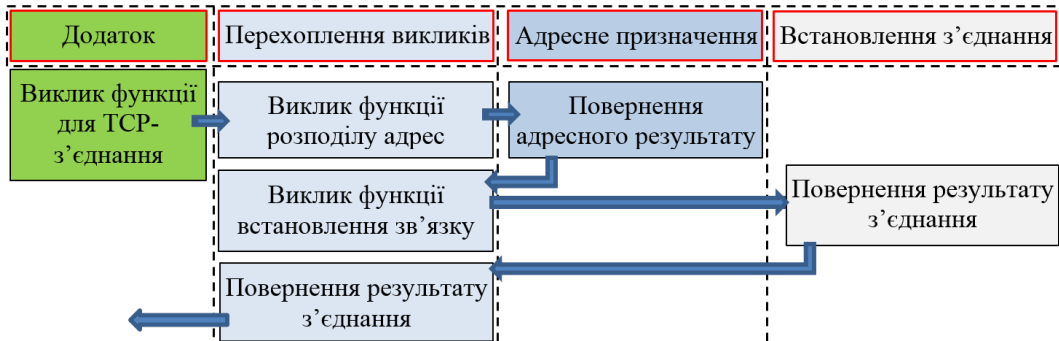


Рис. 5 – Діаграма логічної послідовності встановлення з'єднання з використанням протоколу спрощеного зв'язку

Наведемо покроковий опис встановлення зв'язку для працюючого додатка в наведеній вище системі з задіяним у роботу протоколом спрощеного зв'язку. Як видно з рисунка 5, діючий додаток, що знаходиться на віртуальній машині, першим кроком викликає сокетну функцію з метою встановлення TCP-з'єднання для обміну повідомленнями. Описана вище надбудова, яка включена в ядро Linux, перехоплює цей виклик і для функції визначення виставляє конкретне завдання виявлення місця знаходження віртуальної машини протилежної сторони зв'язку. В цьому завданні головним є визначити, чи вона знаходиться на тій же самій чи іншій фізичній машині. Функція визначення обов'язково повинна повернути певний результат щодо місця знаходження віртуальної машини протилежної сторони зв'язку. Якщо виявлено, що місцем знаходження віртуальних машин є різні фізичні машини, то за допомогою протоколу спрощеного зв'язку виконується спроба виклику функції встановлення з'єднання. В цей же час здійснюється виділення потрібних структур даних та ресурсів, необхідних для встановлення зв'язку. Виконавши описані процедури, функція встановлення зв'язку намагається повернути очікуваний результат, використовуючи цей же протокол спрощеного зв'язку. У випадку, коли дві гостьові співрезидентні віртуальні машини намагаються встановити активне мережеве з'єднання між собою, створюється динамічний двосторонній канал зв'язку типу FIFO для передачі пакетів між ними, використовуючи відомий нам традиційний протокол "двостороннього рукоштовкування". Детально описана діаграма, яка висвітлює послідовність зв'язку всередині домена, зображена на рисунку 6. IP-протокол надсилає датаграму повідомлення, яку сепарує призначена функція перехоплення. Використовуючи інформацію модуля відслідковування повідомлень, дана функція визначає приналежність датаграми з використанням для передачі тільки внутрідоменної комунікації. Якщо модулем XenLoop визначається, що датаграма буде використовувати комунікацію всередині домена, то її подальша передача буде виконуватися через високошвидкісний канал між віртуальними машинами, працюючими всередині домена. Звідси також і здійсниться повернення результату про з'єднання.

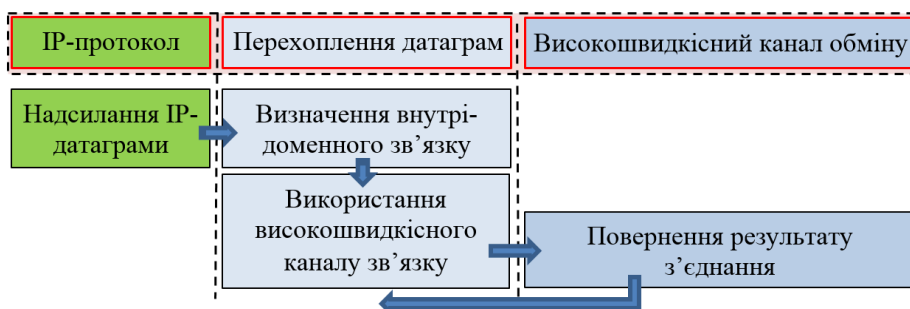


Рис. 6 – Вигляд послідовності комунікації всередині домена

Зупинимося на особливостях роботи модуля, який відтворює протокол спрощеного зв'язку. Всі інші робочі модулі, задіяні в реалізації системи згідно механізми спрощеної комунікації досить добре представлені в роботі [20]. Модуль, який відтворює протокол спрощеного зв'язку, як і наведений в цій роботі модуль XWay, містить в собі три розглянуті нами програмні модулі, такі як додаток до Linux-ядра, модуль завантаження для ядра та рівня доменного використання. Як уже згадувалося, додаток до ядра працює в сторону виконання бінарної сумісності з іншими сокетними додатками та узгоджується в роботі з модулем виконання завантажень для ядра під час передачі пакетів даних з використанням протоколу спрощеного зв'язку. Програмний модуль завантаження для ядра також включає в себе протокол спрощеного зв'язку, з яким допомагає встановити зв'язок і сам домен на рівні користувача. В додатку до ядра проходить модифікація двох відомих структур ядра Linux, таких як *struct inet_stream_ops* та *struct tcprot*, які в традиційному вигляді містять адресні вказівники на функції, які реалізують базові операції контенту виконання для мережі TCP, до яких відносяться з'єднання, відправка, отримання, виключення і інші. Дані вказівники було заміщено вищезгаданими функціями. Кожна із згаданих функцій володіє власним твердженням, яке чітко визначає обробку поточної операції протоколом спрощеного зв'язку чи спеціальною функцією ядра Linux. У випадку використання спрощеного зв'язку вона буде оброблена протоколом спрощеного зв'язку. В іншому випадку операція повинна оброблятися звичайним традиційним TCP/IP-протоколом. Детальний опис програмних структур та їх застосування для відтворення комунікації описані в [16].

Будь-яка датаграма протоколу спрощеного зв'язку містить набір полів в спеціально організованому порядку. Машина-одержувач повинна знати, як потрібно вести зчитування і розшифрування отриманого потоку даних. Заголовок датаграми протоколу спрощеного зв'язку, формат якої показаний на рисунку 7, містить вісім основних полів. Протокол спрощеного зв'язку володіє двома типами датаграм: датаграма розміщення даних та датаграма контрольної інформації для надісланого пакета. Останнє поле номера каналу зберігає номер буферного каналу для активізації з'єднання з використанням протоколу спрощеного зв'язку. Для забезпечення надійної передачі датаграм система спрощеного зв'язку використовує спеціально розроблений віконний алгоритмом передачі повідомлень, описаний в [22].

Контрольна сума	Номер послідовності	Початковий індекс	Довжина корисної частини датаграми
Порядковий номер підтвердження	Індексний номер підтвердження	Тип	Номер каналу
Корисна частина датаграми			

Рис. 7 – Формат датаграми для протоколу спрощеного зв'язку

Оцінимо вплив введеного в запропоновану систему механізму спрощеного зв'язку [23] на основі отриманих результатів з експерименту, які розділимо на дві мережеві частини: внутрішню та позадоменну комунікації, з введеним протоколом для віддаленого доступу до пам'яті та механізмом спрощеної комунікації подвійно сумісним з працюючими на віртуальних машинах мережевими додатками. У ході експерименту було використано дві однакові фізичні машини, кожна з яких була оснащена двох-ядерним процесором з частотою 2,5 ГГц, об'ємом пам'яті 4 ГБ, SATA-диском, мережевою платою Gigabyte для комунікації в Gigabyte Ethernet мережі. Програмне забезпечення CentOS 5.0 з ядром Linux 2.6.18.8 і версією Xen 3.2.0 встановлювалося однаковим на обох фізичних машинах. Кожна робоча віртуальна машина була оснащена 2 VCPU з оперативною пам'яттю 512 Мбайт.

Експериментальна перевірка продуктивності для частини мережі, яка представляла позадоменний зв'язок, здійснювалася між двома фізичними машинами. Оцінка продуктивності частини мережі для внутрішнього зв'язку виконувалася на одній фізичній машині, на якій згідно моделі були встановлені дві віртуальні машини, які, згідно системної моделі, відображали схематично сам домен та внутрішній зв'язок в ньому. Отримання кількісних результатів та оцінка параметрів продуктивності двох вищезгаданих мережевих ділянок здійснювалася за допомогою методу паралельного набору *NAS*-міток і *NETPERF*-міток, згідно методики, описаної в роботі [24]. Метод тестування з паралельним набором *NAS*-міток базується на наборі часових міток [25, 26], які відтворюють такі пристрої, як процесор, кеш-пам'ять, оперативна пам'ять і, звичайно, робочі пристрої вводу-виводу паралельно в часі у відповідному наборі працюючих додатків. Відомо [24], що

блок для отримання експериментальних даних методом NAS-міток з паралельним набором об'єднує в собі п'ять ядер EP, MG, CG, FT, IS та трьох динамічно працюючих додатків BT, SP, LU. Для проведення тестів згідно вказаної методики була вибрана проблема в розмірі А-класу. Слід також додати, що модуль для NAS-міток NPB3.3-MPI був розроблений і скомпільований в середовищі OpenMPI-1.4.2 на базі системи GCC 4.1.2 [24].

Графік залежності величини відносного робочого часу з використанням протоколу спрощеного зв'язку для отриманих наборів міток на основі п'яти вищеописаних діючих ядер та ядра BT зображено на рисунку 8. Порівнюючи отримані результати можна чітко помітити, що для діючого протоколу він є значно менший, ніж для TCP/IP протоколу. В ході виконання дослідження для отримання більшої достовірності експериментальних результатів був залучений тест з набором *Netperf*-міток. Одна з робочих віртуальних машин, яка знаходилася на фізичній машині, виконувала роль *Netperf*-сервера, а друга віртуальна машина, що знаходилася на іншій фізичній машині, повинна була в якнайшвидшому режимі організувати передачу пакетів великих обсягів на віртуальну машину-сервер.

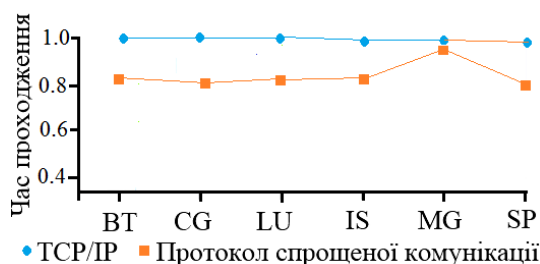


Рис. 8 – Залежність величини часу проходження від типу ядра для тесту з паралельним набором NAS-міток для класу А

В наступному було виконано тестування зі змінним часовим інтервалом вимірювання, як подано в роботі [27]. Отримані тестові результати (рисунок 9) констатують, що пропускна здатність даної мережевої ділянки з використанням протоколу спрощеного зв'язку, яка є оберненою величиною часу проходження і помітно вищою, порівняно із TCP/IP протоколом. Хоча вважається, що традиційний протокол TCP/IP відносно добре виконується в цьому середовищі і не вимагає його оптимізації роботи на якихось визначених конкретних мережевих ділянках, де б активізувався час затримки пакетів. Пропускна здатність зростає в основному на 2,11 %. Таке незначне покращення відбувається внаслідок скорочення часу запуску пакетів повідомлень та зменшення заголовкового пакета за розміром.

Для дослідження було задіяно декілька робочих віртуальних машин, що знаходилися на одній і тій же фізичній машині. Після їх старту в домені 0 був завантажений модуль відокремлення повідомлень, а на кожену робочу віртуальну машину завантажувався модуль XenLoop. Здійснимо порівняння пропускної здатності для традиційного протоколу TCP між двома робочими віртуальними машинами, що використовують *Netperf* між вхідним і вихідним драйверами механізму зв'язку та XenLoop-каналом. Далі перевіримо пропускну здатність для локального каналу замикання *loopback* та виконаємо порівняння з пропускну здатністю для каналу XenLoop.

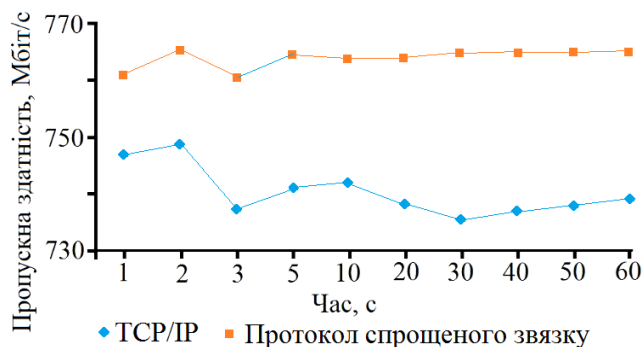


Рис. 9 – Результати зміни пропускної здатності для протоколів ПСЗ і TCP/IP в ході зміни часового інтервалу вимірювання

Рисунок 10 демонструє нам результат, який очевидно доводить, що для здійснення передачі великих пакетів даних пропускна здатність з використанням TCP-протоколу наближено покращується на півтора. Отже, пропускна здатність між взаємодіючими віртуальними машинами всередині домена, тобто співрезидентами на фізичній машині, покращується порівняно з пропускною здатністю для локального *loopback*-каналу.

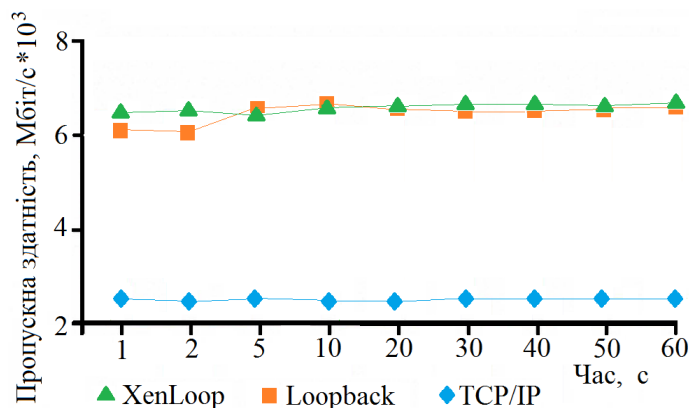


Рис. 10 – Результати тестування пропускної здатності для TCP/IP, XenLoop та Loopback

Виконаємо також порівняння отриманих даних з експерименту для запропонованої в роботі системи з протоколом спрощеного зв'язку і системи з діючим протоколом віддаленого доступу до пам'яті, які в дослідженні були використані окремо. Через них передавалася однакова визначена кількість пакетів з однієї робочої віртуальної машини до іншої віртуальної машини, розміщених на різних фізичних машинах, які реально представляли позадоменний мережевий зв'язок. Результати, наведені на рисунку 11 підтверджують, що пропускна здатність, яка обернено пропорційна часу затримки, значно покращується у системі з використанням протоколу спрощеного зв'язку. Порівнюючи даний результат результатом, отриманим таким же підходом для системи з використанням протоколу віддаленого доступу до пам'яті, зростання приблизно спостерігається на 7,86–7,92 % в залежності від об'єму пакетів передавання. Не зважаючи на те, що протокол спрощеного зв'язку та протокол віддаленого доступу до пам'яті здійснюють відправку стільки даних в кожному пакеті, скільки максимально вони спроможні, все ж між ними відчувається суттєва різниця в пропускній здатності. Хоча протокол віддаленого доступу до пам'яті призначений для організації передавання сторінок пам'яті по чергово (одна за одною), однак заключний пакет даних з розміром максимальної сторінкової пам'яті, призначений для мережевої передачі, не досягає точки призначення, тобто кінцевої віртуальної машини-одержувача. Отже, для протоколу віддаленого доступу до пам'яті порівняно з протоколом спрощеного зв'язку потрібно надіслати на кілька сторінок більше для пересилання наперед визначеної кількості інформації. Це говорить про те, що у випадку використання системи з механізмом спрощеного зв'язку та його протоколом, результуюча пропускна здатність значно покращується.

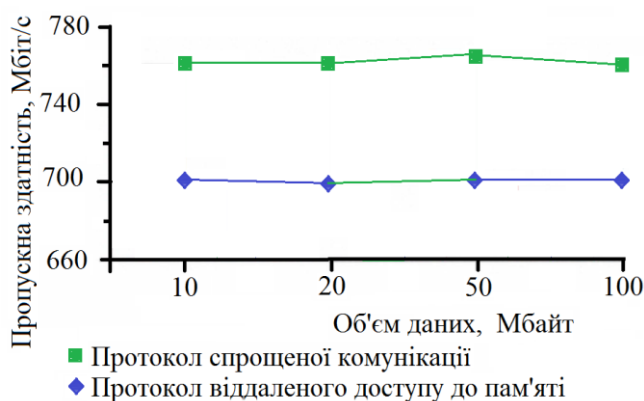


Рис. 11 – Залежність пропускної здатності від об'єму пакетів передавання для систем з протоколом спрощеного зв'язку та протоколом віддаленого доступу до пам'яті

Для перевірки на бінарну сумісність в ході роботи з різними додатками в системі з механізмом спрощеного зв'язку, потрібно було об'єднати в комунікації звичайні мережеві програми запущені на виконання. Для перевірки бінарної сумісності були запущені SCP, WGET, APACHE, MYSQL, NETPERF, TELNET, VSFTPD та інші додатки, які успішно були протестовані, за винятком тих, які не використовують традиційний інтерфейс для стандартних сокетів. Для мережевих додатків, у яких використовується стандартний сокетний інтерфейс, пропонується система з механізмом спрощеного зв'язку повністю підтримує бінарну сумісність в роботі з ними.

7. Висновки

В роботі представлено результати досліджень покращення внутрішніх та зовнішніх параметрів швидкодії зв'язку на кластері комунікуючих віртуальних машин для високопродуктивних обчислень. Для поліпшення внутрішньої та зовнішньої комунікації було застосовано систему з мережевим механізмом спрощеної комунікації, призначеною для підвищення продуктивності передачі пакетів у кластері. За рахунок введеного в систему спрощеного механізму зв'язку значно поліпшується пропускна здатність для внутрішньої та позадоменої ділянок мережі в кластері, підтримуючи сумісність з одночасно працюючими мережевими додатками, які використовують звичайний сокетний інтерфейс, та уникаючи сповільнення, характерного для початкового етапу TCP-протоколу. Даний механізм дозволяє також використовувати менший заголовок для кожного пакета передавання, що також стимулює підвищення мережевої швидкодії та продуктивності.

Виявлено, що пропускна здатність між двома комунікуючими віртуальними машинами з використанням протоколу спрощеного зв'язку зростає приблизно на 2,11 %, ніж для системи таких же віртуальних машин з використанням традиційного TCP/IP протоколу. Поліпшення параметрів внутрішнього зв'язку в системі зі спрощеною комунікацією базується на введенні в неї петлі XenLoop, за рахунок якої виконувалося розпізнавання та перехоплення надісланих пакетів на рівні нижче мережевого і перевірка призначення його будь-якій із співрезидентних машин, що утворювали внутрішній зв'язок. Пропускна здатність для мережі всередині домена покращується в середньому на півтора. Порівняння отриманих показників для системи з використанням протоколу спрощеного зв'язку клімтатує відносно підвищення швидкодії передачі повідомлень на 7,8–7,9 %.

Список бібліографічного опису

1. S. Iannucci, V. Cardellini, O. D. Barba, I. Banicescu. (2020) A hybrid model-free approach for the near-optimal intrusion response control of non-stationary systems. // *Future Generation Computer Systems*. – 2020, V 109, p. 111-124.
2. D. Black, P. Jones (2015) Differentiated Services and Real-Time Communication, Informational. RFC 7657, – Nov. 2015. – IETF, ISSN 2070-1721.
3. D. Black, P. Jones, Differentiated Services and Real-Time Communication, Informational RFC 7657, IETF, ISSN 2070-1721 (Nov. 2015).
4. R. Barik, M. Welzl, A. Elmokashfi, T. Dreiholz, S. Islam, S. Gjessing. (2019) On the utility of unregulated IP DiffServ Code Point (DSCP) usage by end systems. *Performance Evaluation*. – 2019, V. 135. – p. 102036.
5. Мельник, В.М. Побудова та використання міждоменого механізму зв'язку для високопродуктивної обробки даних [Текст] / В.М. Мельник, П.А. Пех, К.В. Мельник, Н.В. Багнюк, О.К. Жигаревич // *Східно-європейський журнал передових технологій*. – 2016. – № 1/9/79. – с. 10-15. doi: 10.15587/1729-4061.2016.60629.
6. Мельник, В.М. Вплив високопродуктивних сокетів на інтенсивність обробки даних [Текст] / Мельник В.М., Багнюк Н.В., Мельник К.В. // *Scientific Journal "ScienceRise"*. – 2015. – Том 6, № 2(11). – с. 38-48. doi: 10.15587/2313-8416.2015.44380.
7. Melnyk, V. High production of java sockets (HPJS) for health clouds in science [Text] / V. Melnyk, K. Melnyk, O. Zhyharevych // *Інженерія програмного забезпечення. Національний авіаційний університет*. – 2015. – Том 19, № 3. – С. 36-40.
8. Melnyk, V. Significance of the socket programming for the laboratory with intensive data communications [Text] / V. Melnyk, P. Pekh, K. Melnyk, O. Zhyharevych // *Всеукраїнська Науково-практична інтернет-конференція «Сучасні методи, інформаційне та програмне забезпечення систем управління організаційно-технологічними комплексами»*. (28 квітня 2015 р.). – Луцьк, Луцький НТУ. / *Науковий журнал «Комп'ютерно-інтегровані технології: освіта, наука, виробництво»*. – 2015, №20. – с. 67-71.
9. Бархем, П. Xen і мистецтво віртуалізації [Текст] / П. Бархем, Б. Драгович, К.Фрейзер та ін. // *Праці симпозиуму ACM на операційних системах принципів (SOSP 2003)*, ACM Press, Нью-Йорк, США: Болтон Landing, Lake George, жовтень 2003. – с. 164-177.
10. Пратт, І. Xen-Віртуалізація. Linux в світі, 2005 [Текст] / І. Пратт // *Віртуалізація BOF-презентація*. 2007.
11. Chisnall, D. Повне керівництво по Xen Hypervisor [Текст] / D. Chisnall // Prentice Hall, 2-е видання, 2007.
12. Менон, А. Оптимізація віртуалізації мережі в Xen. У 2006 р [Текст] / А. Менон, А. Л. Кокс, В. Звейніпоель // *USENIX щорічна технічна конференція*, Бостон, штат Массачусетс, США, червень 2006. – с. 15-28.
13. Jian, W. XenLoop: Прозора висока продуктивність між віртуальними машинами через мережевий шлейф [Текст] / В.Цзянь, К. Л.Райт, К.Гопалан // *Бостон, штат Массачусетс, США, HPDC'08*, 23-27 червня 2008.
14. Kim, K. Міжсокетний зв'язок і підтримка високої продуктивності і повної бінарної сумісності на Xen [Текст] / К. Кім, С. Кім, С.-І. Юнг, Г.-С. Шін, Дж.-С. Кім // *Сієгл Вашингтон, США, VEE'08*, 5-7 березня 2008.
15. Закон Амдаля. Електронний ресурс. Режим доступу: <http://home.wlu.edu/~whaley/classes/parallel/topics/amdahl.html>.
16. V. Melnyk, N. Bahnyuk, K. Melnyk, O. Zhyharevych, N. Panasyuk. Implementation of the simplified communication mechanism in the cloud of high performance computations. *East-European journal of Enterprise Technologies*. – Kharkiv (DOI: 10.15587/1729-4061.2017.98896). – 2017. – № 2/2/86. – p. 24–32.

17. Хуан, Дж. Л. В. Висока продуктивність обходу між МВМ для вводу/виводу в віртуальних машинах [Текст] / Дж. Л. В. Хуан, Б. Абали, Д. К. Панда // USENIX щорічна технічна конференція Архів, 2006.
18. Хайнс, М. Р. MemX: Підтримка великих робочих навантажень пам'яті в Xen для віртуальних машин [Текст] / М. Р. Хайнс, К. Гопалан // VTDC '07, Рено, Невада, США. 12 листопада, 2007.
19. Дешпанде, У. Memx: Віртуалізація кластера з широкою пам'яттю [Текст] / Дешпанде У., Ван Б., Хак С., Хайнс М., К. Гопалан // ICPP'Ю: Праці 39-ї Міжнародної конференції з паралельної обробки (2010), С. 663–672.
20. Кім, Й.-С. Проектування та реалізація рівня користувача та рівня сокета над віртуальною архітектурою інтерфейсу [Текст] / Й.-С. Кім, К. Кім, С.-І. Юнг, С. Ха // Паралельне Обч.: Практ. Експер, 15 (7-8). – 2003. – с. 727–749.
21. Син, С. VOP: сокет-інтерфейс для ВО [Текст] / С. Син, Д. Кім, Є. Лім, С. Юнг // В Інтернеті та мультимедійних системах і додатках, 2004.
22. Кларк, Д. Д. Вікно і стратегія визнання TCP [Текст] / Д. Д. Кларк // RFC 813, Цільова група Інтернет інженерії. 1982. Електронний ресурс. Режим доступу: <https://tools.ietf.org/html/rfc813>.
23. Menon A. Diagnosing Performance Overheads in the Xen Virtual Machine Environment [Текст] / A. Menon , J.R. Santos, Y. Turner, J. Janakiraman, W. Zwaenepoel. – May 3, 2005. Електронний ресурс. Режим доступу: <https://www.hpl.hp.com/techreports/2005/HPL-2005-80.pdf>.
24. Семёнушкин А.В. Тестирование пропускной способности сети. Електронний ресурс. Режим доступу: <http://semenushkin.ru/2010/07/01/тестирование-пропускной-способности/>.
25. Бейлі, Д.Г. "Паралельні тести NAS [Текст] / Д.Г. Бейлі, Е. Баржш, Дж. Бартон, Д. Браунінг, Р. Картер, Д. Дагум, П. А. Фатоугі, П. О. Фредріксон, Т. Ласінські, Р. С. Шрайбер, Н. Д. Симош, Б. Венкатакрішнан, С. К. Віратунга // " Міжнародний журнал суперкомп'ютерних додатків. – (осінь +1991), т. 5, № 3, С. 63-73.
26. Обзор некоторых пакетов измерения производительности кластерных систем. Електронний ресурс. Режим доступу: <http://www.ixbt.com/cpu/cluster-benchtheory.shtml>.
27. Netperf: мітки мережевої продуктивності. Редакція 2.0/ Компанія Гевлет-Пекерд. Електронний ресурс. Режим доступу: <http://www.netperf.org/netperf/training/Netperf.html>.

References

1. S. Iannucci, V. Cardellini, O. D. Barba, I. Banicescu. (2020) A hybrid model-free approach for the near-optimal intrusion response control of non-stationary systems. // Future Generation Computer Systems. – 2020, V 109, p. 111-124.
2. D. Black, P. Jones (2015) Differentiated Services and Real-Time Communication, Informational. RFC 7657, – Nov. 2015. – IETF, ISSN 2070-1721.
3. D. Black, P. Jones, Differentiated Services and Real-Time Communication, Informational RFC 7657, IETF, ISSN 2070-1721 (Nov. 2015).
4. R. Barik, M. Welzl A. Elmokashfi, T. Dreiholz S. Islam S. Gjessing (2019) On the utility of unregulated IP DiffServ Code Point (DSCP) usage by end systems. Performance Evaluation. – 2019, V. 135. – p. 102036.
5. Melnyk, V., Pekh, P., Melnyk, K., Bahnyuk, N., Zhyharevych O. (2016) The cross-domain communication mechanism construction and use for high-performance data computing. The East-European Journal of Advanced Technologies. Kharkiv. 1/9/79, 10-15. DOI: 10.15587/1729-4061.2016.60629.
6. Melnyk, V., Bahnyuk, N., Melnyk, K. (2015). The influence of high-performance sockets on the data computing intensity. Scientific Journal ScienceRise, Kharkiv, 6, 2(11), 38-48. DOI: 10.15587/2313-8416.2015.44380.
7. Melnyk, V. High production of java sockets (HPJS) for health clouds in science [Text] / V. Melnyk, K. Melnyk, O. Zhyharevych // Software engineering. National Aviation University. – 2015. – Vol. 19, № 3. – p. 36–40.
8. Melnyk, V. Significance of the socket programming for the laboratory with intensive data communications [Text] / V. Melnyk, P. Pekh, K. Melnyk, O. Zhyharevych // Ukrainian Scientific and Practical Internet Conference "Modern Methods, Information and Software for Management Systems of Organizational and Technological Complexes". (April 28, 2015). □ Lutsk, Lutsk NTU. / Scientific journal "Computer-integrated technologies: education, science, production." - 2015, №20. – p. 67–71.
9. Barham, P. Xen and the art of virtualization [Text] / P. Barham, B. Dragovich, K. Fraser and others. // Proceedings of the ACM Symposium on Operating Systems Principles (SOSP 2003), ACM Press, New York, USA: Bolton Landing, Lake George, October 2003. – p. 164–177.
10. Pratt, I. (2007). Xen Virtualization. Linux world 2005 Virtualization BOF Presentation.
11. Chisnall, David. (2007). The Definitive Guide to the Xen Hypervisor. Prentice Hall, 2-nd edition.
12. Menon, A., Cox, A. L., Zwaenepoel, W. (2006). Optimizing network virtualization in Xen. In 2006 USENIX Annual Technical Conference, Boston, Massachusetts, USA, 15–28.
13. Jian, W. XenLoop: Transparent high performance between virtual machines through a network loop [Text] / W. Jian, K. L. Wright, K. Gopalan // Boston, Massachusetts, USA, HPDC'08, 23–27 June 2008.
14. Kim, K., Kim, C., Jung, S.-I., Shin, H.-S., Kim, J.-S. (March 5–7, 2008). Inter-domain Socket Communications Supporting High Performance and Full Binary Compatibility on Xen.VEE'08, Seattle, Washington, USA.
15. Amdal's law. Electronic resource. Access mode: <http://home.wlu.edu/~whaley/t/classes/parallel/topics/amdahl.html>.
16. V. Melnyk, N. Bahnyuk, K. Melnyk, O. Zhyharevych, N. Panasyuk. Implementation of the simplified communication mechanism in the cloud of high performance computations. East-European journal of Enterprise Technologies. – Kharkiv (DOI: 10.15587/1729-4061.2017.98896). – 2017. – № 2/2/86. – p. 24–32.
17. Juan, J. L. V. High productivity of bypass between MVM for input/ output in virtual machines [Text] / J. L. V. Juan, B. Abali, D. K. Panda // USENIX annual technical conference Archive , 2006.
18. Hines, MR MemX: Support for large memory workloads in Xen for virtual machines [Text] / MR Hines, K. Gopalan // VTDC '07, Renault, Nevada, USA. November 12, 2007
19. Deshpande W. Memx: Virtualization of a cluster with a wide memory [Text] / Deshpande W., Van B., Huck S., Hines M., K. Gopalan // In ICPP'Ю: Proceedings of the 39th International Conference on Parallel Processing (2010), pp. 663–672.
20. Kim J.-S. Design and implementation of user level and socket level over virtual interface architecture [Text] / J.-S. Kim, K. Kim, S.-I. Jung, S. Ha // Parallel Obch. : Practice. Expert, 15 (7–8). – 2003. – p. 727–749.
21. Sin S. VOP: socket-interface for VO [Text] / S. Sin, D. Kim, E. Lim, S. Jung // In the Internet and multimedia systems and applications, 2004.
22. Clark D.D. Window and TCP recognition strategy [Text] / DD Clark // RFC 813, Target group of Internet engineering. 1982

23. Menon A. Diagnosing Performance Overheads in the Xen Virtual Machine Environment [Текст] / A. Menon , J.R. Santos, Y. Turner, J. Janakiraman, W. Zwaenepoel. – May 3, 2005. Electron resource. Access mode: <https://www.hpl.hp.com/techreports/2005/HPL-2005-80.pdf>.
24. Semenushkin A. Network bandwidth testing. Electronic resource. Access mode: /http://semenushkin.ru/2010/07/01/testirovanie-propusknoy-posobnosti/.
25. Bailey, D.G. "Parallel NAS tests [Text] / DG Bailey, E. Barges, J. Barton, D. Browning, R. Carter, D. Dagum, PA Fatougi, P. O. Fredrickson, T. Lasinski, RS Schreiber, ND Simon, B. Venkatakrishnan, SK Viratunga // "International Journal of Supercomputer Applications. - (autumn +1991), vol. 5, № 3, pp. 63-73.
26. Overview of some cluster performance measurement packages. Electronic resource. Access mode: <http://www.ixbt.com/cpu/cluster-benchtheory.shtml>.
27. Netperf: network performance labels. Edition 2.0/ Hewlett-Packerd Company. Electronic resource. Access mode: <http://www.netperf.org/netperf/training/Netperf.html>.

DOI: 10.36910/6775-2524-0560-2020-39-29

УДК: 004.932

Міскевич Оксана Іванівна, асистент

<https://orcid.org/0000-0002-5009-2391>

Багнюк Наталія Володимирівна, к.т.н., доцент

<https://orcid.org/0000-0002-7120-5455>

Хрестинець Наталія Анатоліївна, ст.викладач

<https://orcid.org/0000-0002-4836-7632>

Марчевська Ольга Романівна, студентка

Луцький національний технічний університет

АВТОМАТИЗАЦІЯ ВИЯВЛЕННЯ ДЕФЕКТІВ МЕТОДАМИ МАШИННОГО НАВЧАННЯ

Міскевич О. І., Багнюк Н. В., Хрестинець Н.А., Марчевська О.Р. Автоматизація виявлення дефектної продукції методами машинного навчання. Представлено основні методи машинного навчання для розпізнавання дефектів у сферах виробництва різнопланової продукції. Детально розглянуто застосування підходів глибокого навчання і комп'ютерного бачення для розпізнавання недоліків продукції апаратного забезпечення, одиночних та комплексних алгоритмів машинного навчання для контролю якості програмного забезпечення.

Ключові слова: автоматизація виробництва, виявлення дефектів, CNN, Machine Learning, Deep Learning, Computer Vision, Quality Assurance.

Міскевич О. И., Багнюк Н. В., Хрестинец Н.А., Марчевская О.Р. Автоматизация выявления дефектной продукции методами машинного обучения. Представлены основные методы машинного обучения для распознавания дефектов в областях производства разнородной продукции. Подробно рассмотрено применение подходов глубокого обучения и компьютерного видения для распознавания недостатков продукции аппаратного обеспечения, одиночных и комплексных алгоритмов машинного обучения для контроля качества программного обеспечения.

Ключевые слова: автоматизация производства, выявления дефектов, CNN, Machine Learning, Deep Learning, Computer Vision, Quality Assurance.

Miskevych O, Bahniuk N., Khrystinets N., Marchevska O. Automation of defective products detection by machine learning methods. The basic methods of machine learning for the detection of defects in the fields of production of various products are presented. The use of deep learning and computer vision approaches to identify hardware faults, single and complex machine learning algorithms for quality control of software are discussed in detail.

Keywords: production automation, defect detection, CNN, Machine Learning, Deep Learning, Computer Vision, Quality Assurance.

Постановка наукової проблеми: Залежно від сфери, виробничий процес (далі ВП) може бути складним та супроводжуватись появою помилок. Хоча ВП відрізняється один від одного залежно від сфери впровадження, при виконанні кожного з них прагнуть досягнення подібних цілей, що стосуються загальної ефективності обладнання (overall equipment effectiveness – OEE) та загальної ефективності лінії (overall line efficiency – OLE). OEE розраховується як добуток між показниками наявності, продуктивності та якості. А значення OLE – середньозважене OEE кожної машини на виробничій лінії, а отже важливість якості як основної виробничої метрики – незаперечна.

Аналіз досліджень. Серед всіх сфер застосування машинного навчання (Machine Learning, далі ML) для контролю якості та надійності виділяють дві найважливіші [1].

Прогнозуючий контроль якості. Контроль якості – вбудований у виробництво процес, де дефектні вироби відсіюються від решти продукції якомога раніше. Під час того, як контроль якості забезпечує виявлення браку та попереджає появу дефектного продукту на ринку, сам процес виявлення головної причини, що призвела до появи дефектних виробів вимагає великих затрат часу і часто вимагає залучення багатьох дисциплін – оптимізації процесу, гарантій якості, механіки та електроніки тощо.

Наприклад, харчове виробництво вимагає жорстких мір при контролі якості на кожному кроці ВП для гарантування безпечності їжі. Good Manufacturing Practices (GMP), Hazard Analysis Critical Control Point (НАССР), Hazard Analysis Risk-based Preventive Controls (HARPC), Codex Alimentarius, та ISO 22000 – лише декілька регуляторів та стандартів, розроблених для цієї мети.

Із впровадженням ML, обладнання та дані про виріб можуть повністю відслідковуватись впродовж усього процесу виготовлення для передбачення втрат якості до того, як такі з'являться. Команди управління та якості попереджаються про точні причини очікуваних помилок, що і є основою Quality 4.0 [2].

Інтегрування машинного навчання в процес управління якістю часто відносять до прогнозування якості, що знижує власне проблеми якості та відходи, скорочує виробничі затрати,

мінімізує відкликання продукції та захищає репутацію бренда. Повідомляють також, що ML може підвищити виявлення браку до 90% при цьому зменшуючи час на знаходження причини його появи від днів до хвилин.

Прогнозуюче технічне обслуговування. Відповідно до останніх досліджень, незаплановані простої промислових машин у виробництві оцінюють у \$50 мільярдів щорічно. А 42% простоїв становлять відмови устаткування [3]. Прогнозуюче технічне обслуговування – моніторинг стану обладнання та передбачення необхідності обслуговування на основі ведучих показників появи дефектів.

Прогресуючі техніки ML все більше застосовують для оптимізації результатів прогнозуючого обслуговування. Такі техніки включають моделювання відповідного процесу розробки на виробничій лінії, а тоді застосування найбільш умісного алгоритму ML в контексті ВП та специфіки продукції, що виготовляється. При прогнозуючому обслуговуванні алгоритми ML не повинні отримувати інформацію для навчання з попередньо встановленим пороговим значенням, а їх необхідно «натренувати» для розпізнавання шаблонів та аномалій на прикладах звичайної поведінки та історії помилок. Хоча перед першим застосуванням алгоритми все одно тренують, проте з часом вони самооптимізуються з мінімальним втручанням людини при цьому. Завдяки цьому фабрики передбачають та попереджають простої здійснюючи правильні та вчасні дії.

Автори статті [4] П. Дека та Р. Міттал описують автоматизацію у виробництві. Як відомо, на сьогоднішній день підвищення рівня автоматизації у виробництві також залежить від автоматизації перевірки якості матеріалів з невеликим втручанням людини. Основний напрям – досягнення рівня точності людини або більша якість автоматизованої перевірки. Такий напрям надає користувачу можливість використання деяких аспектів DL та відображає потребу в оптимізації всього циклу (алгоритми, структура виводу, апаратні прискорювачі) для отримання оптимальної продуктивності.

Виклад основного матеріалу й обґрунтування отриманих результатів. Для досягнення виробничих стандартів, інспектори з якості на виробництвах перевіряють якість зазвичай після виготовлення продукції – це ручна робота, що вимагає великих затрат часу, а відхилений виріб призводить до підвищення втрат заводських потужностей, виробничих матеріалів та праці. Відповідно до сучасного тренду штучного інтелекту, заводи та фабрики шукають шляхи впровадження глибокого навчання на основі технологій комп'ютерного бачення (Computer Vision, далі CV) у сам цикл виробництва для автоматизації перевірки якості матеріалів.

Глибоке навчання (Deep learning, далі DL) – сфера вивчення глибоких структур та неструктурованих відображень даних, зростаючий напрям у ML для отримання кращих результатів, при даних великих розмірів та складності.

Згорткові нейронні мережі (Convolution neural Network, далі CNN) – клас глибоких нейронних мереж, що зазвичай використовують для аналізу зображень. Згорткові шари додають згорткові операції для початкової передачі результату до наступного шару. Згорткові операції є рішенням проблеми великої кількості ознак, адже ці операції скорочують число вільних атрибутів, дозволяючи мережі бути глибшою при меншій кількості параметрів.

CNN складають різні типи шарів:

- Згортковий (створює карту атрибутів для передбачення класових ймовірностей для кожного атрибута, застосовуючи фільтр, що сканує все зображення по декілька пікселів за раз);
- Об'єднуючий (зменшення вибірки) (зменшує масштаби інформації, згенерованої згортковим шаром для кожного атрибута та утримує найбільш суттєву інформацію);
- Повністю з'єднаний вхідний ("сплющує" виходи згенеровані попереднім шаром для перетворення їх в один вектор, що може бути використаний як вхідний для наступного шару);
- Повністю з'єднаний (додає ваги на входи, що створені за допомогою аналізу ознак для прогнозування точної мітки);
- Повністю з'єднаний вихідний (створює фінальні ймовірності для класифікації екземпляра).

Є дві головних переваги використання згорткових шарів порівняно з повністю з'єднаними – розподіл параметрів та розрідження зв'язків [5]. Згорткові нейронні мережі здійснюють пошук шаблонів у зображенні. Зображення згортається в меншу матрицю і ця згортка використовується для пошуку шаблонів у зображенні. Декілька перших шарів можуть визначати лінії, кути, краї тощо. Далі ці шаблони передаються в глибші шари нейронної мережі для розпізнавання більш складних ознак. Така властивість CNN чудово спрацьовує при визначенні об'єктів на зображенні.

Наприклад, для задачі розпізнавання пошкоджень на плиті певного матеріалу (рис. 1) застосовано два згорткових шари, повністю зв'язаний (ReLU) та об'єднуючий (Max Pooling).

Очевидно, що архітектура CNN фокусується на пошкоджених блоках та їх поширенні по всій площині:

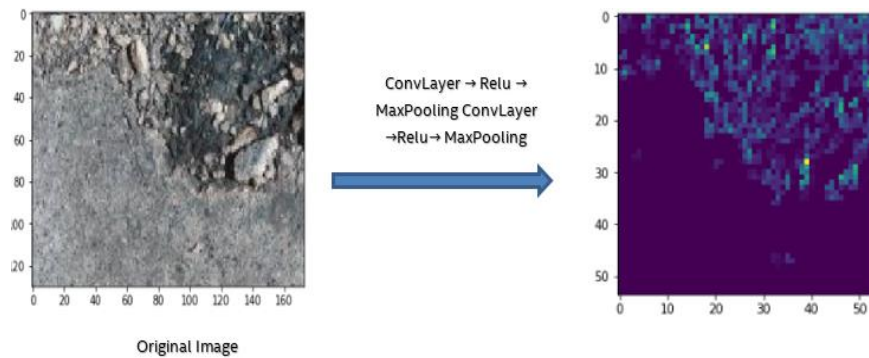


Рисунок 1 - Розпізнавання пошкоджень матеріалу за допомогою CNN

Визначення матеріалів поганої якості у виготовленні апаратного забезпечення – це процес схильний до помилок та вимагає великих затрат часу, що також спричиняє велику кількість помилково позитивних значень (false positives, далі FP), тобто віднесення браку до категорії якісної продукції. Тому у статті [2] пропонують декілька підходів вирішення цієї проблеми.

Перший з них полягає в поєднанні чистого CV для виявлення ділянок зацікавлення (region of interest, далі ROI) із вхідного зображення та чистого DL для визначення дефектів на цих ділянках. В такому порядку дії виконують через те, що експериментальним шляхом було визначено, що продуктивність DL суттєво підвищується за умови фокусування нейронної мережі лише на певних ділянках замість повної площині (рис. 2).

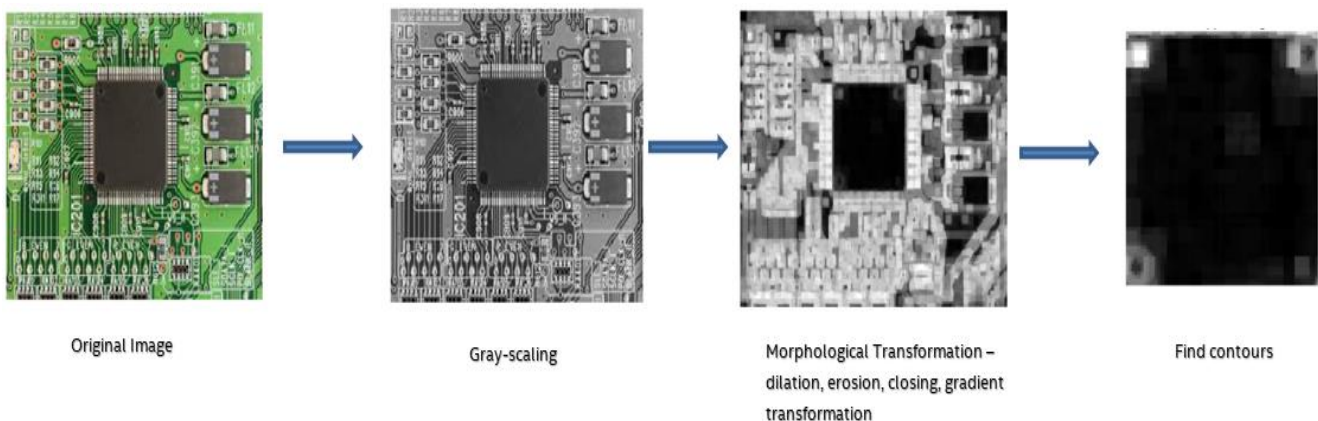


Рисунок 2 - Фокусування нейронної мережі

Надалі виявлення дефектів відбувається за допомогою глибоких нейронних мереж використовуючи перевірені топології Inception Net (Google Net), Res Net, Dense Net (рис. 3).

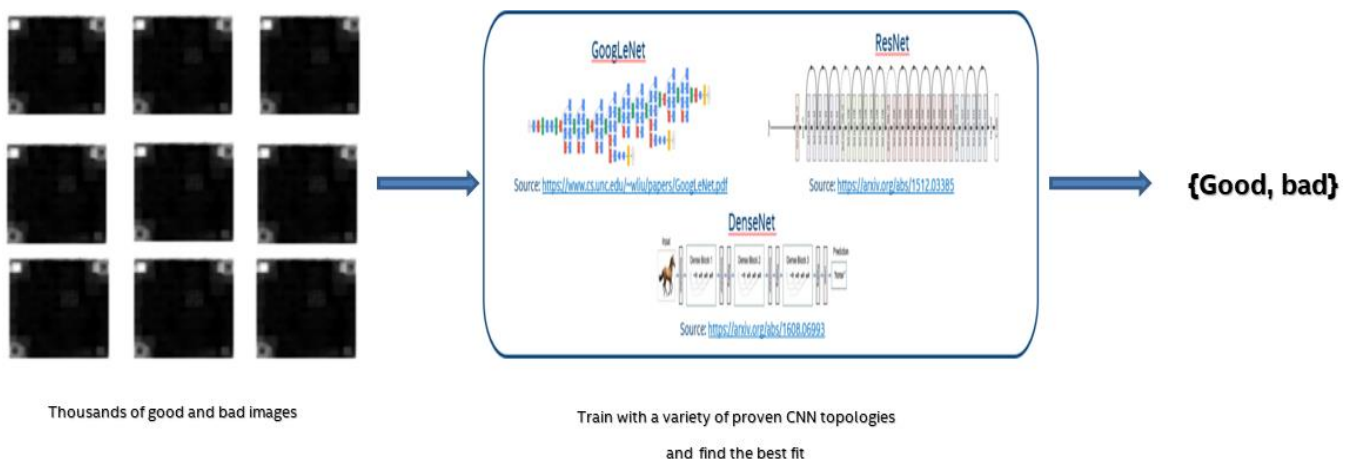


Рисунок 3 - Процес двокрокового розпізнавання дефектів

Другий підхід полягає в критиці до першого. Під час фокусування на ROI необхідно переписувати код незалежно від того чи є зміни щодо типу виробу, мікросхеми, чипу (як в даному прикладі), налаштувань та напрямів камери тощо, тобто масштабування відсутнє. Вирішення: будують закінчену двокрокову архітектуру DL.

На першому кроці замість технік CV використовують DL для передбачення самих ROI. Вручну створюють маркований набір даних за допомогою обмежуючого інструменту, а тоді тренують архітектуру DL для передбачення ROI. Один з мінусів такої техніки є, те що маркований набір даних має бути чітким та охоплювати достатню кількість різних типів продуктів для того, щоб глибока нейронна мережа була здатна добре узагальнювати отримуючи нові зображення (рис. 4).

Оптимальна модель – це завжди компроміс між метриками FPR (false positive rate) та FNR (false negative rate) або Precision та Recall. Для даного випадку головним є успішне виявлення дефектів, а отже модель оптимізують у напрямку нижчого значення FNR (Вищий значення метрики Recall).

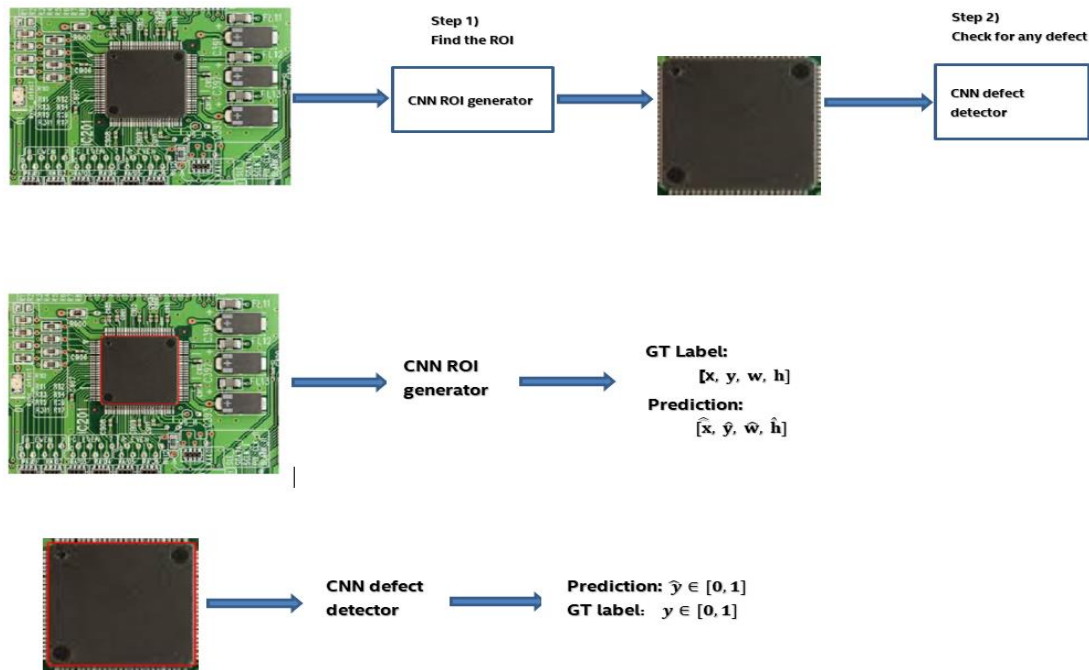


Рисунок 4 - Процес двокрокового розпізнавання дефектів з передбаченням ROI

Окрім перевірки якості у виготовленні матеріальних виробів, ML також широко застосовується для оцінки якості та надійності при розробці програмного забезпечення [6].

Баг означає неочікувану поведінку системи. Така поведінка визначається під час тестування та маркується, як дефект (рис. 5). Виявлення дефектів та виправлення наслідків їх появи – вимагають затрат цінного часу розробки. Досліджено, що лише невелика частина модулів містять більшість недоліків програмного забезпечення [7]. Отже, вчасне виявлення таких дефектів полегшує розподіл ресурсів тестування та дозволяє розробнику покращити архітектуру системи з а допомогою визначення сегментів високого ризику [8].

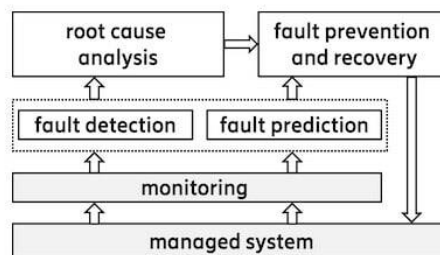


Рисунок 5 - Компоненти системи управління дефектами

Згідно з дослідженнями [6] застосовують різні підходи ML для виявлення багів програмного забезпечення. Для отримання результатів застосовано 3 типи метрик та 5 різних наборів даних. Було доведено ефективність SVM, MLP та технік бегінгу для такого типу задач, що дають в середньому 90% точності при виявленні дефектів. Для вибору найбільш відповідного методу для передбачення багів дослідники радять звертати увагу на різноманітні фактори, такі як: тип набору даних, область проблеми, неоднозначність даних та природу самого проекту.

Також застосовують комплексні підходи ML при тестуванні програмного забезпечення.

Test Scenario Mining. Метод, що дозволяє виконувати тестові завдання, що базуються на тих ділянках, що мають потенційно високу ймовірність помилковості. Це сучасний підхід, що заснований на алгоритмі тестування на основі ризику, який є вирішуючим фактором для виконання тесту. Така техніка підвищує ефективність тестування, а також скорочує затрати, адже дефекти визначаються на ранніх стадіях.

Test Suite Optimizer (Оптимізатор набору тестів). ML використовується також для оптимізації набору тестів. Зростаючий репозиторій тестів з великою кількістю тестової інформації створює надлишковість даних, яку можливо скоротити на 15% за допомогою оптимізатора набору тестів. Це зменшує кількість необхідної роботи за рахунок скорочення регресивного набору для виконання тестових задач у вікнах за фіксований проміжок часу. Структуроване тестування на основі ризику перевіряє додаток, що тестується на всі можливі помилки та ризики.

Analysis of impact (Аналіз впливу). Метод визначає вплив меншої участі експерта, відсутність діаграм UML, встановлення взаємозв'язків між елементами в документі. Машинне навчання пропонує такі переваги як полегшену передачу знань, що дозволена на фазі підтримки, забезпечує полегшений аналіз, ефективне визначення дефектів, що також включає пріоритезацію багів на основі частоти їх появи, серйозності, ризику.

Customer Sentiment Analytics (Аналіз відгуків користувачів). Даний метод використовується для отримання відгуків кінцевого користувача, його бачення та потреб. Такий підхід допомагає знайти точні проблеми, що направляють поведінку користувача. Також це забезпечує навчання в режимі реального часу для отримання постійного зворотного зв'язку, що одночасно покращує менеджмент ризиками.

Висновки та перспективи подальшого дослідження. З використанням DL на основі CV досягають точності діагностування браку у виробництві рівня людського спостерігача в обох підходах CV + DL та DL + DL. Таке рішення є унікальним, адже тут DL використовується не лише для класифікації, а також і для відображення пошкоджених ділянок за допомогою теплових карт. Хоча людський фактор не може бути цілком відкинутий, проте втручання експерта можна суттєво мінімізувати. Також за допомогою ML можливо покращити якість програмного забезпечення, а його впровадження в процес розробки дозволяє розробляти кращі системи на основі відгуків та досвіду користувачів. Підтримка високої якості у створенні продукції – стратегічна ціль для виробників, адже це прямим чином впливає на репутацію та дохід всієї компанії. Інтеграція технологій ML у процес управління якістю може мінімізувати недоліки продукції та скоротити виробничі затрати. Більше того, так як ML – система самонавчання, з часом вона гарантовано покращуватиме власні результати. Подальші дослідження можуть бути спрямовані на застосування Quality 4.0 для розглянутих вище задач.

Список бібліографічного опису

1. Oren Erza, How Machine Learning Slashes Quality Control Costs in Manufacturing [Електронний ресурс]. - Режим доступу: <https://blog.seebo.com/machine-learning-quality-control/>.
2. D. Juran, Quality 4.0: The Future of Quality? [Електронний ресурс]. - Режим доступу: <https://www.juran.com/blog/quality-4-0-the-future-of-quality/>.
3. Graham Immerman, The actual cost of downtime in the manufacturing industry [Електронний ресурс]. - Режим доступу: <https://iiot-world.com/predictive-maintenance/the-actual-cost-of-downtime-in-the-manufacturing-industry/>
4. Partha Deka, Quality inspection in manufacturing using deep learning based computer vision [Електронний ресурс]. - Режим доступу: <https://towardsdatascience.com/quality-inspection-in-manufacturing-using-deep-learning-based-computer-vision-daa3f8f74f45>.
5. Convolutional Neural Network Architecture: Forging Pathways to the Future [Електронний ресурс]. - Режим доступу: <https://missinglink.ai/guides/convolutional-neural-networks/convolutional-neural-network-architecture-forging-pathways-future/>
6. S. Aleem, L. Fernando Capretz, A. Faheem, Benchmarking Machine Learning Techniques for Software Defect Detection [Електронний ресурс]. - Режим доступу: https://www.researchgate.net/publication/279252183_Benchmarking_Machine_Learning_Techniques_for_Software_Defect_Detection.
7. N. E. Fenton & N. Ohlsson (2000) "Quantitative analysis of faults and failures in a complex software system", IEEE Transactions on Software Engineering, с. 797- 4.
8. T. Menzies, J. Greenwald & A. Frank (2007) "Data mining static code attributes to learn defect predictors", IEEE Transaction Software Engineering., с. 2-13.

9. Гринюк С.В., Бортник К.Я., Міскевич О.І., Паливода Д.І. Огляд інструментальних засобів для створення ігор під ОС Android. / Комп'ютерно-інтегровані технології: освіта, наука, виробництво. №35., ст. 124-128, 2019р.
10. Міскевич О.І., Сичов Д.І., Христинець А.О. Про модернізацію локально-обчислювальної мережі на ПрАТ "Волинбленерго" на основі GRE-tunnel з використанням шифрування IPSec. / Комп'ютерно-інтегровані технології: освіта, наука, виробництво. №30-31., ст. 100-103, 2018р.
11. Міскевич О.І., Войтович І.В. Формати зображень та доцільність їх використання в сучасному світі. / Комп'ютерно-інтегровані технології: освіта, наука, виробництво. №38., ст. 85-90, 2018р.
12. Христинець Н.А. Дослідження методу вібраційної сегрегації у формуванні градієнтної структури порошкових матеріалів. // Рудь В.Д., Христинець Н.А. // [Матеріали Міжнародна конференція молодих науковців «Сучасні технології в механіці», – 21-23 квітня 2016 року], м. Хмельницький. /Вісник Хмельницького національного університету, 2016. – Вип.3. – С. 34-41.
13. Христинець Н.А., Рудь В.Д. Стохастичні методи моделювання процесів вібраційного змішування у сипких середовищах. Міжвузівський збірник "Комп'ютерно-інтегровані технології: освіта, наука, виробництво" – Луцьк: Видавництво ЛНТУ. – Вип. 7. – 2011. – 96–98.
14. Христинець Н.А., Рудь В.Д., Колядинський М.І. Модель поведінки часток сипкого середовища під дією вібраційної сегрегації. Міжвузівський збірник "Комп'ютерно-інтегровані технології: освіта, наука, виробництво" – Луцьк: Видавництво ЛНТУ. – Вип. 7. – 2011. – 99–103.

References

1. Oren Ezra, How Machine Learning Slashes Quality Control Costs in Manufacturing [Електронний ресурс]. - Режим доступу: <https://blog.seebo.com/machine-learning-quality-control/>.
2. D. Juran, Quality 4.0: The Future of Quality? [Електронний ресурс]. - Режим доступу: <https://www.juran.com/blog/quality-4-0-the-future-of-quality/>.
3. Graham Immerman, The actual cost of downtime in the manufacturing industry [Електронний ресурс]. - Режим доступу: <https://www.machinemetrics.com/blog/the-real-cost-of-downtime-in-manufacturing>.
4. Partha Deka, Quality inspection in manufacturing using deep learning based computer vision [Електронний ресурс]. - Режим доступу: <https://towardsdatascience.com/quality-inspection-in-manufacturing-using-deep-learning-based-computer-vision-daa3f8f74f45>.
5. Convolutional Neural Network Architecture: Forging Pathways to the Future [Електронний ресурс]. - Режим доступу: <https://missinglink.ai/guides/convolutional-neural-networks/convolutional-neural-network-architecture-forging-pathways-future/>
6. International Journal of Software Engineering & Applications (IJSEA), Vol.6, No.3, May 2015
7. S. Aleem, L. Fernando Capretz, A. Faheem, Benchmarking Machine Learning Techniques for Software Defect Detection [Електронний ресурс] - Режим доступу: https://www.researchgate.net/publication/279252183_Benchmarking_Machine_Learning_Techniques_for_Software_Defect_Detection
8. N. E. Fenton & N. Ohlsson (2000) "Quantitative analysis of faults and failures in a complex software system", IEEE Transactions on Software Engineering, pp. 797-814.
9. T. Menzies, J. Greenwald, A. Frank (2007) "Data mining static code attributes to learn defect predictors", IEEE Transaction Software Engineering, pp. 2-13.
10. 6. S.V. Grynyuk, K.Ya. Bortnik, O.I. Miskevych, D.I. Palivoda An overview of tools for creating games on Android. / Computer-integrated technologies: education, science, production. No. 35, Art. 124-128, 2019.
11. 7. Miskevych O.I., Sychev D.I., Khrystynets N.A. About modernization of the local area network on PJSC "Volynoblenergo" based on GRE-tunnel using IPSec encryption. / Computer-integrated technologies: education, science, production. №30-31., Art. 100-103, 2018
12. Miskevych O., Ilya Voytovich. Image formats and the appropriateness of their use in the modern World. / Computer-integrated technologies: education, science, production. №38., Art. 85-90, 2018
13. N. A. Khrystynets, A. A. Sakhnyuk, E. A. Sviridyuk, O. I. Miskevich. Use of bem-blocks when creating a site. / Computer-integrated technologies: education, science, production. №35., Art. 206-210, 2019
14. N. A. Khrystynets, Rud V.D Stochastic methods for modeling vibration mixing processes in bulk media./ Computer-integrated technologies: education, science, production. №7., Art. 96-98, 2011
15. N. A. Khrystynets, Rud V.D, Kolyadinsky M.I. Model of behavior of particles of bulk medium under the action of vibration segregation./ Computer-integrated technologies: education, science, production. №7., Art. 99-103, 2011

DOI: 10.36910/6775-2524-0560-2020-39-30

УДК: 004.415.2

Муляр Вадим Петрович, к. пед. н., доцент

<https://orcid.org/0000-0003-4774-3947>

Східноєвропейський національний університет імені Лесі Українки

РОЗРОБКА JAVAFX-ДОДАТКІВ ІЗ ВИКОРИСТАННЯМ SCENE BUILDER

Муляр В. П. Розробка JavaFX-додатків із використанням Scene Builder. У статті розкрито особливості створення додатків на основі технології JavaFX. Доведено високу ефективність середовища Scene Builder у процесі розробки графічного інтерфейсу користувача. На прикладі створення діалогового середовища розглянуто основні етапи проектування JavaFX-додатків в інтегрованому середовищі розробки NetBeans із використанням Scene Builder.

Ключові слова: платформа JavaFX, інтегроване середовище розробки NetBeans, конструктор макетів Scene Builder, графічний інтерфейс користувача, компонування.

Муляр В. П. Разработка JavaFX-приложений с использованием Scene Builder. В статье раскрыты особенности создания приложений на основе технологии JavaFX. Доказана высокая эффективность среды Scene Builder в процессе разработки графического интерфейса пользователя. На примере создания диалоговой среды рассмотрены основные этапы проектирования JavaFX-приложений в интегрированной среде разработки NetBeans с использованием Scene Builder.

Ключевые слова: платформа JavaFX, интегрированная среда разработки NetBeans, конструктор макетов Scene Builder, графический интерфейс пользователя, компоновка.

Muliar V. P. Developing JavaFX applications using Scene Builder. The features of JavaFX based application development are discussed in the article. High efficiency of Scene Builder environment in the process of developing a graphical user interface has been proven. The example of creating a dialog discusses the basic stages of designing JavaFX applications in an integrated NetBeans development environment using Scene Builder.

Keywords: JavaFX platform, NetBeans integrated development environment, Scene Builder layout designer, graphical user interface, layout.

Постановка наукової проблеми. Створення сучасного, ефективного та повнофункціонального інструментарію для розробки клієнтських додатків на основі Java неможливо уявити без технології JavaFX. За її допомогою можна створювати програми як для різних операційних систем (Windows, MacOS, Linux), так і для різноманітних пристроїв (десктопів, смартфонів, планшетів, вбудованих пристроїв, ТБ). Однак тепер JavaFX розширює свою підтримку на Android і iOS за допомогою таких технологій, як JavaFX Ports і Gluon Mobile. Остання є платформою для написання, компіляції та підготовки додатків JavaFX для розгортання на iOS і Android. Для кінцевого користувача додаток виглядає і поводить себе точно так само, як і нативний (native) додаток [12].

JavaFX дозволяє створювати програми з багатою насиченою графікою завдяки використанню апаратного прискорення графіки і можливостей графічного процесора. JavaFX має великий набір елементів управління та широкі можливості для роботи з мультимедіа, двомірною і тримірною графікою. Характерним для JavaFX є декларативний спосіб опису інтерфейсу за допомогою мови розмітки FXML, можливість стилізації інтерфейсу за допомогою CSS і багато іншого.

На даний час для розробки програмного забезпечення мовою Java потрібно встановити на комп'ютері декілька основних програмних засобів. Першим є JDK (Java Development Kit – комплект для Java-розробки), який, як правило, містить у собі ще й віртуальну машину Java (пакет JRE – Java Runtime Environment). Його можна встановити з офіційного сайту Oracle: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. Другим засобом є JavaFX SDK, який завантажують за адресою <https://gluonhq.com/products/javafx/>. Третім є інтегроване середовище розробника Java-програм (NetBeans), яке завантажують безпосередньо з сайту NetBeans (<https://netbeans.org>). Останнім засобом є візуальний конструктор графічного інтерфейсу користувача Scene Builder, який завантажують за адресою <https://gluonhq.com/products/scene-builder/>.

Аналіз досліджень. Огляд сучасних технологій створення RIA-додатків (Rich Internet Applications) здійснено у статті К. Афанасьєва та К. Лебедева [1]. Розгляду архітектури платформи JavaFX 2.0, її основним компонентам графічного інтерфейсу користувача, використанню CSS-стилів, створенню візуальних ефектів, трансформації й анімації зображень, використанню компонентів JavaFX NetBeans, мові FXML присвячена робота Т. Машніна [4]. Створенню насичених інтернет-додатків за допомогою JavaFX присвячено дослідження В. Герасимова та В. Левицької [2], Ю. Парфенова та В. Федорченко [6]. Особливості побудови графічного контенту додатків із використанням JavaFX і даних, взятих із баз даних, розкрито в дослідженні В. Карашецького [3]. На думку дослідників, JavaFX є інструментарієм наступного покоління для створення графічного

інтерфейсу користувача. Технологія забезпечує кросплатформні додатки з графічним інтерфейсом такими складними функціями як плавна анімація, веб-представлення, відтворення аудіо та відео, стилі на основі CSS [7; 8; 11].

Формулювання цілей статті. Мета статті – розкрити особливості створення JavaFX-додатків із використанням Scene Builder.

Виклад основного матеріалу й обґрунтування отриманих результатів. JavaFX-технології представлені екземпляром класу GUI-компонента. Компоненти графічного інтерфейсу користувача (графічний інтерфейс користувача, GUI) JavaFX-додатка створюють сцену, логічна структура якої описується графом сцени. Відображення GUI-інтерфейсу JavaFX-додатка включає в себе графічне представлення графа зі сценами. Графом сцени є структура даних, колекція вузлів (вузол) дерева, яка використовує логічну структуру сцен. Сцена – це скомпонований у робочих областях набір моделей та об'єктів, які викликають різні ефекти, наприклад, джерело світла та камера, які створюють ефекти освітленості та перспективи. Під моделлю розуміють опис або набір даних, що представляє форму об'єкта. Моделі всередині сцени характеризуються розміром і взаємним розташуванням. Сцена має знаходитися на підмостках, які є вікном верхнього рівня, якщо програма виконується на робочому столі операційної системи, або прямокутною областю, якщо програма виконується в вигляді аплета [4, с. 16].

Для забезпечення гнучкого і динамічного розміщення елементів управління в графі сцени JavaFX-додатку використовуються контейнери схем компоновання або панелі. JavaFX Layout API включає в себе наступні контейнерні класи, які автоматизують загальні моделі схем компоновання.

Клас `BorderPane` розміщує вузли його контенту у верхній, нижній, правій, лівій, або центральній області.

Клас `HBox` розміщує вузли його контенту горизонтально в один рядок.

Клас `VBox` розміщує вузли його контенту вертикально в один стовпець.

Клас `StackPane` розміщує вузли його контенту в стек.

Клас `GridPane` дозволяє розробнику створити гнучку сітку з рядків і стовпців, в яких розміщуються вузли контенту.

Клас `FlowPane` розміщує вузли його контенту в горизонтальний або вертикальний «потік», огинаючи зазначені межі по ширині або висоті.

Клас `TilePane` розміщує вузли його контенту в комітках однакового розміру.

Клас `AnchorPane` дозволяє розробникам створювати вузли-якорі для прив'язки до верхньої, нижньої або лівої сторони, або в центрі макета.

Для досягнення бажаної структури розташування, різні контейнери можуть бути вкладені.

У JavaFX підтримуються різні способи компоновання елементів управління графічного інтерфейсу. Проте, основним засобом розробки візуального інтерфейсу користувача є JavaFX Scene Builder, оскільки:

– інтерфейс перетягування дозволяє швидко створити макет інтерфейсу користувача без необхідності запису вихідного коду;

– можна додавати, комбінувати та редагувати елементи керування інтерфейсом JavaFX у свій макет за допомогою бібліотеки елементів управління інтерфейсом та панелі вмісту;

– інтеграція з будь-яким Java IDE є простою, оскільки це окремий інструмент розробки;

– автоматичне генерування коду FXML відбувається під час створення та зміни макета інтерфейсу користувача: створений код FXML зберігається в окремому файлі;

– функції редагування в реальному часі та попереднього перегляду дозволяють швидко візуалізувати зміни макета інтерфейсу без необхідності компіляції;

– отримуємо доступ до повної бібліотеки управління інтерфейсом JavaFX: підтримка CSS дозволяє гнучко керувати зовнішнім виглядом інтерфейсу програми.

За умовчанням головне вікно JavaFX Scene Builder містить наступні розділи (рис. 1).

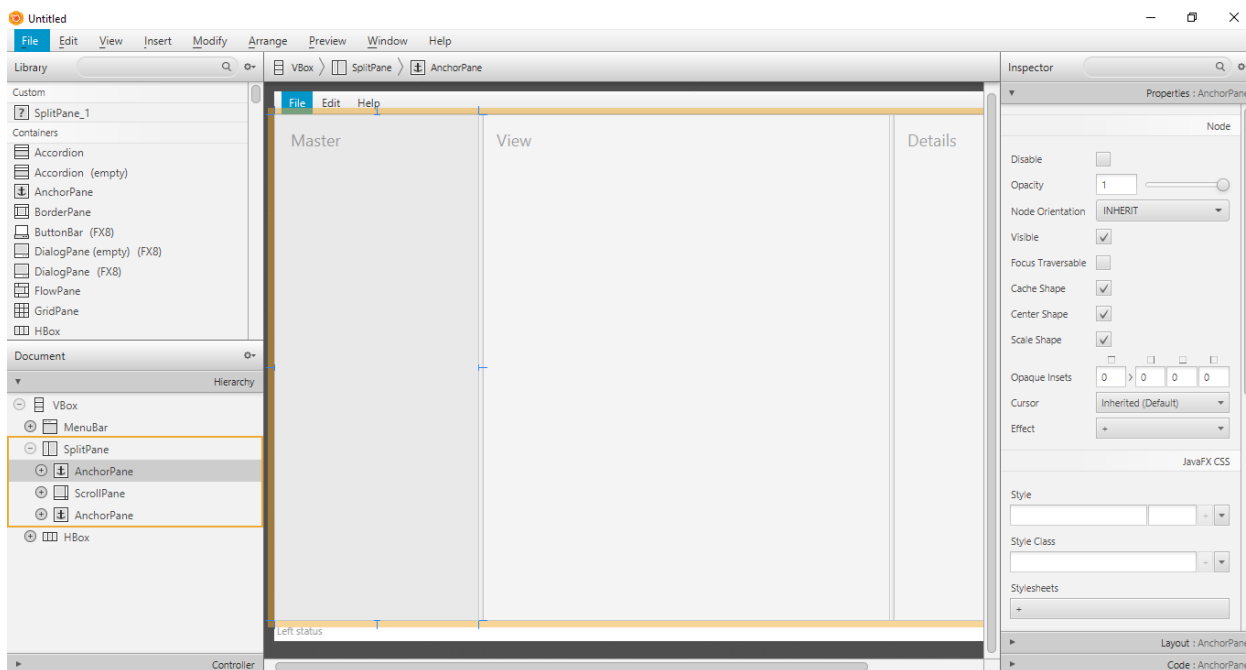


Рис. 1. Головне вікно Scene Builder

Рядок меню: надає доступ до меню команд, доступних у JavaFX Scene Builder.

Вибір та панель повідомлень: відображає шлях до обраного елемента. Він також відображає повідомлення про помилки або статус.

Панель вмісту: контейнер сцени для елементів інтерфейсу, що складають макет FXML. За умовчанням новий файл FXML, відкритий у JavaFX Scene Builder, містить кореневий (верхній) контейнер AnchorPane.

Панель бібліотеки: містить доступні елементи інтерфейсу або елементи управління JavaFX, які можна використовувати для створення макета FXML. Елементи інтерфейсу вибирають на цій панелі та додають їх на панель «Зміст» або «Ієрархія».

Панель ієрархії: відображає подання з дерева на макет FXML, який будують на панелі «Зміст». Елементи, яких не видно на панелі «Зміст», можна поставити у фокус, вибравши його на панелі «Ієрархія».

Панель інспектора: містить розділи Властивості, Макет та Код. Розділи «Властивості та макет» допомагають керувати властивостями вибраного елемента інтерфейсу на панелі «Зміст» або на панелі «Ієрархія». Розділ «Код» дозволяє визначити ім'я об'єкта (fx: id), а також для кожного компонента задати реакцію на події.

Панель «Інспектор» також містить текстове поле пошуку, яке дозволяє виділити конкретні властивості, які можна змінити.

Наступна панель відображається в головному вікні, коли в головному меню вибрати пункт Перегляд, а потім Показати CSS Analyzer.

Панель аналізатора CSS: дозволяє вивчити усі властивості CSS, доступні для компонента JavaFX на макеті FXML, та допомагає створити правила CSS [10].

Файл FXML можна об'єднати з проектом Java наступним чином:

```
public void start(Stage stage) {  
    try {  
        Parent root = FXMLLoader.load(getClass().getResource("dialog.fxml"));  
        stage.setScene(new Scene(root));  
        stage.show();  
    } catch (IOException ex) {  
        ex.printStackTrace();  
        System.exit(0);  
    }  
}
```

Розглянемо основні можливості JavaFX як засобу розробки графічного інтерфейсу користувача на прикладі лабораторного практикуму з комп'ютерного моделювання фізичних процесів і явищ [5].

Лабораторна робота Створення діалогового середовища

Мета: ознайомитись із основними етапами розробки *JavaFX*-додатків з використанням конструктора макетів *Scene Builder* під час створення проекту й головної форми графічних побудов.

Створення головної форми

1. Відкрийте середовище *NetBeans* із підтримкою платформи *JavaFX 2.0*. У меню *Файл* виберіть *Створити проект | JavaFX | JavaFX FXML Application*, натисніть кнопку *Далі*, введіть ім'я проекту *model*, дайте назву FXML-файлу *FXMLModel*, установіть прапорець *Create Application Class* і натисніть кнопку *Готово*. В результаті середовищем *NetBeans* буде згенеровано проект, що містить у каталозі *src* папку пакета головного класу додатка з трьома файлами – *Java*-файлом головного класу додатка *Model.java*, *FXMLModel.fxml* і *FXMLModelController.java*.

2. Відкрийте файл *FXMLModel.fxml* у середовищі *JavaFX Scene Builder*, клацнувши двічі по *FXMLModel.fxml*. Внесіть зміни в макет FXML-файлу (рис. 2).

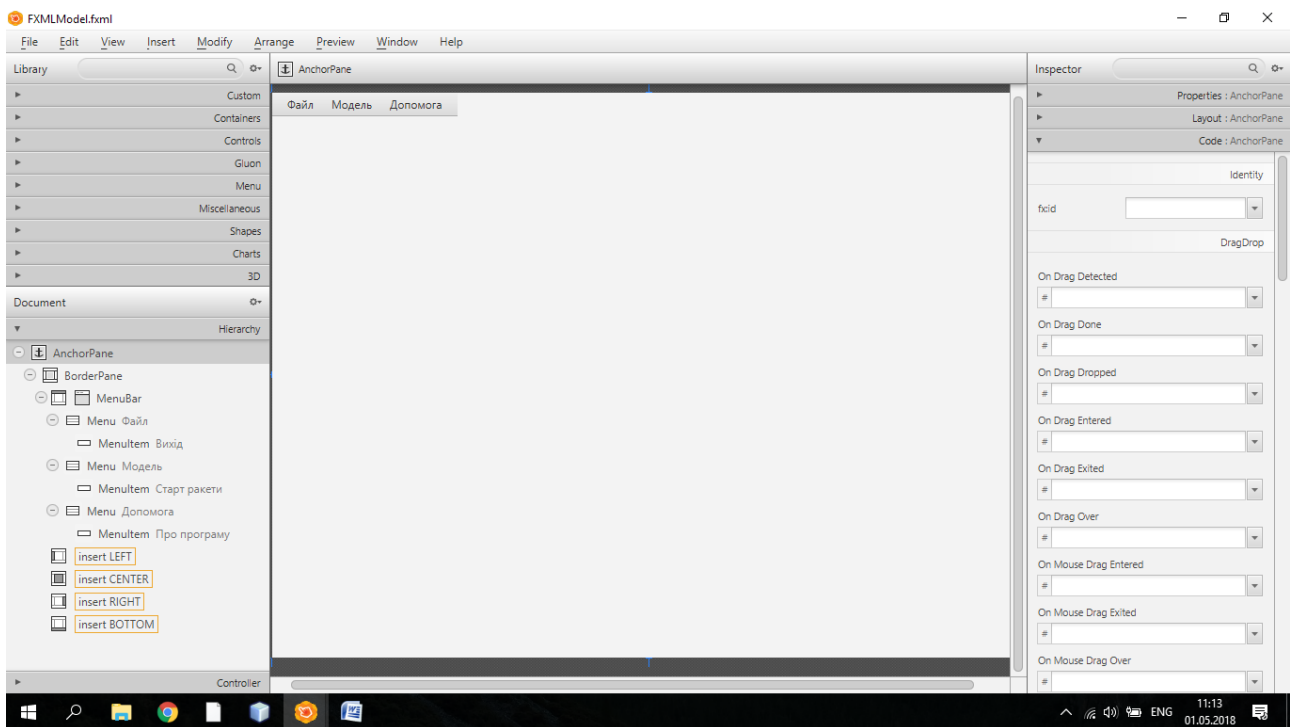


Рис. 2. Макет файлу *FXMLModel.fxml* у середовищі *JavaFX Scene Builder*

3. Виберіть компонент *AnchorPane* на вкладці *Hierarchy*, перейдіть на вкладку *Layout* і в полі *Pref Width* установіть значення 800, а в полі *Pref Height* – значення 600.

Далі перейдіть на вкладку *Code* і для компонента *AnchorPane* в полі *fx:id* установіть значення *root*.

Для компонента *MenuItem Close* у полі *fx:id* установіть значення *menuItemClose*, а в полі *On Action* задайте значення *menuItemCloseOnAction*. Аналогічно для компонента *MenuItem Rocket* в полі *fx:id* установіть значення *menuItemRocket*, а в полі *On Action* задайте значення *menuItemRocketOnAction*.

4. У меню *View* виберіть *Show Sample Controller Skeleton*. Далі виберіть у вікні діалогу наступний текст.

```
@FXML
private AnchorPane root;
@FXML
private MenuItem menuItemClose;
@FXML
private MenuItem menuItemRocket;
@FXML
void menuItemCloseOnAction(ActionEvent event) {
}
```


@FXML

```
void menuItemRocketOnAction(ActionEvent event) {  
}
```

Після цього замініть код у `public class FXXMLModelController implements Initializable { ...}` виділеним фрагментом.

У результаті `FXXMLModelController.java` матиме такий вигляд.

```
package model;  
import static com.sun.corba.se.impl.util.Utility.printStackTrace;  
import java.io.IOException;  
import java.net.URL;  
import java.util.ResourceBundle;  
import javafx.event.ActionEvent;  
import javafx.fxml.FXML;  
import javafx.fxml.FXMLLoader;  
import javafx.fxml.Initializable;  
import javafx.scene.control.MenuItem;  
import javafx.scene.layout.AnchorPane;  
public class FXXMLModelController implements Initializable {  
    @FXML  
    private AnchorPane root;  
    @FXML  
    private MenuItem menuItemClose;  
    @FXML  
    private MenuItem menuItemRocket;  
    @FXML  
    void menuItemCloseOnAction(ActionEvent event) {  
    }  
    @FXML  
    void menuItemRocketOnAction(ActionEvent event) throws IOException {  
    }  
    @Override  
    public void initialize(URL url, ResourceBundle rb) {  
        // TODO  
    }  
}
```

5. Вставте в обробник події `menuItemCloseOnAction` наступний код:

```
printStackTrace();  
System.exit(0);
```

6. Внесіть зміни у файл `Model.java`, вставивши в метод `start` наступний код:

```
stage.setResizable(false);  
stage.setTitle("Комп'ютерне моделювання фізичних процесів і явищ");
```

7. Запустіть додаток на виконання. Загальний вигляд головної форми подано на рис. 3.

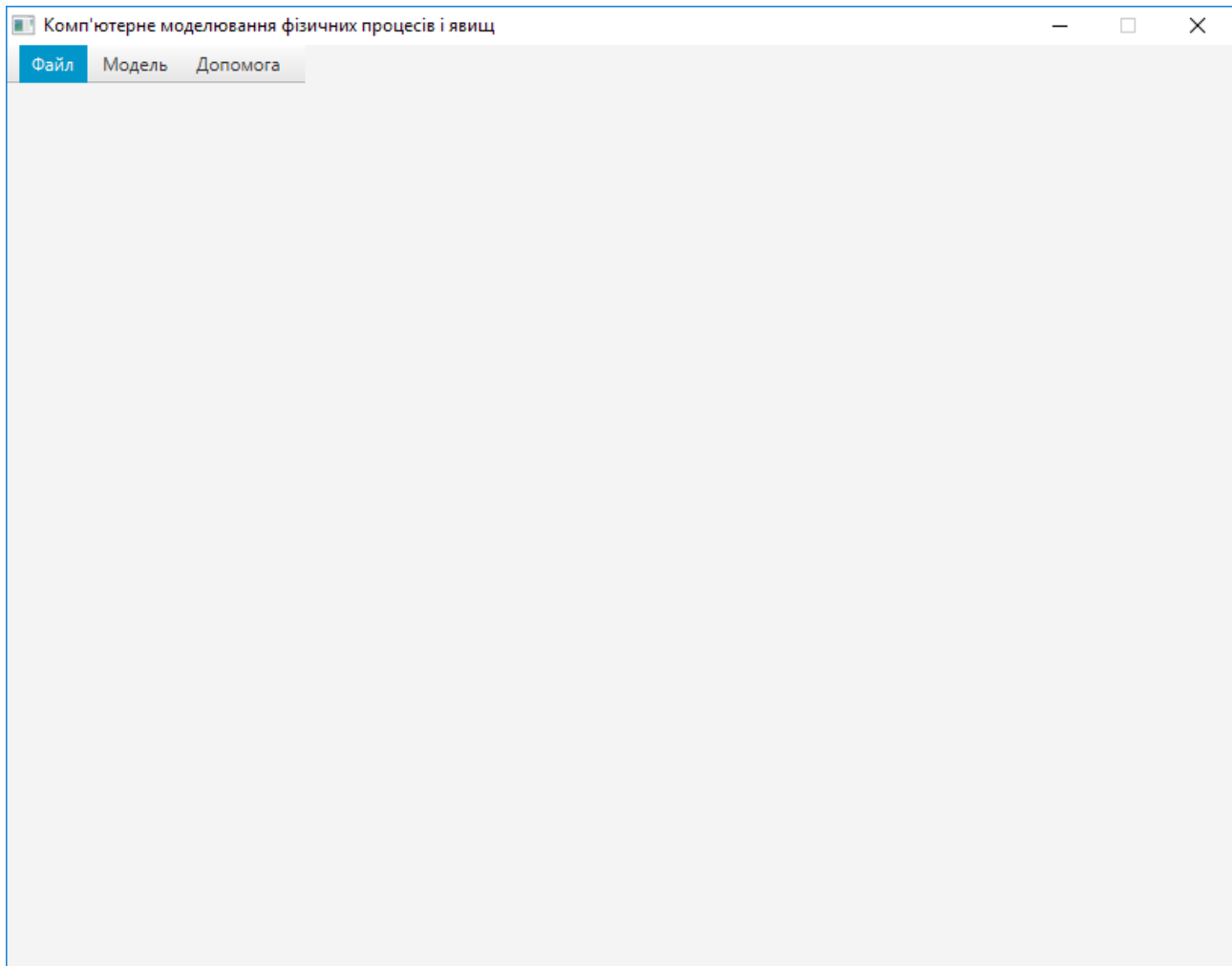


Рис. 3. Загальний вигляд головної форми

Створення макету форми графічних побудов

1. У меню *Файл* виберіть *Створити файл | JavaFX | Empty FXML*, натисніть кнопку *Далі*. Дайте назву FXML-файлу *FXMLAll*, натисніть кнопку *Далі*. Встановіть прапорець *Use Java Controller* і натисніть кнопку *Далі*. Натисніть кнопку *Готово*. В результаті середовищем *NetBeans* буде згенеровано два файли – *FXMLAll.fxml* і *FXMLAllController.java*.

2. Відкрийте файл *FXMLAll.fxml* у середовищі *JavaFX Scene Builder*, клацнувши двічі по *FXMLAll.fxml*. Створіть макет FXML-файлу як показано на рис. 4.

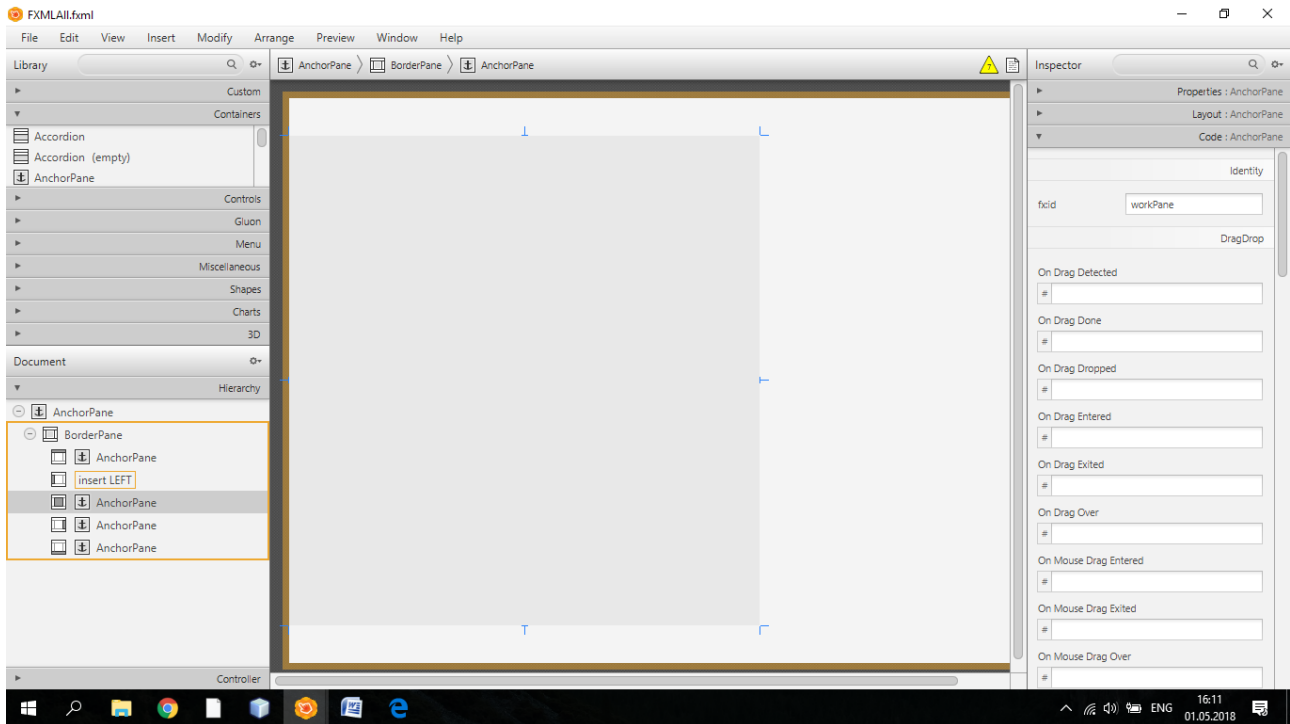


Рис. 4. Макет файлу *FXMLAll.fxml* у середовищі *JavaFX Scene Builder*

3. Виберіть компонент *AnchorPane* на вкладці *Hierarchy*, перейдіть на вкладку *Layout* і в полі *Pref Width* установіть значення 800, а в полі *Pref Height* – значення 600. Далі перейдіть на вкладку *Code* і для компонента *AnchorPane* в полі *fx:id* установіть значення *root*.

Виберіть компонент *AnchorPane*, який знаходиться в *TOP BorderPane*, перейдіть на вкладку *Layout* і в полі *Pref Width* установіть значення 800, а в полі *Pref Height* – значення 40. Далі перейдіть на вкладку *Code* і для компонента *AnchorPane* в полі *fx:id* установіть значення *topPane*.

Аналогічно виберіть компонент *AnchorPane*, який знаходиться в *BOTTOM BorderPane*, перейдіть на вкладку *Layout* і в полі *Pref Width* встановіть значення 800, а в полі *Pref Height* – значення 40. Далі перейдіть на вкладку *Code* і для компонента *AnchorPane* в полі *fx:id* установіть значення *bottomPane*.

Виберіть компонент *AnchorPane*, який знаходиться в *CENTER BorderPane*, перейдіть на вкладку *Layout* і в полі *Pref Width* установіть значення 500, а в полі *Pref Height* – значення 520. Далі перейдіть на вкладку *Code* і для компонента *AnchorPane* в полі *fx:id* установіть значення *workPane*.

Виберіть компонент *AnchorPane*, який знаходиться в *RIGHT BorderPane*, перейдіть на вкладку *Layout* і в полі *Pref Width* установіть значення 300, а в полі *Pref Height* – значення 520. Далі перейдіть на вкладку *Code* і для компонента *AnchorPane* в полі *fx:id* установіть значення *rightPane*.

4. У меню *View* виберіть *Show Sample Controller Skeleton*. Виберіть у вікні діалогу наступний текст.

```
@FXML
private AnchorPane root;
@FXML
private AnchorPane topPane;
@FXML
private AnchorPane rightPane;
@FXML
private AnchorPane bottomPane;
@FXML
private AnchorPane workPane;
```

Вставте виділений код у *public class FXMLAllController implements Initializable { ...}*. Замініть *private* на *public*.

```
У результаті FXMLAllController.java матиме такий вигляд.
package model;
import java.net.URL;
```

```
import java.util.ResourceBundle;
import javafx.fxml.FXML;
import javafx.fxml.Initializable;
import javafx.scene.layout.AnchorPane;
public class FXMLAllController implements Initializable {
    @FXML
    public AnchorPane root;
    @FXML
    public AnchorPane topPane;
    @FXML
    public AnchorPane rightPane;
    @FXML
    public AnchorPane bottomPane;
    @FXML
    public AnchorPane workPane;
    @Override
    public void initialize(URL url, ResourceBundle rb) {
        // TODO
    }
}
```

5. Запустіть додаток на виконання (F6) та збережіть проект (File\Save All).

Висновки та перспективи подальшого дослідження. У роботі розкрито особливості розробки додатків на основі технології JavaFX з використанням конструктора макетів Scene Builder. Показано, що програмний інтерфейс JavaFX API дає можливість створювати RIA-додатки, код яких поєднує широкі можливості платформи Java з багатою графікою та медіафункціональністю платформи JavaFX. Основним засобом розробки візуального інтерфейсу користувача є JavaFX Scene Builder. Його можуть використовувати як Java-розробники, так і дизайнери. Перші можуть швидко створювати прототипи інтерфейсу користувача і окремо розробляти логіку додатку. Другі можуть не тільки швидко створювати візуальний інтерфейс без написання будь-якого коду, а й проектувати і переглядати макет візуального інтерфейсу, змінювати зовнішній вигляд інтерфейсу користувача за допомогою таблиць стилів CSS.

На прикладі створення діалогового середовища розглянуто основні етапи проектування JavaFX-додатків в інтегрованому середовищі розробки NetBeans засобами Scene Builder.

Список бібліографічного опису

1. Афанасьев К.С., Лебедев К.С. Обзор современных технологий создания RIA-приложений. *Кибернетика. Управление в сложных системах. Вестник ИрГТУ.* № 4 (44). 2010. С. 6–12.
2. Герасимов В. В., Левицька В. Я. Аналіз технологій розробки насичених інтернет-додатків на платформі Java. *IT проектування, моделювання, дизайну, WEB,* 2017. С. 355–363.
3. Карашецкий В. П. Побудова графічного контенту додатків з використанням JavaFX і Swing компонентів і даних, взятих із баз даних. *Науковий вісник НЛТУ.* 2015. Вип. 25.1. С. 386–392.
4. Машнин Т. С. *JavaFX 2.0: разработка RIA-приложений.* СПб.: БХВ-Петербург, 2012. 320 с.
5. Муляр В. П., Федонюк А. А. *Комп'ютерне моделювання фізичних процесів і явищ: навч. посіб.* Луцьк: ПП Іванюк В. П., 2018. 212 с.
6. Парфенов Ю. Э., Федорченко В. Н. Разработка «насыщенных» интернет-приложений с помощью JavaFX. *Системы обработки информации.* 2012. Вип. 8 (106). С. 40–46.
7. Хорстманн К. *Java SE 8. Вводный курс.* М.: Вильямс, 2014. 208 с.
8. Carl Dea. *JavaFX 2.0: Introduction by Example.* Apress, 2011. 181 p.
9. JavaFX – Application. URL: https://www.tutorialspoint.com/javafx/javafx_application.htm
10. JavaFX Scene Builder. URL: <http://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-info-2157684.html>
11. Kishori Sharan. *Learn JavaFX 8 (The Expert's Voice in Java): Building User Experience and Interfaces with Java 8.* New York, 2015. P. 1–1173.
12. Stephen Chin, Johan Vos, James Weaver. *The Definitive Guide to Modern Java Clients with JavaFX: Cross-Platform Mobile and Cloud Development.* Apress, 2019. 621 p.

References

1. Afanas'ev K.S., Lebedev K.S. Obzor sovremennykh tehnologiy sozdanija RIA-prilozhenij. *Kibernetika. Upravlenie v slozhnyh sistemah. Vestnik IrGTU.* № 4 (44). 2010. S. 6–12.
2. Herasymov V. V., Levytska V. Ya. Analiz tekhnolohii rozrobky nasychenykh internet-dodatkov na platformi Java. *IT proektirovaniya, modelirovaniya, dizajna, WEB,* 2017. S. 355–363.

3. Karashetskyi V. P. Pobudova hrafichnoho kontentu dodatviv z vykorystanniam JavaFX i Swing komponentiv i danykh, vziatykh iz baz danykh. Naukovyi visnyk NLTU. 2015. Vyp. 25.1. S. 386–392.
4. Mashnin T. S. JavaFX 2.0: razrobotka RIA-prilozhenij. SPb.: BHV-Peterburg, 2012. 320 s.
5. Muliar V. P., Fedoniuk A. A. Kompiuterne modeliuvannia fizychnykh protsesiv i yavyshch: navch. posib. Lutsk: PP Ivaniuk V. P., 2018. 212 s.
6. Parfenov Ju. Je., Fedorchenko V. N. Razrobotka «nasyshhennyh» internet-prilozhenij s pomoshh'ju JavaFX. Systemy obrobky informatsii. 2012. Vyp. 8 (106). S. 40–46.
7. Horstmann K. Java SE 8. Vvodnyj kurs. M.: Vil'jams, 2014. 208 s.
8. Carl Dea. *JavaFX 2.0: Introduction by Example*. Apress, 2011. 181 p.
9. JavaFX – Application. URL: https://www.tutorialspoint.com/javafx/javafx_application.htm
10. JavaFX Scene Builder. URL: <http://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-info-2157684.html>
11. Kishori Sharan. *Learn JavaFX 8 (The Expert's Voice in Java): Building User Experience and Interfaces with Java 8*. New York, 2015. P. 1–1173.
12. Stephen Chin, Johan Vos, James Weaver. *The Definitive Guide to Modern Java Clients with JavaFX: Cross-Platform Mobile and Cloud Development*. Apress, 2019. 621 p.

Рецензенти:

Пех П. А., завідувач кафедри комп'ютерної інженерії та кібербезпеки Луцького національного технічного університету, кандидат технічних наук, доцент.

Яцюк С. М., в.о. завідувача кафедри вищої математики та інформатики Східноєвропейського національного університету імені Лесі Українки, кандидат педагогічних наук, доцент.

DOI: 10.36910/6775-2524-0560-2020-39-31

УДК 004.056:33(045)

Ніколіна Ірина Іванівна, канд. наук держ. упр., доцент

<https://orcid.org/0000-0001-7718-8599>

Гулівата Інна Олександрівна, канд. пед наук, доцент

<https://orcid.org/0000-0003-4752-535X>

Вінницький торговельно-економічний інституту Київського національного торговельно-економічного університету, м. Вінниця, Україна

МОДЕЛЮВАННЯ КІБЕРЗЛОЧИННОСТІ ЯК ЗАГРОЗИ ЦИФРОВІЗАЦІЇ ЕКОНОМІКИ

Ніколіна І.І., Гулівата І.О. Моделювання кіберзлочинності як загрози цифровізації економіки. Здійснено аналітичний огляд наукових досліджень зарубіжних вчених щодо реалізації тренду цифрової трансформації, методологічного апарату оцінки розвитку цифрової економіки, низки нормативно-правових актів з питань цифрових змін в Україні, напрацьовань з моделювання окремих ризиків цифровізації економіки. Проведено аналіз наукової літератури щодо специфіки трактування дефініції «цифровізація». Обґрунтовано, що впровадження цифровізації в економіку спричиняє ряд загроз, основна з яких є зростання рівня кіберзлочинності. Визначено основні види кіберзлочинів. Змодельовано динаміку головної загрози розвитку цифрової економіки – кіберзлочинність – задля оптимізації управління економічними системами. На основі статистичних даних показано, що темп зростання кіберзлочинності пришвидшиться. Передбачено основні види кіберзлочинів, які нестимуть найбільшу небезпеку. Виявлено, що основними порушеннями з кіберзлочинів є злочини у сфері платіжних систем. Запропоновано превентивні шляхи діяльності.

Ключові слова: цифровізація, цифрова економіка, цифрові технології, кіберзлочинність, моделювання.

Николина И.И., Гуливата И.О. Моделирование киберпреступности как угрозы цифровизации экономики. Осуществлен аналитический обзор научных исследований зарубежных ученых по реализации тренда цифровой трансформации, методологического аппарата оценки развития цифровой экономики, ряда нормативно-правовых актов по вопросам цифровых изменений в Украине, наработок по моделированию отдельных рисков цифровизации экономики. Проведен анализ научной литературы о специфике трактовки дефиниции «цифровизация». Обосновано, что внедрение цифровизации в экономику вызывает ряд угроз, основная из которых является рост уровня киберпреступности. Определены основные виды киберпреступлений. Смоделирован динамику главной угрозы развития цифровой экономики - киберпреступность - для оптимизации управления экономическими системами. На основе статистических данных показано, что темп роста киберпреступности ускорится. Предусмотрены основные виды киберпреступлений, которые будут нести наибольшую опасность. Выявлено, что основными нарушениями со киберпреступлений являются преступления в сфере платежных систем. Предложено превентивные пути деятельности.

Ключевые слова: цифровые технологии, Mobile ID, электронная услуга, электронная подпись, идентификация.

Nikolina I., Hulivata I. Modeling cybercrime as a threat to the digitalization of the economy. An analytical review of scientific research of foreign scientists on realization of the digital transformation trend is conducted, as well as on the methodological apparatus for assessing the digital economy development, a series of regulatory acts concerning digital changes in Ukraine and the experience of modelling individual risks of economy digitalization. An analysis of scientific literature on the specifics of the interpretation of the definition "digitalization" is carried out. It is substantiated that the implementation of digitalization into economy creates a number of threats, the main of which is an increase in the level of cybercrime. The main types of cybercrime are defined. The dynamics of the main threat to the digital economy development – cybercrime – are modelled to optimize the management of economic systems. The statistics show that the growth rate of cybercrime will accelerate. The main types of cybercrime that will carry the highest risk are foreseen. It is found that the main violations are crimes in the field of payment systems. Preventive activities are suggested.

Key words: digitalization, digital economy, digital technologies, cybercrime, modelling.

Постановка наукової проблеми.

Світ вступив у нову епоху цифрової глобалізації, яка визначається безперервними потоками даних, які містять інформацію, знання, ідеї та інновації. Розвинені країни, завершивши індустріалізацію, успішно цифровізують економіки, прискореними темпами розвиваючи цифрові технології, де домінують технології «відкритих даних» (Open Data), «цифрових платформ» (Digital Platform), «блокчейну» (Blockchain) «цифрового робочого місця» (Digital Workplace), «багатоканального інформування та залучення громадян» (Multichannel citizen engagement), «Інтернет послуг» (IoS), «Кіберфізичні системи» (Cyber-Physical System), «Смарт-факторія» (Smart Factory), «Спільні сервіси 2.0» Shared services 2.0 (Shared services 2.0), штучного інтелекту (AI).

Трансформаційні зміни сучасної України обумовлюються глобальною цифровізацією, яка відображає причинно-наслідковий зв'язок четвертої індустріальної революції («Industry 4.0») та розвитку суспільства («цифрове суспільство»). Масштаб і темп цифрових трансформацій стали основними характеристиками економічного розвитку [12].

Реалізація цифрової стратегії розвитку економіки України вимагають від уряду зважених нормативної, організаційно-функціональних складових механізму управління відповідно і до зростання загроз, що спричиняє тотальна цифрова трансформація.

Аналіз досліджень. Питання становлення цифрової економіки, її наслідки для суспільства й держави, реалізації тренду цифрової трансформації стали об'єктом наукових досліджень зарубіжних вчених, експертів міжнародного індексу оцінки цифрової економіки та суспільства [17-21].

Впродовж попередніх років міжнародними організаціями здійснюється оцінка тих чи інших сегментів цифрової економіки, зокрема, на сьогодні існує достатній методологічний апарат оцінки розвитку цифрової економіки: Digital Economy and Society Index, Digital Evolution Index, The UN Global E-Government Development Index, ICT Development Index, Networked Readiness Index, E-Participation Index, The Global Innovation Index, Global Connectivity Index.

В Україні питання цифрових змін втілені в розпорядженні КМУ: «Про схвалення Концепції розвитку цифрової економіки та суспільства України на 2018-2020 роки та затвердження плану заходів щодо її реалізації» від 17 січня 2018 № 67-р [12] та постанові КМУ «Деякі питання цифрового розвитку» від 30 січня 2019 р. № 56 [4]. Створення умов для розвитку цифрової економіки закладено у проекті закону «Про цифровий порядок денний України» [16]. В листопаді 2018 р. вступив в дію закон України «Про електронні довірчі послуги», який спрямований на реформування нормативної бази у сфері цифрового підпису, яка уможливує зростання швидкості розвитку цифрової економіки з обов'язковою цифровою партисипацією (digital participation) [11].

Серед вітчизняних науковців досліджуваний проблематиці та загрозам цифровізації приділяли увагу О. Вінник [1], Н. Краус, О. Голобородько та К. Краус [9], В. Ляшенко [12], М. Руденко [13], Н. Ткачук [14], С. Яремко [18]. Питанню моделювання окремих ризиків цифровізації присвячено праці [1, 2, 5-7, 13-16].

Виділення не вирішених раніше частин загальної проблеми. Проте ми вважаємо, що потребує подальшого дослідження питання моделювання загроз цифровізації економіки, зокрема кіберзлочинності.

Мета статті. Головною метою цієї роботи є моделювання проблем, що можуть виникнути в процесі цифрової трансформації, зокрема кіберзлочинності як загрози розвитку цифрової економіки задля оптимізації управління економічними системами.

Виклад основного матеріалу. Диференціація розуміння багатогранності категорії «цифровізація» М. Руденком проявляється у трактуванні цього поняття з чотирьох позицій — держави, науковців, практиків (підприємців) та суспільства, оскільки визначення вищенаведеного поняття значно видозмінюється в залежності від сфери його застосування, категорії осіб, які давали визначення та кінцевої мети надання [13].

Цифровізація економіки трактується науковцями Т. Юдіною та І. Тушкановим у вузькому та широкому значенні. Зокрема, у вузькому значенні — створення на різних рівнях економіки (глобальному, мега, макро, мезо, мікро, нано) інформаційно-цифрових платформ і операторів, які дозволяють вирішувати різні завдання, у тому числі стратегічні: розвиток медицини, науки, освіти, транспорту, нової індустріалізації, державного регулювання економіки; у широкому сенсі — зміна природи виробничих або економічних відносин, зміну їх суб'єктивно-об'єктивної орієнтованості [177]. Під час цифрової трансформації змінюються виробничі сили суспільства та (або) фактори виробництва [10].

Цифровізація — насичення фізичного світу електронно-цифровими пристроями, засобами, системами та налагодження електронно-комунікаційного обміну між ними, що фактично уможливує інтегральну взаємодію віртуального та фізичного, тобто створює кіберфізичний простір [12].

Суспільство вбачає у цифровізації абсолютно нову парадигму розвитку держави, основу на повсякденному та повсюдному використанні цифрових технологій з обов'язковою наявністю цифрових компетенцій, що забезпечить революційні зміни та становлення цифрового суспільства.

Проведений аналіз наукової літератури дозволяє стверджувати, що специфіка трактування дефініції «цифровізація» обумовлена суб'єктом надання визначення, що спричиняє неоднозначність розуміння досліджуваного поняття та підкреслює дискусійність окремих положень і тверджень.

Цифровізація економіки забезпечує ефективну двосторонню взаємодію держави, суспільства, бізнесу, людини за допомогою цифрових технологій за наявності в усіх учасників комунікації належних цифрових компетенцій.

Цифрова економіка нерозривно пов'язана з наявністю інформаційного суспільства. В основу терміну «інформаційне суспільство» покладено положення про те, що кількісні зміни у сфері інформації привели до виникнення якісно нового типу соціального устрою – інформаційного суспільства.

Цифровізацію варто розглядати як інструмент, а не як самоціль. При системному державному підході «цифрові» технології будуть стимулювати розвиток відкритого інформаційного суспільства як одного з істотних факторів підвищення продуктивності, економічного зростання, створення робочих місць, а також покращення якості життя громадян України [166].

«Цифрові» технології необхідні для зростання ефективності української промисловості, а в деяких секторах вони стають основою продуктивних та виробничих стратегій. Їх перетворююча сила змінює традиційні моделі бізнесу, виробничі ланцюжки та обумовлює появу нових продуктів та інновацій. Цифровізація для України носить позитивний соціальний характер, адже зосереджена на поліпшенні якості інфраструктури соціального забезпечення, якості соціальних послуг, організації прозорості та адресності соціальної допомоги, та скорочення витрат [166].

Цифровізація стрімко проникає як в повсякденне життя людей, так і надає нові можливості для всіх секторів економіки України у розрізі модернізації методів роботи та управління, змінюючи виробничі цикли, логістику.

Вагома проблема, яка з'явилася з початком цифрової трансформації та загострюватиметься з часом – кіберзлочинність. Науковці комп'ютерні злочини умовно поділяв на дві основні групи, виходячи з класифікаційної ознаки категорії доступу до засобів комп'ютерної техніки: 1) внутрішні користувачі; 2) зовнішні користувачі, де користувач – суб'єкт, який звертається до інформаційної системи, або посередника, за отриманням необхідної йому для користування інформації [5].

Водночас із поширенням використання інтернет-технологій пропорційно зростає й загроза правопорушень, метою яких є хакерські атаки, викрадення персональної інформації, блокування роботи інформаційних служб, шантаж, шахрайство тощо. Це зумовлено низкою причин, зокрема зростанням довіри до електронних засобів обробки інформації, розширенням кола суб'єктів – учасників інформаційних відносин у глобальній мережі, збільшенням кількості різноманітних сервісів, переходу до обслуговування банківських установ. В Інтернеті сьогодні набули поширення різноманітні схеми, спрямовані на отримання коштів з недосвідчених і довірливих користувачів інтернет-магазинів, віртуальних аукціонів, сайтів знайомств тощо. Зазвичай для такого виду шахрайства використовуються інтернет-сайти, що візуально та за назвою нагадують відомі міжнародні ресурси. Проте, на відміну від добре зарекомендованих брендів, на отримання замовленого товару або повернення коштів годі й сподіватися. Причина користування такими ресурсами – бажання отримати замовлення за надзвичайно низькою ціною. Іноді зловмисники використовують і протилежні якості людини, наприклад, створюючи фіктивний сайт благодійного фонду або школи-інтернату [5].

З поширенням технологій змінився і характер злочинів. Якщо раніше більшість з них припадала на махінації з пластиковими картками, то тепер відбувається справжній бум у сфері онлайн-платежів. Найбільш професійні хакери вже перейшли у сферу крадіжок через клієнт-банки (системи дистанційного банківського обслуговування). Махінації з картками відходять на другий план, натомість збільшується кількість крадіжок з рахунків компаній або з електронних гаманців [16].

Визначаючи сучасний стан кіберзлочинності в Україні, зауважимо, що вона, як і будь-яке інше соціальне явище, піддається передбаченню за допомогою певних індикаторів, що відбивають її кількісні та якісні характеристики. Здійснити таке оцінювання і моделювання можна через аналіз показників поширеності кіберзлочинності в Україні: її рівня, географії, структури, динаміки тощо.

Стосовно рівня кіберзлочинності та її динаміки зазначимо, що у 2009 р. в Україні було зареєстровано 217 злочинів у сфері використання електронно-обчислювальних машин (комп'ютерів), систем та комп'ютерних мереж і мереж електрозв'язку, у 2010 р. – 190, у 2011 р. – 131, у 2012 р. – 138, у 2013 р. – 595, у 2014 р. – 443, у 2015 р. – 598, у 2016 р. – 865, у 2017 р. – 2573, за січень-серпень 2018 р. – 1885 злочинів (рис. 1).

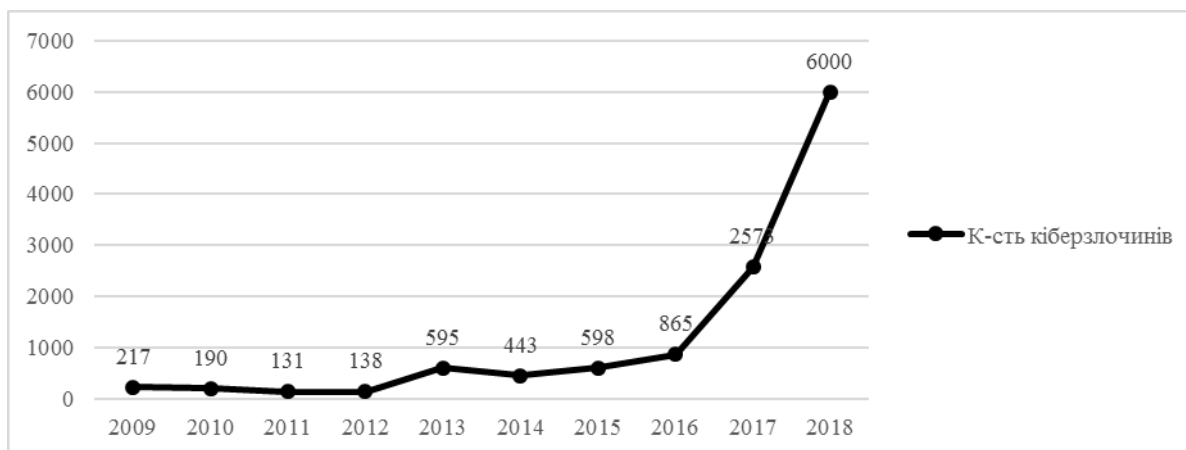


Рис. 1. Динаміка кількості кіберзлочинності в Україні

Джерело: розроблено авторами

Питома вага кіберзлочинів у загальній кількості зареєстрованих злочинів становить 0,05 % від загальної кількості зареєстрованих злочинів у 2009 р., 0,04 % – у 2010, 0,03 % – у 2011, 0,03 % – у 2012, 0,11 % – у 2013, 0,08 % – у 2014, 0,11 % у – 2015, 0,15 % – у 2016, 0,49 % – у 2017 р. та 0,51 % від злочинів,

зареєстрованих за січень-серпень 2018 р. [8].

Суттєве збільшення кількості зареєстрованих у 2013 р. кіберзлочинів окремі вчені пов'язують із тим, що «зростання вказаного виду злочинності обумовлено щорічним зростанням користувачів Інтернет-ресурсу в Україні» [6].

Як бачимо з отриманих статистичних даних по злочинності, кіберзлочини мають значний позитивний приріст (рис. 2).

Наступним не менш важливим кроком є визначення видів кіберзлочинів, які нестимуть найбільшу небезпеку, для цього скористаємось методом АВС-аналізу.

Для розв'язання поставленої проблеми проведемо АВС-аналіз методом «суми» на основі статистичних даних звіту кіберполіції України по кіберзлочинам за 2018 рік.

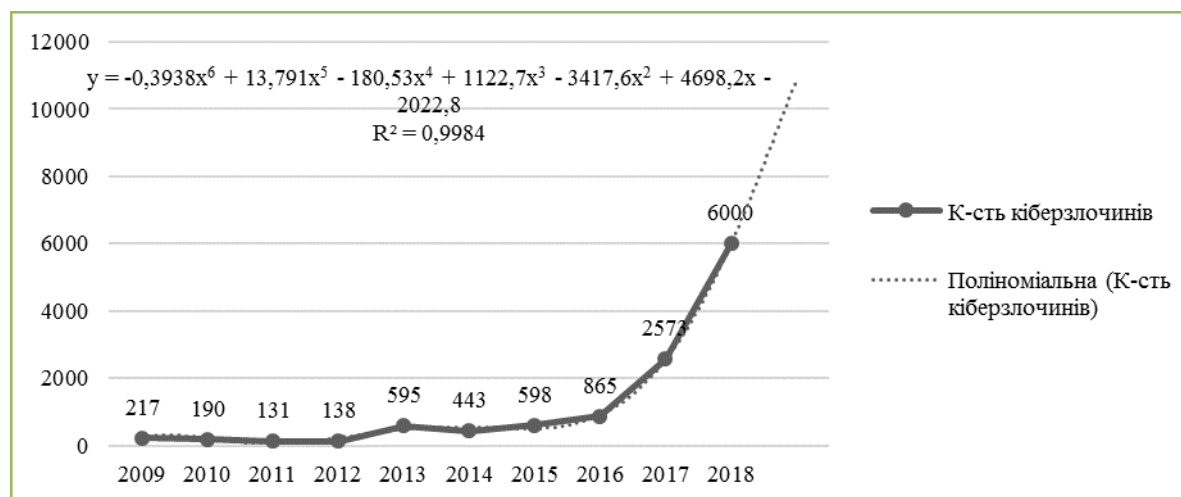


Рис. 2. Прогнозування кіберзлочинності в Україні

Джерело: розроблено авторами

Зведемо результати у таблицю 1.

Таблиця 1. ABC-аналіз кіберзлочинів

Кіберзлочин	Кількість	Частка фактора у сумі значень фактора за даними [8], %	Наростаюче значення ВО, %	Наростаюче значення ЧО, %	Сума ЧО та ВО, %	Група
у сфері платіжних систем	2398	39,96000667	39,96001	25	64,96001	A
у сфері е-комерції	1598	26,62889518	66,5889	50	116,5889	B
у сфері кібербезпеки	1325	22,07965339	88,66856	75	163,6686	B
у сфері протиправного контенту	680	11,33144476	100	100	200	C

Джерело: розроблено авторами

На рис. 3 представлена ABC-крива, яка графічно інтерпретує поділ кіберзлочинів на групи.

Узагальнюючи отримані результати, зауважимо, що найуразливішою є сфера платіжних систем. Впровадження контрзаходів у ній може зменшити кількість кіберзлочинів майже на 40% від усіх можливих.

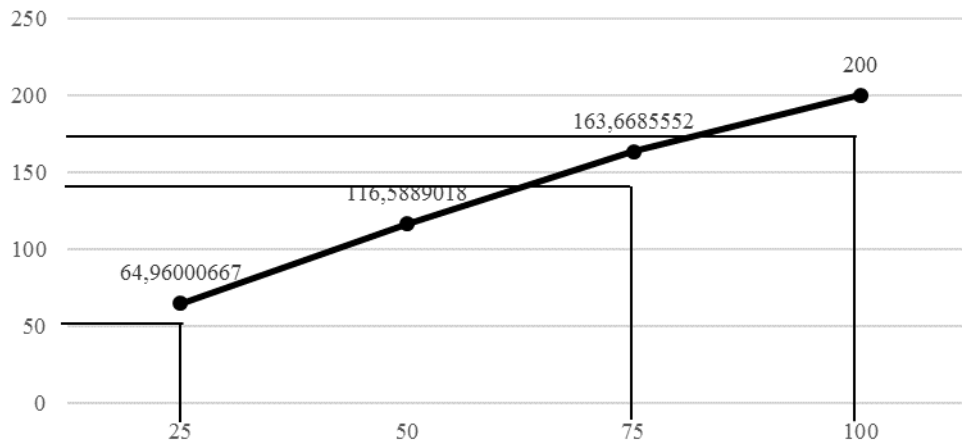


Рис. 3. Графічний результат ABC-аналізу

Джерело: розроблено авторами

Ми вважаємо, що цифровізація економіки збільшить приріст кіберзлочинів. Адже більшість населення, установ, фірм та організацій почнуть переходити на повне цифрове забезпечення, тобто основну інформацію будуть зберігати у цифровому просторі та на серверах.

Уже сьогодні Україна активно бореться з даною проблемою, створено уповноважений орган – Кіберполіція України. На даний момент вона виконує такі функції як: реалізація державної політики у сфері протидії кіберзлочинності; завчасне інформування населення про появу новітніх кіберзлочинів; впровадження програмних засобів для систематизації та аналізу інформації про кіберінциденти, кіберзагрози та кіберзлочини; реагування на запити закордонних партнерів, що надходять каналами національної цілодобової мережі контактних пунктів; участь у підвищенні кваліфікації працівників поліції щодо застосування комп'ютерних технологій у протидії злочинності; участь у міжнародних операціях та співпраця в режимі реального часу. забезпечення діяльності мережі контактних пунктів між 90 країнами світу; протидія кіберзлочинам.

На майбутній період пропонуємо удосконалити національну систему кібербезпеки України, щоб вона могла: забезпечити передню лінію оборони проти кіберзагроз шляхом посилення загальної ситуаційної обізнаності щодо інцидентів, вразливостей та загроз у середовищі державних установ, на об'єктах критичної інфраструктури, у громадському сегменті; запобігати вторгненням завдяки обміну інформацією та впровадженням контрзаходів, здатних зменшити поточні вразливості; захищати від повного спектру загроз шляхом підвищення контррозвідувальних та розвідувальних можливостей; зміцнити середовище кібербезпеки через освітянські, медійні громадські ініціативи; стимулювати та забезпечувати проведення кібернавчань, досліджень і розробок у сфері кібербезпеки.

Отже, після проведення нами моделювання впровадження цифровізації в економіку, робимо висновок, що Україна потерпає від зростання рівня кіберзлочинності.

З тотальною цифровізацією економіки лише пришвидшить даний темп, так за нашими прогнозами у 2019 році кількість кіберзлочинів може досягти відмітки 9000. Основними з яких становлять злочини у сфері платіжних систем (близько 40%). Тому потрібно покращити заходи безпеки у даній сфері, щоб зменшити рівень кіберзлочинності.

Нагальним залишається і покращення захисту е-комерції та забезпечення кібербезпеки (протидію селерам, кодерам, викрадення баз даних і т.д.).

Висновки і пропозиції. Під час дослідження ми обґрунтували, що впровадження цифровізації в економіку спричиняє ряд проблем та ризиків, основним, на нашу думку, з яких є зростання рівня кіберзлочинності. З провадженням цифрових та інформаційних технологій у повсякденне життя та економіку спричиняє поступовий зріст кіберзлочинності, про що свідчать результати наших досліджень. З повною цифровізацією економіки лише пришвидшить даний темп. Основними порушеннями з кіберзлочинів є злочини у сфері платіжних систем. Тому потрібно покращити заходи безпеки у даній сфері, щоб зменшити рівень кіберзлочинності.

Список бібліографічного опису

1. Бойченко О.В. Моделювання сучасних систем захисту інформаційних ресурсів (2009) *Вісник НАУ*. №1. URL: <http://www.lib.nau.edu.ua/Journals/frmDoc.aspx?param=689>.
2. Вінник О. М. Правове забезпечення цифрової економіки та електронного бізнесу. Монографія (2018), 212 с.
3. Гуцалюк М.В. Впровадження ID-web як необхідна умова безпеки в Інтернет. URL: http://archive.nbuv.gov.ua/portal/soc_gum/bozk/18text/g18_30.htm.
4. Деякі питання цифрового розвитку: постанова Кабінету Міністрів України від 30 січня 2019 р. № 56. URL: <https://zakon.rada.gov.ua/laws/show/56-2019-%D0%BF>.
5. Злочинність в Україні : стат. зб. (2011), 117 с.
6. Книженко О. О. Сучасний стан злочинів у сфері використання електронно-обчислювальних машин (комп'ютерів), автоматизованих систем, комп'ютерних мереж і мереж електров'язку в Україні (2014) *Бюлетень Міністерства юстиції України*. № 7, С. 122–127.
7. Кононович В., Копитін Ю. Використання АВС аналізу для оптимізації систем захисту інформації (2010) *Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні*. Вип. 2 (21), URL: https://ela.kpi.ua/bitstream/123456789/9099/1/21_p26.pdf
8. Кравцова М.О. Сучасний стан і напрями протидії кіберзлочинності в Україні. *Вісник кримінологічної асоціації України*. 2018. № 2(19). URL: <http://dSPACE.univd.edu.ua/xmlui/handle/123456789/3848>
9. Краус Н.М., Голобородько О.П., Краус К.М. Цифрова економіка: тренди та перспективи авангардного характеру розвитку (2018) *Ефективна економіка*, №1. URL: <http://www.economy.nayka.com.ua/?op=1&z=6047>
10. Ляшенко В.І. Цифрова модернізація економіки України як можливість проривного розвитку: монографія (2018), 252 с.
11. Ніколіна І.І., Ніколіна І.І., Януш М.П. Особливості впровадження та перспективи Mobile ID в Україні (2019) *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*. №34, С.91-95.
12. Про схвалення Концепції розвитку цифрової економіки та суспільства України на 2018-2020 роки та затвердження плану заходів щодо її реалізації: розпорядження Кабінету Міністрів України від 17 січня 2018 р. № 67-р. URL: <https://zakon.rada.gov.ua/laws/show/67-2018-%D1%80#n13>
13. Руденко М. В. Цифровізація економіки: нові можливості та перспективи (2018) *Економіка та держава*. № 11. URL: http://www.economy.in.ua/pdf/11_2018/13.pdf
14. Ткачук Н.А. Цифрова грамотність та кібергігієна в Україні (2019) *Побудова інформаційного суспільства: ресурси і технології : матеріали XVIII Міжнародної науково-практичної конференції*, Київ, 19-20 вересня 2019 р., 404 с. URL: http://www.uitei.kiev.ua/sites/default/files/materyaly_mon_end.pdf.
15. Україна 2030E — країна з розвинутою цифровою економікою. URL: <https://strategy.uifuture.org/kraina-z-rozvinutoyu-cifrovoyu-ekonomikoyu.html#6-2-11>
16. Цифрова адженда України – 2020. Концептуальні засади (версія 1.0) (2016), 90 с.
17. Юдина Т.Н. Тушканов І.М. Цифрова економіка сквозь призму філософії хозяйства и политической экономики (2017) *Філософія хозяйства*. №1. URL: <https://istina.msu.ru/publications/article/56607736/>
18. Яремко С. А. Управління інформаційною безпекою корпоративних систем на основі сучасних інформаційних технологій (2012), *Економічна кібернетика*. №1, С.96-101.
19. Geissbauer R., Vedso J., Schrauf S. Industry 4.0: Building the digital enterprise. URL: <https://www.pwc.com/gx/en/industries/industries-4.0/landing-page/industry-4.0-building-your-digital-enterprise-april-2016.pdf>
20. Schwab K. The Fourth Industrial Revolution (2016), 172 p.
21. The Digital Economy and Society Index (DESI). URL: <https://ec.europa.eu/digital-single-market/en/desi>.

References

1. Boichenko O.V. Modeliuvannya suchasnykh system zakhystu informatsiinykh resursiv (2009) *Visnyk NAU*. №1. URL: <http://www.lib.nau.edu.ua/Journals/frmDoc.aspx?param=689>.
2. Vinnyk O. M. Pravove zabezpechennia tsyfrovoy ekonomiky ta elektronnoho biznesu. Monohrafiia (2018), 212 s.
3. Hutsaliuk M.V. Vprovadzhennia ID-web yak neobkhdna umova bezpeky v Internet. URL: http://archive.nbuv.gov.ua/portal/soc_gum/bozk/18text/g18_30.htm.
4. Deiaki pytannia tsyfrovoho rozvytku: postanova Kabinetu Ministriv Ukrainy vid 30 sichnia 2019 r. № 56. URL: <https://zakon.rada.gov.ua/laws/show/56-2019-%D0%BF>.
5. Zlochyynnist v Ukraini : stat. zb. (2011), 117 s.

6. Knyzhenko O. O. Suchasnyi stan zlochniv u sferi vykorystannia elektronno-obchysliuvalnykh mashyn (komp'uteriv), avtomatyzovanykh system, komp'uternykh merezh i merezh elektrozv'iazku v Ukraini (2014) *Biuletyn Ministerstva yustytzii Ukrainy*. № 7, S. 122–127.
7. Kononovych V., Kopytin Yu. Vykorystannia ABC analizu dlia optymizatsii system zakhystu informatsii (2010) *Pravove, normatyvne ta metrolohichne zabezpechennia systemy zakhystu informatsii v Ukraini*. Vyp. 2 (21), URL: https://ela.kpi.ua/bitstream/123456789/9099/1/21_p26.pdf
8. Kravtsova M.O. Suchasnyi stan i napriamy protydiv kiberzlochnosti v Ukraini. *Visnyk kryminolohichnoi asotsiatsii Ukrainy*. 2018. № 2(19). URL: <http://dspace.univd.edu.ua/xmlui/handle/123456789/3848>
9. Kraus N.M., Holoborodko O.P., Kraus K.M. Tsyfrova ekonomika: trendy ta perspektyvy avanhardnoho kharakteru rozvytku (2018) *Efektivna ekonomika*, №1. URL: <http://www.economy.nayka.com.ua/?op=1&z=6047>
10. Liashenko V.I. Tsyfrova modernizatsiia ekonomiky Ukrainy yak mozhlyvist proryvnoho rozvytku: monohrafiia (2018), 252 s.
11. Nikolina I.I., Nikolina I.I., Yanush M.P. Osoblyvosti vprovadzhenia ta perspektyvy Mobile ID v Ukraini (2019) *Komp'uterno-intehrovani tekhnologii: osvita, nauka, vyrobnytstvo*. №34, С.91-95.
12. Pro skhvalennia Kontseptsii rozvytku tsyfrovoy ekonomiky ta suspilstva Ukrainy na 2018-2020 roky ta zatverdzhennia planu zakhodiv shchodo yii realizatsii: rozporiadzhennia Kabinetu Ministriv Ukrainy vid 17 sichnia 2018 r. № 67-r. URL: <https://zakon.rada.gov.ua/laws/show/67-2018-%D1%80#n13>
13. Rudenko M. V. Tsyfrovizatsiia ekonomiky: novi mozhlyvosti ta perspektyvy (2018) *Ekonomika ta derzhava*. № 11. URL: http://www.economy.in.ua/pdf/11_2018/13.pdf
14. Tkachuk N.A. Tsyfrova hramotnist ta kiberhihiiena v Ukraini (2019) *Pobudova informatsiinoho suspilstva: resursy i tekhnologii : materialy XVIII Mizhnarodnoi naukovo-praktychnoi konferentsii*, Kyiv, 19-20 veresnia 2019 r., 404 s. URL: http://www.uitei.kiev.ua/sites/default/files/materyaly_mon_end.pdf.
15. Ukraina 2030E — kraina z rozvynutoiu tsyfrovoy ekonomikoy. URL: <https://strategy.uifuture.org/kraina-z-rozvinutoyu-cifrovoyu-ekonomikoyu.html#6-2-11>
16. Tsyfrova adzhenda Ukrainy – 2020. Kontseptualni zasady (versii 1.0) (2016), 90 c.
17. Yudyna T.N. Tushkanov Y.M. Tsyfrovaia ekonomika skvoz pryzmu fylosofyy khoziaistva y polytycheskoi ekonomyy (2017) *Fylosofiya khoziaistva*. №1. URL: <https://istina.msu.ru/publications/article/56607736/>
18. Yaremko S. A. Upravlinnia informatsiinoy bezpekoii korporatyvnykh system na osnovi suchasnykh informatsiinykh tekhnologii (2012), *Ekonomichna kibernetika*. №1, S.96-101.
19. Geissbauer R., Vedso J., Schrauf S. Industry 4.0: Building the digital enterprise. URL: <https://www.pwc.com/gx/en/industries/industries-4.0/landing-page/industry-4.0-building-your-digital-enterprise-april-2016.pdf>
20. Schwab K. The Fourth Industrial Revolution (2016), 172 p.
21. The Digital Economy and Society Index (DESI). URL: <https://ec.europa.eu/digital-single-market/en/desi>.

Рецензенти:

С.А. Яремко – кандидат технічних наук, доцент, доцент кафедри економічної кібернетики та інформаційних систем Вінницького торговельно-економічного інституту КНТЕУ

І.Г. Юник – кандидат наук з державного управління, доцент, доцент кафедри правових наук та філософії Вінницького державного педагогічного університету імені Михайла Коцюбинського

DOI: 10.36910/6775-2524-0560-2020-39-32

УДК: 004.415.3

Пех Петро Антонович., к.т.н., доцент

<https://orcid.org/0000-0002-6327-3319>

Костюк Юрій Юрійович, студент

Кравченко Максим Богданович, студент

Луцький національний технічний університет

ДО ПИТАННЯ КОНСТРУЮВАННЯ КЛАСІВ З КОНСТРУКТОРАМИ РІЗНОГО ТИПУ

Пех П.А., Костюк Ю.Ю., Кравченко М.Б. До питання конструювання класів з конструкторами різного типу.

В статті запропоновано кілька програм мовою С++ на базі класів з різними типами конструкторів. Досліджуються також питання передачі об'єктів з головної функції у підпрограми-функції і навпаки. Наведені коди запропонованих програм.

Ключові слова: клас, функція, формальний параметр, конструктор з параметрами, конструктор копіювання

Пех П.А., Костюк Ю.Ю., Кравченко М.Б. К вопросу конструирования классов с конструкторами различных типов. В статье предложены несколько программ на языке С++ на базе классов с различными типами конструкторов. Исследуются также вопросы передачи объектов с главной функции подпрограммам-функциям и наоборот. Приведены коды предложенных программ.

Ключевые слова: класс, функция, формальный параметр, конструктор с параметрами, конструктор копирования

Pekh Petro, Kostyuk Yuri, Kravchenko Maxim. On the question of designing classes with constructors of different types. The article proposes several C++ language programs based on classes with different types of constructors. The transfer of objects from the main function to the subroutines-functions and vice versa is also investigated. The codes of the offered programs are resulted.

Keywords: class, function, formal parameter, constructor with parameters, copy constructor

Постановка задачі. Загальновідомо, що програми мовою С++ найчастіше розробляються на базі класів [1,2,4,7,8]. Клас є фундаментальним поняттям мови С++. Класи опрацьовують дані за допомогою методів [3,5]. Однією з функцій класу є конструктор. Саме конструктор є функцією, за якою програми створюють об'єкти даного класу і виконують їх попередні налаштування [4,6]. Конструктори бувають різних типів. Задача полягає у дослідженні дії різного типу конструкторів на дані класу.

Метою нашого дослідження було розроблення засобами мови С++ комплексу програм на базі класів з різними типами конструкторів.

Новизна полягає у підході до вирішення проблеми з позицій об'єктно-орієнтованого програмування [1,4,6].

Основна частина. Оскільки під час розробки класів використовуються конструктори різних типів – конструктор за замовчуванням, конструктори з одним, двома або більшою кількістю параметрів, конструктор копіювання, ми розробили комплекс програм, частина яких пропонується в даній статті для всебічного дослідження даного питання. Крім того, ми досліджували процес передачі об'єктів з головної функції у функції-підпрограми і навпаки.

Програма, код якої наведений нижче, розроблена на базі класу `PlanePoint`, вирішує досить просту задачу. Вона дозволяє ввести координати x, y довільної точки $M(x, y)$ площини і обчислити віддаль від цієї точки до точки початку координат $O(0, 0)$. Ми передбачили у програмі три конструктори: конструктор без параметрів `PlanePoint(void)`, конструктор з двома параметрами `PlanePoint(double nx, double ny)` та конструктор копіювання `PlanePoint(PlanePoint & ref_point)`. Клас `PlanePoint` має два власні методи: `int GetX(void){return x;}` та `int GetY(void){return y;}`, з допомогою яких можна отримати значення координат x, y довільної точки M площини. Функція `double GetLength(PlanePoint tmp)` не належить до методів класу `PlanePoint`, тому вона є зовнішньою по відношенню до класу функцією. Родзинкою є те, що ця функція має у якості формального параметра об'єкт `tmp` класу `PlanePoint` і з його допомогою передаються координати точки M .

Після запуску програми на виконання спочатку спрацьовує конструктор з двома параметрами під час виконання команди `PlanePoint Ob1(5, 8)`, який створює об'єкт `Ob1` з параметрами $x=5, y=8$, а потім під час виконання команди `PlanePoint Ob2` спрацьовує конструктор без

параметрів, який створює об'єкт `Ob1` з параметрами $x=7, y=21$. Процес контролюємо шляхом виведення на екран значень абсцис обох об'єктів за допомогою методу `GetX()`. Далі виконується команда побітового копіювання об'єктів `Ob2 = Ob1`. Тут конструктор копіювання не використовується, оскільки обидва об'єкти вже існували. Процес побітового копіювання об'єктів контролюємо виведенням на екран абсциси точки $x=5$ за допомогою методу `GetX()`. Конструктор копіювання спрацьовує під час виконання команди `PlanePoint Ob3 = Ob1`, яка забезпечує створення нового об'єкта `Ob3` і копіювання у нього значень $x=5, y=8$ об'єкта `Ob1`, у чому переконуємося шляхом виведення на екран значення $x=5$, отриманого методом `Ob3.GetX()`. Далі команда `PlanePoint Ob4(5,5)` створює об'єкт `Ob4`, а команда `len = GetLength(Ob4)` передасть цей об'єкт з головної функції у зовнішню функцію `double GetLength(PlanePoint tmp)`, яка обчислить і передасть у змінну `len` головної функції значення віддалі від точки $M(5,5)$ до точки $O(0,0)$. Головна функція виведе це значення на екран.

```
#include <iostream.h>
#include <conio.h>
#include <math.h>
#include <windows.h>
// клас PlanePoint
class PlanePoint {
    double x, y;
public:
    // конструктор класу без параметрів:
    PlanePoint(void);
    // конструктор класу з двома параметрами:
    PlanePoint(double nx, double ny);
    // конструктор копіювання класу PlanePoint -
    // передається посилання на об'єкт класу PlanePoint:
    PlanePoint(PlanePoint & ref_Point);
    // методи, реалізовані в класі
    int GetX(void) { return x; };
    int GetY(void) { return y; };
};
PlanePoint:: PlanePoint(void) {
    x =7;
    y = 21;}
PlanePoint:: PlanePoint(double nx, double ny) {
    x = nx;
    y = ny;}
PlanePoint:: PlanePoint(PlanePoint & ref_Point) {
    // створення нового об'єкта та копіювання даних в нього
    x = ref_Point.x;
    y = ref_Point.y;}
// зовнішня функція, що обчислює відстань від точки
// з координатами tx, ty до початку координат
// об'єкт PlanePoint mp є вхідним параметром функції
double GetLength(PlanePoint tmp);
int main() {
    clrscr();
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int d;
    double len;
    // створення об'єкта Ob1 за допомогою конструктора
    PlanePoint Ob1(5, 8);
    // створення об'єкта Ob2 - викликається
    // конструктор за замовчуванням
    PlanePoint Ob2;
    // Перевірка роботи конструкторів
    d = Ob1.GetX();
    cout<<"\n Працює функція Ob1.GetX():";
```

```
cout<<"\n Абсциса точки об'єкта Ob1 дорівнює "<<d<<endl;
cout<<"\n Натисніть будь-яку клавішу, щоб продовжити...\n";
getch();
d = Ob2.GetX();
cout<<"\n Працює функція Ob2.GetX():";
cout<<"\n Абсциса точки об'єкта Ob2 дорівнює "<<d<<endl;
cout<<"\n Натисніть будь-яку клавішу, щоб продовжити...\n";
getch();
// Побітове копіювання: конструктор копіювання не викликається
Ob2 = Ob1;
// дані скопіювались, але не за допомогою
// конструктора копіювання
d = Ob2.GetX();
cout<<"\n Працює функція Ob2.GetX():";
cout<<"\n після побітового копіювання Ob2 = Ob1:";
cout<<"\n Абсциса точки об'єкта Ob2 дорівнює "<<d<<endl;
cout<<"\n Натисніть будь-яку клавішу, щоб продовжити...\n";
getch();
// Код, що викликає до роботи конструктор копіювання
PlanePoint Ob3 = Ob1;
// Створюється новий об'єкт Ob3
// і в нього копіюється об'єкт Ob1,
// для чого викликається конструктор копіювання
d = Ob3.GetX();
cout<<"\n Працює функція Ob3.GetX() після ";
cout<<"\n виклику конструктора копіювання PlanePoint Ob3 = Ob1";
cout<<"\n Абсциса точки об'єкта Ob3 дорівнює "<<d<<endl;
cout<<"\n Натисніть будь-яку клавішу, щоб продовжити...\n";
getch();
PlanePoint Ob4(5,5);
// оголошено об'єкт Ob4 - екземпляр класу PlanePoint
// передача об'єкта Ob4 у функцію GetLength,
// викликається конструктор копіювання
len = GetLength(Ob4);
cout<<"\n Працює конструктор копіювання PlanePoint tmp, який";
cout<<"\n створює об'єкт tmp і копіює його параметри в об'єкт Ob4";
cout<<"\n функції GetLength(Ob4), що обчислює віддаль між точками.";
cout<<"\n Віддаль від точки Ob4(5,5) до початку координат: ";
cout<<"\n len = "<<len<<endl;
cout<<"\n Розв'язок задачі завершено!"<<endl;
getch();
return 0;
}
// зовнішня функція GetLength(PlanePoint tmp)
double GetLength(PlanePoint tmp) {
    double length;
    double tx, ty;
    tx = tmp.GetX();
    ty = tmp.GetY();
    length = pow((tx*tx + ty*ty),1./2.);
    return length;}

```

Результати тестування програми:

```
Працює функція Ob1.GetX():
Абсциса точки об'єкта Ob1 дорівнює 5
Натисніть будь-яку клавішу, щоб продовжити...
Працює функція Ob2.GetX():
Абсциса точки об'єкта Ob2 дорівнює 7
Натисніть будь-яку клавішу, щоб продовжити...
Працює функція Ob2.GetX():
після побітового копіювання Ob2 = Ob1:
Абсциса точки об'єкта Ob2 дорівнює 5

```

Натисніть будь-яку клавішу, щоб продовжити...
Працює функція Ob3.GetX() після
виклику конструктора копіювання PlanePoint Ob3 = Ob1
Абсциса точки об'єкта Ob3 дорівнює 5
Натисніть будь-яку клавішу, щоб продовжити...
Працює конструктор копіювання PlanePoint, який
створює об'єкт tmp і копіює в нього дані об'єкта Ob4.
Функція GetLength(Ob4) обчислює віддаль між точками.
Віддаль від точки Ob4(5,5) до початку координат:
len = 7.07107
Розв'язок задачі завершено!

Програма, код якої подано нижче, розроблена на базі класу Person з конструктором за замовчуванням – він у програмі не згадується - та двох зовнішніх функцій void rename(Person &x, string s) та void print(const Person &x). Як бачимо з прототипів обох функцій, вони в якості формального параметра мають посилання на об'єкт класу Person. У цій програмі потрібно змінити одне ім'я персони на інше.

Після запуску програми на виконання команда Person ob створює об'єкт ob, а команда ob.name="Петро" задає ім'я персони. Команда виклику зовнішньої функції rename(ob, s) передає об'єкт ob у зовнішню функцію void rename(Person &x, string s), яка змінить ім'я персони на інше і поверне його у головну функцію. Команда виклику зовнішньої функції print(ob) передає об'єкт ob у зовнішню функцію void print(const Person &x), яка виведе змінене ім'я персони на екран.

```
#include <string>
#include <iostream.h>
#include <conio.h>
#include <windows.h>
using namespace std;
class Person {
public:
    string name;
};
// Функції, що не входять в клас Person
void rename(Person &x, string s);
void print(const Person &x);
int main() {
    clrscr();
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    string s;
    // Задання імені персони за допомогою
    // об'єкта ob класу Person
    cout<<"\n Задання імені персони за допомогою";
    cout<<"\n об'єкта ob класу Person: \n";
    Person ob;
    ob.name="Петро";
    print(ob);
    cout<<"\n Натисніть будь-яку клавішу, щоб продовжити...\n";
    getch();
    // Зміна імені персони за допомогою
    // зовнішньої функції rename(ob, s):
    cout<<"\n Зміна імені персони за допомогою";
    cout<<"\n зовнішньої функції rename(ob, s): \n";
    s= "Микола";
    rename(ob, s);
    print(ob);
    cout<<"\n Розв'язок задачі закінчено!\n";
    getch();
    return 0;
}
void rename(Person &x, string s) {
```



```
x.name = s;  
};  
void print(const Person &x) {  
    cout<<"\n "<<x.name<<endl;  
};
```

Результати тестування програми:

```
Задання імені персони за допомогою  
об'єкта об класу Person:  
Петро  
Натисніть будь-яку клавішу, щоб продовжити...\n  
Зміна імені персони за допомогою  
зовнішньої функції rename(ob, s):  
Микола  
Розв'язок задачі закінчено!\n
```

Програма, код якої подано нижче, також розроблена на базі класу Person з конструктором за замовчуванням та двох зовнішніх функцій Person create1() та Person create2(). кожна з яких має такий самий тип, як і клас. Це означає, що ці дві функції повертають об'єкти. Як бачимо з прототипів обох функцій, вони не мають формальних параметрів. Призначення першої функції полягає у тому, що вона повертає об'єкт з порожніми (обнуленими) полями. Призначення другої функції полягає у тому, що вона повертає об'єкт з наперед заданими полями. У цій програмі потрібно обнулити або ж задати ім'я персони.

Після запуску

```
#include <string>  
#include <iostream.h>  
#include <conio.h>  
#include <windows.h>  
using namespace std;  
class Person {  
    public:  
        string name;  
};  
// Функції, що не входять в клас Person  
Person create1();  
Person create2();  
int main() {  
    clrscr();  
    SetConsoleCP(1251);  
    SetConsoleOutputCP(1251);  
    // Задання імені персони за допомогою  
    // об'єкта об класу Person  
    Person ob;  
    ob.name="Peter";  
    cout<<"\n Задання імені персони за допомогою";  
    cout<<"\n об'єкта об класу Person: \n";  
    cout<<" ob.name="<<ob.name<<endl;  
    cout<<"\n Натисніть будь-яку клавішу, щоб продовжити...\n";  
    getch();  
    // Передано об'єкт класу Person з обнуленими полями  
    cout<<"\n Передано об'єкт класу Person ";  
    cout<<"\n з обнуленими полями:"<<endl;  
    ob=create1();  
    cout<<" ob.name="<<ob.name<<endl;  
    cout<<"\n Натисніть будь-яку клавішу, щоб продовжити...\n";  
    getch();  
    // Передано об'єкт tmp класу Person з визначеними  
    // у функції create2() полями  
    cout<<"\n Передано об'єкт tmp класу Person з визначеними ";  
    cout<<"\n у функції create2() полями:"<<endl;
```

```
ob=create2();
cout<<" ob.name="<<ob.name<<endl;
cout<<"\n Розв'язок задачі закінчено!\n";
getch();
return 0;
}
// Функція createl поверне через своє ім'я
// об'єкт класу Person з обнуленими полями
Person createl() {
    return Person();
};
// Функція create2 поверне через своє ім'я
// об'єкт tmp класу Person з визначеними
// у тілі функції полями та їх значеннями
Person create2() {
    Person tmp;
    tmp.name = "Paul";
    return tmp;
};
```

Результати тестування програми:

Задання імені персони за допомогою

об'єкта ob класу Person:

ob.name=Peter

Натисніть будь-яку клавішу, щоб продовжити...

Передано об'єкт класу Person

з обнуленими полями:

ob.name=

Натисніть будь-яку клавішу, щоб продовжити...

Передано об'єкт tmp класу Person з визначеними

у функції create2() полями:

ob.name=Paul

Розв'язок задачі закінчено!

Висновок. В статті запропоновано комплекс програм мовою C++, за допомогою яких досліджується робота конструкторів різних типів на дані класу. Проблема вирішується з позицій об'єктно-орієнтованого програмування. Наведені коди запропонованих програм та результати їх тестування.

Список бібліографічного опису.

1. Архангельский А.Я. Программирование в C++ Builder – М.: ООО "Бином-Пресс", 2010. – 896 с.
2. Архангельский А.Я. Язык C++ в C++Builder. Справочное и методическое пособие –М: Бином, 2007. – 1012 с.
3. Бобровский С. Самоучитель программирования на языке C++ в системе Borland Builder 5.0. –М.: ДЕСС КОМ, I-PRESS, 2000. –272 с.
4. Браян В. Керніган, Денс М. Річі Мова програмування мовою C++. –Київ: КНЕУ, 2014. – 232 с.
5. Глинський Я.М., Анохін В.Є., Рязька В.А. C++ і C++Builder – Львів:СПД Глинський, 2011. – 192 с.
6. Грицюк Ю.І., Рак Т.Є. Програмування мовою C++. - Львів: ЛБУ БЖД, 2011. - 292 с.
7. Дейтел Х.М., Дейтел П.Дж. Как программировать на C++. –М.: М.: ООО "Бином-Пресс", 201. – 1456с.
8. В. Ермолаев, Т. Сорока C++ Builder: Книга рецептов. –КУДИЦ–Образ, 2006.-208с.

References.

1. Arkhangelsk A.Y. Programming in C++ Builder - M.: Binom-Press LLC, 2010. - 896 p.
2. Arkhangelsk A.Y. C++ language in C++ Builder. Reference and methodological manual –M: Binom, 2007. - 1012 p.
3. Bobrovsky S. Tutorial programming in C++ in the Borland Builder 5.0 system. –M.: DESS COM, I-PRESS, 2000. –272 p.
4. Brian V. Kernigan, Dance M. Richi Mova program my C++. –Kiev: KNEU, 2014. - 232 p.
5. Glinsky Ya.M., Anokhin V.E., Ryazhska V.A. C++ i C++ Builder - Lviv: SPD Glinsky, 2011. -- 192 p.
6. Gritsyuk Yu.I., Canser T.E. Programming my C++. - Lviv: LBU BZD, 2011. -- 292 p.
7. Daytel H.M., Daytel P.J. How to program in C++. –M.: M.: Binom-Press LLC, 201. - 1456s.
8. V. Ermolaev, T. Soroka C++ Builder: Recipe Book. – KUDIC – Obraz, 2006.-208s.

DOI: 10.36910/6775-2524-0560-2020-39-33

УДК: 004.01

Сіваковська Олена Миколаївна, к.т.н.,

<https://orcid.org/0000-0002-9300-0039>

Ліщина Валерій Олександрович, к.т.н., доцент,

<https://orcid.org/0000-0002-2371-3850>

Ящук Андрій Анатолійович, к.т.н.,

<https://orcid.org/0000-0003-4872-7949>

Матвій Юрій Ярославович, д.т.н., доцент,

<https://orcid.org/0000-0002-2236-2200>

Повстяна Юлія Славомирівна, к.т.н., доцент.

<https://orcid.org/0000-0001-5426-4157>

Луцький національний технічний університет, м. Луцьк, Україна

АНАЛІЗ ОСОБЛИВОСТЕЙ СТАНДАРТИЗАЦІЇ ПРОГРАМНИХ ПРОДУКТІВ ТА РОЗПОДІЛЕНИХ СИСТЕМ КЕРУВАННЯ

Сіваковська О. М., Ліщина В. О., Ящук А. А., Матвій Ю. Я., Повстяна Ю. С. Аналіз особливостей стандартизації програмних продуктів та розподілених систем керування. Здійснено аналіз становлення світової стандартизації. Означено етапи створення міжнародних організацій зі стандартизації. Розглянуто найактуальніші стандарти програмних продуктів та розподілених систем керування. Обґрунтовано переваги та недоліки процесу стандартизації.

Ключові слова: стандарти, стандартизація, управління якістю, контроль, виробництво, технології.

Сиваковская Е. Н., Лищина В. А., Ящук А. А., Матвиив Ю. Я., Повстяная Ю. С. Анализ особенностей стандартизации программных продуктов и распределенных систем управления. Осуществлен анализ становления мировой стандартизации. Отмечено этапы создания международных организаций по стандартизации. Рассмотрены актуальные стандарты программных продуктов и распределенных систем управления. Обоснованы преимущества и недостатки процесса стандартизации.

Ключевые слова: стандарты, стандартизация, управление качеством, контроль, производство, технологии.

Olena Sivakovska, Valerii Lishchyna, Andrii Yashchuk, Yurii Matviiv, Yuliia Povstiana. Analysis of Features of Software Standardization and Distributed Control Systems. The analysis of formation of world standardization is carried out. The stages of creation of international standardization organizations are marked. Current standards of software products and distributed control systems are considered. The advantages and disadvantages of the standardization process are substantiated too.

Keywords: standards, standardization, quality management, control, production, technologies.

Постановка проблеми. Сучасні умови бізнесу та ринку змушують кожне підприємство запроваджувати дієвий комплексний механізм управління якістю продукції та дотримуватись його вимог. Визначальними елементами цього специфічного менеджменту, що справляють найбільш істотний вплив на процес постійного забезпечення виробництва й постачання на ринок конкурентоспроможної продукції є стандартизація та сертифікація продукції.

Стандартизація – це встановлення й застосування єдиних правил з метою впорядкування діяльності в певній галузі. Здійснюється вона за певними принципами, і результати стандартизації відображаються в спеціальній нормативно-технічній документації [1].

Аналіз останніх досліджень та публікацій. Значну роль у виготовленні конкурентоспроможної якісної продукції, економному використанні ресурсів як основи сталого розвитку національної економіки відіграє стандартизація. Саме стандартизація є важливою частиною економічної політики розвинутих держав, потужним представником національних інтересів на міжнародному ринку.

Постійний систематизований розвиток напрямків стандартизації повинен стати невід'ємною частиною програм наукових досліджень, просування нових технологій. Міжнародні стандарти допомагають упорядкувати відносини між різними країнами з економічних і наукових питань та уникати додаткових витрат на отримання сертифікатів або проведення багаторазових випробувань [2].

Питаннями, пов'язаними з управлінням якістю продукції, знайшли відображення в працях вітчизняних і зарубіжних авторів, зокрема А.О. Болотнікова, Л.С. Кириченка, В.М. Мазура, Н.Г. Салухіної, А.А. Шапошникова та ін.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. Стандартизація бере свої витоки ще з 200-х рр. до н. е. Перший китайський імператор Цінь Ши Хуанг

(260 – 210 pp. до н. е.) стандартизував не тільки китайські символи, але і систему одиниць і вимірювань, а також валюту та ширину осей візка.

Процес війни теж долучився до становлення стандартизації. Більше 2500 років тому Геракліт заявив: «Війна є батьком і царем усіх речей». В Громадянській війні США (1861 – 1865) однією з причин перемоги Союзу над Півднем була стандартизація його залізничних колій. Проблема полягала в різній ширині залізничних колій. Залізнична мережа противника в основному мала широкий формат колії, а Північна Кароліна та Вірджинія мали стандартні колії. Південні залізниці на захід від Міссісіпі дуже відрізнялися форматом колій, роблячи їх ізольованими та від'єднаними.

Під час громадянської війни уряд Союзу визнав військово-економічні переваги наявності стандартизованої колії. Уряд працював з залізницями, щоб сприяти використанню найбільш поширеної на той час залізничної колії в США, яка була за розміром 4 фути і 8 ½ дюйма, що бере свій початок з Англії. Ця магістраль була погоджена для використання на Трансконтинентальній залізниці в 1864 році, і до 1886 року стала американським стандартом.

Іншим прикладом стандартизації під час воєн стали стандартизовані частини гвинтівки, що зробили їх взаємозамінними між даними типами зброї. Це була революційна ідея від Томаса Джефферсона та Елі Вітні, який був інженером-машиністом наприкінці XVIII століття. Також одним із ключових органів заснування стандартизації стала DIN (Німецька національна організація зі стандартизації) у 1917 р., яка на той час мала військовий досвід. Німецька промисловість хотіла оптимізувати виробництво під час Першої світової війни (1914 – 1918 pp.), оскільки велась і матеріальна битва між Німеччиною та Францією.

Існують також соціальні приклади успішної стандартизації. У 1904 році в Балтиморі спалахнула пожежа. Підкріплення з Нью-Йорка, Філадельфії та Вашингтона приїхали до Балтімору для боротьби з полум'ям. Після прибуття вони зрозуміли, що їх пожежні шланги не можуть бути підключені до пожежних гідрантів. У зв'язку з «отриманим уроком», США започаткували реалізацію багатьох проектів зі стандартизації.

У 1904 р. було створено Американський національний інститут стандартів (ANSI). Кількома роками раніше був заснований також Британський інститут стандартизації (BSI). Зростання міжнародного бізнесу, що вимагало стандартизації у всьому світі, призвело до народження Міжнародної електротехнічної комісії (IEC). Перше засідання відбулося в 1906 році. Спочатку її розташування знаходилося в Лондоні (Великобританія), а згодом комісія переїхала до своєї нинішньої штаб-квартири в Женеві в 1948 році.

Роком раніше, 25 країн заснували Міжнародну організацію зі стандартизації (ISO) для сприяння розвитку стандартизації і суміжних видів діяльності у світі з метою забезпечення міжнародного обміну товарами і послугами, а також розвиток співробітництва в інтелектуальній, науково-технічній і економічній галузях [3].

Попередник ISO був створений набагато раніше, у 1926 році, але його функціонування призупинилося під час Другої світової війни.

Сьогодні розподіл за класифікаційними ознаками – це уже історія, оскільки електрообладнання використовується в багатьох галузях промисловості і його потрібно стандартизувати. ISO також стандартизує електроніку, особливо для галузей, які не мають електричних витоків. Спочатку автомобілі не належали до електричної чи електронної сфери, ось тоді CAN і стандартизувався автомобільними технічними комітетами ISO.

ISO – міжнародна організація, метою діяльності якої є ратифікація стандартів, розроблених спільними зусиллями делегатів від різних країн [3]. Члени організації щорічно збираються на Генеральній Асамблеї, щоб обговорити стратегічні цілі ISO. Центральний секретаріат координує діяльність з стандартизації та публікує стандарти ISO. Є 250 технічних комітетів та тисячі підкомітетів, робочих груп та оперативних груп.

Міжнародна електротехнічна комісія (IEC) має аналогічну кількість технічних працівників. Близько 10 000 фахівців з електротехніки та електроніки в промисловості, уряді, академіях, дослідницьких лабораторіях та інших, зацікавлених у цій темі, що розробляють стандарти. Стандарти IEC мають число в межах від 60000 до 79999. IEC складається з членів, котрі називаються національними комітетами (НК). Кожна НК представляє електротехнічні інтереси своєї країни.

Міжнародна організація стандартизації (ISO) в рамках Технічного комітету TC97 (сьогодні це – ISO / IEC JTC1) також працювала над створенням мовних реляційних баз даних.

У 80-х роках як ANSI, так і ISO затвердили стандарти SQL (ANSI – в 1986 р., а ISO – на початку 1987 р.). Надалі з 1989 року обидві організації працювали над новими стандартами SQL, що призвело до появи стандарту SQL-92 (рис. 1).

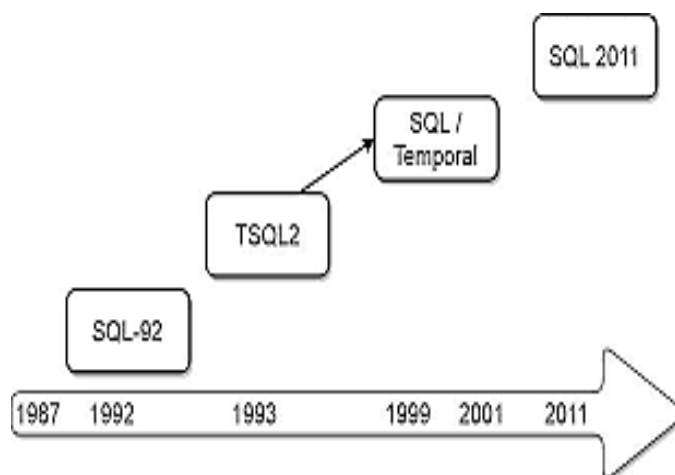


Рисунок 1 – Становлення стандарту SQL

В якості ще одного прикладу становлення стандартизації можна навести появу нині популярного UML (Unified Modeling Language). Основні розробки за методами об'єктно-орієнтованого аналізу і проектування з'явилися між 1988 і 1992 роках. До 1994 р була велика кількість неформальних лідерів розробників-практиків (близько півтора десятка), які просували свої методології.

Усі їхні методи були схожі між собою, лише відмінності між ними полягали у другорядних деталях. Нависала розмова про стандартизацію. Команда з OMG намагалася розглянути проблему стандартизації, але у відповідь отримала відкритий лист з протестом від всіх авторів. У 1996 р. три провідних фахівці в області об'єктно-орієнтованого аналізу і проектування Джеймс Рамбо, Гра-ді Буч, Івар Якобсон об'єдналися. На світ з'явився Уніфікований метод версії 0.8. А в 1996 р. ці фахівці працювали над своїм методом, який отримав назву Unified Modeling Language. У січні 1997 р. різні організації представили свої пропозиції по стандартизації методів, що передбачають в першу чергу можливість обміну інформацією між різними моделями. В результаті цієї праці сьогодні ми маємо єдину пропозицію – стандарт UML (рис. 2).

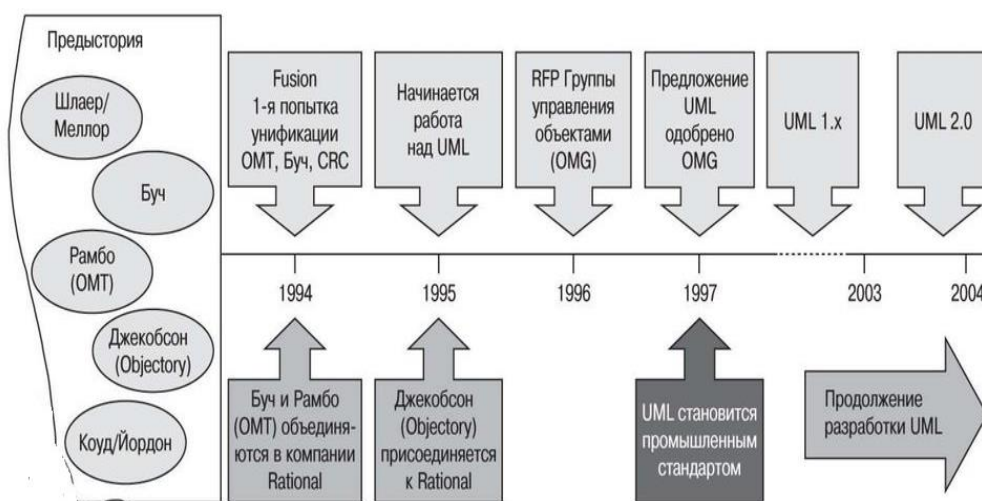


Рисунок 2 – Становлення стандарту UML

Варто також взяти до уваги CAN-стандарти, пов'язані з ISO та IEC. Протокол CAN вперше був описаний у специфікації, опублікованій Bosch. Тому все ще багато людей з тих пір використовують терміни "CAN 2.0A" і "CAN 2.0B". Однак у 1993 році був випущений стандарт ISO 11898, замінивши «своїх попередників», включаючи специфікацію Bosch. Стандарт ISO містив рівень зв'язку даних CAN і високошвидкісну передачу CAN. У 1995 році стандарт ISO 11898 був розширений доповненням, що описує формат розширеного кадру за допомогою 29-бітного ідентифікатора CAN. Через десять років після публікації ISO 11898, документ було розділено на частини. Перша частина містила рівень зв'язку даних та фізичні сигнали, а друга частина стандартизувала швидкісну передачу. Передача несправності та малопотужна передача перейшла в стандарт ISO 11898-3. Одночасно ISO розпочала стандартизацію взаємодії (зв'язку) вантажівки з причепом на базі CAN. Результати визначені у серії ISO 11992, яка включає, власне, фізичне рішення передачі та протокол вищого рівня на основі J1939. Ця серія також спеціалізується на групах параметрів передачі (сигналах, зібраних в повідомлення CAN). Усі ці стандарти публікуються під дахом Технічного комітету (TC) 22. В рамках цього TC також існує стандартизований транспортний протокол ISO (ISO 15765-2), який слугує базою для декількох стандартів (ISO 15765-4), пов'язаних з діагностикою викидів на основі CAN. Стандарт ISO 14229-3 базується на передачі діагностичних повідомлень по локальній мережі контролера через CAN [4]. Ще одна серія, пов'язана з CAN, – це ISO 16844, що стандартизує тахографічну комунікацію для комерційних транспортних засобів. Цей стандарт посиляється на європейські регламенти, але його промисловість не дуже любить, особливо це стосується далекобійників. Вони не люблять, щоб за ними велося спостереження в «електронному вигляді».

Одразу після 2000 року Bosch почав розширювати протокол CAN за допомогою протоколу, керованого часом. Питання про те, чи це протокол сеансового рівня, чи він взагалі не входить в протоколи взаємозв'язку відкритих систем (OSI), залишається без відповіді. Але він був стандартизований в ISO 11898-4 і відомий як TTCAN. На сьогоднішній день він не добився успіху в галузі. Деякі виробники чіпів реалізували його, але він не використовується.

Те ж саме сталося зі стандартом ISO 15745-2-2, що описує базову структуру XML для мереж на основі CAN; він також ніколи не застосовувався в системах промислової автоматизації. Це один із стандартів, що використовується лише для оформлення документів.

Не можна не згадати про CEN та Cenelec, європейські органи зі стандартизації, що були створені відповідно у 1961 та 1973 роках. Сьогодні вони працюють у тісній співпраці з IEC та ISO, щоб уникнути подвійної стандартизації. Або іншими словами, вони не намагаються «винаходити колесо».

Сьогодні ми не можемо уявити нашого життя без комп'ютерів та програмних засобів. Варто зупинитися і на цій невід'ємній сфері нашого життя. Термін "стандарти розробки програмного забезпечення" майже став оксимороном. Виробникам та розробникам комп'ютерів доводиться конкурувати на дуже високих рівнях, а наявність декількох фірмових вигадок часто є єдиною реальною перевагою одного розробника перед іншим. Тому утопія комп'ютерного світу, де все відповідає набору стандартів, ніколи не буде лише мрією. Однак є певні інші стандарти, які стають критичними, яких слід дотримуватися, якщо ви бажаєте конкурувати у світі фінансів та фінансових систем.

У 60-і і 70-і роки ХХ століття створення стандартів ставило користувачів в залежне від виробників становище при використанні основних засобів обробки даних і телекомунікацій. Важливий аспект сьогоднішньої роботи зі стандартизації – подолання цієї залежності через просування стандартних інтерфейсів. Довгий час такими стандартами були SQL (Structured Query Language) і мова діаграм Д. Росса SADT (Structured Analysis and Design Technique) [5].

Стандарт розробляється при дотриманні правил консенсусу в процесі відкритої дискусії, в якій кожен має можливість взяти участь. Жодна група не може діяти незалежно, створюючи стандарти для промисловості. Якщо будь-яка група постачальників створить стандарт, що не враховує вимоги користувачів, вона зазнає невдачі. Те ж саме відбувається, якщо користувачі створюватимуть стандарт, з яким не можуть або не будуть погоджуватися постачальники, – цей стандарт також не буде успішним.

Стандарти не можуть бути змінені, не пройшовши процес узгодження під контролем організації, що розробляє стандарти. Стандарти OSI (Open Systems Interconnection), Ethernet, POSIX, SQL і більшість стандартів мов – приклади такого роду стандартів.

Якщо говорити відверто, то можна сказати, що всі стандартизаційні заходи керуються інтересами окремих осіб або компаній – іноді і обох. Стандартизація має не лише технічний вимір; вона також має політичний характер.

Процес стандартизації іноді повільний порівняно з розвитком технології. Через це технології часто впроваджуються спочатку та стандартизуються пізніше. Стандартизація проходить повільно, оскільки всі сторони повинні мати можливість коментувати надані проекти та пропозиції. З позитивного боку, повільний процес уникає стандартизації технології, яка незабаром знову зникає.

Звичайно, процес стандартизації не повинен бути настільки повільним, щоб документ ніколи не публікувався. Тому ISO та IEC встановлюють терміни виконання проектів, які повинні бути дотримані. Однак читати стандарти нудно, особливо якщо ви не знайомі з конкретною мовою стандартизації. Але нам потрібні ці формальні правила щодо того, як писати стандарти. Нам потрібен стандарт, який говорить нам, як створити стандарти. Однією з причин є уникнення непорозумінь та неправильних тлумачень при перекладі стандарту на інші мови, крім двох офіційних: англійської та французької.

Висновки та перспективи подальших досліджень. Стандартизація достатньо ефективний засіб для оптимізації виробництва та посилення конкурентних позицій на ринку. У сучасному світі стандарти існують у динамічному та змінному середовищі. Стандартизація і сертифікація є важливим при виході продукції на нові конкуруючі ринки, що підвищують престиж підприємства, а також її конкурентоспроможність. Одним із перспективних напрямків подальшої роботи у цій сфері є детальне вивчення досвіду роботи у національних міжнародних організаціях зі стандартизації. Аналіз та дослідження міжнародних стандартів допоможе в становленні української стандартизації та управлінні якістю продукції сучасного ринку.

Список бібліографічного опису

1. Стандартизація та сертифікація продукції (2015) Економіка і підприємництво, менеджмент: електрон. версія підручника, URL : https://pidruchniki.com/80409/ekonomika/yakist_konkurentospromozhnist_produktsiyi (дата звернення: 30.04.2020). АСМ.
2. Зосереджуючись на майбутньому. Німецька стратегія стандартизації (2010) Стандартизація, сертифікація, якість, № 5, С. 23–28. АСМ.
3. Міжнародна організація зі стандартизації (2019) Вікіпедія: віл. енцикл., URL : https://uk.wikipedia.org/wiki/%D0%9C%D1%96%D0%B6%D0%BD%D0%B0%D1%80%D0%BE%D0%B4%D0%BD%D0%B0_%D0%BE%D1%80%D0%B3%D0%B0%D0%BD%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%8F_%D0%B7%D1%96_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D1%96%D1%97 (дата звернення: 30.04.2020). АСМ.
4. ГОСТ Р ИСО 15765-1-2014 (2015) Транспорт дорожный. Передача диагностических сообщений по локальной сети контроллера (DoCAN), Ч. 1, Москва, 12 с. АСМ.
5. Стандарти в області програмного забезпечення (2020) Лекції по дисципліні Сертифікація і стандартизація, URL: <https://werr.ru/standartizaciya2.php> (дата звернення: 30.04.2020). АСМ.

References

1. Products standardization and certification (2015) Economics and entrepreneurship, management, available at: https://pidruchniki.com/80409/ekonomika/yakist_konkurentospromozhnist_produktsiyi (accessed 30 April 2020). АСМ.
2. Focusing on the future. German standardization strategy (2010), Standardization, Certification, Quality, N 5, pp. 23–28. АСМ.
3. International Organization for Standardization (2019) Wikipedia, available at: https://uk.wikipedia.org/wiki/%D0%9C%D1%96%D0%B6%D0%BD%D0%B0%D1%80%D0%BE%D0%B4%D0%BD%D0%B0_%D0%BE%D1%80%D0%B3%D0%B0%D0%BD%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%8F_%D0%B7%D1%96_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D1%96%D1%97 (accessed 30 April 2020). АСМ.
4. GOST R ISO 15765-1-2014 (2015) Road transport. Transmission of diagnostic messages on the local area network of the controller (DoCAN), P. 1, Moscow, 12 p. АСМ.
5. Software Standards (2020) Lectures in the discipline “Certification and Standardization”, available at: <https://werr.ru/standartizaciya2.php> (accessed 30 April 2020). АСМ.

DOI: 10.36910/6775-2524-0560-2020-39-34

УДК: 004.415.25

Христинець Наталія Анатоліївна, ст.викладач

<https://orcid.org/0000-0002-4836-7632>

Скупейко Дмитро Степанович, студент

Луцький національний технічний університет

ОСОБЛИВОСТІ ПРОЕКТУВАННЯ КОМПОНЕНТ МІКРОЯДРА ОПЕРАЦІЙНОЇ СИСТЕМИ ЗАСОБАМИ GCC, GNU BINARY UTILITIES В КОМПОЗИЦІЇ З МОВАМИ АСЕМБЛЕРА ТА С

Христинець Н.А., Скупейко Д.С. Особливості проектування компонент мікроядра операційної системи засобами GCC, GNU Binary Utilities в композиції з мовами асемблера та С. Висвітлено питання розробки мікроядра операційної системи: керування пам'яттю, розробка початкового завантажувача та розглянуті питання написання макросів і спеціальних функцій у процесі програмування конфігураційного файлу. Сформовано схему збірки ядра від початкового коду до етапу емуляції.

Ключові слова: мікроядро операційної системи, Assembler, FASM, TASM, початковий завантажувач, обробник переривань, функція ядра операційної системи, бібліотека, макрос, GCC, GNU Binary Utilities, компіляція, системні ресурси, збірка ядра.

Христинець Н.А., Скупейко Д.С. Особенности проектирования компонент микроядра операционной системы средствами GCC, GNU Binary Utilities в композиции с языками асемблера и С. Освещены вопросы разработки микроядра операционной системы: управление памятью, разработка начального загрузчика и рассмотрены вопросы написания макросов и специальных функции в процессе программирования конфигурационного файла. Сформирована схема сборки ядра от исходного кода к этапу эмуляции.

Ключевые слова: микроядро операционной системы, Assembler, FASM, TASM, начальный загрузчик, обработчик прерываний, функция ядра операционной системы, библиотека, макрос, GCC, GNU Binary Utilities, компиляция, системные ресурсы, сборник ядра.

Khrystynets N., Skupeyko D. Features of designing components of the microkernel of the operating system by means of GCC, GNU Binary Utilities in the composition with the assembly language and C. The issues of developing the microkernel of the operating system are covered: memory management, bootloader development and macro writing and special functions in the process of programming the configuration file. The scheme of assembly of a kernel from a source code to an emulation stage is generated.

Keywords: operating system microkernel, Assembly, FASM, TASM, bootloader, interrupt handler, operating system kernel function, library, macro, GCC, GNU Binary Utilities, compilation, system resources, kernel compilation.

Постановка проблеми: Операційна система приховує апаратну складність, керує обчислювальними ресурсами та забезпечує ізоляцію та захист. Основними її компонентами є файлова система, обробник переривань, менеджер пам'яті, планувальник та драйвер пристрою. Ядро операційної системи, завантажене при запуску комп'ютера, керує ресурсами комп'ютера та обробляє запити від системних програм та додатків. Питання створення мікроядра операційної системи передбачає функціональні можливості та стан системи, необхідний для підтримки програм на базі користувача.

Аналіз досліджень. Більшість операційних систем, драйверів та утилітних програм написані комерційними організаціями, які поширюють виконувані версії свого програмного забезпечення - версії, які не можуть бути доповнені або змінені [1-3]. Відкритий код вимагає розповсюдження оригінальних матеріалів, які можна вивчити, змінити та доповнити, а результати буде знову вільно поширюватись. Самотужки створити операційну систему непросто. Навіть добре відомі операційні системи, такі як Android та Windows, створені командою з сотень людей. Це говорить про те, що створення операційної системи з нуля це ще й досить фінансово-затратний проект. Проте, якщо володіти переліком речей, які потрібно знати, перш ніж намагатися розпочати розробку операційної системи, то ядро операційної системи з мінімальними вимогами до функціонування написати все ж можна. Для цього необхідний час, бажання, основні знання з комп'ютерних наук, теорії комп'ютерного програмування та знання мов програмування низького та високого рівня.

Є багато принципів проектування операційної системи: принцип віртуалізації, генерованості, сумісності і т.д. Часто в літературі [4-5] вони розподілені по двом групам. Перша полягає в розмежуванні механізму та політики шляхом впровадження гнучких механізмів підтримки, друга - в оптимізації для загального випадку з огляду на те, де буде використовуватися ОС, що користувач захоче виконати на цій машині і які вимоги до навантаження. Існує три типи операційних систем, що часто використовуються. Перша - Monolithic OS, де вся ОС працює в просторі ядра і знаходиться в режимі супервізора. Друга - це модульна ОС, в якій деяка частина системного ядра буде розташована в незалежних файлах, званих модулями, які можна додати до системи під час виконання. І третя -

Micro OS, де ядро розбивається на окремі процеси, відомі як сервери. Деякі сервери працюють у просторі ядра, а деякі - у просторі користувача. Для створення операційної системи потрібно орієнтовно тисячі рядків коду. Їх розробка використовує мови програмування, такі як C, C++ та програмування на Assembler.

Виклад основного матеріалу й обґрунтування отриманих результатів. Фактично створення мікроядра має на меті створення програми для виконання процесором, або для керування його регістрами. Використовували мову Assembler, зокрема асемблер FASM. На цій мові повністю написаний код початкового завантажувача (рис. 1), у якому відбувалось, наприклад:

- завантаження глобальної таблиці дескрипторів сегментів gdt32 у спеціальний регістр gdt32;
- заборона переривань (cli);
- перехідно до захищеного режиму роботи процесора, зміна біта PE на 1 та близький стрибок на мітку start32 (Intel рекомендує так робити);
- зміщення на адресу кратну 16 (align 16), для швидшого звернення до даних;
- опис дескрипторів глобальної таблиці дескрипторів (мітка gdt32);
- опис розміру структури таблиці дескрипторів та її адреса;
- мітка start32: підготовка регістрів до стрибка на ядро, заповнення додаткового регістрів даними про ідентифікатор диску, список завантажених модулів ядра та карту оперативної пам'яті;
- стрибок на ядро (jmp 0FFC0000h).

```
610  
611  
612 ; Load values into GDTR (Global Description Table Register)  
613 lgdt [gdt32]  
614 ; Unable interrupts  
615 cli  
616 ; Switch to protected mode  
617 mov EAX, CRO  
618 or EAX, 80000001h  
619 mov CRO, EAX  
620 ; Switch to 32-bit code  
621 jmp 8:start32  
622 ; Description Segment Table for 32-bin kernel  
623 align 16  
624 gdt32:  
625 dq 0 ; NULL - 0  
626 dq 00CF9A000000FFFFh ; CODE32 - 8  
627 dq 00CF92000000FFFFh ; DATA32 - 16  
628 dq 00009A000000FFFFh ; CODE16  
629 dq 000092000000FFFFh ; DATA16  
630 gdt32:  
631 dw $ - gdt32 - 1  
632 dd gdt32  
633 ; 32-bit code  
634 use32  
635 start32:  
636 ; Set segment register and stack  
637 mov EAX, 16  
638 mov DS, AX  
639 mov ES, AX  
640 mov FS, AX  
641 mov GS, AX  
642 mov SS, AX  
643 mov ESP, 0FFFFFFFh  
644  
645 ; Put in DL the boot disk number  
646 mov DL, [disk_id]  
647 ; Put in EBX the address of the list of downloaded files  
648 mov EBX, module_list  
649 ; Put in ESI memory map address  
650 mov ESI, memory_map  
651  
652 ; Pass to the kernel  
653 jmp 0FFC0000h
```

Рисунок 1 – Фрагмент кінцівки коду початкового завантажувача

Команда *cli* забороняє переривання від контролера переривань, щоб не можна було перервати даний набір команд (на рисунку 1), щоб забезпечити атомарність операції та весь перехід до захищеного режиму роботи процесора.

Після написання на FASM початкового завантажувача, здійснювалась ще певна підготовка на асемблері для передачі параметрів у головну функцію ядра (*kernel_main*): надання відповідних параметрів стеку та виклик функції ядра. Коли з'явилась можливість переходу на головну функцію, було написано власно розроблену частину функціоналу стандартної бібліотеки C: у *stdlib* оголошено базові типи даних, різні макроси для роботи з портами вводу-виводу, створено функції для роботи з ділянками пам'яті та рядками. На основі цієї стандартної бібліотеки (рис. 2) базується робота модулів ядра та робота драйверів:

- оголошено тип даних *bool*;

- тип NULL;
- цілочисельні типи з різною розрядністю у версіях зі знаком та без;
- тип size_t в залежності від розрядності машини;
- макроси для визначення максимального та мінімального значення;
- макроси для роботи з портами I/O.

```

1 // standard library header file
2 # ifndef STDLIB_H
3 # define STDLIB_H
4
5 // define a boolean type
6 typedef enum
7 {
8     false = 0,
9     true = 1
10 } bool;
11
12 // define a null pointer
13 # define NULL ((void*)0)
14
15 // determine the types of specified bit depth
16 // the character "u" in the type name indicates an unsigned integer
17 typedef unsigned char uint8;
18 typedef signed char int8;
19
20 typedef unsigned short uint16;
21 typedef signed short int16;
22
23 typedef unsigned long uint32;
24 typedef signed long int32;
25
26 typedef unsigned long long uint64;
27 typedef signed long long int64;
28
29 // if our architecture is x86_64, then declare a non-signed integer
30 // type of maximum length for the current architecture
31 # ifdef __x86_64__
32     typedef uint64 size_t;
33 # else
34     typedef uint32 size_t;
35 # endif
36
37 // define useful macros min and max
38 // (return their minimum or maximum argument)
39 # define min(a, b) (((a) > (b)) ? (b) : (a))
40 # define max(a, b) (((a) > (b)) ? (a) : (b))
41
42 // macros for working with ports that are used to
43 // read or write values to I/O ports
44 # define outportb(port, value) asm("outb %b0, %w1::"a"(value), "d"(port));
45 # define outportw(port, value) asm("outw %w0, %w1::"a"(value), "d"(port));
46 # define outportl(port, value) asm("outl %0, %w1::"a"(value), "d"(port));
47
48 # define inportb(port, out_value) asm("inb %w1, %b0::"a"(out_value):"d"(port));
49 # define inportw(port, out_value) asm("inw %w1, %w0::"a"(out_value):"d"(port));
50 # define inportl(port, out_value) asm("inw %w1, %0::"a"(out_value):"d"(port));

```

Рисунок 2 – Фрагмент програмного коду хедера бібліотеки з модулями ядра і драйверами

Наприклад, перший макрос outportb(port, value) – робить зручним вивід у заданий порт (параметр port) значення розміром 1 байт (параметр value). А функція для роботи з ділянками пам'яті void memset (void *mem, char value, size_t count) – спрощує розміщення символу value у перших count позиціях ділянки mem та повертає вказівник на цю ділянку пам'яті mem (рис. 3):

```

42 // macros for working with ports that are used to
43 // read or write values to I/O ports
44 # define outportb(port, value) asm("outb %b0, %w1::"a"(value), "d"(port));
45 # define outportw(port, value) asm("outw %w0, %w1::"a"(value), "d"(port));
46 # define outportl(port, value) asm("outl %0, %w1::"a"(value), "d"(port));
47
48 # define inportb(port, out_value) asm("inb %w1, %b0::"a"(out_value):"d"(port));
49 # define inportw(port, out_value) asm("inw %w1, %w0::"a"(out_value):"d"(port));
50 # define inportl(port, out_value) asm("inw %w1, %0::"a"(out_value):"d"(port));
51
52 // declare prototypes of the functions of the standard library;
53 // first functions for working with sections of memory,
54 // then functions for working with strings;
55 // what each function does is described in the file stdlib.c
56 void memset(void *mem, char value, size_t count);
57 void memset_word(void* mem, uint16 value, size_t count);
58 void memcpy(void *dest, void *src, size_t count);
59 int memcmp(void *mem1, void *mem2, size_t count);
60 void *memchr(void *mem, char value, size_t count);
61
62 size_t strlen(char *str);
63 void strcpy(char *dest, char *src);
64 void strncpy(char *dest, char *src, size_t max_count);
65 int strcmp(char *str1, char *str2);
66 char *strchr(char *str, char value);
67
68 # endif
69

```

Рисунок 3 – Фрагмент коду з прототипами функцій

Робочий процес програмування конфігураційного файлу тривав досить довгий час, оскільки природно під час написання такого об'єму коду виникали питання вірного компілювання через не читання файлів з ядром (його не виявляла система), деякі питання збору образу системи, правильного читання карти пам'яті оперативної пам'яті і ще різного роду помилки з неправильною ініціалізацією стеку ядра та таблиці сторінок ядра. Проте, результат проектування з додатковим вивченням тем з відповідної літератури та інтернет-джерел [6-11] показав позитивні вирішення цих та інших питань.

Керування пам'яттю при розробці мікроядра зводиться до:

- ініціалізації самої пам'яті на основі даних з карти пам'яті RAM;
- виділення фізичних сторінок;
- звільнення фізичних сторінок;
- отримання об'єму вільної пам'яті;
- проектування фізичної сторінки на віртуальну адресу (необхідно для спрощення реалізації роботи функцій по виділенню та звільненню фізичних та віртуальних сторінок);
- отримання даних про конкретну віртуальну адресу.

Проектування фізичної сторінки (чи кількох сторінок) на задану віртуальну адресу здійснювалось за допомогою спеціально заданого блоку фізичної пам'яті, який має спеціально задану віртуальну адресу.

Функція набуває наступних параметрів:

1. Фізична адреса каталогу сторінок, тобто початкова адреса таблиці сторінок рівня 1.
2. Віртуальна адреса, яка буде використовуватись для спілкування зі спроектованою фізичною сторінкою.
3. Фізична адреса початкової фізичної сторінки, з цієї адреси бере початок виділення пам'яті.
4. Кількість фізичних сторінок, під які буде виділятися пам'ять.
5. Прапорець властивостей сторінок.

Функція повертає *bool* значення: *true* – якщо робота функції була успішною та *false* – у протилежному випадку. Сама функція – це цикл *for*, умовою якого, є «поки не виділиться необхідна кількість фізичних сторінок». В середині циклу відбувається розбивка віртуальної 32-бітної адреси на індекси 1-го рівня, 0-го рівня та зміщення – це необхідно для того, щоб знайти необхідну фізичну сторінку у таблиці сторінок. Далі, знаходиться необхідний блок фізичної сторінки (чи кілька блоків), якщо він доступний (тобто не був виділений раніше), то він позначається як виділений, увесь записується нулями і встановлюються його властивості (присутність в оперативній пам'яті, доступність до запису, і чи сторінка доступна для користувача). Якщо ж фізична сторінка була успішно виділена, то збільшується значення адреси фізичного блоку – переміщення до наступного блоку, щоб його виділити і т.д. Усі блоки фізичної пам'яті мають розмір 4КБ. Дана процедура відбувається по чергово, поки не виділиться необхідна кількість фізичних сторінок. В кінці функція повертає значення *true*.

Процес збірки образу системи від *source*-коду до вже готового образу системи здійснювався по чергово. У *Makefile* були написані (рис. 4) усі команди для роботи з *source*-кодом від компіляції до збору в єдиний бінарний файл ядра.

Порядок написання наступний:

- вказується цільова архітектура (параметр для лінкера);
- параметри для компілятора *GCC*, тут сказано, що об'єктні файли компілюються як 32-бітні, без стандартної бібліотеки мови *C* (адже використовується самописний функціонал частини стандартної бібліотеки *C*), з рівнем оптимізації коду *-O2*;
- перераховуються усі об'єктні файли (*all*), які необхідні для успішного лінування;
- відбувається безпосереднє лінування (команда *ld*).

Після цього перераховані команди асемблерування та компілювання *source*-коду перетворюються в об'єктні файли.

Загалом, структуру збірки можна подати такою схемою (рис. 4):

Кожен *source*-файл з кодом на *C* та асемблері компілюється/асемблерується в об'єктні файли (у випадку з кодом початкового завантажувача — створюється бінарний файл). Об'єктні файли (модулі ядра, драйвери) у свою чергу за допомогою лінкера *LD* із набору *GNU Binary Utilities* збираються у файл ядра, параметри для лінкера описані в окремому скрипті. Далі відбувається безпосереднє створення образу системи, який містить у собі початковий завантажувач (розташовується на початку образу) та саме ядро ОС. Код програми створення образу написано на *C*, сама програма приймає параметри з окремого *linux*-скрипту, де вказано що первинний завантажувач

має міститись у перших 512 байтах, за ним вторинний і далі вже ядро. Образ системи має формат .img і може вже бути запущений на якійсь віртуальній машині або на реальному ПК.



Рисунок 4 – Етапи збірки мікроядра

Таким чином змонтована найсуттєвіша частина операційної системи, що керує системними ресурсами – ядро, яке працює у привілейованому режимі.

Висновки та перспективи подальшого дослідження. З використанням мов програмування асемблера та C, засобів GCC та GNU Binary Utilities було змонтовано код, який містить необхідну мінімальну кількість функцій, даних і є найважливішою частиною для правильної реалізації операційної системи. Архітектура розробленого мікроядра невелика і ізольована, подальші ж дослідження можуть бути спрямовані на розробку великого програмного комплексу, що складатиметься з десятків модулів.

Список бібліографічного опису

1. Таненбаум Э. Современные операционные системы / Э. Таненбаум, П. Босх. – Питер, 2017. – 1120 с. – (4-е издание).
2. Лав Р. Ядро Linux: описание процесса разработки / Роберт Лав., 2019. – 496 с. – (3-е издание).
3. Назаров С. Операционные среды, системы и оболочки Основы структурной и функциональной организации / Сергей Назаров., 2007. – 504 с.
4. Дейтел Х. М. Операционные системы. Основы и принципы / Х. М. Дейтел, П. Д. Дейтел, Д. Р. Чофнес., 2016. – 382 с. – (3-е изд).
5. Рузайкин Г. Принципы проектирования и организации ОС [Электронный ресурс] / Геля Рузайкин // Открытые системы. СУБД, №6. – 1998. – Режим доступа до ресурсу: <https://www.osp.ru/os/1998/06/179610/>
6. Мартин Р. Идеальный программист. Как стать профессионалом разработки ПО / Роберт Мартин., 2019. – 224 с. – (Программирование).
7. Аблязов Р. З. Программирование на асемблере на платформе x86-64 / Р. З. Аблязов. – Москва: ДМК Пресс, 2016. – 302 с. – (Профессиональное программирование).
8. Пильщиков В. Н. Программирование на языке асемблера IBM PC / В. Н. Пильщиков., 2014. – 288 с. – (Диалог-МИФИ).
9. Эриксон Д. Хакинг. Искусство эксплойта / Джон Эриксон., 2014. – 496 с.
10. Максимов А. В. Оптимальное проектирование асемблерных программ математических алгоритмов. Лабораторный практикум / Александр Викторович Максимов. – Санкт-Петербург: Лань, 2017. – 128 с. – (Учебники для вузов. Специальная литература).
11. Как писать на асемблере в 2018 году [Электронный ресурс] – Режим доступа до ресурсу: <https://habr.com/ru/post/345748/>

References

1. Tanenbaum E. Modern operating systems / E. Tanenbaum, P. Bosch. - Peter, 2017. -- 1120 s. - (4th edition).
2. Love R. The Linux kernel: a description of the development process / Robert Love., 2019. - 496 p. - (3rd edition).
3. Nazarov S. Operating environments, systems and shells Fundamentals of the structural and functional organization / Sergey Nazarov., 2007. - 504 p.
4. Deitel H. M. Operating Systems. Fundamentals and principles / Kh. M. Daytel, P.D. Daytel, D.R. Chofnes., 2016. -- 382 p. - (3rd ed).
5. Ruzaykin G. Principles of design and organization of OS [Electronic resource] / Gelya Ruzaykin // Open Systems. DBMS, No. 6. - 1998. - Access mode to the resource: <https://www.osp.ru/os/1998/06/179610/>
6. Martin R. The ideal programmer. How to become a professional in software development / Robert Martin., 2019. -- 224 p. - (Programming).
7. Aplyazov R.Z. Programming in assembler on the x86-64 platform / R.Z. Aplyazov. - Moscow: DMK Press, 2016. -- 302 p. - (Professional programming).
8. Pilshchikov V. N. Programming in assembly language IBM PC / V. N. Pilshchikov., 2014. - 288 p. - (Dialogue-MIFI).
9. Erickson D. Hacking. The Art of Exploit / John Erickson., 2014. -- 496 p.
10. Maksimov A. V. Optimal design of assembler programs of mathematical algorithms. Laboratory workshop / Alexander Viktorovich Maksimov. - St. Petersburg: Doe, 2017. -- 128 p. - (Textbooks for universities. Special literature).
11. How to write in assembler in 2018 [Electronic resource] - Access to resource mode: <https://habr.com/en/post/345748/>

DOI: 10.36910/6775-2524-0560-2020-39-35

УДК: 004.2

Христинець Наталія Анатоліївна, ст. викладач

<https://orcid.org/0000-0002-4836-7632>

Михалик Артур Валентинович, студент

Міскевич Оксана Іванівна, асистент

<https://orcid.org/0000-0002-5009-2391>

Луцький національний технічний університет

ПРОДУКТИВНІСТЬ ТЕХНОЛОГІЇ CROSSFIRE X ПРИ НАВАНТАЖЕННІ ВІДЕОАДАПТЕРІВ МІКРОПРОЦЕСОРІВ AMD

Христинець Н.А., Михалик А.В., Міскевич О.І Продуктивність технології CrossFire X при навантаженні відеоадаптерів мікропроцесорів AMD. Розглянута технологія графічних процесорів компанії AMD: принципи організації та алгоритми побудови зображень. Проведені тестові випробовування процесорів архітектури Fusion для дискретних та інтегрованих зображень.

Ключові слова: архітектура мікропроцесорів, AMD, Radeon, CrossFire X, графічний процесор, відеоадаптер.

Христинець Н.А., Михалик А.В., Міскевич О.І. Производительность технологии CrossFire X при нагрузке видеоадаптеров микропроцессоров AMD. Рассмотрена технология графических процессоров компании AMD: принципы организации и алгоритмы построения изображений. Проведенные тестовые испытания процессоров архитектуры Fusion для дискретных и интегрированных изображений.

Ключевые слова: архитектура микропроцессоров, AMD, Radeon, CrossFire X, графический процессор, видеоадаптер.

Khrystynets N., Mikhalyk A., Miskevych O. Performance of CrossFire X technology when loading microprocessor video adapters on the AMD. The technology of AMD graphics processors is considered: principles of organization and algorithms of image construction. Tests of Fusion processors for discrete and integrated images were performed.

Keywords: microprocessor architecture, AMD, Radeon, CrossFire X, graphics processor, video adapter.

Постановка проблеми.

На сьогодні технології сучасних процесорів розвиваються дуже стрімко. З кожною новою архітектурою з'являються нові технології і інструкції, що максимально адаптують процесори під виконувані задачі та покращують їх енергоефективність. Тому, очевидно, що виробництво кожного наступного покоління відеокарт також прогресує. Головною проблемою для типового споживача стає вартість нових, більш потужних відеокарт. Проблема, яка досліджена у роботі, полягає в аналізі сумісної роботи кількох відео карт різних поколінь та дослідження продуктивності такої системи в цілому. Проведення аналізу поєднання таких структур, а також оцінка складності реалізації є досить актуальним питанням.

Аналіз досліджень.

Дослідження CrossFire X проводили переважно іноземні ІТ-інженери. Дану технологію розібрав Wes Compton на прикладі зв'язки двох відеокарт AMD RX590. Він дійшов до висновку, що приріст до продуктивності відбувається приблизно на 35% [1]. Така перевага компенсується складністю реалізації та підтримкою програмних продуктів даною техноогією. Тому, для простих користувачів це застосування вкрай недоцільне.

Якщо звернутись до інтернет джерел [2-5], то інформації в СНД-регіоні про дану технологію та її тестування обмаль, а якщо вона і зустрічається у публікаціях, то перефразована чи просто дубльована з іноземних досліджень та статей. Якщо говорити про саму компанію AMD, то вона заявляє, що завдяки технології CrossFire X збільшується продуктивність графічної системи персонального комп'ютера до 40%. Щоправда, це ствердження працює лише тоді, коли будуть виконуватись наступні умови:

- програма, яку ви використовуєте, повинна мати підтримку двох графічних ядер;
- робота лише в повноекраному режимі;
- наявність апаратної частини, яка підтримує дану технологію.

Переглянувши форуми та пабліки за данною тематикою, на Always more digital знайдено статтю про технологію CrossFire X, у якій описується тестування відеокарток за усіма наявними характеристиками. За результатами огляду можна зробити наступні висновки: продуктивність відеокартки з CrossFire X зростає приблизно на 30% поряд із збільшенням шумових ефектів, енергоспоживання. Важливою особливістю досліджень є те, що системи кількох відеокарт

попереднього покоління виявляють більшу продуктивність, ніж споживання більш потужної однієї нової відеокарти. Щоправда тести в іграх дуже сильно різняться, але це пояснюється оптимізацією ігор під певні відеокарти, точніше під компанію виробника. Тобто, ігрові додатки, що оптимізовані під AMD, мають серйозний приріст до fps, а ті, які оптимізовані під Nvidia – мають не значний приріст.

Технологія CrossFire X підтримується найбільш потужними відеокартами AMD починаючи від архітектури 10/Bulldozer, тому, щоб досягти ще більшої їх потужності, використовують зв'язки кількох відеокарт для професійної обробки графіки.

Виклад основного матеріалу й обґрунтування отриманих результатів.

«AMD CrossFireX (укр. перехресний вогонь) – технологія, що дозволяє одночасно використовувати потужності від двох до чотирьох графічних процесорів відеокарт Radeon для побудови тривимірного зображення одночасно» [3].

Існує два способи організації CrossFireX-системи.

Внутрішній спосіб – цим методом з'єднання здійснюється за допомогою гнучкого містка з аналогічною назвою CrossFireX. Якщо відеокарт більше двох, тоді решта з'єднуються послідовно, жодних спеціальних інших містків не потрібно.



Рисунок 1 – Дві відеокарти AMD Radeon HD 5870, з'єднані спеціальними гнучкими містками CrossFireX

Програмний метод – відеокарти не з'єднуються між собою фізично, а обмін даними відбувається через шину PCI Express x16, а взаємодія реалізована за допомогою програмного забезпечення. Значним недоліком такого методу є втрата продуктивності на 10-16% порівняно з внутрішнім способом. Метод дуже рідко реалізується, організовується здебільше лише для малопродуктивних відеокарт. Для високопродуктивних даних метод реалізувати неможливо через те, що налаштування драйвера не підтримують без внутрішнього з'єднання таку структуру.

Особливість Crossfire X полягає в тому, що він не сповільнює швидкість тактової частоти GPU, коли запускається разом з іншими графічними процесорами. Наприклад, якщо запускаємо Radeon 7950 та 7870 разом у конфігурацію CrossFire X, це значно відрізняється від зв'язок SLI, які вимагають поєднання однакових GPU. Фактично, міст CrossFire- це просто короткий кабель, який пролягає між двома або більше відеокартами, щоб з'єднання xfire могло відбуватися. Той факт, чи потрібен мостовий кабель між картками, залежить від графічних процесорів та комбінацій цих карт, які потрібно об'єднати разом. Наприклад, більшості серіям 280x потрібні з'єднання з мостом CrossFire, а картки серії 290 та 290x мають підтримку CrossFire без мостів.

Провівши дослідження та знайшовши інформацію на офіційному сайті AMD про технологію і про той факт, що вона може принести приріст до потужності відеоадаптерів через їх поєднання до 40%, вирішено провести ряд натурних експериментів, щоб з'ясувати, як ця технологія працює в реальних умовах на прикладі роботи відеоігор з різною текстурою, трасуванням променів у сучасних 3D іграх, завантаженістю тайтлів різних категорій – від А до AAA та іншими відео-характеристиками.

Подальші тести, аналіз стосуються відео картки AMD Radeon HD 5870. Системні вимоги, характеристики даної картки і відеопроцесора грають важливу роль в результатах досліджень. Споживання енергії та роз'єми ATI Radeon HD 5870 вимагають значної електро-потужності для

роботи. Для цього потрібні два шестиконтактні підключення живлення PCI Express, кожен з яких повинен бути розрахований на 75 Вт потужності. Враховуючи власну структуру будови материнської плати, що відповідає додатковим компонентам системи, комп'ютер, який використовує карту, повинен мати джерело живлення потужністю не менше 500 Вт. У випадку конфігурації подвійної картки з такою ж технологією CrossFire X, компанія AMD дозволяє використовувати їх разом. Тобто, дві карти Radeon HD 5830/5870. Можливість розподілу обов'язків щодо обробки графіки між двома пов'язаними картками може майже подвоїти графічну продуктивність комп'ютера. Важливо, що для використання двох карт потрібна сумісна з CrossFire материнська плата, інша карта та блок живлення потужністю не менше 600 Вт із загальною кількістю чотирьох шестиконтактних роз'ємів. Для найкращої роботи AMD рекомендується сертифікований блок живлення. Справа у тому, що сертифіковані джерела живлення пройшли безліч тестів, щоб пересвідчитись, що вони дійсно можуть забезпечити стабільну подачу сигналу живлення, достатню для задоволення потреб відеокарт ATI у конфігурації подвійних карт CrossFire. У наш час агато виробників, включаючи Antec, Corsair, PC Power and та інші виробляють сертифіковані джерела живлення. Інші системні вимоги: Radeon HD 5870 не просто вимагає багато електроенергії. Для найкращої роботи потрібен дійсно потужний комп'ютер. По-перше, комп'ютер повинен мати один канал PCI Express x16 для кожної карти, яку він буде використовувати. По-друге, комп'ютер повинен мати не менше 1 Гб системної пам'яті. Також знадобиться охолоджувальний блок, який може вмістити фізично велику карту та відводити додаткове тепло, яке вона виробляє.

Для тестів було обрано Tess Mark та дві гри, які, за заявою розробників, працюють з технологією AMD – це World of Tanks і Call of Duty: Black Ops 2.

Першою грою була World of Tanks. Є різна кількість варіантів збірок цієї гри. Ці збірки оптимізовані для конкретної роздільної здатності екрана. Можна отримати збірку, призначену для 4K (з більшою роздільною здатністю) та гру в 1080p (нижча роздільна здатність) без проблем, але очікувати якісну збірку, призначену для 1080p на 4K марно. Для прикладу, відповідність збірки цієї гри з роздільною здатністю екрану ПК 1920 x 1080 дасть найоптимальніші показники з плавними 60 fps.

Результат роботи при низьких налаштуваннях графіки World of Tanks становить 80fps, при аналогічних налаштуваннях інтегрована відеокарта показала результати 55fps, а дискретна відеокарта – 72fps. Отже, ми бачимо приріст fps на 5,76%, що є непоганим результатом.

Але цей приріст відразу анулюється значним мінусом – це спотворення графіки. Програмно реалізація технології (у цій грі) дуже на поганому рівні. Менш потужний інтегрований відеоадаптер, не встигає за обробкою зображення дискретного чіпа. В результаті ділянки кадру за які відповідає інтегрований чіп стають просто чорними, що сильно спотворює враження від кадру. Хоча AMD стверджували, що їх технологія буде працювати навіть з різницею в два рази між відеоадаптерами.

До всього цього додається ще один мінус. Через погану синхронізацію чіпів між собою fps нестабільний і просідає на 25% на активних сценах Стандартного бою з технікою Рангового бою з максимальним навантаженням сцен (рис.1).



Рисунок 2 – Фрагмент роботи Dual Graphics в грі World of Tanks (сцена

Зазначимо, що у ігровому режимі online важлива характеристика fps, і чим більший цей показник (80-100), тим швидша і якісніша ігрова картина, а у ігровому режимі single роль відеоадаптера і усіх його характеристик: частоти, об'єму, суміжних технологій і інструкцій виходить на перший план навіть за наявності fps 30-60 одиниць. Також була врахована характеристика fps серверів. Зокрема, налаштування максимальних гравців має бути оптимізовано, пов'язане з підключенням по висхідній лінії зв'язку до сервера. Протестивши вхідну та вихідну швидкість каналу з'єднання з інтернет, фактично отримаємо цифри про отримання даних з серверу. Завантаження, або вихідна швидкість показує, наскільки швидко сервер може надсилати дані гравцям.

В основному для підключення до Інтернету (біт/с):

128k uplink = 4 гравці (2 гравці за srcds)

256k uplink = 7 гравців (4 гравці за srcds)

320k uplink = 9 гравців (6 гравців за srcds)

512k uplink = 14 гравців (9 гравців за srcds)

768k uplink = 21 гравець (14 гравців за srcds)

1024k uplink = 28 гравців (19 гравців за srcds – зверніть увагу, що 16 гравців максимум для HL2DM)

1140k uplink = 32 гравці (максимум для HLDS) (21 гравець для CS: S)

1708k uplink = 32 гравці для CS: S (макс.)

Звичайно, оптимальне налаштування може відрізнятись залежно від з'єднання. Місцеві гравці в локальній мережі не включаються до цього рахунку. Хорошим правилом для HLDS є 35,6 Кбіт на гравця. Хорошим правилом для SRCDS (Source Games) є 53,4 КБ на гравця (приблизно в 1,5 рази більше, ніж HLDM).

Переходимо до аналізу другої гри, а саме Call of Duty: Black Ops 2. Дану гру було обрано тому, що не тільки розробники гри заявили про підтримку технології Cross Fire X, а й сама компанія AMD.

Дана гра на низьких налаштуваннях на інтегральній графіці показує 54 fps, а на дискретній – 67 fps. Слід зауважити, що це були стабільні fps. При увімкнені технології результат становив 78 fps. Що становить на 8,58% більше за дискретну графіку і на 18,72% більше за інтегровану графіку.

При довгій активній роботі гри fps залишався стабільним. Переходим до якості картинки. Кадр стабільний, цілісний на відміну від попередньої гри де спостерігались деякі мікрофрзи та страттери (рис. 2).



Рисунок 2 – Фрагмент кадру гри Call of Duty: Black Ops 2.

Тестування поведінки і продуктивності відеосистеми з технологією CrossFire X проводили за допомогою тестуючого комплексу Tess Mark.

Вибір серед різноманіття тестуючих програм саме TessMark обумовлено тим, що це тестовий показник OpenGL та яскрава демонстрація технологій, яка використовує нові функції апаратної тесселяції GPU, що підлягає API OpenGL 4 і доступна навіть на найсучасніших відеокартах, таких як

NVIDIA GeForceGTX480 або ATI RadeonHD5870. Високий рівень тесселяції вимагає великих процесорних потужностей GPU.

На рисунку 3 зображено параметри, за яких відбувалось тестування (а) і результати тестування (б):

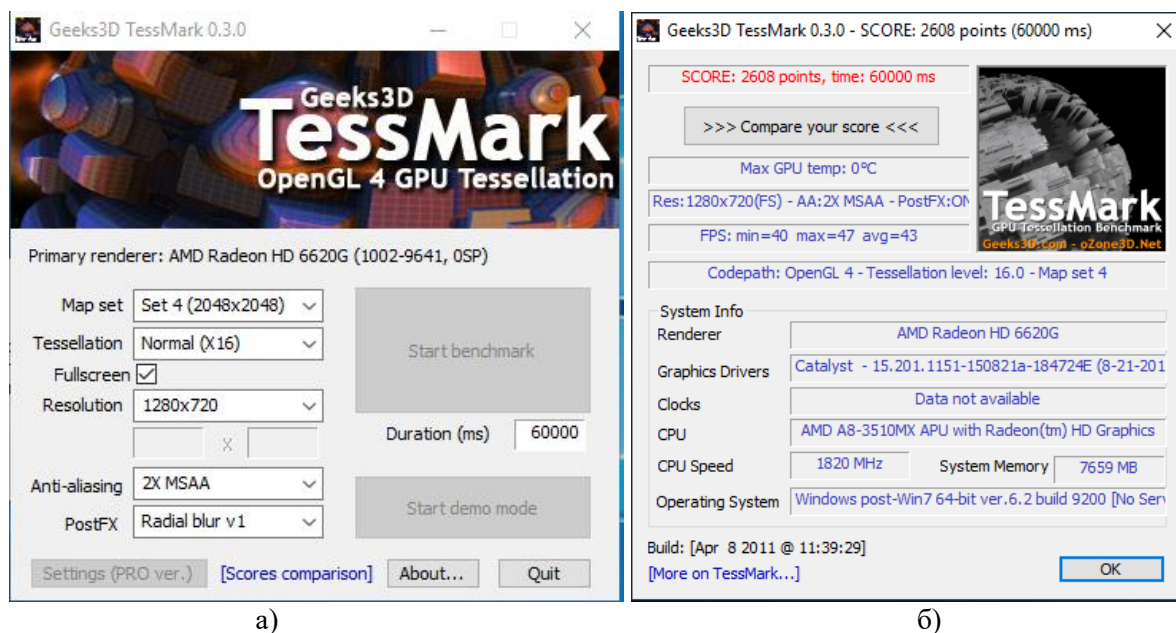


Рисунок 3 – Тестування поведінки відеоадаптера: а) початкові параметри; б) результат тесту.

З тестів видно, що застосування технології Cross Fire X (Dual Graphics) має сенс. Можна збільшити продуктивність графіки, але існує дуже багато дрібних нюансів, серед яких: працює технологія лише в повноекранному режимі, багато програмного забезпечення (особливо старого) не підтримують технологію.

Висновки та перспективи подальшого дослідження. На підставі проведених досліджень можна зробити висновки щодо продуктивності і доцільності використання технології AMD CrossFireX. Розглянуто принцип роботи технології на прикладі інтегрованої та дискретної графіки. Проведено дослідження алгоритмів роботи технології та принципи підключення і дослідження продуктивності відеоадаптерів окремо та з застосуванням технології Cross Fire X (Dual Graphics). У порівнянні можна стверджувати що для середнього користувача мабуть пріоритетніше працювати з однією більш продуктивною картою, ніж об'єднувати їх у зв'язки з даною технологією. Тести проводились для процесорів архітектури Fusion, подальші дослідження продуктивності технології можуть стосуватись і систем з архітектурами нижчого техпроцесу.

Список бібліографічного опису

1. AMD RX590 Crossfire Testing [Електронний ресурс] – Режим доступу до ресурсу: <https://lanoc.org/review/video-cards/7882-amd-rx590-crossfire-testing>
2. Jargosch R. Advanced Micro Devices (AMD). Patent Landscape Analysis / R. Jargosch, J. Jurich. // IPGenix LLC. – 2014.
3. AMD CrossFireX [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/AMD_CrossFireX
4. AMD Crossfire™ [Електронний ресурс] – Режим доступу до ресурсу: <https://www.amd.com/en/technologies/crossfire>
5. Lander S. What Is ATI CrossFireX Technology [Електронний ресурс] / Steve Lander – Режим доступу до ресурсу: <https://smallbusiness.chron.com/ati-crossfirex-technology-58205.html>
6. Відеокарты [Електронний ресурс] – Режим доступу до ресурсу: <http://www.facebook.com/Always.More.Digital>

References

1. AMD RX590 Crossfire Testing [Electronic resource] – Resource access mode: <https://lanoc.org/review/video-cards/7882-amd-rx590-crossfire-testing>
2. Jargosch R. Advanced Micro Devices (AMD). Patent Landscape Analysis / R. Jargosch, J. Jurich. // IPGenix LLC. – 2014.
3. AMD CrossFireX [Electronic resource] – Resource access mode: https://uk.wikipedia.org/wiki/AMD_CrossFireX
4. AMD Crossfire™ [Electronic resource] – Resource access mode: <https://www.amd.com/en/technologies/crossfire>
5. Lander S. What Is ATI CrossFireX Technology [Electronic resource] / Steve Lander – Resource access mode: <https://smallbusiness.chron.com/ati-crossfirex-technology-58205.html>
6. Videocards – Resource access mode: <http://www.facebook.com/Always.More.Digital>

CONTENTS

AUTOMATION AND MANAGEMENT

<i>AUTOMATION AND MANAGEMENT</i>	
L. V. Bulatetska, V. V. Bulatetsky. Features of studying the language of SQL queries in the profile course of informatics of institutions of general secondary education.	5
Holovachuk I., Velychko V., Burban O., Bobokalo S. Substantiation of physical parameters and software of 3D-holographic projections.	11
Zhurilo A. G., Sivak E. M. Building the perspective of a circle that belongs to the subject plane.	17
Kaganyuk A.K., Melnyk V.M. A Mathematical model for calculating controller parameters for a moving object.	22
Kasianchuk Dmytro. Comparative analysis of documentation generators.	30
Klekho O. , Lekhitskyi T. Formation of professional competences of students of pedagogical college in the conditions of vocational training	35
Kostiuchko S.M., Kyryliuk L.M., Kalys O.V., Sibanda Z.F., Havryliuk S.A. The monodromy matrix construction for executive object of a nonlinear system.	40
Kuzmina O.M., Yaremko S.A. Improved business process management based on simulation modeling.	44
Lepky MI, Podoliak VM, Koshelyuk V.A. Hardware and software for creating and using 3D blueprints	50
V.V. Lyshuk, M.M. Yevsiuk, Y.R. Selepyna, N.Yu. Kopylets'. Mathematical models of converting equipment devices.	55
Lupenko S.A., Lytvynenko Ia.V., Stadnyk N.B., Zozulia A.M., Sverstiuk A.S. Conditional cyclic random process of a discrete argument as a generalized mathematic model of cyclic signals with double stochasticity.	60
Mamchych T. I., Myroniuk L. P., Royko L. L. Information and communication technologies usage experience at teaching of mathematical disciplines in the terms of the distance learning.	70
Martynenko A.A., Moroz B.I., Hulina I.G. An intelligent decision support system for cultural property identification.	78
Marchenko O.O., Marchenko O.I. Resource model for heterogenous distributed computer system with local connections and its graph.	83
Poplavskaya G.V. Analysis of the use of the Google Classroom online service for distance learning.	89
Prykhodko A.O., Prykhodko O.S. Development of a universal device for mechanical testing of materials.	94
Savytskyi T.P., Orlova M. M. Using Artificial Intelligence for a Smart Home System.	99
Fedotov V. V. Mathematical model of spin photocurrent excitation in of antiferromagnetic isolators nanostructures.	105
Chybiriak Ya. I., Konoplianchenko Ie. V., Marchenko A. V. Technological patterns and mathematical models of the synthesis of a rational sequence of product assembly.	110

<i>INFORMATICS AND COMPUTER SCIENCE</i>	
Voitenko I., Orlova M. Analysis of the security vulnerability of BGP functioning and the reasons for complexity of dealing with them.	117
Hryniuk S.V., Polishchuk M.M. Use information encryption technology for secure network transmission.	122
Kozubtsova L. M. Improved method of diagnostics of cybernetic protection of the information system taking into account destructive cybernetic influences.	127
Kostiuchko S.M., Sahniuk A.A., Melnyk K.V. Bypass site protection by means SQL injections and protection against them.	136
Kurdus Anastasiia. Comparative analysis of online compilers.	141
Martsenyuk V.P., Didmanidze I.Sh., Sverstiuk A.S., Andrushchak I.Ye., Rud K.I. Automated method of building exploits in analysis software testing.	146
Melnyk K., Melnyk V.M., Hryhorshyn A. Automatic collection of information (parsing) in the network.	151
Melnyk K.V., Kostiuchko S.M., Melnyk D.S. Optimization of accounting and financial analytics using a mobile application based on the Android operating system.	157
V. Melnyk, K. Melnyk, O. Kuzmych, N. Bahniuk, O. Kravets. Internal and external communication performance improving research on a cluster of communicated virtual machines.	162
Miskevych O, Bahniuk N., Khrystynets N., Marchevska O. Automation of defective products detection by machine learning methods.	175
Muliar V. P. Developing JavaFX applications using Scene Builder.	181
Nikolina I., Hulivata I. Modeling cybercrime as a threat to the digitalization of the economy.	190
Pekh Petro, Kostyuk Yuri, Kravchenko Maxim. On the question of designing classes with constructors of different types.	197
Olena Sivakovska. Valerii Lishchyna, Andrii Yashchuk, Yurii Matviiv, Yuliia Povstiana. Analysis of Features of Software Standardization and Distributed Control Systems.	203
Khrystynets N., Skupeyko D. Features of designing components of the microkernel of the operating system by means of GCC, GNU Binary Utilities in the composition with the assembly language and C.	208
Khrystynets N., Mikhalyk A., Miskevych O. Performance of CrossFire X technology when loading microprocessor video adapters on the AMD.	213

ВИМОГИ ДО СТРУКТУРИ ТА ОФОРМЛЕННЯ МАТЕРІАЛУ СТАТЕЙ

- **Наукова стаття обов'язково повинна мати наступні необхідні елементи:**
 - 1) **постановка проблеми** у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями;
 - 2) **аналіз останніх досліджень і публікацій**, в яких започатковано розв'язання даної проблеми і на які спирається автор,
 - 3) **виділення невирішених раніше частин загальної проблеми**, котрим присвячується означена стаття;
 - 4) **формулювання мети дослідження** (постановка завдання);
 - 5) **виклад основного матеріалу дослідження** з повним обґрунтуванням отриманих наукових результатів;
- **висновки** з даного дослідження, у тому числі з науковою новизною і
- б) **перспективи подальших досліджень** у даному напрямку.
- Статтю можна подавати українською, російською або англійською мовами. Вона повинна бути набрана у текстовому редакторі MS WORD 03/07/10 і надрукована на лазерному або струменевому принтері на білих листах формату А4 (297×210 мм). **Нумерацію сторінок** не виконувати. **Обсяг статті** 5-10 сторінок (не менше).
- **Параметри сторінки**. Верхнє, нижнє та праве поле – 1,5 см, лівє – 2 см. Від краю до верхнього колонтитула – 1,25 см, нижнього – 1,25 см.
- **Шапка статті**. УДК, ORCID (якщо є), автори (ім'я та прізвище повністю), місце роботи кожного автора. Назва організації та назва статті набираються з нового рядка шрифтом Time New Roman Суг розміром 11 пт з одинарним міжрядковим інтервалом та вирівнюються по лівому краю. Назва статті розміщується через один рядок нижче назви організації (розмір шрифту 11 пт з напівжирним виділенням та вирівнюванням по центру).
- **Анотації** (українською, російською та англійською мовами) повинні містити прізвища та ініціали авторів, назву статті та короткий її зміст і розміщуються через один рядок нижче назви статті та набираються з абзацного відступу 1 см шрифтом Time New Roman Суг розміром 9 пт з одинарним міжрядковим інтервалом і вирівнюються по ширині. Нижче анотацій обов'язково вказуються **ключові слова**.
- **Основний текст** розміщується на через один рядок нижче анотацій, набирається з абзацного відступу 1 см шрифтом Time New Roman розміром 11 пт з одинарним міжрядковим інтервалом та вирівнюється по ширині.
- **Формули** набираються у редакторі формул MS WORD (використовувати шрифти: Symbol, Time New Roman Суг; розміри шрифтів: звичайний 12 пт, крупний індекс 7 пт, дрібний індекс 5 пт, крупний символ 18 пт, дрібний символ 12 пт). Формула вирівнюється по центру і не повинна займати більше 5/6 ширини рядка.
- **Ілюстрації**, що присутні у статті, необхідно розташовувати у тексті по центру, вирівнюючи підписи по центру (Рис. 1. Назва). Другий екземпляр ілюстрації необхідно подати на окремому листі. Ілюстрації повинні бути чіткими та контрастними.
- **Таблиці** потрібно розташовувати у тексті по центру, причому їх ширина повинна бути на 1 см менша ширини рядка. Над таблицею ставиться її порядковий номер і назва (Таблиця 1. Назва) та вирівнюється по центру.
- **Посилання** на ту чи іншу роботу повинні позначатися в тексті у квадратних дужках за порядковим номером у списку літератури в кінці статті; посилання на джерела статистичних даних обов'язкові; посилання на публікації дослідників обов'язкові; посилання на підручники, навчальні посібники, газети і ненаукові журнали – небажані; посилання на власні публікації допускаються тільки у випадку крайньої необхідності; роботи авторів, на прізвища яких є посилання в тексті, мають бути в списку літератури до **цієї статті**.
- **Список бібліографічного опису та References**. Список літератури («References») потрібно приводити повністю окремим блоком, повторюючи список літератури, який подається українською / російською мовою, незалежно від того, є в ньому іноземні джерела чи ні. Тобто, після статті подається 2 списки: «Список бібліографічного опису» (звичайний список літератури) і «References» (список для міжнародних БД). Необхідно в опис джерела вносити всіх авторів, не скорочуючи їх до трьох, як це рекомендовано діючими у нас державними стандартами. References - повинен бути укладений англійською мовою або транслітерований. Оформлювати згідно з одним із найбільш уживаних у світі стандартів: APA – American Psychological Association; CBE – Council of Biology Editors, Citation-Sequence; Chicago (Author-Date System); Harvard; Harvard – British Standard; MLA (Modern Language Association) – Single Spaced Reference List; NLM – National Library of Medicine; Uniform Requirements for Manuscripts Submitted to Biomedical Journals. У жодному з перелічених стандартів не використовуються розділові знаки: «//», «-». Назва джерела та вихідні дані відокремлюються від авторів і заголовка статті типом шрифту, найчастіше, курсивом (italics), крапкою або комою. Існує багато безкоштовних програм для створення бібліографічних описів у романській абетці, що дають можливість автоматично створювати посилання за одним із світових стандартів наприклад: <http://www.easybib.com/>, <http://www.bibme.org/>, <http://www.sourceaid.com/>, <https://vak.in.ua/>.
- До статті обов'язково додається дві **рецензії**: зовнішня і внутрішня. Редакція залишає за собою право направляти статті на додаткову рецензію та відхиляти їх в разі відсутності рецензій.
- Стаття обов'язково подається на лазерному диску. Її також можна переслати електронною поштою за адресою: cit@lntu.edu.ua.
- В кінці статті обов'язково вказуються ПІБ, посада, науковий ступінь, вчене звання рецензентів статті.
- Рукописи, що не відповідають вище вказаним вимогам, не розглядаються і до друку не приймаються.

ЗРАЗОК ОФОРМЛЕННЯ СТАТТІ
(поля верхнє, нижнє -1.5 см, ліве та праве 2см. дзеркальні поля)

ЗРАЗОК

УДК 621.391

Мороз Борис Іванович, д.т.н., професор,

<https://orcid.org/0000-0002-5625-0864>

Антіпов Олександр Андрійович, аспірант,

Журавльов Володимир Сергійович, аспірант.

<https://orcid.org/0000-0002-7366-9552>

Національний технічний університет «Дніпровська політехніка», м. Дніпро, Україна.

АВТОМАТИЗОВАНА СИСТЕМА ДОСТАВКИ МЕДИКАМЕНТІВ ЗА ДОПОМОГОЮ БЕЗПЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ (МУЛЬТИКОПТЕРІВ) ЗА ЗАПИТОМ СПОЖИВАЧА

Мороз Б. І., Антіпов О.А., Журавльов В. С. Автоматизована система доставки мед **Times New Roman 9**
безпілотних літальних апаратів (мультикоптерів) за запитом споживача. Представлено концепт системи доставки
медикаментів за допомогою безпілотних літальних апаратів. Запропоновано архітектуру системи автоматичної диспетчеризації
замовлень від споживача, зберігання замовлень, та планування доставки дронами. Також було розглянуто юридичні обмеження
роботи запропонованої системи.

Ключові слова: мультикоптер, дрон, доставка, клієнт-серверна архітектура, RSA, APM, HTTPS, Mission Planner.

Мороз Б. И., Антипов А.А., Журавлев В. С. Автоматизированная система доставки медикаментов с помощью
беспилотных летательных аппаратов (мультикоптеров) по запросу потребителя. Представлен концепт системы доставки
медикаментов с помощью беспилотных летательных аппаратов. Предложена архитектура системы автоматической
диспетчеризации заказов от потребителя, хранения заказов, и планирование доставки дронами. Также были рассмотрены
юридические ограничения работы предложенной системы.

Ключевые слова: мультикоптер, дрон, доставка, клиент-серверная архитектура, RSA, APM, HTTPS, Mission Planner.

Moroz B., Antipov A., Zhuravlev V. Automated system for the delivery of medical supplies using unmanned aerial
vehicles (multicopter) at the request of the consumer. The concept of medical supplies delivery system using unmanned aerial vehicles
is presented. The architecture of the system of automatic dispatching orders from the consumer, storage of orders, and scheduling delivery
by drones are proposed. The legal limitations of the proposed system were also considered.

Keywords: multicopter, drone, delivery, client-server architecture, RSA, APM, HTTPS, Mission Planner.

Постановка наукової проблеми.

.....

Аналіз досліджень.

.....

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження.

.....

Висновки та перспективи подальшого дослідження.

.....

Times New Roman 11

Список бібліографічного опису

Times New Roman 9

1. Сін Лю, Ціньян Сяо, Віджай Гопалакришнан, Маттео Варвело (2017) Дослідження 360 ° Інновації для панорамного відеопотоку, С. 50-55. АСМ.
2. Б. Хань, Ф., Цянь, Л. Джі та В. Гопалакришнан. (2017) MP-DASH: Адаптивна відео-трансляція через перевагу, орієнтовану на багатифункціональність. У матеріалах 12-ї Міжнародної конференції з нових мережевих експериментів та технологій, С. 129–143. АСМ.

References

1. Xing Liu, Qingyang Xiao, Vijay Gopalakrishnan, Matteo Varvello (2017) Research 360° Innovations for Panoramic Video Streaming, P. 50-55. ACM.
2. Han, B., Qian, F., Ji, L. & Gopalakrishnan, V. (2017) MP-DASH: Adaptive Video Streaming Over Preference-Aware Multipath. In Proceedings of the 12th International on Conference on emerging Networking Experiments and Technologies, P. 129-143. ACM.

Довідки з питань публікації та прийому матеріалів у науковий журнал «Комп'ютерно-інтегровані технології: освіта, наука, виробництво» можна отримати у відповідального секретаря – Свиридюк Катерини Анатоліївни за тел. (0332) 74-61-15, або (063)-940-69-42.

Адреса: 43018, м. Луцьк, вул. Львівська, 75, ауд. 141

Автор статті отримує 1 примірник збірника. Вартість друку однієї сторінки становить **40 – грн.** (для працівників Луцького НТУ), **50 грн – для інших ЗВО.**

Окремо, кожній статті, буде присвоєний **DOI (digital object identifier)** - ідентифікатор цифрового об'єкту, що веде за собою додаткову оплату **45грн.**

Кошти можна перерахувати на рахунок:

Луцький національний технічний університет
43018 м.Луцьк, вул.Львівська, 75
р/р **UA86 820172 0 3132 4 1 002 2 02 017820**

З призначенням платежу: РВВ ЛНТУ. «За інформаційно-аналітичне забезпечення видання «Комп'ютерно-інтегровані технології: освіта, наука, виробництво» від _____ (ПБ)»