

**CONFERENCE PROCEEDINGS**  
**МАТЕРІАЛИ КОНФЕРЕНЦІЇ**

**XI, XII МІЖНАРОДНА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ  
З ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

**InfoCom**

**Advanced Solutions 2021**

**11,12 INTERNATIONAL SCIENTIFIC AND PRACTICAL CONFERENCE  
ON INFORMATION SYSTEMS AND TECHNOLOGIES**

**Україна, Київ**

**Ukraine, Kyiv**



**InfoCom**

**Advanced Solutions 2021**

**ISBN 978-966-2344-81-3**

Міністерство освіти і науки України  
Інститут модернізації змісту освіти  
КПІ ім. Ігоря Сікорського

# **INFOCOM ADVANCED SOLUTIONS 2021**

## **МАТЕРІАЛИ**

**XI та XII Міжнародних науково-практичних  
конференцій**

**з інформаційних систем та технологій**

## **CONFERENCE PROCEEDINGS**

**11<sup>th</sup> and 12<sup>th</sup> Scientific and practical conference**

**КИЇВ, УКРАЇНА**

**KYIV, UKRAINE**

**13-14 травня 2021 року / May 13-14, 2021**

**1-2 грудня 2021 року / December 1-2, 2021**

## УДК 004

### *Редакційна колегія:*

Теленик С.Ф., д.т.н., проф., КПІ ім. Ігоря Сікорського, Україна, Київ

Павлов О.А., д.т.н., проф., КПІ ім. Ігоря Сікорського, Україна, Київ

### *Головний редакт ор:*

Писаренко А.В., к.т.н., доц., КПІ ім. Ігоря Сікорського, Україна, Київ

### *Програмний коміт ет :*

Голова: проф. Олександр Ролік, Україна

### Члени:

проф. Mięczyślaw Zajac, Польща

д-р. Zbigniew Kokosiński, Польща

проф. Тетяна Ланге, Німеччина

проф. Сергій Теленик, Україна

проф. Ігор Грішин, Росія

проф. Володимир Самотий, Польща-Україна

проф. Олександр Павлов, Україна

проф. Анатолій Дорошенко, Україна

проф. Петро Бідюк, Україна

проф. Валерій Данилов, Україна

InfoCom 2021: Матеріали XI та XII Міжнародних науково-практичних конференцій з інформаційних систем та технологій, м. Київ, 13-14 травня 2021р., 1-2 грудня 2021р. – К.: Вид-во ТОВ "Інжиніринг", 2021. – 70с. – Мови укр., рос., англ.

Наказ ректора КПІ ім. Ігоря Сікорського № НМКП/42/2021 від 23 квітня 2021р.

Усі права застережено. Передруки та переклади дозволяються лише за згодою автора та редакції. За достовірність фактів, цитат, назв та іншої інформації несуть відповідальність автори. Редакційна колегія дотримується прийнятих міжнародною спільнотою принципів публікаційної етики, відображених, зокрема, в рекомендаціях Комітету з етики наукових публікацій (Committee on Publication Ethics, COPE), а також враховує досвід авторитетних міжнародних видавництв. Щоб уникнути недобросовісної практики в публікаційній діяльності (плагіат, виклад недостовірних відомостей та ін.), з метою забезпечення високої якості наукових публікацій, визнання громадськістю отриманих автором наукових результатів, кожен член редакційної колегії, автор, рецензент, видавець, а також установи, які беруть участь в видавничому процесі, зобов'язані дотримуватися етичних стандартів, норм і правил та вживати всіх можливих заходів для запобігання їх порушень. Дотримання правил етики наукових публікацій усіма учасниками цього процесу сприяє забезпеченню прав авторів на інтелектуальну власність, підвищенню якості видання і виключення можливості неправомірного використання авторських матеріалів в інтересах окремих осіб.

ISBN 978-966-2344-81-3

**ПРОГРАМА КОНФЕРЕНЦІЇ**

**CONFERENCE PROGRAM**



***Інформаційні системи та технології /  
Information systems and technologies***

Репнікова Н. Сверчков В.	Дослідження ефективності моделі двоногого крокуючого робота
Лаушкін В. Писаренко А.	Система керування магнітним мікророботом для доправлення лікарських засобів всередині організму людини
Адаменко Д. Писаренко А.	Система керування експериментальною установкою для дослідження імпульсних високовольтних розрядів
Тісна М. Букасов М.	Генерування іоніт-тестів з використанням машинного навчання
Бессмертний Р.	Використання високопродуктивних мікроконтролерів для автоматизованої платформи дослідження території

***Теорія керування / Control theory***

Репнікова Н. Гнип В.	Синтез багатовимірних систем керування з неквадратними матрицями
Висоцька Ю. Репнікова Н.	Дослідження методу цифрової апроксимації для вибору періоду квантування за часом

***Розподілені та паралельні обчислення /  
Distributed and Parallel Computing***

Козлова О.	Використання динамічного кворуму у процесі вибору лідера алгоритму Raft для прийняття консенсусу у розподілених системах
------------	--

***Технології програмування / Programming  
Technologies***

Гатіна Т.	Персоналізація страхових полісів: розробка шаблонів
-----------	---

***Інформаційні системи та технології /  
Information systems and technologies***

<b>Комарський О. Дорошенко А.</b>	Система генерації музики за допомогою рекурентних нейронних мереж
<b>Лопата В. Дорошенко А.</b>	Алгоритм генерації розкладів занять для закладів вищої освіти
<b>Шибецький В. Дорошенко А.</b>	Автоматична система безперервного тестування веб-додатків
<b>Мамонтов В. Березінський Г.</b>	Аналіз наукометричних даних науковців на основі графу співавторства
<b>Кривич О. Резніков С.</b>	Критерій оцінки вартості експлуатації мобільного транспортного робота
<b>Фомін І.</b>	Прогнозування фінансових рядів стану фондового ринку із застосуванням вейвлет-перетворення
<b>Тюрін В. Дорошенко А.</b>	Системи зберігання великих даних

***Штучний інтелект / Artificial intelligence***

**Нікулін О.** Система розподіленого навчання нейронних мереж

***Безпека та захист інформації / Security and protection of information***

**Ходарченко А.  
Полторак В.** Смарт-контракти NFT для оренди нерухомості

**МАТЕРІАЛИ КОНФЕРЕНЦІЇ**

**CONFERENCE PROCEEDINGS**





# ЗМІСТ / CONTENTS

<i>Інформаційні системи та технології / Information systems and technologies</i> .....	11
<b>Репнікова Н., Свєрчков В.</b> Дослідження ефективності моделі двоногого крокуючого робота.....	12
<b>Лаушкін В., Писаренко А.</b> Система керування магнітним мікророботом для доправлення лікарських засобів всередині організму людини.....	14
<b>Адаменко Д., Писаренко А.</b> Система керування експериментальною установкою для дослідження імпульсних високовольтних розрядів.....	16
<b>Тісна М., Букасов М.</b> Генерування юніт-тестів з використанням машинного навчання.....	18
<b>Бєссмертний Р.</b> Використання високопродуктивних мікроконтролерів для автоматизованої платформи дослідження території.....	20
<b>Комарський О., Дорошенко А.</b> Система генерації музики за допомогою рекурентних нейронних мереж.....	22
<b>Лопата В., Дорошенко А.</b> Алгоритм генерації розкладів занять для закладів вищої освіти.....	24
<b>Шибєцький В., Дорошенко А.</b> Автоматична система безперервного тестування веб-додатків.....	26
<b>Мамонтов В., Березінський Г.</b> Аналіз наукометричних даних науковців на основі графу співавторства.....	28
<b>Фомін І.</b> Прогнозування фінансових рядів стану фондового ринку із застосуванням вейвлет-перетворення.....	30
<b>Тюрін В., Дорошенко А.</b> Системи зберігання великих даних.....	32
<b>Кривич О., Резніков С.</b> Критерій оцінки вартості експлуатації мобільного транспортного робота.....	34
<i>Теорія керування / Control theory</i> .....	37
<b>Репнікова Н., Гнип В.</b> Синтез багатовимірних систем керування з неквадратними матрицями.....	38
<b>Висоцька Ю., Репнікова Н.</b> Дослідження методу цифрової апроксимації для вибору періоду квантування за часом.....	40
<i>Штучний інтелект / Artificial intelligence</i> .....	43
<b>Нікулін О.</b> Система розподіленого навчання нейронних мереж.....	44
<i>Безпека та захист інформації / Security and protection of information</i> .....	47

Ходарченко А., Полторак В. Смарт-контракти NFT для оренди нерухомості.....	48
<i>Розподілені та паралельні обчислення / Distributed and Parallel Computing.....</i>	<i>51</i>
Козлова О. Використання динамічного кворуму у процесі вибору лідера алгоритму Raft для прийняття консенсусу у розподілених системах.....	52
<i>Технології програмування / Programming Technologies.....</i>	<i>55</i>
Гатіна Т. Персоналізація страхових полісів: розробка шаблонів.....	56
<i>Abstracts.....</i>	<i>59</i>

**ІНФОРМАЦІЙНІ СИСТЕМИ  
ТА ТЕХНОЛОГІЇ**

**INFORMATION SYSTEMS  
AND TECHNOLOGIES**

# Дослідження ефективності моделі двоногого крокуючого робота

Репнікова Наталія, Свєрчков Владислав

*КПІ ім. Ігоря Сікорського*

м. Київ, Україна

n.repnikova@gmail.com, vladenigmatic@gmail.com

**Анотація**—В даній публікації за допомогою параметру Cost of Transport (CoT) досліджується ефективність моделі двоногого крокуючого робота Darth EvadeR, котрий вмiє обходити перешкоди. Проведені дослідження дозволили визначити оптимальне значення CoT та виконати порівняння з ефективністю вже існуючих крокуючих роботів. Розроблено алгоритм розрахунку загальної потужності, що споживає модель робота, з використанням значень оберտального моменту та кутової швидкості приводів робота.

**Ключові слова**—двоногий крокуючий робот, CoT, ефективність робота, обертальний момент, кутова швидкість, Darth EvadeR.

## Вступ

Під час розробки роботів, що здатні змінювати своє положення у просторі, виникає необхідність у визначенні їх ефективності. Для цієї задачі служить показник Cost of Transport, найчастіше вживається аббревіатура CoT. Це безрозмірний енергетичний критерій визначення ефективності роботів та інших рухомих транспортних засобів (включаючи живих істот), що досліджує енерговитрати на переміщення одиниці ваги на одиницю відстані [1]. Дане визначення більше підходить для роботи з живими істотами, а для робототехнічних систем краще користуватися формулою, яка використовується у діаграмах Кармана-Габрієлі [2]. У зазначеному джерелі CoT ще називається питомою силою тяги, питомим опором або ж індексом енергії і обчислюється за формулою (1):

$$CoT = \frac{P}{mgv}, \quad (1)$$

де  $P$  — потужність робототехнічної системи, Вт;  $g$  — гравітаційна стала ( $9,8 \text{ м/с}^2$ );  $v$  — швидкість пересування системи, м/с.

Потужність визначається як добуток сили струму та напруги на ділянці електричного кола — цією формулою зручно описувати середнє значення потужності на приводах вже існуючого робота, де є можливість здійснювати точні виміри сили струму. Якщо ж наявна лише модель робота, доцільно знаходити  $P$  через обертальний момент та кутову швидкість кожної частини кінцівки робота [3]. Середовище Simulink дозволяє легко відстежувати ці параметри. Сама формула має вигляд (2):

$$P = M \cdot \omega, \quad (2)$$

де  $M$  — обертальний момент, Нм;  $\omega$  — кутова швидкість обертання валу,  $\text{с}^{-1}$ .

Формули (1) та (2) дозволяють у знаходженні оптимального індексу CoT двоногого робота Darth EvadeR (детальніше про робота у [4]).

## АЛГОРИТМ РОЗРАХУНКУ ЗАГАЛЬНОЇ ПОТУЖНОСТІ РОБОТА DARTH EVADER

Особливістю авторського робота є його здатність обходити перешкоди з відновленням початкової прямолінійної ходьби. Обчислення середнього споживання потужності вимагає врахування двох фаз робота: прямолінійної ходьби та обходу перешкод. Тому був розроблений наступний алгоритм обрахування сумарної потужності робота:

- 1) Розбиття ходьби на 2 фази: 1 фаза — прямолінійна ходьба, 2 — обхід перешкоди.
- 2) Знаходження середнього значення потужності для кожного приводу робота за один цикл ходьби, що триває 0,8 с. Ці значення сумуються, а результат — сумарне споживання роботом потужності за один цикл ходьби у фазі 1. Далі знаходиться сумарна потужність, витрачена роботом на здійснення ходьби у фазі 1 при відомому часі виконання завдання.
- 3) Дії з п. 2 застосовуються і для 2 фази робота. Результуюче значення — сумарна потужність, витрачена роботом на здійснення ходьби у фазі 2.

Сума знайдених потужностей з п.2, 3 і є сумарною потужністю робота при виконанні завдання. Важливо розуміти, що завдання робота передбачає реалізацію наступних дій: робот має пройти 1,5м, зупинитися перед перешкодою на відстані 1м, обійти 0,4м перешкоди, відновити прямолінійний рух і зайти за перешкоду. Демонстрація дій, які має виконати робот, на рис.1.

## ОБЧИСЛЕННЯ CoT

Darth EvadeR має наступні параметри: вага — 172 кг, середня швидкість — 0,139 м/с, споживана потужність — 13,999 кВт, час виконання завдання — 22,11 с. Підставляємо дані до (1) та отримуємо 58,4. Проводимо серію дослідів, змінюючи тривалість циклу ходьби робота, кути

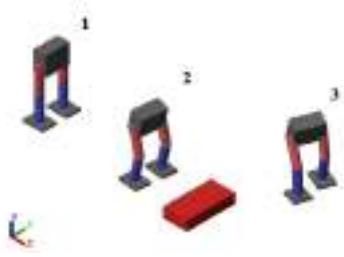


Рис. 1. Переміщення робота Darth EvadeR: 1 — початкова позиція робота, 2 — робот виявив перешкоду і зупинився, 3 — робот обійшов перешкоду і зайшов за неї

згинання-розгинання кінцівок та шукаємо оптимальне значення CoT (найменше). Результати дослідів у табл.1.

Табл. 1  
РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ CoT

№	P, кВт	v, м/с	CoT	t, с	Що було змінено
1	14,01	0,093	89,37	21,9	Збільшено кути відхилення стегон на 4°(фаза 1)
2	20,74	0,1171	105	24,6	Збільш. кути відхилення стегон, колін та стоп на 5-8°(фаза 1)
3	13,34	0,1299	60,9	27,4	Збільш. цикл ходьби з 0,8с до 0,95с (фаза 2)
4	14,26	0,1291	65,5	24,6	Збільш. цикл ходьби до 0,86с (фаза 2)
5	18,81	0,093	120,4	31,7	Збільш. цикл ходьби до 0,83с (фаза 1) та до 0,86с у фазі 2
6	19,08	0,1879	60,25	15,2	Зменш. цикл ходьби до 0,52с (фаза 1)
7	18,02	0,2535	42,16	14	Зменш. цикл ходьби до 0,47с (фаза 1), збільш. до 0,81с у фазі 2
8	19,1	0,2046	55,4	17,1	Зменш. цикл ходьби до 0,47с (фаза 1) та до 0,77с у фазі 2
9	11,09	0,143	45,9	17,6	Збільш. цикл ходьби до 0,83с (фаза 1) та до 0,81с у фазі 2
10	14,96	0,106	83,3	29,7	Збільш. цикл ходьби до 0,86с (фаза 1) та до 0,81с у фазі 2

Як можна бачити з табл. 1, оптимальне значення CoT у досліді №7 — 42,16. Був отриманий приріст швидкості у розмірі 82,25%, час виконання роботом завдання скоротився на 36,64%, однак енергоспоживання зросло на 31,5%.

#### Порівняння CoT з існуючими роботами

Аби зрозуміти адекватність значення CoT, котре було назване оптимальним, слід порівняти його зі значеннями CoT вже наявних крокуючих роботів. Розглянемо наступні робототехнічні системи (наведені у табл. 2): 1 — двонога варіація робота RHex [5], 2 — двоногий Cornell [6], 3 — шестиногий Gregor I, 4 — шестиногий DASH, 5 — двоногий Asimo, 6 — чотириногий Big Dog [7], 7 — Darth EvadeR. Їх CoT наведені у табл. 2.

Як можна бачити, CoT авторського робота більше ніж в 2 рази перебільшує найближчого

Табл. 2

CoT різних роботів, а також моделі Darth EvadeR

Робот	1	2	3	4	5	6	7
Маса, кг	9	13	1	0,0062	50	110	172
CoT	1	0,2	70	147	3,2	15	42,16

вагового конкурента — чотириноного Big Dog. Двоногий робот Asimo з вагою 50 кг тим часом має в 13 разів менший CoT.

Насправді, ефективність Darth EvadeR 39,76 не є «поганим» показником, оскільки при великій вазі робота високі затрати потужності є передбачуваними при тому, що він досить швидко зближується з перешкодою та обходить її.

#### Висновки

У даній публікації був обчислений показник ефективності моделі двоногого крокуючого робота Darth EvadeR. За допомогою серії дослідів було визначено оптимальний індекс CoT, а також було проведено порівняння отриманого індексу з показниками вже існуючих роботів. Оптимальний показник CoT у порівнянні з найпершим розрахованим дав вииграш у швидкості пересування (+82,25%) та у часі виконання роботом завдання з наближення до перешкоди та її обходу (-36,64%), однак споживання потужності при цьому збільшилося на 31,5%.

Також, зважаючи на існування робота Darth EvadeR лише у якості моделі у середовищі MATLAB/Simulink, для знаходження потужності використовувалася формула, що спирається не на класичні силу струму та напругу, а на обертальний момент та кутову швидкість приводів моделі робота. Обчислений CoT показав, що модель Darth EvadeR'a потребує подальшого удосконалення, а саме: зменшення ваги робота і, відповідно, зменшення енергоспоживання; вдосконалення алгоритму пересування та обходження перешкод.

#### ЛІТЕРАТУРА

- [1] Yang T. Control of aperiodic walking and the energetic effects of parallel joint compliance of planar bipedal robots: – Columbus, Ohio, 2007. – 214 с.
- [2] Halloran M. Wing in Ground Effect Craft Review (DSTO-GD-0201) [Електронний ресурс] / DSTO Aeronautical and Maritime Research Laboratory. – 1999. – Режим доступу: <https://apps.dtic.mil/sti/pdfs/ADA361836.pdf>.
- [3] [https://eti.su/articles/elektricheskie-mashini/elektricheskie-mashini\\_1571.html](https://eti.su/articles/elektricheskie-mashini/elektricheskie-mashini_1571.html) (дата звернення 15.04.2021).
- [4] Репнікова Н. Б. Вдосконалення функціональних можливостей моделі крокуючого робота / Н. Б. Репнікова, В. А. Сверчков // Проблеми інформатизації та управління. – 2021. – Т. 1, №65 – С. 62-67.
- [5] Neville N. A bipedal running robot with one actuator per leg // IEEE International Conference on Robotics and Automation: – Orlando, 2006 – С. 848-853.
- [6] Collins S. Efficient Bipedal Robots Based on Passive-Dynamic Walkers // Science. – 2005. – Т. 307, Вип. 5712. – С. 1082-1085.
- [7] Luneskas M. Hexapod Robot Gait Switching for Energy Consumption and Cost of Transport Management Using Heuristic Algorithms / Appl. Sci. – 2021. – Т. 11, Вип.3.

# Система керування магнітним мікророботом для доправлення лікарських засобів всередині організму людини

Лаушкін Володимир, Писаренко Андрій  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна

**Анотація**—Метою даної роботи є описання принципу роботи та структури системи керування магнітним мікророботом. Основною задачею такої системи є доправлення лікарських засобів у тілі людини без інвазивного втручання. Принцип дії заснований на маніпулюванні за допомогою зовнішнього магнітного поля невеликими за розміром магнітними мікрочастинками, які пов'язані між собою та утворюють корпус робота. Така система забезпечує простіші та безпечніші методи лікування деяких захворювань.

**Ключові слова**—мікроробот, магнітна взаємодія, доправлення ліків.

## Вступ

Метою створення системи керування магнітним мікророботом є зменшення кількості випадків необхідності інвазивного втручання до тіла людини. Вирішеною прикладною задачею є процес транспортування, а саме доставка лікарських засобів у необхідне місце в організмі людини. При деяких онкологічних захворюваннях застосовується методика радіотерапії [1], яка являє собою вплив радіоактивними речовинами на злоякісні клітини. При застосуванні розроблюваного пристрою для даного виду терапії можливо опромінювати тканину дуже локально невеликими дозами випромінювання у важкодоступних місцях та без хірургічного втручання.

## Концептуальний опис системи

Зовнішньо основна частина приладу має форму тунелю та нагадує форму апарату МРТ. Людина розміщується всередині апарату, після чого у вену вводиться катетер, який забезпечує інтерфейс для введення та виведення робота у тіло людини. На наступному етапі робот отримує частку лікарського препарату, який поступає через мікрошприц та приймається роботом. Після цього під впливом зовнішнього магнітного поля робот починає рух та потрапляє у тіло людини через катетер. У тілі людини виконується переміщення робота та захопленої ним мікрочастинки ліків по кровоносній системі завдяки магнітному полю. Досягнувши необхідної ділянки тіла робот зупиняється та деякий час впливає на тканину лікарським препаратом. Після виконання опромінення робот повертається у вихідне положення.

Отже, можна зауважити що, система складається з таких основних елементів: магнітний мікроробот; установка для генерації зовнішнього магнітного поля; засоби відстеження положення; блок управління системою.

Найважливішим елементом є магнітний мікроробот (зображений на рис. 1), який представляє собою магнітну тканину сконфігуровану таким чином, щоб робот мав змогу приймати необхідні форми та виконувати певні дії.



Рис. 1. Зовнішній вигляд магнітного мікроробота

Дана тканина виготовляється з використанням спеціального гідрогелю [2] та дуже малих магнітних часточок (діполей). На початку виробництва гелю знаходиться в рідкому стані, у нього розміщуються магнітні часточки та, за допомогою впливу зовнішнього магнітного поля та лазера, усі частки всередині гелю приймають фіксоване необхідне положення. Таким чином утворюється тканина певної форми та певного розміщення магнітних мікрочастинок всередині, яка утворює корпус робота.

З рис. 1 видно, що робот має чотири основні частини: основу та три руки-захоплювачі. Така конфігурація надає роботу можливість ефективно переміщуватись та переміщувати вантаж.

Для приведення робота у рух необхідно застосувати зовнішнє магнітне поле. Воно виникає у електромагнітах, які встановлені у тунелі установки.

Робот має два типи рухів: рухи руками-захоплювачами та обертання [3]. Рухи руками-захоплювачами застосовуються для фіксування лікувального препарату. Для даної операції застосовується градієнт магнітного поля. При змінненні градієнту робот або підіймає руки-захоплювачі – відкривається або опускає – за-

кривається. Для переміщення робота застосовується інший тип руху – обертання. У закритому стані корпус робота стає схожим на кулю і, коли робот знаходиться у рідинному середовищі, він обертається відштовхуючись від рідини. У кожному з випадків обертання робота виконується завдяки зовнішньому обертовому магнітному полю.

Для отримання зворотного зв'язку у системі використовуються два типи сканування: комп'ютерна томографія та ультразвуковий метод (сонограф). Застосування декількох засобів відстеження положення робота обумовлюється обмеженнями системи які накладає взаємодія з тілом людини [4-5]. Комп'ютерна томографія (КТ) має велику роздільну здатність тому її необхідно застосовувати для отримання точної моделі необхідної ділянки тіла. Ця модель буде використовуватись як мапа для навігації робота.

Сонограф застосовується для миттєвого відстеження робота у тілі. Завдяки великій частоті оновлення та абсолютній безпеці для людини, ультразвуковий метод добре підходить для цієї задачі. Усі дані, що надходять з обох джерел оброблюватимуться за допомогою системи розпізнавання образів.

#### ОПИС СТРУКТУРИ СИСТЕМИ

Оскільки сам мікроробот є майже елементарною частиною системи з точки зору кількості елементів та їх конструкції, у даному розділі буде описана структура блоків які забезпечують переміщення та відстеження робота. Загальну структуру представлено на рис.2. З рис. 2 ви-



Рис. 2. Зовнішній вигляд магнітного мікроробота

дно, що в загальному вигляді структура пристрою складається з таких модулів: блок давачів забезпечення переміщення, блок давачів розпізнавання положення, блок обробки інформації

зворотного зв'язку, обчислювальний пристрій, блок виконавчих пристроїв забезпечення переміщення, блок виконавчих пристроїв забезпечення розпізнавання положення та блок пристроїв вводу/виводу.

Блок давачів забезпечення переміщення містить: блок давачів положення магнітних напрямних, давач поздовжнього переміщення, давач сили магнітного потоку, давач вектору магнітного потоку, давач магнітної проникності, давач ультразвукової проникності. Блок давачів розпізнавання положення містить у собі чутливі елементи, які необхідні для отримання інформації про моментальне положення робота та складається з газових та твердотілих детекторів. Ці детектори є частинами апарату комп'ютерної томографії. Блок виконавчих пристроїв забезпечення переміщення складається з наступних елементів: блок двигунів магнітних напрямних, двигун поздовжнього переміщення, котушки генерації магнітного поля, трансформатор, блок керування частотою змінного струму. Блок виконавчих пристроїв забезпечення розпізнавання положення містить: електронно-променеву трубку (елемент КТ), генератор (елемент КТ), випромінювач-детектор (елемент сонографа). Блок пристроїв вводу/виводу містить запам'ятовуючий пристрій. Основним елементом системи є обчислювальний пристрій який збирає інформацію та генерує керуючі сигнали для виконавчих пристроїв. При обробці зображень для розпізнавання образів необхідний швидкий та надійний пристрій, який матиме змогу отримувати, корегувати та надавати інформацію щодо положення магнітів та сили їх магнітного поля, аби керувати мікророботом в реальному часі.

#### ВИСНОВКИ

В роботі розглянута концепція побудови та функціонування системи керування магнітним мікророботом для доправлення лікарських засобів всередині організму людини. Розглянуті основні складові системи, їх особливості та призначення. Серед перспектив подальшого розвитку теми є побудова математичної моделі системи керування рухом мікроробота за визначеної траєкторією.

#### ЛІТЕРАТУРА

- [1] Applications of micromechatronics in minimally invasive surgery / F.Tendick, S. Sastry, R. Fearing, M. Cohn. // IEEE/ASME Trans. Mechatron. – 1998. – С. 34–42.
- [2] Microrobots for Minimally Invasive Medicine / B. Nelson, I. Kaliakatsos, J. Abbott. // Annual Review of Biomedical Engineering. – 2010. – С. 79.
- [3] Millimeter-scale flexible robots with programmable three-dimensional magnetization and motions / T. Xu, J. Zhang, M. Salehizadeh, O. Onaizah, E. Diller. // Robotic Science. – 2019. – С. 5-7.
- [4] A Tumbling Magnetic Microrobot System for Biomedical Applications / E. Niedert, C. Bi, G. Adam, E. Lambert, L. Solorio, C. Goergen, D. Cappelleri. // Micromachines. – 2020. – С. 6.
- [5] Control of Magnetically-Driven Screws in a Viscoelastic Medium / Z. Zhang, A. Klingner, S. Misra, S. Khalil. // Micromachines. – 2020. – С. 17.





## ОПИС СТРУКТУРИ СИСТЕМИ

Блок давачів температури. Датчики розташовані на екранах, що нагріваються від випромінювання. За допомогою цих датчиків можливий збір даних про зміну режиму роботи лазерів та зміну параметрів діода. Блок давачів кутів



Рис. 3. Структурна схема

нахилу коліматорів лазерів. Ці датчики потрібні для точного розуміння у яку точку напрямлене випромінювання з кожного коліматора. Блок давачів зміщення коліматорів лазерів призначені для відслідковування точного положення фіксатора коліматорів, який, за допомогою електродвигуна матиме можливість переміщуватись вздовж осі діода. Дачч положення анода. Даний датчик відповідає за надання інформації про те, на якому віддалені від кінця катода знаходиться кінець анода. Блок лазерів. У розроблюваній системі для створення плазми використовуються лазери. По колу надходять промені які створюють плазму з поверхні матеріалу, на яку вони спрямовані. Утворена плазма стимулює пробій діода. Двигун переміщення аноду виконує функцію збільшення або зменшення відстані між катодом та анодом. Перед центральним обчислювальним пристроєм стоїть задача керувати механізмами та роботою системи таким чином, аби накопичити статистику, яка покаже ту конфігурацію

налаштувань, що є найбільш енергоефективною. Головним критерієм ефективності є температура, яку давачі температури зможуть зафіксувати на екранах.

## АЛГОРИТМ ЗБОРУ ТА ОБРОБКИ ДАНИХ

Навколо діода встановлено металеву кулю, що поглинає випромінювання та теплову енергію. По «паралелях» кулі встановлені термомпари, та для кожної паралелі вираховується середнє арифметичне температури. Таким чином з усіх рівнів давачів отримано  $t_1, t_2, \dots, t_n$  відповідно. Зібрана енергія дорівнює

$$E = m \cdot c \cdot (t_{a.s.} - t_{a.s.0}),$$

де  $E$  – теплова енергія;  $c$  – теплоємність матеріалу кулі;  $m$  – маса кулі.

Параметр  $l$  – проміжок між катодом та анодом діода, – параметр, що автоматично або в ручному режимі змінюється при кожному експерименті для визначення оптимальної величини по критерію максимізації зібраної енергії. Параметр – затримка або випередження спрацьовування лазера відносно спрацьовування розрядників. автоматично або в ручному режимі змінюється при кожному експерименті для того, аби визначити його оптимальну величину по критерію максимізації зібраної енергії. Спочатку встановлюється статичним та проводяться експерименти по пошуку максимального енерговиділення ( $\cdot$ ) зі зміною  $l$ , знаходиться оптимальне  $l$ . Отримане значення  $l$  фіксується та запускається алгоритм пошуку оптимального  $\cdot$ . Як результат – отримується оптимальна комбінація  $T, l$ .

Результатом роботи даного алгоритму є таблиця, що автоматично формується та містить у собі такі дані :  $E, l, T, I, U$ , де  $I$  – максимум сили струму;  $U$  – максимум напруги. У таблиці виконується сортування по значенню  $E$ . Таким чином можна буде визначити при яких параметрах вдалось отримати найбільше теплової енергії.

## ВИСНОВКИ

У роботі було описано ідею створення системи керування роботою лазерного обладнання, що має за мету покращити роботу експериментальної установки. Розраховується, що розроблювана система зробить можливим автоматичний збір даних, крім того, за допомогою алгоритму, буде виконуватись певне керування роботою лазерів на основі зібраних результатів експериментів, що зробить можливим пришвидшити пошук оптимальних налаштувань для кожної з конфігурацій, що застосовуються.

## ЛІТЕРАТУРА

- [1] Controlled Nucleosynthesis / Stanislav Adamenko, Franco Selleri, Alwyn van der Merwe. // Springer Science & Business Media. – 2007. – С. 16–40.
- [2] r-Process Nucleosynthesis in Hot Accretion Disk Flows from Black Hole-Neutron Star Mergers / R. Surman, G. C. McLaughlin, M. Ruffert, H.-Th. Janka, W. R. Hix. // The Astrophysical Journal 679. – 2008. – С. 5-30.

# Генерування юніт-тестів з використанням машинного навчання

Тісна Марія, Букасов Максим  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна

**Анотація**—Метою цієї роботи є дослідження існуючих підходів автоматизованого генерування юніт-тестів та виявлення найбільш перспективного з них до застосування. Проаналізовано існуючі рішення та виявлено їх основні недоліки. Розроблено архітектуру системи генерування юніт-тестів з використанням машинного навчання, представлено алгоритм пов'язання між тестами та кодом, що тестується.

**Ключові слова**—юніт-тест, машинне навчання, нейронні мережі, генератор, тренування моделі.

## Вступ

Фундаментальним рівнем тестування є юніт-тестування, яке дозволяє перевіряти на коректність роботу кожного індивідуального модуля коду. Відсутність таких юніт-тестів призводить до високовартісних дефектів, які зазвичай виправляють уже на наступних стадіях тестування. Актуальним рішенням проблеми відсутності юніт-тестів або неякісно написаних юніт-тестів є генератор юніт-тестів.

Генератори тестів можуть використовуватися у програмних проектах, де відсутній початковий набір тестів, тести можуть бути згенеровані, розробники надалі будуть їх підтримувати та покращувати. Ефективність набору тестів визначається саме як їх можливість знаходження помилок у програмі. Найсучасніші генератори тестів, хоча і є однією з активно досліджуваних тем, наразі можуть знайти лише невелику кількість потенційних недоліків у програмному забезпеченні і є менш ефективними за тестове покриття, яке може бути досягнуте мануально створеними тестами. Низька ефективність наборів тестів, створених генераторами юніт-тестів, є проблемою, яку необхідно вирішити для розроблення та тестування програмного забезпечення. Техніки машинного навчання продемонстрували високий потенціал, вони застосовуються у таких задачах розроблення програмного забезпечення, як автодоповнення коду, баг-фіксинг, генерування коментарів, псевдокоду та інше. Через це проблема використання алгоритмів глибинного навчання для генерування юніт-тестів є актуальною.

## МЕТОДИ АВТОМАТИЗОВАНОГО ГЕНЕРУВАННЯ ЮНІТ-ТЕСТІВ

Для досягнення високої ефективності набору тестів існує багато різних підходів. Деякими з них є:

- Рандомізоване тестування. Базовий та прямолінійний підхід, який складається з виклику функцій з випадковими вхідними даними [1].
- Тестування, основане на пошуку. Алгоритми пошуку, або генетичні алгоритми, перетворюють задачі генерування юніт-тестів у проблеми оптимізації, де ціль генерування тестів реалізована через функцію допасованості, яка керує пошуком [2].
- Символічне тестування. Символічний підхід представляє шляхи виконання через програму як обмеження, встановлене на вхідних значеннях. Найбільш відомим є динамічне символічне виконання (Dynamic Symbolic Execution) [3].

Існує багато алгоритмів машинного навчання з нейронними мережами, які можуть перекладати послідовності. Для даного дослідження були обрані sequence-to-sequence (seq2seq) нейронні мережі [4], основані на рекурентних нейронних мережах [5] або згорткових нейронних мережах [6], тому що вони можуть швидко оброблювати довгі послідовності з високою точністю.

## АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

Найпопулярнішими існуючими рішеннями для генерування юніт-тестів для мови програмування Java є EvoSuite, Randoop та Symbolic PathFinder [7]. EvoSuite є відкритим програмним забезпеченням, створеним за методом тестування пошуком, та використовує еволюційний підхід, оснований на генетичному алгоритмі, для генерування юніт-тестів з метою задоволення критерію покриття. Randoop використовує рандомізоване тестування, направлене на зворотній зв'язок. Symbolic PathFinder є прикладом інструменту на основі символічного тестування, який комбінує символічне виконання з перевіркою моделі та рішенням обмежень. Головними недоліками вищеперерахованих підходів є складність читання та розуміння згенерованих тестів, які виглядають як машинний код. Також дані інструменти мають такі недоліки, як незадовільні якість коду та вміння виявлення помилок. Дані рішення переважно фокусуються на покритті коду, ігноруючи інші фактори, які можуть бути важливими для розробників.

## СТРУКТУРНА СХЕМА СИСТЕМИ ГЕНЕРУВАННЯ ЮНІТ-ТЕСТІВ

Модуль збору даних знаходить відкрите програмне забезпечення за певними фільтрами, наприклад, усі проекти мають використовувати одну і ту ж мову програмування та один і той же фреймворк юніт-тестування. Далі до зібраного та проаналізованого програмного коду застосовується алгоритм пов'язання юніт-тестів до вихідного коду, завдяки якому створюються пари з методів юніт-тестів та методів вихідного коду, які тестуються. Ці тест-пари поділяються на тренувальний сет, який використовується для тренування моделі машинного навчання; валідаційний сет, який дозволяє визначити, коли необхідно припинити тренування; та верифікаційний сет для оцінки та порівняння моделі. Потім починається етап тренування. Процес тренування завершується валідацією для впевнення, що подальше тренування не вплине негативно на результат, зменшуючи можливість узагальнення. Врешті-решт, модель проходить процес верифікації, використовуючи верифікаційний сет, який складається з даних, до яких модель не є оптимізованою. В результаті отримуємо модель, готову до передбачення тестів. В подальшому модель можна використовувати для генерування юніт-тестів, передаючи їй у якості вхідних даних вихідний код проекту без юніт-тестів. Описаний процес зображений на рис. 1.

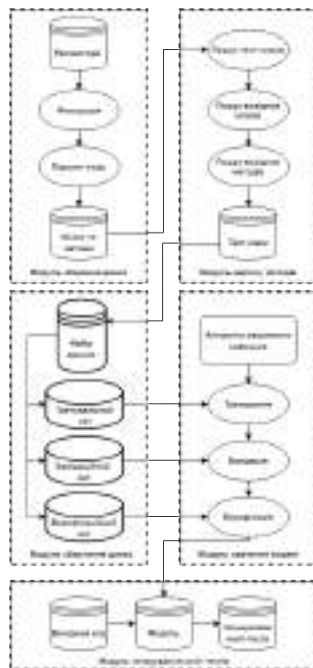


Рис. 1. Структурна схема системи генерування юніт-тестів

### АЛГОРИТМ ПОВ'ЯЗАННЯ КОДУ ДО ТЕСТІВ

Для тренування алгоритму машинного навчання необхідні тренувальні приклади. Щоб створити їх, необхідний набір даних з пар, які складаються з методів вихідного коду та юніт-тестів.

Однак, не існує прямого зв'язку між тестом та кодом, який тестується. Таким чином, щоб мати змогу створити пари, необхідний алгоритм пов'язання юніт-тестів з методами.

Спочатку необхідно знайти у проектах класи, які містять у собі тест-кейси. Це можна зробити за допомогою анотації @Test для мови програмування Java та фреймворку тестування JUnit. Якщо клас містить хоча б один метод з анотацією @Test, його можна назвати тестовим класом.

Далі для кожного тестового класу необхідно визначити відповідний клас вихідного коду, який тестується. Для виконання цього пошуку, можна застосувати наступні методи:

- Пошук за шляхом. Взавши за основу стандартну структуру організації пакетів в Java-проектах, почнемо пошук вихідного коду з кореневого каталогу (src/main/java), шлях якого має відповідати тестовим класам у кореневому каталозі (src/test/java).
- Пошук за назвою. За загальноприйнятою згодою щодо оформлення коду, назва тестового класу зазвичай складається з назви класу, що тестується, з префіксом або суфіксом "Test".

Наступним кроком необхідно знайти безпосередньо метод, який тестується. Це можна зробити за допомогою пошуку за назвою. Якщо ж цей спосіб виявився неефективним, для знаходження методу, який тестується, необхідно визначити, який метод викликається у тест-кейсі, та чи співпадає він з певним методом з класу, що тестується.

### ВИСНОВКИ

У роботі проведено огляд існуючих методів автоматизованого генерування юніт-тестів та означено найбільш перспективний, а саме використання нейронних мереж. Також проведено аналіз існуючих рішень та виявлено їх основні недоліки. У результаті було спроектовано структурну схему системи автоматизованого генерування юніт-тестів з використанням машинного навчання, розроблено алгоритм її функціонування. Також запропоновано алгоритм пов'язання юніт-тестів до вихідного коду.

### ЛІТЕРАТУРА

- [1] Arcuri A. Random testing: Theoretical results and practical implications / A. Arcuri, M. Iqbal, L. Briand., 2012. – 277 с.
- [2] McMinn P. Search-Based Software Test Data Generation: A Survey / Phil McMinn., 2004. – 156 с.
- [3] [https://en.wikipedia.org/wiki/Symbolic\\_execution](https://en.wikipedia.org/wiki/Symbolic_execution) (дата звернення 04.11)
- [4] <https://www.analyticsvidhya.com/blog/2020/08/a-simple-introduction-to-sequence-to-sequence-models> (дата звернення 05.11.2021)
- [5] <https://www.ibm.com/cloud/learn/recurrent-neural-networks> (дата звернення 05.11.2021).
- [6] [https://en.wikipedia.org/wiki/Convolutional\\_neural\\_network](https://en.wikipedia.org/wiki/Convolutional_neural_network) (дата звернення 06.11.2021).
- [7] [https://knapsackpro.com/testing\\_frameworks/difference\\_between/evosuite/vs/randoop-net](https://knapsackpro.com/testing_frameworks/difference_between/evosuite/vs/randoop-net) (дата звернення 09.11.2021).

# Використання високопродуктивних мікроконтролерів для автоматизованої платформи дослідження території

Бессмертний Роман  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна  
roman.bezsmertniy@gmail.com

**Анотація**—В даній роботі надані результати розробки програмної інфраструктури автоматизованої платформи дослідження території що використовують можливості архітектури Cortex-A. Була проведена робота з метою розробки мініатюрної недорогої системи збору і обробки відео інформації. Саме ця інформація є основою базового елемента системи визначення місцезнаходження, що дотепер здійснюється за допомогою супутникової навігації, що є вразливою до втрати зв'язку. Результат розробки прототипу автоматизованої системи управління наземною рухомою платформою на основі фреймворку Spring з вбудованою системою визначення місцезнаходження на основі відеоданих дозволив визначити стандарти такої системи і забезпечити високу надійність таких систем.

**Ключові слова**—АСУ, рухома платформа, Spring, одометрія, обробка відеоданих.

## Вступ

При виконанні пошукових робіт на небезпечних для людини ділянках використовуються дистанційно керовані мобільні платформи. В даний час найбільш поширене застосування наземних колісних і гусеничних мобільних платформ[1]. Основна перевага їх застосування це можливість розміщення фахівців, в даному випадку оператора мобільної платформи, поза межами потенційно небезпечної зони, зберегти здоров'я і життя людей.

При втраті зв'язку, погіршенні видимості і інших випадках, коли відсутня можливість візуального спостереження, виникає проблема ручного управління платформою оператором. Такі ситуації потребують використання системи автоматичного повернення рухомої платформи до місця перебування оператора. Саме вони засновані на використанні автономної навігації.

## Аналіз останніх досліджень і публікацій

Аналіз останніх публікацій [1-4], що пов'язані з вищеописаним, показав, що на теперішній час активно іде розвиток контролерів на базі архітектури Arm(cortex-A). Автоматизація управління реалізується у тому числі і для систем управління мобільними платформами, діаграма розгортання яких показана на рис.1. Основою контролера автоматизованої системи управління мобільною платформою (АСУМП) є OrangePi

Zero Plus. Далі розкриті технічні характеристики мікрокомп'ютера OrangePi Zero Plus, що виконує роль контролера системи управління рухомою платформою:

- процесор 64-бітний SoC Allwinner H5 Quad-core Cortex-A53;
- графічний процесор Mali450MP4;
- оперативна пам'ять 512Mb DDR3 synchronous dynamic random-access memory (включаючи GPU);
- вбудована пам'ять 8GB embedded Multi-MediaCard;
- можливість підтримки карт пам'яті microSD (об'ємом до 64GB);
- аудіо- і відеовивід через HDMI з підтримкою Consumer Electronics Control;
- вбудований WiFi 802.11 b/g/n + Bluetooth 4.0 Low Energy;
- microUSB 2.0 On-The-Go x1;
- camera Serial Interface;
- налагоджувальний порт;
- 26 пінів General Purpose Input-Output, сумісних з Raspberry Pi B +;
- світлодіоди стану і активності;
- живлення від microUSB роз'єму;
- розміри: 48 x 46 mm;
- вага: 20g.

Базовими причинами вибору такого рішення є невисока вартість, \$15 на момент написання статті, наявність безкоштовної ОС "Armbian", сумісність даної ОС з технологіями JAVA і наявність безкоштовних, надійних бібліотек для виконання поставленої задачі [5-8].

## АКТУАЛЬНІСТЬ ТА ПОСТАНОВКА ЗАДАЧІ

Метою даної роботи є зменшення трудомісткості (вартості) розробки програмної складової системи управління мобільною платформою за рахунок використання загальнодоступного фреймворку, а також описати її роботу на рівні діаграми розгортання.

## Виклад основного матеріалу

Метою статті є опис системи автоматизованого керування мобільними платформами на рівні

діаграми розгортання, у стандартах UML. Отже задачею є показати технологічні можливості практичної реалізації АСУМП на базі мікроконтролерів Arm(cortex-A) на прикладі OrangePi.

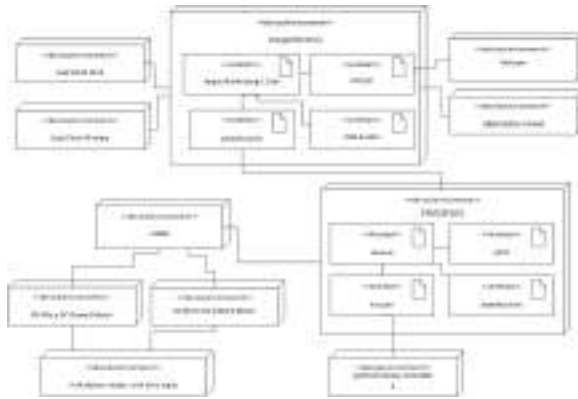


Рис. 1. Діаграма розгортання АСУМП

На рис. 1 показана діаграма розгортання частини АСУМП на базі мікрокомп'ютера cortex-A, а саме OrangePi Zero Plus. В нижній частині діаграми розгортання зображені виконавчі механізми, що керуються за допомогою мікроконтролера STM32F103, підключеного до OrangePi Zero Plus через інтерфейс General Purpose Input-Output (GPIO). До виконавчих механізмів відносяться два електричних двигуни, а також оптичні датчики швидкості обертання. Оскільки двигуни потребують різкі зміни напруги, їх живлення відбувається через окремий коло живлення. Мікроконтролер дозволяє керувати двигунами в реальному часі.

У верхній частині рисунку показана плата мікрокомп'ютера Orange Pi, і операційна система Armbian, встановлена на ній. До мікрокомп'ютера за допомогою USB під'єднується живлення та веб-камера. WI-FI, встановлений у плату мікрокомп'ютера, дозволяє підключитися до встановленого-веб-сервера для керування наземним роботом.

Таким чином, можна виділити два основні контури управління – контур управління двигунами, а також контур управління відеокамерою та обробки візуальної інформації.

Система на основі діаграми розгортання може бути використана як прототип системи з можливістю візуального спостереження.

## Висновки

У ході виконання роботи було побудовано прототип автоматизованої системи управління рухомою платформою, що показав можливість зменшення трудомісткості і вартості розробки програмної складової системи управління за рахунок використання загальнодоступного фреймворку. Також було описано його роботу на рівні діаграми розгортання у стандартах UML.

Побудований прототип може бути використаний як основа для системи автоматичного повернення рухомої платформи до місця перебування оператора.

## ЛІТЕРАТУРА

- [1] Szewczyk, R. Progress in Automation, Robotics and Measuring Techniques. : Volume 2 Robotics / R. Szewczyk, C. Zieliński, M. Kaliczynska // Springer, 2015. P. 81 – 92
- [2] The Best Single-Board Computer for Robotics [електронний ресурс]: <https://www.electromaker.io/blog/article/best-single-board-computer-for-robotics>
- [3] Використання високопродуктивних мікроконтролерів для підвищення економічної ефективності виробництва джему / Р. С. Бессмертний, П. Ю. Катін // Стандартизація. Сертифікація. Якість. - 2019. - № 3. - С. 69-77. - Режим доступу: [http://nbuv.gov.ua/UJRN/ssia\\_2019\\_3\\_11](http://nbuv.gov.ua/UJRN/ssia_2019_3_11)
- [4] Wiring Pi GPIO Interface library for the Raspberry Pi [електронний ресурс]: <http://wiringpi.com>
- [5] Java Communications API [електронний ресурс]: <https://www.oracle.com/technetwork/java/index-jsp-141752.html>
- [6] Spring Framework Overview [електронний ресурс]: <https://docs.spring.io/spring/docs/current/spring-framework-reference/overview.html>
- [7] A java package to access the Capture interface of the Video4Linux API [електронний ресурс]: <https://code.google.com/archive/p/v4l4j/>
- [8] rxtx - a Java cross platform wrapper library for the serial port [електронний ресурс]: <https://github.com/rxtx/rxtx>

*Рецензент: доцент кафедри інформаційних систем та технологій КПІ ім. Ігоря Сікорського к.т.н. доцент Катін Павло*

# Система генерації музики за допомогою рекурентних нейронних мереж

Комарський Олександр, Дорошенко Анатолій  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна  
komarskiy33@gmail.com, doroshenkoanatoliy2@gmail.com

**Анотація**—В роботі розглядається можливість генерації музичних композицій використовуючи рекурентні нейронні мережі. Запропоновано та розглянуто два підходи генерації музичних творів, а саме за допомогою методу нот та методу акордів. Проведено дослідження обох методів і сформульовані їх переваги та недоліки. Детально описано процес пошуку та обробки даних для навчання музичної нейронної мережі. Запропоноване рішення дозволяє легко генерувати музичні твори без втручання людини.

**Ключові слова**—генерація музики, машинне навчання, рекурентні нейронні мережі, LSTM, RNN.

## ВСТУП

Машинне навчання покращило багато аспектів нашого повсякденного життя, як очевидних, так й непомітних. Штучний інтелект відіграє ключову роль у таких речах, як системи рекомендацій в медіа-сервісах, розумні будинки, комп'ютерний зір, системи виявлення підозрілих дій, тощо. І навіть незважаючи на вже реалізовані застосування, досі існує величезний потенціал для покращення використання машинного навчання в різних сферах людської діяльності, включаючи медицину, віртуальні помічники, електронну комерцію та сферу фінансів.

Водночас не менш важливим і цікавим напрямом, в якому машинне навчання може зіграти свою роль є творчість. Комп'ютерна творчість – це дійсно захоплююча сфера, в якій штучний інтелект перетинається з мистецтвом, а метою стоїть розробка систем, які можуть моделювати, вдосконалювати або розуміти творчість людини.

Генерація музики є найбільш абстрактним напрямком творчої діяльності, оскільки на відміну від написання книжок, картин або віршів, музика не прив'язана до контексту оточуючого нас світу.

## ГЕНЕРАТИВНЕ МИСТЕЦТВО

Автоматична генерація музики – це процес створення музичного твору з мінімальним втручанням людини.

Алгоритмічний підхід до створення музики існує вже кілька століть, починаючи з Гвідо д'Ареццо, який в 1024 винайшов перший алгоритм для складання музики. Саме через величезні можливості, які може запропонувати обчислювальна техніка, алгоритмічні музичні композиції почали розквітати з початку 1950-х років та до сьогодні.

## ВИБІР ТЕХНІЧНИХ ЗАСОБІВ

Для побудови нейронної мережі необхідно було обрати базову модель машинного навчання. Проаналізувавши всі наявні моделі рекурентних нейронних мереж, як базову було обрано модель AWD-LSTM [1], яка найчастіше використовується для побудови нейронних мереж в сфері мовних досліджень. Для реалізації програмної частини було обрано мову програмування Python, оскільки має велику кількість бібліотек та модулів для роботи в цьому напрямку. Основним фреймворком для реалізації архітектури мережі було обрано PyTorch [2]. Також був використаний модуль CUDA [3], для реалізації навчання мережі використовуючи графічний процесор.

## МЕТОДИ НОТ ТА АКОРДІВ

Як одне з можливих рішень для генерації музики – це кожного разу запитувати модель, чи грати їй зараз дану ноту для кожної з 88 клавіш фортепіано на кожному музичному кроці. Однак, оскільки кожна окрема нота мовчить більшість часу, то для нейронної мережі було б дуже важко навчитись передбачати ствердну відповідь «так» для нот. Тому було розглянуто два можливі способи для рішення цієї проблеми. Мовні нейронні мережі часто працюють на рівні символів, або на рівні слів. Так само для генерації музики, було досліджено два аналогічних рівня – нот та акордів. Акордом вважається комбінація нот, які коли-небудь зустрічались в музичній композиції. Тобто ставитись до акорду, можна так само як до слів мови. Це означає, що можливо запитати нейронну мережу, яким буде наступний акорд (слово), надавши їй попередню послідовність акордів (речення). Для методу, який використовує ноти, вважаємо, що початок та кінець кожної ноти є різними словами, наприклад «p28» та «endp28» в текстовому форматі MIDI є різними словами, та будуть використовуватись для початку та закінчення ноти №28.

В обох методах було використано трохи зменшений діапазон нот – 62 ноти замість повних 88 на фортепіано. Ноти, які виходили з цього діапазону, було переміщено вниз або вгору на октаву. В результаті експериментів були визначені характеристики кожного з двох методів.

Характеристика методу акордів:

- Нейронна мережа добре генерує довгі музичні патерни, але не може вийти за рамки навчальних даних, та згенерувати щось нове.
- Зазвичай для прийняття рішення, щодо наступного акорду, найефективніше вибрати найбільш ймовірний прогноз на кожному часовому відрізку.
- Зустрічається ефект, коли час від часу музична композиція перестрибує на іншу неочікувану ноту, але потім одразу продовжує грати мелодію в звичайному ритмі.

Характеристика методу нот:

- Нейронна мережа створює гармонійні ритмічні малюнки, які можуть легко та гармонійно переходити в один з одного.
- Легко обробляє параметри тривалості натискання ноти, за рахунок чого мелодія здається більше інтуїтивно зрозумілою для людини.
- Для прийняття рішення, щодо наступної ноти найефективніше чергувати між собою три найбільш ймовірні прогнози на кожному часовому відрізку.

Отже в результаті цих порівнянь, було прийнято рішення про використання методу нот, як основного для генерації музики в системі.

#### НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ

Навчання моделі нейронної мережі потребує багато обчислювальних ресурсів, та велику кількість якісних навчальних даних [2].

Для прискорення навчання були використані хмарні обчислення. Обчислення робились за до-

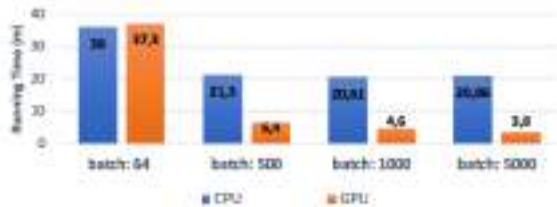


Рис. 1. Порівняння швидкості навчання нейронної мережі з використанням CPU та GPU

помогою ресурсів графічного процесора, оскільки такий підхід забезпечує більшу швидкість опрацювання даних, ніж центральний процесор. Це видно з результатів порівняння зображених на рис. 1. Для порівняння були використані центральний процесор i7 8550 U на 4 ядра, та графічний процесор Geforce MX150 з 2 гігабайтами оперативної пам'яті. Як видно з графіків, зі зростанням кількості даних – зростала швидкість їх обробки, особливо використовуючи графічний процесор, який при максимальній кількості даних був швидше ніж центральний процесор в 5,5 разів.

Вибір даних для тренування є важливою складовою будь-якої нейронної мережі, оскільки вона

отримує всі свої «знання» про музику лише з навчальних даних. Також необхідно мати дані для тестування нейронної мережі, які повинні відрізнятися від даних для тренування, щоб визначити, чи змогла модель абстрагуватись від даних тренування та скласти нові мелодії, а не просто відтворити дані, якими її навчали.

До тренувальних даних є дві вимоги. Перша – музичні композиції повинні бути в одному жанрі та написані людьми, а не іншою нейронною мережею. Друга вимога – мелодії повинні приємно звучати. Оскільки зазвичай різні жанри музики для людей звучать по-різному та залежать від їх смаку, було вирішено обрати для навчання класичну музику як найбільш нейтральний жанр.

Для збору навчального датасету було написано декілька скриптів, за допомогою яких були зібрані файли з найбільшого сайту агрегатора класичної музики – «Classical Piano Midi Page», де представлені композиції таких композиторів як Моцарт, Бах, Бетховен та багато інших.

Для навчання моделі необхідно було перетворити MIDI-файли у формат, який зможе зрозуміти нейронна мережа. Файли MIDI [3] надають інформацію про час початку гри та зупинки на кожному ноту, а також інформацію про музичний інструмент, значення гучності та темпу гри. На виході нейронної мережі музичні композиції подаються спочатку у вигляді власного текстового формату, а далі кодуються в MIDI та MP3.

#### ВИСНОВКИ

В роботі була розглянута та реалізована можливість використання рекурентних нейронних мереж для задачі генеративного мистецтва. Були розглянуті два підходи, які використовують метод нот та метод акордів. В результаті досліджень як основний був обраний метод нот, оскільки результатом його роботи були більш цікаві та гармонійні музичні композиції, на відміну від результатів які вдалось отримати методом акордів. Також було проведено порівняння швидкості навчання нейронної мережі з використанням центрального та графічного процесорів, за результатами якого графічний процесор оброблював дані швидше ніж центральний у 5,5 разів.

#### ЛІТЕРАТУРА

- [1] <https://paperswithcode.com/method/awd-lstm> (дата звернення 10.11.2021).
- [2] <https://pytorch.org/docs/stable/index.html> (дата звернення 10.11.2021).
- [3] <https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html#introduction> (дата звернення 10.11.2021).
- [4] Ian Goodfellow Deep Learning (Adaptive Computation and Machine Learning series) / Ian Goodfellow, Yoshua Bengio, Aaron Courville – 2016. – p 276.
- [5] <http://www.music.mcgill.ca/~ich/classes/mumt306/StandardMIDIfileformat.html> (дата звернення 11.11.2021).



# Алгоритм генерації розкладів занять для закладів вищої освіти

Лопата Владислав, Дорошенко Анатолій

КПІ ім. Ігоря Сікорського

м. Київ, Україна

lopatavlad@gmail.com, doroshenkoanatoliy2@gmail.com

**Анотація**—У даній статі запропонований алгоритм генерації розкладів для ЗВО. Матеріал в статі враховує необхідність регенерації розкладів на основі оцінки з вагових коефіцієнтів та пропонує алгоритм табуйованого локального пошуку альтернативних розкладів для побудови нових розкладів на основі початкового з метою покращення оцінки якості розкладу. Вводиться поняття якості розкладу і метод оцінки якості розкладу виходячи з явних вагових коефіцієнтів.

**Ключові слова**—алгоритм генерації розкладів, табуйований локальний пошук, NP-повна задача, індивідуальний розклад студента, оптимізація згенерованого розкладу, якість згенерованого розкладу, свобода розташування заняття у розкладі.

## ВСТУП

Функціонування закладів вищої освіти сильно залежить від ритму навчання, закладеного в організаційних документах закладів. Одним з основних документів, що регулює ритм навчання та вносить ясність в принципи функціонування розкладу для студентів і викладачів.

Утім, задача складання розкладів не є простою, оскільки для кожного закладу існує ряд обмежень, які необхідно врахувати при побудові оптимального розкладу. Таких обмежень може бути дуже багато, і це одна з причин, чому задачі складання розкладів присвоюють NP-повний рівень складності.

Варіанти вирішення задачі складання розкладу включають як повний перебір варіантів, що базуються на обмеженнях, так і різні оптимізації на основі генетичних алгоритмів та інших видів алгоритмів, які допомагають знизити кількість варіантів, які необхідно перевірити для побудови оптимального розкладу.

Дана стаття пропонує лінійний алгоритм генерації першого розкладу, який буде задовольняти мінімальному рівню обмежень, а також алгоритм регенерації розкладу, що буде знаходити більш оптимальні з точки зору обмежень рішення шляхом використання алгоритму табуйованого локального пошуку.

## АЛГОРИТМ ГЕНЕРАЦІЇ РОЗКЛАДІВ ЗАНЯТЬ

Оцінюючи свободи розташування  $i$ -го заняття у розкладі проводиться підрахунок:

- кількості аудиторій  $a_i$ , які підходять для проведення заняття, на підставі вимог заняття до обладнання аудиторії та кількості робочих місць;
- кількості занять на тиждень  $p_i$ , які проводить викладач цього заняття;
- кількості занять на тиждень  $g_i$  для заданого студента (не рекомендується розглядати групу студентів, оскільки наразі існує практика дисциплін на вибір).

На підставі цих даних визначається оцінка свободи розташування заняття у розкладі:

$$S_i = \frac{a_i}{g_i \cdot p_i},$$

де  $S_i$  – оцінка свободи розташування  $i$ -го заняття у розкладі;

$n$  – кількість занять.

Після оцінювання свободи розташування занять у розкладі проводиться сортування списку занять щодо зростання їх оцінок свободи розташування:

$$S_i < S_{i+1}, i = 1..n.$$

## АЛГОРИТМ РЕГЕНЕРАЦІЇ РОЗКЛАДІВ

Нехай є  $n$  критеріїв для оцінки розкладу, кожен з яких може бути числом в діапазоні  $[0;1]$ . Тоді на їх основі можна оцінити якість розкладу, застосувавши наступні розрахунки:

$$R_{il} = \sum_{j=1}^m w_j k_{jl},$$

де  $R_{il}$  – якість розташування  $i$ -го заняття на  $l$ -й позиції у розкладі;

$k_{jl}$  – значення, отримане за  $j$ -м критерієм оцінки якості розташування заняття на  $l$ -й позиції у розкладі;

$w_j$  – ваговий коефіцієнт  $j$ -го критерію оцінки якості;

$m$  – кількість критеріїв оцінки якості.

Після оцінки якості всіх можливих варіантів розташування заняття у розкладі вибирається варіант, при якому досягається максимальне значення оцінки

$$R_i = \max_l (R_{il}), l = 1..h,$$

де  $l$  – можлива позиція  $i$ -го заняття в розкладі;

$R_i$  – якість розташування  $i$ -го заняття у розкладі;

$h$  – кількість можливих варіантів розташування заняття у розкладі.

Після розміщення всіх занять у розкладі проводиться оцінка якості складеного розкладу. Для оцінки якості розкладу вирішено використати суму оцінок якості розташування всіх занять у розкладі. З огляду на те, що оцінка якості розташування кожного заняття за критеріями появи та зникнення вікна у розкладах студентів та викладачів залежить від взаємного розташування занять у розкладі – необхідно провести повторну оцінку якості розташування кожного заняття у складеному розкладі. Для оцінки якості складеного розкладу використовується формула:

$$R = \sum_{i=1}^n R_i. \quad (1)$$

На основі отриманих оцінок можна оптимізувати розклад, застосувавши алгоритм табуованого локального пошуку.

Шляхом застосування незначних змін (як-от перестановка двох сусідніх занять з мінімальною різницею ступеня свободи  $S_i - S_j$ ) можна отримати набір розкладів, що є «сусідніми рішеннями» для даної задачі. Для кожного з цих розкладів визначається  $R$  за (1) і найкращий запам'ятовується. З метою запобігання зациклювання алгоритму на локальних мінімумах, коли усі сусідні рішення схожі і є гіршими за поточне оптимальне, застосовується табування – рішення, що були відхилені як неоптимальні запам'ятовуються і не застосовуються в подальших ітераціях.

Алгоритм табуованого локального пошуку ітеративний, тож знайшовши рішення з кращим  $R$ , до нього можна застосувати наступну ітерацію пошуку і шукати більш оптимальне рішення серед сусідів поточного.

### Висновки

Отже, для подальшої оптимізації розкладу, що генерується лінійним алгоритмом можна застосувати табуований локальний пошук і отримати оптимізоване за обмеженнями рішення, яке буде більш ефективним з точки зору обмежень.

Такий спосіб регенерування розкладів можна застосувати після додавання нових обмежень до системи, щоб вихідний розклад змінився не надто сильно (що гарантується локальним пошуком, оскільки зміни, які пропонуються для сусідніх рішень є мінімальними перестановками занять, а не повним пересортуванням розкладу).

Алгоритмічне рішення, що було описано у тексті статті може бути застосовано в системах генерації розкладів, оскільки вирішує основні проблеми, що виникають при розробці цього типу систем, а саме: генерування розкладу і регенерування розкладу з мінімальними змінами за

виникнення додаткових умов, за якими розклад необхідно оптимізувати.

Дане алгоритмічне рішення буде застосовано як один із способів генерування і регенерування розкладів у системі генерації розкладів занять для ЗВО, де для нього буде додано графічний інтерфейс з можливістю генерувати та регенерувати розклад, а також опублікувати отриманий розклад і надати студентам можливість отриманий розклад переглянути.

Також, дане алгоритмічне рішення можна застосувати в рамках будь-яких закладів освіти, у тому числі середньої у ручному і автоматичному форматі для побудови розкладів, що будуть використовуватися у якості розкладів занять у відповідному закладі.

### ЛІТЕРАТУРА

- [1] Берегових Ю. В. Алгоритм составления расписания занятий / Ю. В. Берегових, Б. А. Васильев, М. О. Володин. // Искусственный интеллект. – 2009. – №2. – С. 50–56.
- [2] Галузин К. С. Математическая модель оптимального учебного расписания с учетом нечетких предпочтений : автореф. дис. на здобуття наук. ступеня канд. фіз.-мат. наук / Галузин Константин Станиславович – Пермь, 2004. – 18 с.
- [3] Fred G. Future Paths for Integer Programming and Links to Artificial Intelligence / Glover Fred. // Computers and Operations Research. – 1986. – №5. – С. 533–549.
- [4] Serial and parallel search techniques for the traveling salesman problem / M. Malek, M. Huruswamy, H. Owens, M. Pandya. // Annals of OR: Linkages with Artificial Intelligence. – 1989. – С. 75–101.

# Автоматична система безперервного тестування веб-додатків

Шибєцький Владислав, Дорошенко Анатолій

КПІ ім. Ігоря Сікорського

м. Київ, Україна

vladyslav.shyb@gmail.com, doroshenkoanatoliy2@gmail.com

**Анотація**—В даній публікації розглядається автоматизація тестування різних рівнів програмного продукту на прикладі веб-додатків. Була виявлена основна проблема наявних рішень та на їх основі запропоновано новий підхід та методику тестування. Запропоновано новий підхід та методику тестування, метою яких є систематизація усього процесу тестування. Створена та випробувана тестова модель, що яка довела свою ефективність нового підходу та методики.

**Ключові слова**—автоматична система безперервного тестування, CI, Jenkins, автоматизація, підхід, методика.

## ВСТУП

Основною проблемою наявних підходів та методик тестування є їх вузько направленість. Завдяки цьому тестування складних багаторівневих систем ділиться на частини та проводиться окремо. Таке рішення видається раціональним, бо надає можливість швидко перевіряти необхідну область об'єкта та значно пришвидшує перший цикл тестування частинами, під кожен з яких можна швидко знайти спеціалістів та залучити їх на короткий термін для базового покриття тестами необхідної області. Але такий підхід не є оптимальним, оскільки він слабо орієнтований на цілісний результат що призводить до складності аналізу загальної якості продукту. Реалізація наявних методів та підходів дозволяє знизити більшість ризиків, проте приводять до вибору між малоефективним використанням часу чи коштів.

## АНАЛІЗ ПРОБЛЕМИ

З класичної моделі тестування (рис.1) нас цікавлять основні 3 рівні тестування – системне, інтеграційне і модульне. Для зручності будемо розглядати їх на прикладі веб-додатків.

Системне тестування покривається Unit-тестами. Воно проводиться у самому коді програмного забезпечення та аналізується, як правило, розробниками за допомогою консольного журналу (логу). Цей вид тестування спрямований на базову перевірку функціоналу. Якщо об'єкт не проходить Unit-тести, проводиться відновлення програмного забезпечення до останньої стабільної версії та аналізуються проблеми. Цей вид

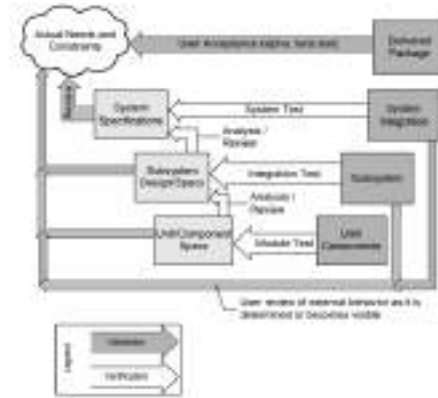


Рис. 1. Класична модель тестування [1]

тестування здебільшого вже автоматизований і покращувати його великого сенсу не має.

Інтеграційне тестування веб-застосунків проводиться через API (прикладний програмний інтерфейс) – набір визначень підпрограм, протоколів взаємодії та засобів для створення програмного забезпечення. Простими словами це можна назвати «вікном спілкування» між серверною (бекендом) та клієнтською (фронтом) частинами веб-додатка. У більшості випадків тестується людиною повністю або розбивається на компоненти. Тестування проводиться за допомогою спеціальних інструментів, таких як Postman, SoapUI та т.п. і аналізується за допомогою логу та згенерованих звітів про тести, які аналізуються бізнес-аналітиками та менеджерами проекту.

Системному тестуванню відповідає UI тестування (тестування інтерфейсу користувача) що може проводитися як вручну так і за допомогою автотестів, написаних будь-якою мовою програмування, що підтримує Selenium WebDriver. Selenium WebDriver – це драйвер, що дозволяє коду взаємодіяти з клієнтською частиною додатку. У результаті автоматизації прикладного інтерфейсу (API) генерується лог та документ, який відправляється на аналіз. Створити повністю автоматичну систему неможливо, оскільки будь-який продукт постійно оновлюється і потрібна людина, яка буде корегувати тести на нижчому рівні. Результатом ручного тестування

є документ з тест-кейсами, які буди перевірені у ході тестування, та звіти, у яких описується, що, як і коли тестувалося.

Також слід зазначити, що зараз для автоматизації тестування використовуються системи контролю версій та неперервного інтегрування. Прикладом цього є використання Git для синхронізації моделей тестування чи використання системи безперервної інтеграції Jenkins [2] для автозапуску тестування. Але ключовою проблемою є їх точкове використання і відсутність єдиного підходу.

Проаналізувавши, як проводиться тестування програмного забезпечення, було виведено спільну схему та створене її схематичне зображення (див. рис.2).

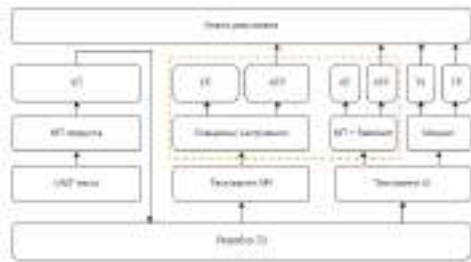


Рис. 2. Загальна схема тестування веб-додатків

Після аналізу проблеми стає очевидно що Unit тести та ручне тестування інтерфейсів UI є задовільними оскільки Unit тести вже ефективно працюють на рівні розробки а мануальне UI тестування на виході видає добре оформлений результат. Залишається оптимізувати блок тестування API та автоматизоване UI тестування (на рис.1 виділені пунктиром), основною вадою яких є нестандартизований результат, який призводить до збільшення витрат часу на його обробку. Для кожного з напрямків є багато чудових інструментів, але це призводить до непов'язаної між собою документації, яка не може дати чіткого розуміння статусу продукту.

#### ЗАПРОПОНОВАНИЙ ПІДХІД

Результатом аналізу проблеми є необхідність створення системи, яка змогла б об'єднати тестування API та автоматизоване тестування UI і стандартизувати їх вихідний результат. На основі цих вимог та аналізу наявних технологій була створена наступна теоретична модель системи (рис. 3).

Сутність запропонованої методики полягає у тому що об'єкт тестування ділиться на частини для перевірки кожної з них, але у кінці результат подається як одне ціле.

Каркасом запропонованої моделі є Jenkins, що це автономним сервером автоматизації з відкритим кодом. З його допомогою можна брати актуальні версії кожного модуля з Git та запускати їх відповідно до налаштованих параметрів.

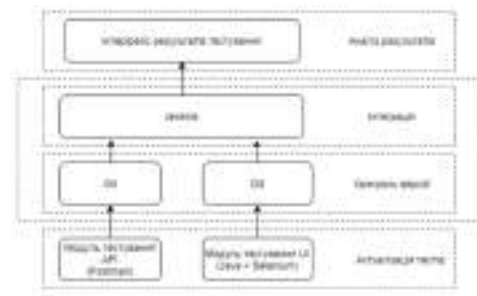


Рис. 3. Теоретична модель системи

На роль основного інструменту тестування API найкраще підходить Postman. Postman дозволяє створювати та використовувати запити, документувати й контролювати сервіси в одному місці.

Для коректної взаємодії Postman та Jenkins необхідно додатково встановити Newman. Для тестування UI підходить будь-яка мова програмування що підтримує Selenium WebDriver.

Також необхідно реалізувати налаштувати розсилку звітів про тестування, приведених до зручного формату на електронну пошту або створити окрему веб-сторінку на яку у формі діаграм виводити результати останнього тестування кожного модуля. Це надає замовнику постійний доступ до результатів тестування, які не потрібно переоформлювати.

#### ЗАСТОСУВАННЯ

Запропонована система тестування не може повністю замінити ручне тестування, проте чудово підходить для проведення регресів [3]. Регресійне тестування – це тип тестування, який проводиться, щоб перевірити, чи зміна коду в програмному забезпеченні не впливає на наявні функціональні можливості продукту. Це робиться для того, щоб переконатися, що продукт працює нормально з новим функціоналом, виправленнями помилок або будь-якими змінами в існуючій функції.

Також плюсом є те, що замовник і кожен член команди у будь-який час може побачити статус тестування.

#### ВИСНОВКИ

У ході дослідження було проведено аналіз недоліків загальної моделі тестування та запропоновані новий підхід і методика тестування. Була детально описана модель цієї методики та на її основі був реалізований прототип запропонованого рішення. Він не зміг покрити все тестування об'єкта, проте значно оптимізував процес та чудово зарекомендував себе у регресах.

#### ЛІТЕРАТУРА

- [1] Mauro Pezze and Michal Young, Software Testing. Techniques, Principles, and Practices / John Wiley&Sons – 2008. – p. 16.
- [2] <https://www.jenkins.io/doc/> (дата звернення 14.11.2021).
- [3] <https://uk.myservname.com/what-is-regression-testing> (дата звернення 14.11.2021).

# Аналіз наукометричних даних науковців на основі графу співавторства

Мамонтов Владислав, Березінський Генадій

*КПІ ім. Ігоря Сікорського*

м. Київ, Україна

vladyslavmamontov@gmail.com, gennady.berezinsky@gmail.com

**Анотація**—Стаття присвячена дослідженню методів аналізу даних наукової діяльності груп науковців, використовуючи графову структуру даних. Ставиться задача візуалізації отриманих даних з метою спрощення прийняття управлінських рішень адміністрацією закладів науки та освіти. Робота проведена на основні інформації про науковців, що діють у рамках закладу вищої освіти КПІ ім. Ігоря Сікорського. Для аналізу використано методи типізації, обчислено показники центральності та вирішена задача про кліку графу.

**Ключові слова**—наукометрія, Google Scholar, графічне представлення, візуалізація, показник центральності, ступінь посередництва, індекс Хірша.

## Вступ

Найбільш широко оцінка наукової діяльності наукових дослідників застосовується по відношенню до результативності праці науково-педагогічних співробітників університетів, наукових інститутів і приватних компаній. Аналіз оцінок, та в цілому наукової діяльності, дозволяє визначити найбільш перспективні галузі, що розвиваються, і, що більш важливо, визначити коло конкретних науковців, які вносять найбільший вклад у розвиток наукової області, в якій працюють. Результати аналізу дозволяють звернути увагу та направити фінансування на найбільш перспективні, з точки зору збільшення наукометричних показників, групи науковців.

## АКТУАЛЬНІСТЬ РОБОТИ

Кожен з індексаторів має свої критерії відбору наукових публікацій, а також свої розрахунки наукометричних показників. Правила відбору журналів в індексах Web of Science і Scopus найжорсткіші. Навпаки, система Google Scholar збирає всі наукові публікації в Інтернеті, і є безкоштовною. Лише після їх загального аналізу можна дати правильну оцінку діяльності науковця. Наразі відсутні запропоновані рішення представлення великого масиву наукометричних даних науковців та їх комплексного аналізу з метою допомогти в прийнятті управлінських рішень.

## МЕТА ДОСЛІДЖЕННЯ

Метою роботи є розробка підходів для прийняття управлінських рішень адміністрацією закладів науки та освіти. Для цього використовую-

ться методи аналізу наукометричних показників групи науковців, що дозволяють визначити найбільш перспективні підгрупи, виявляти вплив на загальну картину наукової діяльності. Для представлення даних у емпіричній формі використано технології візуалізації. Результати роботи показані на прикладі вчених, що ведуть наукову та викладацьку діяльність у межах КПІ ім. Ігоря Сікорського.

## АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

Дослідження є продовженням робіт [1, 2], що полягають у зборі та попередньому аналізі наукометричних даних науковців, які працюють на базі зазначеного вище закладу вищої освіти (ЗВО). Із недоліків, які виправлені у даній роботі, можна назвати наступні:

- візуалізація даних не спрямована на відображення узагальнених результатів;
- аналіз даних має вузький спектр;
- аналіз проводиться у рамках заздалегідь визначених підрозділів. У даній роботі підрозділи для аналізу формуються, у тому числі, на основі отриманих показників (наприклад, визначення кліки графу).

## ОПИС ПРЕДСТАВЛЕННЯ ДАНИХ

Інструментом подальшого аналізу та візуалізації було обрано графову структуру даних. Побудова графу відбувається на основі відношень науковців за кількістю спільних публікацій. Це дозволило легко візуалізувати дані. На графі відображені вузли (науковці), ребра, що їх з'єднують за наявністю спільних наукових робіт, та вага ребер (кількість спільних наукових робіт). Петлі на графі дозволяють відобразити загальну кількість робіт автора.

## ДАНІ ДЛЯ АНАЛІЗУ

Дані для аналізу отримано з наукометричних баз Google Scholar та Scopus за допомогою синтаксичного аналізатора. Система проведення аналізу може використовувати дані з обох ресурсів. Проте, у цій статті розглядаються наукометричні дані, взяті з бази Google Scholar, оскільки ця база є відкритою, та містить інформацію про більшу кількість науковців та їх робіт. Обрана

частина бази даних містить записи про 2694 науковців та 63456 статей з ресурсу Google Scholar. Усього маємо 6664 пари науковців з різною кількістю наукових робіт, що є показником спільної наукової активності. Для побудови інформативної графової структури необхідно провести очищення даних. Було видалено науковців, що не працювали у співавторстві з іншими авторами у межах даних, що аналізуються. Вузли графа, що залишилися, було відфільтровано за значеннями величини компонентів [3], що у даному випадку залежать від кількості публікацій науковця та спільних наукових робіт. Даний алгоритм реалізується бібліотекою NetworkX [4]. У результаті фільтрації кількість досліджуваних науковців скоротилася до 1519. Було виконано типізацію даних для можливості отримання візуальних характеристик впливовості того чи іншого типу. Колір вузла використовується для позначення науковців в рамках одного інституту або факультету. За параметр розміру вузлів відповідає ступінь посередництва вузла графу.

**АНАЛІЗ ДАНИХ НА ОСНОВІ ВІЗУАЛІЗАЦІЇ ГРАФУ**  
Важливим показником при аналізі графу є ступінь посередництва вузла. Він визначається як сума частки всіх пар найкоротших шляхів, які проходять через обраний вузол [5]. Показник центральності вершини визначає найважливіші вершини графу – вони є найбільш впливові у соціальних мережах [6]. Зі збільшенням показника центральності зростає значимість вузла (автора) у загальній тенденції написання наукових робіт та співробітництві науковців. На рис.1 наведено

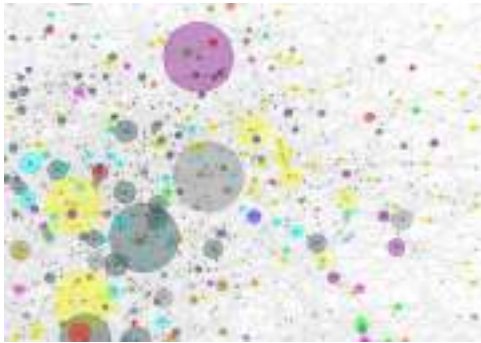


Рис. 1. Приклад фрагменту побудованого графу

фрагмент побудованого графу на основі даних про всіх науковців КПІ ім. Ігоря Сікорського. Наочно представлено різницю між вузлами з різним показником центральності, області скупчення вузлів, кількісну характеристику взаємодії науковців. Використовуючи побудований граф за заданими підрозділами, можна отримати інформацію про:

- кооперацію між членами різних підрозділів;
- порівнювати ступінь взаємодії науковців у межах одного підрозділу;
- визначати підрозділи з найбільш впливовими науковцями.

## КЛІКА ГРАФУ

Задачу про знаходження найбільш ефективних груп науковців можна звести до задачі пошуку клік графу. Розмір кліки визначається як число вершин в ній [7]. Кліки графа у контексті задачі, що розглядається, є група вчених, кожен з яких проводив наукову діяльність з кожним іншим представником цієї групи. Ставиться задача знаходження найбільшої кліки, у відповідності до класичної задачі теорії графів [8]. Продемонструємо на побудованому по всьому ЗВО графі. Максимальний розмір знайденої кліки рівний 6. Її складають науковці ІФФ, кафедри фізики металів, для яких спільне значення h-індексу дорівнює 11.

## ВИСНОВКИ

Робота показує можливості прикладного аналізу з метою прийняття управлінських рішень, на основі методів візуалізації даних показників наукової діяльності. Запропоновано спосіб застосування інформації про діяльність науковців для дослідження ефективності і визначення найбільш перспективних напрямків. Наведено методи для побудови графу наукової діяльності; знаходження даних, що можуть відігравати ключову роль при вирішенні питань фінансування, та необхідності визначення впливу груп на загальну картину наукометричних показників.

## ЛІТЕРАТУРА

- [1] Березінський Г. В. Аналіз наукометричних даних та складання рейтингу підрозділів КПІ ім. Ігоря Сікорського на основі публіцистичної діяльності викладачів /В.В. Мамонтов, Г.В. Березінський // Матеріали конференції «Інформаційні системи та технології управління» (ІСТУ-2020) – м. Київ.: 24-30 квітня 2020. – С. 110-117.
- [2] Мамонтов В. В. Інформаційна система визначення критеріїв технології підтримки науково-публіцистичної діяльності науковців закладу вищої освіти: / Мамонтов В.В., Березінський Г.В.– Київ, 2020. – С. 121.
- [3] Networkx Algorithms Connected components [Електронний ресурс] // Networkx. – 2020. – Режим доступу: [https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.components.connected\\_components.html](https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.components.connected_components.html).
- [4] Software for Complex Networks [Електронний ресурс] // NetworkX. – 2020. – Режим доступу: <https://networkx.org/documentation/stable/index.html>.
- [5] Ulrik Brandes. A faster algorithm for betweenness centrality // Journal of Mathematical Sociology. – 2001. – Т. 25, вип. 2. – С. 163-177.
- [6] Newman M.E.J. Networks: An Introduction. // Oxford, UK: Oxford University Press – 2010.
- [7] Cook, Stephen A. The Complexity of Theorem-Proving Procedures. Proceedings of the Third Annual ACM Symposium on Theory of Computing. // Shaker Heights, Ohio. – 1971 – С. 151-158.
- [8] Blair, Jean R. S.; Peyton, Barry "An introduction to chordal graphs and clique trees Graph theory and sparse matrix computation. // IMA Vol. Math. Appl., 56, Springer, New York – 1993. С. 1-2.

*Рецензент: доцент кафедри інформатики та програмної інженерії КПІ ім. Ігоря Сікорського  
к.т.н. доцент Фіногенов Олексій*

# Прогнозування фінансових рядів стану фондового ринку із застосуванням вейвлет-перетворення

Фомін Ілля  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна

**Анотація**—Метою дослідження є розробка архітектури нейронної моделі для визначення трейдингової стратегії та відстеження руху тенденцій цін. Досліджується вплив кількості вхідних змінних на точність моделі глибокого навчання у задачі прогнозування стану фондового ринку. Була проведена попередня обробка набору вхідних даних, де використовується як стандартні методи перед обробки, так і метод перетворення довжини хвилі часового ряду - вейвлет-перетворення, щоб позбутися шуму у цьому наборі. Надалі було проведено навчання та тестування архітектури моделі, використовуючи вид рекурентних нейронних мереж LSTM для прогнозування майбутнього стану фондового ринку. Проведено порівняння із базовою моделлю машинного навчання, що не має вейвлет-перетворень даних. Грунтовні емпіричні дослідження на основі набору даних біржових цін NASDAQ100 продемонстрували точність розробленої LSTM моделі, яка перевершує методи прогнозування тенденцій фінансових часових рядів без використання методу вейвлет-перетворень.

**Ключові слова**—прогнозування фінансових рядів, короткочасне прогнозування, рекурентна нейронна мережа, вейвлет-перетворення.

## ВСТУП

Прогнозування тенденцій фінансових часових рядів є складним завданням. Багато дослідників в цій галузі досліджень використовують статистичні моделі, моделі часових рядів та модель машинного навчання. Систем, що пропонують рішення задач прогнозування фінансових рядів для моніторингу стану фондового ринку зараз не так і багато. Але з'явилися моделі глибокого навчання, здатні обробляти вхідні дані фінансових рядів.

Пов'язано це з малою точністю прогнозування вже існуючих систем і на перший погляд взагалі неможливості пов'язати попередні дані цін валют з майбутніми. Тому має сенс розглядати дані про діяльність трейдингових систем як часовий ряд, а одним із можливих видів їх прогнозування є застосування рекурентних нейронних мереж. Такі мережі мають можливість короткострокової пам'яті, що є її особливістю, яка може принести значно кращі результати, порівняно з традиційними підходами в системах машинного навчання. Однак новизна полягає в використанні вейвлет-перетворень на підготовчих даних, щоб

позбавити часовий ряд шуму, що покращить здатність нейронної моделі навчатись.

## ОСОБЛИВОСТІ ПРОГНОЗУВАННЯ ФОНДОВОГО РИНКУ

Одним із поширених підходів до прогнозування часових рядів є використання систем машинного навчання для вивчення даних історії цін та подальшого їх прогнозування [1-3]. Але проблема у застосуванні нейронних моделей до фінансових рядів полягає в тому, що параметри транзакцій здебільш не піддаються встановленню залежності. Однак застосування технічних індексів є більш прогнозованими на такому ринку, які подаються у виді різниць між низькою ціною, високою ціною тощо [5, 6].

Через складність середовища фінансової біржі ціни на акції містять багато шуму, що ускладнює досягнення хорошої точності моделі при спробі спрогнозувати рух її тенденцій. Оскільки існує зацікавленість інвесторів у отриманні хорошої моделі прогнозування, виникає необхідність зменшення шуму у наборі вхідних даних, оскільки на стан фондового ринку впливає дуже багато зовнішніх факторів, зокрема, новини із різних джерел інформації [5]. Тому виникає необхідність у застосуванні певних математичних перетворень до підготовчих даних, щоб позбавити вхідний часовий ряд шуму. Найкращим варіантом буде короткотермінове прогнозування, тому що нелінійність таких систем призводить до великих змін через незначні зміни в довгостроковій перспективі. Це вказує на необхідність застосування специфічної моделі з використанням рекурентних нейронних мереж. Застосовним алгоритмом є модель LSTM (довго-короткотермінової пам'яті) для навчання та тестування архітектури моделі прогнозування. LSTM - це тип рекурентної сітки, яка здатна розділяти останні та ранні послідовності даних, даючи різні ваги для кожного, відкидаючи при цьому пам'ять, яку вона вважає недоречною для прогнозування наступного результату [4, 5].

## ПРОЕКТУВАННЯ МОДЕЛІ ПРОГНОЗУВАННЯ

Для прогнозу послідовностей фінансових часових рядів застосована LSTM модель, що має

вентилі забування, які дозволяють не сильно спиратися і враховувати занадто давні історичні дані. Тому у проект нейронної моделі прогнозування стану фондового ринку як основний включено LSTM шар нейронів.

Зібрані у набори вхідні дані необхідно передобробити та визначити розмір вікна даних набору із зсувом на 1 значення шириною в довільну кількість для того, щоб подавати на вхід моделі пакети даних, які будуть зберігати послідовність.

Для передобробки даних використовуються наступні методи: нормалізація та регуляризація для можливості їх подачі на нейронну модель. Але для випадку прогнозування фінансових часових рядів, дані мають так званий "шум", що впливає на функцію залежності прогнозу від вхідних параметрів [4].

Для позбавлення зашумленості даних застосовується метод вейвлет-перетворення, який буде трансформувати вікна даних у вейвети. Обраний метод має переваги над іншими методами обробки сигналів, оскільки враховує їх частотні характеристики. Для утворення вейвлетів, які будуть містити частотні характеристики даних, до моделі додається окремий згортковий шар передобробки.

Побудова вейвлетів за допомогою комбінації згорткових шарів нейронної моделі, здійснюється за допомогою перетворення вхідного вікна (сигналу) шляхом застосування на кожний елемент послідовності іншого фільтру. В результаті отриманий сигнал буде містити саме ті характеристики, які необхідні для виявлення частотних характеристик часового ряду. Дані нормалізовані і локалізовані в часі для того, щоб мережа була більш універсальною. Отже, отримані вейвети будуть подаватись на основний рекурентний шар нейронної моделі.

Завершальним етапом проектування – додавання шару прямого поширення, який на основі сигналу з рекурентного шару буде обчислювати прогноз наступного стану фондового ринку.

Таким чином, отримуємо нейронну модель, з інтегрованим вейвлет перетворювачем і рекурентним шаром аналізу отриманих вейвлетів, яка здатна до прогнозу стану фондового ринку на короткотермінових інтервалах часу, оскільки тенденції часто змінюються на основі впливу зовнішніх факторів.

На рис. 1 та 2 продемонстровано результати оцінки роботи нейронної моделі з та без вейвлет-перетвореннями відповідно. Виконано оцінку 3 варіантів зсуву вікна даних для прогнозу з різною дальністю від моменту завершення навчання моделі. Підтверджено, що найкращими за точністю прогнозу є перші два найближчі варіанти, а на в останньому характер поведінки прогнозної кривої вже не співпадає з фактичними даними, що демонструє складність виконання довгострокових прогнозів. Крім того, проведено

порівняльний аналіз моделі із застосуванням і без застосування вейвлет перетворень, який підтверджує необхідність і обґрунтованість їх застосування для прогнозування стану фінансових часових рядів.

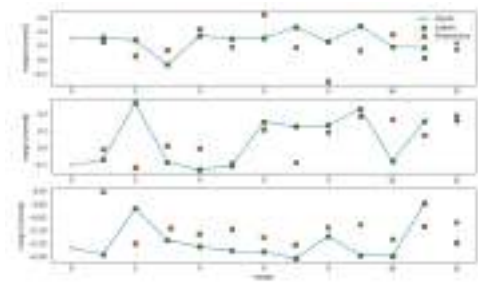


Рис. 1. Результати роботи нейронної моделі з вейвлет перетворенням

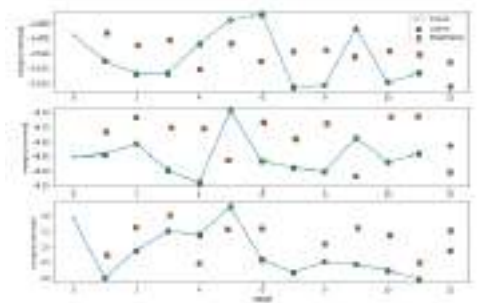


Рис. 2. Результати роботи нейронної моделі без вейвлет перетворення

## ВИСНОВКИ

Розроблена і відпрацьована методологія проектування LSTM архітектури моделі продемонструвала високу здатність здійснювати прогнозування фінансових часових рядів, якщо вони були попередньо позбавлені шуму методом вейвлет-перетворень. Її застосування створює передумови до створення систем, здатних визначати ефективні трейдингові стратегії на фінансових біржах з контролю цінних тенденцій.

## ЛІТЕРАТУРА

- [1] Grandell J. Time series analysis <https://www.math.kth.se/matstat/gru/sf2943/ts.pdf> (Last accessed 01.04.2020).
- [2] Бідюк П.І. Аналіз часових рядів: навчальний посібник - Київ: Політехніка, 2010. – 317 с.
- [3] Николенко С. Глубокое обучение - Санкт-Петербург: Питер, 2018. – 480 с.
- [4] Гудфеллоу Я. Глубокое обучение - Москва: ДМК Пресс, 2018. – 652 с.
- [5] Иванов Д.В. Прогнозирование финансовых рынков с использованием искусственных нейронных сетей – Владивосток, 2000. – 104 с. 6.
- [6] Сорнетте Д. Как предсказывать крахи финансовых рынков – Москва: Издательство И-трейд, 2008. – 400с.

Рецензент: доцент кафедри інформаційних систем та технологій КПП ім. Ігоря Сікорського к.т.н. доцент Остапченко Костянтин



# Системи зберігання великих даних

Тюрін Валерій, Дорошенко Анатолій

*КПІ ім. Ігоря Сікорського*

м. Київ, Україна

tiurINVALERY@gmail.com, a-y-doroshenko@ukr.net

**Анотація**—Робота присвячена створення нового методу зберігання та підготовки до аналізу великих даних. Метод використовує різні типи баз даних, потокову обробку й передачу даних, визначення дельти (різниця) неупорядкованих даних, мікропакетну обробку даних й маскуванню критично важливих даних. Розроблено робочий прототип системи, на якому проведено експеримент, що демонструє переваги методу.

**Ключові слова**—Data Warehouse (DWH), BigData, BigQuery, ETL.

## ВСТУП

Системи зберігання великих даних, які ще називають Data Warehouse (DWH), наразі є необхідними для кожної компанії, що має ПЗ.

Створення DWH надає компанії можливість використовувати власні історичні дані для прийняття важливих бізнес-рішень, не на основі припущень, а відштовхуючись від даних.

Для подібних систем є кілька ключових вимог: швидкий запис, висока доступність, Prevention Data Loose, Governance/Law control for PII data, наявність інтеграцій із засобами ВІ для візуалізації даних та інші.

Через постійні атаки хакерів, зловживання приватними даними великими корпораціями, державний контроль, вимоги до DWH, в тому числі на рівні законодавства, стають дедалі суворішими, що робить цю галузь важкою, але й привабливою, адже потрібно вирішувати абсолютно нові задачі. Разом зі зміною сховищ, змінюються й створювачі (продюсери) даних – смарт холодильник, годинник, смартфон, система керувати доставкою цих подій, їх обробкою, збереженням, тощо.

Створений метод під час роботи має значну практичну й наукову цінність, адже він дозволяє безпечно зберігати, швидко передавати й аналізувати приватні (PII) та лікарські (PHI) дані [1]. Це дозволяє робити трансформації у сфері охорони здоров'я, наприклад, дає можливість кожному наступному лікарю пацієнта знати всі його попередні прийоми й хвороби, при цьому не порушуючи особисті права пацієнта, що значно підвищує якість діагностування й лікування хвороби людини.

## РЕАЛІЗАЦІЯ МЕТОДУ

Незалежно від конкретної реалізації (як то BigQuery) суть (алгоритм) подібних рішень полягає у тому, щоб не працювати з усіма даними

в DWH, а лише з інкрементальними (новими даними за певний проміжок часу, або у певному вимірі) – такий підхід робить одночасно можливим збереження великої кількості даних та швидку обробку найбільш свіжих даних для аналітики.

Метод передбачає декілька основних дій з обробки та збереженню даних, а саме: прийняття повідомлень за моделлю Pub/Sub (це дозволяє організувати чергу повідомлень й гарантувати доставку кожного повідомлення, як мінімум один раз), потокова обробка прийнятих подій із window-size function, підготовка даних для збереження (первинне сортування), збереження даних у рівень початкових незмінних даних, подальша обробка й перевірка якості даних й їх переміщення на рівень чистих даних, з'єднання окремих подій у плоскі таблиці бізнес-моделі, підготовка даних плоских моделей для використання кінцевими користувачами на рівні view, архівування історичних даних на терміни, визначені вимогами законодавства до DWH.

Для збереження конфіденційності приватних даних використовується технологія Data Masking, усі поля, що мають PII дані, помічені окремою міткою Label у GCP, а Google забезпечує захист цих даних, навіть від розробників системи.

Захист PHI даних реалізовано за визнаним стандартом FHIR Data, за допомогою GCP FHIR DATA API.

Передача даних між рівнями системи реалізована за допомогою Dataflow, Airflow, що дозволяє передавати 100 тис повідомлень за хвилину на 2 робочих віртуальних машинах.

За допомогою розгортання інфраструктури на хмарних ресурсах досягається широке масштабування системи. Масштабування відбувається повністю автоматично (розробник вказує межі масштабування, тобто мінімальне, початкове і максимальне значення перед запуском системи й провайдер хмарних послуг масштабує систему в цих межах) і, що важливо, система може масштабуватися до 0, тобто, коли вона не обробляє нові дані й не зберігає даних більше ніж tier, то вона є безкоштовною – це є дуже важливо на стадії розробки для комерційних компаній, а також для наукових робітників та студентів – вони можуть спробувати модель не витрачаючи

значні кошти на інфраструктуру.

#### ТЕХНОЛОГІЇ РЕАЛІЗАЦІЇ МОДЕЛІ

Для реалізації моделі використано наступні технології.

Pub/Sub – це ресурс для прийняття повідомлень для їх подальшої потокової обробки та передачі у сховище чи іншу систему.

BigQuery – це безсерверне, масштабоване і ефективно рішення для сховища даних, що знаходиться в кількох хмарах [2].

Dataflow – це середовище виконання поточкових та пакетних задач по обробці й передачі даних засноване на основі Apache Beam SDK. Це безсерверне середовище, що забезпечує швидку обробку великих масивів даних за помірну ціну [3].

Apache Airflow – це набір бібліотек для моніторингу, запуску, організації, CI/CD інтеграції ETL/ELT задач. Широко використовується для контролю потокової передачі даних між різними середовищами, організаціями, сховищами, і т.п. [4].

DAG – смислове об'єднання задач, яке можна виконати в строгій послідовності у певний час. Зазвичай DAG включає в себе більш ніж один оператор, а основні види операторів включають: BashOperator (для запуску скрипта на Bash), PythonOperator (для виконання скрипта на мові Python), EmailOperator (для відправки листа – зазвичай використовується для внутрішніх нотифікацій), HttpOperator (для обробки Http запитів), SQLOperator (для виконання SQL запитів), Sensor (оператор очікування події – наступу певного часу, відповіді від БД, тощо) [5].

Composer – це система, що відповідальна за планування виконання DAG-ів та розподілення задач по виконавчим процесам.

#### МОНІТОРИНГ СИСТЕМИ

У системах зберігання великих даних дуже важливими складовими є моніторинг даних та узгодженість даних [6]. В даній роботі для забезпечення узгодженості й контролю за тим, що дані не втрачаються при передачі, використовувались як існуючі методи, що пропонуються хмарним провайдером, наприклад Cloud Monitoring, Cloud Logging та інші, так і власну систему й алгоритм контролю узгодженості даних – він передбачає створення окремого рівня бази даних, що контролює переміщення даних, шляхом запису усіх транзакцій, й основних ідентифікаторів транзакції й часу виконання. Таким чином завжди можна проконтролювати переміщення даних за певний проміжок часу, їх початковий та кінцевий рівень, успішність перенесення, тощо.

Системи, що працюють з BigData завжди мають мати систему автоматизованого моніторингу реального часу, адже людина не може опрацювати такі об'єми даних і виявляти нестачу незначної кількості елементів.

#### ВИСНОВКИ

Безпечно та вигідно для компанії збереження користувацьких даних вимагає значних інвестицій людських та фінансових ресурсів, проте провідну роль відіграє саме вибір правильного (безпечного, швидкого, надійного) методу збереження й передачі даних.

У результаті тестування було продемонстровано, що розроблений метод дозволяє обробляти сотні тисячі запитів за хвилину на обмежених (безкоштовних) обчислювальних потужностях.

Було продемонстровано шляхи, як система може масштабуватись від безкоштовного прототипу до системи реального розміру.

Було продемонстровано модель різних рівнів та їх призначення у DWH.

Було продемонстровано спосіб обробки й передачі даних між рівнями DWH.

Було описано способи захисту інформації у системах зберігання даних та спосіб маскування приватних даних.

Були оглянуті особливості розробки систем, що працюють з приватними даними користувачів.

Було продемонстровано можливості інтеграції систем машинного навчання для побудови системи прийняття рішень або системи допомоги у прийнятті рішень на основі даних, збережених новим методом.

Використання правильних інструментів для побудови DWH допомогло: реалізувати задачу швидше, зекономити ресурси, підвищити якість, інтегруватись з іншими системами швидко, захистити приватні дані, відобразити дані, використати дані для побудови систем прийняття рішень, в тому числі на основі машинного навчання.

Результати, що отримані під час виконання роботи можуть бути використаними ІТ-компаніями при проектуванні власних або клієнтських DWH, міграції DWH на хмарні технології, державними компаніями для організації сховищ, що мають значну кількість приватних даних громадян.

#### ЛІТЕРАТУРА

- [1] <https://www.hhs.gov/hipaa/index.html> (дата звернення 18.11.2021).
- [2] <https://cloud.google.com/bigquery> (дата звернення 18.11.2021).
- [3] <https://cloud.google.com/dataflow> (дата звернення 18.11.2021).
- [4] <https://airflow.apache.org> (дата звернення 18.11.2021).
- [5] <https://habr.com/ru/company/vk/blog/339392/> (дата звернення 18.11.2021).
- [6] <https://aws.amazon.com/data-warehouse/> (дата звернення 18.11.2021).

# Критерій оцінки вартості експлуатації мобільного транспортного робота

Кривич Ольга, Резніков Сергій

*КПІ ім. Ігоря Сікорського*

м. Київ, Україна

olgakrivich@gmail.com, reznikov\_sa@ukr.net

**Анотація**—У даному науковому дослідженні здійснений огляд проблеми пошуку оптимальних шляхів транспортними мобільними роботами. Розглянуто розроблену підсистему формування маршрутів для промислових приміщень з визначеними технологічними позиціями для виконання типових операцій, яка базується на алгоритмі пошуку найкоротшого шляху  $A^*$ . Представлено критерій для оцінювання вартості експлуатації робота на базі одного з критеріїв оптимальності маршрутів, а саме довжини пройденого шляху. Виконано порівняння значень запропонованого критерію для  $A^*$  з різними модифікаціями.

**Ключові слова**—ТСО, критерій оцінки, мобільний робот, пошук шляху, вартість експлуатації.

## Вступ

Незважаючи на сучасну роботизацію промислових підприємств за допомогою мобільних транспортних роботів, проблема оптимізації їх використання залишається актуальною дотепер [1]. Рішення, які існують зараз, вимагають повного переобладнання приміщення спеціальними коліями та кріпленнями, що вимагає великих коштів, та не завжди підходить під невеликі склади або теплиці. При виконанні завдання досягнення деякої поставленої цілі, транспортні мобільні роботи опираються на свою поточну позицію, позицію цілі та показники з різних датчиків, та слідує у напрямку до заданої точки по спеціальним коліям [1]. Такі роботи не можуть реагувати на несподівані перешкоди, шукати оптимальний шлях або бути адаптивними до змін приміщення. На жаль, такий підхід є не оптимальним, у разі появи на шляху робота перешкоди, яку неможливо об'їхати, що в свою чергу може спричинити повну зупинку технологічного процесу. Тому розробка ефективного алгоритму планування шляхів для приміщення з технологічними позиціями для завантаження та розвантаження вантажу є важливим питанням навігації мобільних транспортних роботів, оскільки оптимальність шляху впливає на ефективність виконання завдань. Побудований шлях повинен задовольняти хоча б одному з критеріїв оптимальності, наприклад, загальній пройденій відстані роботом, яка в свою чергу може впливати на використані їм ресурси, такі як електроенергія, паливо або загальний час виконання технологічної задачі. Тож у даній робо-

ті розроблено підсистему пошуку оптимального маршруту для мобільного транспортного робота на базі алгоритму  $A^*$ , на основі якого розраховується критерій оцінки вартості експлуатації залежно від критерію оптимальності маршруту, а саме довжини шляху.

## ПІДСИСТЕМА ПОШУКУ ОПТИМАЛЬНОГО МАРШРУТУ

При виконанні черги послідовних типових операцій (завантаження, розвантаження, очікування) на визначених технологічних позиціях по попередній відомій карті, постає проблема створення карти, визначення усіх ключових позицій, подальше їх запам'ятовування системою керування роботом, а також побудова оптимального шляху залежно від поточної задачі.

В сучасних умовах для побудови карти можна використати малогабаритного робота для побудови первинної карти, потім завантажити її в систему управління вантажними роботами та скласти граф доріг, з урахуванням технологічних позицій, розмірів вантажних роботів та інших параметрів середовища. Таким чином, отримати шляхи до кожної технологічної ділянки, та оптимальні маршрути руху, без небезпеки для обладнання, вантажу та персоналу.

Важливим критерієм розв'язання даної задачі складає час виконання циклу задач. Висока швидкість роботи для зменшення часу може перешкоджати точності та повторюваності руху та бути небезпечною для вантажу, який перевозить робот, оскільки від виконавчих механізмів та системи управління потрібні надзвичайні характеристики. Тому приділено особливу увагу створенню траєкторії руху, яка могла б бути виконана за відносно малий час, але водночас безпечна для робота, з точки зору уникнення надмірних прискорень приводів та вібрацій механічної конструкції.

Для розбиття карти приміщення на опорні точки та побудову графу шляхів розраховується кількість метрів, що припадає на одну клітинку карти (піксель), тобто роздільна здатність карти. На основі цих даних розраховується міні-

мальний розмір ділянки мапи, в яку може поміститися робот, уникаючи перешкоди, за (1).

$$cellSize = [robotSize/mapResolution], \quad (1)$$

де  $cellSize$  – розмір комірки,  $robotSize$  – розмір робота,  $mapResolution$  – роздільна здатність карти.

Далі визначаються опорні точки рівновіддалені одна від одної на розмір ділянки та розташовані в прохідних зонах мапи. Використовуючи метод сітки можна перевірити досяжність кожної точки графу шляхів, виконавши перевірки на наявність непрохідних ділянок в кожній з клітинок та з'єднавши всі сусідні точки в один граф. Отримаємо карту (граф) шляхів, показаний на рис 1.

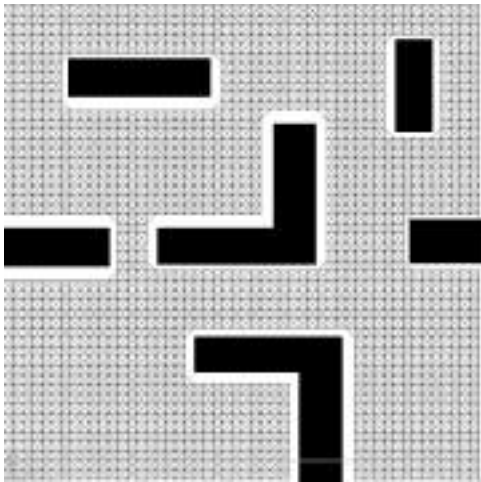


Рис. 1. Граф шляхів

Далі проводиться формування маршруту. Суть розробленого алгоритму складається з того, щоб замість пошуку повного найкоротшого шляху від початкової точки до кінцевої, з врахуванням черги технологічних позицій, проводити поступове розбиття усього шляху на найкоротші відрізки від одної технологічної позиції до іншої (рис. 2).

#### КРИТЕРІЙ ОЦІНКИ ВАРТОСТІ ЕКСПЛУАТАЦІЇ МОБІЛЬНОГО ТРАНСПОРТНОГО РОБОТА

Транспортні мобільні роботи довели, що здатні підвищувати продуктивність і знижувати витрати на автоматизацію складських та промислових приміщень. Хоча прямі витрати продовжують знижуватися, часто виникають приховані витрати, пов'язані з їх розгортанням, підтримкою, обслуговуванням та модернізацією. Впроваджуючи таких роботів на підприємстві, важливо враховувати як початкові, так і поточні зусилля з налаштування, розгортання та реконфігурації, які можуть мати велику вартість, якщо роботизованою системою складно користуватися під час їх експлуатації. Для оцінки економічної ефективності розробленого алгоритму пошуку

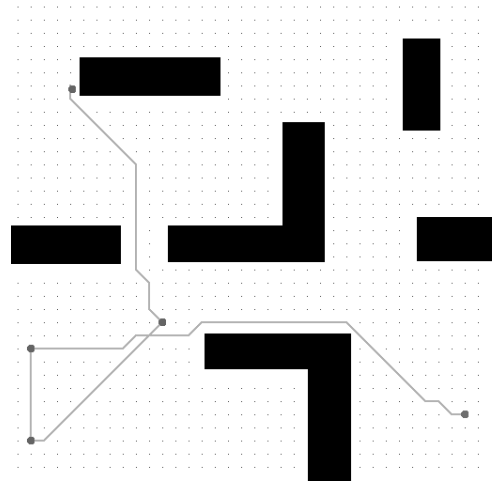


Рис. 2. Повний шлях від початкової до кінцевої точки

найкоротшого шляху та переваг його використання застосований метод ТСО. ТСО, або Сукупна вартість володіння, – це сучасний метод оцінки ефективності придбання, володіння та експлуатації обладнання та комплектуючих [2]. ТСО представляє суму всіх очевидних і всіх прихованих витрат, пов'язаних з активом за загальний період володіння, за загальний термін корисного використання та протягом терміну експлуатації активу. На сьогоднішній день ТСО майже не використовувався для розрахунку вартості експлуатації на базі критеріїв оптимальності. Основні розрахунки ТСО проводяться за (2) [3]:

$$TCO = C_{ownership} + C_{service} + C_{idleTime}, \quad (2)$$

де  $C_{ownership}$  – це витрати на закупівлю,  $C_{service}$  – витрати на обслуговування,  $C_{idleTime}$  – витрати на простій. Для детального розрахунку ТСО необхідно врахувати прямі та непрямі витрати [4], які можуть виникнути при впровадженні та використанні розробленої підсистеми формування руху робота (табл. 1).

Одним з головних показників ефективності роботи даної підсистеми пошуку шляху є довжина шляху [5]. Вона безпосередньо впливає на швидкість виконання операцій роботом, витрати на електроенергію, знос деталей, бо чим менше проїде робот, тим повільніше зносяться його комплектуючі, і т.д. Тому для підсистеми пошуку шляху критерій оптимальності за місяць розрахований як відношення довжини шляху до вартості експлуатації за (3)

$$TCO_{month} = \frac{avgPath * pathPerMonth}{\sum C_{initialCost} + \sum C_{hiddenCost}}, \quad (3)$$

де  $avgPath$  – середня довжина шляху відвідування технологічних позицій, обчислена за формулою (5),  $pathPerMonth$  – довжина шляху, яку пройшов робот за місяць, розраховане за

Табл. I  
ПРЯМІ ТА ПРИХОВАНІ ВИТРАТИ

Вид витрат	Витрати	
	Прямі	Приховані
Витрати на закупівлю	1) початкова ціна за підсистему пошуку шляху; 2) придбання транспортного мобільного робота; 3) налаштування системи.	1) навчання працівників. 2) створення мапи.
Витрати на обслуговування	1) заміна комплектуючих робота; 2) оновлення підсистеми пошуку шляху.	1) електроенергія; 2) витрати на Інтернет; 3) переналаштування мапи.
Витрати на простій	1) запланований простій через оновлення підсистеми.	1) аварійна зупинка робота або його частин; 2) збій системи.

формулою (4), *initialCosts* – це прямі витрати, *hiddenCosts* – непрямі витрати,

$$pathPerMonth = avgPath * repeatCount * 28, \quad (4)$$

де *repeatCount* – кількість повних об'їздів роботом 4 технологічних позицій за день,

$$avgPath = \frac{\sum_{i=0}^n pathLength}{n}, \quad (5)$$

де *pathLength* – середня довжина шляху, виміряна на основі *n* вимірів.

Порівняння ТСО для різних алгоритмів пошуку шляху

Для аналізу ефективності роботи системи на основі алгоритму A\* необхідно порівняти ТСО зі значеннями роботи різних модифікацій. Для даних алгоритму проведено заміри довжини шляху та кількості поворотів, розраховані для прикладу з відвідуванням 4 технологічних позицій в приміщенні розміром 2500m<sup>2</sup> (рис. 3).

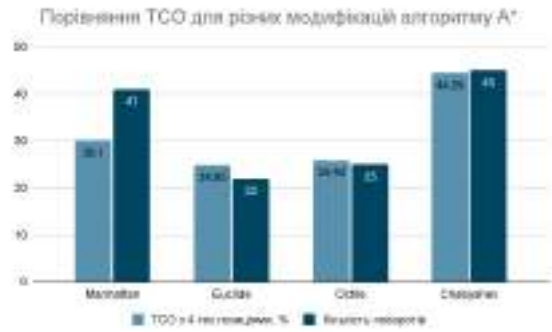


Рис. 3. Діаграма результатів ТСО для різних модифікацій алгоритму A\*

ВИСНОВКИ

Запропоновано підсистему пошуку оптимального маршруту для мобільного транспортного робота на основі алгоритму A\* для виконання типових задач в технологічних позиціях у промислових приміщеннях. Сформульовано критерій для оцінювання вартості експлуатації мобільного транспортного робота, який базується на критеріях оптимальності маршруту – довжині шляху та кількості поворотів. Проведено порівняльний аналіз різних модифікацій для алгоритму A\* та розраховано значення критерію для кожного з них.

ЛІТЕРАТУРА

- [1] Chen, Xiaoqi&Chen, Y.Q.&Chase, James. (2009). Mobiles Robots - Past Present and Future. 10.5772/6986.
- [2] Total Cost of Ownership [Електронний ресурс] // Режим доступу: <https://www.wallstreetmojo.com/total-cost-of-ownership/>.
- [3] Barbusova, Miroslava&Medvecká, Iveta&Gašo, Martin. (2019). Use of TCO Analysis in Industry 4.0. 10.24132/PI.2019.08948.010-017.
- [4] The Hidden Costs of Autonomous Mobile Robots – And How to Avoid Them [Електронний ресурс] // Режим доступу: <https://www.roboticsbusinessreview.com/opinion/the-hidden-costs-of-autonomous-mobile-robots-and-how-to-avoid-them/>.
- [5] Лавренов Р.О., Афанасьев И.М., Магид Е.А. Планирование маршрута для беспилотного наземного робота с учетом множества критериев оптимизации // Третий Всероссийский научно-практический семинар «Беспилотные транспортные средства с элементами искусственного интеллекта. 2016. С. 10–20.

**ТЕОРІЯ КЕРУВАННЯ**

**CONTROL THEORY**

# Синтез багатовимірних систем керування з неквадратними матрицями

Репнікова Наталія, Гнип Владислав

КПІ ім. Ігоря Сікорського

м. Київ, Україна

n.repnikova@gmail.com, vladyslavgnyp@gmail.com

**Анотація**—В роботі запропоновано розширення можливостей синтезу структур лінійних багатовимірних систем з неквадратними матрицями на базі формули Р. Ізермана. Розглядаються цифрові системи керування, що описуються моделями простору станів. Показано, що виконання умови коректності вибору розмірностей матриць рівнянь стану і виходу, дозволяє для досліджуваного класу систем, отримати результати синтезу з заданими показниками якості. Виконано моделювання цифрової системи з неквадратними матрицями з використанням прикладного пакету MATLAB/Simulink.

**Ключові слова**—цифрові системи, багатовимірні системи, підвищення якості, цифровий регулятор станів, модель простору станів, неквадратні матриці, моделі MATLAB.

## Вступ

На теперішній час опубліковано цілий ряд робіт, які розглядають теоретичні питання аналізу та синтезу цифрових систем на базі методу простору станів. Як відомо, при використанні зазначеного методу, динаміка систем описується диференціальними рівняннями стану та виходу виду:

$$\begin{aligned} \mathbf{x}[n+1] &= \mathbf{A}\mathbf{x}[n] + \mathbf{B}\mathbf{u}[n], \\ \mathbf{y}[n] &= \mathbf{C}\mathbf{x}[n] + \mathbf{D}\mathbf{u}[n], \end{aligned} \quad (1)$$

де  $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$  — матриці відповідних розмірностей.

Хоча кожна досліджувана модель має свої особливості, при виконанні синтезу, всі вони об'єднуються головним змістом, а саме створенням зворотного зв'язку за станом, який забезпечує задане розташування полюсів замкнутої системи. При цьому вираз для керуючого впливу має вигляд:

$$\mathbf{u}[n] = -\mathbf{K}\mathbf{x}[n], \quad (2)$$

де  $\mathbf{K}$  - матриця зворотних зв'язків за станом.

Слід зазначити, що теоретичне обґрунтування задач синтезу моделей з одним входом та одним виходом широко представлені у літературі [1]. Для багатовимірних систем існує поняття автономних та багатозв'язних систем [2]. Якщо для перших є достатня кількість прикладів вирішення задач синтезу, то для багатозв'язних систем іноді дослідникам доводиться виконувати авторське теоретичне обґрунтування запропонованих рішень. В роботі розглядаються питання синтезу багатозв'язної цифрової системи на базі

відомої формули Р. Ізермана. При цьому задача ускладнюється вибором неквадратних матричних структур.

## ОСНОВНИЙ МАТЕРІАЛ

У відповідності до формули Р. Ізермана [3], матриця зворотних зв'язків за станом розраховується як

$$\mathbf{K} = [\mathbf{C}\mathbf{B}]^{-1}[\mathbf{C}\mathbf{A} - \mathbf{T}\mathbf{C}], \quad (3)$$

де  $\mathbf{T}$  - матриця власних значень системи.

Запишемо вираз (3) через розмірності матриць в загальному вигляді для визначення результуючої розмірності матриці  $\mathbf{K}$ .

$$\begin{aligned} [\mathbf{K}] &= [(l \times n)(n \times r)]^{-1} [(l \times n)(n \times n) - (l \times l)(l \times n)] = \\ &= [(l \times r)]^{-1} [(l \times n) - (l \times n)] = (r \times l)(l \times n) = (r \times n). \end{aligned} \quad (4)$$

Для забезпечення заданих показників якості використовуємо формулу визначення корегуючих коефіцієнтів.

$$\mathbf{K}^* = \mathbf{T}^{-1} \mathbf{K}_h^{-1} \begin{bmatrix} \frac{1}{1-z_1} & 0 \\ 0 & \frac{1}{1-z_2} \end{bmatrix}^{-1} \mathbf{E}. \quad (5)$$

$$\mathbf{K}_h = \begin{bmatrix} k_{h11} & 0 & 0 \\ 0 & k_{h22} & 0 \\ 0 & 0 & k_{hlr} \end{bmatrix}, \quad (6)$$

$$k_{hij} = c_{ii} b_{jj} \quad i = 1, 2..l; j = 1, 2..r,$$

де  $z_1, z_2$  - бажані корені.

Підставивши в формулу (5) розмірності відповідних матриць, визначимо умову коректності вибору розмірностей для можливості проведення синтезу багатозв'язної цифрової системи з неквадратними матрицями:

$$\begin{aligned} [\mathbf{K}^*] &= (l \times l)^{-1} (l \times r)^{-1} (l \times l)^{-1} (l \times l) = \\ &= (l \times l)(r \times l)(l \times l)(l \times l). \end{aligned} \quad (7)$$

Таким чином, відповідно до виразу (7), використати метод синтезу для неквадратних матриць можна лише у випадку коли  $r = l$ .

## ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

Для векторно-матричної моделі, яка описує динаміку цифрової багатовимірної системи з відповідними матрицями виду (8).

$$\mathbf{A} = \begin{bmatrix} -0,3 & 3 & -1 \\ -0,1 & -0,5 & 0 \\ 0,3 & -0,7 & -0,1 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix}, \quad (8)$$

$$\mathbf{C} = \begin{bmatrix} 1,2 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

Виконаємо синтез регулятора, який буде забезпечувати нульове перерегулювання, нульову усталену помилку та скорочення часу керування. В системі діють два вхідні впливи, відповідно 1 та 4.

Моделювання виконується за допомогою пакету MATLAB/Simulink.

На рис.1 представлені графіки перехідних процесів вихідної системи за двома виходами.

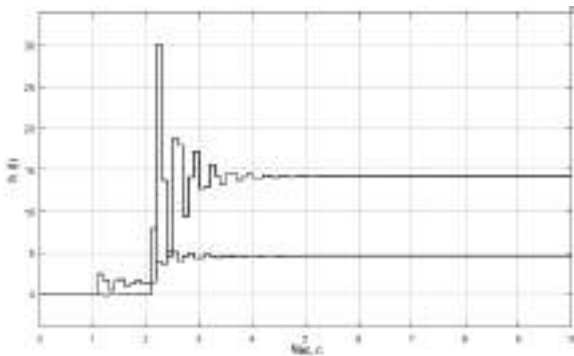


Рис. 1. Графіки перехідних процесів вихідної системи

Як видно з графіків, кожен стан системи впливає на інший, присутні велике перерегулювання та усталена помилка керування.

З урахуванням умови коректності вибору розмірностей матриць формул (1), (3) розраховуємо матрицю коефіцієнтів спостерігаючого пристрою (9).

$$\mathbf{H} = \begin{bmatrix} -0,25 & 3 \\ -0,0833 & -0,4 \\ 0,25 & -0,7 \end{bmatrix}. \quad (9)$$

Використовуючи (9) розраховуємо матриці моделі простору-станів спостерігаючого пристрою, на основі яких будується об'єднаний регулятор (11) за формулами (10).

$$\mathbf{A}_p = [\mathbf{A}_{sp} - \mathbf{BK}],$$

$$\mathbf{B}_p = [\mathbf{B} \quad \mathbf{H}],$$

$$\mathbf{C}_p = [-\mathbf{K}],$$

$$\mathbf{D}_p = [\mathbf{E} \quad \mathbf{0}]. \quad (10)$$

$$\mathbf{A}_p = \begin{bmatrix} 0,4 & -3 & 0 \\ 0,1 & 0,5 & 0 \\ 0 & 0 & -0,1 \end{bmatrix},$$

$$\mathbf{B}_p = \begin{bmatrix} 2 & 0 & -0,25 & 3 \\ 0 & 2 & -0,833 & -0,4 \\ 0 & 0 & 0,25 & -0,7 \end{bmatrix}, \quad (11)$$

$$\mathbf{C}_p = \begin{bmatrix} 0,2 & -1,5 & 0,5 \\ 0,05 & 0,3 & 0 \end{bmatrix},$$

$$\mathbf{D}_p = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$



Рис. 2. Структурна схема з об'єднаним регулятором

Результати моделювання представлені на рис.3.

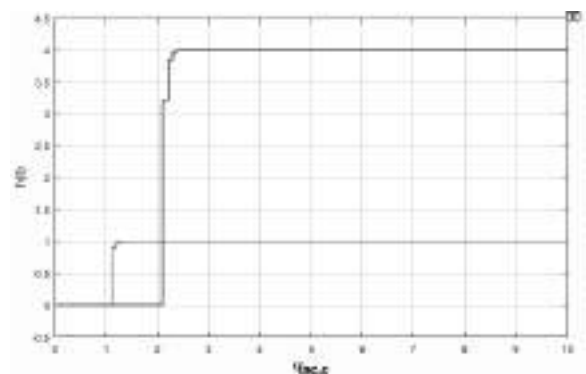


Рис. 3. Перехідні процеси моделі з об'єднаним регулятором

Як видно з графіків перехідного процесу, синтез регулятора забезпечує задану якість багатовимірній цифровій системі.

## ВИСНОВКИ

Таким чином, у роботі виведена умова коректності вибору розмірностей відповідних матриць для забезпечення можливості проведення синтезу цифрової багатовимірної системи з неквадратними матрицями.

При цьому розширюється границя області використання формули Р.Ізермана для забезпечення розв'язання перехресних зв'язків за станом.

## ЛІТЕРАТУРА

- [1] Куо Б. Теория и проектирование цифровых систем управления / Перевод с англ. – М.: Машиностроение, 1986, 448 с.
- [2] Ким Д.П. Теория автоматического управления Т.2 Многомерные, нелинейные, оптимальные и адаптивные системы/ Д.П. Ким – М.:Физматлит, 2007.-440 с.
- [3] Изерман Р. Цифровые системы управления/ Пер. с англ.- М.: Мир, 1984, 541с.



# Дослідження методу цифрової апроксимації для вибору періоду квантування за часом

Репнікова Наталія, Висоцька Юлія

КПІ ім. Ігоря Сікорського

м. Київ, Україна

n.repnikova@gmail.com, julja.vysotska@gmail.com

**Анотація**—В роботі розглядаються питання синтезу цифрових систем керування на базі математичної платформи передавальних функцій та структурних схем. Запропоновано використання методу цифрової апроксимації для уточнення значення періоду квантування за часом для забезпечення функціонування цифрової системи в стійкій області. Показано, що виконання умови коректності вибору періоду квантування, дозволяє для досліджуваного класу систем, отримати результати синтезу з заданими показниками якості. Виконано моделювання цифрової системи з використанням прикладного пакету MATLAB/Simulink.

**Ключові слова**—цифрові системи, передавальні функції, Z-перетворення, період квантування за часом, цифрова апроксимація, підвищення якості, моделі MATLAB.

## ВСТУП

На теперішній час опубліковано цілий ряд робіт, які розглядають теоретичні питання аналізу та синтезу цифрових систем на базі передавальних функцій та структурних схем. Всі автори однозначно відзначають важливість моменту вибору такого параметру цифрової системи, як періоду квантування за часом [1, 2, 3]. По-перше, його слід обирати з умови імпульсної теореми (теорема Котельникова), по-друге, з умови забезпечення стійкості досліджуваної цифрової системи.

Як відомо, при використанні зазначеного методу, динаміка систем описується диференціальними рівняннями з подальшим визначенням передавальної функції (області  $s$ ) та переходу до передавальної функції цифрової системи з використанням Z-перетворення  $W(z)$ .

Хоча описані процедури є класичними кожна досліджувана цифрова система має свої особливості, і при виконанні аналізу та синтезу, інженери—науковці створюють умови функціонування систем у стійкій площині, опікуючись питанням вибору періоду квантування за часом.

В роботі запропоновано для цифрової апроксимації керованих процесів використовувати метод Z-форм та визначати співвідношення параметру періоду квантування з іншими параметрами системи.

## ОСНОВНИЙ МАТЕРІАЛ

У відповідності до методу Z-форм [1] передавальну функцію цифрової системи можна записати у відповідності за наступним алгоритмом:

- за заданою передавальною функцією у Лапласі знаходять передавальну функцію замкнутої системи;
- записують перетворення за Лапласом вихідного сигналу  $Y(s)$  у вигляді раціональної функції за степенями  $s$ ;
- замінюють  $s$  відповідними Z-формами, та в кінці ділять на період квантування за часом ( $T$ );
- з використанням Z-перетворення одиничного стрибка, записують передавальну функцію.

Пропонується для визначення значення періоду квантування за часом, який забезпечує функціонування цифрової системи у стійкій області використовувати співвідношення зазначеного параметру до значення коефіцієнту підсилення ( $K$ ) досліджуваної системи. При цьому до визначеної передавальної функції замкнутої системи та відповідного характеристичного рівняння застосовуються критерії стійкості та будується залежність значень  $K$  і  $T$ , які відповідають стійкості системи.

## ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

На прикладі системи третього порядку, яка описується передавальною функцією виду

$$W(s) = \frac{K}{(T_1s + 1)(T_2s + 1)(T_3s + 1)} \quad (1)$$

виконаємо вибір періоду квантування за часом  $T$  запропонованим методом.

Обчислимо передавальну функцію замкнутої системи як:

$$W_3(s) = \frac{K}{As^3 + Bs^2 + Cs + 1 + K}, \quad (2)$$

де  $A = T_1T_2T_3$ ,  $B = T_1T_2 + (T_1 + T_2)T_3$ ,  $C = (T_1 + T_2)T_3$ .

Представимо вихідну функцію, як

$$Y(s) = \frac{Ks^{-4}}{A + Bs^{-1} + Cs^{-2} + (1 + k)s^{-3}} \quad (3)$$

та використовуючи відповідні Z-форми, отримаємо

$$Y(z) = \frac{K \frac{T^4}{6} \cdot \frac{z^{-1} + 4z^{-2} + z^{-3}}{(1-z^{-1})^4} \cdot \left(-\frac{T^4}{720}\right)}{A + B \frac{T}{2} \left(\frac{1+z^{-1}}{1-z^{-1}}\right) + C \frac{T^2}{12} \cdot \left(\frac{1+10z^{-1}+z^{-2}}{(1-z^{-1})^2}\right) + (1+K) \frac{T^3}{1} \cdot \frac{z^{-1}+z^{-2}}{(1-z^{-1})^3}} \quad (4)$$

Використовуючи критерії стійкості можна показати, що значення  $K$  і  $T$ , відповідають стійкій моделі, яка зв'язана залежністю приведеною на рис.1

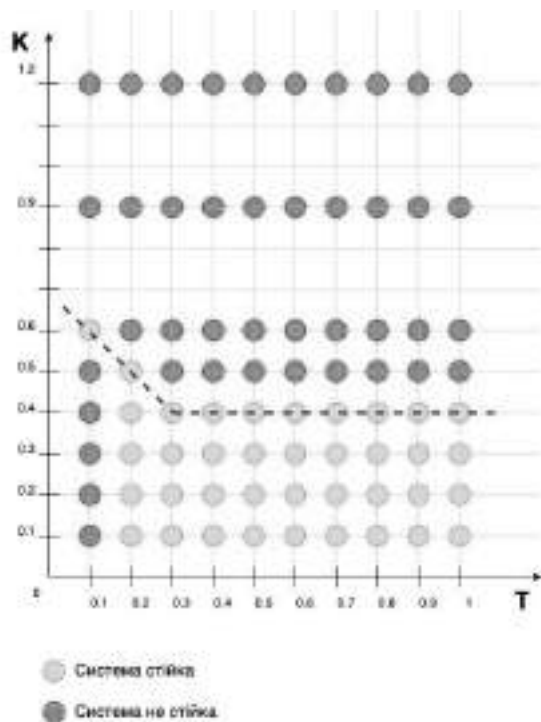


Рис. 1. Результати досліджень стійкості системи, де  $T_1=10$ ,  $T_2=0.5$ ,  $T_3=20$

Перевіримо отримані результати з використанням моделювання досліджуваної цифрової системи за допомогою MATLAB/Simulink. Будемо виконувати моделювання цифрових систем з вибраним співвідношенням коефіцієнту підсилення та періоду квантування області стійкої та нестійкої систем (рис.1).

Варіант 1. Модель передавальної функції цифрової системи в Simulink при  $\omega=0.1$  і  $\omega=0.5$ с (стійка область) та відповідно графік перехідного процесу рис.2, 3.

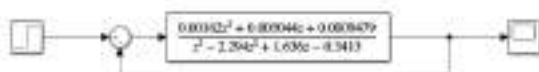


Рис. 2. Модель цифрової системи

Варіант 2. Модель передавальної функції цифрової системи в Simulink при  $\omega=0.5$  і  $\omega=0.5$ с (нестійка область) та відповідний перехідний процес зображено на рис.4, 5.

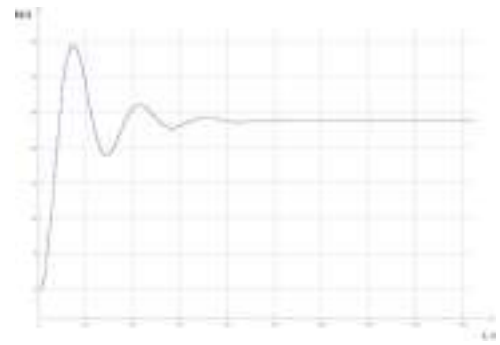


Рис. 3. Графік перехідного процесу

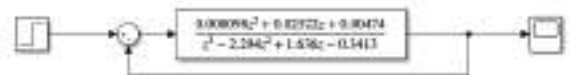


Рис. 4. Модель цифрової системи

Результати моделювання представлені на рис.5.

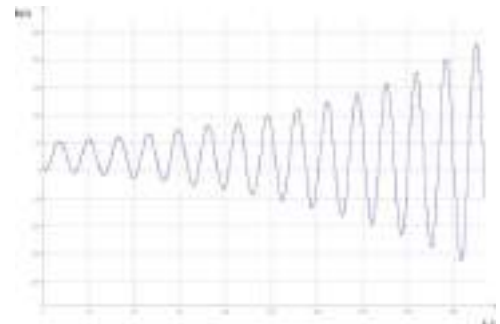


Рис. 5. Графік перехідного процесу

Як видно, з представлених графіків перехідних процесів, вибране співвідношення коефіцієнту підсилення та періоду квантування за часом з стійкої та нестійкої областей демонструє повну адекватність запропонованої методики.

#### ВИСНОВКИ

У даній публікації було обчислено співвідношення параметрів цифрової системи до періоду квантування за часом, яке забезпечує функціонування системи у стійкій області. Для цифрової апроксимації керованого процесу використовувався метод z-форм. За допомогою проведених дослідів з моделювання досліджуваних цифрових систем показана достовірність отриманих результатів.

#### ЛИТЕРАТУРА

- [1] Куо Б. Теория и проектирование цифровых систем управления / Перевод с англ. – М.: Машиностроение, 1986, 448 с.
- [2] Олссон Г. Цифровые системы автоматизации и управления / Г. Олссон, Д. Пиани / перевод с англ. – Невский диалект, 2001, 3-е издание.
- [3] Поляков К.Ю. Основы цифровых систем управления / Изд-во СПбГМУ, 2012, 154с.



**ШТУЧНИЙ ІНТЕЛЕКТ**

**ARTIFICIAL INTELLIGENCE**

# Система розподіленого навчання нейронних мереж

Нікулін Олексій  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна  
firlbit@gmail.com

**Анотація**—У даній роботі наведено огляд розподіленої системи, її основні характеристики та задачі, які необхідно вирішити при реалізації системи розподіленого навчання нейронних мереж. Зокрема, розглядаються процеси отримання, трансформації, завантаження даних та безпосередньо розподіленого навчання нейронних мереж. Також коротко описано можливі способи реалізації цих задач.

**Ключові слова**—розподілена система, розподілений розрахунок, розподілене навчання нейронних мереж, ETL.

## Вступ

З кожним роком людство генерує все більше і більше даних, які потребують подальшого опрацювання та аналізу. Такі процеси на сьогодні стали майже буденністю, і важко знайти сферу, у якій не відбувається аналітичних оцінок та розрахунків. Дані процеси можуть потребувати багато часу або вимагати створення складних алгоритмів для їх вирішення. Сьогодні для вирішення подібних задач все частіше застосовують штучні нейронні мережі.

Зазвичай, при збільшенні кількості даних, збільшується їх варіативність та складність. І, як результат, зростає складність як самих задач, що вимагають вирішення, так і моделей нейронних мереж. Навчання таких нейронних мереж може потребувати великої кількості часу та розрахункових ресурсів комп'ютера. Одним із варіантів вирішення даної проблеми є використання розподіленого навчання. Таким чином для того, щоб побудувати систему розподіленого навчання потрібно вирішити дві основні задачі: отримання, трансформації та завантаження даних і безпосередньо саму задачу розподіленого машинного навчання.

## Визначення розподіленої системи

Розподілена система є системою, що складається з декількох частин розміщених на різних машинах. Ці машини обмінюються даними і координують свої дії таким чином, щоб для кінцевого користувача вони виглядали, як єдине ціле. Компонентами розподіленої системи можуть бути комп'ютери, фізичні сервери, віртуальні машини, контейнери або будь-які інші вузли, що під'єднані до спільної мережі, мають локальну пам'ять і взаємодіють між собою [1]. Є два

основних підходи реалізації розподіленої системи. Перший – коли кожна машина працює для досягнення загальної мети і кінцевий користувач отримує результати у вигляді єдиного цілого. Другий – коли кожна машина має свого кінцевого користувача, а розподілена система допомагає у використанні спільних ресурсів.

Хоч структура розподілених систем може бути складною у них є три основні характеристики [1]: всі компоненти мережі працюють одночасно; мережі не мають глобального годинника; компоненти мережі виходять з ладу незалежно один від одного.

Зважаючи на вище описані характеристики, можна виділити переваги та недоліки розподілених систем. Першою перевагою даних систем є горизонтальна масштабованість. Оскільки розрахунки виконуються незалежно на кожному вузлі системи то недорого та просто додавати нові вузли та функції. Другою перевагою розподілених систем є відмовостійкість. Оскільки розподілена система може складатися з сотень вузлів, які працюють разом, то вона не перестане виконувати свою задачу, якщо одна машина виходить з ладу. Третьою основною перевагою є продуктивність. Вона досягається за рахунок того, що робоче навантаження на систему може бути розділене і відправлене на декілька різних машин.

До недоліків можна віднести наступне: розподілені системи повинні вирішувати, які завдання, коли та де необхідно виконати; затримка, чим більше система розподілена, тим більше затримок може виникати при обміні даними; спостереження за станом системи, збір, обробка і моніторинг показників використання ресурсів для великих кластерів може створити велику проблему.

## ОТРИМАННЯ, ТРАНСФОРМАЦІЯ ТА ЗАВАНТАЖЕННЯ ДАНИХ

Задача про отримання, обробці та завантаженню даних є однією з основних при розробці системи розподіленого машинного навчання. Оскільки розміри наборів даних для навчання нейронних моделей можуть бути досить великими, то завантаження та обробка даних із використанням класичних підходів буде займати велику

кількість часу. Тому у даному випадку варто реалізувати ETL процес. ETL (Extract, Transform і Load) процес – це налагоджений конвеєр даних, що використовується для збору і трансформації даних із наступним завантаженням їх до відведеного сховища даних [2]. Робота по перетворенню в ETL виконується у спеціальному рушії даних та вимагає використання додаткових проміжних таблиць для тимчасового збереження даних до моменту завантаження кінцевого результату.

Трансформація даних зазвичай передбачає використання таких операцій, як фільтрування, сортування, агрегування, перевірка даних, видалення дублікатів, очистка і об'єднання даних. Часто фази отримання, трансформації та завантаження даних відбуваються паралельно для збереження часу. На практиці для реалізації ETL процесу використовують рушії даних, які, зазвичай, також працюють як розподілені системи для досягнення необхідного рівня паралелізації та ефективності виконання конвеєру даних [2]. Прикладом такого інструменту є Apache Spark. На сьогодні це один з найрозповсюдженіших рушіїв даних, що використовують для реалізації розподіленої обробки неструктурованих і слабо-структурованих даних у реальному часі із малими затримками. Проміжним або цільовим сховищем даних на практиці є розподілена файлова система. Прикладом може бути HDFS з Apache Hive або Apache Spark.

#### Розподілене машинне навчання

З ускладненням задач, збільшенням наборів даних та їх варіативності, що використовуються для навчання моделей нейронних мереж, збільшилась кількість ресурсів та час необхідний для навчання моделі. Тому для підвищення швидкості та ефективності виконання машинного навчання почали розробляти та використовувати розподілене машинне навчання.

Розподілене машинне навчання – це підхід, при якому процес навчання відбувається розподілено з використанням принципів розподілених систем. За способом розділення воно може бути реалізоване за допомогою розподілення наборів даних по вузлах кластеру (Data Parallelizing) [3]. При такому підході кожен вузол містить однакову модель, але проводить навчання на різних вхідних даних. Data Parallelizing корисно використовувати, коли набір даних є занадто великим для опрацювання його одним вузлом.

Іншим підходом є Model Parallelizing. При такому підході модель ділиться на частини та розподіляється по вузлах кластеру, але дані на всі вузли поступають однакові. Model Parallelizing корисно використовувати у випадку, коли модель машинного навчання є занадто складною для виконання навчання на одному вузлі [3].

За способом взаємодії вузлів, реалізації можна поділити на централізоване та децентралізоване

[3]. При першому підході виділяється окремий вузол або група вузлів, що відповідають за синхронізацію інших, що виконують навчання. При децентралізованому кожен з вузлів відповідає за навчання і взаємодіють один з одним напряму для оновлення параметрів моделі.

За способом синхронізації: синхронний та асинхронний [3]. При першому, вузли надсилають значення параметрів моделі серверу параметрів та очікують доки всі інші теж не надішлють. При використанні даного підходу значно збільшується точність моделі, але і збільшується час навчання. А при асинхронному, вузли відправляють свої результати так швидко наскільки можливо. При використанні даного підходу збільшується завантаження кластеру, але також із ним і швидкість навчання.

На практиці існує багато інструментів, що реалізують різні підходи розподіленого машинного навчання. Одні з найбільш розповсюджених це TensorFlow, Horovod, PyTorch та ін. [4].

#### ВИСНОВКИ

У роботі було розглянуто поняття розподіленої системи, визначено її основні характеристики, переваги та недоліки. Також було виділено та розглянуто дві основні задачі, які повинні бути вирішені у системі розподіленого навчання нейронних мереж. Першою є процес отримання, трансформації та завантаження даних (ETL). Він використовується для більш ефективного та швидкого збору та обробки даних і реалізується за допомогою використання рушіїв даних для розподіленої обробки даних. Другою – розподілене машинне навчання. Є декілька способів її реалізації (Model або Data Parallelizing, централізований або децентралізований та синхронний або асинхронний), які потрібно обирати у відповідності до поставленої задачі. Результати даної роботи можуть бути використані для побудови розподілених систем, що повинні працювати із великими даними та/або систем для розподіленого машинного навчання.

#### ЛІТЕРАТУРА

- [1] Gibb R. What is a Distributed System? [Електронний ресурс] / Robert Gibb – Режим доступу до ресурсу: <https://blog.stackpath.com/distributed-system/>.
- [2] Wilson M. Extract, transform, and load (ETL) [Електронний ресурс] / M. Wilson, D. Kshirsagar – Режим доступу до ресурсу: <https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl>.
- [3] Machiraju S. How to train your deep learning models in a distributed fashion. [Електронний ресурс] / Srikanth Machiraju – Режим доступу до ресурсу: <https://towardsdatascience.com/how-to-train-your-deep-learning-models-in-a-distributed-fashion-43a6f53f0484>.
- [4] A Survey on Distributed Machine Learning [Електронний ресурс] / [J. Verbraeken, W. Matthijs, K. Jonathan та ін.] – Режим доступу до ресурсу: <https://arxiv.org/ftp/arxiv/papers/1912/1912.09789.pdf>

*Рецензент: доцент кафедри інформаційних систем та технологій КПІ ім. Ігоря Сікорського к.т.н. доцент Цьопа Наталія*



**БЕЗПЕКА ТА ЗАХИСТ  
ІНФОРМАЦІЇ**

**SECURITY AND  
PROTECTION OF  
INFORMATION**



# Смарт-контракти NFT для оренди нерухомості

Ходарченко Андрій, Полторак Вадим  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна

**Анотація**—В роботі розглянуто NFT для оренди нерухомості, а саме основні проблеми зберігання даних у блокчейні, варіанти їх вирішення, система смарт-контрактів NFT для оренди нерухомості, основні проблеми такої системи та варіанти вирішення цих проблем. Розглянуто недоцільність зберігання великих за об'ємом даних в блокчейні та можливість зберігання цих даних поза системою. Також наведено приклад системи для оренди нерухомості, проблему реєстру, розташованого поза блокчейном та можливі варіанти вирішення цієї проблеми, шляхом встановлення довіри до реєстра.

**Ключові слова**—блокчейн, NFT, невзаємозамінні токени, база даних, смарт-контракт, криптовалюта, оренда нерухомості.

## ВСТУП

Невзаємозамінні токени (NFT, non-fungible token) — це спеціальний вид криптовалюти, який є невзаємозамінним, тобто його унікальне і єдине існування в блокчейні підтверджує смарт-контракт. Це його головна відмінність від більшості криптовалют та багатьох мережевих або службових tokenів, які є взаємозамінними. Токен представляє собою сертифікат унікальності цифрового об'єкта, який підтверджує право на власність цифровим активом. Важливо розуміти, що NFT не забороняє копіювати цифровий актив, він тільки закріплює право на володіння оригіналом цього активу.

Вперше смарт-контракт NFT був створений в 2017 році на базі блокчейну Ethereum. На базі цієї технології почали створюватися стартапи у різних сферах діяльності: ігри, нерухомість, документи, мистецтво тощо [1, 2].

## Принцип роботи NFT

На дуже високому рівні більшість NFT є частиною блокчейну Ethereum. Ethereum — це криптовалюта, така як Bitcoin, але його блокчейн також підтримує смарт-контракти NFT. Ethereum дає змогу зберігати додаткову інформацію, так звані метадані. Це головна відмінність NFT від самої монети Ethereum. Варто зазначити, що інші блокчейни також можуть реалізувати власні версії NFT [2]. В метаданих може зберігатися різна інформація, така як опис, назва, посилання на будь-які ресурси, ціну, власника, документи тощо.

Табл. 1

РОЗРАХУНКИ ВАРТОСТІ ПАМ'ЯТІ У БЛОКЧЕЙНІ ETHEREUM

Назва	Кількість байт	Кількість необхідного Gas	Ціна, \$
1 Б	1	625	6250
1 кБ	1024	640000	6400000
1 МБ	1048576	655360000	6553600000
1 ГБ	1073741824	6.71089E+11	6.71089E+12

## ОБґРУНТУВАННЯ ТИПІВ ЗБЕРІГАННЯ ДАНИХ

Існує два способи зберігання даних в блокчейні: в самому ланцюгу і поза ним. Як впливає з назви блокчейну, всі дані зберігаються всередині кожного блоку в ланцюжку. Таким чином, дані можуть бути відновлені та використані знову. Але це достатньо дорога операція, крім цього такі відновлення впливають на швидкість системи. Тож багато інформації зберігається поза ланцюгом, щоб зменшити навантаження та ціну блокчейну.

Саме сховище поза блокчейном зберігає не всі дані, а лише метадані. Однак у нього є недолік. Якщо трапиться якийсь системний збій, дані можуть не відновитися. Але це набагато економніше, що робить цей підхід більш доцільним [3].

Розглянемо блокчейн Ethereum, оскільки це один із найпростіших прикладів для розуміння проблем, пов'язаних із зберіганням даних.

Подібно до вузлів і майнерів, які підтримують блокчейн Bitcoin, Gas — це паливо, яке підтримує функціонування блокчейна Ethereum. Усі транзакції, такі як надсилання Ethereum з одного облікового запису на інший, створення смарт-контракту та інші дії, мають свою комісію. Ця комісія вимірюється в одиницях Gas. Таким чином, будь-який фрагмент програмованих обчислень має вартість. Кожна транзакція має певну кількість Gas, яка пов'язана з ним. Чим вище складність проблеми, тим більшу кількість Gas вона буде коштувати. Gas коштує 10 доларів, і якщо ви плануєте зберігати дані або, скажімо, 256-бітове слово, це буде коштувати вам 20 000 Gas. Таким чином, байт становить 625 Gas або 6250 доларів США за поточним курсом Gas. Розрахунки вартості пам'яті в блокчейні Ethereum наведено у табл. 1 [4].

Тож вартість пам'яті в мережі Ethereum вимірюється мільйонами доларів. Для порівняння, ціна ГБ пам'яті на хмарних сервісах, таких як AWS, становить 0,05 доларів США на місяць. Архітектура блокчейну має бути зовсім іншою, щоб можна було зберігати великі файли в системі і це було економічно доцільно. Тож більшість NFT рішень використовує IPFS - протокол та систему розповсюдження даних за змістом, яка побудована на концепції ідентифікування даних за хешем [5]. Це означає, що сам NFT не містить всі байти файлу, а лише зберігає посилання на елемент у сховищах (рис.1).



Рис. 1. Схема зберігання даних поза блокчейном

#### NFT у НЕРУХОМОСТІ ТА ПРОБЛЕМА ДОВІРИ ДО РЕЕСТРУ

Для будь-якого виду цінного майна (нерухомість, автомобілі, мистецтво) важливо мати точні записи, які ідентифікують поточного власника і надають підтвердження того, що він справді є власником. Ці записи можна використовувати для: захисту права власників (наприклад, у випадку крадіжки); вирішення суперечок; того, щоб переконатися, що право на оренду правильно передано орендарю після укладання угоди; запобігання шахрайству з продажу.

Коли передача майна забезпечується блокчейном, більше не потрібно покладатися на довірену сторону для їх перевірки. Однак для того щоб переконатися, що в системі тільки один такий об'єкт нерухомості, необхідно створити систему перевірки унікальності.

Для цього потрібно створити реєстр, який буде відповідати за зв'язок між об'єктами та відповідними NFT (нерухомістю в блокчейні). Припустимо, що для певного типу об'єктів, які нас цікавлять, ми можемо генерувати певні ідентифікатори, які однозначно вказують на об'єкт (наприклад, координати, вулична адреса тощо). Реєстр буде пов'язувати нерухомість в блокчейні та ідентифікатори.

Реєстр повинен бути надійною третьою стороною. Ми не можемо повністю відійти від цієї моделі, однак ми можемо спробувати мінімізувати залежність від довіри та нав'язати правила, які зроблять реєстр прозорим.

Зокрема, довіра викликає набагато менше занепокоєння, коли реєстр змушений працювати прозоро і криптографічні протоколи використовуються для автентифікації інформації, що надходить із реєстру.

Цього можна досягти, зробивши повний каталог реєстру відкритим для всіх. Єдина можлива проблема тут полягає в тому, що в реєстр можна додати нерухомість, яка не належить власнику. Власник може виявити це, коли спробує додати своє майно. [6] Рішенням цієї проблеми може стати додаткова система, яка буде перевіряти власність майна, за допомогою додакових звернень у державний реєстр власності майна, за допомогою модераторів, які зможуть перевірити документи на власність майном або їх комбінація, при чому система автоматичної перевірки повинна бути ізольована від зовнішнього впливу. Іншим рішенням може стати імплементація одного з алгоритмів довіри до користувача [7] – при взаємодії користувачів системи формується певна довіра між ними і можемо припустити, що при певній оцінці довіри до користувача, можна стверджувати, що це його власність.

#### ВИСНОВКИ

В статті розглянуто основні ідеї та проблеми NFT в сфері оренда нерухомості. NFT-система може зробити процес передачі майна в оренду прозорим та стійким до атак і шахрайства. Проте основною проблемою даного підходу є система перевірки власності майна, адже сам смарт-контракт не зможе перевіряти цю інформацію, через суттєві недоліки блокчейну (дорога вартість зберігання даних). Але якщо з'являється система перевірки та зберігання даних поза блокчейном, то це робить систему вразливою до атак.

#### ЛІТЕРАТУРА

- [1] [https://uk.wikipedia.org/wiki/Невзаємозамінний\\_токен](https://uk.wikipedia.org/wiki/Невзаємозамінний_токен) (дата звернення 13.11.2021).
- [2] <https://www.theverge.com/22310188/nft-explainer-what-is-blockchain-crypto-art-faq> (дата звернення 13.11.2021).
- [3] <https://www.upgrad.com/blog/how-to-use-blockchain-to-store-data/> (дата звернення 13.11.2021).
- [4] <https://www.linkedin.com/pulse/costs-storing-data-blockchain-rohan-pinto/> (дата звернення 13.11.2021).
- [5] <https://uk.wikipedia.org/wiki/IPFS> (дата звернення 13.11.2021).
- [6] Mizrah A. A blockchain-based property ownership recording system.
- [7] Ахрамович В. М. Моделі довіри та репутації користувачів в соціальних мережах // Сучасний захист інформації № 4(40), 2019.



**РОЗПОДІЛЕНІ ТА  
ПАРАЛЕЛЬНІ ОБЧИСЛЕННЯ**

**DISTRIBUTED AND  
PARALLEL COMPUTING**

# Використання динамічного кворуму у процесі вибору лідера алгоритму Raft для прийняття консенсусу у розподілених системах

Козлова Ольга  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна

**Анотація**—У роботі представлено модифікацію Raft алгоритму досягнення консенсусу у розподілених системах, а саме зміну протоколу голосування для вибору сервера-лідера, і показано, що кластер з чотирма вузлами, керований модифікованим алгоритмом, буде мати майже такі ж показники доступності, як звичайний Raft кластер з п'ятьма вузлами, і завжди сприймає непоправну відмову будь-якого вузла без втрати даних.

**Ключові слова**—розподілені системи, алгоритм, консенсус, кластер, вузли, розподілений лог.

## Вступ

При спробі досягнення горизонтальної масштабованості системи постає питання синхронізації даних всередині кластера. Тоді і з'являється проблема консенсусу – як саме можна забезпечити консистентність даних у вузлах. Для цього існують алгоритми прийняття консенсусу, що вирішують задачу отримання узгодженого значення групою учасників у ситуації, коли можливі відмови деяких вузлів, надання недостовірної інформації або спотворення переданих значень середовищем передачі даних.

## АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ

Алгоритми Raft [1] і Paxos [2] – це алгоритми прийняття консенсусу у розподілених системах. Обидва вони засновані на підході лідерства: спочатку обирається лідер, що є відповідальним за управління розподіленим логом. Лідер приймає запити від клієнтів і реплікує їх на інші сервери в кластері. Основна відмінність у роботі алгоритмів Raft і Paxos це процес вибору лідера. Алгоритм Paxos сам розподіляє терміни між серверами. Raft же дозволяє будь-якому серверу стати лідером у будь-який термін. Якщо фоловер не отримуватиме жодних повідомлень деякий час, він збільшує на 1 свій номер терміну  $T$ , переходить у стан «кандидат», голосує сам за себе і розсилає запит на голосування всім іншим серверам. Коли лідер обраний, він бере на себе управління розподіленим логом. Лідер записує в свій лог новий запис, що містить клієнтську команду, а потім надсилає запит на запис всім

фоловерам, для того щоб вони реплікували запис у свій лог. Коли запис буде успішно репліковано на більшості серверів, лідер вважає запис остаточно збереженим і надсилає клієнту інформацію про успішно виконану операцію [3].

## ВВЕДЕННЯ ДИНАМІЧНОГО ПРОТОКОЛУ ГОЛОСУВАННЯ

Основною причиною високої ресурсоемності алгоритму Raft є використання консенсусного голосування більшості для визначення кворуму  $Q$  запису і кворуму  $Q$  виборів. Оскільки ці кворуми повинні містити більшість серверів кластера, кластер Raft повинен містити щонайменше  $2n + 1$  сервери, щоб мати можливість приймати  $n$  збоїв серверів. Ефективнішим рішенням є зробити кворуми динамічними та вимагати, щоб вони містили більшість активних серверів даного кластера. Таким чином протокол динамічного голосування вимагає  $n + 2$  сервери на кластер, щоб мати змогу приймати  $n$  збоїв. Реалізація цієї властивості у алгоритмі Raft, призводить до 25% зменшення ресурсоемності Raft. Для збереження додаткових метаданих про стан вузлів пропонується використовувати когортні набори [4].

## ПОРІВНЯННЯ РЕАЛІЗАЦІЙ

Розглянемо кластер, що складається із чотирьох вузлів, який вимагає як два сервера, щоб підтримувати нові оновлення даних. У якості показника ефективності було обрано доступність сервера, тобто відрізок часу, коли він буде спроможний обробляти запити клієнта. Для того, аби вивести значення доступності, необхідно зробити деякі припущення щодо роботи кластера. Нехай кожен вузол працює незалежно один від одного і так само незалежно може відмовити. Кожного разу, коли вузол падає, система автоматично запускає процес його відновлення. Ми припускаємо, що падіння серверів це незалежні випадкові події розподілені за експоненціальним законом з середнім  $\lambda$ . За аналогією, час відновлення роботи вузла має експоненційний розподіл з середнім

$\mu$ . Обидві гіпотези необхідні для того, аби пред- ставити систему у вигляді процесу Маркова з кінцевою кількістю можливих станів. Ми також припускаємо, що лідери можуть швидко визна- чати відмову вузлів з допомогою періодичних фіктивних запитів. На рис.1 зображена схема переходів між станами кластера.

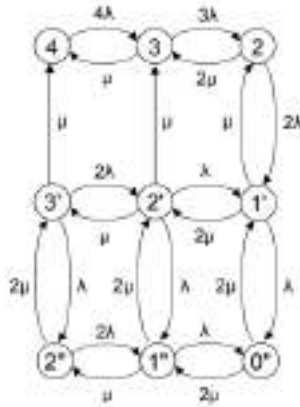


Рис. 1. Схема зміни станів кластера

Кластер має 9 можливих станів. Стан 4 - це початковий стан системи, коли усі вузли працюють коректно. Відмова будь-якого із вузлів переводить систему у стан 3 і кворум складатиметься з двох вузлів із трьох можливих. Падіння ще одного вузла переведе систему у стан 2, кворум складатиметься з двох вузлів із двох можливих.

Умови рівноваги для такого кластеру:

$$\begin{aligned} 4\lambda p_4 &= \mu(p_3 + p_{3'}) \\ (3\lambda + \mu)p_3 &= 4\lambda p_4 + 2\mu p_2 + \mu p_{2'} \\ (3\lambda + \mu)p_{3'} &= \mu p_{2'} + 2\mu p_{2''} \\ (2\lambda + 2\mu)p_2 &= 3\lambda p_3 + \mu p_{1'} \\ (2\lambda + 2\mu)p_{2'} &= 2\lambda p_{3'} + 2\mu(p_{1'} + p_{1''}) \\ (2\lambda + 2\mu)p_{2''} &= \lambda p_{3'} + \mu p_{1''} \\ (\lambda + 3\mu)p_{1'} &= 2\lambda p_2 + \lambda p_{2'} + 2\mu p_{0''} \\ (\lambda + 3\mu)p_{1''} &= \lambda p_{2'} + 2\lambda p_{2''} + 2\mu p_{0''} \\ 4\mu p_{0''} &= \lambda(p_{1'} + p_{1''}), \end{aligned}$$

де  $p_i$  це ймовірність перебування кластера у стані  $i$  з додатковою умовою, що  $\sum_i p_i = 1$ .

Розв'яжемо систему лінійних рівнянь і замі- нимо відношення  $\frac{\lambda}{\mu} = \rho$ . Отримаємо наступну формулу доступності кластера:

$$A_{4R}(\rho) = p_4 + p_3 + p_2 = \frac{3\rho^6 + 23\rho^5 + 68\rho^4 + 97\rho^3 + 21\rho + 3}{(\rho + 1)^5(3\rho^3 + 8\rho^2 + 6\rho + 3)}$$

Такі ж формули можна вивести і для систем, що складаються з трьох або з п'яти вузлів. Та- кі системи вимагають більшості від початкової кількості вузлів для коректної роботи кластеру. Виходячи з цього, доступність такого кластера вираховується за формулою

$$\begin{aligned} A &= \frac{1}{\rho + 1}, A_3(\rho) = A^3 + 3A^2(1 - A) = \frac{3\rho + 1}{(\rho + 1)^3}, \\ A_5(\rho) &= A^5 + 5A^4(1 - A) + 10A^3(1 - A)^2 = \\ &= \frac{10\rho^2 + 5\rho + 1}{(\rho + 1)^5}. \end{aligned}$$

Побудуємо графіки на основі отриманих фор- мул. На рис.2 зображено графік доступності кластера при трьох конфігураціях: динамічне голосування, класична конфігурація алгоритму Raft з трьома та п'ятьма вузлами. Доступність кластера тут залежить від відношення частоти відмов вузлів до частоти їх відновлення - , що належить проміжку від 0 до 0,20. Нульова відмі- тка означає те, що кластер ніколи не відмовить, а відмітка 0,20 - те, що кластер буде відповідати на запити клієнта 83,3 відсотки часу.

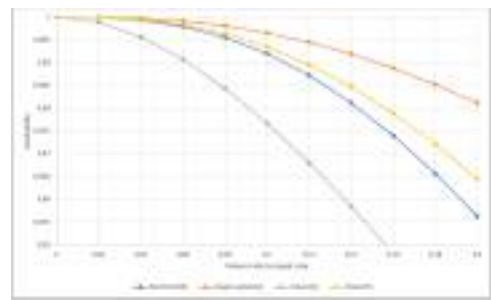


Рис. 2. Графік порівняння доступності кластера

Як бачимо, динамічний протокол показує пра- ктично такі ж показники доступності як і кон- венційний Raft на 5-ти вузлах і значно кращі показники, ніж Raft на 3-ох вузлах.

### ВИСНОВКИ

Хоча алгоритм консенсусу Raft і легкий для розуміння, і простіший у реалізації, ніж алго- ритм Paxos, він вимагає п'яти серверів, щоб за- безпечити як доступність, так і довговічність оновлення логу. В роботі показано як можна зменшити цю енергоємність, замінивши статичні кворуми, які зараз використовує Raft, кворума- ми, які змінюються залежно від кількості досту- пних на даний момент серверів, а також, що ця конфігурація буде майже такою ж доступною, як і звичайний кластер Raft з п'ятьма учасниками.

### ЛІТЕРАТУРА

- [1] Ongaro D. In Search of an Understandable Consensus Algorithm / D. Ongaro, J. Ousterhoud. // USENIX Annual Technical Conference. – 2014.
- [2] Lamport, L., et al.: Paxos made simple. ACM SIGACT News 32(4), 18–25 (2001).
- [3] Howard H. Analysis of Raft Consensus / Heidi Howard. // Cambridge Computer Laboratory. – 2014. – №857.
- [4] Jehan-François P. Pirogue, a lighter dynamic version of the Raft distributed consensus algorithm / P. Jehan-François, L. Darrell D. E. // IEEE. – 2015.

Рецензент: доцент кафедри інформаційних систем та технологій КПІ ім. Ігоря Сікорського к.т.н. доцент Сперкач Майя



**ТЕХНОЛОГІЇ  
ПРОГРАМУВАННЯ**

**PROGRAMMING  
TECHNOLOGIES**



# Персоналізація страхових полісів: розробка шаблонів

Гатіна Тетяна  
КПІ ім. Ігоря Сікорського  
м. Київ, Україна

**Анотація**—Проаналізовано потреби та актуальний стан програмного забезпечення страхової галузі, виявлено недоліки існуючих пропозицій. Змодельоване архітектурне рішення для універсалізації та оптимізації розробки і супроводження персоналізованих страхових полісів.

**Ключові слова**—страхова галузь, SOA архітектура, BPM, BPMS, Camunda, страховий поліс.

## ВСТУП

Розвиток страхової діяльності в Україні тісно пов'язаний з розробкою та впровадженням діджитал-технологій у діяльність страхових компаній. Налаштувати продуктивну B2B та B2C комунікацію між страховиками, посередниками та страхувальниками дозволяє використання сучасних інформаційних систем. Страхова галузь є одним з найбільш інформаційно залежних видів економічної діяльності. Страховий ринок України продовжує розвиватися та наближається до світових стандартів діяльності, що викликає зростання попиту на об'єктивну оцінку фінансового стану підприємств та змушує страхові компанії використовувати автоматизовані інформаційні системи та технології. Зростання клієнтської бази, посилення конкуренції, необхідність впровадження нових сервісів стимулює страхові компанії до впровадження інформаційних систем та сучасних інноваційних технологій.

Страхові компанії використовують діджитал-технології як засіб комунікації з клієнтом для нагадування про пролонгацію контракту, надання інформації про статус збитку, терміни та обсяг страхових виплат, пропозиції продукту. Однак страхові компанії все ще обмежено використовують цифрові технології для продажу та просування страхових послуг на ринку [1]. Це пов'язано, з одного боку, з необхідністю підтримувати паперовий документообіг з клієнтами та страховими посередниками, з іншого боку - пояснюється структурою каналів збуту, де частка прямих продажів залишається на низькому рівні. Використання веб-додатків дає можливість досить швидко створювати та впроваджувати інтегровані системи страхової діяльності [2]. Швидка зміна умов, висока конкуренція на ринку, персоналізований підхід до взаємодії з клієнтами та при цьому необхідність врахування змін у законодавстві, документообігу та прозорості процесів обумовлює необхідність розробки гнучкої

Табл. 1

Елементи діджитал-технологій	Страхові компанії «non-life»	Страхові компанії «life»
Страховий калькулятор	23	-
Онлайн-магазин, у т.ч.:	19	3
з одним страховим продуктом	6	-
з двома і більше страховими продуктами	13	-
Онлайн-оплата	10	-
Онлайн-кабінет	9	3
Зв'язок із соціальними мережами, месенджерами та відеоплатформами	29	3

та безпечної системи програмного забезпечення страхової компанії.

У табл. 1 наведено елементи діджитал-технологій, які використовуються страховими компаніями на веб сайтах [3].

Аналіз наведених в таблиці даних свідчить, що частина страхових компаній мають онлайн ринок збуту та дають можливість клієнтам обрати та придбати страховий поліс онлайн. З метою забезпечення конкурентних переваг страховим компаніям потрібна можливість створення персоналізованих страхових полісів для залучення клієнтів та якісне подальше супроводження клієнтів для збільшення їх лояльності до компанії та збереження залучених клієнтів.

У результаті аналізу прийнято наступні вимоги до проєктованого рішення: можливість розробки персоналізованих страхових продуктів; забезпечення зручності супроводу страхового полісу; можливість інтеграції з різними системами. При прийнятті архітектурного рішення потрібно враховувати наступні фактори: безпека збереження інформації клієнтів; гнучкість системи; можливість впливати на процес. Відповідно до сформованих вимог та потреб страхових компаній доцільно використовувати веб-додатки на базі архітектури SOA [4].

## ТЕХНІЧНЕ РІШЕННЯ

Сервісно-орієнтована архітектура (SOA) – це стиль архітектури програмного забезпечення, який складається з дискретних та слабо пов'язаних програмних агентів, які виконують потрібну функцію [5]. Концепція SOA полягає в

наступному: програма може бути спроектована і побудована таким чином, щоб її модулі були легко інтегрованими і могли бути легко використаними повторно [6]. Інтерфейси компонентів у сервіс-орієнтованій архітектурі інкапсулюють деталі реалізації від інших компонентів, таким чином дозволяють комбінування та багаторазове використання компонентів для побудови складних розподілених програмних комплексів. (рис.1).



Рис. 1. Структура SOA

Ключовим об'єктом страхової сфери є страховий поліс, а саме процес видачі та супроводження страхового полісу. За цей час потрібно зберігати кожен етап процесу та впливати на нього. Також ці бізнес-процеси мають бути різноманітними та гнучкими, тому прийнято рішення про доцільність використання BPM (Business Process Management) [7]. BPM розглядає бізнес-процеси як особливі ресурси підприємства, що безперервно адаптуються до змін, і дотримуються принципів зрозумілості та видимості бізнес-процесів за рахунок надання суб'єктам управління можливості їх моделювання, моніторингу та аналізу, динамічної перебудови моделей. Для технологічної підтримки концепції BPM використовується спеціальний клас програм – BPMS (Business Process Management System) [8]. Функції BPMS можуть бути поділені на чотири групи відповідно до стадій життєвого циклу процесу PDCA (Цикл Демінга): планування (моделювання), виконання, контроль і моніторинг, поліпшення процесу. Під час розробки технічного рішення було розглянуто найпопулярніші BPMS [9] (рис.2).



Рис. 2. Порівняння BPMS

В результаті розгляду BPM систем за кри-

теріями зручності підключення API, огляду та аналізу процесу Camunda Platform має перевагу. Також вона є безкоштовною і має наступні переваги: вилучення журналів подій та моделей процесів, висока пропускна здатність, можливість інтеграції і зручний веб-інтерфейс [10].

#### ВЛАСНА РЕАЛІЗАЦІЯ

Для вирішення поставленої задачі було розроблено архітектурне рішення, складовими частинами якого є клієнт-серверний підхід та сукупність мікросервісів, що складає SOA архітектурний стиль (рис.3).

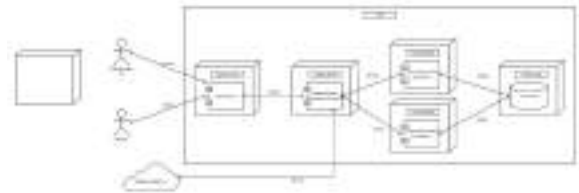


Рис. 3. Діаграма розгортання

В межах розробленої системи реалізовано за допомогою платформи Camunda, Process Engine: головний процес, який забезпечує виконання загальної логіки для всіх процесів страхових полісів та процеси персоналізованих страхових полісів. Це дозволить досягти поставленої мети та реалізувати шаблони персоналізованих страхових полісів з можливістю автоматизації їх супроводження та централізованого збереження інформації протягом дії страхового полісу.

#### ЛІТЕРАТУРА

- [1] Галушак В.В. Інформаційні технології страхування – рушійна сила страховика у взаємодії із клієнтами // Регулювання банківської діяльності, бізнесу та підприємництва: національна та міжнародна практика. – м. Одеса: 2018. С. 87-91.
- [2] Rubtsova N. M. Особливості функціонування страхового ринку України в умовах діджиталізації // Fundamental and applied research in the modern world – Boston, USA, 2021. Pp. 21-27.
- [3] Шубенко І. А. Тенденції діджиталізації на страховому ринку України / І.А. Шубенко // Бізнес інформ. - 2020. - №2. - С. 273-279.
- [4] <https://rubygarage.org/blog/monolith-soa-microservices-serverless>(дата звернення 12.11.2021).
- [5] <https://www.ibm.com/cloud/blog/soa-vs-microservices>(дата звернення 12.11.2021).
- [6] <https://www.chakray.com/soa-the-perfect-architecture-to-adapt-to-business-needs/>(дата звернення 12.11.2021).
- [7] [https://club.cnews.ru/blogs/entry/import\\_itsistemy\\_dlya\\_funktionalnogo\\_i\\_protsechnogo\\_upravleniya\\_bfba](https://club.cnews.ru/blogs/entry/import_itsistemy_dlya_funktionalnogo_i_protsechnogo_upravleniya_bfba)(дата звернення 12.11.2021).
- [8] <https://globalcio.ru/discussion/895/>(дата звернення 12.11.2021).
- [9] <https://www.g2.com/compare/camunda-platform-vs-epicor-kinetic-vs-kissflow-vs-pega-platform>(дата звернення 12.11.2021).
- [10] Alessandro Berti An Open-Source Integration of Process Mining Features into the Camunda Workflow Engine // Software Engineering (cs.SE) - Aachen, Germany, 2020.

Рецензент: доцент кафедри інформаційних систем та технологій КПП ім. Горя Сікорського к.т.н. Штифурак Юрій



# **ABSTRACTS**



---

# INFORMATION SYSTEMS AND TECHNOLOGIES

Page 12

## *Exploration of effectiveness of a bipedal walking robot`s model*

Nataliya Repnikova, Vladyslav Sverchkov  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine  
n.repnikova@gmail.com, vladenigmatic@gmail.com

*Abstract.* In this publication the Cost of Transport (CoT) parameter was used to find out effectiveness of the robot`s model called Darth EvadeR, which is able to avoid obstacles. Conducted researches allowed to determine the optimal value of CoT and to make comparisons with efficiency of already existing walking robots. An algorithm for calculating total power consumed by the robot model using the values of torque and angular velocity of robot`s drive units has been developed.

*Keywords:* two-legged robot, bipedal walking robot, CoT, efficiency of robot, torque, angular velocity, Darth EvadeR.

---

Page 14

## *Magnetic micro robot control system for drug delivery inside the human body*

Laushkin V., Pysarenko A.  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

*Abstract.* The purpose of this article is to describe the principle of operation and structure of the control system of the magnetic microrobot. The main task of the system is to deliver drugs inside human body without invasive intervention. The principle of operation is based on the manipulation small magnetic microparticles by external magnetic field, these particles are interconnected and form the body of the robot. This system provides simpler and safer treatments for some diseases.

*Keywords:* microbots, magnetic interaction, drug delivery.

---

Page 16

## *Control system of experimental setup for pulsed high-voltage discharges research*

Adamenko D., Pysarenko A.

Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

*Abstract.* The purpose of this work is to describe a system that should make possible experiments that will investigate the effect of certain variable parameters on pulsed high-voltage discharges. The developed prototype is designed to change the parameters of experiments, automatic collection and analysis of the obtained data.

*Keywords:* *thermonuclear fusion, diode, gap, lasers, measured energy.*

---

Page 18

## *Machine-learning Based Unit Test Generation*

Mariia Tisna, Maksym Bukasov

Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

*Abstract.* In this paper, an analysis of existing methods for automated unit-test generation is presented and the most fitting and promising one is determined and chosen. An analysis of existing solutions is performed and their main downsides are detected. Architecture for machine-learning based unit test generation system is developed as well as the linking algorithm between unit-tests and source code under the test.

*Keywords:* *unit-test, machine learning, neural networks, generator, model training.*

---

Page 20

## *Use of high performance microcontrollers for an automated area exploration platform*

Bessmertnyi Roman  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine  
roman.bezsmertniy@gmail.com

*Abstract.* This paper presents the results of the development of the software infrastructure of an automated area exploration platform using the capabilities of the Cortex-A architecture. Work was carried out to develop of a miniature inexpensive system for collecting and processing video information. This information is the basis of the basic element of the localization system, which is still carried out by means of satellite navigation, which is vulnerable to loss of communication. As a result of the development of a prototype of the automated control system of a ground mobile platform based on the Spring framework with the built-in localization system based on video information, it is possible to determine the standards of such a system and ensure high reliability of such systems.

*Keywords:* ACS, mobile platform, Spring, odometry, video data processing.

---

Page 22

## *Music generation system using recurrent neural networks*

Komarskyi Oleksandr, Anatoliy Doroshenko  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine  
komarskiy33@gmail.com, doroshenkoanatoliy2@gmail.com

*Abstract.* This paper considers the possibility of generating musical compositions using recurrent neural networks. Two approaches to the generation of musical compositions are proposed and considered, namely using the method of notes and the method of chords. The research of both methods is carried out, and their advantages and disadvantages are formulated. The process of searching and processing data for training a musical neural network is described in detail. The proposed solution allows you to easily generate music without human intervention.

*Keywords:* music generation, machine learning, recurrent neural networks, LSTM, RNN.

---



Page 24

## *Class Generation Algorithm for Universities*

Vladyslav Lopata, Anatoliy Doroshenko  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

lopatavlad@gmail.com, doroshenkoanatoliy2@gmail.com

*Abstract.* The following article covers the class generation algorithm for universities. The material in the article considers the need to regenerate schedules based on the assessment of weights and offers an algorithm for taboo local search for alternative schedules to build new schedules based on the original one in order to improve the assessment of schedule quality. The concept of schedule quality and the method of assessing the quality of the schedule based on explicit weights are introduced.

*Keywords:* scheduling algorithm, taboo local search, NP-complete task, individual schedule, optimization of the generated schedule, quality of the generated schedule, freedom of placement of classes in the schedule.

---

Page 26

## *Automatic system of continuous testing of the web application*

Shybetskyi Vladyslav, Anatoliy Doroshenko  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

vladyslav.shyb@gmail.com, doroshenkoanatoliy2@gmail.com

*Abstract.* This publication covers the problem of testing automation of different levels of a software product using a mobile application. The main problem of existing solutions was identified and based on those solutions the new approach and method were introduced. Technical solution was created with an aim of systematisation of a whole testing process. Test model that was created and tested proved the efficiency of the new approach and method.

*Keywords:* automatic system of continuous testing, CI, Jenkins, automation, approach, method.

---

---

Page 28

## *Analysis of scientists' scientometric data based on the co-authority graph*

Vladyslav Mamontov, Hennadii Berezinskyi  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

vladyslavmamontov@gmail.com, gennady.berezinsky@gmail.com

*Abstract.* The objective of the work is to study the methods of data analysis of scientific activities of groups of scientists united in scientific units, using a graph data structure. Another task is to visualize the obtained data in order to simplify management decisions by the administration of science and education institutions. The work is based on basic information about scientists working within the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". Examples of conclusions based on visualized data are given. The result of this work is a demonstration of a practical way to use information about the activities of scientists.

*Keywords:* *scientifics, google scholar, graph, visualization, centrality indicator, betweenness centrality, hirsch number.*

---

Page 30

## *Financial Series Forecasting in the Stock Market using Wavelet Transform*

Fomin Ilya  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

*Abstract.* This paper is about developing a neural network architecture to determine a trading strategy and track the movement of price trends. It is also necessary to check the influence of the number of input variables on the accuracy of the model of deep learning in the problem of forecasting the state of the stock market. The input data set was pre-processed using both standard pre-processing methods and a time series wavelength conversion method to get rid of noise in this set. Further training and testing of the model architecture was performed, using the type of recurrent neural networks LSTM to predict the future state of the stock market. A comparison is made with the basic model of machine learning, which has no wavelet transforms of data. Thorough empirical research based on the NASDAQ100 exchange price data set has demonstrated the accuracy of the developed LSTM model, which surpasses the methods of forecasting financial time series trends without the use of the wavelet transform method.

*Keywords:* *financial series forecasting, short-term forecasting, recurrent neural network, wavelet transform.*

---

## Page 32

## *BigData Storages*

Tiurin Valerii, Doroshenko Anatoliy  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

tiurinvalery@gmail.com, a-y-doroshenko@ukr.net

*Abstract.* The goal of the work is an approach how to store and proceed for a future analysis a huge amount of data. The prototype of this system has been created in scope of this work. System prototype will be used to pros and cons of a system design and implementation. Design of these functionality include usage of a different kinds of data storages, data streaming, delta-processing for non-ordering data, micro-batch upload and data-masking for PII and PHI data.

*Keywords:* Data Warehouse (DWH), BigData, BigQuery, ETL.

---

## Page 34

## *A criterion for estimating the cost of ownership of a mobile transport robot*

Olha Kryvych, Serhii Reznikov  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

olgakrivich@gmail.com, reznikov\_sa@ukr.net

*Abstract.* This scientific research provides an overview of the subsystem for formation of routes of transport mobile robot for industrial premises with the technological positions for performance of typical operations which is based on algorithm A\*.The criterion for estimating the cost of ownership of the robot on the basis of one of the criteria of optimality of routes, named the length of the distance traveled, is proposed. A comparison of the values of the proposed criterion for A \* and other algorithms is presented.

*Keywords:* TCO, evaluation criterion, mobile robot, path search, cost of ownership.

---

---

## CONTROL THEORY

Page 38

### *Synthesis of multidimensional control systems with non-square matrices*

Repnikova Natalya, Vladyslav Hnyp  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

n.repnikova@gmail.com, vladyslavgnyp@gmail.com

*Abstract.* The paper proposes expansion of possibilities of synthesis of structures of linear multidimensional systems with non-square matrices on the basis of R.Iserman's formula. Considered digital control systems described by state-space models. It is shown that fulfillment of condition of correctness of choice of dimensions of the matrices of state and output equations allows to obtain results of synthesis with preset quality indicators for the studied class of systems. Simulation of digital system with non-square matrices was performed using MATLAB/Simulink application package.

*Keywords:* digital systems, multidimensional systems, quality improvement, digital state controller, space-state model, non-square matrices, MATLAB models.

---

Page 40

### *Research of the digital approximation method for selecting the quantization period over time*

Repnikova Natalya, Vysotska Yuliia  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

n.repnikova@gmail.com, julja.vysotska@gmail.com

*Abstract.* This publication discusses the synthesis of digital control systems based on the mathematical platform of transfer functions and structural diagrams. It is proposed to use the digital approximation method to clarify the value of the quantization period over time to ensure the functioning of the digital system in a stable region. It is shown that the fulfillment of the condition for the correctness of the choice of the quantization period allows for the studied Class of systems to obtain synthesis results with specified quality indicators. A digital system is modelled using the MATLAB/Simulink application package.

*Keywords:* digital systems, transfer functions, z-transformations, time quantization period, digital approximation, quality improvement, MATLAB models.

---

# ARTIFICIAL INTELLIGENCE

Page 44

## *Distributed neural networks learning system*

Oleksii Nikulin  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine  
firlbit@gmail.com

*Abstract.* This paper provides an overview of the distributed system, its main characteristics and tasks that need to be solved in the implementation of a distributed learning system of neural networks. In particular, the processes of obtaining, transforming, downloading data and directly distributed learning of neural networks are considered. Possible ways to implement these tasks are also briefly described.

*Keywords:* distributed system, distributed computing, distributed machine learning, ETL.

---

# SECURITY AND PROTECTION OF INFORMATION

Page 48

## *NFT smart contracts for real estate rental*

Andrii Khodarchenko, Vadym Poltorak  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine

*Abstract.* This article discusses NFT for real estate rental, namely the main problems of data storage in the blockchain and possible solutions for this problem, NFT smart contract system for real estate, the main problems of such a system and options for solving these problems. The inexpediency of storing large amounts of data in a blockchain and the possibility of storing this data outside the system are considered. There is also an example of a system for renting real estate, the problem of the register located outside the blockchain and possible solutions to this problem by establishing trust for the register.

*Keywords:* blockchain, NFT, non-fungible tokens, database, smart contract, cryptocurrency, real estate rental.

---

# DISTRIBUTED AND PARALLEL COMPUTING

Page 52

## *Using a dynamic quorum in the process of selecting the leader of the Raft algorithm to reach consensus in distributed systems*

Olha Kozlova

Igor Sikorsky Kyiv Polytechnic Institute

Kyiv, Ukraine

*Abstract.* This paper presents a modification of the Raft algorithm for reaching consensus in distributed systems, namely changing the voting protocol for selecting a leader server, and shows that a four-node cluster controlled by a modified algorithm will have almost the same availability as a conventional Raft cluster with 5 nodes, and always perceives the irreparable failure of any node without data loss.

*Keywords:* distributed systems, algorithm, consensus, cluster, nodes, distributed log.

---

# PROGRAMMING TECHNOLOGIES

Page 56

## *Personalization of insurance policies: development of templates*

Hatina Tetiana  
Igor Sikorsky Kyiv Polytechnic Institute  
Kyiv, Ukraine  
t.gatina2@gmail.com

*Abstract.* The needs and current state of the software of the insurance industry are analyzed, the shortcomings of the existing proposals are revealed. An architectural solution for universalization and optimization of development and maintenance of personalized insurance policies is modeled.

*Keywords:* insurance industry, SOA architecture, BPM, BPMS, Camunda, insurance policy.

---