

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна

ВІСНИК

Харківського національного університету
імені В.Н. Каразіна

Серія

«Математичне моделювання.
Інформаційні технології.

Автоматизовані системи управління»

Випуск 53

Серія заснована 2003 р.

BULLETIN

of V.N. Karazin Kharkiv National University

Series

«Mathematical Modeling.
Information Technology.
Automated Control Systems»

Issue 53

First published in 2003

Харків
2022

Статті містять дослідження у галузі математичного моделювання та обчислювальних методів, інформаційних технологій, захисту інформації. Висвітлюються нові математичні методи дослідження та керування фізичними, технічними та інформаційними процесами, дослідження з програмування та комп'ютерного моделювання в наукоємних технологіях.

Для викладачів, наукових працівників, аспірантів, працюючих у відповідних або суміжних напрямках.

Наказом Міністерства освіти і науки України від 17.03.2020 № 409 наукове фахове періодичне видання Вісник Харківського національного університету імені В.Н. Каразіна серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління» включено до Категорії «Б» Переліку наукових фахових видань України за наступними спеціальностями: 113 – Прикладна математика; 122 – Комп'ютерні науки та інформаційні технології; 123 – Комп'ютерна інженерія; 125 – Кібербезпека.

Затверджено до друку рішенням Вченої ради Харківського національного університету імені В. Н. Каразіна (протокол № 7 від 11.04.2022 р.)

Редакційна колегія:

Азаренков М.О. (гол. редактор),

д.ф.-м.н., академік НАН України, проф., ІВТ ХНУ імені В.Н. Каразіна

Жолткевич Г.М. (заст. гол. редактора), д.т.н., проф., ФМІ ХНУ імені В.Н. Каразіна

Лазурик В.Т. (заст. гол. редактора), д.ф.-м.н., проф., ФКН ІВТ ХНУ імені В.Н. Каразіна

Споров О.Є. (відповідальний секретар), к.ф.-м.н., доц. ФКН ІВТ ХНУ імені В.Н. Каразіна

Замула О. А., д.т.н., доц., ФКН ІВТ ХНУ імені В.Н. Каразіна

Золотарьов В.О., д.ф.-м.н., проф., ФТІНТ імені Б.І. Веркіна НАН України

Куклін В.М., д.ф.-м.н., проф., ФКН ІВТ ХНУ імені В.Н. Каразіна

Мацевитий Ю.М., д.т.н., академік НАН України, проф., фізико-енергетичний ф-т ХНУ імені В.Н. Каразіна

Рассомахін С. Г., д.т.н., доц., ФКН ІВТ ХНУ імені В.Н. Каразіна

Руткас А.Г., д.ф.-м.н., проф., ФМІ ХНУ імені В. Н. Каразіна

Стервєдєв М.Г., к.т.н., доц., ФКН ІВТ ХНУ імені В.Н. Каразіна

Толстолузька О. Г. д.т.н., с.н.с., доц., ФКН ІВТ ХНУ імені В.Н. Каразіна

Угрюмов М. Л., д.т.н., проф., ІВТ ХНУ імені В.Н. Каразіна

Шейко Т.І., д.т.н., проф., фізико-енергетичний ф-т ХНУ імені В.Н. Каразіна

Шматков С. І., д.т.н., проф., ФКН ІВТ ХНУ імені В.Н. Каразіна

Щербина В.А., д.ф.-м.н., проф., ФМІ ХНУ імені В.Н. Каразіна

Раскін Л.Г., д.т.н., проф., Національний технічний університет "ХПІ"

Стрельнікова О.О., д.т.н., проф. Ін-т проблем машинобудування НАН України

Соколов О.Ю., д.т.н., проф., кафедра прикладної інформатики, університет імені Миколая Коперника, м. Торунь (Польща)

Prof. **Harald Richter**, Dr.-Ing., Dr. rer. nat. habil. Professor of Technical Informatics and Computer Systems, Institute of Informatics, Technical University of Clausthal, Germany

Prof. **Philippe Lahire**, Dr. habil., Professor of computer science, Dep. of C. S., University of Nice-Sophia Antipolis, France

Адреса редакційної колегії: 61022, м. Харків, майдан Свободи, 6, ХНУ імені В. Н. Каразіна, к. 534.

Тел. +380 (57) 705-42-81, Email: journal-mia@karazin.ua.

Мова публікації: українська, англійська, російська.

Статті пройшли внутрішнє та зовнішнє рецензування.

Свідоцтво про державну реєстрацію КВ № 21578-11478 Р від 18.08.2015.

The articles are present research in the field of mathematical modeling and computing methods, information technologies, information security. New mathematical methods of research and management of physical, technical and information processes, research on programming and computer modeling in science-intensive technologies are covered.

For teachers, researchers, graduate students working in relevant or related fields.

By the order of the Ministry of Education and Science of Ukraine from 17.03.2020 № 409 scientific professional periodical Bulletin of V.N. Karazin Kharkiv National University series "Mathematical modeling. Information Technologies. Automated control systems" is included in Category "B" of the List of scientific professional publications of Ukraine in the following specialties: 113 – Applied Mathematics, 122 – Computer Science and Information Technology; 123 – Computer engineering; 125 – Cybersecurity.

Approved for publication by the decision of the Academic Council of V.N. Karazin Kharkiv National University (Minutes № 7 of 11.04.2022).

Editorial Board:

Azarenkov M.O. (Chief Editor), Acad. Of the NAS of Ukraine, Dr. Sc., Prof., HTI V.N. Karazin Kharkiv National University

Zholtkevich G.M. (Deputy Editor), Dr. Sc, Prof. MCS V.N. Karazin Kharkiv National University

Lazurik V.T. (Deputy Editor), Dr. Sc, Prof. CSD HTI V.N. Karazin Kharkiv National University

Sporov O.E., (Executive Secretary), Ph.D. Assoc. Prof, CSD HTI V.N. Karazin Kharkiv National University

Zamula A.A., Ph.D. Assoc. Prof, CSD HTI V.N. Karazin Kharkiv National University

Zolotarev V.A., Dr. Sc, Prof. B. Verkin Institute for Low Temperature Physics and Engineering of the National Academy of Sciences of Ukraine

Kuklin V.M., Dr. Sc, Prof. CSD HTI V.N. Karazin Kharkiv National University

Matsevity Yu.M., Acad. Of the NAS of Ukraine, Dr. Sc., Prof., DPE V.N. Karazin Kharkiv National University

Rossomakhin S.G., Dr. Sc, Prof. CSD HTI V.N. Karazin Kharkiv National University

Rutkas A.G., Dr. Sc, Prof. MCS V.N. Karazin Kharkiv National University

Styervoyedov N.G., Ph.D. Assoc. Prof, CSD HTI V.N. Karazin Kharkiv National University

Tolstoluzka O.G., Dr. Sc, Assoc. Prof. CSD HTI V.N. Karazin Kharkiv National University

Ugryumov M.L., Dr. Sc, Prof. HTI V.N. Karazin Kharkiv National University

Sheyko T.I., Dr. Sc, Prof. DPE V.N. Karazin Kharkiv National University

Shmatkov S.I., Dr. Sc, Prof. CSD HTI V.N. Karazin Kharkiv National University

Shcherbina V.A., Dr. Sc, Prof. MCS V.N. Karazin Kharkiv National University

Raskin L.G., Dr. Sc, Prof. National Technical University "Kharkiv Polytechnic institute"

Strelnikova E.A., Dr. Sc, Prof., NASU A. Pidgorny Institute of Engineering Problems

Sokolov O.Yu., Dr. Sc, Prof. Nicolaus Copernicus University, Torun, Poland

Prof. **Harald Richter**, Dr.-Ing., Dr. rer. nat. habil. Professor of Technical Informatics and Computer Systems, Institute of Informatics, Technical University of Clausthal, Germany

Prof. **Philippe Lahire**, Dr. habil., Professor of computer science, Dep. of C. S., University of Nice-Sophia Antipolis, France

Editorial Address: 61022, Kharkiv, Svobodi sq., 6, V.N. Karazin Kharkiv National University, r. 534.

Phone. +380 (57) 705-42-81, Email: journal-mia@karazin.ua.

Language of publication: Ukrainian, English, Russian.

The articles pass internal and external review.

Certificate of state registration: KV № 21578-11478P dated 18.08.2015

ЗМІСТ

- **Вервейко К.Є., Друшляк В. В., Кізілова Н. М., Корякіна О. О. 6**
Фрактальні структури і механічні властивості раковин амонітів

- **Гамзаєв Р. А., Мурадова В. Х., Ткачук М. В. 12**
Дослідження альтернативних технологій контейнерів для віртуалізації процесів розміщення компонентів лінійок програмних продуктів

- **Лихач О. Ю., Угрюмов М. Л., Шевченко Д. О., Шматков С. І. 21**
Методи виявлення викидів в пробних вибірках при управлінні процесами в системах за станом

- **Подгайко В.О., Рассомахін С.Г. 41**
Анализ алгоритмов идентификации на основе биометрии в системах электронных доверительных услуг

- **Помогаєв К. О., Толстолузька О. Г., Артюх О. А. 47**
Дослідження можливостей моделі 3d-візуалізації навчальних приміщень кафедри

- **Усатова О.О., Крютченко Д.В. 56**
Гідропружні коливання оболонок обертання при комбінованих вертикальних та горизонтальних навантаженнях

- **Ніколюк П.К. 67**
Методологія вибору оптимального міського автомобільного маршруту

CONTENTS

▪ Vervejko K., Drushliak V., Kizilova N., Koryakina O.	6
Fractal structures and mechanical properties of ammonite shells	
▪ Gamzayev R., Muradova V., Tkachuk M.	12
A study on alternative container-based technologies for virtualization of components deployment in software product lines	
▪ Lykhach O., Ugryumov M., Shevchenko D., Shmatkov S.	21
Anomaly detection methods in sample datasets when managing processes in systems by the state	
▪ Podhaiko V.O., Rassomakhin S.G.	41
Analysis Of Biometric-Based Identification Algorithms In Electronic Trust Services Systems	
▪ Pomohaiev K., Tolstoluzskaya O., Artiuh A.	47
Research of the possibilities of 3d-visualisation model of classrooms of the department	
▪ Usatova O., Kriutchenko D.	56
Hydroelastic vibrations of shells of revolution under coupled vertical and horizontal loadings	
▪ Nikolyuk P.K.	67
Navigation methodology for vehicle city route optimal choice	

УДК 532.546+514.86

Фрактальні структури і механічні властивості раковин амонітів

К.Є. Вервейко, В.В. Друшляк, Н.М. Кізілова, О.О. Корякіна,

- Вервейко Катерина Євгенівна** студент, кафедра прикладної математики, Харківський національний університет імені В. Н. Каразіна, пл. Свободи, 4, Харків-22, Україна, 61022; e-mail: katerynaverveiko@gmail.com
<https://orcid.org/0000-0002-2539-5403>
- Друшляк Владислав Віталійович** студент, кафедра прикладної математики, Харківський національний університет імені В. Н. Каразіна, пл. Свободи, 4, Харків-22, Україна, 61022; e-mail: vladislav.drushliak@gmail.com
<https://orcid.org/0000-0003-3937-6139>
- Кізілова Наталія Миколаївна** д.ф.-м.н., професор; професор кафедри прикладної математики, Харківський національний університет імені В. Н. Каразіна, пл. Свободи, 4, Харків-22, Україна, 61022; e-mail: n.kizilova@gmail.com
<https://orcid.org/0000-0001-9981-7616>
- Корякіна Ольга Олександрівна** студентка, кафедра прикладної математики, Харківський національний університет імені В. Н. Каразіна, пл. Свободи, 4, Харків-22, Україна, 61022; e-mail: korykina67@gmail.com
<https://orcid.org/0000-0003-2510-3615>

Досліджується фрактальна структура лопатевих ліній раковин вимерлих форм амонітів різних типів. Ці лінії відповідають з'єднувальному швам між попередньою та новою камерами, які будує молюск протягом свого поступового зростання. Еволюційно старі форми мають майже прямі або хвилясті шви, які мають низьку звивистість (відношення довжини лінії до відстані між кінцевими точками). Протягом еволюції форма цих ліній суттєво складніша і набула фрактальні властивості, так що еволюційно нові типи мають з'єднувальні шви складної форми. З використанням методу box-counting були виміряні фрактальні розмірності D_f лопатевих ліній. Було показано, що раковини різного еволюційного віку мають лінії, фрактальні розмірності яких зростають від $D_f=1.1$ до $D_f=1.9$. Крім того, зі зростанням номеру камери значення D_f зростають, причому залишаються більшими на зовнішній поверхні камери ніж на її внутрішній. Обговорюється зв'язок фрактальної структури з підвищенням механічної міцності шву.

Ключові слова: математичне моделювання, фрактальні структури, нарощування порожнистих структур, механічна міцність, еволюція

Fractal structures and mechanical properties of ammonite shells

- Vervejko Kateryna** student
V.N. Karazin Kharkov National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine
- Drushliak Vladyslav** student
V.N. Karazin Kharkov National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine
- Kizilova Nataliya** Doctor of Physical and Mathematical Sciences, Professor; Professor of the Department of Applied Mathematics
V.N. Karazin Kharkov National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine
- Koryakina Olga** student
V.N. Karazin Kharkov National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine

The fractal structure of the blade-type lines of shells of extinct forms of ammonites of various types is investigated. These lines correspond to the connecting seams between the previous and new chambers that the mollusc builds during its gradual growth. Evolutionarily older forms have nearly straight or wavy seams with low tortuosity (the ratio of line length to end-point distance). Over the course of evolution, the shape of these lines has become more complex and acquired fractal properties, so that evolutionarily new types have connecting seams of a very complex shape. The fractal dimensions D_f of the blade lines were measured using the box-counting method. It was shown that the fractal dimensions of the lobed lines of shells of different evolutionary ages increase from $D_f=1.1$ to $D_f=1.9$. In addition, with an increase in the chamber number, the D_f values increase

and remain larger on the outer surface of the chamber than on the inner one. The relationship between the fractal structure and an increase in the mechanical strength of the weld is discussed.

Keywords: *mathematical modeling, fractal structures, build-up of hollow structures, mechanical strength, evolution.*

1 Вступ

Самоподібні (фрактальні) структури зустрічаються в живій і неживій природі майже на всіх рівнях від макроскопічного (гірські хребти, річкові системи, берегові лінії материків і островів) до мікро/нано рівня (мікроструктури ґрунтів, кристали, клітини і субклітинні структури) [1]. Ці структури утворюються з використанням простих фізичних принципів, які повторюються на поступово зменшуваних масштабах, що приводить до формування достатньо складних геометричних форм. Деякі з них, наприклад, бінарні дерева трубок, що утворюють системи кровоносних судин, дихальних шляхів, судин рослин та ін., з достатньо високою точністю відповідають моделі оптимального трубопроводу, який постачає та розподіляє рідину між розподіленою системою споживачів (клітин організму) [2]. Фрактальні структури овочів відповідають розв'язку задачі про оптимальне компакне заповнення простору. Розгалужені системи від мікрОВОЛОКНИН в клітинах до гілок дерев відповідають механічним структурам з максимальною міцністю [3]. Саме тому принципи фрактальної будови 1D, 2D і 3D структур в природі відповідають універсальним принципам оптимальності [4]. Найбільш поширеною моделлю утворення фракталів в природі є агрегація з обмеженою дифузією (diffusion-limited aggregation, DLA), тобто рух, агрегація і осадження (абсорбція) частинок на поверхнях, коли з поступовим зростанням складності поверхні зменшується можливість постачання нових частинок із розчину або повітря. В результаті рівняння моделі DLA достатньо точно описують такі різні процеси як електроосадження металевих частинок з утворенням розгалужених металевих кристалів, зростання бактеріальних колоній, утворення і розповсюдження тріщин і зростання агломерацій міст.

Пористі структури в верхніх дихальних шляхах тварин забезпечують швидкий нагрів холодного повітря у арктичних тварин і ефективне охолодження і насичення вологою – у тварин пустель. Фрактальна розмірність таких структур близька до розмірності килиму Серпінського ($D_f=1.893$) і має дуже низькі величини гідравлічного опору потоку повітря і високі значення P/S (P – периметр, S – площа перерізу), що забезпечує ефективне зволоження повітря в системі [5]. Принципи будови природних пористих структур можуть використовуватися для підвищення ефективності технічних тепломасообмінників, нагрівачів і охолоджувачів.

В даній роботі вивчаються фрактальні властивості сполучних швів у раковин амонітів. Подібні шви також є в панцирах діатомових водоростей, з'єднаннях плоских кісток у тварин і людини, форма яких варіює від синусоїдної, трикутної, анти-трапецієвидної до фрактальних [6]. В останні роки було опубліковано кілька статей, в яких вивчалися фрактальні властивості сполучних швів раковин деяких класів амонітів [7,8].

2 Аналіз геометрії раковин амонітів

Амоніти є групою вимерлих морських молюсків, які жили у спіралеподібних раковинах. М'яке тіло молюска розташується у зовнішній камері і тому, зі зростанням тіла, молюск «добудовує» нові камери більшого розміру, що приводить до спіралеподібної геометрії (Рис.1). Якщо судити з сучасних наутілусів, більшість «нежилих» камер заповнена газом (повітряні камери), а кілька – рідиною (гідростатичні камери). Завдяки активному гіперосмосу молюск виводив або додавав воду з цих камер раковини, що дозволяло керувати плавучістю і підніматися чи опускатися в воді. Перегородка між камерами має гофрований край, який утворює складну лінію прикріплення до раковини – лопатеву лінію, геометрія якої є однією з систематичних ознак амонітів. Протягом еволюції від ~409 до 66 млн.р. до н.е. лопатеві лінії істотно змінювали форму від майже прямих (Nautiloid, Рис.1а) і «хвилеподібних» (Goniatitic, Рис.1б) до зубчастих (Ceratic, Рис.1в) і складних розгалужених (Amonitic, Рис.1г). Внутрішні процеси добудови нової камери проходять за рахунок агломерації і осадження солей Са шляхом DLA, що приводить до утворення міцного 3D з'єднувального шву (Рис.2а), фрактальна форма якого добре помітна на уламках раковин (Рис.2б).

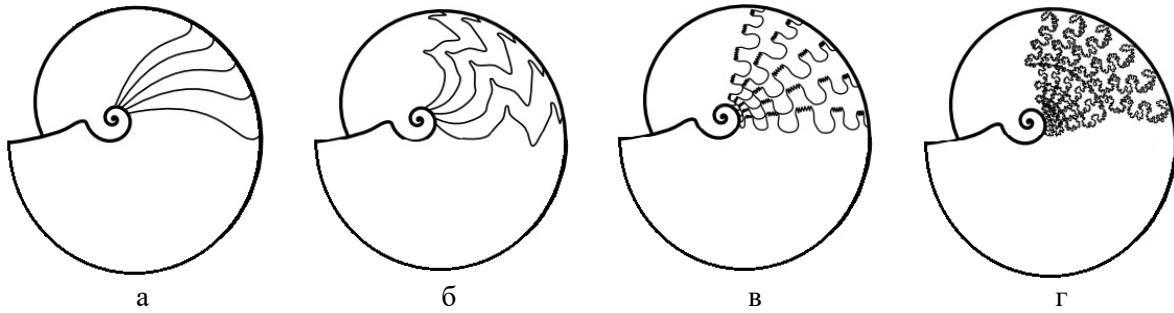


Рис.1. Чотири типи лопатевих ліній амонітів: Nautiloid (а), Goniatitic (б), Ceratitic (в), Amonitic (г)

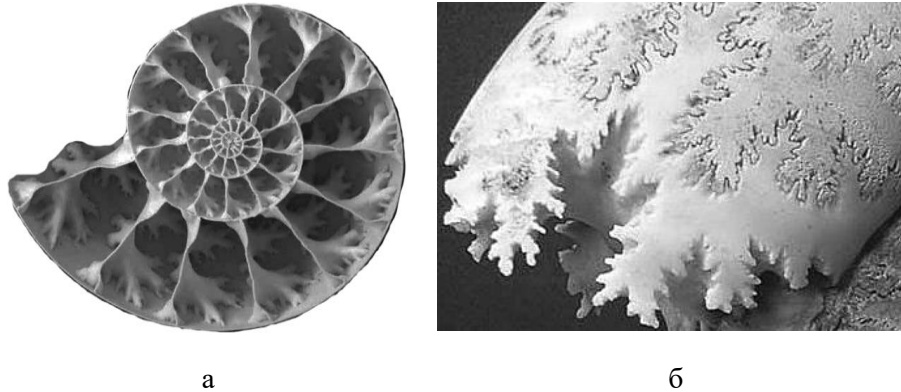


Рис.2. Внутрішня будова стінок камер (а) і лопатевої лінії (б) амонітів.

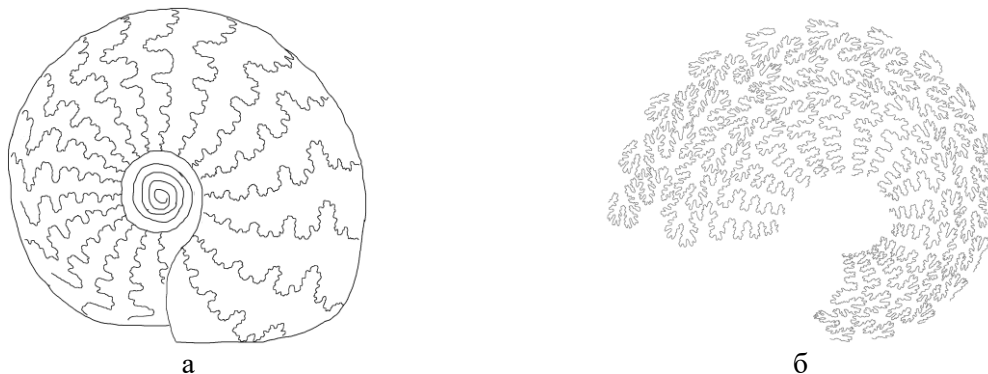
З фотографій високої роздільності (600dpi) раковин амонітів із відкритих онлайн колекцій палеонтологічних музеїв оцифрували за допомогою градієнтних фільтрів контури границь між сусідніми камерами (Рис.3). Результати показують, що складність ліній, які розташовані поблизу зовнішньої поверхні раковини вище, ніж ліній, які розташовані поблизу внутрішньої поверхні. Крім того, складність ліній між першими (малими) камерами значно нижча у порівнянні зі складністю ліній між більш крупними камерами. Таким чином, фрактальні властивості з'єднувального шву можуть відповідати потрібній механічній міцності з'єднання камер.

Фрактальні розмірності ліній вимірювали за допомогою власного коду і вбудованих бібліотек пакету Python. Був використаний метод box-counting, який складається з підрахунку числа квадратів N_ε з довжиною сторони ε та з послідовним зменшенням ε обчислити значення

$$D_f = - \lim_{\varepsilon \rightarrow 0} \frac{\ln(N_\varepsilon)}{\ln(\varepsilon)} \quad (1)$$

з достатньою точністю.

Валідація коду проводилася шляхом вимірювань фрактальної розмірності відомих кривих фон Коха, Мінковського та Гілберта. Розрахунки проводилися як для цілих ліній (Рис.3в,г), так і для їх першої (I), другої (II), третьої (III) та останньої (IV) четвертин.



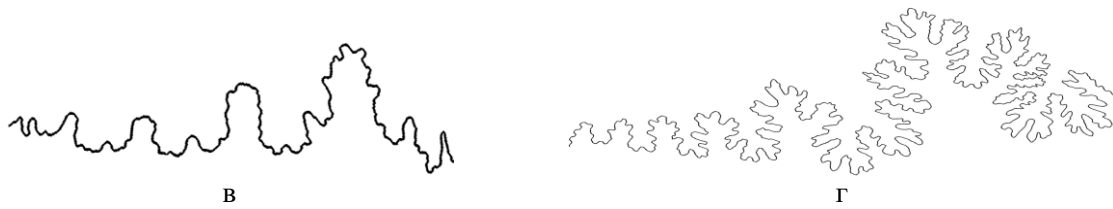


Рис.3. Оцифровані лопатеві лінії раковин тунів Amonitic (а,б) та їх окремі лінії (в,г).

3 Результати розрахунків і їх обговорення

Результати розрахунків D_f для 16 лопатевих ліній амонітового типу (Рис3б) наведені на Рис.4. Значення фрактальних розмірностей ліній лежать в діапазоні $1.25 < D_f < 1.48$, причому нижнє значення відповідає розмірності кривої фон Коха з квадратом замість трикутника ($D_f=1.26186$), а верхнє значення – модифікованій кривій фон Коха з квадратом замість трикутника ($D_f=1.46497$). Внутрішні частини лопатевих ліній (I і II) мають значно простіші форми ($D_f \sim 1.2-1.3$) у порівнянні з зовнішніми частинами (III, IV), які мають $D_f \sim 1.4-1.5$.

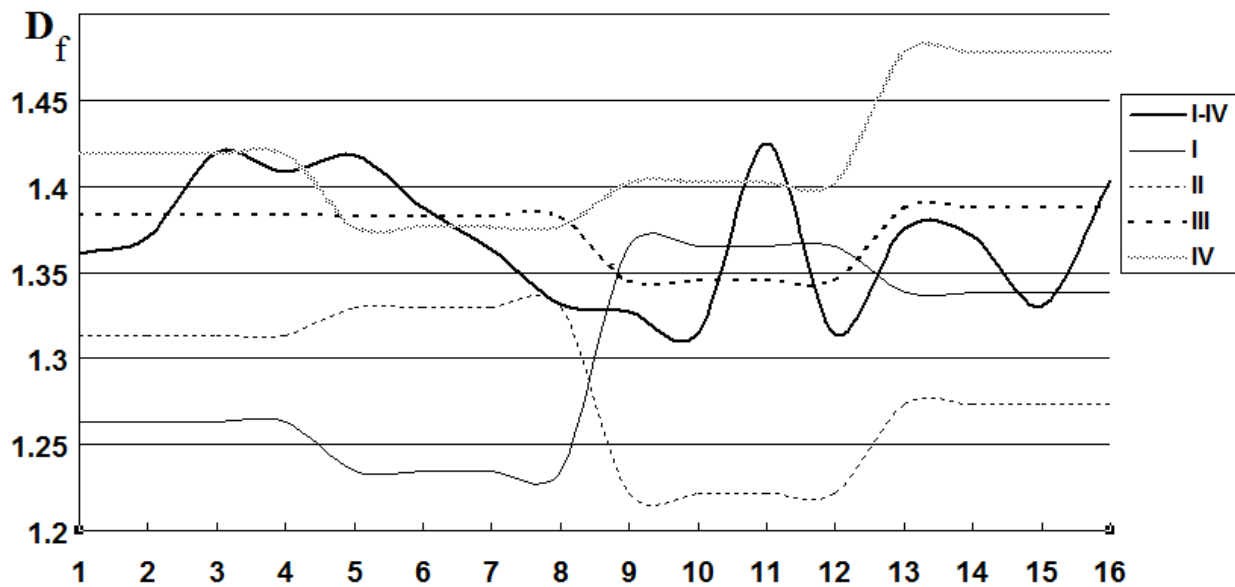

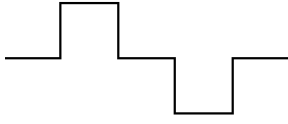

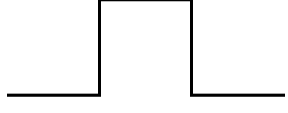
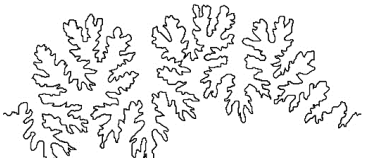
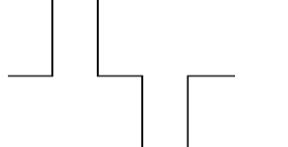

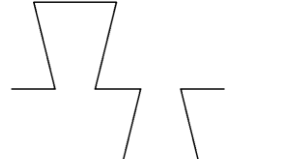
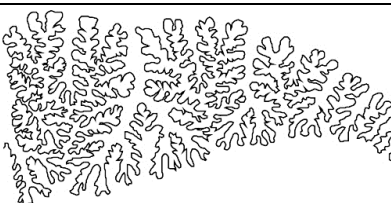
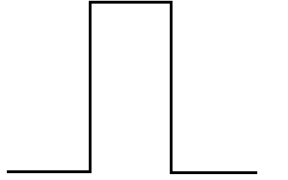
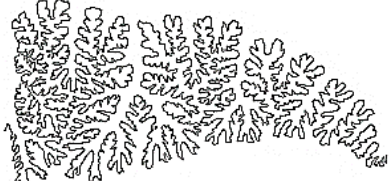
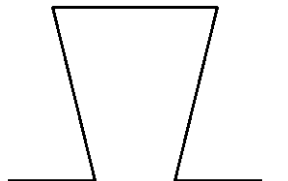


Рис.4. Значення D_f для лопатевих ліній в цілому (I-IV) і їх частин I, II, III, IV.

Для еволюційно молодих форм Goniatic, Seratic фрактальні розмірності внутрішніх частин ліній наближуються до $D_f \sim 1.1-1.2$. Спираючись на результати розрахунків можна виділити наступні за складністю типи лопатевих ліній та відповідні їм фрактальні моделі (Табл.1). Таким чином, найбільш складні лінії з $D_f \sim 1.6-1.8$ відповідають найміцнішим механічним закріпленням швів у вигляді анти-трапецієвидної форми [6]. Цікаво, що природні пористі структури дихальних шляхів арктичних тварин теж мають фрактальної розмірності з цього ж діапазону значень $D_f \sim 1.6-1.8$, а найближчою моделлю для таких структур є килими Серпінського різної форми [5].

Таблиця 1. Характерні типи лопатевих ліній та їх математичні моделі.

	Лінія	D_f лінії	Модель	D_f моделі
1		1.1-1.2		$\frac{\ln 7}{\ln 5} \approx 1.20906$
2		1.2-1.3		$\frac{\ln 4}{\ln 3} \approx 1.26186$

3		1.3-1.4		$\frac{\ln 9}{\ln 5} \approx 1.36521$
4		1.4-1.5		$\frac{\ln 5}{\ln 3} \approx 1.46497$
5		1.5-1.6		$\frac{\ln 13}{\ln 5} \approx 1.5937$
6		1.6-1.7		$\frac{\ln 15}{\ln 5} \approx 1.6826$
7		1.7-1.8		$\frac{\ln 7}{\ln 3} \approx 1.7712$
8		1.8-1.9		$\frac{\ln 8}{\ln 3} \approx 1.8928$

5 Висновки

Еволюційні зміни раковин амонітів протягом ~340 млн. років набували значних змін форми лопатевих ліній, які відповідають з'єднувальним швам між сусідніми камерами, які надбудує молюск в порожнинній раковині. Структура цих ліній змінювалася від майже плоских (кола в гофрованій структурі, Рис.1а) у Nautiloid до само подібних звивистих форм з фрактальними розмірностями $D_f \sim 1.1-1.2$ у Goniatic, $D_f \sim 1.2 - 1.5$ - у Ceratic, і $D_f \sim 1.2-1.9$ - у Amonitic. При цьому частини ліній, які розташовані на зовнішній поверхні раковини і зазнають значніших механічних навантажень, є більш складнішими і мають вищі значення D_f . Ті лінії, які розташовані на внутрішніх поверхнях, що звиті у спіраль і, таким чином, мають додаткове механічне підкріплення, є більш простішими і мають менші значення D_f . Побудовані математичні моделі відповідних фрактальних ліній, які можна запропонувати до використання в якості з'єднувальних швів в технічних конструкціях.

ЛІТЕРАТУРА

1. Mandelbrot B. *The Fractal Geometry of Nature*. Echo Point Books & Media, LLC. 1982. - 490 p.
2. Кизилова Н.Н., Попова Н.А. Критерии оптимального функционирования ветвящихся транспортных систем живой природы. // *Вестник Харьковского университета. Сер. Математика, прикладная математика и механика*. - 1999, N444. - С.148-156.

3. Kizilova N. Biomimetic composites reinforced by branched nanofibers. // *Nanoplasmonics, Nano-Optics, Nanocomposites, and Surface Studies. Springer Proceedings in Physics*, Vol.167. O. Fesenko and L. Yatsenko, (Eds.). – 2015. – P. 7–23.
4. Ball Ph. *The self-made tapestry: pattern formation in nature*. Oxford: Oxford University Press. 2001. – 288 p.
5. Кізілова Н.М., Світличний О.В., Чуйко В.В. Дослідження зв'язку фрактальних та гідравлічних властивостей пористих структур верхніх дихальних шляхів деяких арктичних тварин. // *Вісник Харківського національного університету. Сер. «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. – 2020. – вип. 46. – С.17-24. <https://doi.org/10.26565/2304-6201-2020-46-02>
6. Li Y., Ortiz Ch., Boyce M.C. A generalized mechanical model for suture interfaces of arbitrary geometry. // *Journal of the Mechanics and Physics of Solids*. 2013. Vol.61. - P.1144–1167. <http://dx.doi.org/10.1016/j.jmps.2012.10.004>
7. Inoue Sh., Kondo Sh. Suture pattern formation in ammonites and the unknown rear mantle structure. // *Scientific Reports*. - 2016. Vol.6. - P.33689. doi: 10.1038/srep3368.
8. Pérez-Claros J.A., Bengtson P. Evolution of complexity and natural selection: Suture complexity and its relation to taxonomic longevity in Cretaceous ammonoids. // *Cretaceous Research*. - 2018. - Vol. 88. P. 55-61. <http://dx.doi.org/10.1016/j.cretres.2017.02.008>

REFERENCES

1. B. Mandelbrot, *The Fractal Geometry of Nature*, Echo Point Books & Media, LLC, 1982, 490 p.
2. N. Kizilova, N. Popova, “Criteria for the optimal functioning of branching transport systems of living nature”, *Bulletin of Kharkov University, Ser. Mathematics, applied mathematics and mechanics*, 1999, N444, P.148-156.
3. N. Kizilova, “Biomimetic composites reinforced by branched nanofibers”, In: *Nanoplasmonics, Nano-Optics, Nanocomposites, and Surface Studies. Springer Proceedings in Physics*, Vol.167. O. Fesenko and L. Yatsenko, (Eds.), 2015, P. 7–23.
4. Ph. Ball, *The self-made tapestry: pattern formation in nature*, Oxford University Press, Oxford, 2001, 288 p.
5. N. Kizilova, O. Svitlichnyj, V. Chujko, “Study on connections between the fractal and hydraulic properties of porous structures of the upper airways of some arctic animals”, V.N. Karazin Kharkov National University, Ser. Mathematic modeling. Information technology. Automated control systems, 2020, Vol. 46, C.17-24. <https://doi.org/10.26565/2304-6201-2020-46-02>
6. Y. Li, Ch. Ortiz, M.C. Boyce, “A generalized mechanical model for suture interfaces of arbitrary geometry”, *Journal of the Mechanics and Physics of Solids*, 2013, Vol.61, P.1144–1167. <http://dx.doi.org/10.1016/j.jmps.2012.10.004>
7. Sh. Inoue, Sh. Kondo, “Suture pattern formation in ammonites and the unknown rear mantle structure”, *Scientific Reports*, 2016, Vol.6, P.33689. doi: 10.1038/srep3368.
8. J.A. Pérez-Claros, P. Bengtson, “Evolution of complexity and natural selection: Suture complexity and its relation to taxonomic longevity in Cretaceous ammonoids”, *Cretaceous Research*, 2018, Vol. 88, P. 55-61. <http://dx.doi.org/10.1016/j.cretres.2017.02.008>

УДК 004.051

Дослідження альтернативних технологій контейнерів для віртуалізації процесів розміщення компонентів лінійок програмних продуктів

Р.О. Гамзаєв¹, В.Х. Мурадова^{1,2}, М.В. Ткачук¹¹Харківський національний університет імені В. Н. Каразіна, майдан Свободи 4, м. Харків, 61022, Україна²Харківський національний університет радіоелектроніки, проспект Науки 14, м. Харків, 61166, Україна**Гамзаєв Рустам
Олександрович***кандидат технічних наук, доцент, докторант кафедри моделювання систем і технологій, Харківський національний університет імені В.Н. Каразіна, м.Харків, Майдан Свободи 4, 61022; e-mail: rustam.gamzayev@karazin.ua; <https://orcid.org/0000-0002-2713-5664>***Мурадова Всаля
Худашірін кизи***кандидат технічних наук, старший викладач кафедри моделювання систем і технологій, Харківський національний університет імені В.Н. Каразіна, м. Харків, Майдан Свободи 4, 61022; e-mail : v.muradova@karazin.ua ; <https://orcid.org/0000-0001-6304-8325>***Ткачук Микола
Вячеславович***доктор технічних наук, професор, завідувач кафедри моделювання систем і технологій, Харківський національний університет імені В.Н. Каразіна, м. Харків, Майдан Свободи 4, 61022; e-mail: mykola.tkachuk@karazin.ua; <https://orcid.org/0000-0003-0852-108>*

У статті пропонується підхід до підтримки гнучкої розробки лінійок програмних продуктів (ЛПП) з використанням методів керування варіабельністю в межах методології Scrum. Головною метою роботи є аналіз контейнерів для віртуалізації середовищ при розгортанні ЛПП. Структурована інформаційна база для запропонованого підходу управління варіабельністю розгортання, показана його роль у загальному методі Scrum та запропоновано концептуальну схему процесу управління на етапі розгортання додатку. Проведено експериментальний аналіз та розраховані метрики для двох типів контейнерів Docker та Vagrant.

Ключові слова: *варіабельність, Docker, Vagrant, варіабельність, контейнеризація, ЛПП*

A study on alternative container-based technologies for virtualization of components deployment in software product lines

R.A. Gamzayev¹, V.Kh. Muradova^{1,2}, M.V. Tkachuk¹¹V. N. Karazin kharkiv national university, freedom square 4, Kharkive, 61022, Ukraine²Kharkiv national university of radioelectronics, avenue science 14, 61166, Ukraine**Gamzayev Rustam***PhD, asc. professor, post-doctorate of the Department of Systems and Technology Modeling, V.N. Karazin Kharkiv National University, Kharkiv, Freedom Square 4, 61022; e-mail: rustam.gamzayev@karazin.ua ; <https://orcid.org/0000-0002-2713-5664>***Muradova Vusala***Candidate of technical sciences, senior lecturer of the Department of Systems and Technology Modeling, V.N. Karazin Kharkiv National University, Kharkiv, Freedom Square 4, 61022; e-mail : v.muradova@karazin.ua; <https://orcid.org/0000-0001-6304-8325>***Tkachuk Mykola***Dc. of techn. science, professor, head of the Department of Systems and Technology Modeling, V.N. Karazin Kharkiv National University, Kharkiv, Freedom Square 4, 61022, e-mail: mykola.tkachuk@karazin.ua; <https://orcid.org/0000-0003-0852-108>*

The application containerization approach allows creating virtualization environments that could be used as a code. It allows running application in the isolated container that could be reproduced on any other hardware or cloud environment. One of the benefits of the containerization approach is the possibility to allocate necessary hardware resources like a RAM, CPU and storage. An approach to support agile development of software product lines (SPL) by using variability management

techniques within the framework of the Scrum methodology has been proposed in the article. The main goal of the work is to analyze containers for virtualization of the runtime environment when deploying SPL. The information base for the proposed approach to managing the variability of deployment has been structured. The role of the approach in the general method of Scrum has been shown, and a conceptual diagram of the management process at the stage of application deployment has been proposed. The experimental analysis has been carried out and metrics for two types of containers, Docker and Vagrant, have been calculated. The following two metrics, namely, portability and productivity, for both containers have been analyzed. These metrics for the test component software solution have been calculated and executed in cloud environment with different configurations. The portability metric indicates how easily the application can be migrated to other platform basing on the time required to start container with the application. The second metric is the time necessary for the same operations in different container.

Keywords: *variability, Docker, Vagrant, containerization, SPL*

1. Вступ. Актуальність дослідження

Сучасні системи розробки програмного забезпечення (ПЗ) складаються з ряду сервісів, компонентів і класів, які можуть бути використані повторно. Це особливо важливо при розробці (ЛПП) і вимагає встановлення механізму управління варіабельністю вимог. Часто такі задачі погано формалізовані і вимагають комплексного програмно-аналітичного рішення. Отримані рішення дозволяють досягти певного ступеня подібності розроблених структур, як результат можна стверджувати, що відмінності виникають внаслідок варіацій програмних артефактів.

Під час розробки ЛПП, процес управління варіабельністю зв'язує етапи аналізу вимог та етап будівництва певної одиниці ЛПП [1]. На етапі впровадження часто виникає необхідність ефективного конфігурування та налаштування розроблених програмних компонентів з урахуванням ресурсів та інтерфейсів зовнішніх бібліотек, операційних платформ, програмних сервісів тощо. Відсутність процесів накопичення даних про деталі етапів розгортання кожного з повторно використовуваних компонентів призводить до необхідності повторення цих етапів у випадку використання варіабельних компонентів.

Зокрема, у роботі [2] показано, що завдяки наявності зв'язків між вимогами та артефактами, було можливим покращити повноту та точність відстеження вимог при гнучкій розробці ПЗ шляхом розробки знання-орієнтованих моделей та технології накопичення інформації, аналіз та управління даними про вимоги користувачів. Проте запропонований у роботі [2] підхід був застосований для розробки окремого продукту та не має можливості застосовувати дані між різними проектами. Таким чином, метою цього дослідження є розробка підходу до аналізу можливостей забезпечення варіабельності на етапі розгортання компонентів ПЗ шляхом використання альтернативних сучасних технологій для контейнеризації застосунків.

2. Технологічна схема управління варіабельністю розгортання програмних компонентів в контексті методології Scrum

В [3] була запропонована інтегрована модельно-технологічна схема розробки та супроводу ЛПП, яка поєднує в собі 2-х рівневу концептуальну модель процесів створення будь-якої ЛПП з роботи К. Поля (К. Pohl et al.) [1] з узагальненою технологічною схемою гнучкої розробки ПЗ (напр., за методологією Scrum та ін.), що дозволяє розглядати всі основні фази життєвого циклу (ЖЦ) ЛПП з урахуванням можливостей забезпечення варіабельності відповідних проектних артефактів, а саме:

- 1) багатовимірних специфікацій вимог користувачів (user stories), які формуються на основі аналізу певної предметної області (ПрО) за участю експертів та з використання методів обробки знань [4];
- 2) альтернативних доменних моделей (domain models), які будуються на основі опрацювання цих вимог, і які можуть бути використані в подальшому для генерації каркас вихідного коду (source code framework) з необхідним коефіцієнтом його повторного використання [5];
- 3) різних варіантів архітектури програмних компонентів (software components), що входять до складу цільової ЛПП, і які мають бути розміщені (components deployment) на відповідній операційній платформі в тій бізнес-організації, яка буде в подальшому використовувати функціональні можливості цієї ЛПП.

В роботах [3 – 5] для автоматизації процесів обробки даних та експертних знань на етапах (1) – (2) були представлені відповідні методи та інструментальні засоби, які структурно і функціонально входять в згадану вище інтегровану модельно-технологічну схему розробки

ЛПП в вигляді контурів управління зі зворотним зв'язком (feedback control loop). Тепер, зважаючи на необхідність забезпечити також і на етапі (3) можливість вибору та використання альтернативних технологій розгортання програмних компонентів, пропонується розширена схема процесів ЖЦ ЛПП у поєднанні с методологією Scrum, яка представлена на рисунку 2.1.

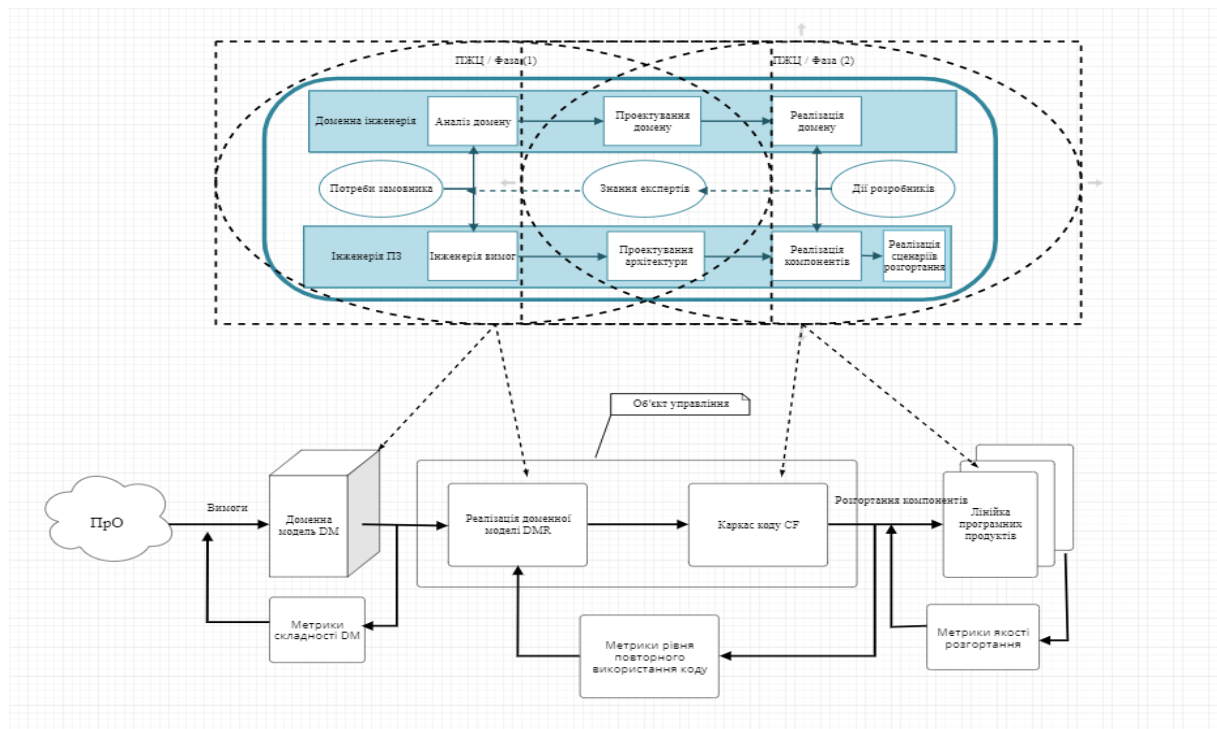


Рис.2.1 Розширена схема процесів ЖЦ ЛПП в контексті методології Scrum

Ця схема містить додатковий контур зворотного зв'язку для управління процесом розгортання програмних компонентів і для забезпечення властивостей варіабельності на цьому етапі ЖЦ ЛПП необхідність розробити підхід до оцінювання якості застосування альтернативних технологій їх розгортання, до яких належать, зокрема, технології контейнеризації та віртуалізації програмних проєктів [6]. Ці питання більш детально розглянуті у наступних розділах цієї статті.

3. Методика дослідження процесів розгортання компонентів з використанням альтернативних технологій Docker та Vagrant

Для вирішення поставлених задач цього дослідження необхідно розробити достатньо функціонально повний прототип компонентного прос[грамного рішення (КПР), для розгортання якого застосовуються альтернативні технології контейнеризації та віртуалізації проєктів, а також запропонувати відповідні метрики обчислення показників якості виконання цих процесів.

3.1. Прототип тестового КПР для дослідження альтернативних технологій розгортання програмних проєктів

Для дослідження особливостей процесів розгортання тестового КПР з використанням альтернативних технологій контейнеризації була розроблена архітектура, яка реалізує функціональні можливості системи адаптивного управління вимогами на основі використання моделі сфокусованого інтерфейсу розробника [7], і вона представлена на рис. 3.1 у вигляді UML – діаграми розгортання компонентів (component deployment diagram).

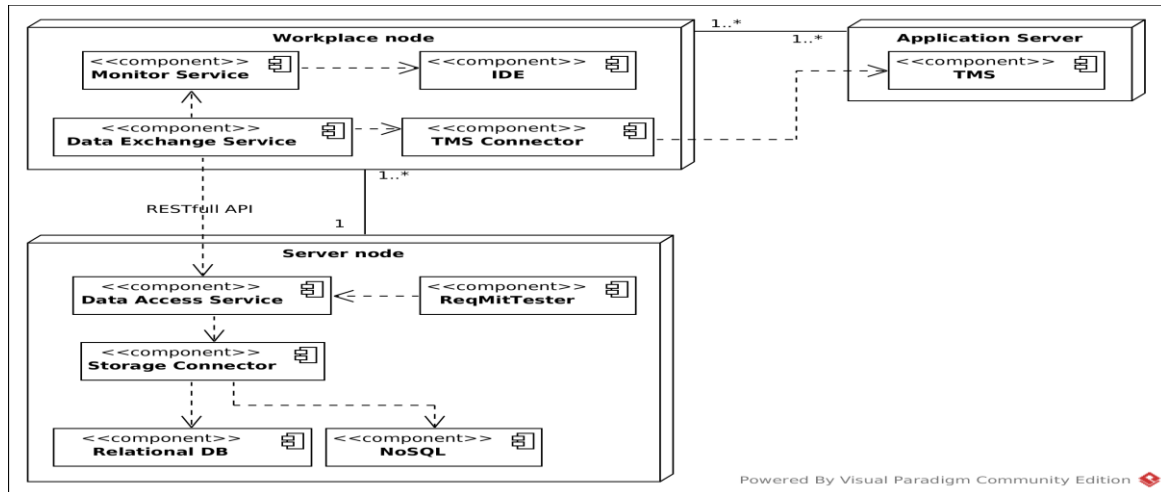


Рис. 3.1 Діаграми розгортання компонентів тестового КПП

У цьому КПП реалізовані наступні основні функціональні можливості: на вузлі Server Node розміщено компоненти Data Access Service є контролером, який займається обробкою запитів від клієнтських застосунків, а також забезпечує доступ з використанням сервісу Storage Connector до ресурсів даних, що знаходяться під управлінням реляційної СКБД Relation DB (MySQL) та альтернативної СКБД NoSQL (Cassandra), і нарешті, компонент ReqMitTester дозволяє розраховувати відповідні метрики і на підставі цього будувати сфокусований інтерфейс розробника ПЗ [7].

Для розгортання цього тестового КПП були написані відповідні скриптові сценарії для технології Docker та Vargant (див. рисунок 3.2).

```

docker.sh
1 cp docker/containers/app/.env.example docker/containers/app/.env
2 cp docker/containers/db/.env.example docker/containers/db/.env
3 docker network create reqmit2-auth-dbnet
4 docker-compose up -d testdb
5 docker-compose -f docker/build.yml build
6 docker-compose build
7 docker-compose -f docker/build.yml up --force-recreate --no-deps build
8 docker-compose up -d db cassandra
9 docker-compose up -d --force-recreate --no-deps api
10 docker exec -t reqmit2-auth-api ./gradlew flywayMigrate --info
11

vagrant.sh
1 service mysqld start
2 service cassandrad start
3 mysql -u mysql -c DROP DATABASE test; CREATE DATABASE test;
4 ./gradlew clean flywayClean check build flywayMigrate
5 vagrant up
    
```

Рис. 3.2 Сценарій розгортання КПП з використанням Docker та Vargant

3.2. Показники для оцінки якості процесів розгортання КПП та метрики їх визначення

Для аналізу використання Docker та Vagrant для розгортання тестового КПП (див. рисунок 3.2) розглядалися наступні показники якості [8]:

- переносимість (Portability - Prb);
- продуктивність (Performance - Prm).

Переносимість є мірою того, наскільки легко відповідне КПП може бути перенесена з однієї операційної платформи на іншу, тобто кількісно вона може бути обчислена як величина, що є зворотно пропорційною витратам часу, який є необхідним для здійснення цього процесу. Таким чином, метрика для Prb має наступний вигляд:

$$Prb = \frac{1}{T(Deployment)}, \quad (1)$$

де $T(Deployment)$ - це витрати часу на розгортання нової версії проекту.

Беручи до уваги деякі технологічні особливості процесу контейнеризації [9] розрахунок за формулою (1) зводиться до визначення витрат часу безпосередньо на процес розгортання операційного середовища (контейнера) « $T(Environment\ deployment)$ » та витрат часу на налаштування та підготовки до розгортання відповідного серверу додатків « $T(Config.\ and\ preparation)$ ». В свою чергу, час « $T(Environment\ deployment)$ » складається з часу, необхідного для створення (build) контейнеру чи віртуальної машини, та часу для їх безпосереднього виконання (run).

Таким чином, для варіанту застосування технології Docker витрати часу на процесу розгортання має наступний вигляд:

$$T(Environment\ deployment)_{Docker} = T(Docker\ build) + T(Docker\ run), \quad (2)$$

а для варіанту застосування технології Vagrant вони обраховуються за формулою:

$$T(Environment\ deployment)_{Vagrant} = T(Vagrant\ setup) \quad (3)$$

де « $T(Vagrant\ setup)$ » - це час створення та запуску віртуальної машини Vagrant.

Отже, враховуючи вирази (1) – (3) кінцевий вираз для обчислення метрики оцінки показника якості переносимості Prb розраховується за формулою:

$$Prb = \frac{1}{T((Env\ deployment)) + T((Config.\ and\ preparation))} \quad (4)$$

Для обчислення загальної метрики оцінки продуктивності Prm процесу розгортання в даній роботі використовуються наступні параметри [10]:

- метрика для визначення ступеню використання центрального процесора $M(CPU)$;
- метрика для визначення об'єму оперативної пам'яті $M(RAM)$;
- метрика для визначення середнього завантаження операційної системи $M(LoadAvg)$.

Метрика $M(CPU)$ може бути розрахована як середнє навантаження на центральний процесор внаслідок виконання певних операцій:

$$M(CPU) = 1 - \frac{\sum_{i=1}^N (U_{avg}(CPU)_i)}{N} \quad (5)$$

де $U_{avg}(CPU)_i$ це середнє навантаження на центральний процесор для кожної операції; а N - кількість виконаних операцій.

Метрика $M(RAM)$ дозволяє визначити об'єму оперативної пам'яті, яка має бути виділена для розгортання КПП з використанням однієї з альтернативних технологій Docker та Vagrant, і вона має наступний вигляд:

$$M(RAM) = \frac{1}{V_2(RAM) - V_1(RAM)} \quad (6)$$

де $V_1(RAM)$ - це об'єм оперативної пам'яті, яка використовується до початку процесу розгортання відповідного КПП, а $V_2(RAM)$ - це об'єм оперативної пам'яті, яка задіяна після завершення процесу його розгортання.

Аналогічним чином може бути визначена метрика середнього завантаження операційної системи $M(LoadAvg)$ під час процесу розгортання КПП [10], а саме:

$$M(\text{Load Avg}) = \frac{1}{L_2 - L_1} \quad (7)$$

де, L_1 це середнє навантаження системи до початку процесу розгортання відповідного КНР, а L_2 – середнє навантаження після його завершення.

Для обчислення остаточного значення метрики Prm на основі окремих метрик (5) – (7) пропонується побудувати їх згортку у вигляді

$$Prm = k_1 M(\text{CPU}) + k_2 M(\text{RAM}) + k_3 M(\text{Load Avg}), \quad (8)$$

де відповідні вагові коефіцієнти k_1 , k_2 , k_3 можуть бути визначені з використанням одного з експертних методів, наприклад, за методом аналізу ієрархій (MAI) [11].

4. Проведення програмних експериментів та аналіз отриманих результатів

Для проведення обчислювальних експериментів використовувався екземпляр хмарного сервісу (cloud instance) «Amazon EC2 m4.2xlarge» [12], який мав 32 Гб оперативної пам'яті та віртуальний процесор 8vCPU. Програмний експеримент з оцінюванням параметра переносимості Prb було проведено 5 разів, і відповідний алгоритм в вигляді UML – діаграми діяльності представлено на рисунку 4.1.



Рис.4.1 Алгоритм проведення експерименту з обчислення метрики Prb

Результати цих експериментів для обох технологій: Docker та Vagrant приведені в Таблиці 1. Де T_1 та T_2 час початку та закінчення експерименту відповідно заданий у Unix часі. ΔT – час на розгортання застосунку у відповідному контейнеру.

Таблиця 1. Результати експериментів для оцінки метрики переносимості

№	T_1 (Docker)	T_2 (Docker)	ΔT (Docker)	T_1 (Vagrant)	T_2 (Vagrant)	ΔT (Vagrant)
1	1527592245	1527592498	253 с	1527593552	1527593879	327 с
2	1527592498	1527592741	243 с	1527593879	1527594168	289 с
3	1527592741	1527592978	237 с	1527594168	1527594479	311 с

4	1527592978	1527593245	267 с	1527594479	1527594812	333 с
5	1527593245	1527593552	307 с	1527594812	1527595091	279 с
T(P(Deployment with Docker))			261 с	T(P(Dep. with Vagrant))		290 с

Використовуючи дані з Табл. 1 та формули (1) – (4) були отримані наступні значення для оцінки якості процесу розгортання тестового КПП за метрикою Prb

$$Prb(\text{Docker}) = 0.0038, \quad Prb(\text{Vagrant}) = 0.0035, \quad (9)$$

Тобто, значення метрики переносимості Prb при використанні технології Docker більше значення цієї ж метрики при використанні технології Vagrant лише приблизно на 8%, що є досить незначною різницею.

Експеримент для обчислення значення метрики продуктивності Prm проводився наступним чином: програмним шляхом виконувались 100000 запитів обробки даних з використанням функціоналу тестового КПП (див. рисунок 4.1), збирались відповідні дані, а потім розраховувались показники метрик з використанням формул (5) – (7), і ці результати наведені в Таблиці 2.

Таблиця 2. Результати проведеного експерименту

Метрика / Технології	Docker	Vagrant
M(CPU)	0.63	0.56
M(RAM)	0.00049	0.00041
M(Load AVG)	0.14	0.083

Наступним кроком для визначення кінцевого значення метрики оцінки продуктивності Prm для варіантів використання Docker та Vagrant є застосування MAI за формулою (8). Для цього спочатку була побудована матриця попарних порівнянь пріоритетів для альтернатив Docker / Vagrant (див. Таблиця 3).

Таблиця 3. Матриця парних порівнянь альтернатив за методом MAI

	M(Load AVG)	M(RAM)	M(CPU)
M(Load AVG)	1	3/1	5/4
M(RAM)	1/3	1	3/6
M(CPU)	4/5	6/3	1

Наступним кроком є визначення значень вагових коефіцієнтів k_1, k_2, k_3 для застосування у формулі (8), див. Таблицю 4.

Таблиця 4. Значення вагових коефіцієнтів для методу MAI

	M(Load AVG)	M(RAM)	M(CPU)	Сума по рядкам	Вагові коефіцієнти k_1, k_2, k_3
M(Load AVG)	1	3	1.25	5.25	0.48
M(RAM)	0.33	1	0.5	1.83	0.17
M(CPU)	0.8	2	1	3.8	0.35

З використанням окремих метрик продуктивності з Табл. 3, значень вагових коефіцієнтів з Табл. 4 та застосовуючи формулу (8), були отримані комплексні оцінки метрики Prm для технологій Docker і Vagrant відповідно

$$\begin{aligned} Prm(\text{Docker}) &= 0.48 \cdot 0.14 + 0.17 \cdot 0.00049 + 0.35 \cdot 0.63 \approx 0.29 \\ Prm(\text{Vagrant}) &= 0.48 \cdot 0.083 + 0.17 \cdot 0.00041 + 0.35 \cdot 0.56 \approx 0.24 \end{aligned} \quad (10)$$

Тобто, аналізуючи результати метрики P_{rm} з виразу (9), можна зробити мотивований висновок, що продуктивність функціонування тестового КПП, який було розгорнуто за допомогою контейнерної технології Docker, є приблизно на 17% вищою, ніж у випадку його розгортання з використанням технології віртуалізації Vagrant.

5. Висновки та напрямки подальших досліджень

Як було показано у цієї статті контейнеризація з використанням Docker має значну перевагу з точки зору портативності, для аналізу продуктивності було розраховано кількісне значення завдяки порівнянню різних критеріїв з використанням методу аналізу ієрархій, а отримане значення також показало перевагу Docker контейнеризації. В подальшому планується розробити декілька ЛПП з використанням різних архітектурних підходів, та дослідити ефективність використання контейнеризації, а також проаналізувати оптимальну структуру поділу системи.

ЛІТЕРАТУРА

1. K. Pohl, et al: *Software Product Line Engineering: Foundations, Principles, and Techniques*. Springer, 2005, 467 pp.
2. Гамзаєв Р. О. *Модель та інформаційна технологія побудови адаптивної матриці трасування вимог у гнучких процесах розробки програмного забезпечення* / Р. О. Гамзаєв, М. В. Ткачук // *Вісник Нац. техн. ун-ту "ХПІ" : зб. наук. пр. Темат. вип. : Системний аналіз, управління та інформаційні технології*. – Харків : НТУ "ХПІ". – 2013. – № 2 (976). – С. 49-59.
3. Гамзаєв Р.О., Ткачук М.В., Шевкопляс Д.О. *Застосування знання-орієнтованих методів і технологій для моделювання варіабельності в розробці лінійок програмних продуктів* // *Матеріали міжн. науков-техн. конференції КМНТ-2021, (м. Харків, 23-25 квітня 2021 року) – Х.: ХНУ імені В.Н. Каразіна, 2021. - С. 104-107.*
4. Gamzayev R.O., Tkachuk M.V., Shevkoptyas D.O. *Handling of Expert Knowledge in Software Product Lines Development with Usage of Repertory Grids Method* // *Вісник Харківського національного університету імені В.Н. Каразіна, Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. - № 47, 2020. – С. 13-24.
5. Мартінкус І. О., Ткачук М. В., Гамзаєв Р. О. *Конструювання лінійок програмних продуктів із застосуванням доменного моделювання та метрик повторного використання коду* / І. О. Мартінкус, М. В. Ткачук, Р. О. Гамзаєв // *Системи управління, навігації та зв'язку: зб. наук. пр. ЦНДІ НУ*. – К., 2017. – Вип. 3(43).– С. 93-97.
6. Wes Felter, Alexandre Ferreira, Ram Rajamony, and Juan Rubio. *An Updated Performance Comparison of Virtual Machines and Linux Containers* // *IBM Research Report, RC25482 (AUS1407-001) July 21, 2014.*
7. Гамзаєв Р. О., *Мета-модель процесу трасування вимог при розробці програмного забезпечення* / Гамзаєв Р. О., Ткачук М. В., Мартінкус І. О. // *Вісник НТУ «ХПІ»*. Серія: *Нові рішення в сучасних технологіях*. – Х: НТУ «ХПІ», – 2014. – 26 (1069). – С.121-128
8. Андон Ф.И. *Основы инженерии качества программных систем* / Андон Ф.И., Коваль Г.И., Коротун Т.М. и др – 2-е изд. – К.: Академперіодика, 2007 – 672с.
9. D. N. Jha, S. Garg, R. Ranjan et al. *A Study on the Evaluation of HPC Microservices in Containerized Environment* // *Concurrency and Computation Practice and Experience* 33(1) Published 2 May 2019
10. Miguel G. Xavier; Marcelo V. Neves; Fabio D. Rossi; et al. *Performance Evaluation of Container-based Virtualization for High Performance Computing Environments* // *21st Euromicro Conference on Parallel, Distributed and Network-Based Processing, 2013, pp. 233 – 240.*
11. Саати, Т.Л. *Принятие решений. Метод анализа иерархий* / Саати Т.Л.; пер. с англ. – М.: Радио и связь, 1993. – 278 с.
12. Офіційний сайт компанії Amazon. - URL: <https://aws.amazon.com/ru/ec2/instance-types> (дата звернення:05.12.2021).

REFERENCES

1. K. Pohl, et al: *Software Product Line Engineering: Foundations, Principles, and Techniques*. Springer, 2005, 467 pp.
2. Gamzaev R.O. Model and information technology for building an adaptive matrix of tracing requirements in flexible software development processes / R.O. Gamzaev, M.V. Tkachuk // *Visnyk Nats. tech. HPI University: Coll. Science. etc. Topic. issue : Systems analysis, management and information technology*. - Kharkiv: NTU "KhPI". - 2013. - № 2 (976). - P. 49-59. [in Ukrainian]
3. Gamzaev R.O., Tkachuk M.V., Shevkoplyas D.O. Application of knowledge-oriented methods and technologies for modeling variability in the development of software product lines // *Materials int. scientific and technical conference KMNT-2021, (Kharkiv, April 23-25, 2021)* - Kh. : KhNU named after V. N. Karazina, 2021. - P. 104-107. [in Ukrainian]
4. Gamzayev R.O., Tkachuk M.V., Shevkopliias D.O. Handling of Expert Knowledge in Software Product Lines Development with Usage of Repertory Grids Method // *Visnyk of Kharkiv National University named after V.N. Karazina, Series "Mathematical modeling. Information Technology. Automated control systems "*. - № 47, 2020. - P. 13-24.
5. Martinkus I.O., Tkachuk M.V., Gamzaev R.O. Construction of software product lines using domain modeling and code reuse metrics / I.O. Martinkus, M.V. Tkachuk, R.O. Gamzaev // *Control, navigation and communication systems: coll. Science. etc. CNDI OU*. - K., 2017. - Vip. 3 (43) .– P. 93-97. [in Ukrainian]
6. Wes Felter, Alexandre Ferreira, Ram Rajamony, and Juan Rubio. An Updated Performance Comparison of Virtual Machines and Linux Containers // *IBM Research Report, RC25482 (AUS1407-001) July 21, 2014*.
1. Gamzaev R.O., Meta-model of the process of tracing requirements in software development / Gamzaev R.O., Tkachuk N.V., Martinkus I.O. // *Bulletin of NTU "KhPI". Series: New solutions in modern technologies*. - X: HTY «XIII», - 2014. - 26 (1069). - P.121-128. [in Russian]
7. Andon F.I. *Fundamentals of software systems quality engineering* / Andon F.I., Koval G.I., Korotun T.M. etc. - 2nd ed. - K. : Академперіодика, 2007 - 672с. [in Russian]
8. D. N. Jha, S. Garg, R. Ranjan et al. A Study on the Evaluation of HPC Microservices in Containerized Environment // *Concurrency and Computation Practice and Experience* 33(1) Published 2 May 2019.
9. Miguel G. Xavier; Marcelo V. Neves; Fabio D. Rossi; et al. Performance Evaluation of Container-based Virtualization for High Performance Computing Environments // *21st Euromicro Conference on Parallel, Distributed and Network-Based Processing, 2013*, pp. 233 – 240.
10. Saati, T.L. *Decision making. Hierarchy analysis method* / Saati TL; lane with English - M. : Radio and communication, 1993. - 278 p. [in Russian]
11. Amazon official website. - URL: <https://aws.amazon.com/ru/ec2/instance-types/> (accessed on 05.12.2021)

УДК 519.6:681.518.2:004.62:519.23

Методи виявлення викидів в пробних вибірках при управлінні процесами в системах за станом

О.Ю. Лихач, М.Л. Угрюмов, Д.О. Шевченко, С.І. Шматков

**Лихач
Олег Юрійович**

студент Харківського національного університету імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022;
e-mail: lykhach2018@gmail.com;
<https://orcid.org/0000-0002-4598-2912>.

**Угрюмов
Михайло Леонідович**

д.т.н., проф., професор кафедри ТПС; факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022
e-mail: ugrymov.mykhaylo52@gmail.com;
<https://orcid.org/0000-0003-0902-2735>

**Шевченко
Дмитро Олександрович**

студент Харківського національного університету імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022
e-mail: dimyich24@gmail.com;
<https://orcid.org/0000-0002-7897-250X>

**Шматков
Сергій Ігорович**

д.т.н., проф., завідувач кафедрою ТПС; факультету комп'ютерних наук; Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків-22, Україна, 61022;
e-mail: sershmat@gmail.com;
<https://orcid.org/0000-0002-0298-7174>.

Розроблене на сьогоднішній день інформаційне забезпечення не дозволяє з досить високим рівнем достовірності вирішувати завдання виявлення викидів у вибірках даних та часових рядах.

Тому, ця робота присвячена вибору метрик для оцінювання правильності виявлення викидів, найкращих математичних моделей та методів для вирішення проблеми виявлення викидів в пробних вибірках при управлінні процесами в системах за станом.

Були використані математичні моделі та методи виявлення викидів (аномальних значень) та програмні засоби на основі мови Python, такі як scikit-learn, Tensorflow, NumPy, Pandas і інші.

В ході виконання роботи було отримано: огляд метрик, які використовуються для оцінки ефективності математичних моделей та методів виявлення викидів; огляд традиційних методів та методів глибокого навчання для виявлення викидів; результати дослідження, щодо ефективності та якості математичних моделей і методів виявлення викидів, використовуючи 12 пробних вибірок; висновки про найкращу метрику та найкращі математичні моделі і методи для вирішення проблеми виявлення викидів в пробних вибірках при управлінні процесами в системах за станом.

Головним напрямком використання обраних методів є моніторинг рівня аномальних значень в різних вибірках при управлінні процесами в системах за станом, що робить ці методи універсальними для використання.

Ключові слова: виявлення викидів, машинне навчання, управління процесів, метрики оцінки якості, глибоке навчання

Anomaly detection methods in sample datasets when managing processes in systems by the state

Lykhach Oleh

student V.N. Kaeazyn National Universite, Svobody Sq 6, Kharkiv, Ukraine, 61022;
e-mail: lykhach2018@gmail.com;
<https://orcid.org/0000-0002-4598-2912>

- Ugryumov Mykhaylo** *Doctor of Technical Sciences, Professor, Professor of the Department of Theoretical&Applied Systems; Faculty of Computer Science; VN Kharkiv National University Karazina, Maidan Svobody, 4, Kharkiv-22, Ukraine, 61022;*
e-mail: ugryumov.mykhaylo52@gmail.com;
<https://orcid.org/0000-0003-0902-2735>
- Shevchenko Dmytro** *student V.N. Karazin National University, Svobody Sq 6, Kharkiv, Ukraine, 61022;*
e-mail: dimyich24@gmail.com;
<https://orcid.org/0000-0002-7897-250X>
- Shmatkov Sergei** *Doctor of Technical Sciences, Professor, Head of the Department of Theoretical&Applied; Faculty of Computer Science; VN Kharkiv National University Karazina, Maidan Svobody, 4, Kharkiv-22, Ukraine, 61022;*
e-mail: sershmat@gmail.com;
<https://orcid.org/0000-0002-0298-7174>

The current information software does not allow solving the problems of detecting outliers in data samples and time series with a sufficiently high level of reliability.

Therefore, this work is devoted to the choice of metrics for assessing the correctness of detecting outliers, as well as the best mathematical models and methods for solving the problem of detecting outliers in test samples when managing processes in systems by state. Mathematical models and methods for detecting outliers (anomalous values) and Python-based software tools such as scikit-learn, Tensorflow, NumPy, Pandas and others have been used.

The results of our work are the overview of the metrics used to assess the effectiveness of mathematical models and methods for detecting outliers; the overview of traditional and deep learning techniques of detecting outliers; the results of researching the efficiency and quality of mathematical models and methods for detecting outliers using 12 datasets; the conclusions about the best metric and the best mathematical models and methods for solving the problem of detecting outliers in test samples when managing processes in systems by state.

The selected methods are mainly used for monitoring the level of anomalous values in various datasets when managing processes in systems by state, which makes these methods universal.

Keywords: outlier detection, machine learning, process control, quality assessment metrics, deep learning

1 Постановка проблеми та її актуальність

Будемо розглядати в якості об'єкту дослідження управління процесів в системах за їх станом, наприклад, економічних систем, заснованих на даних моніторингу контрольованих змінних стану. Результати моніторингу – це вибірки даних та часові ряди. Вибірка даних представляє собою набір значень за певній проміжок часу, які можуть суттєво змінюватись в залежності від ситуації в системі. В той час як часові ряди являють собою сукупність вимірних значень змінних, одержуваних на певних інтервалах часу, що нерозривно примикають один до одного та протягом яких значення змінних істотно не змінюються. Часові ряди, будучи дискретною моделлю контролю стану динамічних систем, зазвичай містять параметричну невизначеність, є нестационарними і зашумленими.

При розв'язанні задачі виявлення викидів у вибірках даних та часових рядах потрібно попередньо обробити вхідні дані та видалити пропущені значення для того, щоб можна було використати методи виявлення викидів. Після того, як вхідні дані були оброблені та пропущені значення були видалені потрібно знайти викиди (також відомі, як аномалії) і видалити їх з наборів даних та часових рядів. Це дозволить підвищити рівень достовірності інформації та покращити управління процесів в системі.

Задача виявлення викидів в результаті її декомпозиції повинна бути представлена, як послідовність вирішення взаємопов'язаних задач таких як:

- моніторинг стану системи (вибір та вимірювання значень контрольованих змінних стану системи через певні проміжки часу);
- попередня обробка даних, яка призводить до приведення даних моніторингу до виду, придатного для виявлення викидів;

- знаходження аномальних значень у часових рядах та вибірках даних моніторингу системи.

Розгляду завдань теорії та практики управління процесів у динамічних системах приділяється велика увага як науковцям в Україні, так і за її межами. На цей час опубліковано безліч робіт, присвячених опису математичних моделей та методів виявлення викидів у процесах управління систем за станом [1-15].

Розглядаючи класифікацію методів (моделей) виявлення викидів в системах можна визначити, що їх можна поділити на традиційні методи та методи глибокого навчання. Класифікація цих моделей та методів виявлення викидів представлена на рис. 1.1-1.2.

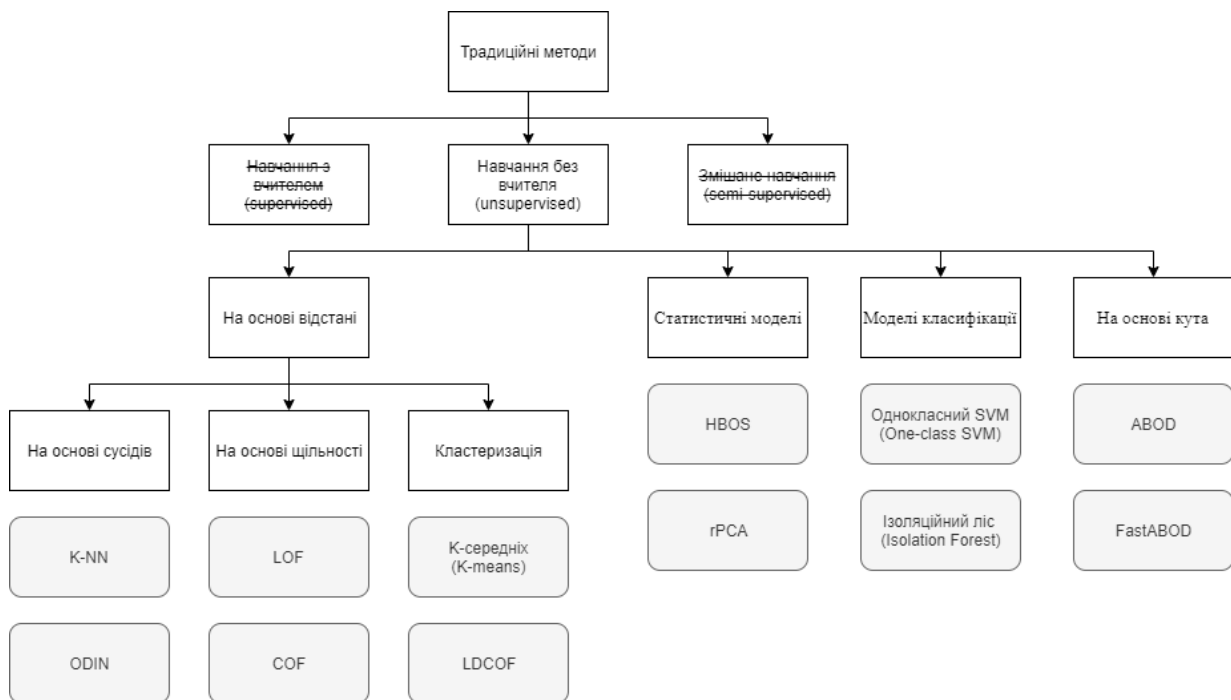


Рис. 1.1 Класифікація моделей та методів виявлення викидів

Можна виділити вищу ланку ієрархії моделей та методів виявлення викидів – це традиційні моделі та методи глибокого навчання (рис. 1.2). Серед них можна виділити чотири основних типи моделей: моделі на основі відстані, статистичні моделі, моделі класифікації та моделі на основі кута. Аналізуючи типи моделей для глибокого навчання можна виділити три типи навчання: глибоке навчання для вилучення функцій, навчання особливостей представлення нормальності та наскрізне навчання для визначення викидів.

До множини методів, моделей на основі відстані відносяться наступні: K-NN[16], ODIN [17], LOF [18] та K-means [19]. Серед статистичних моделей можна виділити: HBOS [20] та rPCA [21]. Прикладами моделей класифікації для виявлення викидів є: SVM [22] та Isolation Forest [23]. До методів, моделей побудованих на основі кута можна віднести ABOD [24] та FastABOD.

До множини методів, моделей на основі відстані відносяться наступні: K-NN[16], ODIN [17], LOF [18] та K-means [19]. Серед статистичних моделей можна виділити: HBOS [20] та rPCA [21]. Прикладами моделей класифікації для виявлення викидів є: SVM [22] та Isolation Forest [23]. До методів, моделей побудованих на основі кута можна віднести ABOD [24] та FastABOD.

Аналіз існуючих літературних джерел показує, при розробці математичних моделей та методів вирішення завдань виявлення викидів у вибірках даних та часових рядах виникає низка проблем:

- невизначеність вхідних даних (обмежений обсяг вибірок, наявність пропущених значень, корельованість змінних станів);
- велика розмірність множини змінних стану;
- невизначеність у виборі форми приведення вхідних даних до нормального вигляду, придатного для моделей виявлення викидів;
- невизначеність у виборі критеріїв якості математичних моделей та методів;
- невизначеність вибору правильних результатів рішень щодо аномальних значень.

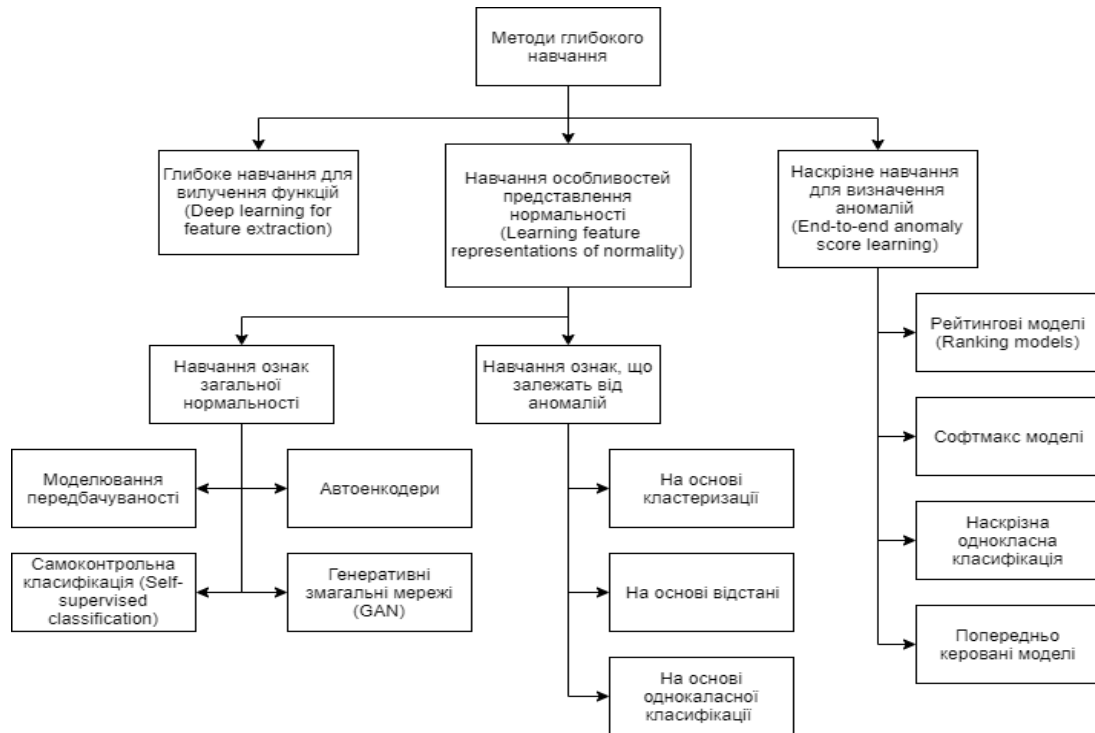


Рис. 1.2 Класифікація методів глибокого навчання для виявлення викидів

Слід зазначити, що у роботах, присвячених вирішенню завдань виявлення викидів, відсутнє визначення щодо найкращих серед існуючих математичної моделі або методу для виявлення аномалій та не враховується метрика для визначення точності цих моделей.

Розроблене на сьогоднішній день інформаційне забезпечення не дозволяє з досить високим рівнем достовірності вирішувати завдання виявлення викидів у вибірках даних та часових рядах.

Таким чином, виникає потреба у виборі метрик для оцінювання правильності виявлення викидів, найкращих математичних моделей, методів та засобів реалізації інформаційної технології при виявленні аномалій у часових рядах та вибірках даних при управлінні процесів в системах за їх станом.

Ця робота присвячена вибору метрик, найкращих математичних моделей та методів для вирішення проблеми виявлення викидів в пробних вибірках при управлінні процесами в системах за станом.

2 Постановка завдання виявлення викидів в пробних вибірках

Наділі будемо розглядати пробні вибірки в яких присутні аномалії (або викиди). Аномалії і викиди – це два терміни, які найчастіше використовуються в контексті виявлення аномалій; іноді взаємозамінні. Аномалії або викиди – це екземпляри даних, які виділяються і несхожі на інші

Необхідно отримати функціональну залежність, яка буде відображати зв'язок викидів, реальних даних та пробних вибірок. А також визначити якість цієї залежності за допомогою єдиної метрики.

Результатом вирішення задачі повинен бути математична модель (або метод) при використанні яких можливо отримати результати, які вказували би на приналежність певного спостереження до викидів, використовуючи лише дані із пробної вибірки.

3 Метрика для оцінювання якості виявлення викидів

Метрика в загальному сенсі машинного навчання – це еталон вимірювання якості математичного методу або моделі. Існує проблема у виборі єдиної метрики для визначення якості математичних методів, моделей виявлення викидів у пробних вибірках.

В даний час є велика кількість метрик, тому розглянемо популярні метрики, які підтвердили свою використовуваність за довгі роки у задачах виявлення викидів.

1. Точність (Ассигасу) – це найпростіший та інтуїтивно зрозумілий показник продуктивності класифікатора. Він розраховується по формулі 3.1, як відношення кількості правильно передбаченого класу до загальної кількості передбачень:

$$\text{Точність} = \frac{TP+TN}{T}, \quad (3.1)$$

де TP – кількість значень правильної класифікації позитивного класу, TN – кількість значень правильної класифікації негативного класу, T – загальна кількість передбачень.

Коли ми стикаємося з проблемою дисбалансу класів, точність є неправильною метрикою для використання. Наприклад, нехай існує 2 класи: клас А становить 99% набору даних, а клас В – це решта 1%. Якщо прогнозувати клас А кожного випадку, буде досягнута точність 99%. За метрикою точності можна вважати, що модель працює чудово, але насправді модель працює дуже погано.

2. Точність (Precision) і відклик (Recall). Загалом, існує компроміс між відкликом (відсоток дійсно позитивних випадків, які були класифіковані як такі) і точністю (відсоток позитивних класифікацій, які дійсно позитивні). Ці метрики можна розрахувати за формулами 3.2-3.3:

$$\text{Відклик} = \frac{TP}{TP+FN}, \quad (3.2)$$

де TP – кількість значень правильної класифікації позитивного класу, FN – кількість значень неправильної класифікації негативного класу.

$$\text{Точність} = \frac{TP}{TP+FP}, \quad (3.3)$$

де TP – кількість значень правильної класифікації позитивного класу, FP – кількість значень неправильної класифікації позитивного класу.

У ситуаціях, коли потрібно виявити екземпляри класу меншості, зазвичай більш важливою метрикою є відклик, ніж точність. Але, як сказано раніше, повинен бути компроміс між цими метриками.

3. Метрика F. Ця метрика використовується в тих випадках, коли потрібно досягнути високого значення точності (Precision) і відклику (Recall) та отримати лише один показник якості. F1 можна розрахувати за формулою:

$$F1 = \frac{2*precision*recall}{precision+recall}, \quad (3.4)$$

де precision – точність, recall – відклик.

Метрику F1 не можна коригувати в залежності від потреб класифікації, тому виник окремий випадок метрики, відомої як F-Beta, яка має корегувальний параметер β . Цей параметер надає змогу корегувати значення відклику та точності в такій залежності, що чим більше значення β тим більша залежність метрики від відклику і тим менша від точності. Дана метрика розраховується за формулою:

$$F_{\beta} = (1 + \beta^2) * \frac{precision*recall}{\beta^2*precision+recall}, \quad (3.5)$$

де precision – точність, recall – відклик, β – параметер корегування.

4. Метрика Карра. Карра або Cohen's Карра схожа на точність класифікації, за винятком того, що вона нормалізується на базовій лінії випадкових шансів у наборі даних. Метрика розраховується за формулою:

$$k = \frac{p_0 - p_e}{1 - p_e}, \quad (3.6)$$

де p_0 – це дотримана угода, p_e – це очікувана угода.

Ця метрика відображає наскільки краще працює класифікатор (p_0) порівняно з продуктивністю класифікатора, який просто вгадує випадковим чином відповідно до частоти кожного класу (p_e).

Стандартизованого способу інтерпретації його значень не існує. Ландіс і Кох [25] пропонують спосіб характеристики цінностей. Згідно з їхньою схемою, значення < 0 вказує на відсутність згоди, 0–0.20 як незначну, 0.21–0.40 як справедливу, 0.41–0.60 як помірну, 0.61–0.80 як суттєву та 0.81–1 як майже ідеальну згоду.

5. Метрика ROC-AUC. Крива ROC-AUC є вимірюванням продуктивності для проблем класифікації, виявлення викидів при різних порогових налаштуваннях. ROC – це крива ймовірності, а AUC – ступінь або міра відокремленості. Ця метрика відображує, наскільки математична модель здатна розрізнити класи. Чим вища міра AUC, тим краще модель правильно прогнозує класи.

Крива ROC зображена на рис. 3.1, де відображена залежність TPR (True Positive Rate) від FPR (False Positive Rate). По осі ординат відображена міра TPR, по осі абсцис — FPR. TPR – це частка правильних прогнозів у прогнозах позитивного класу. Ця міра розраховується за формулою:

$$TPR = \frac{TP}{TP+FN}, \quad (3.7)$$

де TP – кількість значень правильної класифікації позитивного класу, FN – кількість значень неправильної класифікації негативного класу.

FPR – це частка неправильних прогнозів у прогнозах позитивного класу. Ця міра розраховується за формулою :

$$FPR = \frac{FP}{FP+TN}, \quad (3.8)$$

де FP – кількість значень неправильної класифікації позитивного класу, TN – кількість значень правильної класифікації негативного класу.

Площа під кривою ROC визначає значення метрики ROC-AUC. Для відмінної моделі значення цієї метрики буде дорівнювати одиниці. Діагональна пряма, яка відображена на рис. 3.1, вказує на модель, яка не має змоги відрізнити позитивний клас від негативного, тобто не має можливості розділення класів.

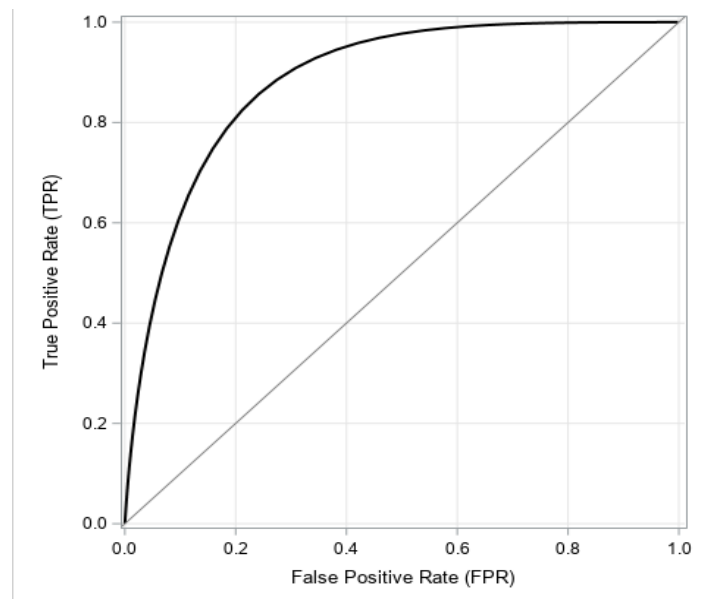


Рис. 3.1 Крива ROC

Цю метрику слід використовувати, коли модель повинна працювати однаково добре як на позитивному, так і на негативному класі.

6. Метрика PR-AUC. Ця метрика розраховується так само, як ROC-AUC. Відмінною рисою є те, що крива будується на основі точності (Precision) і відклику (Recall).

Цю метрику слід використовувати, коли модель повинна відмінно визначати або позитивні класи, або негативні. Тобто за допомогою PR-AUC можна сфокусуватися на правильному прогнозуванні одного із класів.

7. Метрика pAUC (Partial AUC). Часткова AUC була запропонована як альтернативний захід до стандартної AUC. При використанні часткової AUC враховується лише конкретна область простору ROC. Приклад кривої pAUC відображено на рис.3.2, де pAUC розраховується для затіненої області.

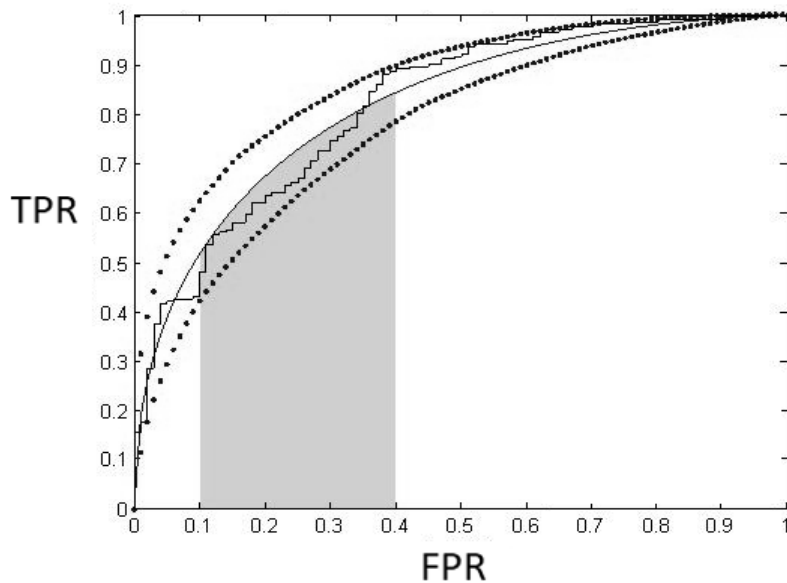


Рис. 3.2 Крива рAUC

8. Метрика двосторонній рAUC (Partial AUC). На відміну від рAUC, замість обмеження лише частоти помилкових позитивних результатів (FPR), двосторонній рAUC фокусується на частковій площі під кривою з обмеженнями як по горизонталі, так і по вертикалі.

Приклад кривої двостороннього рAUC відображено на рис.3.3. Двосторонній рAUC позначає площу заштрихованої області А. Ця заштрихована область безпосередньо визначається явною верхньою межею FPR ($p_0 = 0,5$) і нижньою межею TPR ($q_0 = 0,65$). На відміну від цього, рAUC позначає площу обох регіонів А і В.

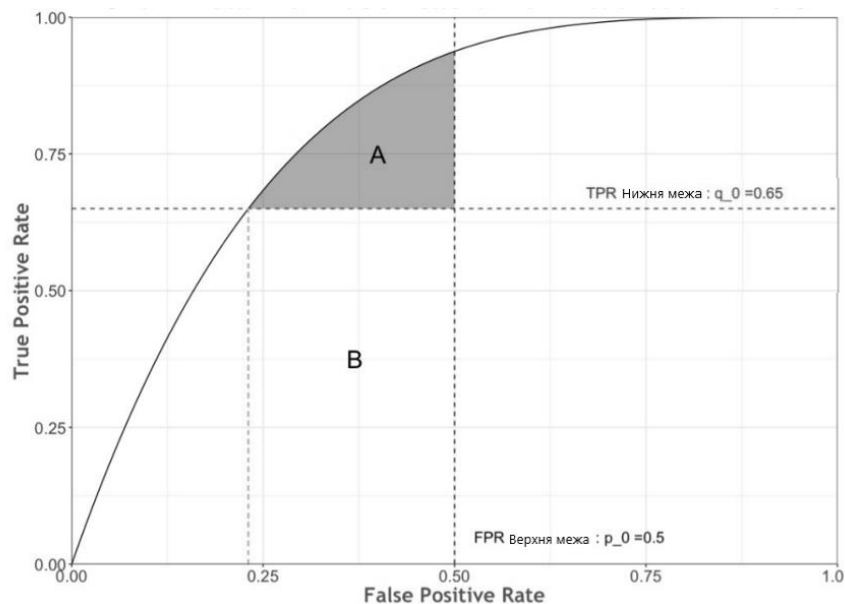


Рис. 3.3 Крива двостороннього рAUC

9. Метрика LogLoss. Вона є однією з найважливіших класифікаційних метрик на основі ймовірностей. Ця метрика розраховується за формулою 3.9. LogLoss вказує на те, наскільки ймовірність прогнозу близька до відповідного фактичного/істинного значення. Та чим більше прогнозована ймовірність відрізняється від фактичного значення, тим вище значення логарифмічних втрат.

$$\text{LogLoss} = -\frac{1}{n} \sum_{i=1}^n [y_i * \log(p_i) + (1 - y_i) * \log(1 - p_i)], \quad (3.9)$$

де y_i – справжній клас, p_i – вірогідність того, що y_i вказує на позитивний клас, n – загальна кількість спостережень з вибірки.

До недоліків цієї метрики можна віднести те, що при незбалансованості класів, мажоритарний клас може домінувати над LogLoss.

Проаналізувавши усі метрики, які використовуються при оцінці якості методів (моделей) виявлення викидів, можна визначити, що найкращою є метрика PR-AUC. Ця метрика була обрана найкращою, через те що вона фокусується на малих позитивних класах, в нашому випадку викидах та дозволяє об'єктивно оцінити якість математичних моделей та методів.

4 Методи виявлення викидів (аномальних значень)

4.1 Традиційні методи для виявлення викидів

Серед традиційних методів можна виділити такі як:

1) Z-оцінка (стандартна оцінка спостереження) – це індикатор, що визначає положення вихідної оцінки з погляду її відстані від середнього значення при вимірі в одиницях стандартного відхилення, при умові гаусовського розподілу.

Метод дозволяє побачити, наскільки вище або нижче середнього знаходиться це значення на кривій розподілу (рис. 4.1).

Це робить z-оцінку параметричним методом. Інколи точки даних не описуються гауссовим розподілом. Ця проблема може бути вирішена шляхом застосування перетворень до даних, таких як масштабування.

Під час обчислення z-оцінки для кожної вибірки в наборі даних необхідно вказати поріг. Аналізуючи точки даних, які лежать за певним порогом, можна визначити чи відносяться вони до аномальних.

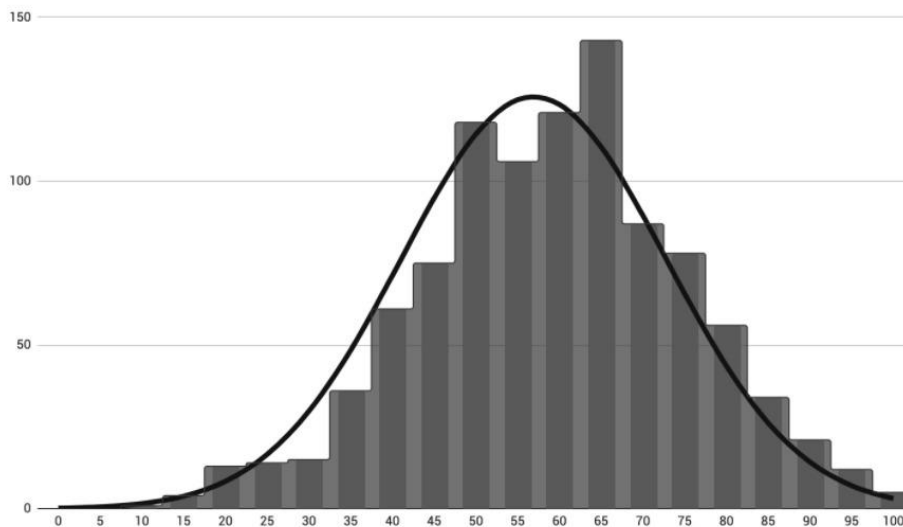


Рис. 4.1 Приклад роботи методу Z-оцінка [26]

Z-оцінку для будь-якої точки даних можливо розрахувати за формулою:

$$Z = \frac{x - \mu}{\sigma}, \quad (4.1)$$

де x – вхідний показник, μ - середнє значення набору даних, σ – стандартне відхилення для набору даних.

Даний метод ефективний та простий для виявлення викидів в наборі даних з параметричними розподілами в малорозмірному просторі об'єктів.

Визначимо переваги та недоліки методу Z-оцінки. До переваг можна віднести наступні:

- ефективний метод, щоб описати значення у просторі ознак за допомогою розподілу Гауса.
- проста реалізація.

Серед недоліків можна виділити наступні:

- метод ефективний у просторі об'єктів низької розмірності;
- якщо розподіли не можна вважати параметричними, метод працює не точно.

2) DBSCAN — це метод кластеризації на основі щільності. Цей метод зосереджений на пошуку сусідів за щільністю на n -вимірній сфері з радіусом ϵ (рис.4.2.). Кластер можна визначити як максимальний набір точок, пов'язаних із щільністю в просторі ознак.

Це ефективний метод для роботи з наборами даних середнього розміру. DBSCAN самостійно оцінює кількість кластерів, немає потреби вказувати кількість бажаних кластерів, це модель машинного навчання без вчителя.

DbSCAN визначає такі класи точок як:

- основна точка: A є основною точкою, якщо її околиця (визначена ϵ) містить принаймні стільки ж або більше точок, ніж параметр MinPts (мінімальна кількість спостережень у кластері);
- гранична точка: C — це гранична точка, яка лежить у кластері, і її околиці не містять більше точок, ніж MinPts , але вона все ще є досяжною для щільності іншими точками в кластері;
- викид: N — це точка викиду, яка не лежить у жодному кластері, і вона не пов'язана з будь-якою іншою точкою.

Таким чином, ця точка матиме власний кластер [27].

Визначимо переваги та недоліки даного методу. До переваг можна віднести наступні:

- точно працює, якщо простір ознак пошуку викидів є багатовимірним;
- легка візуалізація результатів.

Серед недоліків можна виділити наступні:

- значення у просторі функцій необхідно відповідно масштабувати;
- це неконтрольована модель, і її необхідно повторно калібрувати щоразу, коли аналізується новий пакет даних.

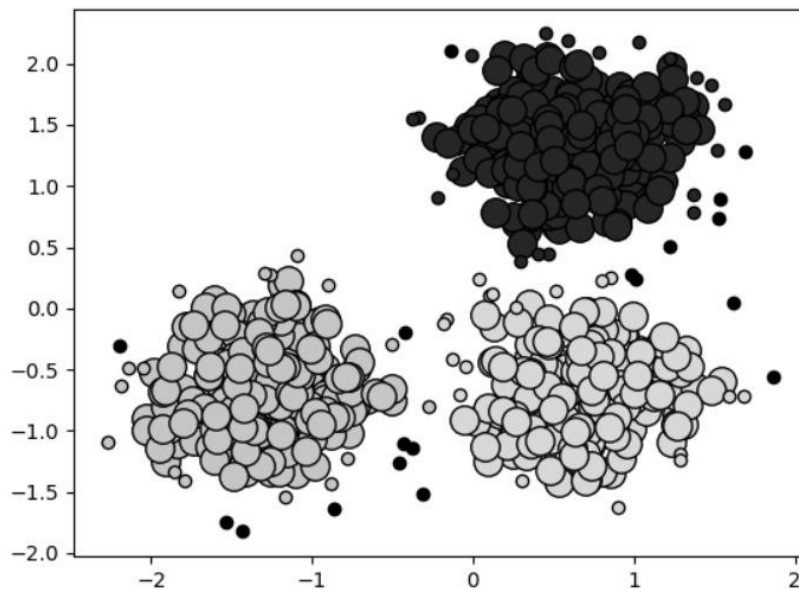


Рис. 4.2 Приклад роботи методу DBSCAN[28]

3) Ізоляційний ліс – ефективний метод для виявлення викидів (аномальних значень) в наборах даних. Цей метод заснований на бінарних деревах рішень. Основний принцип ізоляційного лісу полягає в тому, що викиди є нечисленними і далекими від решти спостережень.

Щоб побудувати дерево, алгоритм випадковим чином вибирає об'єкт із простору ознак і випадкове розділене значення в діапазоні між максимумами та мінімумами. Це робиться для всіх

спостережень у навчальному наборі. Для побудови лісу складається ансамбль дерев із усередненням усіх дерев у лісі.

Даний метод порівнює спостереження зі значенням розщеплення у вузлі, цей вузол матиме два дочірні вузла, на яких буде зроблено ще одне випадкове порівняння. Кількість розщеплень, зроблених алгоритмом для екземпляра, називається довжиною шляху. Викиди матимуть коротшу довжину шляху, ніж решта спостережень.

Оцінку аномалії можна обчислити за формулою:

$$S(x, n) = 2^{-\frac{E(h(x))}{e(n)}}, \quad (4.2)$$

де $E(h(x))$ – середня довжина шляху вибірки, $S(n)$ – невдалий пошук довжини, N – кількість зовнішніх вузлів.

Розглянемо переваги та недоліки методу. До переваг можна віднести відсутність необхідності масштабувати значення у просторі функцій.

Серед недоліків можна виділити наступні:

- складна візуалізація результатів;
- при неправильній оптимізації, для великого набору даних, довгий час навчання.

4) Local Outlier Factor (LOF) — метод машинного навчання без вчителя для виявлення аномалії, який обчислює локальне відхилення щільності точки щодо її сусідів. Даний метод вважає викидами зразки, які мають значно нижчу щільність, ніж їхні сусіди.

Кількість сусідів, що розглядаються зазвичай встановлюється:

- більшою, ніж мінімальна кількість вибірок, яку повинен містити кластер, так що інші вибірки можуть бути локальними викидами щодо цього кластера;
- менше, ніж максимальна кількість близьких сусідів за вибірками, які потенційно можуть бути локальними викидами [29].

LOF дає кращі результати, аніж глобальний підхід до пошуку викидів. Оскільки граничного значення LOF немає, вибір точки як викиду залежить від користувача.

Розглянемо переваги та недоліки методу. До переваг можна віднести, що точка буде вважатися викидом, якщо вона знаходиться на невеликій відстані від надзвичайно щільного скупчення, глобальний підхід може не розглядати цей момент як вибір. Але LOF може ефективно ідентифікувати локальні викиди.

Серед недоліків можна виділити наступні:

- немає певного порогового значення, вище якого точка визначається як викид;
- ідентифікація викиду залежить від проблеми та користувача.

4.2 Методи глибокого навчання для виявлення викидів (аномальних значень)

Серед методів глибокого навчання можна виділити наступні.

1) Автоенкодер - це нейронні мережі, призначені для вивчення низькорозмірного уявлення за деякими вхідними даними (рис. 4.3). Вони складаються з двох компонентів: кодувальника, який вчиться відображати вхідні дані в низькорозмірне уявлення (так зване вузьке місце), і декодер, який вчиться відображати це низькорозмірне уявлення назад у вихідні вхідні дані. Структуруючи проблему навчання таким чином, мережа кодувальника вивчає ефективну функцію стиснення, яка відображає вхідні дані в помітне уявлення нижчої розмірності, так що мережа декодера може успішно відновлювати вихідні вхідні дані.

Модель навчається шляхом мінімізації помилки відновлення, яка є різницею (середньоквадратичною помилкою) між вихідним введенням і відновленим висновком, створеним декодером [30].

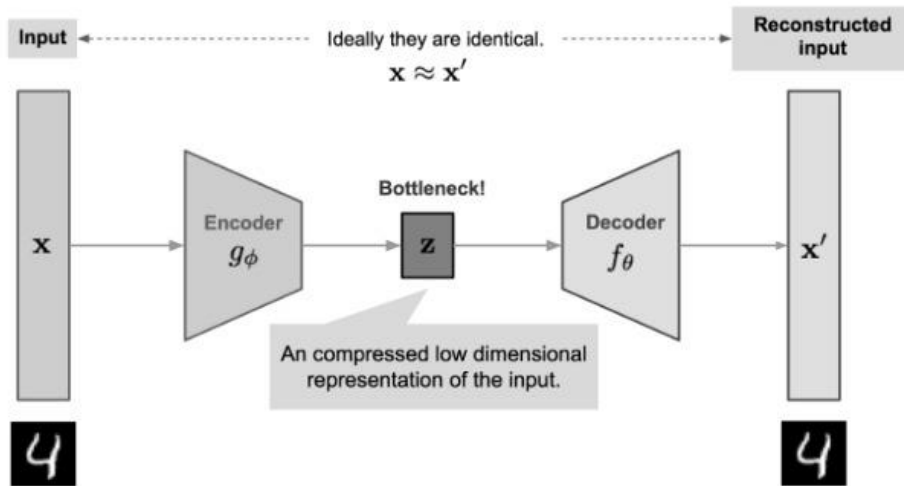


Рис. 4.3 Архітектура AutoEncoder[31]

Застосування автоенкодера для виявлення аномалій слідує загальному принципу: спочатку потрібно змодельовати нормальну поведінку, а потім генерувати оцінку аномалії кожної нової вибірки даних. Щоб змодельовати нормальну поведінку, слід використовувати напівконтрольований підхід, тобто коли навчання моделі проходить на нормальних вибірках даних. Таким чином модель вивчає функцію відображення, яка успішно відновлює нормальні вибірки даних з дуже невеликою помилкою відновлення. Така поведінка відтворюється під час тестування, коли помилка відновлення мала для нормальних вибірок даних та велика для аномальних вибірок даних.

Щоб ідентифікувати аномалії, використовується обробка помилки відновлення як оцінка аномалії, після цього можливо вибрати зразки з помилками відновлення, що перевищують заданий поріг.

Розглянемо переваги та недоліки методу. До переваг можна віднести наступні:

- компактність та швидкість кодування;
- зменшення розмірності даних, що прискорює навчання моделі.

Серед недоліків можна виділити такі як:

- складна візуалізація результатів;
- складне навчання.

2) Варіаційний автоенкодер (VAE) - це розширення автоенкодера (рис. 4.4). Подібно до автоенкодера, він складається з кодувальника та мережевого компонента декодера, але він включає важливі зміни в структурі завдання навчання, щоб пристосуватися до варіаційного висновку.

На відміну від навчання зіставлення вхідних даних з фіксованим вектором вузького місця (точкова оцінка), VAE вивчає зіставлення вхідних даних з розподілом і вчиться відновлювати вихідні дані шляхом вибірки цього розподілу з використанням прихованого коду.

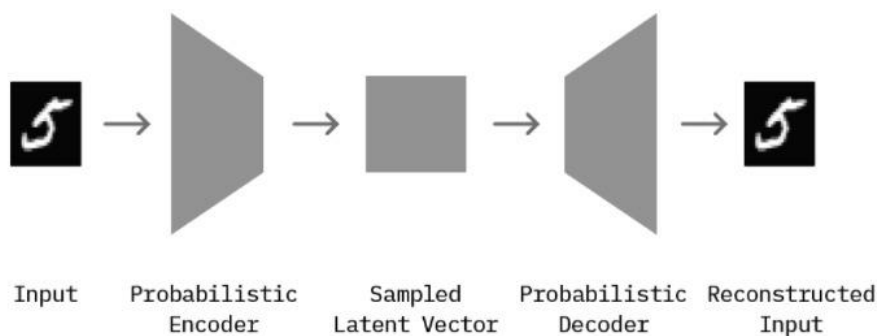


Рис. 4.4 Модель варіаційного автоенкодера [31]

Модель VAE навчається шляхом мінімізації різниці між розрахунковим розподілом, створеним моделлю, та реальним розподілом даних. Ця різниця оцінюється за допомогою дивергенції Кульбака-Лейблера, яка кількісно визначає відстань між двома розподілами, вимірюючи, скільки інформації втрачається, коли один розподіл використовується для подання іншого.

Використання VAE для виявлення аномалій: подібно автоенкодеру, навчання VAE починається на нормальних вибірках даних. Під час тестування можна визначити оцінку аномалії двома способами. По-перше, вилучити зразки прихованого коду з кодера з урахуванням наших вхідних даних, відстежити відновлені значення з декодера та обчислити середню помилку відновлення. Аномалії позначаються з урахуванням порогу помилки відновлення.

В якості альтернативи потрібно вивести середнє значення та параметр дисперсії з декодера та обчислити ймовірність того, що нова точка даних належить розподілу нормальних даних, на якому була навчена модель. Якщо точка даних знаходиться в області низької щільності (нижче деякого порога), це позначається як аномалія (рис. 4.5).

Розглянемо переваги та недоліки методу. До переваг можна віднести наступні:

- чіткий спосіб оцінки якості моделі (логарифмічна ймовірність);
- легка візуалізація.

Серед недоліків можна виділи такі як:

- неоптимальні фактори варіації;
- може мати градієнти високої дисперсії.

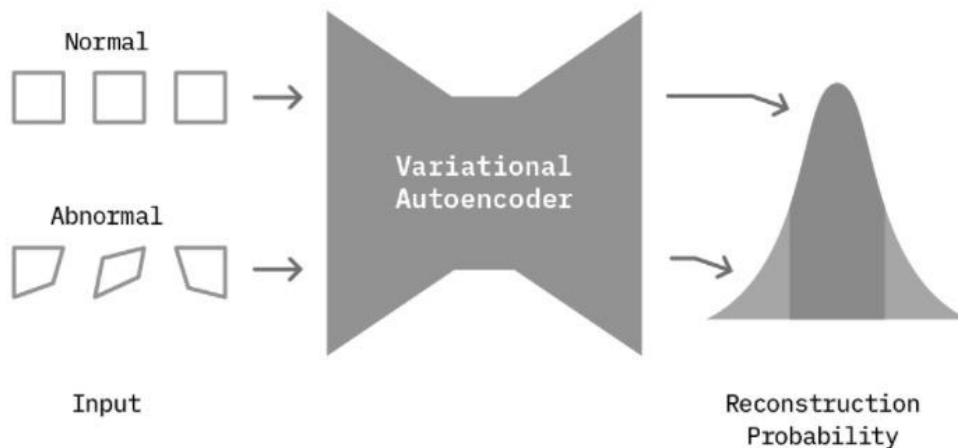


Рис.4.5 Оцінка аномалій за допомогою VAE. [31]

3) Генеративні змагальні мережі (GAN) – це нейронні мережі, розроблені для вивчення генеративної моделі розподілу вхідних даних (рис. 4.6). У класичному формулюванні вони складаються з пари (зазвичай з прямим зв'язком) нейронних мереж, а саме з генератора G і дискримінатора D. Обидві мережі навчаються спільно з кінцевою метою визначити розподіл вихідних даних X.

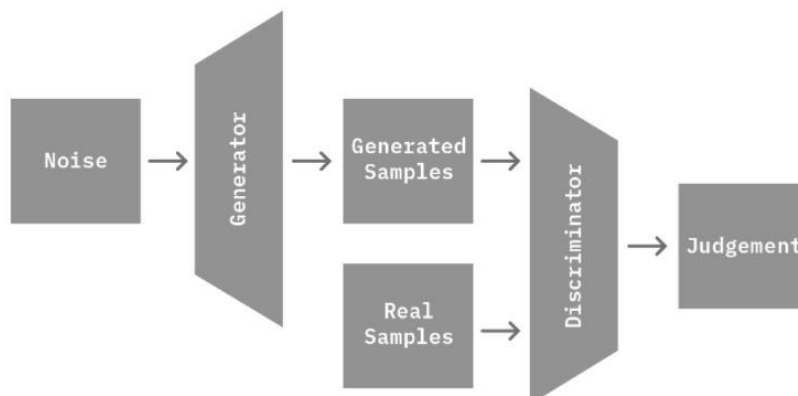


Рис. 4.6 Класична модель GAN [31]

Щоб ідентифікувати аномалії, використовується напівкерований підхід, тобто потрібно навчати модель від послідовності до послідовності на нормальних даних. Моделі послідовність-послідовність (sequence-to-sequence) - це клас нейронних мереж, в основному призначений для вивчення зіставлень між даними, які найкраще представлені у вигляді послідовностей. Потім під час тестування можна порівняти різницю (середньоквадратичну помилку) між вихідною послідовністю, згенерованою моделлю, та її вхідними даними. Це значення використовується як показник аномалій.

Розглянемо переваги та недоліки методу. Недоліком генеративних змагальних мереж є складність оцінки моделі. До переваг можна віднести наступні:

- GAN генерує дані, схожі вхідні;
- GAN деталізує дані та може легко інтерпретуватися у різні версії.

5. Огляд результатів та ефективності методів

В якості наборів даних були обрані різні вибірки, які використовуються для оцінки якості методів виявлення викидів, наприклад ForestCover, Satellite та інші. Також було створено власний набір даних, який буде позначатися, як «Econometrical (наш)». Цей набір даних відображає соціально-економічний розвиток країн за 2012-2020 роки. Він представляє 115 альтернатив з 32 змінними стану. Набір даних відображено на рис. 5.1.

Country Name	EGI_2020	EPI_2020	OSI_2020	HCI_2020	TII_2020	EGI_2018	EPI_2018	OSI_2018	HCI_2018	TII_2018	...	ICT_16	ICT_15	ICT_13	ICT_12	SPI_20	SPI_18	SPI_17	SPI_16
Albania	73.99	84.52	84.12	80.01	57.85	65.19	75.84	73.61	78.77	43.18	...	4.90	4.73	4.62	4.42	75.41	71.77	71.65	70.
Algeria	51.73	15.48	27.65	69.66	57.87	42.27	20.22	21.53	66.40	38.89	...	4.32	3.71	4.46	4.22	69.92	66.83	66.83	65.
Argentina	82.79	85.71	84.71	91.00	72.65	73.35	62.36	75.00	85.79	59.27	...	6.68	6.40	6.62	6.38	80.66	74.98	74.84	74.
Armenia	71.36	75.00	70.00	78.72	65.36	59.44	56.74	56.25	75.47	46.60	...	5.56	5.32	5.64	5.55	76.46	70.87	70.53	69.
Australia	94.32	96.43	94.71	100.00	88.25	90.53	98.31	97.22	100.00	74.36	...	8.08	8.29	8.23	8.14	91.29	88.32	88.23	87.
...
United States of America	92.97	100.00	94.71	92.39	91.82	87.69	98.31	98.61	88.83	75.64	...	8.13	8.19	7.78	7.75	85.71	84.78	85.27	86.
Uruguay	85.00	85.71	84.12	85.14	85.74	78.58	91.57	88.89	77.19	69.67	...	6.75	6.70	7.05	6.80	82.99	79.40	NaN	N.
Viet Nam	66.67	70.24	65.29	67.79	66.94	59.31	69.10	73.61	65.43	38.90	...	4.18	4.28	4.48	4.39	68.85	NaN	NaN	N.
Zambia	42.42	30.95	25.88	67.45	33.94	41.11	39.89	47.92	56.89	18.53	...	2.19	2.04	2.68	2.63	55.34	NaN	NaN	N.
Zimbabwe	50.19	45.24	52.35	61.35	36.88	36.92	27.53	32.64	56.68	21.44	...	2.85	2.90	3.12	2.99	52.26	45.26	43.76	42.

Рис. 5.1 Набір даних «Econometrical (наш)»

Було проведено дослідження щодо ефективності традиційних математичних моделей та методів для виявлення викидів. В якості показника ефективності було значення обраної метрики PR-AUC.

Для тестування та оцінки якості було розроблено програмне забезпечення, яке дозволяє:

- виконувати тренування математичних моделей та методів;
- кластеризувати вибірки на різних наборах даних;
- розраховувати показник ефективності.

Програмне забезпечення розроблено за допомогою мови Python. Також були використані наступні бібліотеки: scikit-learn, NumPy, Pandas і інші.

В якості наборів даних були використані різні вибірки, які були створені саме для оцінки якості математичних моделей та методів виявлення викидів (аномальних значень).

Дослідження якості проводилося таким чином, що набори даних подавалися до програмного забезпечення, яке в свою чергу виконувало усі необхідні дії з ними. В результаті роботи програми видавалися результати виявлення викидів та значення метрики PR-AUC.

Результати оцінки якості виявлення викидів для різних наборів даних приведені в таблиці 1.

Таблиця 1. Результати тестування традиційних методів для виявлення викидів

Набір даних	PR-AUC			
	Ізоляційний ліс	DBSCAN	LOF	Z-оцінка
Http (KDDCUP99)	0.90	0.60	0.45	0.50
ForestCover	0.85	0.83	0.72	0.60
Mulcross	0.89	0.33	0.37	0.41
Smtп (KDDCUP99)	0.83	0.80	0.65	0.60
Shuttle	0.81	0.60	0.55	0.67
Mammography	0.86	0.77	0.67	0.65
Satellite	0.82	0.68	0.72	0.66
Pima	0.71	0.65	0.52	0.60
Breastw	0.67	0.71	0.49	0.85
Arrhythmia	0.87	0.86	0.37	0.41
Ionosphere	0.80	0.78	0.73	0.63
Economical (наш)	0.84	0.73	0.65	0.54

Також було проведено дослідження щодо ефективності математичних моделей та методів глибокого навчання для виявлення викидів. Це дослідження є подібним до того, що проводилося для оцінки якості традиційних методів виявлення викидів. Відмінність полягає тільки у різних математичних моделях і методах, що оцінюються. Тому, було проведена імплементація автоенкодера, варіаційного автоенкодера та генеративних змагальних мереж. Головною бібліотекою для їх реалізації була Tensorflow, яка є комплексною платформою з відкритим вхідним кодом для машинного навчання.

Результати дослідження оцінки якості виявлення викидів (аномальних значень) для визначених наборів даних приведені в таблиці 2.

Таблиця 2. Результати тестування методів глибокого навчання для виявлення викидів

Набір даних	PR-AUC		
	Автоенкодер	Варіаційний автоенкодер	GAN
Http (KDDCUP99)	0.90	0.97	0.95
ForestCover	0.88	0.93	0.97
Mulcross	0.92	0.95	0.76
Smtп (KDDCUP99)	0.89	0.91	0.90
Shuttle	0.93	0.95	0.94
Mammography	0.87	0.89	0.90
Satellite	0.85	0.87	0.88
Pima	0.81	0.84	0.86
Breastw	0.75	0.77	0.79
Arrhythmia	0.89	0.90	0.93
Ionosphere	0.85	0.86	0.90
Economical (наш)	0.88	0.90	0.86

Вибираючи найкращі методи серед традиційних та глибокого навчання, не важко бачити, що кращими є ізоляційний ліс та метод, що базується на генеративних змагальних мережах (GAN).

Побудуємо графіки ящиків з вусами для кожної змінної стану із вхідного набору даних та результуючого набору, де викиди були видалені за допомогою найкращих методів. В якості вхідної вибірки буде набір даних «Economical (наш)». Графік ящиків з вусами для вхідної вибірки представлено на рис. 5.2.

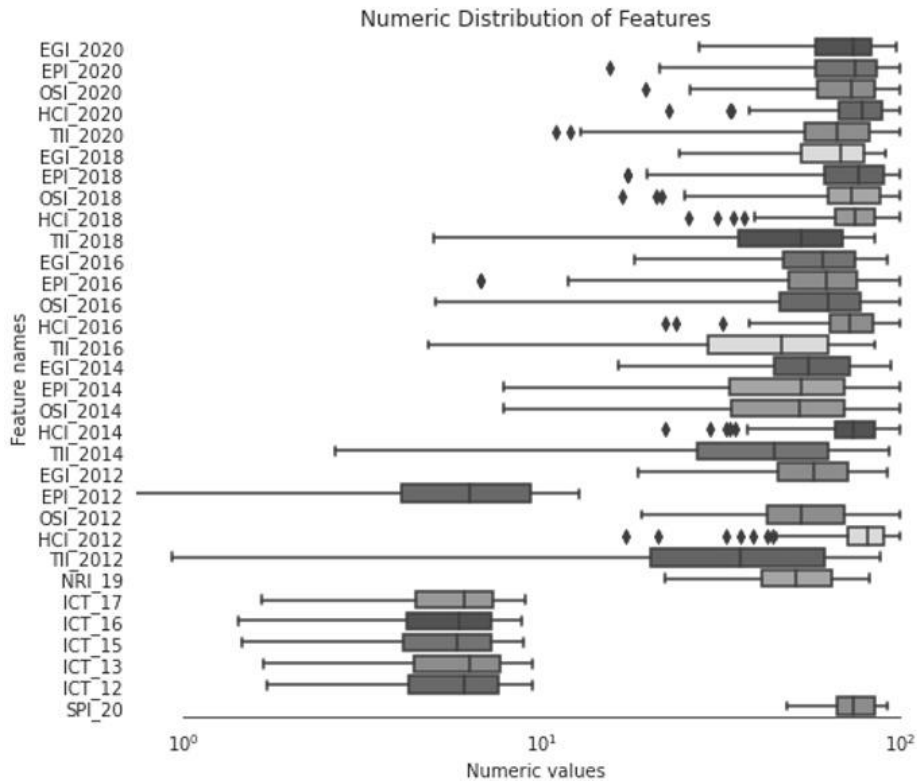


Рис. 5.2 Графік ящиків з вусами для змінних вхідної вибірки

На рис 5.2 можна побачити, що деякі змінні стану мають викиди, які відображені точками на графіку ящиків з вусами. Використаємо ізоляційний ліс для виявлення та видалення викидів з вхідної вибірки і побудуємо графік ящиків з вусами для набору даних, де викиди були видалені за допомогою ізоляційного лісу. Цей графік відображено на рис. 5.3.

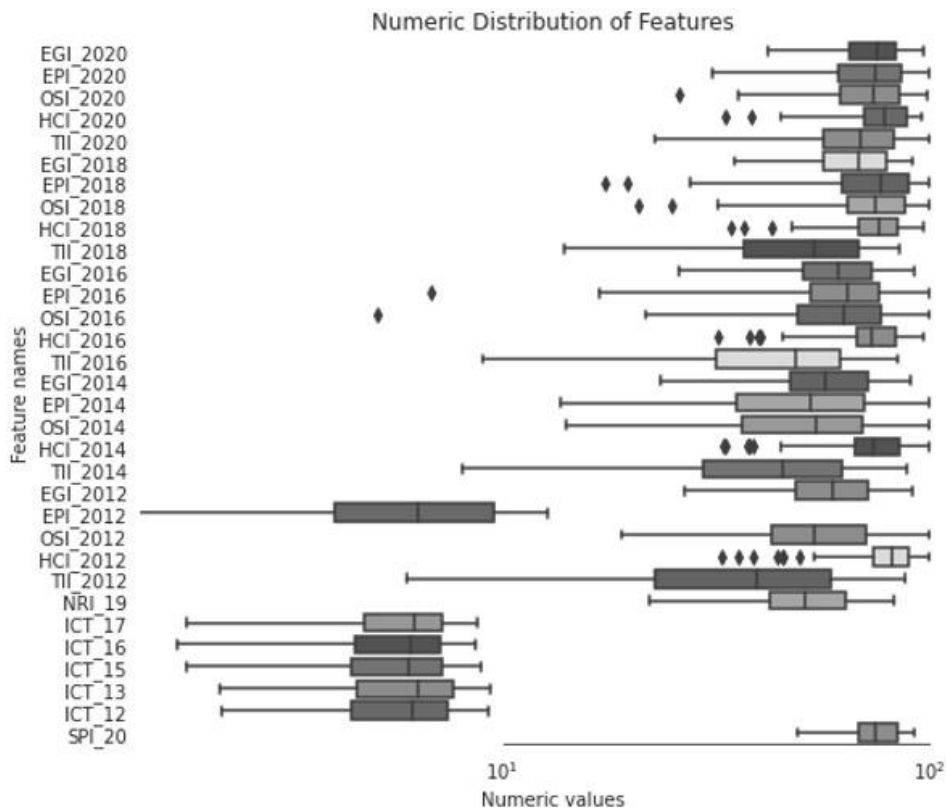


Рис. 5.3 Графік ящиків з вусами для набору даних, де викиди були видалені за допомогою ізоляційного лісу

Також використаємо генеративні змагальні мережі для виявлення та видалення викидів з вхідної вибірки і побудуємо графік ящиків з вусами для набору даних, де викиди були видалені за генеративних змагальних мереж. Цей графік відображено на рис. 5.4.

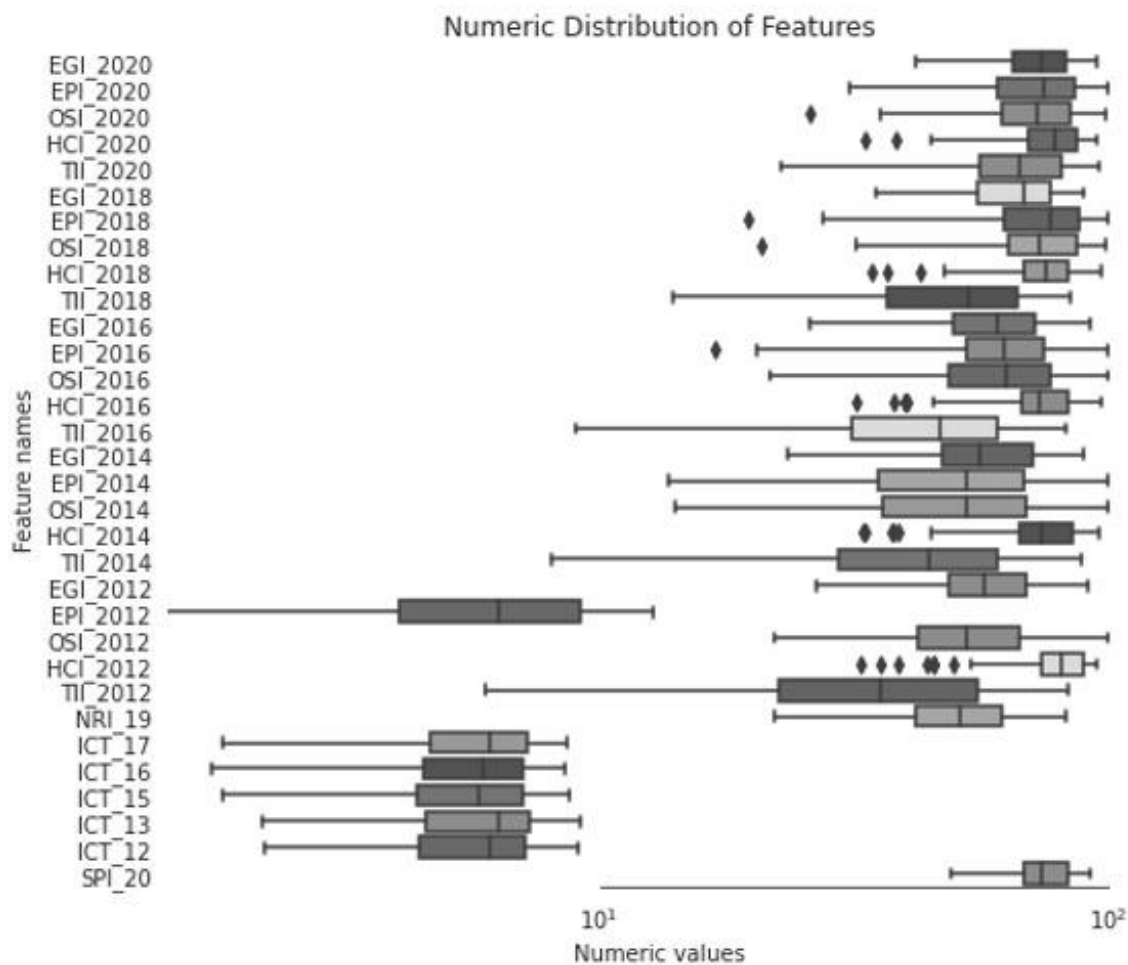


Рис. 5.4 Графік ящиків з вусами для набору даних, де викиди були видалені за допомогою генеративних змагальних мереж

6 Результати та висновки

Проведено аналіз метрик, які використовуються при оцінці якості математичних моделей та методів виявлення викидів. Визначено, що найкращою є метрика PR-AUC. Ця метрика була обрана найкращою, через те що вона фокусується на малих позитивних класах, в нашому випадку викидах, та дозволяє об'єктивно оцінити якість математичних моделей та методів.

Виконано класифікацію методів (моделей) виявлення викидів. Ці методи поділяються на традиційні та глибокого навчання. На основі аналізу результатів дослідження щодо ефективності традиційних математичних моделей та методів і аналізу їх переваг та недоліків, було визначено, що найбільш ефективним є метод «ізоляційний ліс». Цей метод здобув найкращі результати показників ефективності. Перевагою методу є відсутність необхідності масштабувати дані в просторі та відсутність великої кількості параметрів, що робить його надійним та простим в підборі параметрів для покращення роботи методу.

Серед математичних моделей та методів глибокого навчання для виявлення викидів було визначено, що найбільш ефективним є метод, що базується на генеративних змагальних мережах (GAN). Він досяг найкращих показників ефективності серед усіх математичних моделей та методів виявлення викидів, які були розглянуті в роботі, використовуючи різні набори даних. Перевагами даного методу є те, що метод деталізує дані та може легко інтерпретуватися у різні версії. Цей метод потребує великі обчислювальні можливості, тому його слід використовувати при їх наявності.

Обираючи між ізоляційним лісом та GAN слід підкреслити, що обидва методи мають досить високі показники ефективності виявлення викидів на пробних вибірках. Вибір між ними залежить від обчислювальних можливостей та поставлених цілей, щодо якості виявлення викидів.

Таким чином, в роботі було отримано:

- огляд метрик, які використовуються для оцінки ефективності математичних моделей та методів виявлення викидів;
- огляд традиційних методів та методів глибокого навчання для виявлення викидів;
- результати дослідження, щодо ефективності та якості математичних моделей і методів виявлення викидів, використовуючи 12 пробних вибірок;
- висновки про найкращу метрику та найкращі математичні моделі і методи для вирішення проблеми виявлення викидів в пробних вибірках при управлінні процесами в системах за станом.

ЛІТЕРАТУРА

1. В.П. Шкодірєв, К.І. Ягафаров, В.А. Баштовенко, Є.Е. Ільїна. Огляд методів виявлення аномалій в потоках даних. URL: http://ceur-ws.org/Vol-1864/paper_33.pdf (дата звернення: 10. 11. 2021).
2. М.В. Ломоносова. Виявлення аномалій у роботі механізмів методами машинного навчання. URL: <http://ceur-ws.org/Vol-2022/paper59.pdf> (дата звернення: 10. 11. 2021).
3. Chalapathy R., Chawla S. Deep Learning for Anomaly Detection: A Survey. URL: <https://arxiv.org/abs/1901.03407> (дата звернення: 10. 11. 2021)
4. Srikanth Thudumu, Philip Branch, Jiong Jin & Jugdutt (Jack) Singh. A comprehensive survey of anomaly detection techniques for high dimensional big data.
5. Deep Learning for Anomaly Detection: A Review: ACM Computing Surveys: Vol 54, No 2. URL: <https://dl.acm.org/doi/10.1145/3439950> (дата звернення: 10. 11. 2021).
6. Muruti G., Rahim F., bin Ibrahim Z. A Survey on Anomalies Detection Techniques and Measurement Methods // 2018 IEEE Conference on Application, Information and Network Security (AINS). 2018.
7. Shikha Agrawal, JitendraAgrawal. Survey on Anomaly Detection using Data Mining Techniques.
8. Pang G. и др. Deep Learning for Anomaly Detection // ACM Computing Surveys. 2021. Т. 54. № 2. С. 1-38.
9. Nassif A. и др. Machine Learning for Anomaly Detection: A Systematic Review // IEEE Access. 2021. Т. 9. С. 78658-78700.
10. Izhak Golan, Ran El-Yaniv. Deep Anomaly Detection Using Geometric Transformations. URL: <https://proceedings.neurips.cc/paper/2018/file/5e62d03aec0d17facfc5355dd90d441c-Paper.pdf> (дата звернення: 10. 11. 2021).
11. Mohammad Braei, Sebastian Wagner. Anomaly Detection in Univariate Time-series: A Survey on the State-of-the-Art. URL: <https://www.semanticscholar.org/paper/Anomaly-Detection-in-Univariate-Time-series%3A-A-on-Braei-Wagner/cf45bce52cca1f6e450ddaa1d19fe6e30661dffb> (дата звернення: 10. 11. 2021).
12. Atiq ur Rehman & Samir Brahim Belhaouari. Unsupervised outlier detection in multidimensional data
13. Victoria J. Hodge and Jim Austin. A Survey of Outlier Detection Methodologies. URL: <https://core.ac.uk/download/pdf/58585.pdf> (дата звернення: 12. 11. 2021).
14. Karanjit Singh and Dr. Shuchita Upadhyaya. Outlier Detection: Applications And Techniques. URL: https://www.researchgate.net/publication/267964435_Outlier_Detection_Applications_And_Techniques (дата звернення: 12. 11. 2021).
15. Wang S. и др. Effective End-to-end Unsupervised Outlier Detection via Inlier Priority of Discriminative Network. URL:

<https://proceedings.neurips.cc/paper/2019/hash/6c4bb406b3e7cd5447f7a76fd7008806-Abstract.html>
(дата звернення: 12. 11. 2021).

16. Karanjit Singh and Dr. Shuchita Upadhyaya. Outlier Detection: Applications And Techniques. URL: https://www.researchgate.net/publication/228686398_k-Nearest_neighbour_classifiers (дата звернення: 14. 11. 2021).

17. Yen-Chang Hsu, Yilin Shen, Hongxia Jin, Zsolt Kira. Generalized ODIN: Detecting Out-of-distribution Image without Learning from Out-of-distribution Data. URL: https://openaccess.thecvf.com/content_CVPR_2020/papers/Hsu_Generalized_ODIN_Detecting_Out-of-Distribution_Image_Without_Learning_From_Out-of-Distribution_Data_CVPR_2020_paper.pdf
(дата звернення: 14. 11. 2021).

18. Breunig M. и др. LOF // Proceedings of the 2000 ACM SIGMOD international conference on Management of data - SIGMOD '00. 2000. URL: https://www.researchgate.net/publication/221214719_LOF_Identifying_Density-Based_Local_Outliers
(дата звернення: 15. 11. 2021).

19. Na S., Xumin L., Yong G. Research on k-means Clustering Algorithm: An Improved k-means Clustering Algorithm // 2010 Third International Symposium on Intelligent Information Technology and Security Informatics. 2010.

20. Markus Goldstein, Andreas Dengel. Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm. URL: https://www.researchgate.net/publication/231614824_Histogram-based_Outlier_Score_HBOS_A_fast_Unsupervised_Anomaly_Detection_Algorithm (дата звернення: 15. 11. 2021).

21. Warp-core. URL: https://workday.github.io/warp-core/contents/anomaly_detection/ (дата звернення: 17. 11. 2021).

22. PDF) Support Vector Machines: Theory and Applications. URL: https://www.researchgate.net/publication/221621494_Support_Vector_Machines_Theory_and_Applications (дата звернення: 17. 11. 2021).

23. Fei Tony Liu, Kai Ming Ting Gipspsland School of Information Technology Monash University, Victoria, Australia. Isolation Forest. URL: https://www.researchgate.net/publication/224384174_Isolation_Forest (дата звернення: 18. 11. 2021).

24. Xuehui Wang, Yong Zhang, Hao Liu, Yang Wang, Lichun Wang, and Baocai Yin. An Improved Robust Principal Component Analysis Model for Anomalies Detection of Subway Passenger Flow.

25. JR L., GG K. The measurement of observer agreement for categorical data. URL: <https://pubmed.ncbi.nlm.nih.gov/843571/> (дата звернення: 18. 11. 2021).

26. Study Finance. URL: <https://studyfinance.com/static/media/z-score.png> (дата звернення: 21. 11. 2021).

27. A Brief Overview of Outlier Detection Techniques .URL: <https://towardsdatascience.com/a-brief-overview-of-outlier-detection-techniques-1e0b2c19e561> (дата звернення: 21. 11. 2021).

28. Demo of DBSCAN clustering algorithm. URL: https://scikit-learn.org/stable/_images/sphx_glr_plot_dbscan_001.png (дата звернення: 22. 11. 2021).

29. Outlier detection with Local Outlier Factor (LOF). URL: [https://scikit-learn.org/stable/auto_examples/neighbors/plot_lof_outlier_detection.html#:~:text=The%20Local%20Outlier%20Factor%20\(LOF,lower%20density%20than%20their%20neighbors.](https://scikit-learn.org/stable/auto_examples/neighbors/plot_lof_outlier_detection.html#:~:text=The%20Local%20Outlier%20Factor%20(LOF,lower%20density%20than%20their%20neighbors.) (дата звернення: 22. 11. 2021).

30. Anomaly Detection using Autoencoders. URL: <https://towardsdatascience.com/anomaly-detection-using-autoencoders-5b032178a1ea> (дата звернення: 22. 11. 2021).

31. Cloudera Fast Forward. Deep Learning for Anomaly Detection. URL: <https://ff12.fastforwardlabs.com/> (дата звернення: 24. 11. 2021)

REFERENCES

1. V.P. Shkodyrev, K.I. Yafagorov, B.A. Bashtovenko, Y.E. Ilyina. Review of methods for detecting anomalies in data streams. URL: http://ceur-ws.org/Vol-1864/paper_33.pdf (Last accessed: 10. 11. 2021). [in Russian]
2. M.V. Lomonosov. Detection of anomalies in the work of mechanisms by machine learning methods. URL: <http://ceur-ws.org/Vol-2022/paper59.pdf> (Last accessed: 10. 11. 2021). [in Russian]
3. Chalapathy R., Chawla S. Deep Learning for Anomaly Detection: A Survey. URL: <https://arxiv.org/abs/1901.03407> (Last accessed: 10. 11. 2021)
4. Srikanth Thudumu, Philip Branch, Jiong Jin & Jugdutt (Jack) Singh. A comprehensive survey of anomaly detection techniques for high dimensional big data.
5. Deep Learning for Anomaly Detection: A Review: ACM Computing Surveys: Vol 54, No 2. URL: <https://dl.acm.org/doi/10.1145/3439950> (Last accessed: 10. 11. 2021).
6. Muruti G., Rahim F., bin Ibrahim Z. A Survey on Anomalies Detection Techniques and Measurement Methods // 2018 IEEE Conference on Application, Information and Network Security (AINS). 2018.
7. Shikha Agrawal, Jitendra Agrawal. Survey on Anomaly Detection using Data Mining Techniques.
8. Pang G. Deep Learning for Anomaly Detection // ACM Computing Surveys. 2021. T. 54. № 2. C. 1-38.
9. Nassif A. Machine Learning for Anomaly Detection: A Systematic Review // IEEE Access. 2021. T. 9. C. 78658-78700.
10. Izhak Golan, Ran El-Yaniv. Deep Anomaly Detection Using Geometric Transformations. URL: <https://proceedings.neurips.cc/paper/2018/file/5e62d03aec0d17facfc5355dd90d441c-Paper.pdf> (Last accessed: 10. 11. 2021).
11. Mohammad Braei, Sebastian Wagner. Anomaly Detection in Univariate Time-series: A Survey on the State-of-the-Art. URL: <https://www.semanticscholar.org/paper/Anomaly-Detection-in-Univariate-Time-series%3A-A-on-Braei-Wagner/cf45bce52cca1f6e450ddaa1d19fe6e30661dffb> (Last accessed: 10. 11. 2021).
12. Atiq ur Rehman & Samir Brahim Belhaouari. Unsupervised outlier detection in multidimensional data
13. Victoria J. Hodge and Jim Austin. A Survey of Outlier Detection Methodologies. URL: <https://core.ac.uk/download/pdf/58585.pdf> (Last accessed: 12. 11. 2021).
14. Karanjit Singh and Dr. Shuchita Upadhyaya. Outlier Detection: Applications And Techniques. URL: https://www.researchgate.net/publication/267964435_Outlier_Detection_Applications_And_Techniques (Last accessed: 12. 11. 2021).
15. Wang S. и др. Effective End-to-end Unsupervised Outlier Detection via Inlier Priority of Discriminative Network. URL: <https://proceedings.neurips.cc/paper/2019/hash/6c4bb406b3e7cd5447f7a76fd7008806-Abstract.html> (Last accessed: 12. 11. 2021).
16. Karanjit Singh and Dr. Shuchita Upadhyaya. Outlier Detection: Applications And Techniques. URL: https://www.researchgate.net/publication/228686398_k-Nearest_neighbour_classifiers (Last accessed: 14. 11. 2021).
17. Yen-Chang Hsu, Yilin Shen, Hongxia Jin, Zsolt Kira. Generalized ODIN: Detecting Out-of-distribution Image without Learning from Out-of-distribution Data. URL: https://openaccess.thecvf.com/content_CVPR_2020/papers/Hsu_Generalized_ODIN_Detecting_Out-of-Distribution_Image_Without_Learning_From_Out-of-Distribution_Data_CVPR_2020_paper.pdf (Last accessed: 14. 11. 2021).
18. Breunig M. и др. LOF // Proceedings of the 2000 ACM SIGMOD international conference on Management of data - SIGMOD '00. 2000. URL: https://www.researchgate.net/publication/221214719_LOF_Identifying_Density-Based_Local_Outliers (Last accessed: 15. 11. 2021).

19. Na S., Xumin L., Yong G. Research on k-means Clustering Algorithm: An Improved k-means Clustering Algorithm // 2010 Third International Symposium on Intelligent Information Technology and Security Informatics. 2010.
20. Markus Goldstein, Andreas Dengel. Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm. URL: https://www.researchgate.net/publication/231614824_Histogram-based_Outlier_Score_HBOS_A_fast_Unsupervised_Anomaly_Detection_Algorithm (Last accessed: 17. 11. 2021).
21. Warp-core. URL: https://workday.github.io/warp-core/contents/anomaly_detection/ (Last accessed: 17. 11. 2021).
22. PDF) Support Vector Machines: Theory and Applications. URL: https://www.researchgate.net/publication/221621494_Support_Vector_Machines_Theory_and_Applications (Last accessed: 17. 11. 2021).
23. Fei Tony Liu, Kai Ming Ting Gippssland School of Information Technology Monash University, Victoria, Australia. Isolation Forest. URL: https://www.researchgate.net/publication/224384174_Isolation_Forest (Last accessed: 18. 11. 2021).
24. Xuehui Wang, Yong Zhang, Hao Liu, Yang Wang, Lichun Wang, and Baocai Yin. An Improved Robust Principal Component Analysis Model for Anomalies Detection of Subway Passenger Flow.
25. JR L., GG K. The measurement of observer agreement for categorical data. URL: <https://pubmed.ncbi.nlm.nih.gov/843571/> (Last accessed: 18. 11. 2021).
26. Study Finance. URL: <https://studyfinance.com/static/media/z-score.png> (Last accessed: 21. 11. 2021).
27. A Brief Overview of Outlier Detection Techniques .URL: <https://towardsdatascience.com/a-brief-overview-of-outlier-detection-techniques-1e0b2c19e561> (Last accessed: 21. 11. 2021).
28. Demo of DBSCAN clustering algorithm. URL: https://scikit-learn.org/stable/_images/sphx_glr_plot_dbscan_001.png (Last accessed: 22. 11. 2021).
29. Outlier detection with Local Outlier Factor (LOF). URL: [https://scikit-learn.org/stable/auto_examples/neighbors/plot_lof_outlier_detection.html#:~:text=The%20Local%20Outlier%20Factor%20\(LOF,lower%20density%20than%20their%20neighbors.](https://scikit-learn.org/stable/auto_examples/neighbors/plot_lof_outlier_detection.html#:~:text=The%20Local%20Outlier%20Factor%20(LOF,lower%20density%20than%20their%20neighbors.) (Last accessed: 22. 11. 2021).
30. Anomaly Detection using Autoencoders. URL: <https://towardsdatascience.com/anomaly-detection-using-autoencoders-5b032178a1ea> (date of application: 22. 11. 2021).
31. Cloudera Fast Forward. Deep Learning for Anomaly Detection. URL: <https://ff12.fastforwardlabs.com/> (Last accessed: 24. 11. 2021)

УДК 681.3.07

Analysis Of Biometric-Based Identification Algorithms In Electronic Trust Services Systems

V.O. Podhaiko, S.G. Rassomakhin

Rassomakhin S.G.*Doctor of Technical Sciences, Associate Professor, Head of the Department of BIST,**V. N. Karazin Kharkiv National University,**Maidan Svobodi, 6, c. Kharkiv, 61022**e-mail: bist@karazin.ua;*<https://orcid.org/0000-0003-1394-3588>**Podhaiko V.O.***Postgraduate**V. N. Karazin Kharkiv National University,**Maidan Svobodi, 6, c. Kharkiv, 61022**e-mail: podhaiko2020pg@student.karazin.ua;*<https://orcid.org/0000-0001-5905-9400>

The article is devoted to the usage of identification algorithms based on biometric personal data (biometrics) in the electronic trust services. This theme is of considerable interest due to the fact that it is becoming one of the most used tools in everyday life. It can include such things as fingerprint and facial scanners in modern cell phones as well as more official aspects such as a personal passport or signature.

The main problem of electronic identification is that the most effective algorithms: based on abstract-minutiae cylindrical codes, where minutiae are unique to each fingerprint and determine the points of change in the structure of the capillary lines (ending, splitting, breaking, etc.), the orientation of capillary lines and coordinates at these points.

Also, algorithms based on fuzzy extractors, where by fuzzy extractor we mean the system (object, algorithm) which transforms biometric data into random sequences, providing the opportunity to apply encryption methods for biometric security. Although they perform their role, they do not always work correctly and present a possible danger to a user. A comprehensive analysis of the advantages and disadvantages of such algorithms requires further investigation and combination of these algorithms to solve existing problems and improve overall response.

Not only could that provide a higher level of protection, but also greatly simplify the mathematical complexity of data processing, as well as lead to an increase in the number of correct triggers and overall increase the efficiency of using biometric technology in electronic trust services.

Keywords: *biometry, minutiae, imprecise extractor, algorithm, code.*

Анализ алгоритмов идентификации на основе биометрии в системах электронных доверительных услуг.

Рассомахин С.Г.*Доктор технічних наук, доцент, завідувач кафедри БІСТ,**Харківський національний університет ім. В.Н. Каразіна**Майдан Свободи, 6, м. Харків, 61022**e-mail: bist@karazin.ua;*<https://orcid.org/0000-0003-1394-3588>**Подгайко В.О.***Аспірант,**Харківський національний університет ім. В.Н. Каразіна**Майдан Свободи, 6, м. Харків, 61022**e-mail: podhaiko2020pg@student.karazin.ua;*<https://orcid.org/0000-0001-5905-9400>

Дана стаття присвячена актуальній на сьогоднішній день темі використання алгоритмів ідентифікації на основі біометричних даних особистості (біометрії) у системі електронних довірчих послуг. Це викликає значний інтерес через те, що це стає одним з найбільш використовуваних засобів в повсякденному житті. До нього можна віднести таке, як сканери відбитків пальців і обличчя у сучасних мобільних телефонах, так і більш офіційні аспекти, наприклад, особистий паспорт або підпис.

Основна проблема застосування електронної ідентифікації виражається в тому, що найбільш ефективні алгоритми: на основі абстрактно-мінуативних циліндричних кодів, де мінуція – це унікальні для кожного відбитку ознаки, що визначають пункти зміни структури папілярних ліній (закінчення, роздвоєння, розрив та ін.), орієнтацію папілярних ліній та координати в цих пунктах. Також алгоритми на основі нечітких екстракторів, де під нечітким екстрактором ми розуміємо систему (об'єкт, алгоритм), яка перетворює біометричні дані в випадкові послідовності, що надають можливість застосувати шифрувальні методи для біометричної безпеки. Вони хоча й виконують свою роль, але спрацьовують не завжди коректно, наражаючи на можливу небезпеку користувача.

Всебічний аналіз переваг і недоліків таких алгоритмів потребує подальшого їх розвитку та використання своєрідного симбіозу цих алгоритмів для вирішення існуючих проблем та покращення загального спрацьовування. Це повинно не тільки забезпечити більш високий рівень захисту, але й значно спростити математичну складність обробки даних за алгоритмами. І саме це призведе до підвищення кількості вірних спрацьовувань та підвищення ефективності використання біометричних технологій у системах електронних довірчих послуг.

Ключові слова: біометрія, мінуція, нечіткий екстрактор, алгоритм, код.

Introduction

Nowadays different identification systems are increasingly being used. We use them for electronic payments, for limiting access to objects, for identity authentication, etc.

Nowadays, the problem of correct identification and authentication of a person is more important than ever. And what can be more personal than individual biometric features, such as voice, retina, gait, fingerprints and, of course, DNA. All of these things together are covered by the term "biometrics." Today, the idea of using biometrics for the authentication and identification is very relevant. Moreover, almost all of us use them, for example, to unlock the phone screen with our fingerprint, or the so-called "Face ID" or face identifier. Biometric mechanisms are also applied in the modern identification documents, both foreign and Ukrainian. But unfortunately, the above-mentioned methods, especially the more complex ones, do not work quite correctly. For example, "Face ID" can unlock the screen only if a person's photo is presented to its scanner. And the fingerprint doesn't work when the fingers are wet or something else makes it difficult to access the fingerprint pattern.

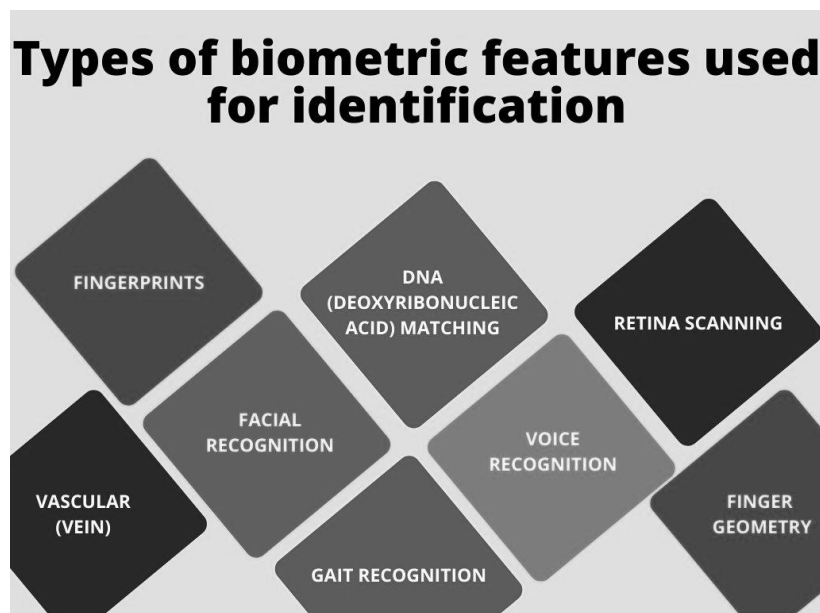


Fig.1 The types of biometric features used for identification

Such systems require more detailed study, all-around improvement and further development in order to avoid false positives and achieve a fully correct identification and authentication, where an error can be an extremely "costly".

1. Problem Statement

To analyze potentially better identification algorithms and determine directions for their further development in order to solve existing problems:

- confidentiality and delimitation (data obtained during biometric registration can be used for purposes that the registered individual did not consent to (was not aware of);

- dangers to owners of protected data (there is a possibility of an attempt on the part of the holder of biometric identifiers to gain access);
- the possibility of cancelling biometric data (the advantage of a password over biometrics is the possibility of changing it). Mainly, the cancellation of biometrics is a distortion of the biometric image or properties prior to their approval.

2. Main points

Fingerprint identification (dactyloscopy) is the most common technology used in biometric access control systems today. The technology is based on the unique pattern of fingerprints on people's fingers. The fingerprint obtained with the scanner is converted into a digital code, which is stored in a database, and then compared to previously entered and converted fingerprint codes.

The biometric access by fingerprint is easy to use, convenient and reliable. The devices that scan the fingerprints are very reliable and cheap. The disadvantages include distortion of the papillary pattern by small scratches, cuts, chemical reagents; inability to read the fingerprint by some scanners if the skin is too wet.

Two types of features can be identified in each fingerprint – global and local. Global features are those that can be seen with the naked eye.

Papillary pattern:

- pattern area – a highlighted fragment of the fingerprint in which all global features are localized;
- core or center – a point localized in the middle of the imprint or some selected area;
- the "delta" point – the starting point. The place where there is a separation or joining of the papillary furrows, or a very short furrow (can reach to a point);
- line type – the two largest lines that start as parallel and then diverge and circle the entire image area;
- line count – the number of lines on the image area, or between the core and the "delta" point.

Papillary pattern types:

- "loop" type patterns (left, right, central, double),
- "delta" or "arc" type patterns (simple and sharp),
- "spiral" type patterns (central and mixed).

Another type is local features. They are called minutiae (peculiarities or special points) which are unique for each fingerprint and determine the points of capillary lines structure changes (ending, splitting, breaking etc.), the orientation of capillary lines and coordinates in these points. Each fingerprint can contain up to 70 or more minutiae.

Practice shows that the fingerprints of different people may have the same global features, but it is completely impossible to have the same minutiae micropatterns. Therefore, global features are used to separate the database into classes and at the stage of authentication. At the second stage of recognition local features are used.

Comparisons of prints by local features (minutiae) include:

1. Improving the quality of the original image of the print. Sharpness of the borders of the lines of strokes is increased.
2. Calculation of the orientation field of the fingerprint's lines. The image is divided into square blocks with sides more than 4 pixels, and the angle t of line orientation for a fragment of the fingerprint is calculated using brightness gradients.
3. Binarizing the image of the print. Binning to a black and white image (1 bit) by thresholding.
4. Thinning the lines of the fingerprint image. Thinning is performed until the lines are 1 pixel wide.
5. Minutiae extraction. The image is divided into blocks of 3x3 pixels. After that, the number of black (non-zero) pixels around the center is counted. The pixel in the center is considered to be a minutia if it is non-zero, and there are one (minutiae "ending") or three (minutiae "branching") neighboring non-zero pixels. Coordinates of detected minutiae and their orientation angles are written in a vector. When registering users, this vector is considered a reference and is written to the database. During recognition, the vector determines the current fingerprint.
6. Matching minutiae. Two fingerprints of the same finger will differ from each other by rotation, offset, change in scale, and/or area of contact, depending on how the user places the finger on the scanner.

Therefore, it is impossible to tell whether a fingerprint belongs to a person or not by a simple comparison (the vectors of the reference and the current fingerprint may differ in length, contain mismatched minutiae, etc.). Because of this, the comparison process must be implemented for each minutia separately.

Comparison stages include data registration, searching for pairs of matching minutiae, evaluating the print matching.

During registration the parameters of affine transformations (rotation angle, scale and shift) are determined, at which some minutiae from one vector correspond to some minutiae from the second vector.

The results of a search are up to 30 rotation values (from -15 to +15 degrees), 500 shift values (from -250 px to +250 px, though sometimes even smaller limits are chosen), and 10 scale values (from 0.5 to 1.5 in steps of 0.1) for each minutia, up to 150,000 steps for each of the 70 possible minutiae in total. (In practice, all possible options are not enumerated – after selecting the right values for one minutia, there is an attempt to substitute them for the other minutiae, otherwise it would be possible to match almost any prints to each other).

The assessment of print matching is done by the formula

$$K = \frac{D^2}{pq} * 100,$$

where D – the number of matching minutiae,

p – the number of minutiae of the reference,

q – the number of minutiae in the recognized print). If the result exceeds 65%, the prints are considered identical (the threshold can be lowered by setting another level of vigilance).

If authentication has been performed, this is the end of the process. For identification this process must be repeated for all fingerprints in the database (then the user with the highest matching level is selected (of course, the result must be above the 65% threshold)).

3. Current identification algorithms

The main directions of solving the specified problems are the algorithms abstract-minutiae cylindrical codes, and the algorithms based on fuzzy extractors. They are the most used in biometrics to provide a high probability of a correct identification.

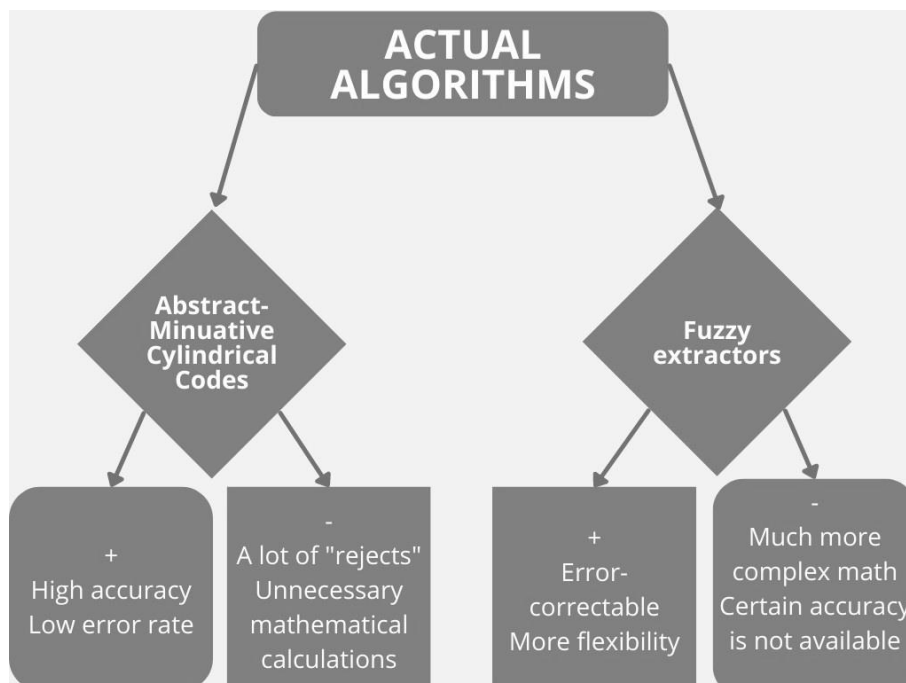


Fig.2 The used algorithms

Abstract-Minutiae Cylindrical Codes.

Abstract-Minutiae cylindrical codes are based on the small details of fingerprint descriptors, which take into account the smallest information in a fingerprint image to match them. Minutiae are features unique to each fingerprint, defining points of change in the structure of the capillary lines (termination,

bifurcation, break, etc.), the orientation of the capillary lines and the coordinates at these points. Each fingerprint can contain up to 70 or more minutiae. It is thanks to their comparison that the local features are compared.

The algorithm based on minutiae cylindrical codes uses three-dimensional data structures called cylinders, where each cylinder is oriented in the direction of the central minutiae throughout the image. Minor locations are spatial points where the orientation, frequency, and energy have a higher differential of change.

The orientation image computed by STFT analysis differs from traditional orientation images computed using simple derivatives. With the exception of the core and delta points, any local area of the fingerprint image provides consistent texture information using STFT. This is not the case in traditional gradient orientation estimation.

The approach by which it is created is called the default function, where not every cell in the cylinder will accumulate the specified contribution. Thus, cells that lie outside the valid territory mask are considered invalid, and cells without neighbors have zero contribution. The cylinder is kept or discarded according to validity constraints. These constraints include the minimum number of neighbors around the central minutiae at constant radius, as a percentage of the total number of valid cells. Only valid cylinders will be part of the imprint pattern.

The disadvantage of this approach is the complexity of mathematical calculations of a large number of minutiae and a significant number of cylinders that are processed but discarded for one reason or another (considered invalid).

Fuzzy extractors.

A fuzzy extractor is a system (object, algorithm) that converts biometric data into random sequences that provide the ability to apply encryption methods for biometric security. They are used to encrypt and authenticate user transactions. In this case, the biometric input is treated as a key. The word "fuzzy" in the extractor's name implies that the values of the resulting sequence have a form close enough to the original and can confirm the authenticity of the identity.

The algorithm using fuzzy extractors is a method that allows us to uniquely recover the secret key from inaccurately reproduced biometric data involving auxiliary data, which is open. According to the algorithm, a sequence of actions is performed:

- initialization, where a security parameter is specified that defines the length of the public and secret keys and the triggering threshold, based on which the algorithm generates a secret master key and public parameters;
- extraction, where a specific identity and the secret master key are specified, and the algorithm itself processes the data and returns the user's secret key;
- encryption, where the algorithm returns a ciphertext based on the user's secret key, his identity, and the embedded message;
- decryption, where by the secret key and the ciphertext encrypted with the identity, the algorithm returns the message if the identity data is confirmed, or stops working otherwise.

Subsequently, the stability of such an algorithm is evaluated from the point of view of force attacks, such as full brute force, collision creation, etc. The stability to them being obtained, the algorithm is moved to the analysis of its stability against analytical attacks.

The disadvantage of this approach is the complexity of mathematical apparatus of data processing in the presence of a significant number of errors and the inability to provide the necessary accuracy when it is necessary to correct these errors.

Conclusions

Thus, the most important and promising areas for further research are those related to the incorrect operation of fuzzy extractors and abstract-minutiae codes, which address the issues of correct identification and authentication of the person. The identification and authentication based on the synthesis of the "strengths" of both algorithms will balance their individual drawbacks.

ЛІТЕРАТУРА

1. Подгайко В.О., Рассомахін С.Г. *Аналіз алгоритмів ідентифікації у системах електронних довірчих послуг.*// Зб. наук. праць I Міжнародної науково-технічної конференції “Системи і технології зв’язку, інформатизації та кібербезпеки: Актуальні питання і тенденції розвитку”, м. Київ, 25–26 листопада 2021 року. С. 239-240.
2. A. Menezes, P. van Oorschot and S. Vanstone. *Handbook of applied cryptography.* – Boca Raton, FL: CRC Press, 1996. P. 306-312.
3. Yen-Lung Lai, Jung-Yeon Hwang, Zhe Jin, Soohyong Kim, Sangrae Cho and Andrew Beng Jin Teoh. *A Symmetric Keyring Encryption Scheme for Biometric Cryptosystems.* – arXiv:1807.02251v1 [cs.CV], 6 Jul 2018. P. 1-15.
4. Wajih Ullah Baig, Umar Munir, Waqas Ellahi, Adeel Ejaz, Kashif Sardar. *Minutiae Texture Cylinder Codes for fingerprint matching.*// Information Sciences. V.502, October 2019. P. 492-509.

REFERENCES

1. Podhaiko V.O., Rassomakhin S.G. *Analysis of identification algorithms in electronic trust services systems.* // Coll. of Scient. Papers of the First International Scientific and Technical Conference "Communication Systems and Technologies, Informatization and Cyber Security: Current Issues and Development Trends", Kyiv, November 25-26, 2021. – P. 239-240.
2. A. Menezes, P. van Oorschot and S. Vanstone, *Handbook of applied cryptography.* – Boca Raton, FL: CRC Press, 1996. – P. 306-312.
3. Yen-Lung Lai, Jung-Yeon Hwang, Zhe Jin, Soohyong Kim, Sangrae Cho and Andrew Beng Jin Teoh. *A Symmetric Keyring Encryption Scheme for Biometric Cryptosystems.* – arXiv:1807.02251v1 [cs.CV], 6 Jul 2018. – P. 1-15.
4. Wajih Ullah Baig, Umar Munir, Waqas Ellahi, Adeel Ejaz, Kashif Sardar. *Minutiae Texture Cylinder Codes for fingerprint matching.*// Information Sciences. V.502, October 2019. – P. 492-509.

УДК 004.9

Дослідження можливостей моделі 3d-візуалізації навчальних приміщень кафедри

Помогаєв К. О., Толстолузька О. Г., Артюх О. А.

**Помогаєв
Костянтин
Олександрович**

*студент;
Харківський національний університет імені В.Н. Каразіна,
площа Свободи 6, м. Харків Україна, 61022
e-mail: xa11867765@student.karazin.ua
<https://orcid.org/0000-0002-5010-2369>*

**Толстолузька
Олена Геннадіївна**

*д. т. н., с.н.с.; професор кафедри теоретичної та прикладної
системотехніки;
Харківський національний університет імені В.Н. Каразіна,
площа Свободи 6, м. Харків Україна, 61022
e-mail: elena.tolstoluzka@karazin.ua
<https://orcid.org/0000-0003-1241-7906>*

**Артюх
Олексій
Анатолійович**

*ст. викладач кафедри теоретичної та прикладної
системотехніки;
Харківський національний університет імені В.Н. Каразіна,
площа Свободи 6, м. Харків Україна, 61022
e-mail: oleksiy.artiuh@karazin.ua
<https://orcid.org/0000-0002-2038-4131>*

Сучасні технології важко уявити без сфери 3D-моделювання та 3D-візуалізації. Найважливішою її перевагою є економія часу, оскільки за допомогою 3D-моделей можна віртуально досліджувати будь-які об'єкти, не створюючи їх в реальному житті. Звичайно, створення 3D-моделей об'єктів потребує значно менше часу ніж створення цих об'єктів фізично. За допомогою даної сфери створюються мультфільми та фільми, рекламуються товари, створюються віртуальні тури будь-якої місцевості. У наш час існує багато програм для створення 3D-моделей, які оновлюються та розвиваються дуже стрімко. У роботі був описаний процес створення 3D-моделі аудиторій університету імені В. Н. Каразіна, за допомогою яких можна проводити віртуальні екскурсії по цим аудиторіям. Були досліджені можливості 3D-моделі аудиторій. Розроблена 3D-модель має можливість інтерактивної взаємодії з користувачем. Для того, щоб кожен користувач міг отримати доступ до моделі, її було розміщено на сайті. Користувачі мають змогу обирати об'єкт або зону огляду, ракурс, кути нахилу. Можна збільшити або зменшити масштаб зображення. Таким чином, за допомогою розробленої 3D-моделі можна отримати уявлення про аудиторії та об'єкти, які там знаходяться. В майбутньому планується розробити способи пришвидшення розробки 3D-моделей аудиторії шляхом створення аддонів до програми Blender на Python.

Ключові слова: *Blender, модель, 3D-модель, 3D-візуалізація.*

Research of the possibilities of 3d-visualisation model of classrooms of the department

**Pomohaiev
Kostiantyn**

*student;
V. N. Karazin Kharkiv National University,
Svobody Sq 6, Kharkiv, Ukraine, 61022*

**Tolstoluzhskaya
Elena**

*doctor of Engineering Sciences; professor of theoretical and applied systems
engineering department;
V. N. Karazin Kharkiv National University,
Svobody Sq 6, Kharkiv, Ukraine, 61022*

**Artyukh
Oleksiy**

*senior lecturer of theoretical and applied systems engineering department;
V. N. Karazin Kharkiv National University, Svobody Sq 6, Kharkiv, Ukraine,
61022*

It is difficult to imagine modern technologies without the 3D modeling and 3D visualization. The most important advantage of this technology is the possibility to create and inspect virtually 3D models of any objects which takes much less time than creating these objects physically. Therefore, this technology allows creating cartoons, films, virtual tours, and advertisements. Nowadays there are many programs for creating 3D models, which are constantly developed and updated. In this work the process of creating the 3D model of the V.N. Karazin University classrooms is described, which allows conducting virtual tours of the university. Blender, the free [3D computer graphics](#) software tool has been used to develop the model. The main advantage of Blender over other programs is that, each user is able to create applications for this program. The development stages, namely, a concept phase, a 3D modeling, an application of textures and colors, lighting, and rendering are described. In order to understand the capabilities and functions of the created 3D-model, the methodology of functional modeling IDEF0 has been used. With the help of IDEF0 notation the scheme of 3D-model has been created. The processes of users' interaction with 3D-model have been considered. The opportunities presented by the 3D model of classrooms have been investigated. The developed interactive 3D model is available to users on the university site. Users can choose an object or field of view, angle, and tilt angles. Users can also zoom in or out of the image. Thus, using the developed 3D model, users can learn about the classrooms and their interior. We are planning to increase the efficiency of the process of creating such 3D models by creating add-ons to Blender in Python language which allows automating the process of creating a 3D object. Their using can reduce the time needed to create 3D models of remaining classrooms, as well as, the number of developers required.

Key words: *Blender, model, 3D-model, 3D-visualization.*

1 Вступ

Модель – це штучно створюваний об'єкт, який замінює собою деякий об'єкт реального світу (об'єкт моделювання) і відтворює обмежене число його властивостей. Процес побудови моделі називають моделюванням. Моделювання – це метод пізнання дійсності, використовуваний різними науками. На сьогоднішній день, моделювання є досить актуальним.

Моделі дозволяють розробляти та досліджувати об'єкти без використання матеріальних ресурсів на їх виготовлення. Моделювання дозволяє скоротити час, витрати та ресурси при введенні об'єктів в господарський обіг та експлуатації на протязі всього життєвого циклу. Наприклад, при тестуванні моделі корабля: модель спочатку розробляється у 3D, а потім створюється її матеріальна копія, яка має менший розмір, та на яку витрачено набагато менше ресурсів та часу ніж на справжній корабель.

За допомогою моделей можна проводити дослідження та ознайомлення з об'єктами у віддаленому режимі. Наприклад, за допомогою схеми в графічній нотації IDEF0 можна ознайомитися з певними процесами та системами, не взаємодіючи з ними в реальному житті. Моделювання надає можливість імітації та створення граничних ситуацій при використанні об'єктів без безпосереднього ризику. Тобто при тестуванні моделі, якщо створюється гранична ситуація, саме модель зазнає ризику, а не реальні об'єкти.

3D-моделювання є одним із напрямів моделювання і являє собою процес створення тривимірних моделей місцевості або об'єктів за допомогою комп'ютерних програм. 3D-моделювання є дуже корисним, коли потрібно за невелику кількість часу відтворити об'єкт, провести на ньому необхідні дослідження, зробити певні зміни. Таким чином, витрачається набагато менше ресурсів, ніж на створення реального об'єкта [1].

Розроблена 3D-модель аудиторій дозволить віртуально, з будь-якого місця оглянути ці приміщення. Користувачі мають змогу отримати повне уявлення про них, об'єкти, які там розміщені, їх зовнішній вигляд та призначення. Кожен має доступ для перегляду 3D-моделей аудиторій. В умовах карантину та дистанційного навчання це має особливе значення.

2 Процес розробки 3D-моделі

Виходячи з призначення спочатку до розроблюваної моделі були висунуті наступні вимоги:

1. Надання повної уяви про приміщення аудиторій, розташовані там об'єкти, відповідно до їх зовнішнього вигляду, пропорцій розмірів, кольорів.

2. Деталізація має бути на такому рівні, який зможе надати уяву про справжній вигляд та призначення обладнання в аудиторіях, складових предметів та окремих частин об'єктів, які там розміщені.
3. Потрібно надати можливість огляду 3D-моделі з різних ракурсів, зберігаючи при цьому пропорції. В модельованому приміщенні огляд має бути здійснений в усіх трьох площинах з місця спостереження.
4. Потрібно надати користувачам можливість інтерактивної взаємодії з 3D-моделлю за допомогою інтуїтивно-зрозумілого інтерфейсу.
5. Забезпечити розроблювану модель можливістю подальшої модернізації та нарощування.
6. Створити можливість віддаленого доступу до 3D-моделі за допомогою засобів комунікації в мережі.

Процес розробки 3D-моделі можна поділити на наступні етапи: фаза концепції, 3D-моделювання, нанесення текстур та кольорів, освітлення, рендеринг.

2.1 Фаза концепції

Спочатку була сформульована ідея, розроблена концепція та обрані аудиторії для моделювання. Було продумано, як саме виглядатиме 3D-модель та були розроблені вимоги до неї. Для більш зручного моделювання були зроблені фото приміщень з різних ракурсів, а також фото обладнання в цих аудиторіях. Для різних об'єктів була зроблена необхідна кількість фото для того, щоб можна було максимально точно їх відтворити. Після цього була обрана програма для 3D-моделювання. Зробивши аналіз програмного забезпечення для створення 3D-моделей, була обрана програма Blender [2].

2.2 3D-моделювання

На другому етапі були змодельовані обрані аудиторії. За допомогою вбудованих інструментів програми Blender були побудовані 3D-моделі об'єктів відповідно до їх розмірів та форм в реальності [3].

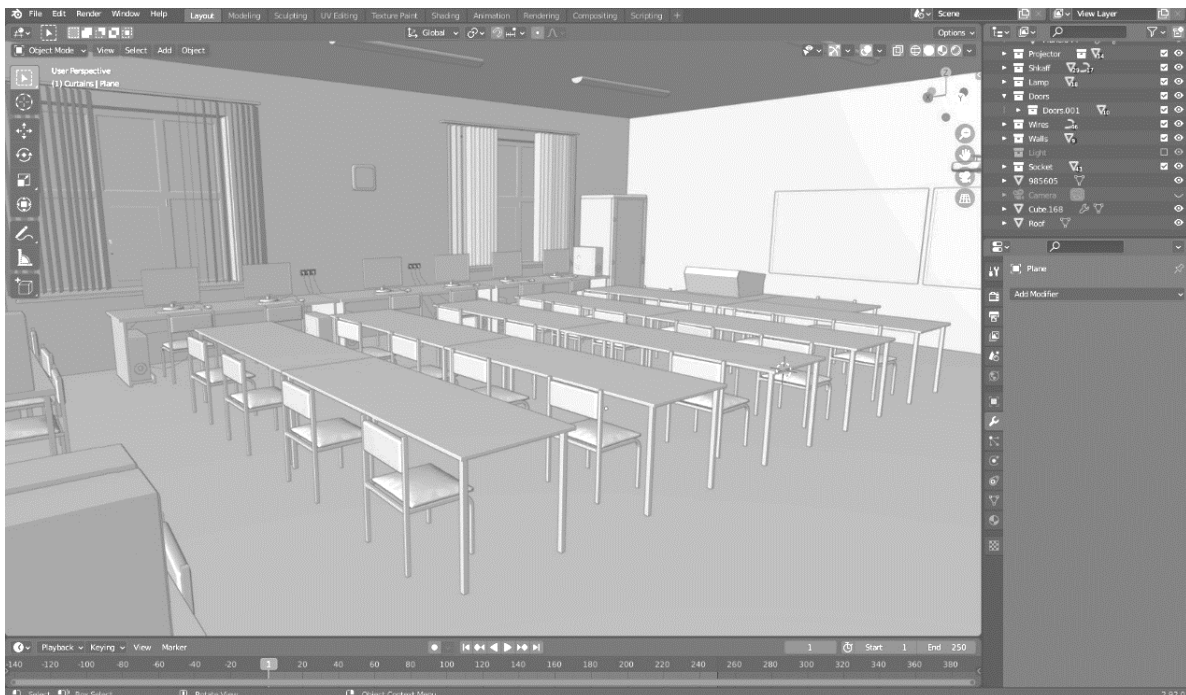


Рис.2.1 – Результат 3D-моделювання аудиторії 317

2.3 Нанесення текстур та кольорів

Наступним кроком було надання кольору та текстур створеним 3D-моделям. Об'єкти, створені за допомогою 3D-моделювання, мають плоский сірий колір, який наявний в програмі за замовчуванням. 3D-текстурування - це, в основному, обгортання 2D- текстури навколо 3D-об'єкта і визначення впливу світла на нього. Воно відповідає за застосування атрибутів кольору та поверхні до 3D-об'єктів. Загальна мета полягає в тому, щоб поверхня моделі відповідала її концептуальному дизайну або аналогу в реальному світі.



Рис.2.2 – Текстура підлоги

За процес нанесення текстур в програмі Blender відповідають панелі, які називаються шейдерами. Поєднуючи ці шейдери між собою, можна створювати різні матеріали. Шейдери повідомляють програмі, який матеріал має той чи інший об'єкт. Кожен шейдер має в собі певну кількість налаштувань, які можна змінювати. Нанести матеріал можна не тільки на певний об'єкт, але і на окрему його частину[4].

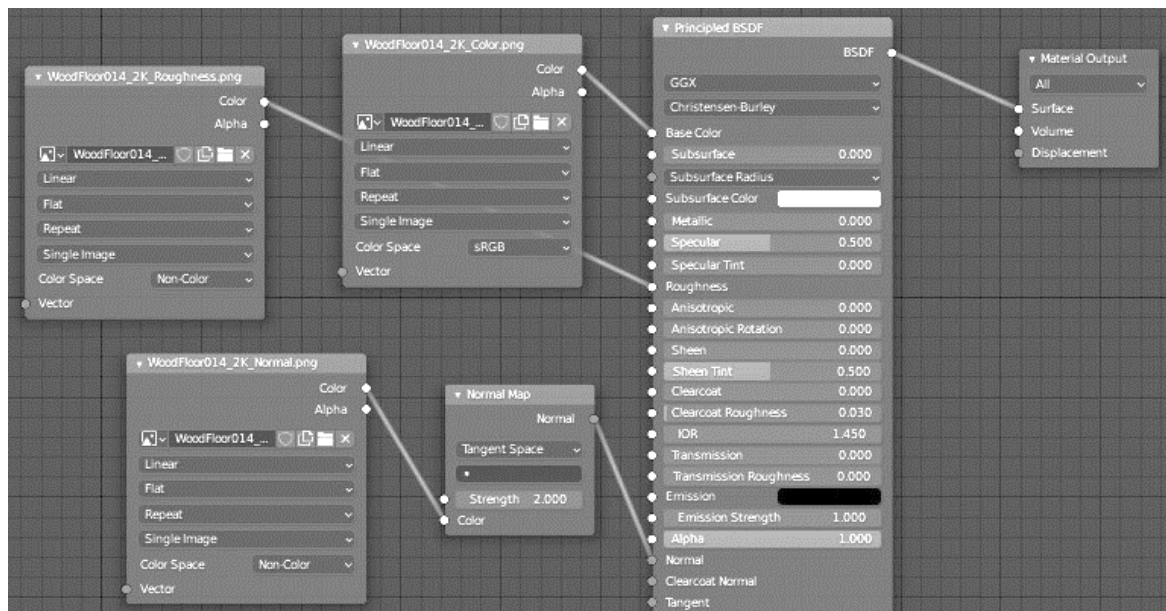


Рис.2.3 – Комбінація шейдерів для створення текстури підлоги

2.4 Освітлення

На етапі освітлення були виставлені джерела світла. Як і в реальних аудиторіях було створене світло від ламп, а також від сонця. Вони були розміщені згідно їх позицій в аудиторіях, також були налаштовані їх характеристики [5]. Іноді добре змодельований 3D-об'єкт виглядає непереконливо через погану реалізацію 3D-освітлення. І навпаки, правильний вибір техніки

освітлення може значно підвищити якість комп'ютерної сцени. Тому цьому процесу треба приділяти значну увагу.

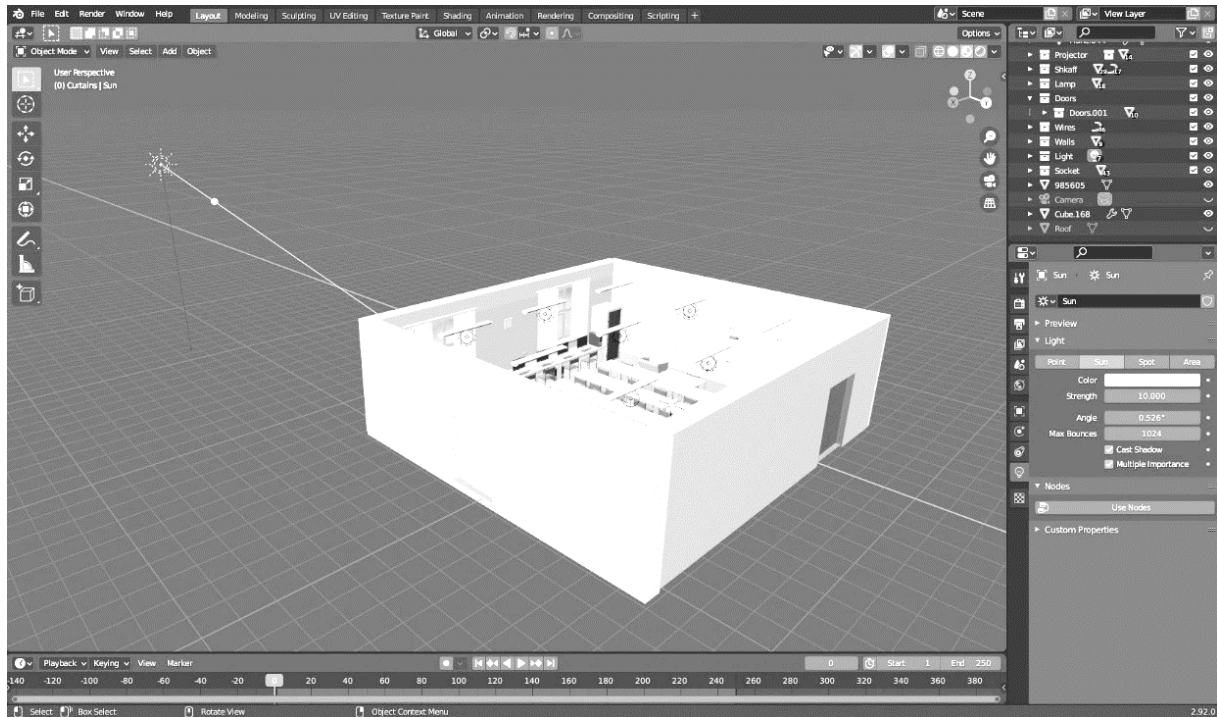


Рис.2.4 – Розміщення джерел світла

2.5 Рендеринг

Рендеринг – це процес отримання остаточного зображення аудиторії. Для цього в програмі був створений об'єкт “камера” і розміщений таким чином, щоб в кадр потрапляла та ділянка, яка буде в результаті. Однак, для того, щоб 3D-модель можна було оглядати з усіх ракурсів, у налаштуваннях камери була обрана опція “360 градусів”. Таким чином на виході буде отримано зображення, яке можна оглядати з різних сторін та змінювати його масштаб[6].



Рис. 2.5 –3D-модель аудиторії 317

Створені 3D-моделі були розміщені на сайті. Користувачі можуть перейти за наступними посиланнями та оглянути їх: <https://kuula.co/post/7w0gp> - аудиторія 317, <https://kuula.co/post/73k96> - аудиторія 320.

3 Можливості розробленої 3D-моделі

Для того, щоб краще зрозуміти можливості та функції створеної 3D-моделі, можна скористатися методологією функціонального моделювання IDEF0. За допомогою нотації IDEF0 можна розглянути процеси, які відбуваються при взаємодії користувача з 3D-моделлю [7].

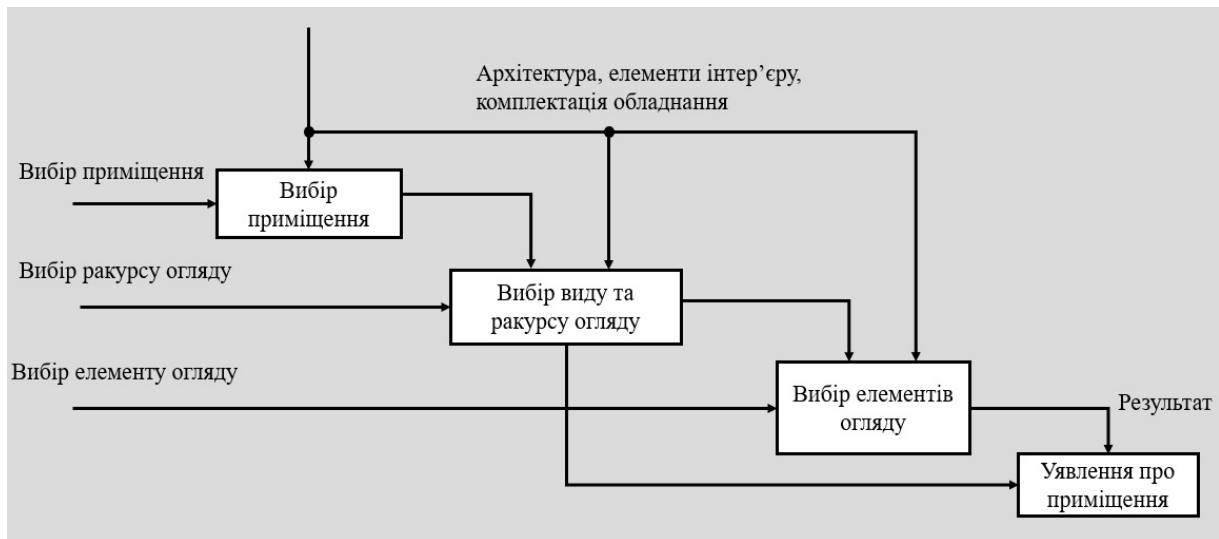


Рис. 3.1 - Модель 3D-візуалізації в нотації IDEF0

На рисунку 3.1 представлена модель в нотації IDEF0. Вхідними технічними даними моделі є архітектура приміщень, елементи інтер'єру, комплектація обладнання. Незалежно від того, яка саме аудиторія буде обрана для 3D-моделювання, спочатку потрібно отримати інформацію про її вигляд, об'єкти, їх розмір та форму [8].

Як показано у першому блоці з боку користувача може бути здійснений вибір приміщення для огляду зі списку, який пропонує модель. На даний момент можна оглянути 3D-моделі аудиторій 317 та 320. В майбутньому можна створити 3D-моделі інших аудиторій, і таким чином буде більше варіантів для огляду.

Після цього, як можна побачити у другому блоці, користувач може обрати вид та ракурс огляду. На сайті для кожної 3D-моделі були створені контрольні точки. Натиснувши на них, можна, таким чином, віртуально переміщатися по аудиторії і змінювати місце огляду. Наприклад, для 3D-моделі аудиторії 317 було створено 4 контрольні точки. Вони були розташовані по різних кутам аудиторії для того, щоб можна було оглянути її з різних сторін. Користувач може оглянути об'єкти під різним кутом, переміщуючи камеру за допомогою миші. Можна оглянути приміщення у панорамному виді або в перспективі. Дивлячись на схему IDEF0 на рисунку 5, можна побачити, що після цього користувач вже може досягти результату та сформувати уявлення про приміщення і отримати про них інформацію.

За необхідності користувач може обрати та детальніше оглянути окремі елементи моделі, збільшивши їх зображення. Це показано у третьому блоці схеми IDEF0. За допомогою колеса миші або клавіш Z та Q користувач може змінювати масштаб зображення. Таким чином, можна детальніше розглядати окремі об'єкти або їх частини. Після цього користувач в результаті отримує необхідну інформацію та формує уявлення про приміщення та об'єкти.



Рис 3.2 – Початкове зображення 3D-моделі аудиторії 317

На прикладі тестування 3D-моделі виглядає наступним чином. При переході за посиланням користувач може побачити перед собою початкове зображення 3D-моделі. Було виставлене початкове положення камери, яке користувач в процесі може змінювати. Контрольні точки можна побачити на самому зображенні у вигляді помаранчевих кругів. Також список контрольних точок наявний у правому верхньому куті.

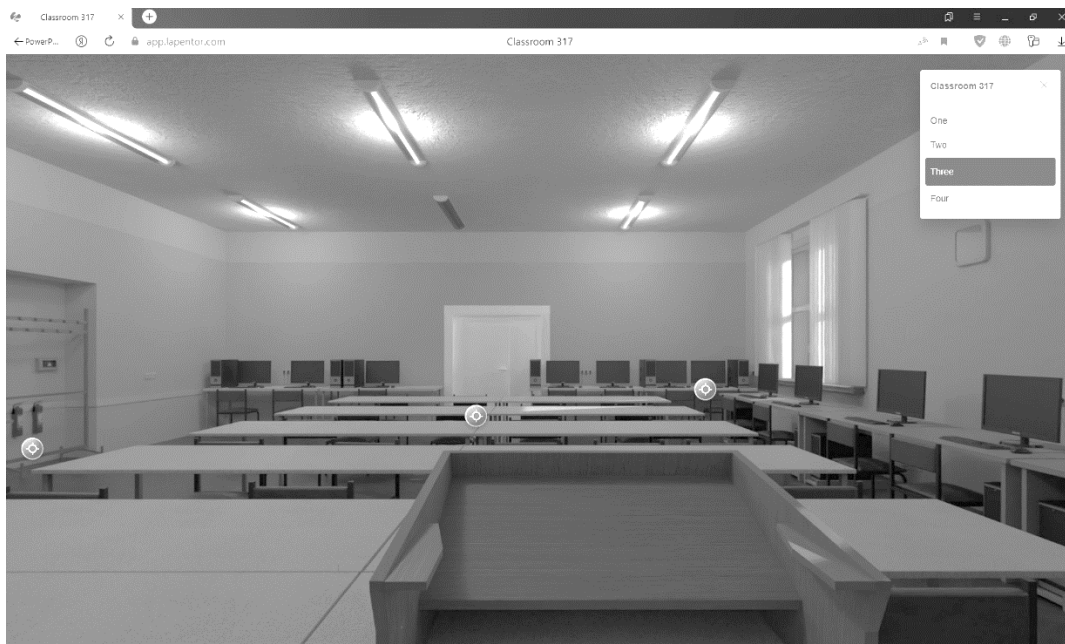


Рис 3.3 – Зображення 3D-моделі після зміни контрольної точки

Після натиску на контрольну точку камера змінить своє положення як показано на рисунку 3.3. Таким чином, користувач може переміщатися по контрольним точкам і оглядати 3D-модель з різних сторін. За допомогою обертів «миші» користувач також може змінювати ракурс зображення.

Також користувач за допомогою колеса «миші» може приблизити зображення і оглянути об'єкт детальніше. Це показано на рисунку 3.4. Це можна зробити, знаходячись у будь-якому місцезнаходженні. Таким чином можна віртуально оглядати інші об'єкти в аудиторії. В залежності від того, що потрібно користувачу.



Рис 3.4 – Зміна масштабу зображення

В майбутньому планується збільшити ефективність процесу створення таких 3D-моделей. Це можна зробити шляхом створення аддонів до програми Blender на мові Python. Аддон – це доповнення до самостійної програми. Відмінністю програми Blender від інших є те, що кожен користувач має доступ до створення додатків до цієї програми. Їх застосування може зменшити час на створення 3D-моделей інших аудиторій або окремих об'єктів, які там знаходяться, а також зменшити витрати людських ресурсів. Якщо для того, щоб створити 3D-об'єкт, потрібно зробити ряд необхідних дій, застосувати певні інструменти та операції, то за допомогою аддонів цей процес можна автоматизувати [9].

4 Висновки

У ході роботи проаналізовані можливості моделі 3D-візуалізації навчальних приміщень кафедри. Спочатку були описані вимоги до 3D-моделі. На основі цих вимог була розроблена 3D-модель. Після цього були описані етапи розробки. Вони включають в себе: фазу концепції, 3D-моделювання, нанесення текстур та кольорів, освітлення, рендеринг. Для того, щоб зрозуміти можливості та функції створеної 3D-моделі, використана методологія функціонального моделювання IDEF0. За допомогою нотації IDEF0 створена схема 3D-моделі, розглянуті процеси, які відбуваються при взаємодії користувача з 3D-моделлю. Спочатку користувач може обрати аудиторію для огляду. Після цього можна обрати ракурс огляду за допомогою контрольних точок. На основі цієї інформації користувач може сформулювати уявлення про приміщення. За необхідності можна змінити масштаб зображення за допомогою колеса «миші». Можливості 3D-моделі також були розглянуті на прикладі. В подальшому планується збільшити ефективність процесу створення таких 3D-моделей.

ЛІТЕРАТУРА

1. Andrew McDonald. A Beginners Guide to 3D, 2019. <https://discover.therookies.co/2019/04/27/a-beginners-guide-to-3d/>.
2. А. А. Прахов. Самоучитель Blender 2.7, 2016. https://books.google.com.ua/books/about/Самоучитель_Blender_2_7.html?id=CEYpDQAAQBAJ&printsec=frontcover&source=kp_read_button&redir_esc=y#v=onepage&q&f=false.
3. Antti Konsén. 3D model of a real world model, 2015. https://www.theseus.fi/bitstream/handle/10024/103541/Konsen_Antti.pdf?sequence=1&isAllowed=y.

4. Megan Ashley Stephens. 3D Modeling Techniques for Print and Digital Media, 2015. <https://core.ac.uk/download/pdf/232140462.pdf>.
5. Siddhartha Chaudhuri. 3d modeling with data-driven suggestions, 2011. https://stacks.stanford.edu/file/druid:vq766tr8762/sid_thesis-augmented.pdf.
6. В. П. Большаков. Основы 3D-моделирования. Изучаем работу в AutoCAD, КОМПАС-3D, SolidWorks, Inventor, 2012. https://books.google.com.ua/books?id=xmtiaDiwp4EC&pg=PA11&hl=ru&source=gbs_toc_r&cad=3#v=onepage&q&f=false.
7. Черемных С.В., Семенов И.О., Ручкин В.С. Моделирование и анализ систем. IDEF-технологии, 2006. <https://studfile.net/preview/961103/>.
8. О.А. Цуканова. Методология и инструментарий моделирования бизнес-процессов. Учебное пособие, 2005. <https://books.ifmo.ru/file/pdf/1720.pdf>.
9. Blender official website. <https://www.blender.org/features/>.

REFERENCES

1. Andrew McDonald. A Beginners Guide to 3D, 2019. <https://discover.therookies.co/2019/04/27/a-beginners-guide-to-3d/>.
2. A. A. Prakhov. Tutorial Blender 2.7, 2016. [in Russian] https://books.google.com.ua/books/about/Blender_2_7.html?id=CEYpDQAAQBAJ&printsec=frontcover&source=kp_read_button&redir_esc=y#v=onepage&q&f=false.
3. Antti Konsén. 3D model of a real world model, 2015. https://www.theseus.fi/bitstream/handle/10024/103541/Konsen_Antti.pdf?sequence=1&isAllowed=y.
4. Megan Ashley Stephens. 3D Modeling Techniques for Print and Digital Media, 2015. <https://core.ac.uk/download/pdf/232140462.pdf>.
5. Siddhartha Chaudhuri. 3d modeling with data-driven suggestions, 2011. https://stacks.stanford.edu/file/druid:vq766tr8762/sid_thesis-augmented.pdf.
6. V.P.Bolshakov. Basics of 3D modeling. We study work in AutoCAD, KOMPAS-3D, SolidWorks, Inventor, 2012. [in Russian] https://books.google.com.ua/books?id=xmtiaDiwp4EC&pg=PA11&hl=ru&source=gbs_toc_r&cad=3#v=onepage&q&f=false.
7. Cheremnykh S.V., Semenov I.O., Ruchkin V.S. System modeling and analysis. IDEF-technologies, 2006. [in Russian] <https://studfile.net/preview/961103/>.
8. O.A. Tsukanov. Methodology and tools for modeling business processes, 2005. [in Russian] <https://books.ifmo.ru/file/pdf/1720.pdf>.
9. Blender official website <https://www.blender.org/features/>.

УДК 593.3

Гідропружні коливання оболонок обертання при комбінованих вертикальних та горизонтальних навантаженнях

О.О. Усатова, Д.В. Крютченко

**Усатова Ольга
Олександрівна***аспірант**Інститут проблем машинобудування ім. А. М. Підгорного НАНУ, вул.
Пожарського, 2/10, Харків, 61046, Україна
e-mail: usatova.olia@gmail.com*<https://orcid.org/0000-0002-5813-0321>**Крютченко Денис
Володимирович***провідний інженер**Інститут проблем машинобудування ім. А. М. Підгорного НАНУ, вул.
Пожарського, 2/10, Харків, 61046, Україна
e-mail: wollydenis@gmail.com*<https://orcid.org/0000-0002-6804-6991>

Функціонування сучасних конструкцій відбувається в умовах дії підвищених технологічних навантажень. Оболонки обертання відносяться до найбільш застосованих елементів конструкцій. Умови експлуатації таких елементів передбачають вплив оточуючого середовища, тобто функціонування конструкцій відбувається в умовах взаємодії пружних оболонкових елементів з рідиною або газом. В роботі розглянуто коливання пружної оболонки обертання, частково заповненої рідиною, за умов дії зовнішніх навантажень в вертикальному та горизонтальному напрямках одночасно. Вважалось, що рідина є ідеальною та нестисливою, а її рух під дією прикладених навантажень є безвихровим. Це дало змогу ввести потенціал швидкостей для обчислення тиску рідини на поверхні оболонки та для визначення функції, що описує рівня підйому вільної поверхні. На першому етапі визначені частоти а форми вільних коливань оболонки з рідиною. Ці частоти зазвичай поділяють на два відокремлені спектри: частоти коливань вільної поверхні та частоти коливань пружних стінок. Для числової реалізації використано метод граничних елементів. В цьому дослідженні встановлено, що для досить тонких пружних стінок резервуара фундаментальна частота при зв'язаних коливаннях може бути набагато меншою, ніж частота рідини в оболонці з жорсткими стінками. Зі збільшенням товщини стінки резервуара цей ефект стає незначним, а нижча частота коливань оболонки з рідиною наближається до частоти коливань рідини в жорсткому резервуарі. Другий етап дослідження пов'язаний із аналізом комбінованих горизонтальних та вертикальних навантажень. Розглянуто випадки, коли частоти навантажень є близькими до фундаментальної частоти плескань та до нижчої частоти коливань пружних стінок. Досліджено ефект параметричного резонансу. З'ясовано вплив наявності перегородок на рух оболонки з рідиною при вільних та вимушених коливаннях.

Ключові слова: оболонки обертання, пружність, коливання, комбіновані навантаження, методи скінченних та граничних елементів, ідеальна нестислива рідина.

Hydroelastic vibrations of shells of revolution under combined vertical and horizontal loadings

Usatova Olga*PhD student**A. Podgorny Institute of Mechanical Engineering Problems NASU, Pozharsky,
st. 2/10, Kharkiv, 61046, Ukraine***Kriutchenko Denis***leading engineer**A. Podgorny Institute of Mechanical Engineering Problems NASU, Pozharsky,
st. 2/10, Kharkiv, 61046, Ukraine*

The operation of modern structures occurs under conditions of high technological loads. Shells of rotation are among the most applicable structural elements. The operating conditions of such elements provide for the influence of the environment such as the interaction of elastic shell elements with liquid or gas. The oscillations of the elastic shell of rotation, which are partially filled with liquid, under external loads in the vertical and horizontal directions simultaneously, are considered in the paper. The liquid is supposed to be ideal and incompressible, and its motion under the applied loads is irrotational. This allows us to introduce the velocity potential to calculate the pressure of the fluid on the surface of the shell and determine the function which describes the level of the free surface lift. At the first stage, the frequencies and forms of free oscillations of the shell with liquid are determined. These frequencies are usually divided into two separate spectra: the free surface vibration frequency and the elastic wall vibration frequency. The boundary element method is used for numerical implementation. It is found that for sufficiently thin elastic tank walls, the fundamental frequency of the coupled vibrations can be much lower than the frequency of the fluid in a shell with rigid walls. As the tank wall thickness increases, this effect becomes insignificant, and the lower vibration frequency of the shell with liquid approaches the vibration frequency of the liquid in a rigid tank. The

second stage of the study is related to the analysis of combined horizontal and vertical loads. The cases when the load frequencies are close to the fundamental frequency of sloshing and to the lower frequency of vibration of elastic walls are considered. The effect of parametric resonance has been investigated. The influence of the presence of partitions on the motion of the shell with the liquid at free and forced oscillations has been found out.

Keywords: shell rotation, elasticity, oscillations, combined loads, finite and boundary element methods, ideal incompressible fluid.

1 Вступ

Значна кількість елементів сучасного обладнання в енергетичній, хімічній промисловості, на транспорті, в авіаційній та космічній техніці працює в умовах підвищених технологічних навантажень, при високих температурах та рівнях тиску. При цьому зазвичай обладнання піддається дії зовнішніх навантажень різної природи. Важливими питаннями тут є гідро та аеро-пружні ефекти. Ці явища притаманні процесам коливань кришок гідротурбін [1], робочих коліс гідротурбін [2], лопатей повітряних установок [3]. Гідродинамічні ефекти спостерігаються й при дослідженні явищ підтоплення міських територій [4], Гідропоужні явища необхідно враховувати і при проектуванні та модернізації резервуарів та сховищ для вогнебезпечної та легкозаймистих речовин. Як моделі цих резервуарів зазвичай використовують оболонки обертання [5], [6] та складені оболонки обертання [7]. Вплив різного виду перегородок на частоти та амплітуди коливань резервуарів вивчався в роботах [5], [8], [9], [10]. Вимушені коливання резервуарів при частковому заповненні рідиною вивчались в [5], [11], [12]. Зазначимо, що вивчення вимушених коливань рідини в резервуарах в уточненому формулюванні залишається актуальним питанням, оскільки дає змогу визначити границі стійкості руху, що дозволяє забезпечити надійність та безпечність функціонування обладнання.

2 Формулювання задачі та основні співвідношення

Для оцінки ефектів пружності стінок оболонки при коливаннях конструкцій з рідиною розроблена математична модель, що базується на таких гіпотезах: рідина є нестисливою і нев'язкою, рух рідини є безвихровим, враховуються лише малі коливання пружних елементів [12]. Введемо потенціал цієї зв'язаної задачі як суму двох функцій

$$\tilde{\Phi}(t, x, y, z) = \Phi(t, x, y, z) + \Phi_e(t, x, y, z), \quad (2.1)$$

де потенціал $\Phi(t, x, y, z)$ визначає коливання вільної поверхні, а функція $\Phi_e(t, x, y, z)$ описує внесок пружних коливань стінки.

Для функції $\Phi_e(t, x, y, z)$ сформульовано таку крайову задачу [12]:

$$\nabla^2 \Phi_e = 0, \quad \left. \frac{\partial \Phi_e}{\partial \mathbf{n}} \right|_{S_r} = 0, \quad \left. \frac{\partial \Phi_e}{\partial \mathbf{n}} \right|_{S_e} = \frac{\partial W}{\partial t}, \quad \left. \frac{\partial \Phi_e}{\partial t} \right|_{S_0} = 0, \quad W = (\mathbf{U}, \mathbf{n}), \quad \iint_{S_0} \frac{\partial \Phi_e}{\partial \mathbf{n}} dS_0 = 0. \quad (2.2)$$

Тут змочена поверхня оболонки подана як $S_1 = S_r \cup S_e$, де S_r і S_e є жорсткою та пружною частинами поверхні оболонки відповідно, $\mathbf{U} = \mathbf{U}(x, y, z, t)$ - вектор пружного переміщення, і W є його нормальним компонентом.

Нехай $\mathbf{u}_{nk}(r, z)$, $k = 1, 2, \dots$ - власні форми коливань пружної порожньої оболонки, які відповідають хвильовому числу n . Для числового аналізу використовуємо метод редукованих скінченних елементів, описаний у [5]. Вектор-функція в циліндричних координатах $\mathbf{U} = \mathbf{U}(\theta, r, z, t)$ може бути подана таким чином:

$$\mathbf{U} = \sum_{n=0}^N \cos n \theta \sum_{k=1}^{M_1} c_{nk}(t) \mathbf{u}_{nk}(r, z). \quad (2.3)$$

Зобразимо функцію $\Phi_e(t, x, y, z)$ у циліндричних координатах як ряд

$$\Phi_e = \sum_{n=0}^N \cos n \theta \sum_{k=1}^{M_1} \dot{c}_{nk}(t) \varphi_{nk}^e(r, z), \quad (2.4)$$

де базисні функції φ_{nk}^e визначаються з граничних задач

$$\nabla^2 \varphi_{nk}^e = 0, \quad \left. \frac{\partial \varphi_{nk}^e}{\partial \mathbf{n}} \right|_{S_r} = 0, \quad \left. \frac{\partial \varphi_{nk}^e}{\partial \mathbf{n}} \right|_{S_e} = (\mathbf{u}_{nk}, \mathbf{n}), \quad \varphi_{nk}^e|_{S_0} = 0. \quad (2.5)$$

Для функції $\Phi(x, y, z, t)$ згідно з [13] маємо такий вираз,

$$\Phi = \sum_{n=0}^N \cos n\theta \sum_{k=1}^{M_1} \dot{d}_{nk}(t) \phi_{nk}(r, z)$$

Таким чином, для сумарного потенціалу $\tilde{\Phi}(t, \theta, r, z)$ маємо подання

$$\tilde{\Phi}(t, \theta, r, z) = \sum_{n=0}^N \cos n\theta \left[\sum_{k=1}^M \dot{d}_{nk}(t) \phi_{nk}(r, z) + \sum_{k=1}^{M_1} \dot{c}_{nk}(t) \phi_{nk}^e(r, z) \right]. \quad (2.6)$$

Аналогічно, для функції $\zeta(t, \theta, r)$ що описує положення та форму вільної поверхні, отримаємо

$$\zeta = \sum_{n=0}^N \cos n\theta \frac{1}{g} \left[\sum_{k=1}^M \chi_{nk}^2 d_{nk}(t) \phi_{nk}(r, 0) + \sum_{k=1}^{M_1} c_{nk}(t) \frac{\partial \phi_{nk}^e(r, z)}{\partial \mathbf{n}} \Big|_{z=0} \right]. \quad (2.7)$$

Крайові задачі (2.2) та (2.5) розв'язані методом граничних елементів з використанням спеціальних процедур, що описані в [13], [14]

Отже, для функції $\tilde{\Phi}(t, \theta, r, z)$ згідно з (2.6), (2.7) маємо такі співвідношення:

$$\nabla^2 \tilde{\Phi} = 0, \quad \frac{\partial \tilde{\Phi}}{\partial \mathbf{n}} \Big|_{S_r} = 0, \quad \frac{\partial \tilde{\Phi}}{\partial \mathbf{n}} \Big|_{S_e} = \frac{\partial W}{\partial t}, \quad \frac{\partial \tilde{\Phi}}{\partial \mathbf{n}} = \frac{\partial \zeta}{\partial t} \Big|_{S_0} = 0.$$

Тому для розв'язання зв'язаної задачі з визначення коливань пружної оболонки, заповненою рідиною, необхідно обчислити невідомі коефіцієнти, що залежать лише від часу

$$c_{nk}(t), d_{nl}(t), n = 0, 1, \dots, N, k = 1, 2, \dots, M, k = 1, 2, \dots, M_1.$$

Для знаходження цих коефіцієнтів використано динамічну граничну умову на вільній поверхні [15]

$$\frac{\partial \tilde{\Phi}}{\partial t} + a_x(t)x + (g + a_z(t))\zeta = 0.$$

Використовуючи рівняння (2.6), (2.7), знаходимо

$$\begin{aligned} & \sum_{n=0}^N \cos n\theta \left[\sum_{k=1}^M \ddot{d}_{nk}(t) \phi_{nk}(r, 0) + \sum_{k=1}^{M_1} \ddot{c}_{nk}(t) \phi_{nk}^e(r, 0) \right] + a_x(t)r \cos \theta + \\ & + (g + a_z(t)) \sum_{n=0}^N \cos n\theta \left[\frac{1}{g} \sum_{k=1}^M \chi_{nk}^2 d_{nk}(t) \phi_{nk}(r, 0) + \sum_{k=1}^{M_1} c_{nk}(t) \frac{\partial \phi_{nk}^e(r, 0)}{\partial \mathbf{n}} \right] = 0 \end{aligned} \quad (2.8)$$

Ще одно співвідношення для визначення невідомих, залежних від часу коефіцієнтів отримано з рівняння руху оболонки у вигляді [12]

$$\mathbf{L}\mathbf{U} + \mathbf{M}\ddot{\mathbf{U}} = p_d \mathbf{n}, \quad p_d = -\rho_l \left[\frac{\partial \tilde{\Phi}}{\partial t} + a_x(t)x + (g + a_z(t))\zeta \right], \quad (2.9)$$

тут \mathbf{L} та \mathbf{M} - матриці жорсткості та мас, відповідно.

Нехай Ω_{nk} є власними частотами порожньої пружної оболонки.

Маємо співвідношення

$$\mathbf{L}\mathbf{u}_{nk} = \Omega_{nk}^2 \mathbf{M}\mathbf{u}_{nk}, \quad (\mathbf{M}\mathbf{u}_{nk}, \mathbf{u}_{nj}) = \delta_{kj}, \quad (\mathbf{L}\mathbf{u}_{nk}, \mathbf{u}_{nj}) = \Omega_{nk}^2 \delta_{kj}. \quad (2.10)$$

Використав рівняння (2.7)-(2.10), приходимо до такого співвідношення:

$$\begin{aligned} & \mathbf{L} \left\{ \sum_{n=0}^N \cos n\theta \left[\sum_{k=1}^{M_1} c_{nk}(t) \phi_{nk}^e(r, z) \right] \right\} + \mathbf{M} \left\{ \sum_{n=0}^N \cos n\theta \left[\sum_{k=1}^{M_1} \ddot{c}_{nk}(t) \phi_{nk}^e(r, z) \right] \right\} = \\ & = -\rho_l \sum_{n=0}^N \cos n\theta \left[\sum_{k=1}^M \ddot{d}_{nk}(t) \phi_{nk}(r, z) + \sum_{k=1}^{M_1} \ddot{c}_{nk}(t) \phi_{nk}^e(r, z) \right] + a_x(t)r \cos \theta + \\ & + (g + a_z(t)) \sum_{n=0}^N \cos n\theta \left[\sum_{k=1}^M d_{nk}(t) \frac{\partial \phi_{nk}(r, z)}{\partial \mathbf{n}} + \sum_{k=1}^{M_1} c_{nk}(t) \frac{\partial \phi_{nk}^e(r, z)}{\partial \mathbf{n}} \right] \end{aligned} \quad (2.11)$$

Рівняння (2.8), (2.11) описують зв'язану задачу визначення коливань оболонки з рідиною при одночасному врахуванні пружності стінок та коливань вільної поверхні рідини.

Враховуючи результати скалярного добутку рівняння (2.10) на функції $\cos n\theta \phi_{nk}$ і рівняння (2.11) на функції \mathbf{u}_{nk} , та ортогональність форм власних коливань рідини в жорстких резервуарах

[16], отримаємо систему диференціальних рівнянь другого порядку для визначення невідомих коефіцієнтів $c_{nk}(t)$, $d_{nl}(t)$.

Якщо розглядаються задачі власних коливань, вважаємо, що

$$c_{nk}(t) = C_{nk} \exp(i\omega t), \quad d_{nk}(t) = D_{nk} \exp(i\omega t), \quad a_x(t) = 0, \quad a_z(t) = 0. \quad (2.12)$$

Тоді з рівнянь (10), (12) отримуємо задачу на власні значення для визначення власних частот і форм з урахуванням як пружності, так і ефектів плескань вільної поверхні. Тут ω - власна частота, C_{nk}, D_{nk} - невідомі константи.

3 Визначення власних частот

Як приклад числового моделювання розглянемо циліндричну оболонку з плоским дном, радіусом $R = 1m$, товщиною $h = 0.01m$, висотою $L = 2m$, з модулем Юнга $E = 2 \cdot 10^5 MPa$, коефіцієнтом Пуассона $\nu = 0.3$, густиною матеріалу оболонки $\rho_s = 7800 kg/m^3$, густиною рідини $\rho_l = 1000 kg/m^3$, рівнем заповнення $H = 1.0m$. Положення перегородки приймаємо як $H_l = 0.5m$, радіус перегородки R_b є змінним, значення $R_b = 0$ відповідає резервуару без перегородки. Передбачається, що оболонка жорстко закріплена за контуром контуром, тобто маємо такі граничні умови: $u_r = u_z = u_\theta = 0$ при $z = -H$ і $r = R$, рис.3.1. Власні форми коливань порожньої оболонки отримано за допомогою методу скінченних елементів, як описано в [15] (це перша система базисних функцій).

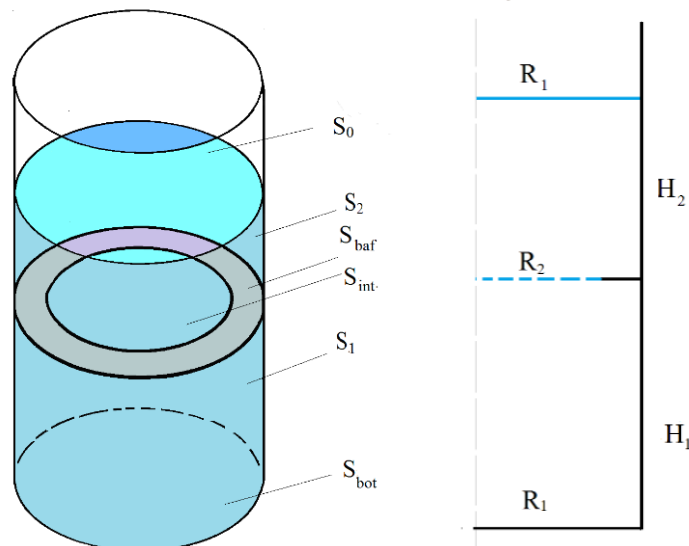


Рисунок 3.1 Циліндричний резервуар з перегородкою

Форми заповненої рідиною пружної оболонки без урахування сили тяжіння Φ_{nk}^e (друга система базисних функцій) знайдені методом, розробленим в [17]. Базисні функції Φ_{nk} що описують коливання вільної поверхні знаходимо з використанням методу інтегральних рівнянь [14]. Коли всі системи базових функцій визначені, підставляємо їх у рівняння (2.10), (2.11) та з використанням (2.12) приходимо до розв'язання задачі на власні значення. При цьому враховуються як коливання пружних стінок оболонки, так і плескання рідини.

Далі досліджено вплив наявності перегородок на власні частоти коливань системи «оболонка-рідина».

У табл. 3.1 наведені числові значення власних частот коливань для порожніх та заповнених рідиною циліндричних пружних резервуарів без перегородок. Тут коефіцієнти n_s, n_l вказують на кількість форм коливань стінок оболонки та форм плескань рідини, що враховуються в зв'язаних вібраціях, K - номер зв'язаної форми коливань. Для числового моделювання використано чотири форми коливань оболонки та п'ять форм плескань.

Таблиця 3.1 – Частоти порожніх та наповнених рідиною пружних резервуарів без перегородок, $n = 0, 1$, Гц

$n = 0$					$n = 1$			
K	n_S	n_L	Оболонка без рідини	Оболонка з рідиною	n_S	n_L	Оболонка без рідини	Оболонка з рідиною
1		1		0.9739		1		0.6418
2		2		1.3208		2		1.1509
3		3		1.5909		3		1.4564
4		4		1.8209		4		1.7054
5		5		2.0249		5		1.9212
6	1	1,2	23.233	7.6591	1,2		48.520	21.902
7	2,1		91.101	43.308	2,1		139.70	79.712
8	3,2		205.25	117.03	3,2,1		232.44	178.42
9	4,3,2		365.79	230.31	4,3		277.30	210.00

Наведені тут результати показують різницю між частотами заповнених рідиною та порожніх оболонок. Але зі збільшенням номера частоти ця різниця поступово зменшується. Частоти резервуарів з пружними стінками істотно перевищують частоти, пов'язані з плесканнями.

Результати впливу перегородки на частоти коливань наведені в табл. 2. Перегородка, що має форму кільця з $R_b = 0.5$ м, встановлюється на висоті $H_l = 0.5$ м в циліндричну оболонку, параметри якої описані вище, рис. 3.1.

Таблиця 3.2 – Частоти порожніх та наповнених рідиною пружних резервуарів із перегородками, $\alpha = 0$, Гц 4 Відомості про авторів та організацію.

K	n_S	n_L	Оболонка без рідини	Оболонка з рідиною
1		1		0.9475
2		2		1.3028
3		3		1.5900
4		4		1.8209
5		5		2.0249
6	1	1	23.2338	5.5213
7	2	1	40.4818	15.172
8	3	2	91.1015	43.769
9	4	2	205.253	119.145
10	5	2	213.551	168.052
11	6	1	365.794	196.125
12	7	2	553.183	370.881
13	8	3	572.280	401.324

Наведені тут результати показують різницю між частотами заповнених рідиною та порожніх оболонок. Але зі збільшенням номера частоти ця різниця поступово зменшується. Частоти резервуарів з пружними стінками істотно перевищують частоти, пов'язані з плесканнями.

У порівнянні з даними табл. 3.1 бачимо вплив наявності перегородки.

Форми осесиметричних коливань циліндричної оболонки з кільцевою перегородкою зображені на рис. 3.2.

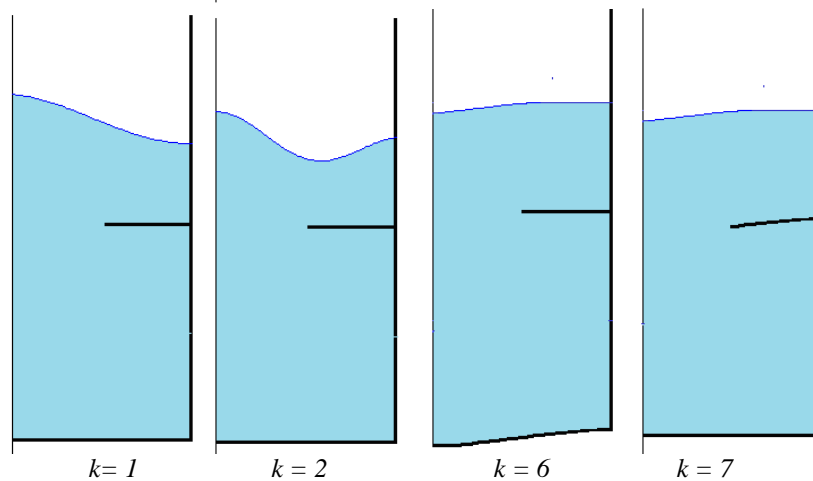


Рисунок 3.2: Форми осесиметричних коливань оболонки з кільцевою перегородкою

Порівнюючи частоти для кругових циліндричних резервуарів із перегородками і без перегородок, можна побачити, що найнижчі частоти осесиметричних коливань для обох резервуарів практично однакові. Нижча частота ω_{06} для обох порожніх пружних резервуарів $\omega_{06} = 23,233$ Гц, і її форма відповідає першій формі осесиметричних коливань днища. Частота ω_{07} для порожнього резервуара та частота ω_{08} для резервуара з перегородкою становлять 91.1015 Гц і відповідають другій формі осесиметричних коливань днища.

Слід зазначити, що встановлення перегородки призводить до більш істотних змін частоти коливань днища. Частота ω_{07} відповідає першій осесиметричній формі перегородки. Подібно до резервуара без перегородок, частоти резервуарів з пружними стінками, істотно перевищують частоти, пов'язані з плесканнями.

Зазначимо також, що зменшення товщини оболонки призводить до зменшення частот пружної оболонки. Частоти порожніх та заповнених рідиною резервуарів з кільцевою перегородкою для різної товщини наведені в табл. 3.3.

Таблиця 3.3: Залежність частот від товщини оболонки

K	Пуста оболонка, Гц				Оболонка з рідиною, Гц			
	h, м							
	0.01	0.005	0.003	0.0015	0.01	0.005	0.003	0.0015
6	23.233	11.838	7.1805	3.6308	5.5213	2.8187	1.7096	0.8644
7	40.482	20.612	12.495	6.3132	15.172	7.9277	4.8058	1.5780
8	91.101	46.271	28.023	14.153	43.769	22.249	13.479	7.0064
9	205.25	100.01	62.922	31.747	119.14	58.148	36.587	15.716
10	213.55	109.10	66.127	31.778	168.05	85.909	52.069	18.692

Відповідно до рис. 3.1, у табл. 3.3 номер $K = 6$ відповідає першій осесиметричній формі коливань пружного днища, а $K = 7$ відповідає першій осесиметричній формі коливань пружної перегородки. Частоти, наведені в табл. 3.3 для товщини $h = 0.01\text{ м}$, $h = 0.005\text{ м}$, $h = 0.003\text{ м}$, вище ніж частоти плескань. Результати табл. 5.3 свідчать про те, що найнижчі частоти пружних оболонок зменшуються зі зменшенням товщини оболонки. Отже, для дуже тонких пружних стінок резервуара фундаментальна частота при зв'язаних коливаннях може бути набагато меншою, ніж частота рідини в оболонці з жорсткими стінками. Зі збільшенням товщини стінки резервуара цей ефект стає незначним, а нижча частота коливань оболонки з рідиною наближається до частоти коливань рідини в жорсткому резервуарі. Але використання таких тонких оболонок як елементів відповідальних конструкцій, що працюють при інтенсивних зовнішніх навантаженнях, вимагає ретельного аналізу напружено-деформованого стану, щоб уникнути втрати стійкості.

У роботі [18] були отримані умови, коли вплив плескань стає незначним при вивченні коливань пружної оболонки. Таким чином, для оцінки найнижчих частот коливань оболонки, заповненої рідиною, доцільно обмежити дослідження жорсткими оболонками, принаймні, якщо відношення товщини до характерного розміру оболонки більше, ніж 0.003.

4 Дослідження взаємної дії горизонтальних та вертикальних навантажень на пружний резервуар частково заповнений рідиною

Розглядається задача про вимушені коливання рідини в пружній оболонці. Передбачається, що рідина ідеальна, нестислива, а її рух, що почався зі стану спокою, є безвихровим. Позначимо змочену поверхню оболонки через S_1 , а вільну поверхню через S_0 . Систему рівнянь руху оболонкової конструкції з відсіками, частково заповненою рідиною, та інтеграл Коші-Лагранжа, застосовуємо у вигляді (2.9).

Будемо шукати власні форми коливань резервуару з рідиною у вигляді (2.3).

Приходимо до системи диференціальних рівнянь (2.8), (2.11) для знаходження невідомі функції часу $c_k(t)$ та $d_k(t)$. Для їх однозначного визначення використовуємо початкові умови

$$c_k(0) = c_{k0}, \quad \dot{c}_k(0) = c_{k1}, \quad d_k(0) = d_{k0}, \quad \dot{d}_k(0) = d_{k1}.$$

Після виконання скалярного добутку отримаємо

$$c_l(t) + \Omega_l^2 c_l(t) = -\rho_l \sum_{k=1}^N \ddot{c}_k(t) (\phi_{1k}, u_l) + \sum_{k=1}^M \ddot{d}_k(t) (\phi_{2k}, u_l) + a_x(t)(x, u_l) + a_z(t)(z, u_l) = 0, \quad (4.1)$$

$$\ddot{d}(\varphi_{2l}, \varphi_{2l}) + \chi_k^2 d_l(\varphi_{2l}, \varphi_{2l}) + (a_z(t) + g) \chi_k^2 d_l(\varphi_{2l}, \varphi_{2l}) + \sum_{k=1}^N c_k \left(\frac{\partial \phi_{1k}}{\partial n}, \varphi_{2l} \right) + a_x(t)(x, \varphi_{2l}) = 0. \quad (4.2)$$

Розв'язок системи (4.1) - (4.2) дає змогу проаналізувати вимушені коливання системи «оболонка-рідина» з різними силами, що збуджують.

При числовому моделюванні розглянуто циліндричну оболонку з плоским днищем, з параметрами, описаними в підрозділі 3.

З наведених в таблиці 3.2 даних щодо плескань рідини у резервуарі, бачимо, що найнижчі частоти є 0.64 Гц та 0.97 Гц. Ці частоти відповідають першій та нульовій гармонікам. Дані таблиці 3.1 свідчать про те, що найнижча частота коливань стінок циліндричного резервуара є 0.947 Гц.

Будемо розглядати одночасну дію горизонтальних та вертикальних навантажень. Припустимо, що

$$a_x(t) = a_h \cos \omega_h t, \quad a_z(t) = a_v \cos \omega_v t.$$

Зауважимо, що нижча частота плескань відповідає першій гармоніці, а нижча частота коливань пружних стінок відповідає осесиметричним коливанням днища. З цього випливає, що $(x, u_1) = 0$ оскільки $x = \rho \cos \theta$. Тому в рівняннях (4.1) буде наявною лише дія вертикального збудження. Але в рівняннях (4.2) будуть ненульовими всі доданки, що відповідають за вплив зовнішнього навантаження.

Припустимо, що $\omega_h = \omega_v = 0.945$ Гц. Обчислимо функції $d_l, c_l, l = 0, 1$ та побудуємо графік зміни рівня вільної поверхні резервуару в точці з координатами $z=H, R=1, \theta=0$.

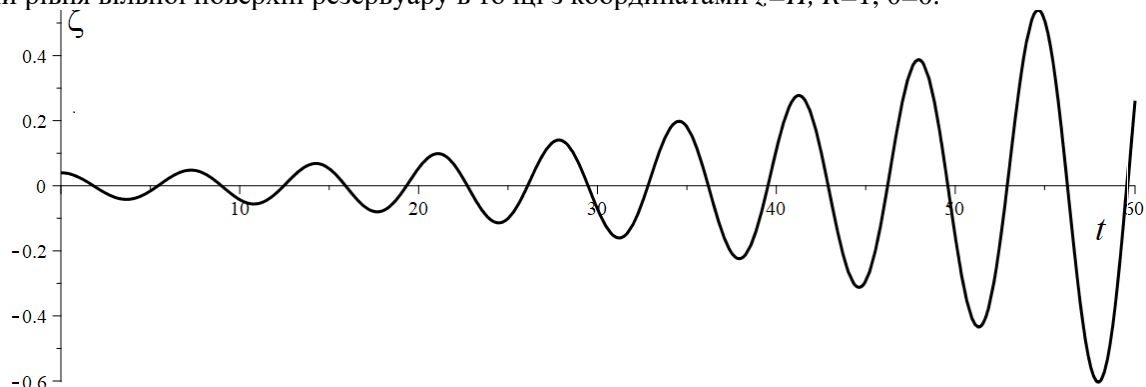


Рисунок 4.1: Рівень підйому вільної поверхні за умови що $\omega_h = \omega_v = 0.945$ Гц.

Заголовки повинні набиратися 11 кеглем шрифту Times New Roman, вирівнюватися по лівому краю. Далі розглянемо випадок, коли $\omega_h = \omega_v = 0.969$ Гц. Зміна рівня вільної поверхні за часом подана на рис. 4.2.

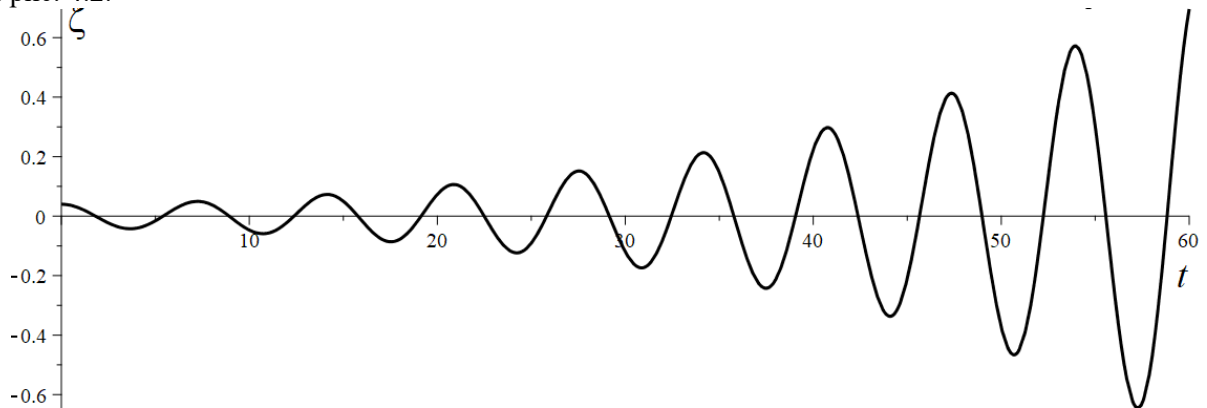


Рисунок 4.2: Рівень підйому вільної поверхні за умови що $\omega_h = \omega_v = 0.969$ Гц

З даних, наведених на рис. 4.1 та 4.2, бачимо що наближення частоти сил, що змушують, до нижчої частоти коливань пружних стінок призводить до втрати стійкості руху.

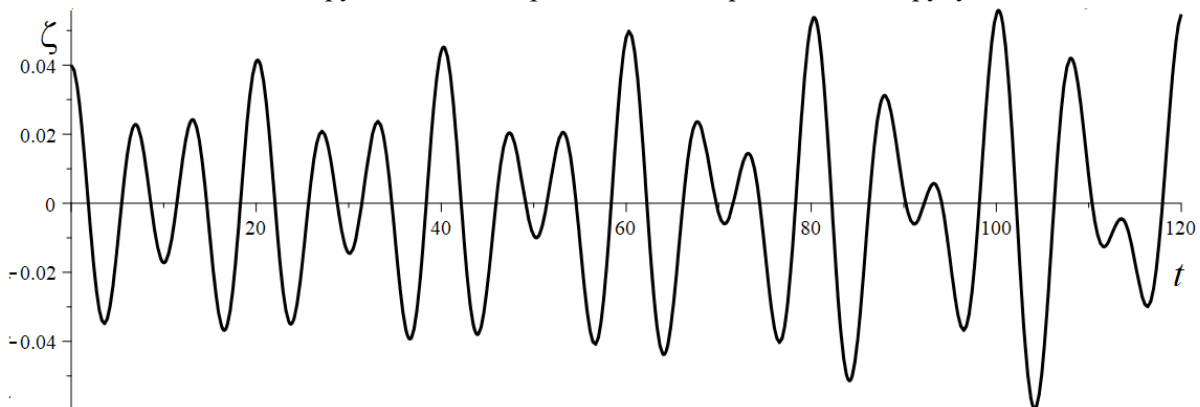


Рисунок 4.3: Рівень підйому вільної поверхні за умови що $\omega_h = \omega_v = 0.639$ Гц

На рис. 4.3. наведено графік зміни рівня вільної поверхні рідини за умови наближення частоти сил, що змушують, до першої фундаментальної частоти плескань.

Рис.4.4. відповідає тій частоті сил, що змушують, що дорівнює подвоєній фундаментальній.



Рисунок 4.4: Рівень підйому вільної поверхні за умови що $\omega_h = \omega_v = 1.28$ Гц.

Рис. 4.5 демонструє зміну рівня вільної поверхні при частоті, яка дорівнює 0.3 Гц, та не є близькою ані до нижчих частот плескань, ані до нижчої частоти коливань пружних стінок.

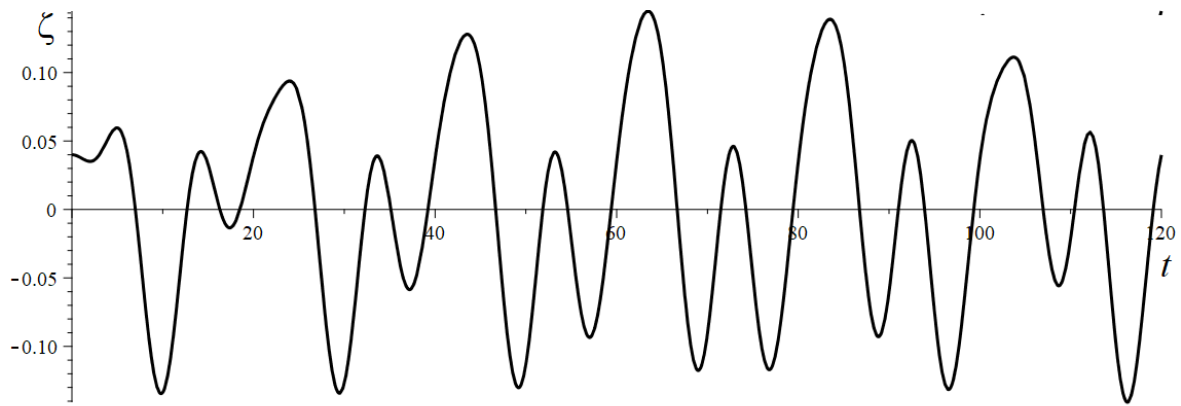


Рисунок 4.5: Рівень підйому вільної поверхні за умови що $\omega_h = \omega_v = 0.3$ Гц.

З наведених результатів можна зробити висновок про те, що в випадку досить тонких оболонок, коли нижча частота коливань пружних стінок стає приблизно рівною нижчим частотам плескань, відбувається втрата стійкості. В цьому випадку спектри частот коливань пружних стінок та коливань вільної поверхні рідини не є відокремленими, і впливом пружності стінок неможливо нехтувати.

5 Висновки

Розглянуто коливання пружної оболонки обертання, частково заповненої рідиною, за умови дії зовнішніх навантажень в вертикальному та горизонтальному напрямках одночасно. Встановлено, що для досить тонких пружних стінок резервуара фундаментальна частота при зв'язаних коливаннях може бути набагато меншою, ніж частота рідини в оболонці з жорсткими стінками. Зі збільшенням товщини стінки резервуара цей ефект стає незначним, а нижча частота коливань оболонки з рідиною наближається до частоти коливань рідини в жорсткому резервуарі. Розглянуто випадки, коли частоти навантажень є близькими до фундаментальної частоти плескань та до нижчої частоти коливань пружних стінок. Встановлено, що спектри частот коливань пружних стінок та коливань вільної поверхні рідини не є відокремленими, і впливом пружності стінок неможливо нехтувати.

ЛІТЕРАТУРА

1. Місюра С. Ю., Сметанкіна Н. В., Місюра Є. Ю. Рациональное моделирование крышки гидротурбины для анализа мощности. Вісн. Нац. техн. ун-ту «ХПІ». Сер. Динаміка і міцність машин. 2019. № 1. С. 34–39. <http://repository.kpi.kharkov.ua/handle/KhPI-Press/44370>
2. Ганчин Е.В., Ржевская И.Е., Стрельникова Е.А. Исследование динамических характеристик лопастей рабочих колес поворотно-лопастных гидротурбин при взаимодействии с жидкостью. Вісник Харківського національного університету, 2009. – № 847. – С. 79-86. <http://mia.univer.kharkov.ua/11/30078.pdf>
3. Дегтярев К.Г., Стрельникова Е. А., Шелудько Г. А. Компьютерное моделирование лопастей ветроустановок с оптимальными параметрами / Вісн. Харківського нац. ун-ту імені В.Н. Каразіна. Серія: Математичне моделювання. Інформаційні технології. Автоматизовані системи управління, No 19, 2012, С.81-86 <http://mia.univer.kharkov.ua/19/30251.pdf>
4. Serikova E., Strelnikova E., Yakovlev V. Mathematical model of dangerous changing the groundwater level in Ukrainian industrial cities. Journal of Environment Protection and Sustainable Development. 2015. Vol.1. P. 86-90. URL: <https://www.researchgate.net/publication/281784323>
5. Strelnikova E., Gnitko V., Krutchenko D., Naumemko Y. Free and forced vibrations of liquid storage tanks with baffles J. Modern Technology & Engineering Vol.3, No.1, 2018, pp.15-52 <http://jomardpublishing.com/UploadFiles/Files/journals/JTME/V3No1/StrelnikovaE.pdf>
6. Karaiev A., Strelnikova E. Liquid Sloshing in Circular Toroidal and Coaxial Cylindrical Shells. In: Ivanov V., Pavlenko I., Liaposhchenko O., Machado J., Edl M. (eds) Advances in Design, Simulation and Manufacturing III. DSMIE 2020. Lecture Notes in Mechanical Engineering. Springer, Cham, 2020. P.3-13. DOI: 10.1007/978-3-030-50491-5_1

7. Дегтярев К.Г., Гнисько В.И., Тонконоженко А.М. Компьютерное моделирование разрушающей нагрузки на топливный бак. Вісник Харківського національного університету імені ВН Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління» N 1105, 2014, С.51-59. http://nbuv.gov.ua/UJRN/VKKhIMAM_2014_1105_24_7
8. Behshad, A., Shekari, M.: A boundary element study for evaluation of the effects of the rigid baffles on liquid sloshing in rigid containers. International Journal of Maritime Technology № 10, 2018, pp.45-54, DOI: 10.29252/ijmt.10.45.
9. Mohammad Yaghoob Abdollahzadeh Jamalabadi. Analytical Solution of Sloshing in a Cylindrical Tank with an Elastic Cover. Mathematics 2019, 7(11), 1070; <https://doi.org/10.3390/math7111070>
10. Крютченко Д. Моделювання коливань рідини в призматичних резервуарах з хрестовими перегородками. Вісник Харківського національного університету імені ВН Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління» N 44, 2019, С. 43-50. <https://doi.org/10.26565/2304-6201-2019-44-05>
11. Wenyuan Wang, Quansheng Zang, Zhijun Wei, Zijian Guo, An isogeometric boundary element method for liquid sloshing in the horizontal eccentric annular tanks with multiple porous baffles, Ocean Engineering, Volume 189, 2019, 106367, <https://doi.org/10.1016/j.oceaneng.2019.106367>
12. Strelnikova E.A., Choudhary N., Kriutchenko D.V., Gnitko V.I., Тонконоженко А.М. Liquid vibrations in circular cylindrical tanks with and without baffles under horizontal and vertical excitations, Engineering Analysis with Boundary Elements, 2020, 120, pp. 13–27. DOI: 10.1016/j.enganabound.2020.07.024
13. Naumenko V.V., Strelnikova H.A. Singular integral accuracy of calculations in two-dimensional problems using boundary element methods. Engineering analysis with boundary elements. №26, pp. 95-98, 2002. DOI: 10.1016/S0955-7997(01)00041-8
14. Karaiev A., Strelnikova E. Singular integrals in axisymmetric problems of elastostatics. International Journal of Modeling, Simulation, and Scientific Computing. □ 2020 □ Vol. 11, № 1, 2050003. DOI: 10.1142/S1793962320500038
15. Gnitko V., Degtyariov K., Karaiev A. and Strelnikova E. Multi-domain boundary element method for axisymmetric problems in potential theory and linear isotropic elasticity, WIT Transactions on Engineering Sciences, Vol. 122, WIT Press, pp.13-25, 2019. DOI: 10.2495/BE410021
16. Raynovskyy, I.A., Timokha, A.N.: Sloshing in Upright Circular Containers: Theory, Analytical Solutions, and Applications. CRC Press/Taylor & Francis Group, (2020). 170 pp.. ISBN 9780367362898
17. Ravnik, J., Strelnikova, E., Gnitko V., Degtyarev, K., Ogorodnyk, U., BEM and FEM analysis of fluid-structure interaction in a double tank, Engineering Analysis with Boundary Elements, 67, pp. 13-25, 2016. <https://doi.org/10.1016/j.enganabound.2016.02.006>
18. Ibrahim, R. A., Pilipchuk, V. N., and Ikeda, T. (March 1, 2001). "Recent Advances in Liquid Sloshing Dynamics." ASME. Appl. Mech. Rev. March 2001; 54(2): 133–199. <https://doi.org/10.1115/1.3097293>

REFERENCES

1. Missouri C. Yu., Smetankina NV, Missouri E. Yu. Rational modeling of a hydroturbine cover for strength analysis. Visn. Nat. tech. KhPI University. Ser. Dynamics and strength of machines. 2019. № 1. S. 34–39. <http://repository.kpi.kharkov.ua/handle/KhPI-Press/44370> [in Ukrainian]
2. Ganchin EV, Rzhetskaya IE, Strelnikova EA Investigation of the dynamic characteristics of the impellers of rotary-vane hydraulic turbines when interacting with a liquid. Bulletin of Kharkiv National University, 2009. - № 847. - P. 79-86. <http://mia.univer.kharkov.ua/11/30078.pdf> [in Ukrainian]
3. Degtyarev KG, Strelnikova EA, Sheludko GA Computer modeling of wind turbine blades with optimal parameters / Visn. Kharkiv National University VN University Karazina. Series: Mathematical modeling. Information Technology. Automated control systems, No 19, 2012, P.81-86 <http://mia.univer.kharkov.ua/19/30251.pdf> [in Ukrainian]
4. Serikova E., Strelnikova E., Yakovlev V. Mathematical model of dangerous changing the groundwater level in Ukrainian industrial cities. Journal of Environment Protection and Sustainable

- Development. 2015. Vol.1. P. 86-90. URL: <https://www.researchgate.net/publication/281784323> [in English]
5. Strelnikova E., Gnitko V., Krutchenko D., Naumemko Y. Free and forced vibrations of liquid storage tanks with baffles J. Modern Technology & Engineering Vol.3, No.1, 2018, pp.15-52 <http://jomardpublishing.com/UploadFiles/Files/journals/JTME/V3No1/StrelnikovaE.pdf> [in English]
 6. Karaiev A., Strelnikova E. Liquid Sloshing in Circular Toroidal and Coaxial Cylindrical Shells. In: Ivanov V., Pavlenko I., Liaposhchenko O., Machado J., Edl M. (eds) Advances in Design, Simulation and Manufacturing III. DSMIE 2020. Lecture Notes in Mechanical Engineering. Springer, Cham, 2020. P.3-13. DOI: 10.1007/978-3-030-50491-5_1 [in English]
 7. Degtyarev K.G., Gnitko V.I., Tonkonozhenko A.M. Computer simulation of the destructive load on the fuel tank. Bulletin of Kharkiv National University named after VN Karazin, series "Mathematical modeling. Information Technology. Automated control systems "N 1105, 2014, P.51-59. http://nbuv.gov.ua/UJRN/VKhimAM_2014_1105_24_7 [in Ukrainian]
 8. Behshad, A., Shekari, M.: A boundary element study for evaluation of the effects of the rigid baffles on liquid sloshing in rigid containers. International Journal of Maritime Technology № 10, 2018, pp.45-54, DOI: 10.29252/ijmt.10.45 [in English]
 9. Mohammad Yaghoub Abdollahzadeh Jamalabadi. Analytical Solution of Sloshing in a Cylindrical Tank with an Elastic Cover. Mathematics 2019, 7(11), 1070; <https://doi.org/10.3390/math7111070> <https://periodicals.karazin.ua/mia/issue/download/773/940> [in English]
 10. Kryutchenko D. Modeling of fluid oscillations in prismatic tanks with cross partitions. Bulletin of Kharkiv National University named after VN Karazin, series "Mathematical modeling. Information Technology. Automated control systems "N 44, 2019, P. 43-50. <https://doi.org/10.26565/2304-6201-2019-44-05> [in Ukrainian]
 11. Wenyuan Wang, Quansheng Zang, Zhijun Wei, Zijian Guo, An isogeometric boundary element method for liquid sloshing in the horizontal eccentric annular tanks with multiple porous baffles, Ocean Engineering, Volume 189, 2019, 106367, <https://doi.org/10.1016/j.oceaneng.2019.106367>[in English]
 12. Strelnikova E.A., Choudhary N., Kriutchenko D.V., Gnitko V.I., Tonkonozhenko A.M. Liquid vibrations in circular cylindrical tanks with and without baffles under horizontal and vertical excitations, Engineering Analysis with Boundary Elements, 2020, 120, pp. 13–27. DOI: 10.1016/j.enganabound.2020.07.024 [in English]
 13. Naumenko V.V., Strelnikova H.A. Singular integral accuracy of calculations in two-dimensional problems using boundary element methods. Engineering analysis with boundary elements. №26, pp. 95-98, 2002. DOI: 10.1016/S0955-7997(01)00041-8 [in English]
 14. Karaiev A., Strelnikova E. Singular integrals in axisymmetric problems of elastostatics. International Journal of Modeling, Simulation, and Scientific Computing., 2020, Vol. 11, № 1, 2050003. DOI: 10.1142/S1793962320500038 [in English]
 15. Gnitko V., Degtyariov K., Karaiev A. and Strelnikova E. Multi-domain boundary element method for axisymmetric problems in potential theory and linear isotropic elasticity, WIT Transactions on Engineering Sciences, Vol. 122, WIT Press, pp.13-25, 2019. DOI: 10.2495/BE410021 [in English]
 16. Raynovskyy, I.A., Timokha, A.N.: Sloshing in Upright Circular Containers: Theory, Analytical Solutions, and Applications. CRC Press/Taylor & Francis Group, (2020). 170 pp.. ISBN 9780367362898 [in English]
 17. Ravnik, J., Strelnikova, E., Gnitko V., Degtyarev, K., Ogorodnyk, U., BEM and FEM analysis of fluid-structure interaction in a double tank, Engineering Analysis with Boundary Elements, 67, pp. 13-25, 2016. <https://doi.org/10.1016/j.enganabound.2016.02.006> [in English]
 18. Ibrahim, R. A., Pilipchuk, V. N., and Ikeda, T. (March 1, 2001). "Recent Advances in Liquid Sloshing Dynamics." ASME. Appl. Mech. Rev. March 2001; 54(2): 133–199. <https://doi.org/10.1115/1.3097293> [in English]

UDC 656.11

Navigation methodology for vehicle city route optimal choice

P.K. Nikolyuk

*doctor of physical and mathematical sciences, professor of information technologies department
Vasil' Stus Donetsk National University, 600-richchia str., 21, Vinnytsia-21, Ukraine, 21021
e-mail: p.nikolyuk@donnu.edu.ua
<https://orcid.org/0000-0002-0286-297X>*

Relevance. The study is a fundamentally new approach to such an extremely important problem as the congestions in large cities. The solution of this global problem is a step in the realization of a smart city concept.

Goal. The aim of the study is to create basic elements of technology that can stabilize urban traffic and bring it to a qualitatively new state. To achieve this goal, the following tasks have been formulated:

- to create a model of a city transport network in the form of an oriented weighted non-planar multigraph with dynamically loaded arcs;
- to activate a city electronic map in the Traffic Management Centre (TMC) which allows tracking each vehicle;
- to navigate the time-optimal routes for all those vehicles that request the route;
- to implement the work of the software algorithm in real time with constant updating of the route of each tracked vehicle. That will allow monitoring changes in city traffic in real time and making adjustments to the route of each vehicle.

Research methods. The research is based on the use of mechanisms for modeling and working with various networks – the graph theory and the A-star algorithm. The latter traces the route on the graph (transport network) between two selected positions of the vehicle graph theory – origin and destination. The heuristic A-star algorithm – a powerful computational method of graph theory has been used in the study. This makes it possible to synchronize vehicles flows and therefore provides a qualitatively new level to the control of urban traffic.

The results. The problem of traffic load registration for the city transport network essential for navigating a vehicle route in metropolis has been solved. Traffic data of the real transport network have been reproduced on the city electronic map. Each vehicle received a unique marker consisting of an origin-destination pair and can be tracked on the map. Since each vehicle is under control of the Traffic Management Center (TMC), it is possible to track it along the optimal route, taking an urban traffic dynamic into account. Support is provided via the General Packet Radio Service (GPRS) channel, which allows each driver to receive instructions as to an optimal travel path.

Conclusions. The study has proposed a working software module that navigates a time-optimized route on the graph that represents the model of the real transport city network.

Key words: *weighted oriented graph, A-star algorithm, city traffic, traffic jams, congestion, Traffic Management Centre, graph model of city traffic network.*

Методологія вибору оптимального міського автомобільного маршруту

**Ніколюк
Карпович**

Петро *доктор фізико-математичних наук, професор кафедри
інформаційних технологій Донецького національного університету
імені Василя Стуса, вул. 600-річчя, 21, Вінниця-21, Україна, 21021
e-mail: p.nikolyuk@donnu.edu.ua
<https://orcid.org/0000-0002-0286-297X>*

Актуальність. Дослідження є принципово новим підходом до вирішення такої надзвичайно актуальної проблеми сучасності, як проблеми заторів у великих містах. Рішення зазначеної глобальної проблеми є прямим шляхом до реалізації основного завдання розумного міста.

Метою дослідження є створення базових елементів технології, які зможуть стабілізувати міський рух і привести його в якісно новий стан. Для досягнення поставленої мети були поставлені наступні завдання:

- створити модель транспортної мережі міста у вигляді орієнтованого зваженого неплоского мультиграфа з динамічно навантаженими дугами;
- активувати електронну карту міста в Центрі Керування Трафіком (ЦКТ) і за допомогою цієї карти супроводжувати кожен транспортний засіб;

– здійснювати оптимальну по часу навігацію за маршрутами для всіх транспортних засобів, які замовили такі маршрути;

– реалізувати роботу програмного алгоритму в режимі реального часу з постійним оновленням маршруту кожного супроводжуваного транспортного засобу.

Це дозволить своєчасно відслідковувати зміни у міському трафіку та вносити корективи в маршрут кожного супроводжуваного автомобіля.

Методи дослідження. Дослідження базується на використанні механізмів моделювання та роботи з різними мережами – теорії графів та алгоритму A-star. Останній простежує маршрут на графіку (транспортній мережі) між двома обраними позиціями транспортного засобу – початковою та кінцевою. Дослідження використовує евристичний алгоритм A-star – потужний обчислювальний метод теорії графів. Це дає змогу синхронізувати транспортні потоки, а тому міський рух виходить на якісно новий рівень.

Результати. Для навігації по маршруту транспортного засобу в мегаполісі вирішено проблему реєстрації транспортного навантаження транспортної мережі міста. Дані про рух реальної транспортної мережі відтворюються на електронній карті міста. Кожен транспортний засіб отримує унікальний маркер, що складається з пари «пункт відправлення - пункт призначення». Кожна така пара відстежується на електронній карті міста. Оскільки кожен транспортний засіб знаходиться під контролем ЦКТ, є можливість супроводжувати його по оптимальному маршруту з урахуванням міської динаміки руху. Підтримка надається через канал General Packet Radio Service (GPRS), який дозволяє кожному водієві отримувати інструкції щодо оптимального маршруту руху.

Висновки. У дослідженні запропоновано робочий програмний модуль, який забезпечує навігацію за оптимальними по часу маршрутами на графі, який є моделлю реальної транспортної мережі міста.

Ключові слова: зважений орієнтований граф, A-star алгоритм, міський трафік, затори, Центр Керування Трафіком, графова модель міської транспортної мережі.

1. Introduction

The research is based on a use of mechanisms for modeling and working with various networks – the graph theory and the A-star algorithm. The latter navigates the optimal route on the graph (or in city transport network) between two selected positions of the vehicle – origin and destination. To navigate the optimal route, it is necessary to know the traffic load of a city transport network. The load of traffic lanes corresponds to the arcs weights of the graph that simulates the city transport network. This kind of comparison allows using an oriented weighted multigraph to model the city transport network. In turn, the graph theory has a number of algorithms that allow, in particular, calculating the optimal routes. In this study we are discussing the optimal routes, namely those that take a minimal time.

In order to track each car along the optimal route, each vehicle assigned a unique marker consisting of an origin-destination pair. Since each vehicle is under the control of the TMC, its tracking on the selected route is equivalent to the tracking of the marker on the city electronic map. In turn, a real support of the vehicle involves online interaction with a driver of the vehicle, which is carried out via a GPRS channel. This allows each driver to receive dynamic real-time instructions for the optimal travel. It is important for such instructions to be correlated with the dynamics of urban traffic at each moment of time.

The end result of this research is: i) support of an optimal route of each guided vehicle, that minimizes a travel time between origin and destination (ordered); ii) synchronization of traffic flows; iii) optimal use of transport arteries throughout the city; (iv) significant reduction in the duration and frequency of congestions.

Therefore, the main goal of the work is creating the algorithm for optimal routes navigating for each vehicle involved in traffic. Research methods: i) the methods of modeling complex systems theory; ii) the methods of graph theory; iii) java programming technologies; iv) optimization methods; v) the A-star algorithm.

The aim of the study is to create basic elements of technology that can stabilize urban traffic and bring it to a qualitatively new state. To achieve the stated goal, the following tasks have been set:

– to create a model of a city transport network in the form of an oriented weighted non-planar multigraph with dynamically loaded arcs;

– activate a city electronic map in the TMC which allows tracking each vehicle;

– to navigate the optimal routes for all those vehicles that request the route;

– to implement the work of the software algorithm in real time with constant updating of the route of each tracked vehicle. That will allow monitoring changes in city traffic in real time and making adjustments to the route of each vehicle.

2. Analytical review of the literature

A fundamental issue of the urban traffic regulation is the representation (bijective reflection) of real urban traffic on the city electronic map [1-3]. The example of real picture of visualization of traffic flows in a big city is shown in Fig. 1 as a rose diagrams applied to the spatiotemporal analysis of traffic congestion in Helsinki. The transparency of each circular segment is used to depict the number of occurring traffic jams. The size of each circular segment represents the duration of traffic jams. Rose diagrams applied to stops and congestion analysis [4].



Fig.1 Visualization of traffic flows in Helsinki.

Understanding the city congestion level is a starting point in solving the problem of traffic regulation. If vehicles get redirected from the central congested streets (in Fig. 1 the corresponding roads are highlighted) to unused ones, the traffic state will approach equilibrium.

The question of equilibrium for any system is a fundamental point in game theory [5,6]. The state of equilibrium is especially relevant for dynamic systems, where deviations from this state are quite common. One of the highly dynamic systems is a transport network of a large modern city. Therefore, the main issue is to create an algorithm for bringing the mentioned system to a state of Nash equilibrium [5], which means an absence of congestion in the transport network.

The study of networks of different nature, including neural, allows making predictions as to urban transport networks [7-9].

Monitoring of the city transport arteries is relevant, in particular, due to the possibility of a certain periodicity (during the day, week, season) in city traffic. This circumstance allows forecasting traffic processes.

Particular attention is paid to a choice of optimal routes in the urban transport network, as this issue is key in terms of urban traffic strategy. In particular, the macroscopic mathematical model to control the dynamic factors of traffic – speed, intensity and congestion of vehicles is used for solving this problem in [10].

It is expedient to study the transport networks principles functioning in different modes. Particular attention in this regard should be paid to the overloaded mode, which causes congestion. The research presented in [11] is devoted to the dynamics of the transition between different phases of traffic.

The authors of [12] propose a real algorithm that allows navigating time-optimized route for travel. It is important that there is a constant correction of the route taking into account the dynamics of changes in the city transport traffic. Each intersection is equipped with special sensors that allow monitoring the load of lanes at any given time and respond to such changes in a timely manner. The transport network is modeled as a weighted planar oriented multigraph. The weight of each arc of the graph corresponds to the load of the real city lane. Drivers who interact with the TMC via GPRS channel have an opportunity to receive constantly updated data as to the optimal route. Plotting a route in the graph that simulates the city transport network is performed by a computer program that implements the A-star algorithm.

The main target of an optimization of urban traffic processes is an urban intersection. It is one of the main causes of congestion on urban transport networks. Therefore, the main task is to maximize the throughput at each intersection. A lot of research has been devoted to this problem. Let's focus on the results of [13] which analyzes a Connected Vehicles Technology (CVT). This work proposes an algorithm to optimize travel through a controlled intersection. The algorithm allows recording a location and speed of connected and identified unconnected vehicles within an area of interest around the intersection. As a result of the algorithm, the optimal sequence of switching phases of the intersection traffic lights and their length is obtained: the algorithm provides for the minimum delay time of vehicles at the intersection. [14-19] are also focused on a similar problem.

The research in [20] is devoted to the prospects of using self-driving cars. The advantages of such innovative approach and real prospects of their use have been described. This problem has recently received special attention from researchers around the world and number of publications in recent years has increased dramatically.

3. Results of the work

Let us formulate the stages for solving a problem of optimal mode of urban traffic management. The first step is a modeling of a city transport network by using a weighted oriented non-planar multigraph (Fig. 2). The fundamental point is that each arc of the graph is compared in accordance with the weight – the dynamic value which matches the actual load in each urban lane. In graph theory this graph is a bijective (maximum objective) reflection of a distribution of vehicles on a city transport artery. The second important point is the A-star algorithm, which is used to plot optimal routes between two pointed vertices.

The next stage is a software algorithm for navigating specific routes, taking into account all features of a transport network dynamic. Such features include high mobility and different levels of congestion at separate lanes on the same section of the road and separate streets – the central highways are usually congested, while the secondary ones – underloaded (Fig. 1). Therefore, one of the tasks of the proposed algorithm is to organize a more or less uniform distribution of vehicles over the city arteries. It could be said, that analyzing the data on the distribution of cars presented in Fig. 1, the task of the study is to achieve uniform color over the entire area, which will mean a uniform distribution of vehicles over urban transport arteries. And, finally, the main – the fourth – point is the registration of the vehicles on the city electronic map through the use of special sensors located near the intersection. The data from these sensors used as input values for the software module that implements the A-star algorithm. As a result of the implementation of above steps, there will be a complete synchronization of traffic flows, which leads to a fundamentally new quality and, consequently, to the disappearance of congestion (or a sharp reduction of it) in the transport network. This will allow each driver to arrive at their destination in the shortest time possible. Thus, urban traffic is transformed into a fundamentally new state – the state of Nash equilibrium [5,6].

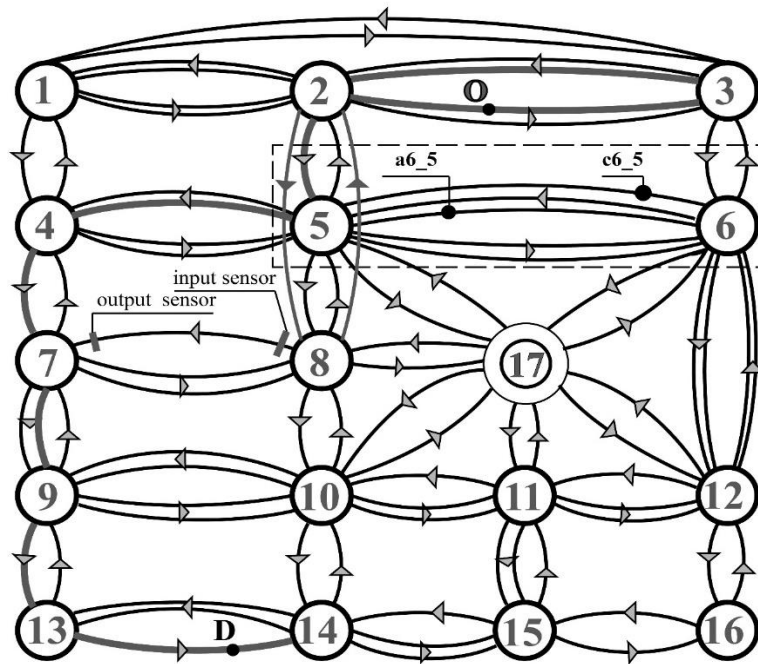


Fig.2. Weighted oriented non-planar multigraph which models a fragment of the city transport network. The vertex 17 is a special case of the intersection, where a circular motion is organized and therefore intersections 5, 6, 8, 10, 11 and 12 are connected by means of 17. The symbol "O" symbolizes the starting position of vehicle, and the symbol "D" – its destination. Between the vertices 5 and 6 there are lane designations (for example, the designation c6_5 means the third lane from the axial dividing line, directed from the intersection 6 to the intersection 5). The intersections 2 and 8 form a transport interchange (the road connecting these intersections, runs over the intersection 5).

The graph (Fig. 2) simulates not just a road between intersections, but reproduces individual lanes, that is a fundamentally important point, because the load of different lanes, as a rule, differs significantly. Therefore, it is necessary to record the load of each lane separately. Full program code for our research is available on a GitHub service [21]. The code navigating a route of vehicle from the starting position O to the destination D (Fig. 2) is presented. The fundamental question is to navigate a time-optimized route for each vehicle. To do this, we need to introduce the basic concepts, which are formulated as follows:

Definition 1. Traffic resistance $R_h^j(t)$ of any intersection j as time function of traffic observation t in the direction of intersection j for the vehicles arranged along line h is computed as time interval for which the cars were delayed at the intersection. Such resistance of the intersection is defined as

$$R_h^j(t) = S_h^j(t) \cdot T_h^j(t), \quad (1)$$

where $T_h^j = t_j^{red}(h) + t_j^{green}(h)$ is duration of traffic light switching cycle at the intersection j (duration of yellow signal is included in $t_j^{red}(h)$); $S_h^j(t)$ specified as follows:

$$S_h^j(t) = \begin{cases} \delta \\ \tau_h^j(t) / T_h^j(t) + \delta \\ \tau_h^{j,red}(t) / T_h^j(t) + C_h^j(t) + \delta \\ \tau_h^{j,green}(t) / T_h^j(t) + C_h^j(t) + \delta \end{cases}, \quad (2)$$

where value $\delta \ll 1$ corresponds to the ratio *time of travel by vehicle at intersection/duration of a switching cycle of traffic lights* $T_h^j(t)$; that is, in the case of the first term, vehicle crosses the intersection without delay; the value in the second term is obtained when the vehicle approaches the intersection at a time of red light, delayed for a while $\tau_h^j(t)/T_h^j(t) + \delta$ and passes this intersection immediately after the inclusion of the green phase; the value $\tau_h^j(t)/T_h^j(t) + C_h^j(t)$ in the third term consists of a fraction $\tau_h^{j,red}(t)/T_h^j(t)$ plus the number of traffic light switching cycles ($C_h^j(t) \geq 1$), during which the car is delayed at an intersection; $\tau_h^{j,green}(t)/T_h^j(t) + C_h^j(t)$ – the part of the fourth term value consisting of a fraction $T_h^j(t)$ in sum with an integer number of traffic light switching cycles ($C_h^j(t)$).

Delay time of car at an intersection

$$R_h^j(t) = \begin{cases} \delta \cdot T_h^j(t) \\ \tau_h^j(t) + \delta \cdot T_h^j(t) \\ \tau_h^{j,red}(t) + C_h^j(t) \cdot T_h^j(t) + \delta \cdot T_h^j(t) \\ \tau_h^{j,green}(t) + C_h^j(t) \cdot T_h^j(t) + \delta \cdot T_h^j(t) \end{cases} . \quad (3)$$

The weight $W_h^j(t)$ of each arc of the graph (Fig. 2) is a sum of the resistance $R_h^j(t)$ of an intersection j on the side of the lane h plus the length L_h^j of the lane, i.e.

$$W_h^j(t) = R_h^j(t) \cdot V_h^j + L_h^j . \quad (4)$$

Here V_h^j means the vehicle average velocity along the lane h to the intersection j .

The strategic goal set in this paper is as follows:

$$\sum_{h=1}^I (R_h^j(t) \cdot V_h^j + L_h^j) \rightarrow \min . \quad (5)$$

Here I is the set of all city lanes along vehicle trip. The expression (5) consists of variable (the first term) and constant components (the second term). The variable component is a function of time because a traffic in city is highly dynamic. In order to synchronize an actual traffic in a city and the program data, a constant recalculation of a route for each vehicle is performed. Thus, the problem of navigating an optimal route in the graph/(city network), synchronized in time with the real situation with the traffic on the city transport network, solves the A-star algorithm by constantly correcting a calculation of an optimal route between selected vertices of the graph. The route adjustment process is shown in Listing 1, and the basic algorithm code is shown in [19].

The value $S_h^j(t)$ represented by (2) can also be specified as follows

$$S_h^j(t) = \sum_{n=1}^l N_{n,h}^j(t) / \left(\sum_{n=1}^l m_{n,h}^j(t) \right) . \quad (6)$$

Here $N_{n,h}^j(t)$ is the number of cars entering the lane n (numeration of this lanes goes in perpendicular direction of median strip of the road) in the direction of the intersection j ; the value $m_{n,h}^j(t)$ is the number of cars passing through the intersection j from the lane n . The dynamics of traffic along the road $i \rightarrow j$ in the direction of an intersection j is due to a ratio between a number of vehicles on the road $i \rightarrow j$ to the number of vehicles leaving this section of the road for a time equal to a green

light $\tau_h^{j,green}$ at the intersection j . Technically the process is organized as follows: a sensor i_h (input sensor) located near the intersection i in the direction of the intersection j , registers the geolocation position of vehicles in accordance with their code marker. The range of such markers is determined by the ratio

$$R = \{(O_1, D_1), (O_2, D_2), \dots, (O_k, D_k), \dots, (O_N, D_N)\}. \quad (7)$$

Here N is a number of vehicles involved in a city traffic. Each car receives its own unique code, consisting of a pair of "origin-destination". Since a geolocation position of the vehicle are projected on an electronic map, it allows tracking a vehicle movement: as soon as the vehicle crosses the input sensor the next instruction from the TMC concerning the further route is transferred by using a GPRS channel.

A complex of output sensors (Fig. 2), located near the intersection j on the lanes directed in $i \rightarrow j$ direction, registers vehicles that have left the road $i \rightarrow j$. Input sensors detect vehicles entering the road section $i \rightarrow j$. The ratio $S_h^j(t)$ of these values indicates a dynamic of traffic on a particular section of road in a particular direction $i \rightarrow j$. It is clear that the closer $S_h^j(t)$ to the unit, the higher dynamics of traffic on the road in the direction $i \rightarrow j$. The task is to find in the web of the urban transport network the sequence of lanes with the minimal total weight (relative to the vehicle travel time). That sequence forms the time-optimized route. The value (6) in the software algorithm [21] is used as a key parameter that determines the values of the graph arcs weights.

The expression (5) can be represented as a record

$$\sum_h W_h^j(t) \rightarrow \min, \quad (7)$$

which is an objective function of the problem: the total weight of all lanes, which are located consecutively along the route, should be minimal. The minimization of the function (7) is performed by the A-star algorithm, the main code of which for the graph in Fig. 2 is shown in Listing 2 [21].

Definition 2. The weight of the route (7) is a minimum virtual path traveled by the vehicle from the starting position to the destination. The passage of such route takes a minimal time (taking into account the same average speed V_h^j of the car movement in the city).

The expression (7) is equivalent to the following

$$\tau^{trip} = \sum_{h=1}^I (\tau_h^j + L_h^j / V_h^j) \rightarrow \min, \quad (8)$$

where τ^{trip} is the travel time of a vehicle from its starting position to the destination.

Definition 3. Equivalence of the expressions (7) and (8) allows us to consider the optimization of vehicle travel time for the selected route. Accordingly, the total time spent by all vehicles involved in traffic will be optimal (minimal) as well.

Thus, the algorithm proposed in the study minimizes an actual travel time of the vehicle on the selected route.

As mentioned above, the calculation of an objective function of the forms (7) and (8) is performed by the programmed A-star algorithm, the code of which is presented in [21]. In order to simulate the weights w_h of the graph arcs a random variable generator `Random k = new Random()` has been used in the program

$$double w_h = (1 + (double)k.nextInt(10)) \cdot L_h. \quad (9)$$

Thus, the weight of the graph arc is represented as the sum (4), the value of which correlates with the congestion of urban transport arteries. Each arc of the graph (Fig. 2) receives a time-varying weight, a value of which corresponds to an actual load of urban transport arteries.

Fig. 3 shows a pair of neighboring intersections – i and j .

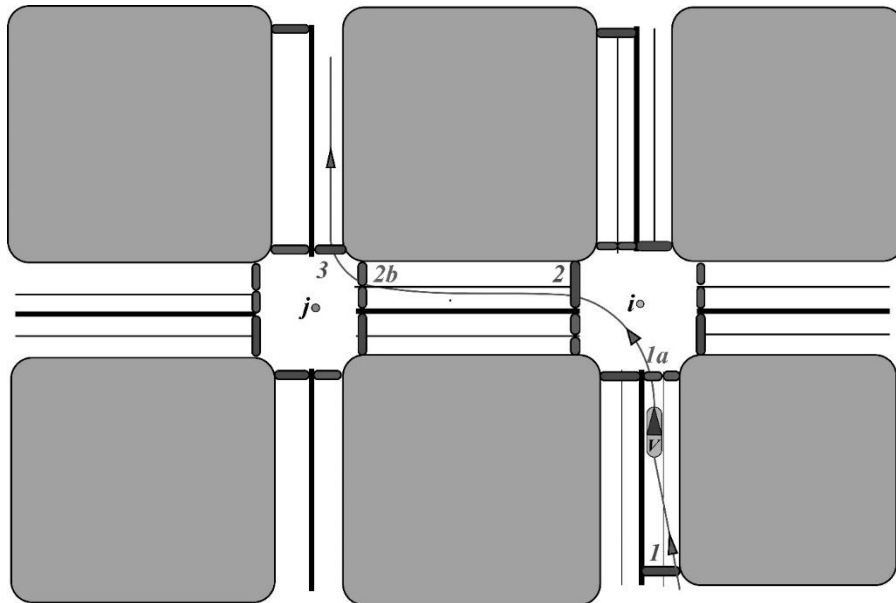


Fig.3. The pair of adjacent intersections on the city electronic map. The route of the vehicle V is shown.

At the moment of crossing the input and output sensors of an electronic map with geolocation marks the signal arrives to the ICC and initiates the start of a re-routing procedure [22] for cars with markers $(O_i, D_i, i \in N)$.

The procedure of constant recalculation of updated routes is necessary in order to respond to changes in urban traffic, in other words – for synchronization between the traffic dynamics and procedure of route navigation. Let us elaborate by presenting the example. In the illustrated case of the vehicle V movement (Fig. 3), the program at the ICC calculates a new route as soon as the vehicle crosses the intersection i . This process initiates a transmission of a control signal via the GPRS channel. A vehicle driver will receive the calculated software instructions for the further route selection at the beginning of *each* road lane (this is important!). Therefore, the driver has the time and ability to move to a recommended lane to perform the right maneuver at the next intersection, after which the driver will receive new instructions until the destination is reached.

The scheme of interaction of each vehicle with TMC is presented in Fig. 4.

CITY ROAD NETWORK TRAFFIC MANAGEMENT CENTRE

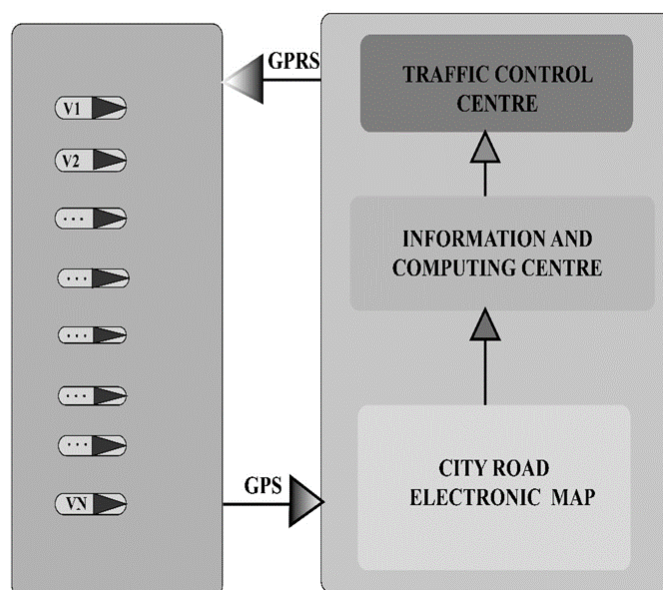


Fig. 4. Scheme, which shows the vehicles interaction with TMC [24].

In [23,24] the problem of map-matching of GPS markers (points) of vehicles with electronic maps is considered in detail. The essence of the map-matching algorithm is to correctly compare the GPS-trajectory of a vehicle with corresponding road segment. In particular, the comparison of GPS data with the city electronic map is carried out by using Automatic Vehicle Location (AVL) systems that are used to track the movement of the vehicle. In turn, an information collected by using AVL can be compared with electronic maps via the Internet or by using special software. For this purpose, the so-called device-based sensors are used, which constantly record and transmit information about the location and movement of the vehicle [25]. The latest applications of urban vehicle mobility visualization systems are based on the use of Geographic Information Systems (GIS). This system allows linking data to an electronic map, in particular to display a geolocation position of the vehicle on the map (Fig. 4).

The study [26] shows that finding the shortest route by using the A-star algorithm facilitates the procedure of comparing vehicle geolocation with road segments on electronic map. That means the Dijkstra or Floyd algorithms is less effective in this case.

A compact representation of congestion of urban transport networks can be done by using matrix analysis, as shown in [27]. The elements of the matrix are the values that represent a ratio of an average flow rate of vehicles in the road segment j to speed of free movement in this segment.

It is principal that an interaction between each vehicle and ICC takes place both in real time and in a mode of constant route correction due to the program, the code of which is presented in [21]. The program actually works and navigates specific routes. The program being run, one of possible options is presented as in Listing 1. The specific form of Listing 1 depends on an operation of random number generators $\text{Random } k = \text{new Random}()$.

Listing 1

Course-1: [b1_2-Liberty avenue, **a2_5-Heroes street**, a5_8-Victory street, a8_10-Lincoln street, a10_14-Candy street, a14_13-Cherry street, a13_14-Cherry street]

Course-2: [a2_5-Heroes street, **a5_8-Victory street**, a8_10-Lincoln street, a10_14-Candy street, a14_13-Cherry street, a13_14-Cherry street]

Course-3: [a5_8-Victory street, **a8_10-Lincoln street**, a10_14-Candy street, a14_13-Cherry street, a13_14-Cherry street]

Course-4: [a8_10-Lincoln street, **a10_14-Candy street**, a14_13-Cherry street, a13_14-Cherry street]

Course-5: [a10_14-Candy street, **a14_13-Cherry street**, a13_14-Cherry street]

Course-6: [a14_13-Cherry street, **a13_14-Cherry street**]

Course-7: [**a13_14-Cherry street**]

Between successive Courses the delay phase takes place, which corresponds to the time required for vehicle to travel one lane with subsequent passage of an intersection. In [21], the delay phase is implemented by using the `Thread.sleep(long (t2-t1))` method. That is, this phase is the time required for a vehicle to travel between two adjacent successive lanes h along the route. The visual interpretation is presented in Fig. 3: car V crosses two adjacent input sensors – on the road $i \rightarrow j$. It is the time between the moments of intersection of these sensors that is the delay phase. The program is "waiting" until a vehicle reaches a new lane. As soon as a vehicle crosses an input sensor of this new lane – the program immediately navigates an updated route, which, in turn, will be recalculated when a vehicle crosses an input sensor of a next lane along the route. And so on until a vehicle reaches a route destination.

The method `AstarSearch()` has been used to build a route between the selected graph vertices [21]. In fact, the driver is given only the directions to the next lane along the route (these lanes in Listing 1 are highlighted in bold). Therefore, the previous route is related to the next one (it is very important!) – the second lane of the previously calculated route is intercepted by the program using the software procedure `Node node = path.get(1)`. Thus, the new calculated route correlates with changes in traffic. That allows the program to follow an urban traffic dynamic. This means that the optimal time routes will be laid for all vehicles in metropolis. Therefore, if the traffic jams do not disappear, the probability of their appearance will be significantly reduced.

We will now describe some features of the program presented in [21], which consists of three classes – `Astar`, `Node` and `Edge`. The first class is very voluminous because it contains the constructors of all nodes (e.g. `static Node ni_j = new Node("ni_j-Liberty avenue", 1250)`). Number 1250 represent a

heuristic distance measured in a straight line from the intermediate vertex of the graph to the lane where the route ends. Each node of the graph is extended, i.e., it covers the adjacent lane, for example, the node *a1_2* includes the node 1 itself and the lane 1_2 propagated from this node. That is done in order to register cars, which, of course, are located in lanes, not at intersections. And when a real route is laid, the starting position of the vehicle, as well as its destination are located on an arc of the graph (Fig. 2), and not in nodes.

The public static void *initGraph()* method of the *Astar* class specifies constructors *ni_j.adjacencies = new Edge[]*, which form an interaction between graph arcs. The traffic regulations must be taken into account when forming incidental arcs with help of such constructors. For example, from lane *a6_5* (Fig. 2) it is impossible to return to lane *a5_2* – this is prohibited by traffic regulations. Therefore, the program will not include such route element. The interaction between all incidental arcs of the graph is similarly presented.

The public static void *AstarSearch (Node source, Node goal)* method of the *Astar* class implements the A-star algorithm [16]. This algorithm has a slight algorithmic complexity $O(D)$, where D is an amount of input data. The peculiarity of this algorithm is that it cuts off obviously unpromising routes by introducing so-called heuristic distance.

Class variables are declared in the *Node* class field, and the public *Node(String val, double hVal)* method returns a variable value.

The *Edge* class returns the values of target and cost variables used for routing in method public static void *AstarSearch (Node source, Node goal)* of the *A-star* class.

It should be noted that the study [28] considers a similar problem but solves it by using the Dijkstra algorithm.

4. Conclusions

1. A multigraph model that reproduces the transport network of a city district, simulates all actually existing lanes. Each arc of the graph receives a weight that is changed synchronously according to changes in traffic.

2. The use of sensors, which are located on an electronic map in an area of each intersection, allows for registration of a traffic flow. That means the entire city transport network is under control of TMC making it possible to track all changes in traffic and calculate the optimal routes for each vehicle dynamically in real-time in order to avoid congestions.

3. The working software module [21] that navigates an optimal time route in graph, and hence in the real transport network has been developed. The heuristic A-star algorithm – a powerful computational method of graph theory – is used. This makes it possible to synchronize vehicle flows and therefore urban traffic takes a qualitatively new level.

4. The implementation of the proposed technology is possible without significant additional costs.

ЛІТЕРАТУРА

1. Sewall J., Van den Berg J., Lin M. C., D. Manocha. Virtualized Traffic: Reconstructing Traffic Flows from Discrete Spatiotemporal Data. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, Issue 1, P. 26-37, 2011. <https://doi.org/10.1109/tvcg.2010.27>
2. Jagannathan P., Gurumoorthy S., Stateczny A., Parameshachari B. D., Sengupta J. Collision-Aware Routing Using Multi-Objective Seagull Optimization Algorithm for WSN-Based IoT. *Sensors*, Vol. 21, Issue 4, P. 8496- 8520, 2021. <https://doi.org/10.3390/s21248496>
3. Tostes A. I. J., Duarte-Figueiredo F. de LP, Assunção R., Salles J., Loureiro A. A. F. From data to knowledge: City-wide Traffic Flows Analysis and Prediction Using Bing Maps. “*UrbComp '13: Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing*”, Vol. 12, P. 1-8, 2013. <https://doi.org/10.1145/2505821.2505831>
4. Sobral T., Galvao T., Bornes J. Visualization of Urban Mobility Data from Intelligent Transportation Systems. *Sensors*, Vol. 19, Issue 2, P. 332-360, 2019. <https://doi.org/10.3390/s19020332>
5. Ganzfried S., Laughlin C., Morefield C. Parallel Algorithm for Nash Equilibrium in Multiplayer Stochastic Games with Application to Naval Strategic Planning. *International Conference on Distributed Artificial Intelligence*, P. 1-13, 2020. https://doi.org/10.1007/978-3-030-64096-5_1
6. Min H.-G., Fang Y.-K., Wu X., Wu G., Zhao X. On-ramp merging strategy for connected and automated vehicles based on complete information static game. *Journal of Traffic and Transportation*

- Engineering (English Edition)*, Vol. 8, Issue 4, P. 582-595, 2021. <https://doi.org/10.1016/j.jtte.2021.07.003>
7. Rahimipour S., Moeinfar R., Hashemi S. M. Traffic prediction using a self-adjusted evolutionary neural network. *J. Mod. Transport*, Vol. 27, P. 306–316. 2019. <http://dx.doi.org/10.1016/j.trc.2015.02.2019>
 8. Emami A., Sarvi M., Bagloee S. A. Using Kalman filter algorithm for short-term traffic flow prediction in a connected vehicle environment. *J. Modern Transport*, Vol. 27, P. 222 – 232, 2019. <http://dx.doi.org/10.1155/2017/8241932>.
 9. Pompigna A., Rupi F. Comparing practice-ready forecast models for weekly and monthly fluctuations of average daily traffic and enhancing accuracy by weighting methods. *Journal of Traffic and Transportation Engineering (English Edition)*, Vol. 5, Issue 4, 239-253, 2018. <https://doi.org/10.1016/j.jtte.2018.01.002>
 10. Majid H., Lu C. Karim H. An integrated approach for dynamic traffic routing and ramp metering using sliding mode control. *Journal of Traffic and Transportation Engineering (English Edition)*, Vol. 5, Issue 2, P. 116-128, 2018. <https://doi.org/10.1016/j.jtte.2017.08.002>
 11. Kidando E., Moses R., Sando T., Ozguven E. E. An application of Bayesian multilevel model to evaluate variations in stochastic and dynamic transition of traffic conditions. *Journal of Modern Transportation*, Vol. 27 Issue 4, P. 235–249, 2019. <https://doi.org/10.1007/s40534-019-00199-2>
 12. Pidgurska A., Nikolyuk P. Intelligent urban traffic. *CERES*, Vol. 6, Issue 1, P. 33-61, 2020.
 13. Liang X., Guler Ilgin S., Gayan V. An equitable traffic signal control scheme at isolated intersections using Connected Vehicle technology. *Transportation Research Part C*, V. 110, P. 81-97. 2020. <https://doi.org/10.1016/j.trc.2019.11.005>
 14. Boguto D.G., Kadomskiy K.K., Nikolyuk P.K., Pidgurska A.I. Algorithm of intelligent urban traffic. *Bulletin of V. Karazin Kharkiv National University, Series "Mathematical Modeling. Information Technology. Automated Control System*, V. 42, P. 12 – 25. 2019.
 15. Xu C., Xu J. Intelligent terminal based intelligent traffic light system and method. Pat. CN104575066, China. 2015.
 16. C Xu, J. Xu. A kind of intelligent transportation road capacity note broadcasting system. Pat. CN104064049B, China. 2017.
 17. Shu A., Xu X., Xu L., Zhang B., Liu C., Cai Y. Urban artery traffic dynamic green wave signal control system and method based on real-time traffic flow data. Pat. CN110136454 (A), China. 2019.
 18. Zhang H., Li T., Hu H., Li L., Hao Y., Chen Q., Zhang W., Chu P., Jiang C., Zhou W. Dynamic prediction intelligent traffic management method for urban road. Pat. CN109920250(A), China. 2019.
 19. Wang T., Hussain A., Cao Y., Sangirov G. An improved channel estimation technique for IEEE 802.11p standard in vehicular communications. *Sensors*, Vol. 19, Issue 1, P. 98. 2019. <https://doi.org/10.3390/s19010098>
 20. Yu H., R. Jiang, Z. Zheng, L. Li, R. Liu, X. Chen, “Automated vehicle-involved traffic flow studies: A survey of assumptions, models, speculations, and perspectives”. *Transportation Research Part C*, Vol. 127, 2021. <https://doi.org/10.1016/j.trc.2021.103101>
 21. Nikolyuk P.K. A-Star algorithm. *GitHub*: 2022. [A-Star_algorithm/Astar.java at main · npk54/A-Star_algorithm \(github.com\)](https://github.com/npk54/A-Star_algorithm/Astar.java)
 22. Pan J., Popa I. S., Zeitouni K., Borcea C. Proactive vehicular traffic rerouting for lower travel time. *IEEE Transactions on Vehicular Technology*, Vol. 62, Issue 8, P. 3551-3568, 2013. <https://doi.org/10.1109/TVT.2013.2260422>
 23. Hashemi M., Karimi H. A. A weight-based map-matching algorithm for vehicle navigation in complex urban networks. *Journal Intelligent Transportation Systems: Technology, Planning, and Operations*, Vol. 20, Issue 6, P. 573-590, 2016. URL: <https://doi.org/10.1080/15472450.2016.1166058>
 24. Wenxin T., Yanhui W. Real-Time Map Matching: A New Algorithm Integrating Spatio-Temporal Proximity and Improved Weighted Circle. *Open Geosciences*, Vol. 11, Issue 1, P. 288-297, 2019. URL: <https://doi.org/10.1515/geo-2019-0023>
 25. Sobral T., Galvao T., Bordes J. Visualization of Urban Mobility Data from Intelligent Transportation Systems. *Sensors*, Vol. 19, Issue 2, P. 332-350, 2019. URL: <https://doi.org/10.3390/s19020332>

26. Quddus M., Washington S. Shortest path and vehicle trajectory aided map-matching for low frequency GPS data. *Transportation Research Part C: Emerging Technologies*, Vol. 55, P. 328-339, 2015. URL: <https://doi.org/10.1016/j.trc.2015.02.017>
27. Ma X., Li Y., Chen P. Identifying spatiotemporal traffic patterns in large-scale urban road networks using a modified nonnegative matrix factorization algorithm. *Journal of Traffic and Transportation Engineering (English Edition)*, Vol. 7, Issue 4, pp. 529-539, 2020. URL: <https://doi.org/10.1016/j.jtte.2018.12.002>
28. Dutta P., Khatua S., Choudhuri S. DB-corouting: Density Based Coordinated Vehicle Rerouting in Smart Environment. *International journal of intelligent transportation systems research*, Vol. 19, P. 539-556, 2021. URL: <https://link.springer.com/article/10.1007/s13177-021-00261-6>

REFERENCES

1. J. Sewall, J. van den Berg, M. C. Lin, D. Manocha, “Virtualized Traffic: Reconstructing Traffic Flows from Discrete Spatiotemporal Data”. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, Issue 1, P. 26-37, 2011. <https://doi.org/10.1109/tvcg.2010.27>
2. P. Jagannathan, S. Gurumoorthy, A. Stateczny, B. D. Parameshachari, J. Sengupta, “Collision-Aware Routing Using Multi-Objective Seagull Optimization Algorithm for WSN-Based IoT”. *Sensors*, Vol. 21, Issue 4, P. 8496- 8520, 2021. <https://doi.org/10.3390/s21248496>
3. A. I. J. Tostes, F. de LP Duarte-Figueiredo, R. Assunção, J. Salles, A. A. F. Loureiro, “From data to knowledge: City-wide Traffic Flows Analysis and Prediction Using Bing Maps”. *UrbComp '13: Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing*, Vol. 12, P. 1-8, 2013. <https://doi.org/10.1145/2505821.2505831>
4. T.Sobral, T. Galvao, J. Bornes, “Visualization of Urban Mobility Data from Intelligent Transportation Systems”. *Sensors*, Vol. 19, Issue 2, P. 332-360, 2019. <https://doi.org/10.3390/s19020332>
5. S.Ganzfried, C. Laughlin, C. Morefield, “Parallel Algorithm for Nash Equilibrium in Multiplayer Stochastic Games with Application to Naval Strategic Planning” *International Conference on Distributed Artificial Intelligence*, P. 1-13, 2020. https://doi.org/10.1007/978-3-030-64096-5_1
6. H.-G. Min, Y.-K. Fang, X. Wu, G. Wu, X. Zhao, “On-ramp merging strategy for connected and automated vehicles based on complete information static game”. *Journal of Traffic and Transportation Engineering (English Edition)*, Vol. 8, Issue 4, P. 582-595, 2021. <https://doi.org/10.1016/j.jtte.2021.07.003>
7. S. Rahimpour, R. Moeinfar, S. M. Hashemi, “Traffic prediction using a self-adjusted evolutionary neural network”, *J. Mod. Transport*, Vol. 27, P. 306–316. 2019. <http://dx.doi.org/10.1016/j.trc.2015.02.2019>
8. A. Emami, M. Sarvi, S. A. Bagloee, “Using Kalman filter algorithm for short-term traffic flow prediction in a connected vehicle environment”. *J. Modern Transport*, Vol. 27, P. 222 – 232, 2019. <http://dx.doi.org/10.1155/2017/8241932>.
9. A. Pompigna, F. Rupi, “Comparing practice-ready forecast models for weekly and monthly fluctuations of average daily traffic and enhancing accuracy by weighting methods”. *Journal of Traffic and Transportation Engineering (English Edition)*, Vol. 5, Issue 4, 239-253, 2018. <https://doi.org/10.1016/j.jtte.2018.01.002>
10. H. Majid, C. Lu, H. Karim, “An integrated approach for dynamic traffic routing and ramp metering using sliding mode control”. *Journal of Traffic and Transportation Engineering (English Edition)*, Vol. 5, Issue 2, P. 116-128, 2018. <https://doi.org/10.1016/j.jtte.2017.08.002>
11. E. Kidando, R. Moses, T. Sando, E. E. Ozguven, “An application of Bayesian multilevel model to evaluate variations in stochastic and dynamic transition of traffic conditions”. *Journal of Modern Transportation*, Vol. 27 Issue 4, P. 235–249, 2019. <https://doi.org/10.1007/s40534-019-00199-2>
12. A. Pidgurska, P. Nikolyuk, “Intelligent urban traffic”. *CERES*, Vol. 6, Issue 1, P. 33-61, 2020.
13. X. Liang, S. Guler Ilgin, V. Gayan, “An equitable traffic signal control scheme at isolated intersections using Connected Vehicle technology”. *Transportation Research Part C*, V. 110, P. 81-97. 2020. <https://doi.org/10.1016/j.trc.2019.11.005>
14. D.G. Boguto, K.K. Kadomskiy, P.K. Nikolyuk, A.I. Pidgurska, “Algorithm of intelligent urban traffic”. *Bulletin of V. Karazin Kharkiv National University, Series "Mathematical Modeling. Information Technology. Automated Control System*, V. 42, P. 12 – 25. 2019.

15. C. Xu, J. Xu, “Intelligent terminal based intelligent traffic light system and method”. Pat. CN104575066, China. 2015.
16. C. Xu, J. Xu, “A kind of intelligent transportation road capacity note broadcasting system”. Pat. CN104064049B, China. 2017.
17. A. Shu, X. Xu, L. Xu, B. Zhang, C. Liu, Y. Cai, “Urban artery traffic dynamic green wave signal control system and method based on real-time traffic flow data”. Pat. CN110136454 (A), China. 2019.
18. H. Zhang, T. Li, H. Hu, L. Li, Y. Hao, Q. Chen, W. Zhang, P. Chu, C. Jiang, W. Zhou, “Dynamic prediction intelligent traffic management method for urban road”. Pat. CN109920250(A), China. 2019.
19. T. Wang, A. Hussain, Y. Cao, G. Sangirov, “An improved channel estimation technique for IEEE 802.11p standard in vehicular communications”. *Sensors*, Vol. 19, Issue 1, P. 98. 2019. <https://doi.org/10.3390/s19010098>
20. H. Yu, R. Jiang, Z. Zheng, L. Li, R. Liu, X. Chen, “Automated vehicle-involved traffic flow studies: A survey of assumptions, models, speculations, and perspectives”. *Transportation Research Part C*, Vol. 127, 2021. <https://doi.org/10.1016/j.trc.2021.103101>
21. P.K.Nikolyuk, “A-Star algorithm”. *GitHub*: 2022. [A-Star_algorithm/Astar.java at main · npk54/A-Star_algorithm \(github.com\)](https://github.com/npk54/A-Star_algorithm)
22. J. Pan, I. S. Popa, K. Zeitouni, C. Borcea, “Proactive vehicular traffic rerouting for lower travel time”. *IEEE Transactions on Vehicular Technology*, Vol. 62, Issue 8, P. 3551-3568, 2013. <https://doi.org/10.1109/TVT.2013.2260422>
23. M. Hashemi, H. A. Karimi, “A weight-based map-matching algorithm for vehicle navigation in complex urban networks”. *Journal Intelligent Transportation Systems: Technology, Planning, and Operations*, Vol. 20, Issue 6, P. 573-590, 2016. URL: <https://doi.org/10.1080/15472450.2016.1166058>
24. T. Wenxin, W. Yanhui, “Real-Time Map Matching: A New Algorithm Integrating Spatio-Temporal Proximity and Improved Weighted Circle”. *Open Geosciences*, Vol. 11, Issue 1, P. 288-297, 2019. URL: <https://doi.org/10.1515/geo-2019-0023>
25. T. Sobral, T. Galvao, J. Bordes, “Visualization of Urban Mobility Data from Intelligent Transportation Systems”. *Sensors*, Vol. 19, Issue 2, P. 332-350, 2019. URL: <https://doi.org/10.3390/s19020332>
26. M. Quddus, S. Washington, “Shortest path and vehicle trajectory aided map-matching for low frequency GPS data”. *Transportation Research Part C: Emerging Technologies*, Vol. 55, P. 328-339, 2015. URL: <https://doi.org/10.1016/j.trc.2015.02.017>
27. X. Ma, Y. Li, P. Chen, “Identifying spatiotemporal traffic patterns in large-scale urban road networks using a modified nonnegative matrix factorization algorithm”. *Journal of Traffic and Transportation Engineering (English Edition)*, Vol. 7, Issue 4, pp. 529-539, 2020. URL: <https://doi.org/10.1016/j.jtte.2018.12.002>
28. P. Dutta, S. Khatua, S. Choudhuri, “DB-corouting: Density Based Coordinated Vehicle Rerouting in Smart Environment”. *International journal of intelligent transportation systems research*, Vol. 19, P. 539-556, 2021. URL: <https://link.springer.com/article/10.1007/s13177-021-00261-6>

Наукове видання

**Вісник Харківського національного університету
імені В. Н. Каразіна**

Серія «Математичне моделювання. Інформаційні технології.
Автоматизовані системи управління»

Випуск 53

Збірник наукових праць

Українською та англійською мовами

Комп'ютерне верстання О.О. Афанасьєва

Підписано до друку 11.04.2022 р.
Формат 60x84/8. Папір офсетний. Друк цифровий.
Ум. друк. арк. – 9.
Обл.– вид. арк. – 11,3.
Наклад 50 пр. Зам. № 24/2022
Безкоштовно

Видавець і виготовлювач
Харківський національний університет імені В. Н. Каразіна
61022, м. Харків, майдан Свободи, 4
Свідоцтво суб'єкта видавничої справи ДК №3367 від 13.01.09

Видавництво Харківський національний університет імені В. Н. Каразіна
тел.: 705-24-32