

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ  
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

**АВТОМАТИЗОВАНІ СИСТЕМИ  
УПРАВЛІННЯ ТА ПРИЛАДИ  
АВТОМАТИКИ**

**Всеукраїнський міжвідомчий  
науково-технічний збірник**

**Заснований в 1965 р.**

**Випуск 177**

Харків  
2021

В сборнике представлены результаты исследований, касающихся компьютерной инженерии, управления, технической диагностики, автоматизации проектирования, оптимизированного использования компьютерных сетей и создания интеллектуальных экспертных систем. Предложены новые подходы, алгоритмы и их программная реализация в области автоматического управления сложными системами, оригинальные информационные технологии в науке, образовании, медицине.

Для преподавателей университетов, научных работников, специалистов, аспирантов.

У збірнику наведено результати досліджень, що стосуються комп'ютерної інженерії, управління, технічної діагностики, автоматизації проектування, оптимізованого використання комп'ютерних мереж і створення інтелектуальних експертних систем. Запропоновано нові підходи, алгоритми та їх програмна реалізація в області автоматичного управління складними системами, оригінальні інформаційні технології в науці, освіті, медицині.

Для викладачів університетів, науковців, фахівців, аспірантів.

**Редакционная коллегия:**

*В.В. Семенец*, д-р техн. наук, проф. (гл. ред.), *В.М. Левыкин*, д-р техн. наук, проф. (отв. ред.), *М.В. Евланов*, д-р техн. наук, доц. (отв. секр.), *Е.В. Бодянский*, д-р техн. наук, проф., *И.В. Гребенник*, д-р техн. наук, проф., *А.Л. Ерохин*, д-р техн. наук, проф., *А.А. Каргин*, д-р техн. наук, проф., *А.И. Михалёв*, д-р техн. наук, проф., *И.Ш. Невлюдов*, д-р техн. наук, проф., *К.Э. Петров*, д-р техн. наук, проф., *Е.П. Пуятин*, д-р техн. наук, проф., *И.В. Рубан*, д-р техн. наук, проф., *В.А. Филатов*, д-р техн. наук, проф., *Г.З. Халимов*, д-р техн. наук, проф.

Свидетельство о государственной регистрации  
печатного средства массовой информации

КВ № 4619 от 18.10.2000г.

*Адрес редакционной коллегии:* Украина, 61166, Харьков, просп. Науки, 14, Харьковский национальный университет радиоэлектроники, комн. 254, тел. 70-21-451

© Харківський національний університет  
радіоелектроніки, 2021

## ЗМІСТ

<b>ПЕТРИЧЕНКО О.В.</b> ВДОСКОНАЛЕННЯ МЕТОДОЛОГІЇ УПРАВЛІННЯ ІТ-ПОСЛУГАМИ ПІДПРИЄМСТВА.....	4
<b>ПРОНЧАКОВ Ю.Л.</b> ДОСЛІДЖЕННЯ СТРУКТУРНИХ РІШЕНЬ ПРИ СТВОРЕННІ РОЗПОДІЛЕНОГО ВІРТУАЛЬНОГО ВИРОБНИЦТВА.....	13
<b>ГУБАРЕНКО Є.В., ГУБАРЕНКО М.С., АНТОНЮК М.В.</b> РОЗПІЗНАВАННЯ ОБ'ЄКТІВ У ВІДЕОПОТОЦІ.....	18
<b>СИТНИКОВА П.Е., ГОВДЕРЧАК О.П.</b> AWS STEP FUNCTIONS ЯК НЕВІД'ЄМНА СКЛАДОВА АРХІТЕКТУРИ СИСТЕМИ З ПОСЛІДОВНИМ ВИКОНАННЯМ БЕЗСЕРВЕРНИХ ФУНКЦІЙ.....	29
<b>САФОНИК А.П., ГРИЦЮК І.М., МІЩАНЧУК М.М., ІЛЬКІВ І.В.</b> ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ВИЗНАЧЕННЯ ЗАЛІЗА В КОАГУЛЯНТІ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ.....	35
<b>СИТНИКОВ Д.Е., АНДРУСЕНКО Ю.О.</b> ЗАСТОСУВАННЯ МОДЕЛІ НА ОСНОВІ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ ДЛЯ ЗАДАЧІ ПРОГНОЗУВАННЯ ПОШИРЕННЯ COVID-19 В УКРАЇНІ.....	43
<b>КРАВЕЦЬ Н.С.</b> ВИЗНАЧЕННЯ ОБМЕЖЕНЬ РЕАЛІЗАЦІЇ ОБРОБКИ ПОДІЙ В ХМАРНОМУ ДОДАТКУ ЗА ДОПОМОГОЮ БЕЗСЕРВЕРНИХ ОБЧИСЛЕНЬ.....	47
<b>ТІТОВСЬКОЮ М.С., ХРЯПКІН О.В.</b> ДОСЛІДЖЕННЯ МЕТОДІВ МІГРАЦІЇ ДАНИХ В СИСТЕМАХ ЕЛЕКТРОННОЇ КОМЕРЦІЇ НА ПРИКЛАДІ SMS MAGENTO.....	51
<b>ЮР'ЄВ І.О.</b> РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ УПРАВЛІННЯ СИСТЕМОЮ НАДАННЯ ІТ-СЕРВІСІВ.....	57
<b>БЛОВА Т.Г., ДЬОМІНА В.М., ПОБІЖЕНКО І.О.</b> КОМПОЗИЦІЙНЕ ПРОЕКТУВАННЯ ТА РЕПРОЕКТУВАННЯ ГЛОБАЛЬНОЇ СХЕМИ МУЛЬТИБАЗИ ДАНИХ.....	63
<b>ЧАЛИЙ С.Ф., ЛЕЩИНСЬКИЙ В.О., ЛЕЩИНСЬКА І.О.</b> ПРЕДСТАВЛЕННЯ УЗГОДЖЕНИХ ЗНАТЬ З УРАХУВАННЯМ ЇХ ЛОГІЧНОЇ НЕСУПЕРЕЧЛИВОСТІ ДЛЯ ЗАДАЧІ ПОБУДОВИ ПОЯСНЕНЬ В ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ ГЛОБАЛЬНОЇ СХЕМИ МУЛЬТИБАЗИ ДАНИХ.....	69
<b>ЧАЛА О.С.</b> ЗГОРТКОВА ПОДВІЙНА РАДІАЛЬНО-БАЗИСНА НЕЙРОННА МЕРЕЖА НА ОСНОВІ АКТИВАЦІЙНИХ ЯДЕРНИХ ФУНКЦІЙ СПЕЦІАЛЬНОГО ТИПУ.....	75
<b>ЧУМАК В.С., АВРУНІН О.Г., ЧУГУЙ Є.А., СВИД І.В.</b> АНАЛІЗ ПРИНЦІПІВ ПОБУДОВИ ТЕЛЕМЕДИЧНИХ КОМПЛЕКСІВ ШИРОКОГО ПРИЗНАЧЕННЯ.....	80
<b>РЕФЕРАТИ.....</b>	86

*О.В. ПЕТРИЧЕНКО*

## **ВДОСКОНАЛЕННЯ МЕТОДОЛОГІЇ УПРАВЛІННЯ ІТ-ПОСЛУГАМИ ПІДПРИЄМСТВА**

Розглянуто основні особливості існуючої методології управління послугами. Виділено основні недоліки, що утрудняють використання існуючої методології для управління інформаційними системами як сукупністю ІТ-послуг, які розробник системи надає замовнику у вигляді множини функцій системи. Запропоновано вдосконалити існуючу методологію управління послугами шляхом розширення набору основних концептів, що утворюють онтологію даної методології. Розроблено схему взаємозв'язку основних концептів вдосконаленої методології управління ІТ-послугами підприємства. Отримано категорні моделі, що встановлюють взаємодію основних описів вдосконаленої методології управління ІТ-послугами підприємства на різних стадіях циклу управління експлуатацією інформаційної системи.

### **1. Вступ**

В даний час переважну більшість інструментальних засобів, що автоматизують роботи з управління експлуатацією інформаційних систем (ІС), технологій (ІТ) і програмних продуктів різного призначення, створено на основі існуючої методології управління послугами. Основні положення даної методології викладені в [1]. Аналіз цих положень дозволяє виділити наступні основні особливості, що впливають на теоретичні та прикладні аспекти управління експлуатацією веб-базованих ІС [1]:

а) представлення управління експлуатованими ІС та ІТ як сукупністю окремих ІТ-послуг, реалізованих у вигляді ІТ-сервісів (при цьому будь-яка ІТ-послуга може бути реалізована як одним, так і множиною пов'язаних між собою ІТ-сервісів);

б) представлення управляючих впливів на множину ІТ-послуг як рішень керівництва, пов'язаних з постійним поліпшенням можливостей ІТ-послуг і будь-якою потребою у зміні як системи управління послугами, так і окремих послуг (сервісів);

в) використання в якості основного кількісного критерію управління послугами поняття «ефективність», визначеного як ступінь реалізації запланованих заходів та досягнення запланованих результатів в управлінні послугами;

г) представлення методологічних основ управління послугами як множини вимог, що висуваються до системи управління послугами та її окремих процесів і засновані на множині базових понять.

Необхідно уточнити, що подібні вимоги не відображають особливості можливих реалізацій системи управління послугами для конкретних замовників. Передбачається, що в ході створення та експлуатації будь-якої можливої системи управління послугами подібні вимоги повинні бути адаптовані до особливостей конкретних ІТ-проектів і ІТ-продуктів. Однак в [1] особливо наголошується, що будь-яка система управління послугами, розроблена для конкретної організації, не може залишити невиконаною жодну з подібних вимог.

Необхідно також відзначити, що множина базових понять і їх визначень, на якій базуються подібні вимоги, розділена на дві наступних підмножини [1]:

а) терміни, характерні для систем управління в цілому (21 термін);

б) терміни, характерні для управління послугами (29 термінів).

По суті, ці терміни утворюють концептно-онтологічну основу існуючої методології управління послугами та сприяють виділенню основних сутностей предметної області в ході будь-якої реалізації системи управління послугами.

Розглянуті особливості дозволяють стверджувати, що в даний час управління експлуатацією веб-базованих ІС являє собою процеси прийняття рішень, які слабо формалізуються і засновані на результатах збору, обробки та аналізу кількісних значень різних множин показників. Ці показники характеризують окремі аспекти експлуатованої ІС, процесів експлуатації і управління експлуатацією даної ІС. Внесення в експлуатовану ІС змін також носить слабо формалізований характер і не дозволяє в достатній мірі оцінити наслідки

реалізації окремих запитів на зміну (request for change, RFC) веб-базованої ІС. Тому проведення досліджень щодо вдосконалення методології управління послугами стосовно управління експлуатацією сучасних веб-базованих ІС є актуальним.

## **2. Мета і задача вдосконалення методології управління послугами**

Розглянуті вище особливості існуючої методології визначають головне протиріччя, що утруднює застосування даної методології для управління експлуатацією наборами ІТ-послуг, які представлені множиною функцій конкретних веб-базованих ІС управління підприємствами. Суть цієї суперечності полягає в наступному:

а) існуюча методологія управління послугами використовує для опису будь-якої ІС один рівень представлення - як множини окремих ІТ-послуг та їх різнорідних компонентів [1] без урахування їх належності Постачальнику або Споживачеві ІТ-послуг;

б) для Споживача та Постачальника ІТ-послуг опис будь-якої ІС передбачає одночасне існування кількох рівнів представлення - як єдиний продукт, як множина функцій, а також як множина ІТ-сервісів, що реалізують ці функції і знаходяться на різних стадіях свого життєвого циклу.

В результаті задача підвищення ефективності експлуатованої веб-базованої ІС в ході управління її експлуатацією представляється як задача оптимізації [2] з великою кількістю змінних. Цими змінними є показники, які описують плани заходів з управління кожною конкретною ІТ-послугою ІС і результати фактичного виконання цих заходів. При цьому темпи зростання кількості таких показників значно перевищують темпи зростання кількості окремих ІТ-послуг (функцій) веб-базованих ІС.

Тому метою даного дослідження є удосконалення існуючої методології управління послугами, що дозволяє спростити рішення задачі ефективного управління експлуатацією веб-базованих ІС з великою кількістю ІТ-послуг (функцій).

Для досягнення даної мети в дослідженні передбачається рішення таких задач:

- розробка основної концепції управління експлуатацією веб-базованої ІС управління підприємством на основі удосконалення існуючої методології управління послугами шляхом визначення додаткових концептів, які адаптують дану методологію до особливостей вирішення задачі управління експлуатацією веб-базованої ІС;

- розробка візуальної моделі онтології управління ІТ-послугами веб-базованої ІС шляхом включення запропонованих додаткових концептів з урахуванням зв'язків, що виникають між даними концептами і концептами існуючої методології управління послугами;

- розробка математичної моделі управління експлуатацією ІС, яка враховує особливості вдосконаленої візуальної моделі онтології управління ІТ-послугами веб-базованої ІС.

## **3. Розробка основної концепції управління експлуатацією веб-базованої інформаційної системи**

Для усунення протиріччя, виділеного в розділі 2 даного дослідження, перш за все, необхідне узгодження описів веб-базованої ІС управління підприємством на різних рівнях представлення цієї системи. Для цього необхідно виділити основні рівні представлення ІС управління підприємствами в ході їх експлуатації. Подібні рівні були виділені в [3] для процесів формування та аналізу вимог до системи. На підставі запропонованого в [3] рішення пропонується виділити такі рівні представлення експлуатованої ІС:

- бізнес-рівень (об'єднує в собі запропоновані в [3] рівень керованих об'єктів і/або процесів і загальносистемний рівень);

- рівень ІТ-послуг;

- рівень ІТ-сервісів;

- рівень ІТ-інфраструктури.

Особливості представлень поняття «експлуатована ІС» на кожному із запропонованих рівнів наведені в табл. 1.

Для узгодження описів ІС на розглянутих рівнях представлення пропонується ввести додаткове поняття, що дозволяє описувати поведінку ІС та її елементів в ході експлуатації. Як таке поняття пропонується використовувати термін «транзакція». Слід зазначити, що до недавнього часу в ІТ-сфері цей термін використовувався переважно для опису дій над даними і визначався як логічна одиниця роботи (точніше, логічна одиниця роботи бази

Таблиця 1

Визначення поняття «експлуатована інформаційна система» на різних рівнях представлення

Рівень представлення	Визначення поняття «експлуатована інформаційна система»
Бізнес-рівень	ІС – ІТ-продукт, який: а) має фінансову і юридичну цінність; б) є одним з механізмів керованого об'єкта та/або процесу; в) одночасно є системою, що складається з персоналу і комплексу засобів автоматизації; г) формує і відображає єдине цілісне інформаційне представлення цього об'єкта та/або процесу відповідно до поставлених перед даним продуктом цілей.
Рівень ІТ-послуг	ІС – система ІТ-послуг (функціональних завдань, функцій), яка: а) експлуатується на основі рішень Споживача про економічну або технічної доцільності даної дії; б) спрямована на формування і відображення єдиного цілісного інформаційного представлення керованого об'єкта або процесу відповідно до поставлених перед системою цілей.
Рівень ІТ-сервісів	ІС - система ІТ-сервісів, яка: а) реалізує сукупність операцій надання або обробки даних; б) спрямована на формування і відображення єдиного цілісного інформаційного представлення об'єкта або процесу відповідно до поставлених перед системою цілей.
Рівень ІТ-інфраструктури	ІС – система взаємопов'язаних елементів ІТ-інфраструктури, необхідних і достатніх для експлуатації всіх ІТ-сервісів ІС, яка: а) складається з комплексів засобів автоматизації; б) спрямована на формування і відображення єдиного цілісного інформаційного представлення об'єкта або процесу відповідно до поставлених перед системою цілей.

даних), яка зазвичай включає кілька операцій бази даних [4]. Однак за останнє десятиліття з'явилися пропозиції використовувати поняття «транзакція» для формального опису бізнес-процесів та їх окремих елементів, в тому числі - тих, що автоматизуються [5]. Відзначено також прикладне застосування поняття «бізнес-транзакція» як концептуальної основи даталогічного опису робіт з автоматизованого проектного управління послугами [6].

Кожна абстрактна транзакція повинна мати такі властивості [4]:

- нерозривність (atomicity);
- правильність (correctness);
- ізольованість (isolation);
- стійкість (durability).

Ці властивості дозволяють розглядати транзакцію як основу для формального опису будь-якої можливої дії, яка виконується ІС та її окремими елементами в ході експлуатації. Подібний опис найкраще використовувати для веб-базованих ІС, які являють собою сукупність взаємодіючих елементів- «чорних ящиків» (властивість нерозривності). При цьому кожен елемент веб-базованої ІС в максимально можливій мірі ізольований один від одного і в ході нормальної експлуатації може приймати значення, заздалегідь визначені множиною допустимих станів (властивість стійкості). Правильність елементів веб-базованої ІС в ході

експлуатації визначається шляхом тестування і приймально-здавальних випробувань цих елементів, в ході яких перевіряється їх відповідність висунутим вимогам або RFC цих вимог.

Для опису ІС та її елементів на розглянутих в табл. 1 рівнях представлення пропонується виділити наступні види транзакцій:

а) бізнес-транзакція - узгоджена зміна стану відносин двох і більше сторін, де кожна сторона готова до цієї зміни і знає, що його узгодять та ухвалить всі сторони [5];

б) функціональна транзакція - логічна одиниця сценарію виконання функції (послуги) ІС, зазвичай включає кілька сервісних транзакцій;

в) сервісна транзакція (транзакція) - логічна одиниця роботи, зазвичай включає кілька операцій бази даних [4];

г) інфраструктурна транзакція - логічна одиниця роботи на верхніх (прикладних) рівнях використовуваних мережевих моделей [7], зазвичай включає кілька операцій нижчих рівнів конкретної мережевої моделі.

Таким чином, будь-яку ІС (в тому числі веб-базовану ІС) можна описати на будь-якому рівні представлення за допомогою транзакцій, які переводять ІС та її елементи з початкового стану в кінцевий. При цьому для уніфікації описів різномірних елементів ІС в ході її експлуатації в [1] введено поняття «конфігураційний елемент» (КЕ) - елемент, який необхідно контролювати для надання однієї або декількох послуг.

Однак таке представлення експлуатованої ІС серйозно ускладнює її моделювання в ході управління експлуатацією. Застосування для управління експлуатацією ІС існуючої методології призводить до того, що швидкість росту кількості показників, що характеризують ІС та її КЕ, значно випереджає швидкість росту кількості КЕ в ІС. Наслідком цього є неможливість масштабування моделей і методів вирішення задач управління в ході управління експлуатацією дрібними, середніми і великими ІС через різке зростання розмірності таких задач. Слід зазначити, що існуючі методи вирішення задач великої розмірності ефективні тільки в окремих випадках і не вирішують цю проблему кардинально.

Як вихід з даної ситуації пропонується застосувати для управління експлуатацією веб-базованих ІС елементи теорії потенційної ефективності. Дана теорія дозволяє оцінити ефективність складної системи за рахунок застосування порівняно простих моделей, що описують окремі якості цієї системи. В цьому випадку ефективність системи представляється як ймовірність вигідного обміну між системою і навколишнім середовищем [8]. Таке визначення не суперечить прийнятому в існуючій методології визначенню поняття ефективності, наведеному вище, і дозволяє представити ступінь реалізації запланованих заходів та досягнення запланованих результатів в управлінні послугами як ймовірність вигідного обміну КЕ і ресурсами, необхідними для їх створення або модифікації, між ІС, її Постачальником і Споживачем.

Пропонований підхід до організації формального рішення задач управління експлуатацією ІС вимагає введення в існуючу методологію управління послугами поняття «властивість системи». При цьому для вирішення задач управління експлуатацією ІС інтерес викликають тільки ті властивості системи, які характеризують прояв окремих якостей цієї системи. Якості ІС, в свою чергу, визначаються множиною вимог (функціональних і нефункціональних), висунутих до ІС до початку її експлуатації, а також множиною RFC ІС та її окремих КЕ, що формуються в ході експлуатації ІС. В цьому випадку ефективність експлуатованої ІС буде визначатися як ступінь досягнення планових результатів по кожному окремому властивості ІС і її КЕ.

Надалі для позначення планових результатів експлуатації ІС введемо поняття «мета управління ІС». Тоді задача досягнення глобальної мети ефективної експлуатації ІС буде розглядатися як завдання досягнення множини окремих цілей управління ІС за кожною конкретною її властивістю.

Однак використання даного визначення ефективності як основи формального опису управління експлуатацією значною мірою залежить від точок зору Споживача і Постачальника на процеси експлуатації ІС. Різниця даних точок зору визначається, головним чином, відмінністю глобальних цілей Споживача і Постачальника, розглянутих в [3, 9]. Тому слід визнати одночасне існування двох наступних підходів:

а) підхід до ефективного управління експлуатацією ІС з точки зору Споживача (підхід до ефективного управління процесом функціонування ІС);

б) підхід до ефективного управління експлуатацією ІС з точки зору Постачальника (підхід до ефективного управління процесом супроводу ІС).

Саме необхідність управління процесами експлуатації ІС на основі одночасного застосування двох розглянутих вище підходів, цілі яких не збігаються один з одним, слід визнати основною проблемою формального опису ефективного управління експлуатацією ІС.

Результати запропонованого вдосконалення існуючої методології управління послугами дозволяють сформулювати основну концепцію управління експлуатацією веб-базованої ІС як набір наступних положень:

а) управління експлуатацією ІС та її КЕ з точки зору споживача розглядається як сукупність проектних та/або операційних заходів, результатом яких є досягнення глобальної ефективності ІС з точки зору всіх співробітників Споживача при підтримці витрат на ці заходи на бажаному для Споживача рівні;

б) управління експлуатацією ІС та її КЕ з точки зору Постачальника розглядається як сукупність проектних та/або операційних заходів, результатом яких є досягнення глобальної ефективності ІС з точки зору всіх співробітників Постачальника при підтримці витрат на ці заходи на бажаному для Постачальника рівні;

в) управління експлуатацією ІС в цілому при дотриманні інтересів Споживача і Постачальника даної ІС повинно розглядатися як окремий випадок задачі багатокритеріальної оптимізації, рішенням якої буде Парето-оптимальна ІС;

г) управління експлуатацією окремих КЕ ІС при дотриманні інтересів Споживача і Постачальника даної ІС повинно розглядатися як сукупність проектних та/або операційних заходів щодо перетворення множини поточних значень показників властивостей КЕ ІС (експлуатаційних характеристик окремих КЕ ІС) в множину запланованих значень цих КЕ, при яких ІС в цілому буде Парето-оптимальною.

#### **4. Розробка візуальної моделі онтології методології управління ІТ-послугами**

З урахуванням запропонованих додаткових концептів і результатів розробки концепції управління експлуатацією веб-базованої ІС візуальна модель онтології методології управління ІТ-послугами веб-базованої ІС набуде вигляду, показано на рис. 1.

Для поліпшення сприйняття на рис. 1 не показані основні слоти фреймів, що описують основні концепти. Підкресленням на рис. 1 виділені концепти, які введені в онтологію в результаті вдосконалення методології управління послугами. Курсивом на рис. 1 виділені концепти, які розширюють методологію управління послугами в ході вирішення прикладних задач управління послугами.

Наведена на рис. 1 схема дозволяє виділити групи основних сутностей предметної області, на основі яких можливо сформулювати такі описи:

а) опис експлуатованої ІС на різних рівнях представлення;

б) опис станів експлуатованої ІС;

в) опис показників, які характеризують ІС, що експлуатується, та її КЕ;

г) опис ефективності експлуатації ІС за її окремими властивостями і в цілому.

Опис експлуатованої ІС на різних рівнях представлення складатиметься з наступних концептів, показаних на рис. 1:

- концепт «Information System» («ІС»);

- концепт «Provider of IS» («Постачальник ІС»);

- концепт «Customer of IS» («Споживач ІС»);

- концепт «Requirement of IS» («Вимога до ІС»);

- концепт «RFC IS» («RFC ІС»);

- концепт «Configuration of IS» («Конфігурація ІС»);

- концепт «Presentation Layer» («Рівень представлення»);

- концепт «Configuration Item» («КЕ»);

- концепт «Transaction» («Транзакція»).

Зазначені концепти визначають основні точки зору, які слід враховувати під час детального формалізованого опису експлуатованої ІС на різних рівнях представлення як сукупності окремих КЕ та транзакцій, створених для виконання вимог Постачальника та Споживача ІС.

Опис станів експлуатованої ІС буде складатися з наступних концептів, показаних на рис. 1:



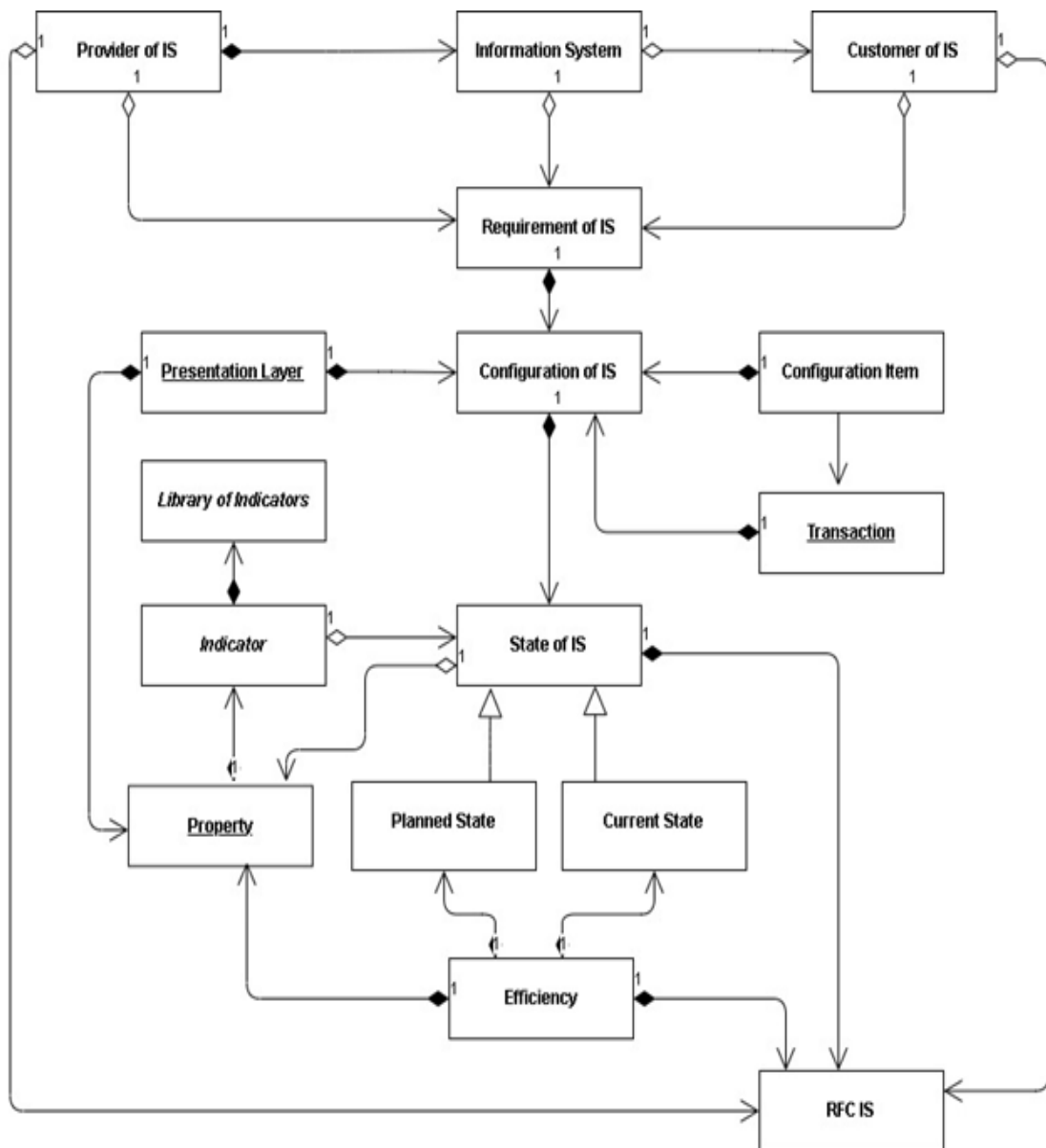


Рис. 1. Візуальна модель онтології методології управління ІТ-послугами веб-базованої інформаційної системи

- концепт «Configuration of IS» («Конфігурація ІС»);
- концепт «Presentation Layer» («Рівень представлення»);
- концепт «Configuration Item» («КЕ»);
- концепт «Transaction» («Транзакція»);
- концепт «State of IS» («Стан ІС»);
- концепт «Planned state» («Планований стан»);
- концепт «Current state» («Поточний стан»).

Зазначені концепти визначають основні точки зору, які слід враховувати під час детального формалізованого опису експлуатованої ІС на різних рівнях представлення як планованих та поточних станів окремих КЕ і транзакцій.

Опис показників, які характеризують експлуатовану ІС та її КЕ, буде складатися з наступних концептів, показаних на рис. 1:

- концепт «State of IS» («Стан ІС»);

- концепт «Indicator» («Показник»);
- концепт «Library of Indicators» («Бібліотека показників»).

Зазначені концепти визначають основні точки зору, які слід враховувати під час детального формалізованого опису експлуатованої ІС набором окремих показників, які базуються на конкретну бібліотеку показників.

Опис ефективності експлуатації ІС по її окремих властивостях і в цілому буде складатися з наступних концептів, показаних на рис. 1:

- концепт «State of IS» («Стан ІС»);
- концепт «Planned state» («Планований стан»);
- концепт «Current state» («Поточний стан»);
- концепт «Indicator» («Показник»);
- концепт «Presentation Layer» («Рівень представлення»);
- концепт «Property» («Властивість»);
- концепт «Efficiency» («Ефективність»);
- концепт «RFC IS» («RFC ІС»).

Зазначені концепти визначають основні точки зору, які слід враховувати під час детального формалізованого опису ефективності експлуатованої ІС на різних рівнях представлення.

Часткове дублювання концептів в різних описах необхідне для забезпечення зв'язків даних описів один з одним.

### **5. Розробка узагальненої математичної моделі управління експлуатацією інформаційної системи**

На основі виділених вище описів груп сутностей предметної області можливо розробити узагальнені математичні моделі управління експлуатацією ІС. Оскільки ці описи є наборами окремих концептів, пов'язаних певним чином між собою, формально їх можна представити у вигляді наступних категорій:

- а) опис експлуатованої ІС на різних рівнях представлення представимо категорією  $L_{IS}$  ;
- б) опис станів експлуатованої ІС представимо категорією  $L_{IS}$  ;
- в) опис показників, що характеризують експлуатовану ІС та її КЕ, представимо категорією  $L_{Ind}$  ;
- г) опис ефективності експлуатації ІС по її окремих властивостях і в цілому представимо категорією  $L_{Eff}$  .

Відповідно до положень [1], управління експлуатацією ІС засноване на циклі «Plan - Do - Check - Act» (PDCA). У цьому циклі управління:

- а) на стадії «Plan» здійснюється розробка планів експлуатації ІС;
- б) на стадії «Do» здійснюються роботи з виконання планів експлуатації ІС;
- в) на стадії «Check» здійснюються роботи з перевірки результатів виконання планів експлуатації ІС;
- г) на стадії «Check» здійснюються роботи з корекції планів експлуатації ІС.

Тоді узагальнена модель  $M_O$  управління експлуатацією ІС може бути представлена як надкатегорія вигляду:

$$M_O = \left[ L_P, L_D, L_{Ch}, L_A, F_{L_D}^{L_P}, F_{L_{Ch}}^{L_D}, F_{L_D}^{L_{Ch}}, F_{L_A}^{L_{Ch}}, F_{L_P}^{L_A} \right], \quad (1)$$

де  $L_P$  - надкатегорія, що формально описує стадію «Plan»;  $L_D$  - надкатегорія, що формально описує стадію «Do»;  $L_{Ch}$  - надкатегорія, що формально описує стадію «Check»;  $L_A$  - надкатегорія, що формально описує стадію «Act»;  $F_{L_D}^{L_P}$  - одномісний коваріантний функтор, що встановлює зв'язок між надкатегоріями  $L_P$  і  $L_D$ ;  $F_{L_{Ch}}^{L_D}$  - одномісний коваріантний функтор, що встановлює зв'язок між надкатегоріями  $L_D$  і  $L_{Ch}$ ;  $F_{L_D}^{L_{Ch}}$  - одномісний коваріан-

тний функтор, що встановлює зв'язок між надкатегоріями  $L_{Ch}$  і  $L_D$ ;  $F_{L_A}^{L_{Ch}}$  - одномісний коваріантний функтор, що встановлює зв'язок між надкатегоріями  $L_{Ch}$  і  $L_A$ ;  $F_{L_P}^{L_A}$  - одномісний коваріантний функтор, що встановлює зв'язок між надкатегоріями  $L_A$  і  $L_P$ .

Функтори  $F_{L_D}^{L_P}$ ,  $F_{L_{Ch}}^{L_D}$  і  $F_{L_A}^{L_{Ch}}$  встановлюють зв'язок між надкатегоріями  $L_P$ ,  $L_D$ ,  $L_{Ch}$  і  $L_A$  відповідно до життєвого циклу управління експлуатованої ІС. Функтор  $F_{L_D}^{L_{Ch}}$  встановлює зв'язок між надкатегоріями  $L_{Ch}$  і  $L_D$  для опису ситуацій, коли експлуатація ІС відбувається відповідно до розроблених планів і необхідність в корекції цих планів відсутня.

Для деталізації формального опису надкатегорій  $L_P$ ,  $L_D$ ,  $L_{Ch}$  і  $L_A$  розділимо введену категорію опису станів експлуатованої ІС  $L_{St}$  на дві підкатегорії: підкатегорію опису планового стану експлуатованої ІС  $L_{St}^P$  і підкатегорію опису поточного стану експлуатованої ІС  $L_{St}^C$ .

Тоді з урахуванням наведених раніше представлень описів груп сутностей предметної області у вигляді категорій надкатегорію  $L_P$  можна представити у вигляді:

$$L_P = \left[ L_{IS}, L_{St}^P, L_{Ind}, F_{L_{St}^P}^{L_{IS}}, F_{L_{Ind}}^{L_{St}^P} \right], \quad (2)$$

де  $F_{L_{St}^P}^{L_{IS}}$  - одномісний коваріантний функтор, що встановлює зв'язок між категоріями  $L_{IS}$  і  $L_{St}^P$ ;  $F_{L_{Ind}}^{L_{St}^P}$  - одномісний коваріантний функтор, що встановлює зв'язок між категоріями  $L_{St}^P$  і  $L_{Ind}$ .

Надкатегорію  $L_D$  можна представити таким чином:

$$L_D = \left[ L_{IS}, L_{St}^C, L_{Ind}, F_{L_{St}^C}^{L_{IS}}, F_{L_{Ind}}^{L_{St}^C} \right], \quad (3)$$

де  $F_{L_{St}^C}^{L_{IS}}$  - одномісний коваріантний функтор, що встановлює зв'язок між категоріями  $L_{IS}$  і  $L_{St}^C$ ;  $F_{L_{Ind}}^{L_{St}^C}$  - одномісний коваріантний функтор, що встановлює зв'язок між категоріями  $L_{St}^C$  і  $L_{Ind}$ .

Надкатегорію  $L_{Ch}$  можна представити у вигляді:

$$L_{Ch} = \left[ L_{IS}, L_{St}^P, L_{St}^C, L_{Ind}, F_{L_{St}^P}^{L_{IS}}, F_{L_{St}^C}^{L_{IS}}, F_{L_{Ind}}^{L_{St}^P}, F_{L_{Ind}}^{L_{St}^C} \right]. \quad (4)$$

Надкатегорію  $L_A$  можна представити у вигляді:

$$L_A = \left[ L_{IS}, L_{Eff}, F_{L_{Eff}}^{L_{IS}} \right], \quad (5)$$

де  $F_{L_{Eff}}^{L_{IS}}$  - одномісний коваріантний функтор, що встановлює зв'язок між категоріями  $L_{IS}$  і  $L_{Eff}$ .

## 6. Висновки і перспективи подальших досліджень

Розроблена концепція управління експлуатацією веб-базованої ІС управління підприємством дозволяє спростити рішення задачі ефективного управління експлуатацією ІТ-послуг за рахунок переходу від рішення задачі глобальної оптимізації з великою кількістю змінних до вирішення задачі Парето-оптимізації окремих властивостей веб-базованої ІС. При цьому кожна окрема задача оптимізації характеризується значно меншою кількістю змінних, що дозволяє в подальшому досліджувати можливості аналітичного рішення таких задач.

Розроблена візуальна модель онтології управління ІТ-послугами веб-базованої ІС дозволяє сформулювати основні елементи концептуальної моделі даних інформаційної технології автоматизованого управління експлуатацією веб-базованої ІС. Крім того, виділені в розробленій онтології основні концепти методології управління ІТ-послугами і зв'язки між ними є основою для виділення описів основних сутностей предметної області, які є основою для формального опису управління експлуатацією ІС.

На основі виділених описів були розроблені узагальнені теоретико-категорні моделі (1)-(5), що встановлюють формальні описи основних стадій циклу управління експлуатацією ІС. Отримані моделі дозволяють формально представити взаємодію виділених раніше описів основних сутностей предметної області в межах кожної конкретної стадії життєвого циклу, а також взаємодію сукупності таких описів при переході від однієї стадії циклу до іншої.

Отримані результати визначають перспективу подальших досліджень як розробку моделей і методів, що дозволяють деталізувати описи окремих елементів моделей (1)-(5). Подібні моделі і методи будуть визначати особливості вирішення конкретних задач управління експлуатацією ІС на різних стадіях життєвого циклу.

**Література:** 1. *ISO/IEC 20000-1. Information technology - Service management - Part 1: Service management system requirements. Third edition 2018-09.* Geneva: ISO Copyright Office, 2018. XX p. 2. *Levykin V., Evlanov M., Neumyvakina O., Petrichenko O.* Development of model of a web-based information system control problem // *Technology audit and production reserves.* 2021. № 1,2 (57). P. 25-31. DOI: 10/12287/2706-5448.2021.225067. 3. *Левыкин В.М., Евланов М.В., Керносов М.А.* Паттерны проектирования требований к информационной системе: моделирование и применение: монография. Харьков: ООО «Компанія СМІТ», 2014. 320 с. 4. *Дейт К. Дж.* Введение в базы данных, 8 издание. М.: Издательский дом «Вильямс», 2006. 1328 с. 5. *Артамонов И. В.* Бизнес-транзакции: характеристики и отличительные особенности // *Бизнес-информатика.* 2012. No 2 (20). С. 29-34. 6. *Ділові транзакції / Microsoft Build.* URL: <https://docs.microsoft.com/uk-ua/dynamics365/project-operations/psa/basic-business-transactions> (дата звернення 04.05.2021 р.). 7. *Танненбаум Э. С., Узеролд Д.* Компьютерные сети. СПб.: Питер Пресс, 2019. 960 с. 8. *Флейшман Б.С.* Элементы теории потенциальной эффективности сложных систем. М.: Советское радио, 1971. 224 с. 9. *Евланов М.В., Неумывакина О.Е., Карамышева А.Ю.* Глобальные цели поставщика и потребителя ИТ-услуг // *Восточно-европейский журнал передовых технологий.* - 2012. - № 5/2 (59). - С. 12-17.

*Надійшла до редколегії 29.04.2021*

**Петриченко Олександр Вячеславович**, кандидат технічних наук, докторант кафедри інформаційних управляючих систем ХНУРЕ. Наукові інтереси: веб-базовані інформаційні системи. Адреса: Україна, 61166, м. Харків, пр. Леніна, 14, тел. (057) 702 14 51.

*Ю.Л. ПРОНЧАКОВ*

## ДОСЛІДЖЕННЯ СТРУКТУРНИХ РІШЕНЬ ПРИ СТВОРЕННІ РОЗПОДІЛЕНОГО ВІРТУАЛЬНОГО ВИРОБНИЦТВА

Поставлено та вирішено науково-прикладну задачу синтезу архітектури розподіленої віртуальної виробничої системи. Актуальність теми дослідження пов'язана зі створенням віртуальних виробництв, що відповідає концепції індустріального розвитку Industry 4.0. Метою дослідження є обґрунтування раціонального складу та топології віртуальної виробничої системи, яка орієнтована на інноваційний портфель замовлень. Дослідження складається з декількох етапів: формування архітектури віртуального виробництва; моделювання виробничого циклу; дослідження вузьких місць у розподіленому віртуальному виробництві. Для оцінки розташування технологічного обладнання використовуються показники у вигляді витрат, часу та ризиків. Проведено моделювання виробничого циклу виконання замовлень з урахуванням часових затримок на виконання операцій у технологічному обладнанні. Для моделювання вузьких місць досліджуються черги, які виникають при виконанні замовлень. Використано математичні методи: методи цілочисельного (булевого) програмування; імітаційне подійне моделювання; агентне моделювання.

### 1. Вступ

Сучасна стратегія розвитку виробничих систем заснована на концепції індустріального розвитку виробництва Industry 4.0, в якій велику увагу приділено задачам віртуалізації, роботизації та інтелектуалізації управління сучасного виробництва [1]. Для підприємства, що розвивається, при формуванні нового портфелю замовлень інноваційного характеру необхідно проводити диверсифікацію своєї діяльності, впроваджувати нові технологічні об'єкти та процеси, сучасні форми організації у вигляді віртуального управління виробництвом [2]. Можлива розподіленість віртуальної виробничої системи (ВВС) призводить до необхідності пошуку раціональних архітектурних рішень при формуванні нових виробничих підсистем з урахування обмежених можливостей підприємства, що розвивається. Тому дослідження архітектурних рішень при створенні ВВС з урахуванням топології виробничої системи та складу нових технологічних об'єктів (ТО), які входять до ВВС, є актуальним.

### 2. Постановка задачі дослідження

В теперішній час у світовій практиці створення нових виробництв існує множина типових топологій для архітектурних рішень розподіленої виробничої системи. Ці структури ефективно використовуються при створенні машинобудівних підприємств, транспортних систем, нафтогазових комплексів, при створенні автострад тощо. До типових топологій в архітектурних рішеннях складних систем можна віднести такі:

- послідовна структура, в якій окремі ТО пов'язані за допомогою комунікацій у послідовний логістичний ланцюг;
- матрична (гратчаста) структура, в якій комунікаційні зв'язки між ТО утворюють решітку;
- радіальна структура з центром у вигляді, наприклад, складального виробництва;
- древовидна (ієрархічна) структура, яка забезпечує виконання виробничих процесів, що відповідають багаторівневій архітектурі складного виробу машинобудування;
- кільцева структура, в якій комунікаційні зв'язки між ТО утворюють кільце.

Окрім того, на практиці часто використовують змішані (комбіновані) структури, в яких переважає той або інший тип топології, наприклад, радіально-кільцева структура.

Вибір архітектури розподіленого віртуального виробництва залежить від специфіки нового портфелю замовлень, логістичних взаємодій між виробничими підсистемами та ТО, від вимог щодо логістики виробничої системи, а також обмежених можливостей підприємства, що розвивається [3]. Тому при аналізі можливих архітектурних рішень експерти у галузі створення нових виробництв повинні обґрунтувати та вибрати топологію архітектури ВВС, виходячи з перелічених вимог.

Метою дослідження, яке розглядається у даній публікації, є вирішення задачі раціонального розташування ТО у вузлах розподіленої архітектури віртуального виробництва.

Для досягнення поставленої мети пропонується вирішити такі задачі дослідження:

а) розробка моделі розташування ТО в вузлах розподіленої архітектури ВВС.

б) дослідження технології моделювання виробничого циклу та вузьких місць у сформованій розподіленій архітектурі ВВС.

### 3. Розробка моделі розташування технологічних об'єктів в вузлах розподіленої архітектури віртуальної виробничої системи

Для пошуку раціональних рішень щодо розташування ТО у вузлах розподіленої архітектури ВВС будемо використовувати наступні показники [4, 5]:

- витрати на придбання або створення нових ТО у складі ВВС (технологічне обладнання, комунікаційні зв'язки тощо), -  $W$ ;

- час витрачений на формування нової ВВС (будівні роботи, монтаж обладнання, формування комунікаційних зав'язків тощо), -  $T$ ;

- ризики, пов'язані зі створенням нової ВВС (проектні, технологічні тощо), -  $R$ .

У подальшому архітектуру ВВС будемо представляти у вигляді топологічного графу  $G$  у вершинах (вузлах), якого будуть розташовані окремі ТО, а ребра будуть представляти комунікаційні зв'язки між ТО. Граф топології  $G$  формально можна представити у вигляді

матриці сумісності  $\|M_{ij}\|$  де  $i, j = \overline{1, L}$ ;  $L$  - кількість вершин графу  $G$ . Для використання вершин та ребер графу  $G$  у наступних розрахунках та з урахуванням його зміни на матрицю  $\|M_{ij}\|$  будемо брати до уваги тільки ті елементи матриці, які задовольняють вимогам  $i < j$ ,  $i \neq j$ . Для оптимізації показників  $W, T, R$  при виборі раціональних архітектурних рішень щодо створення ВВС, скористаємося цілочисельним лінійним програмуванням з булевими змінними:

$$x_{ik} = \begin{cases} 1, \text{ якщо } k\text{-й ТО розташований} \\ \text{у } i\text{-й вершині } G \text{ графу ВВС,} \\ 0, \text{ в протилежному випадку} \end{cases} \quad (1)$$

У подальшому будемо враховувати наступні обмеження:  $\sum_{k=1}^m x_{ik} = 1, i = \overline{1, L}$ , що значить обов'язкове розташування  $k$ -го ТО у графі  $G$ ,  $m \leq L$ .

Для рішення задачі оптимізації архітектурних рішень ВВС введемо допоміжні булеві змінні:

$$y_{je} = \begin{cases} 1, \text{ якщо } y_j\text{-у вершину } G \text{ графу} \\ \text{розташований } e\text{-й ТО,} \\ 0, \text{ в іншому випадку.} \end{cases} \quad (2)$$

$$z_{ik,je} = \begin{cases} 1, \text{ якщо } y_i\text{-у вершину } G \text{ графу} \\ \text{розташований } k\text{-й ТО,} \\ a \text{ у } y_j\text{-у вершину розташований } e\text{-й ТО,} \\ 0, \text{ в іншому випадку.} \end{cases} \quad (3)$$

При цьому необхідно виконати наступні обмеження:

$$x_{ik} - y_{je} = 0, i, j = \overline{1, L}, k = \overline{1, m}, i = j, k = e, \quad (4)$$

$$L - \sum_{j=1}^L \sum_{e=1}^m z_{ik,je} = 0, i = \overline{1, L}, e = \overline{1, m}, \quad (5)$$

$$L - \sum_{i=1}^L \sum_{k=1}^m z_{ik,je} = 0, j = \overline{1, L}, k = \overline{1, m} . \quad (6)$$

Розглянемо мінімізацію показника витрат  $W$  для раціонального вибору архітектурних рішень при синтезі ВВС. Необхідно мінімізувати витрати  $W$ :

$$\min W, W = \sum_{i=1}^L \sum_{k=1}^m w_k x_{ik} + \sum_{j=1}^L \sum_{i=1}^L \sum_{k=1}^m \sum_{l=1}^m w_{ij} z_{ik,je} , \quad (7)$$

де  $w_k$  - витрати, пов'язані з використанням  $k$ -го ТО в архітектурі ВВС;  $w_{ij}$  - витрати на комунікацію між  $i$ -им та  $j$ -им вузлами в архітектурі ВВС.

При цьому необхідно виконати наступні обмеження:

$$T \leq T', T = \sum_{i=1}^L \sum_{k=1}^m t_k x_{ik} + \sum_{j=1}^L \sum_{i=1}^L \sum_{k=1}^m \sum_{l=1}^m t_{ij} z_{ik,je} , \quad (8)$$

де  $T'$  - максимально допустимі часові витрати на виконання робіт щодо створення ВВС;  $t_k$  - час, пов'язаний з монтажем  $k$ -го ТО у ВВС;  $t_{ij}$  - час витрачений на формування комунікацій між  $j$ -им та  $i$ -им вузлами обраної топології архітектури ВВС;

$$R \leq R', R = \sum_{i=1}^L \sum_{k=1}^m r_k x_{ik} + \sum_{j=1}^L \sum_{i=1}^L \sum_{k=1}^m \sum_{l=1}^m r_{ij} z_{ik,je} , \quad (9)$$

де  $R'$  - допустимий ризик, пов'язаний зі створенням ВВС;  $r_k$  - ризик, пов'язаний з придбанням та монтажем  $k$ -го ТО у ВВС;  $r_{ij}$  - ризик, пов'язаний з організацією комунікацій між  $i$ -им та  $j$ -им вузлами у обраній топології архітектури ВВС.

Для пошуку компромісного рішення, з урахуванням усіх показників, скористаємося мінімізацією комплексного показника  $F$  у вигляді суми зважених локальних показників  $\overline{W}$ ,  $\overline{T}$ ,  $\overline{R}$ :

$$\min F, F = \alpha_W \cdot \overline{W} + \alpha_T \cdot \overline{T} + \alpha_R \cdot \overline{R} , \quad (10)$$

де  $\alpha_W, \alpha_T, \alpha_R$  - "ваги" (значимість) показників  $\overline{W}$ ,  $\overline{T}$ ,  $\overline{R}$ ;  $\alpha_W + \alpha_T + \alpha_R = 1$ , тут  $\overline{W}$ ,  $\overline{T}$ ,  $\overline{R}$  - пронормовані (безрозмірні) представлення показників:

$$\overline{W} = \frac{W - W^*}{W' - W^*}, \quad \overline{T} = \frac{T - T^*}{T' - T^*}, \quad \overline{R} = \frac{R - R^*}{R' - R^*}, \quad (11)$$

де  $W^*, T^*, R^*$  - мінімальні значення показників  $W, T, R$ , які було отримано у результаті їх незалежної оптимізації;  $W', T', R'$  - допустимі значення показників.

Можлива така постановка задачі формування архітектури ВВС, коли вузли пов'язані з певним типом ТО, які розміщують у ВВС.

В цьому випадку в розглянутій задачі оптимізації (1)-(11) необхідно ввести фільтр у вигляді матриці  $\|p_{ik}\|$ , яка вказує на дозвіл або заборону розташування  $k$ -го ТО у  $i$ -й вузол ВВС, де:

$$p_{ik} = \begin{cases} 1, & \text{якщо дозволено розмістити у } i\text{-й вузол ВВС } k\text{-й ТО,} \\ 0, & \text{в іншому випадку.} \end{cases}$$

З урахуванням запропонованого фільтру  $\|p_{ik}\|$  задача мінімізації витрат, пов'язаних з раціональним розміщенням ТО для використання у вузлах архітектури ВВС, буде мати наступний вигляд.

Необхідно знайти

$$\min W, W = \sum_{i=1}^L \sum_{k=1}^m w_k p_{ik} x_{ik} + \sum_{j=1}^L \sum_{l=1}^L \sum_{k=1}^m \sum_{e=1}^m w_{ij} z_{ik,je} \quad (12)$$

з урахуванням обмежень

$$T \leq T', T = \sum_{i=1}^L \sum_{k=1}^m t_k p_{ik} x_{ik} + \sum_{j=1}^L \sum_{l=1}^L \sum_{k=1}^m \sum_{e=1}^m t_{ij} z_{ik,je} , \quad (13)$$

$$R \leq R', R = \sum_{i=1}^L \sum_{k=1}^m r_k p_{ik} x_{ik} + \sum_{j=1}^L \sum_{l=1}^L \sum_{k=1}^m \sum_{e=1}^m r_{ij} z_{ik,je} . \quad (14)$$

#### 4. Дослідження технології моделювання виробничого циклу та вузьких місць у сформованій розподіленій архітектурі віртуальної виробничої системи

Для дослідження часу виконання виробничого циклу у сформованій архітектурі віртуальної виробничої системи скористаємося імітаційним подійним моделюванням, яке реалізоване на агентній платформі JADE з використанням наступних агентів [6, 7]:

- 1 - агент «архітектура ВВС»;
- 2 - агент «часова затримка в вузлах» ВВС;
- 3 - агент «часова затримка між вузлами» ВВС;
- 4 - агент «технологічний маршрут»;
- 5 - агент «генератор замовлень»;
- 6 - агент «статистика»;
- 7 - агент «монітор».

Для ініціалізації замовлень «генератор замовлень» генерує заявки на обслуговування, які надходять до ВВС та рухаються у відповідності до вимог агента «технологічний маршрут». При цьому заявка проходить скрізь вузли та зв'язки між ними з урахуванням агентів «часова затримка в вузлах» та агента «часова затримка між вузлами». Планування руху заявок здійснює агент «монітор». Результати моделювання видає агент «статистика». До результатів моделювання включено: кількість замовлень, які було обслужено у ВВС, середній час виконання одного замовлення, завантаження окремих ТО, порушення планових строків виконання замовлень тощо. На рис. 1 представлено структурну схему агентної моделі.



Рис. 1. Структурна схема агентної моделі

Для дослідження вузьких місць у сформованій структурі ВВС в агентну модель добавлено агент «черга» (рис. 1). Виникнення черг при виконанні замовлень пов'язане з вузькими місцями у вигляді ТО з низькою пропускнуою здатністю, що призводить до збільшення часу



виконання замовлень. Наявність вузьких місць вимагає прийняття конструктивних рішень щодо їх усунення (включення до ВВС більш продуктивних ТО, формування раціонального план-графіка запуску замовлень тощо).

### **5. Висновки та перспективи подальших досліджень**

Проведено дослідження архітектурних рішень при формуванні віртуальної розподіленої виробничої системи за допомогою оптимізаційної моделі. Проведено пошук раціональних рішень щодо створення архітектури віртуального виробництва у вигляді множини технологічних об'єктів та комунікацій між ними в умовах обмежених можливостей підприємства, що розвивається. Для дослідження динамічних процесів, пов'язаних з виконанням інноваційних замовлень, використовується подійне імітаційне моделювання у вигляді агентної моделі. Досліджуються вузькі місця у віртуальному розподіленому виробництві у вигляді великих черг, які виникають при обслуговуванні заявок і викликають затримки виконання замовлень.

Наукова новизна проведеного дослідження пов'язана з розробкою методу синтезу архітектури розподіленої виробничої системи, заснованої на статичному (структурному) та динамічному представленні розподіленого віртуального виробництва з пошуком раціональної структури у вигляді вузлів з технологічним обладнанням і комунікацій між ними та моделювання процесу виконання інноваційного портфелю замовлень.

Запропонований підхід дозволяє на початкових етапах синтезу віртуальної розподіленої виробничої системи сформувати раціональну архітектуру з урахуванням топології, зв'язків між окремими віддаленими технологічними об'єктами та обґрунтувати комунікаційну інфраструктуру для задач управління виробництвом.

Подальші дослідження будуть спрямовані на вирішення задач розташування віртуального виробництва на земній поверхні, що призводить до витрат, пов'язаних з проведенням земельних та будівельних робіт, появою додаткових ризиків, у тому числі екологічних, необхідністю врахування кліматичних факторів, пов'язаних з різними природними зонами експлуатації віртуального розподіленого виробництва.

**Список літератури:** 1. *Lindgren, M., Bandhold, H.* Scenario Planning The link between future and strategy // Palgrave Macmillan UK, 2002. - 180 p. DOI: 10.1057/9780230511620. 2. *Uskenbayeva, R. K., Kurmangaliyeva, B. K., Yedilkhan, D.* Situational Management for Process Implementation of Working Operations of the Business Process // 54th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE). - Hangzhou, China, 2015. - P. 292-297. DOI: 10.1109/SICE.2015.7285573. 3. *Федорович, О.Є., Прончаков, Ю.Л.* Метод формування логістичних транспортних взаємодій для нового портфелю замовлень розподіленого віртуального виробництва // *Авіаційно-космічна техніка і технологія*, 2020, № 2, С. 102-108. 4. *Pawluczuk, Ju.* К проблеме управления производственными ресурсами предприятия // *Zarz?dzanie : Teoria i praktyka*. - 2011. - № 1(3). - С. 17-26. 5. *Paulsen, S., Boens, J.* Summary of the Workshop on information and communication technologies supply chain risk management // National Institute of Standards and Technology, 2012. - 21 p. 6. *Кравець, Р. О.* Динамічна координатія стратегій мультиагентних систем // *Бюлетень Національного університету "Львівська політехніка"*. - 2011. - No. 699. - P. 134-144. 7. *Fedorovich, Oleg, Uruskiy, Oleg, Pronchakov, Yuri, Lukhanin, Mikhail.* Method and information technology to research the component architecture of products to justify investments of high-tech enterprise // *Радіоелектронні і комп'ютерні системи*, 2021. - № 1 (2021).

*Надійшла до редколегії 17.06.2021*

**Прончаков Юрій Леонідович**, кандидат технічних наук, доцент, декан факультету програмної інженерії та бізнесу Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут». Наукові інтереси: логістика розподілених виробничих систем. Адреса: Україна, м. Харків, вул. Чкалова, 17, тел. +38 (095) 577 38 57, +38 (050) 637 87 68.

*С.В. ГУБАРЕНКО, М.С. ГУБАРЕНКО, М.В. АНТОНЮК*

## РОЗПІЗНАВАННЯ ОБ'ЄКТІВ У ВІДЕОПОТОЦІ

Описані проблеми попередньої обробки зображень у відеопотоці для подальшого аналізу і розпізнавання об'єктів. Розглянуті основні способи та підходи до компенсації недоліків чи дефектів зображень, серед яких оптимізація контрастності зображення, різкості, нормалізація освітлення, пошук і відбракування схожих зображень. Запропоновано використання методу еквіваріантного детектора для розпізнавання об'єктів, що швидко рухаються. Синтезована нейрона мережа, яка пройшла навчання для розпізнавання транспортних засобів та обличч людини, ймовірно водія. Запропоновано підхід для вдосконалення алгоритму пошуку локальних ознак при навчанні нейронної мережі. Наведені та проаналізовані результати експерименту з використанням навченої мережі при розпізнаванні окремих об'єктів та композитних сцен.

### 1. Вступ

Розпізнавання зображень є важливим компонентом систем управління, обробки інформації та прийняття рішень. Завдання, пов'язані з класифікацією і ідентифікацією предметів, явищ і сигналів, що характеризуються кінцевим набором деяких властивостей і ознак, виникають в таких сферах як робототехніка, інформаційний пошук, моніторинг та аналіз візуальних даних, дослідження штучного інтелекту. На даний момент широко використовуються системи розпізнавання рукописного тексту, автомобільних номерів, відбитків пальців або людських обличч, що знаходять застосування в інтерфейсах програмних продуктів, системах безпеки та аутентифікації особи [1].

За останній час з появою методів зниження розмірності, згортальних нейронних мереж, *deep learning* і констеляційних моделей у розпізнаванні візуальних образів був досягнутий істотний прогрес. Однак, незважаючи на досягнуті успіхи, сучасні дослідження підтверджують той факт, що алгоритми розпізнавання об'єктів не можуть повноцінно замінити людину.

Актуальним залишається питання розпізнавання зображень тривимірних об'єктів під різними кутами зору, що піддаються перетворенням обертання, масштабування і трансляції. Сучасні підходи до вирішення цього питання, такі як багатошарові згорткові нейронні мережі, а також використання інваріантних детекторів ознак SIFT і ORB [2], в даний момент пропонують часткові рішення, що не забезпечують достатньої точності розпізнавання і втрачають інформацію про структуру об'єкта. Існують проблеми з обробкою потокового відео і виявлення об'єктів, що рухаються. Також проблемою є розпізнавання нечітких розмитих зображень або зображення перекриті іншими об'єктами в отриманих кадрах.

Використання камер як уніфікованого пристрою для визначення множини параметрів рухомих об'єктів (відстані, швидкості, метричні параметри) дозволить знизити собівартість системи і спростити формалізацію одержуваної інформації за рахунок зменшення різновидів застосовуваних технічних пристроїв, а також, без додаткових налаштувань робочого місця і додавання апаратних засобів, підвищити багатofункціональність системи контролю.

**Мета дослідження** полягає у вдосконаленні методу розпізнавання об'єктів, які рухаються, за рахунок використання еквіваріантного детектора на етапі навчання та використання нейронної мережі, що повинно покращити роботу систем автоматизованого та автоматичного моніторингу.

У рамках дослідження приділялась увага геометричним, кінематичним та динамічним характеристикам стану і поведінки рухомих та нерухомих об'єктів, які виявляються інформаційно-вимірjuвальними системами моніторингу в системах з рухомими об'єктами.

Теоретичне і практичне значення дослідження полягає у тому, що розроблені і реалізовані алгоритмічне, математичне і програмне забезпечення становлять основу вимірjuвальної системи, яка може використовуватися для віддаленого контролю і моніторингу транспортних засобів, людей, технологічних процесів в різних сферах діяльності, а також входити

до складу комплексу технічного зору, що забезпечує автономне функціонування транспортно-технологічних комплексів, безпілотних систем.

## 2. Проблеми на етапах розпізнавання об'єктів у відеопотоці

Процес розпізнавання об'єктів у відеопотоці може бути розділений на наступні етапи (рис. 1).

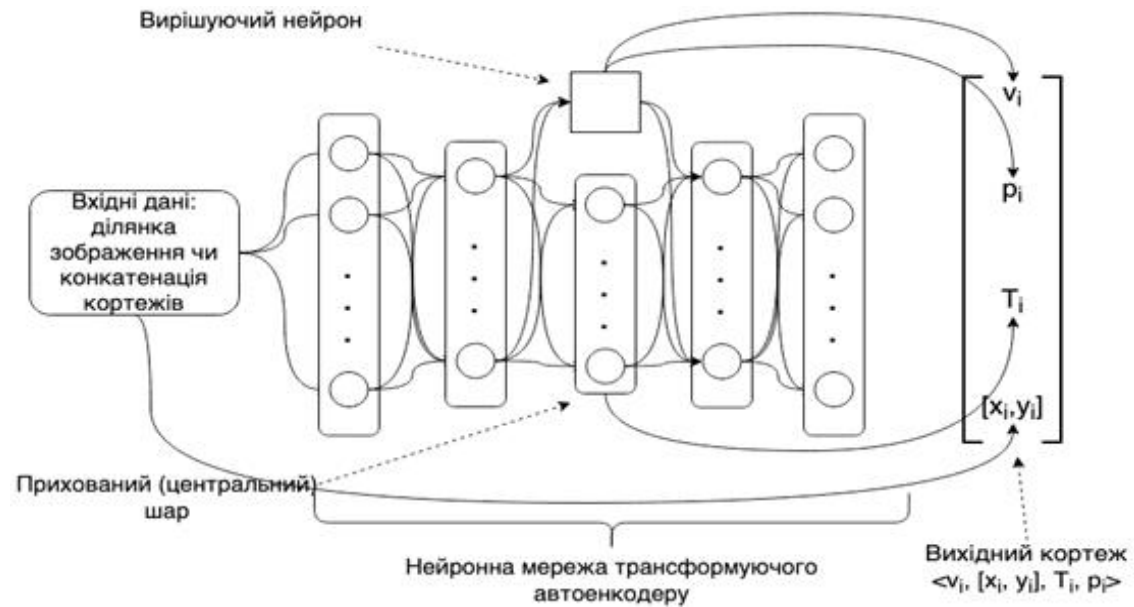


Рис.1. Схема еквіваріантного детектора на базі трансформуючого автоенкодера

Етап 1. Порівняння кадрів у відеопотоці - серед проблем, які виникають на цьому етапі, слід зазначити незмінність кадрів. Частіше за все відеопотік має 16-30 кадрів у секунду, часто виникають ситуації, коли кадри ідентичні, або мають незначні відмінності. Існують три основних підходи до вирішення такої проблеми [3]: порівняння значення хеш-функцій двох кадрів, які змінюються; обчислення коефіцієнту кореляції; побудова та аналіз SURF-дискриптів.

Етап 2. Оцінка якості зображення - серед проблем, які виникають на цьому етапі, слід зазначити розмиття, наявність шумів та засвічення кадрів. З багатьох причин зображення у кадрі може бути пошкоджено, іноді якість кадру може бути дуже низькою і не придатною до розпізнавання, такі кадри слід відкинути. На цьому етапі слід провести оцінку контрастності, різкості та чіткості [4]. Можна також підвищити різкість або компенсувати недоліки якості зображення.

Етап 3. Зменшення розмірності зображення - серед проблем, які виникають на цьому етапі, слід зазначити вхідні кадри високої розмірності. Кадри надходять у вигляді матриці пікселів, чим більша розмірність, тим більше операцій буде виконано і більше часу на розпізнавання буде витрачено. Іноді є можливим зменшити розмірність зображення, при цьому залишивши достатньо даних для обробки [5].

Етап 4. Розпізнавання об'єктів на зображенні - серед проблем, які виникають на цьому етапі, слід зазначити оклюзії та трансформацію. Іноді об'єкти повертаються, віддаляються або перекриваються іншими об'єктами. Такі проблеми вирішуються завдяки використанню відповідних алгоритмів розпізнавання [5].

Етап 5. Отримання результату розпізнавання.

### **3. Удосконалення методу еківаріантного детектора розпізнавання об'єктів, що швидко рухаються**

Головні особливості методологій та алгоритмів розпізнавання об'єктів, що швидко рухаються, а також програмного забезпечення, створеного на їх основі, визначаються особливостями предметної області та апаратного забезпечення. На даний момент складно створити універсальну систему для розпізнавання різних класів об'єктів та подальшого їх аналізу. Вузкими місцями подібної системи зараз є обчислювальні можливості комп'ютерів та недосконалість алгоритмів. У дослідженні пропонується розглядати переважно розпізнавання рухомих об'єктів транспортного типу. Програмне забезпечення, яке використовується для вирішення задач подібного типу, зараз є одним з найбільш затребуваних. Таке забезпечення може використовуватися для спостереження за дотриманням правил дорожнього руху на шляхбаумах, пропускних пунктах, залізничних шляхах, магістралях, для моніторингу трафіку, знаходження вузьких місць в трафіку на дорогах, для перепустки в якісь місця тощо.

Вхідними даними є кадри з відеопотоку чи запис трафіку.

Виходом є навчена нейронна мережа, результат розпізнавання у вигляді областей знайдених об'єктів та ідентифікований об'єкт.

Слід виділити наступні етапи роботи алгоритму розпізнавання:

Етап 1. Оптимізація обробки відеопотоку. На даному етапі вибираємо зображення з серії кадрів, які відрізняються, знаходимо кращий кадр за якістю серед схожих для більш детального аналізу зображення. Даний етап потрібен для оптимізації швидкості роботи алгоритму: не має сенсу повністю аналізувати кожен кадр з відеопотоку, адже існує велика ймовірність того, що зображення ідентичні або майже незмінні. Важливо також вибрати для подальшого аналізу чіткіше і контрастніше зображення серед вибірки з декількох схожих задля покращення процесу розпізнавання.

Етап 2. Стиснення, зменшення розмірності зображення з мінімальною втратою інформативності. Даний крок необхідний для оптимізації швидкодії алгоритму, адже, як правило, необроблені зображення мають дуже високі розмірності, що ускладнює обробку і потребує більше ресурсів і часу для аналізу.

Етап 3. Знаходження границь об'єктів на зображенні. На даному етапі знаходимо контури об'єктів, встановлюємо їх кількість і розташування на зображенні. Таким чином, надалі ми будемо аналізувати лише частини зображення, які нас цікавлять.

Етап 4. Аналіз інформативних параметрів об'єкта. На даному етапі знаходимо колір, розміри, віддаленість від камери, швидкість руху, позицію відносно камери, напрям руху. Всі ці характеристики далі використовуються для вирішення бізнес-задач конкретної системи.

Етап 5. Виділення класу об'єкта та його виду. На даному етапі аналізуємо, що за об'єкт був знайдений, його клас та тип, вирішується, чи потрібен він для роботи системи, чи є аномалією, далі він може класифікуватися, а також використовуватися для навчання адаптивних нейронних мереж для самовдосконалення системи.

Було проведено експериментальну оцінку результатів дослідження. Отримані результати роботи алгоритму розпізнавання було порівняно з результатами роботи відомих аналогічних алгоритмів, зокрема, були порівняні долі успішно розпізнаних зображень. Як тестові вибірки було взято зображення, згенеровані спеціальними програмними засобами, а також відомі тестові набори, які знаходяться у відкритому доступі.

### **4. Синтез моделі нейронної мережі з використанням еківаріантного детектора на базі трансформуючого автоенкодера**

Алгоритм розпізнавання з використанням еківаріантного детектора, побудованого на базі трансформуючого автоенкодера (рис. 1), де вихідними даними для мережі є ділянка зображення (для першого рівня ієрархії) або конкатенація вихідних кортежів з детекторів нижчих рівнів (для другого та вищих за номером рівнів). Результатом є кортеж розмірністю  $n \langle v_i, [x_i, y_i], T_i, p_i \rangle$ , де  $v_i$  - значення ідентифікатора функції детектора, вираженого як вектор-маска;  $[x_i, y_i]$  - відносні координати центру детектора;  $T_i$  - значення трансформації, вира-

жене як матриця афінної трансформації;  $p_i$  - значення впевненості детектора на відріжку  $[0;1]$ .

Детектори є частинами ієрархічної структури, кожен рівень ієрархії відповідає локальним ознакам або групі ознак, які описують певну частину об'єкта (рис. 2). Перший рівень ієрархії містить детектори мінімальних локальних ознак об'єкта (наприклад, фара, лобове скло чи колесо автомобіля). Наступний рівень описує групи локальних ознак, які можна об'єднати у більш складну локальну ознаку. Останній рівень ієрархії містить один детектор, який описує весь об'єкт і включає у себе локальні ознаки попередніх рівнів. Між першим та останнім рівнем ієрархії може бути скільки завгодно рівнів.

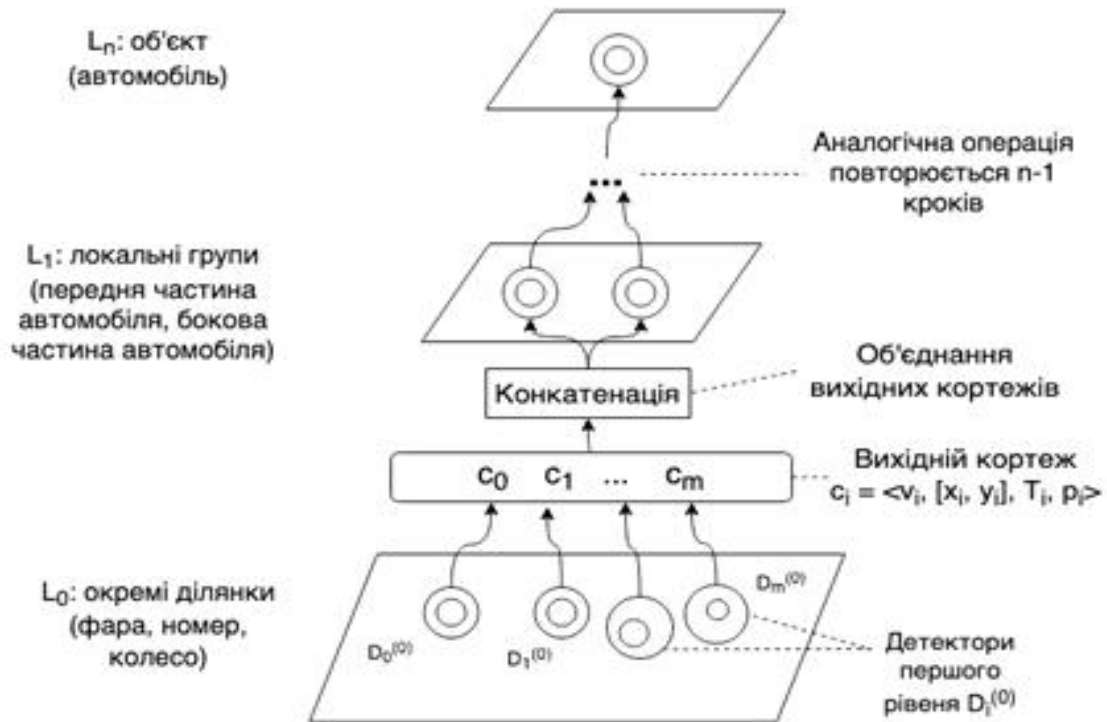


Рис.2. Схема ієрархічної організації локальних еквіваріантних детекторів

## 5. Вдосконалення алгоритму пошуку локальних ознак

Алгоритм розпізнавання складається з послідовної активації детекторів моделі, починаючи з першого рівня, на заданому зображенні. Нехай  $g: X \rightarrow Y$ , аргументами якої є зображення  $x_n \in X$ , представлені в вигляді вектору довжини  $n$ , а значеннями функції - множина класів (категорій)  $y \in Y$ , варійована в залежності від поставленого завдання. Є також підмножина пар аргументів і значень функції  $D = \{(x_0, y_0), \dots, (x_n, y_n)\}$ . Таким чином, ієрархія детекторів, що послідовно активуються, реалізує функцію  $h: X \rightarrow Y$ , яка апроксимує функцію  $g$  на всій її області визначення, в тому числі в точках, які не включені в  $D$ . Для розрахунку значення  $h(x)$  зображення надходить на вхід першого шару навченої системи, потім виконується послідовна активація локальних детекторів на кожному з рівнів. Вихідне значення ієрархії являє собою бінарне число, що визначає належність зображення до класу, при цьому вихідний рівень також проводить оцінку параметрів локалізації зображеного об'єкта, якщо значення активації дорівнює 1 (зображення успішно розпізнано і належить класу).

У такому вигляді розроблена система розпізнавання використовується для вирішення задачі унарної класифікації, коли множина класів  $y \in Y$  представлена двома елементами

$\{0,1\}$ . Функція, таким чином, дорівнює 1 у випадках, коли зображення, що служить її аргументом, містить об'єкт, що належить класу, і 0 в іншому випадку. Для випадків, коли потрібно розпізнати зображення серед кількох можливих класів (завдання мультикласової класифікації), проводиться навчання окремої ієрархії детекторів для кожного конкретного класу, і потім проводиться послідовна перевірка зображення на позитивну відповідність кожного з них. В цьому випадку розглядається функція  $g'$ , визначена на множині  $X'$ , значеннями якої є множина класів, така що для обраного  $j$ -го класу  $Y' = \{y_i, U_{j \neq i} y_i\}$ .

Нехай дана навчена ієрархія детекторів  $M_c$  для деякого класу зображень  $c$  (наприклад, людських обличчя), або кілька ієрархій для завдання мультикласового розпізнавання, і зображення  $I$ , яке необхідно розпізнати. Алгоритм розпізнавання складається з наступних кроків:

Крок 1. Вибираємо ієрархію детекторів  $M_c$ .

Крок 2. Для  $l$ -того рівня ієрархії, починаючи з першого, і для кожного детектора відповідного рівня  $D_j^{(l)}$  складаємо вектор вихідних даних  $Z^{(l)}$ .

Крок 3. Якщо  $l=0$ , то заповнимо вектор вихідних даних наступним чином: розраховуємо значення функції ідентифікації детектора  $d_{I_j}^{(l)}$  і значення впевненості  $d_{p_j}^{(l)}$  для кожної ділянки зображення  $I(x..x+w, y..y+h)$ , де  $w$  та  $h$  відповідають розмірам локальної ділянки зображення, і проведемо конкатенацію вектора  $Z^{(l)}$  з вихідними значеннями детектора для тих ділянок, де  $d_{I_j}^{(l)} = 1$ , та  $d_{p_j}^{(l)} > t$ , де  $t$  - обрана ступінь впевненості.

Крок 4. Якщо  $l \neq 0$ , вектор вхідних даних отримаємо як  $Z^{(l)} = (D_0^{(l)}(Z^{(l-1)})) \parallel \dots \parallel (D_j^{(l)}(Z^{(l-1)})) \parallel (0_0 \parallel \dots \parallel 0_{N_l})$ , де  $N_l$  - максимальна кількість можливих детекторів на рівні  $l$ .

Крок 5. Якщо для рівня  $l$  активації всіх детекторів цього рівня негативні (дорівнюють нулю), то зображення не належить класу  $c$ .

Крок 6. В іншому разі інкрементуємо значення  $l$  і повторюємо алгоритм з другого кроку.

Крок 7. Якщо активація останнього рівня позитивна, то зображення належить класу  $c$ . Інакше повторюємо алгоритм з першого кроку.

Використовуючи той факт, що низькорівневі локальні ділянки зображення (взяті з достатнім масштабом), як правило, представляють собою край, межі і кути, слід зазначити, що має місце тенденція до зростання різноманітності серед еквіваріантних детекторів на більш високих рівнях моделі. Експерименти показують, що кількість детекторів першого рівня при навчанні не перевищує 10, при цьому ієрархії, навчені на об'єктах різних категорій, здатні розділяти між собою частину детекторів першого рівня, демонструючи ефект, що нагадує трансферне навчання або попереднє навчання без вчителя.

## 6. Аналіз результатів експерименту

Для оцінки ефективності розробленого алгоритму розпізнавання зображень проведемо експерименти для трьох варіантів алгоритму об'єднання в одній категорії зображень, різних з інформатико-теоретичної точки зору (таких, як зображення людського обличчя в профіль і в фас), але відповідних при цьому одній категорії приналежності.

Варіант 1. Навчання локального еквіваріантного детектора в якості окремого класифікатора. Метою експериментів є визначення точності відновлення фрагментів зображення за компактними репрезентаціями, адекватності сформованих репрезентацій, їх відповідність значенням трансформацій. Крім цього, тестується ефективність роботи детектора в якості опції-ідентифікатора. У завершальній частині проводиться зіставлення з іншими алгоритмами унарної класифікації та відновлення трансформацій.

Варіант 2. Розпізнавання зображень за допомогою повноцінної ієрархії локальних еквіваріантних ознак. Об'єктами експерименту є показники помилки розпізнавання (в порівнянні з відомими алгоритмами), а також сформовані моделлю просторові структури еквіваріантних детекторів.

Варіант 3. Робота системи на узагальнених відеофрагментах, що містять множину об'єктів різних класів. Досліджується здатність моделі кластеризувати сцену, виявляючи окремі об'єкти і формуючи відповідні їм репрезентації.

Оскільки представлена модель вимагає для навчання обмежену вибірку з повнозв'язних відеофрагментів - тип даних, який не отримав поширення серед сучасних методів розпізнавання зображень, - навчання моделей для експерименту проводилося на даних, представлених автором. Навчальні вибірки включають в себе як відеофрагменти, отримані в результаті обробки згенерованих з використанням комп'ютерної графіки тривимірних моделей, так і фрагменти живої зйомки навколишнього світу.

Оцінка ефективності розпізнавання людських обличч проводилася на відкритих базах даних LFW [6], HPID (Head Pose Image Database) [7]. Деякі моделі, що використовують альтернативні методи розпізнавання, надані бібліотекою Caffe [8]. Також для експериментів використовувалися дані, отримані за допомогою програми тривимірного моделювання FaceGen [9], що представляють собою марковані зображення людських обличч в різних орієнтаціях щодо камери.

Експериментальна перевірка розробленого методу розпізнавання проведена відповідно до рекомендацій з навчання і порівняння моделей розпізнавання образів в рамках апарату теорії розпізнавання образів і машинного навчання [10, 11]. Для оцінки ефективності методів, які використовуються в експерименті, використовувалися показники точності і повноти (що відображають помилки першого і другого роду).

Для оцінки результативних показників застосовувався метод розрахунку довірчих інтервалів. Відповідно до прийнятої практикою постановки експерименту в області розпізнавання і рекомендаціями щодо вибору рівня довіри, значення рівня довіри  $p$  вибрано рівним 0.95 [12]. Експериментальні дані продемонстрували, що для досягнення відповідного рівня досить вибірки, яка складається з близько 500 примірників, що для авторського методу відповідає короткому відеофрагменту, який за часом не перевершує 15 секунд. Частота зміни кадрів дорівнює 24 кадри за секунду, за умови використання методу пермутації.

Оскільки специфіка розробленого методу полягає у визначенні просторової локалізації об'єктів під різними кутами зору, то експериментальні вибірки для навчання і тестування містять набори відповідних зображень. Для оцінки здатності методу розпізнавання стійким чином обробляти об'єкти під впливом відповідних інваріантних перетворень, для кожного етапу експерименту результати обчислювалися за допомогою агрегації показників серії експериментальних перевірок. При цьому кожна серія складається з наступних елементів: навчання і тестування на загальній вибірці, навчання і тестування випадковою вибіркою, навчання і тестування на кластеризованій вибірці. Кластеризація включає в себе розбивку вибірки на відеофрагменти, що містять зображення окремих об'єктів (конкретних людських осіб), при цьому дані відеофрагментів навчання не перетинаються з тестовими. Таким чином, результатом експерименту є агрегований показник серії з двох стадій навчання і тестування з урахуванням розрахованої на базі складових показників стандартної помилки.

В рамках експерименту проводиться зіставлення результатів використання розробленого методу і альтернативних методів розпізнавання зображень, що використовуються в даний час. Вибрані альтернативи широко застосовуються як у виробництві (метод Віюлі-Джонса [13], згорткові нейронні мережі [14]), так і в академічних дослідженнях. При виборі альтернатив використовувалися дані відкритих змагань з машинного навчання в області розпізнавання зображень, таких як ILSVRC [15]. Реалізації алгоритмів, що використовувалися в ході експерименту, представлені авторами відповідних методів і отримані з відкритих джерел [10,16].

На базі отриманої вибірки, згрупованої попарно, проводиться навчання розрідженого трансформуючого автоенкодера. В експерименті брали участь чотири моделі автоенкодерів, позначені нижче як А, В, С і D, що відрізняються кількістю нейронів першого і останнього прихованих шарів.

- А - 36x36 нейронів;
- В - 64x64 нейрони;
- С - 256x256 нейронів;
- D - 768x768 нейронів.

Вибір кількості нейронів зроблено відповідно до відомих експериментів з навчання автоасоціативних нейронних мереж [17]. Трансформації представляють собою поворот камери по осях  $X$  і  $Z$ .

У табл. 1 наведені результати розпізнавання для різних категорій фрагментів зображень, взятих з відеофрагментів (складання вибірки проводилося в автоматичному режимі, методом трекінгу; категорії промарковані назвами для спрощення і зручності аналізу).

Таблиця 1

	A (36x36), %	B (64x64), %	C (256x256), %	D (768x768), %
Обличчя: очі	79±4	95±3	96±3	96±3
Обличчя: ніс	82±3	92±3	93±3	94±2
Обличчя: рот	84±3	97±4	97±3	97±3
Обличчя: вуха	72±3	91±3	93±4	93±3
Обличчя: деталі контуру	92±4	96±3	97±4	98±3
Автомобіль: колеса	72±4	92±4	94±3	95±4
Автомобіль: фари	74±3	91±3	92±2	92±2
Автомобіль: лобове скло	80±3	94±4	95±2	95±3
Автомобіль: заднє скло	82±4	92±2	93±3	93±3
Автомобіль: бампер	79±4	90±3	92±4	93±4
Автомобіль: деталі контуру	78±2	89±4	90±3	91±3

З отриманих даних можна зробити висновок про оптимальність моделі - збільшення числа нейронів веде до незначного підвищення точності реконструкції, при цьому збільшуючи обчислювальне навантаження. Також більше число нейронів вимагає більшого числа ітерацій навчання для досягнення збіжності. Найбільш проблемними випадками для роботи еківаріантного детектора є аномальні ситуації, які не зустрілися у відеофрагменті - наприклад, оклюзія шуканої ділянки зображення сторонніми предметами. Крім ідентифікації ділянок зображення, вторинною функцією еківаріантного детектора є оцінка параметрів інстанціювання або оцінка позиції об'єкта. Для експериментальної перевірки точності цієї оцінки використовувалися змішані дані вибірки HPID і згенерованої вибірки зображень осіб FaceGen з подальшою пост-обробкою візуальними ефектами розмиття і випадкової оклюзії. Для зіставлення результатів використовувалися такі методи як класичний алгоритм POSIT [18], і навчається з учителем регресор - випадковий ліс [19]. Оскільки прогноз трансформації являє собою регресію, а не класифікацію, як показник ефективності обрано відносну похибку, виражену у відсотках. Результати представлені у табл. 2.

В результаті експерименту виявлено, що еківаріантний детектор показує більш точну оцінку просторових параметрів об'єкта в порівнянні з альтернативними методами. Істотна перевага над класичними методами комп'ютерного зору пояснюється тим, що алгоритм POSIT вимагає для оцінки позиції наявності маркерів, які можуть бути спроектовані на об'єкт за допомогою методів епіполярної геометрії.

Було проведено дослідження ефективності простої дворівневої моделі на трьох категоріях зображень людських обличчя: звичайні зображення осіб, зображення з штучним зашумленням за допомогою оклюзії і розмиття. Ефективність розпізнавання розраховувалася за допомогою показників точності і повноти та порівнювалася з аналогічними показниками альтернативних методів розпізнавання.

Оскільки однією з основних переваг даної моделі є еківаріантність - здатність до ідентифікації об'єктів на зображенні в різних орієнтаціях - то експериментальні вибірки згруповані таким чином: розглядається деяка вихідна позиція об'єкта з координатами обертання (кутів Ейлера)  $(0,0,0)$ , при цьому в групу, що характеризується значеннями  $(\varphi_{\min}, \varphi_{\max})$ , входять зображення об'єктів, які зазнали трансформації обертанням  $(\varphi_i, \varphi_j, \varphi_k)$ , такої, що для будь-якого  $\varphi \in (\varphi_i, \varphi_j, \varphi_k)$  є вірною нерівність  $\varphi_{\min} < \varphi < \varphi_{\max}$ .



Таблиця 2

	POSIT, %	Випадковий ліс, %	Авторський метод, %
Обличчя: очі	25±3	17±5	11±3
Обличчя: ніс	27±3	14±2	12±3
Обличчя: рот	18±4	16±4	10±4
Обличчя: вуха	24±4	12±4	11±2
Обличчя: деталі контуру	20±2	14±3	14±4
Розмиття: очі	27±3	19±3	12±3
Розмиття: ніс	32±3	18±3	14±2
Розмиття: рот	25±3	15±3	12±4
Розмиття: вуха	21±2	17±2	14±3
Розмиття: деталі контуру	31±4	16±4	12±3
Оклюдія: очі	32±4	20±3	10±5
Оклюдія: ніс	28±2	16±5	14±2
Оклюдія: рот	31±5	16±2	13±3
Оклюдія: вуха	31±3	18±4	11±4
Оклюдія: деталі контуру	30±4	17±4	13±4

Оскільки для цього експерименту потрібна значна кількість зображень об'єктів з різних кутів огляду, для його проведення була використана вибірка, отримана з використанням комп'ютерної графіки і генерації осіб програмою FaceGen.

Тестування розпізнавання осіб проводилося шляхом зіставлення методом Віоли-Джонса, класифікатора SVM в поєднанні з обчисленням гістограми орієнтованих градієнтів і згорткових мереж, навчених на вибірці ImageNet. Навчені моделі були надані бібліотеками Caffe з OpenCV. Результати наведені у табл.3.

Таблиця 3

	Метод Віоли-Джонса, %	SVM+HOG, %	CaffeNet, %	Авторський метод, %
(0°,15°)	85±3	84±5	87±4	92±3
(15°,30°)	72±3	75±2	85±4	91±3
(30°,45°)	67±4	72±4	86±2	89±4
(45°,60°)	66±4	78±4	83±3	86±2
(60°,90°)	86±2	77±3	82±3	90±4
(90°,120°)	67±3	80±3	83±3	86±3

Метод Віоли-Джонса потребує наявності окремої стадії навчання для кожної орієнтації. В рамках експерименту використовувалася попередньо навчена модель, яка продемонструвала зіставні результати для фронтальної і профільної орієнтації осіб, але вкрай низькі результати в проміжних станах. При використанні SVM в поєднанні з методами зниження розмірності і підвищення інваріантності (гістограма орієнтованих градієнтів) для класифікатора характерно зниження точності в міру включення до вибірки зображень різних орієнтацій. Така поведінка пов'язана з тим, що модель в процесі навчання намагається виробити ознаки, які однаково підходять для всіх зображень у вибірці, в результаті отримуємо рівномірно розподілені невисокі значення точності. Серед порівнянних методів авторська модель поступається тільки глибиною згорткової мережі, здатної до навчання різних (таких, що не змішуються) локальних ознак для різних орієнтацій об'єкта.

Слід зазначити, що на відміну від методу Віоли-Джонса, розроблена система розпізнавання дозволяє виділити комплексну структуру голови людини, не обмежуючись ділянкою,

що містить очі і рот. Такий ефект є наслідком використання ознак деталей контуру обличчя.

Порівняння результатів розпізнавання зображень обличчя, підданих впливу, розмиття проводилося за допомогою методів SVM + HOG, глибокої згорткової мережі і алгоритму випадкового лісу. Вибірка проводилася на базі CVLAB Dataset і включала в себе об'єкти, подані з різних кутів (табл. 4). Шумом є розмиття по Гаусу зі значенням  $\sigma = 0.5 \dots 2.5$ .

Таблиця 4

	Випадковий ліс, %	SVM+HOG, %	CaffeNet, %	Авторський метод, %
(0°,15°)	72±4	72±4	85±2	91±4
(15°,30°)	73±3	74±3	81±3	88±3
(30°,45°)	75±2	76±4	80±4	88±4
(45°,60°)	72±3	74±4	82±4	89±4
(60°,90°)	71±3	73±4	83±3	91±3
(90°,120°)	73±3	72±4	83±3	87±4

При впливі ефекту розмиття дисперсія результатів по відношенню до орієнтації об'єкта знижується.

Як завершальні групи експериментальної вибірки використовувалися зображення людських обличчя під впливом шуму оклюзії (часткового перекриття). Шум оклюзії згенерований за допомогою випадкового розміщення на зображеннях осіб непрозорих геометричних фігур. При генерації шуму оклюзії параметри розміщення фігур підібрані таким чином, щоб залишати відкритою мінімум дві третини площі зображення. Результати наведені у табл. 5.

Таблиця 5

	Випадковий ліс, %	SVM+HOG, %	CaffeNet, %	Авторський метод, %
(0°,15°)	72±4	72±3	84±3	90±5
(15°,30°)	69±3	66±4	82±3	89±4
(30°,45°)	70±3	68±4	83±2	88±3
(45°,60°)	71±4	70±4	81±4	89±4
(60°,90°)	70±2	67±3	84±5	88±3
(90°,120°)	72±3	67±3	85±4	87±3

Як демонструють результати, оклюзія є суттєвою перешкодою для алгоритмів розпізнавання, що використовують компактні цілісні представлення, таких як випадковий ліс і SVM+HOG - для деяких груп орієнтацій спостерігається падіння точності розпізнавання до 10%. Інструменти розпізнавання, які використовують локальні ознаки (наприклад, розроблена система і мережа CaffeNet), менш чутливі до часткового перекриття локальних областей зображення. У таких ситуаціях продуктивність представленого методу наближається до показників основного конкурента - згорткових мереж.

## 7. Аналіз композитних сцен

В даному експерименті перевірялася здатність розробленої системи за допомогою багаторівневої моделі аналізувати сцени, що містять безліч об'єктів, і автономно (без наявності вчителя) класифікувати їх між собою. Як вибірки використовувалися відкриті дані, що містять відеозаписи камер спостереження за дорожнім трафіком. Оцінка продуктивності алгоритму проводилася в такий спосіб: підраховувалася кількість об'єктів в кадрі, аналогічним чином вручну робилося розбиття по групах, потім ці значення зіставлялися зі значеннями знайденими алгоритмом (рис. 3).

У табл. 6 значення наведені у відсотках від фактичної кількості об'єктів. Слід зазначити, що на відміну від ситуації з контрольованим трекінгом і рухом камери навколо об'єкта, алгоритм в режимі аналізу композитних сцен зі сторонніх відеофрагментів позбавлений інформації про фактичний рух об'єкта навколо камери. Так система не здатна без додаткової інформації визначити, що рух від камери автомобіля

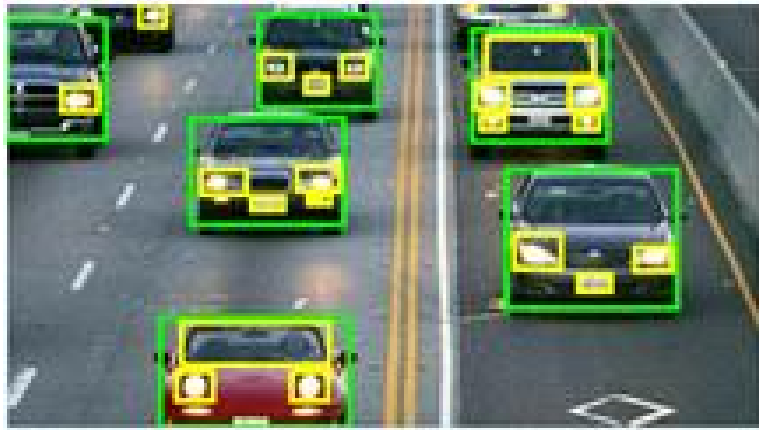


Рис.3. Результат розпізнавання запису «трафік 2»

що віддаляється являє собою переміщення по площині, непаралельній площині сенсору камери. У загальному випадку в такій ситуації алгоритм знаходиться в умовах невизначеності і здатний сформувати тільки обмежене представлення об'єкта за допомогою моделі еківаріантних детекторів. Існує можливість, використовуючи методи проєктивної геометрії, ввести деякі припущення в розрахунок оптичного трекінгу - так для розглянутих експериментальних випадків стадія оптичного трекінгу була доповнена умовою жорсткості (фіксованою формою) об'єктів в кадрі. При цьому трекер інтерпретував зменшення розмірів об'єкта на зображенні як видалення від камери, і на підставі відносної зміни площі об'єкта і показників оптичного потоку обчислював напрямок його руху.

Таблиця 6

	Об'єктів знайдено, %	Груп об'єктів знайдено, %
Запис трафіку 1	70	96
Запис трафіку 2	72	100
Запис трафіку 3	75	75
Запис трафіку 4	74	75

## 8. Висновки

Результатом дослідження стала модернізація алгоритму розпізнавання. Запропоновано використання еківаріантного детектора на базі трансформуючого автоенкодера алгоритму розпізнавання. Створена та навчена нейронна мережа для розпізнавання об'єктів у відеопотоці. Проведено експеримент, у процесі якого ефективність розробленої системи порівнювалася з показниками альтернативних відомих методів розпізнавання. Точність розпізнавання у разі використання запропонованого методу зростає на 3-5%. Розроблена система розпізнавання більш стійка до локального шуму: для зображень, що піддаються розмиттю і оклюзії, падіння точності розпізнавання розробленої системи становить 3-6% проти 5-10% у альтернативних відомих методах.

Результати досліджень становлять практичний інтерес при проектуванні систем управління та обробки інформації в області комп'ютерного зору і розпізнавання зображень, для тих завдань, де існує необхідність визначення просторових параметрів зображених об'єктів. Запропонований підхід навчання системи базується на використанні відеозаписів, тобто система може навчатися на встановленому пункті моніторингу для адаптації до локальних образів. Для використання системи для моніторингу в режимі реального часу слід проаналізувати і спроектувати систему на базі розподілених обчислень для паралельного аналізу кадрів, адже результати локальних експериментів показують труднощі при багатьох кадрах в секунду, активація детекторів багатьох рівнів потребує значних обчислювальних ресурсів.

**Список літератури:** 1. *Nechiporenko A.S., Gubarenko E.V., Gubarenko M.S.* Authentication of users of mobile devices by their motor reactions. *Telecommunications and Radio Engineering*. 2019. V. 78 (11). P. 987-1003. doi: 10.1615 / TelecomRadEng.v78.i11.60. 2. *Ebrahim Karami, Siva Prasad, and Mohamed Shehata.* Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images. 2017. arXiv preprint arXiv:1710.0272. 3. *Локтев Д.А., Кочнев В.А., Локтев А.А.* Вивчення функцій розмиття зображення у вигляді інформативного параметра стану і поведінки аналізованого об'єкта. *Динаміка складних систем - XXI століття*. 2020. № 2. С. 16-27. 4. *Loktev D., Loktev A.* Image blurring function as an informative criterion. *Advances in Intelligent Systems and Computing*. 2021. V. 1258. P. 173-183. 5. *Фісенко В.Т., Фісенко Т.Ю.* Комп'ютерна обробка і розпізнавання зображень: навчальний посібник. СПб: СПбГУ ІТМО, 2008. 192 с. 6. *Huang G.B., Ramesh M., Berg T., Learned-Miller E.* Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts. 2007. No. 1 (2). pp. 3-37. 7. *Gourier N., Hall D., Crowley J.L.* Estimating face orientation from robust detection of salient facial structures. *FG Net Workshop on Visual Observation of Deictic Gestures*, 2004. V. 6(4). 8. *Gourier N., Hall D., Crowley J.L.* Caffe: Convolutional architecture for fast feature embedding. *FG Net Workshop on Visual Observation of Deictic Gestures*. 2004. P. 1-9. 9. *Singular Inversions.* FaceGen modeller (Version 3.3). *Singular Inversions*, 2008. 10. *Bishop C.M.* Neural networks for pattern recognition. Oxford: Oxford university press, 1995. P. 482. 11. *Bishop C.M.* Pattern recognition and machine learning. New York: Springer, 2006. P. 758. 12. *Tan X., Triggs B.* Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions. *IEEE Transactions on image processing*. 2010. Vol. 19, No 6. P. 1635-1650. 13. *Viola P., Jones M.* Rapid object detection using a boosted cascade of simple features. 2001. No. 1. P. 502-511. 14. *Szegedy et al.,* Going deeper with convolutions. 2015 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015. P. 1-9, doi: 10.1109/CVPR.2015.7298594. 15. *Hinton G.E.* A practical guide to training restricted Boltzmann machines. *Momentum*. 2010. No. 9 (1). P. 926. 16. *Hubel D.H., Wiesel T.N.* Brain and visual perception. ISBN13, 2005. 17. *Hinton G.E., Salakhutdinov R.R.* Reducing the dimensionality of data with neural networks. *Science*. 2006. No.313 (5786). P. 504-507. 18. *Duін R.P., Pekalska E.* Open issues in pattern recognition. *Computer Recognition Systems*. 2005. P. 27-42. 19. *Leo B.* Random forests. *Machine learning*. 2001. No. 45 (1). P. 5-32.

*Надійшла до редколегії 24.05.2021*

**Губаренко С.В.**, кандидат технічних наук, доцент, доцент кафедри системотехніки ХНУРЕ. Наукові інтереси: теорія прийняття рішень, управління соціально-економічними системами, системи штучного зору. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. +38 (050) 741 01 74.

**Губаренко М.С.**, асистент кафедри системотехніки ХНУРЕ. Наукові інтереси: згорткові нейронні мережі, глибоке навчання, розпізнавання зображень. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. +38 (050) 532 61 23.

**Антонюк М.В.**, магістрант кафедри системотехніки ХНУРЕ. Наукові інтереси: проблеми розпізнавання зображень. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. +38 (068) 342 43 74.

*П. Е. СИТНИКОВА, О. П. ГОВДЕРЧАК*

## **AWS STEP FUNCTIONS ЯК НЕВІД'ЄМНА СКЛАДОВА АРХІТЕКТУРИ СИСТЕМИ З ПОСЛІДОВНИМ ВИКОНАННЯМ БЕЗСЕРВЕРНИХ ФУНКЦІЙ**

У статті досліджено використання сервісу Step Functions хмарного провайдера Amazon Web Services (AWS) у системах, що мають послідовне виконання функцій/дій в алгоритмі свого виконання. Досліджено також сервіс AWS, що може бути використаний в будь-якій організації для будь-якої системи задля отримання стану виконання усього ланцюга функцій.

### **1. Вступ**

Створюючи безсерверні додатки, розробники, як правило, хочуть масштабувати програму відповідно до зростаючого навантаження, зберігаючи при цьому низькі витрати і дозволяючи декільком командам одночасно працювати над різними частинами програми. У безсерверній моделі однією з найкращих практик для досягнення цих цілей є розділення бізнес-логіки програми на набір нерозв'язаних служб. Велика безсерверна програма може складатися з десятків або сотень безсерверних служб [1].

Проблема виникає, коли така велика кількість служб потребує доступу до різних частин спільного стану. Для ефективної роботи цих служб команди також повинні мати можливість організувати потік даних через усі служби додатків в одному місці, і це саме те, що обробляє крокові функції AWS. Step Functions стали найважливішим елементом безсерверної екосистеми завдяки всьому стану та керуванню даними, необхідними для ефективної роботи безсерверних систем у масштабі.

Більше того, AWS Step Functions дозволяє організувати ваші завдання AWS за допомогою візуального редактора робочих процесів. Ви використовуєте цей редактор для створення діаграм стану машини або покрокових функцій, які потім стають основою побудови, спільного використання та модифікації поведінки додатків [2].

### **2. Постановка задачі**

На сьогоднішній день дуже багато додатків використовують хмарні обчислення для спрощення або покращення їхньої роботи, тому питання, чи використовувати безсерверні обчислення у даній роботі не розглядається, це взято за основу системи. Як провайдер сервісів хмарних обчислень розглядається Amazon Web Services (AWS).

У даній статті розглядатиметься проблема послідовного виклику безсерверних функцій (Serverless, FaaS - Function as a Service), які у даного представника хмарних послуг називаються AWS Lambda, та можливості вирішення цих проблем з допомогою сервісу AWS Step Functions.

Досліджуючи необхідність мікросервісних додатків запускати деякі частини логіки системи послідовно, розбиваючи їх на окремі сервіси, можна дійти до висновку, що найпоширенішим рішенням є використання синхронного виклику хмарних функцій як сервісів, у випадку AWS - Lambdas. Накопичений досвід експлуатації веб-служб Amazon (AWS) дозволяє зробити висновок про те, що багато організацій-користувачів AWS спочатку використовували AWS Lambda (продукт Amazon's Function-as-a-Service) для автоматизації робочих процесів навколо запущеного та масштабованого коду. Хоча таке рішення дозволило розробникам простих програм заощаджувати час, з підвищенням складності програм у компаній, які використовували AWS Lambda, виникали проблеми [3].

Ці проблеми об'єднувалися у дві основні групи: збільшення тривалості роботи та управління станом. Зокрема, під час виконання front-end сервісів AWS Lambda дозволяє виконувати функції не довше 15 хвилин. Після цього сервіс закінчується, залишаючи завдання незавершеними [4].

Очевидно, цей недолік є критичним для організацій, які використовують складні програми, що вимагають тривалих обчислень. Крім того, наслідком є неможливість повторного

виконання незавершених задач, оскільки функція, що виконувалася, заново викликатися не може. Кожен новий запуск цієї функції сприймається як окремий новий запуск із своїм особистим станом.

Для виходу з подібної ситуації можливо скористатися контейнерами за допомогою Amazon Elastic Container Service (ECS). Але у випадку, коли контейнери зайняті іншими завданнями, треба додати стан.

Ще один недолік полягає в тому, що AWS Lambda не створена для ефективного управління станами. Розробники повинні писати код у своїх програмах для зберігання цього самого стану, що ускладнює управління та збільшує час обробки. Отже, користувачі змушені вибирати між запуском вимогливих програм та забезпеченням достатнього стану для максимального використання [5].

З огляду на описану вище проблему, метою даної статті є дослідження, чи є AWS Step Functions самостійним вирішенням питання контексту та довготривалості виконання та чи варто використовувати цей підхід з точки зору ціноутворення.

### **3. Використання Step Functions архітектури системи з послідовним виконанням функцій**

Розглянемо, що є AWS Step Functions. AWS Step Functions - це послуга, що надається Amazon Web Services, яка спрощує організацію декількох служб AWS для виконання завдань. Step Functions дозволяють створювати кроки в процесі, де результат одного кроку стає входом для іншого кроку, і це все робиться за допомогою візуального редактора робочого процесу.

Step Functions забезпечують досить зручну функціональність: автоматичну обробку повторних спроб, активацію та відстеження для кожного кроку робочого процесу та забезпечення виконання кроків у правильному порядку. Хоча спочатку цей список може здатися не вражаючим, виявляється, зовсім не тривіально забезпечувати, щоб усі ці речі відбувалися правильно в робочих процесах, що містять десятки кроків і сотні паралельних виконань. Step Functions виконує багато функцій, які потребує робити програма. Іншими словами Step Functions є автоматом стану, його основні абстракції називаються станами. Конфігурація функцій кроку є картою всіх можливих кроків та переходів між ними. Стани та їх переходи визначаються з використанням мови штатів Амазонки. Ця мова базується на JSON і є власністю Amazon [6].

Крокові функції AWS є невід'ємною частиною безсерверної екосистеми завдяки наступному. Створюючи безсерверні додатки, розробники, як правило, хочуть масштабувати програму відповідно до зростаючого навантаження, зберігаючи при цьому низькі витрати і дозволяючи декільком командам одночасно працювати над різними частинами програми. У безсерверній моделі однією з найкращих практик для досягнення цих цілей є розділення бізнес-логіки програми на набір нерозв'язаних служб. Велика безсерверна програма може складатися з десятків або сотень безсерверних сервісів [7]. Проблема виникає, коли така велика кількість служб потребує доступу до різних частин спільного стану. Для ефективної роботи цих служб команди також повинні мати можливість організувати потік даних через усі сервіси додатків в одному місці, і саме цим займаються AWS Step Functions. Вони стали найважливішим елементом безсерверної екосистеми впродовж усього стану системи і управління даними, необхідними для ефективної роботи безсерверних систем у масштабі.

### **4. Переваги AWS Step Functions**

По-перше, Step Functions «полегшують життя» безсерверного розробника, дозволяючи швидко створювати складні послідовності задач в AWS, одночасно беручи на себе функцію обробки помилок і повторної спроби виконання логіки та дозволяючи відокремити бізнес-логіку додатка від логіки оркестрації. Нижче буде розглянуто особливості використання цієї технології, що може допомогти під час дизайну та розробки системи.

По-друге, завдяки цьому сервісу надається можливість швидкого створення складних послідовностей задач. Організація послідовності з десяти окремих безсерверних додатків, управління їх повторними спробами та усунення будь-яких помилок може бути надзвичайно складним завданням. Коли ви додаєте ще більше функцій, складність управління ними зростає в геометричній прогресії.

По-третє, завдяки графічному інтерфейсу та вбудованому механізму оперативного контролю Step Functions управляє послідовністю задач і знімає великий оперативний тягар з команди розробників.

Крім того є можливість керування станом між виконанням різних безсерверних та безстанових функцій. Для багатьох безсерверних робочих процесів налаштування черг та баз даних для зв'язку між усіма безсерверними службами може зайняти багато часу та спричинити помилки, не кажучи вже про те, що не всі випадки використання вимагають реальної бази даних або реальної системи черг.

Step Functions дозволяють також легко налаштувати управління станом на ранніх етапах, і це продовжує працювати добре з приростом масштабу завдань, коли використовуються багато сервісів разом.

Із використанням цієї технології логіка робочого процесу додатків стає від'ємною від бізнес-логіки. Тут є ще одна гарна практика безсерверної розробки. Додавання логіки робочого процесу та складних серверних конфігурацій до додатків, які повинні обробляти лише бізнес-логіку, збільшує складність програм і може легко створювати проблеми. Крім того, управління станом окремо від бізнес-логіки дозволяє розробникам зберігати ясність при роботі в безсерверній системі.

До плюсів можна також віднести ефективніші робочі процеси з паралельним виконанням. Коли є необхідність самостійно керувати станом різних функцій, отримати високу продуктивність від системи може бути складно. Деякі оркестровані фрагменти можуть обробляти лише одну задачу за раз, що може сповільнити роботу всієї програми. Step Functions дозволяють одночасно виконувати багато паралельних робочих процесів, переконавшись, що продуктивність масштабується залежно від завантаження програми [8].

Пропонується розглянути дві однакові системи, де у базі даних будуть 100 000 записів, запускатимуться декілька послідовних функцій, кожна з яких виконуватиме операцію фільтрації за своїм параметром. Різниця у тому, що в першій системі кожна функція викликати наступну за протоколами Amazon, без використання Step Functions (щільно зв'язані функції), а друга система буде запускати Step Functions, де функції будуть незалежними. Виконаємо 10 запитів та порівняємо результати (рис. 1).

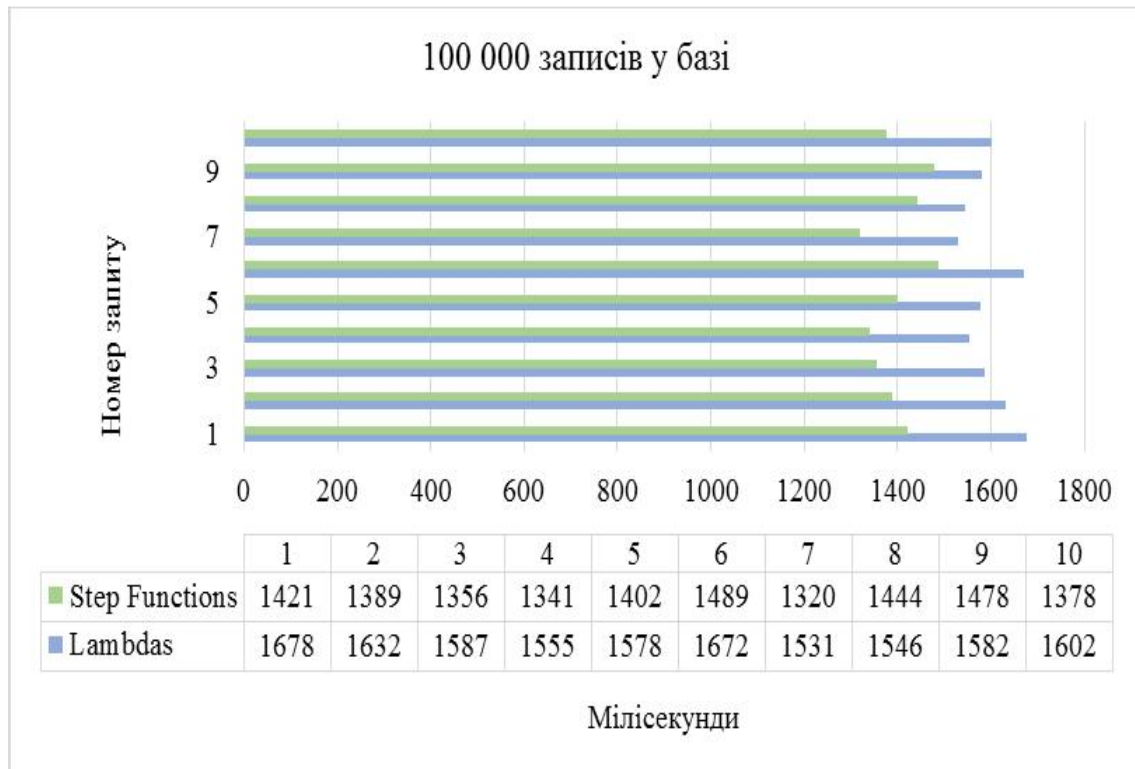


Рис 1. Порівняння підходів з/без використання Step Functions

З рисунку можна зробити висновок, що продуктивність системи з використанням Step Functions виявилась кращою за продуктивність системи без використання цього сервісу.

Функції кроку AWS можуть підвищити продуктивність роботи в кількох напрямках.

Перший напрям - обробка даних. Інші види обробки даних також добре підходять для функцій кроку AWS, особливо якщо джерело та адреса знаходяться всередині AWS. Функції кроку спрощують взаємозв'язок декількох кроків обробки даних, включаючи ті, що зроблені за допомогою функцій AWS Lambda, простої служби черги та простої служби сповіщень.

Другий напрям - організація робочого процесу без серверів. Основна цінність крокових функцій AWS полягає в простій організації програм, які вимагають взаємозв'язку множини безсерверних функцій. Для сукупності бізнес-процесів, для отримання кінцевого результату яких потрібна комбінація декількох роз'єднаних безсерверних програм, функція Step може бути правильним вибором для зручності організації.

## 5. Недоліки AWS Step Functions

Усі зазначені вище переваги можуть забезпечити високу цінність для багатьох команд та випадків використання, але перед тим, як взяти участь у запуску функцій Step у виробництві, слід також врахувати потенційні недоліки використання цієї програми у додатку. Хоча AWS Step Functions вирішують багато проблем для безсерверної спільноти, перехід усіх шарів оркестрації на Step Functions не обов'язково є правильним рішенням. Наступне зауваження треба враховувати при використанні покрокових функцій у виробництві.

Конфігурація з мовою станів Amazon є досить складною; його синтаксис заснований на JSON і тому оптимізований для машинної читабельності замість читабельності людьми. Вивчення мови може бути складним завданням, і, оскільки воно є власністю Amazon, набуті навички не застосовуватимуться ніде, крім контексту AWS Step Functions [9]. Ця мова є власністю Amazon і може використовуватися лише на Amazon Web Services. Тому при зміні постачальника хмарних послуг потрібно повторно впроваджувати рівень оркестровки або замінити його альтернативою іншого постачальника.

## 6. Механізм ціноутворення AWS Step Functions

Механізм ціноутворення є додатковою причиною переконатися у необхідності використання Step Functions в архітектурі системи, що має ланцюг послідовно виконуваних функцій.

Розглянемо основні характеристики цього механізму. Безкоштовний рівень AWS обмежений 4000 переходів стану функцій кроку AWS на місяць. Цей показник щомісячно поновлюється, тому можливо скористатися безкоштовним рівнем, навіть якщо обліковий запис AWS не є абсолютно новим.

За межею безкоштовного рівня використання Step Functions коштує 0,025 доларів США за 1000 переходів стану. Розглянемо роботу механізму ціноутворення в цьому випадку на такому прикладі. Нехай є робочий процес, який перетворює зображення для веб-сайту у чотири різних розміри, причому робочий процес складається із десяти переходів стану в середньому для кожного зображення і близько 10% виконання робочого циклу передбачає одну повторну спробу. Припустимо, що завантажуються 100 000 зображень на місяць. Кількість переходів робочого процесу буде складати 1 000 000 (10 переходів x 100 000 оброблюваних зображень). Додатково повторні спроби обробки дадуть 100 000 переходів (((100 000 оброблюваних зображень/100) x 10) x 10) переходів). Усього за місяць робочий процес потребує 1 100 000 переходів, 4 000 з яких будуть безкоштовними. У цьому випадку вартість використання функцій кроку AWS буде дорівнювати 27,4 \$ ((1 100 000 - 4 000) x 0,025 / 1 000).

Плата за AWS Step Functions доповнює будь-які збори за передачу даних та вартість будь-яких послуг AWS, якими ви користуєтесь у робочих процесах. Наприклад, якщо ці 100 000 виконання робочого процесу призводять до плати за AWS Lambda 600 \$ на місяць та до плати за передачу даних 100 \$ на місяць, загальна вартість за місяць складе 600 \$ + 100 \$ + 27,4 \$ (плата за використання крокових функцій) = 727,4 \$ на місяць для всієї системи.



## 7. Аналіз конкурентоспроможності AWS Step Functions

Розглянемо конкурентоспроможність AWS Step Functions відносно альтернатив від інших хмарних постачальників. Насамперед, слід зазначити, що Step Functions не мають прямого конкурента серед лідерів хмарних провайдерів таких, як Google (Google Cloud Platform, GCP) або Microsoft (Azure). Найближчим за походженням можна взяти Google Cloud Dataflows (GCP - Google Cloud Platform)

Розглянемо основні відмінності AWS Step Functions та Google Cloud Dataflow.

AWS Step Functions - це програмне забезпечення для створення розподілених додатків за допомогою візуальних робочих процесів. Ця технологія спрощує координацію компонентів розподілених програм та мікропослуг за допомогою візуальних робочих процесів. Створення програм з окремих компонентів, кожна з яких виконує дискретні функції, дозволяє швидко масштабувати та змінювати програми.

Google Cloud Dataflow є повністю керованою хмарною послугою та моделлю програмування для пакетної та потокової обробки великих даних. Google Cloud Dataflow також може використовуватися як керований сервіс для розробки та виконання широкого спектру шаблонів обробки даних, включаючи пакетні обчислення та безперервні обчислення. Cloud Dataflow забезпечує автоматичне управління ресурсами та оптимізацію продуктивності.

AWS Step Functions належить до категорії «Хмарне управління задачами» у технологічному стеці, тоді як Google Cloud Data Flow можна класифікувати як продукт, що належить до категорії «Обробка даних у реальному часі».

SWOT-таблицю аналізу цих продуктів наведено у табл. 1.

Таблиця 1 -  
SWOT AWS Step Functions порівняно з конкурентами (GCP Dataflows)

Ключові фактори	Краще за конкурентів	Гірше за конкурентів
Графічне відображення виконання алгоритму програми	✓	
Конфігурування послідовних або паралельних кроків алгоритму програми	✓	
Контроль переходу стану між різними сервісами	✓	
Легкість конфігурування		✓
Швидка обробка великих об'ємів даних		✓
Рівень популярності серед розробників	✓	
Рівень ціноутворення	✓	
Рівень популярності бренду		✓
Підтримка інтеграції з різними сервісами того ж самого хмарного провайдера	✓	

Переваги використання саме AWS Step Functions стають очевидними, тому що наразі немає конкурентного сервісу від інших постачальників, що може графічно відображувати інтеграції з різними сервісами, контролювати стан переходу між ними та їх вхідні параметри.

Стосовно найближчого конкурента, GCP Dataflows, слід зазначити: це сервіс, створений більше для моніторингу обробки великого набору даних, автоматичного збільшення пам'яті для їх обробки, але не для конфігурації послідовного чи паралельного виконання різних сервісів Google.

За іншими критеріями - популярність у всьому світу та легкість конфігурування - AWS Step Functions виявилася гіршою, ніж система Google. Щодо популярності, звісно, Амазон не може сперечатися з Google, але не у світі розробників, де AWS існує вже давно та досі є безперечним лідером з постачання сервісів для хмарних обчислень. Щодо легкості конфігурування, AWS Step Functions конфігурується спеціальною мовою, заснованою на JSON шаблоні, тому розробникам потрібно буде спочатку навчитися її використовувати.

Слід також пам'ятати про ціноутворення, а саме, про те, що AWS надає 4000 безкоштовних виконань цього сервісу на місяць. Це означає, що якщо система буде використовуватися нечасто, то кошти не потрібно буде сплачувати взагалі. Також слід зазначити, що неважливо, скільки сервісів було викликано під час одного виконання автомату стану Амазону, це все буде вважатися одним виконанням Step Functions, і кошти підуть окремо за сервіси та окремо за одне виконання Step Functions.

## 8. Висновки

У даній роботі було розглянуто проблему використання безсерверних функцій, а саме, відсутність стану використання системи, відсутність механізму повторних запитів до ланцюга виконуваних послідовно функцій у випадку некоректної праці будь-якого елемента, та сильну зв'язаність між усіма послідовними функціями у вигляді прямого запиту кожної з них до наступної.

Також було досліджено особливості AWS Step Functions як окремого сервісу та шляхи вирішення усіх проблем архітектури системи з послідовним виконанням безсерверних функцій. Для цього пропонується знову звернути увагу на визначення Step Function та на проблему, що розглядалася.

Step Functions слід належним чином описувати як «стан як послуга». Без нього неможливо було б підтримувати стан кожного виконання, що має множину лямбда-функцій/дій.

Технологія мікросервісів передбачає, що кожен сервіс - це незалежна частина системи, яка може бути навіть написана іншою мовою програмування. Тому коли потрібне послідовне виконання декількох функцій як сервісів, у випадку AWS - Lambdas, то впровадження Step Functions - необхідний крок для завдання порядку виконання, який залишає сервіси незалежними. Без цієї технології доведеться на кожному кроку, у кожній функції задавати ім'я наступної, тобто сервіси будуть знати кожен про кожного, що означає - тісне зв'язування і порушення концепту мікросервісної архітектури.

Як один з недоліків Step Functions було визначено, що їх доволі складно конфігурувати, тому що потрібно знати нову мову AWS - Amazon State Language. Складно також уявити конфігурацію функцій, що послідовно викликаються та знають одна про одну, як вже було сказано, для того, щоб виключити або просто змінити один з кроків, щоб після 3-го кроку, наприклад, викликався 6-й, а потім 4-й. Функція, викликаючи наступну у своєму ж коді, тісно зв'язана із цією наступною функцією. Тому зміна порядку викликів або виключення одного з етапів виконання призведе до зміни коду майже кожної функції окремо у той час, коли, опанувавши ASL та сконфігурувавши Step Functions, можна з легкістю маніпулювати порядком або робити щось інше, не торкаючись окремих функцій.

AWS Step Functions вирішують питання контексту між усіма послідовними функціями, що, безперечно, є їхнім плюсом, адже концепт Lambda як раз говорить, що це функція безконтекстна та швидка. Тому вигадування логіки передачі його без Step Functions суперечить самому принципу безсерверних функцій та робить логіку виконання та підтримки складнішою та повільнішою. Як говорить один з принципів розробки SOLID, літера S - Single Responsibility, що означає «єдиний обов'язок». Тобто, якщо лямбди не вміють працювати з контекстом і за своїм принципом не повинні, - не треба їх змушувати, нехай про це подбає той сервіс, що був створений для цього, - AWS Step Functions.

І нарешті ціноутворення. Як вже було сказано вище, Step Functions - не дуже дорогий сервіс. Якщо компанія дозволяє собі систему, яка побудована на мікросервісній архітектурі і використовує AWS, то у випадку послідовного виконання функцій необхідно замислитись над використанням технології автомату стану Амазона. Адже, навіть якщо сто або більше функцій виконується послідовно, коли вони сконфігуровані у Step Functions - це все одно лише одне виконання автомату, а тисяча таких виконань коштує лише 0,025 \$.

Тому можна дійти до висновку, що саме цей сервіс є невід'ємною складовою мікросервісної архітектури системи із послідовним виконанням функцій. Step Functions - це відносно новий продукт AWS, який, безсумнівно, змінить показники продуктивності, дозволивши розбити програми на основні компоненти служби з можливістю маніпулювати кожним із цих компонентів окремо.

**Список літератури:** 1. *Ньюмен С.* Создание микросервисов. СПб.: Питер, 2018. 304 с. 2. *AWS Step Functions: Developer Guide.* Режим доступу: <https://www.amazon.com/AWS-Step-Functions-Developer-Guide-ebook/dp/B078XBSLY53> 3. *Fowler M.* Microservices. Режим доступу: <https://martinfowler.com/articles/microservices.html> 4. *Witting A., Witting M.* Amazon Web Services in Action. Shelter Island, 2015. 200p. 5. *Офіційний сайт C++ Micro Services.* - Режим доступу: <http://cppmicroservices.org/> 6. *Boricha V, Rajani M., Atanna A.* Learn AWS Serverless Computing. Packt Publishing, 2019. 174 p. 7. *Richardson, C.* Pattern: Microservice Architecture // Kong. - Access mode: <http://microservices.io/patterns/microservices.html> 8. *AWS Step Functions.* Режим доступу: <https://aws.amazon.com/ru/step-functions/?step-functions.sort-by=item.additionalFields.postDateTime&step-functions.sort-order=desc> 9. *Офіційний сайт Google:* Compare AWS and Azure services to Google Cloud. Режим доступу: <https://cloud.google.com/free/docs/aws-azure-gcp-service-comparison>

*Надійшла до редколегії 06.05.2021*

**Ситнікова Поліна Едуардівна**, кандидат технічних наук, доцент, доцент кафедри системотехніки ХНУРЕ. Наукові інтереси: Data Mining and Knowledge Discovery, логічне моделювання даних, прийняття рішень в умовах невизначеності. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. (057) 702 10 06; e-mail: [polina.sytnikova@nure.ua](mailto:polina.sytnikova@nure.ua).

**Говдерчак Олексій Павлович**, студент групи СПРМ-19-2 ХНУРЕ, магістрант. Наукові інтереси: розробка програмного забезпечення, розробка архітектури систем. Адреса: Україна, м. Харків, 61166, пр. Науки, 14, тел. (057) 702 14 46; e-mail: [oleksii.hovderchak@nure.ua](mailto:oleksii.hovderchak@nure.ua).

---

УДК 004.048:004.89

DOI: 10.30837/0135-1710.2021.177.035

*А.П. САФОНИК, І.М. ГРИЦЮК, М.М. МІЩАНЧУК, І.В. ІЛЬКІВ*

## **ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ВИЗНАЧЕННЯ ЗАЛІЗА В КОАГУЛЯНТІ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ**

Розглядаються питання побудови інтелектуальної системи для визначення концентрації заліза в коагулянті за його кольором на базі нейронної мережі. На основі проведеного аналізу різних типів нейронних мереж обрано найбільш відповідну архітектуру нейронної мережі для вирішення задачі визначення концентрації заліза в коагулянті. Описано процес проектування архітектури, дослідження методів навчання, підготовки даних для проведення навчання нейронної мережі для визначення концентрації заліза в коагулянті за його кольором. Розроблено структурно-функціональну схему нейронної мережі, яка складається із вхідного, прихованих та вихідного шарів, описано активаційні функції. Проаналізовано точність навчання нейронної мережі шляхом порівняння даних, отриманих з використанням різних оптимізаторів, із використанням бібліотеки TensorFlow. Розроблений веб-додаток може бути використаний в якості складової інформаційно-аналітичної системи автоматизованого керування технологічним процесом електрокоагуляційної очистки.

### **1. Вступ**

Очищення стічних вод - це одна з проблем сучасного людства. Останнім часом для очищення води набувають популярності невеликі очисні споруди. Оснащення водоочисних споруд такими системами особливо важливе для сільського господарства, легкої промисловості тощо. Але використання реагентів зумовлює створення служб забезпечення їх доставки та зберігання, що є накладним в порівнянні з електрохімічними методами, які, в свою чергу, дають змогу добувати реагенти на місці з наявної сировини. Одним із найбільш перспективних методів, які надають таку можливість, є метод електрокоагуляції. На даний час все більше уваги приділяється дослідженням цього процесу за допомогою математичних моделей, що допомагає поліпшити конструктивні особливості пристрою та зменшити витрати в експлуатації, а також прогнозувати ефективність процесу в широкому діапазоні операцій з

мінімальними виробничими витратами. Дані дослідження також є основою для розробки автоматизованих систем керування процесами електрокоагуляції. Процес отримання коагулянту шляхом електрокоагуляції включає складні та дорогі натурні експерименти, що дозволяють визначати вміст корисного елементу (заліза) в коагулянті. Одним з лабораторних методів знаходження заліза в коагулянті є фотоколориметричний метод, який полягає у кількісному визначенні концентрації речовини за допомогою поглинання світла у видимій і ближній ультрафіолетовій області спектра. Даний метод дозволяє точніше і швидше оцінювати якість багатокомпонентних сполук. Крім того, даний метод включають до стандартизованих методів визначення вмісту заліза у воді, оскільки інтенсивність забарвлення води та оптична щільність середовища змінюються зі зміною концентрації заліза. На даний момент практично не існує датчиків для визначення концентрації заліза в коагулянті в режимі реального часу, тому, актуальною є задача розробки автоматизованої інформаційної системи для електрохімічного виробництва коагулянту на основі фотоколориметричного аналізу.

Аналіз публікацій за даною тематикою показав, що ряд досліджень проводилися з використанням фотоколориметрії та фотоколориметричного аналізу. В роботі [1] розроблено цифровий колориметр із мобільним додатком на базі iOS з використанням теорії Евкліда для визначення вмісту. Цей додаток виявляє кольорові компоненти, запам'ятовуючи дані щодо червоного, зеленого та синього кольорів, а також розраховуючи значення відтінку, насиченості, яскравості за допомогою стандартної теорії кольорів. Простий і досить ефективний метод визначення концентрації заліза у зразках води згідно з правилами United States Environmental Protection Agency висвітлюється в роботі [2]. В дослідженні [3] вміст заліза у воді визначали за допомогою цифрової колориметрії зображення з використанням веб-камери. Для отримання рівняння, що описує залежність вмісту заліза від кольору, на основі відтінків червоного, зеленого та синього, а також їх насиченості та прозорості був використаний метод найменших квадратів. Два методи визначення концентрації хрому (Cr) та заліза (Fe) описані в [4]. Всі ці дослідження базуються на випробуваннях, спрямованих на визначення концентрації в лабораторії, які є дуже дорогими і трудомісткими. У роботі [5] було розроблено портативний пристрій, який дозволяє проводити колориметричний аналіз на місці та пропонує широку доступність з обмеженими ресурсами. У роботі [6] описується розробка мобільного інструменту колориметричного аналізу - PhotoMetrix, який використовує прості методи лінійної регресії для одновимірного аналізу та аналізу основних компонентів для багатовимірного аналізу досліджень. Ці зображення фіксуються основною камерою пристрою та перетворюються на червоні, зелені та сині гістограми.

## **2. Постановка задачі**

З огляду на те, що при зміні концентрації заліза у воді змінюється колір та інтенсивність забарвлення розчину, розроблену нами експериментальну установку для лабораторних досліджень було вирішено доповнити автоматизованою інформаційною системою, що забезпечить визначення концентрації заліза у воді за зміною кольору та інтенсивності світлового потоку в реальному часі. З цією метою було вирішено розробити штучну нейронну мережу для визначення концентрації заліза в коагулянті. Для розробки такої мережі необхідно вирішити ряд задач, а саме: провести аналіз різних типів нейронних мереж та обрати найбільш відповідну архітектуру нейронної мережі для вирішення задачі визначення концентрації заліза в коагулянті; описати процес проектування архітектури, дослідження методів навчання, підготовки даних для проведення навчання нейронної мережі; розробити структурно-функціональну схему нейронної мережі; проаналізувати точність навчання нейронної мережі шляхом порівняння даних отриманих з використанням різних оптимізаторів; розробити веб-додаток, який надасть можливість використання розробленої нейронної мережі як складової системи автоматизованого керування технологічним процесом електрокоагуляційної очистки.

## **3. Вирішення поставленої задачі**

Для вирішення поставленої задачі було розроблено програмне забезпечення для мікрокомп'ютера Raspberry Pi 4 та під'єданого датчика кольору TCS230.

На сьогодні існує велика кількість різних типів нейронних мереж, а також варіантів реалізації мереж одного і того ж типу. Серед різних видів нейронних мереж найрозповсюдженішими є: багат шаровий перцептрон; згортоква нейронна мережа; рекурентна нейронна мережа.

Для вибору типу мережі, що буде використана для вирішення задачі визначення заліза в коагулянті, розглянемо детальніше наведені типи мереж.

Багатошаровий перцептрон (MLP) - це клас штучної мережі, який має мінімум три шари. Перший - вхідний шар з нейронами від  $I_1$  до  $I_n$ , другий - прихований шар з нейронами від  $H_1$  до  $H_m$  та третій - вихідний шар з нейронами від  $O_1$  до  $O_k$ . Ці шари з'єднані таким чином, що кожен нейрон попереднього шару з'єднаний з кожним нейроном наступного [7] (див рис. 1).

Прихованих шарів може бути більше ніж один, а мережі, що містять декілька прихованих шарів, називають «глибокими» нейронними мережами. Нейронні мережі MLP часто використовуються для вирішення проблем класифікації та прогнозування [8].

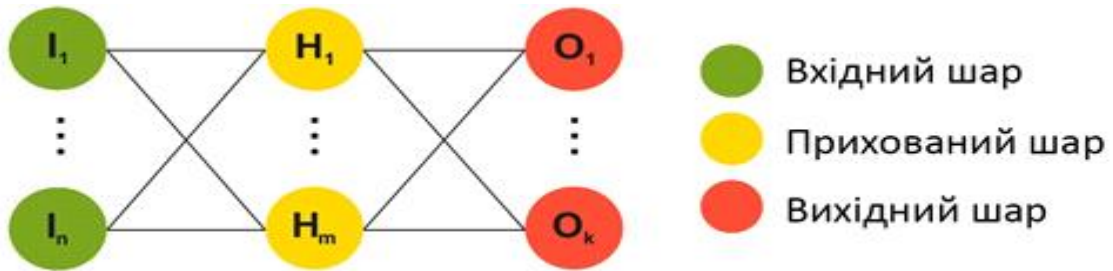


Рис. 1. Багатошаровий перцептрон

Згорткові нейронні мережі - це клас глибоких нейронних мереж, які широко використовуються для аналізу візуальних зображень. Такі нейронні мережі використовують математичні операції згортки. Дані нейронні мережі мають таку ж структуру, як і MLP мережі.

Рекурентні нейронні мережі - це клас мереж, у яких з'єднання між вузлами утворюють орієнтований у часі граф. До таких мереж можна віднести мережі довгострокової/короткочасної пам'яті [9] (LSTM). Структура такої мережі зображена на рис. 2, де нейрони від  $I_1$  до  $I_n$  - це вхідні нейрони, від  $H_1$  до  $H_m$  - це нейрони прихованого шару із пам'яттю та можливістю повторення, від  $O_1$  до  $O_k$  - вихідні. Такі нейронні мережі часто використовують для аналізу послідовності слів, розпізнавання мови або рукописного вводу. Дані нейронні мережі погано працюють із даними, які представлені у вигляді таблиць.

Задачу визначення концентрації заліза в коагулянті за його кольором можна віднести до класу задач розпізнавання образів на основі класифікації вхідних даних. Найпоширенішим типом нейронних мереж для такого класу задач є багатошарові перцептрони.



Рис. 2. Нейронна мережа LSTM

Розроблена нейронна мережа має вхідний та вихідний шар, між якими розміщено два прихованих шари (див. рис. 3). Вхідний шар має п'ять нейронів, в якості функції активації використано гіперболічний тангенс. В двох прихованих шарах - по 10 нейронів з функціями активації SeLU та ReLU відповідно. У вихідному шарі 1 нейрон, як функцію активації використано експоненційну функцію.

#### 4. Запропонована методологія

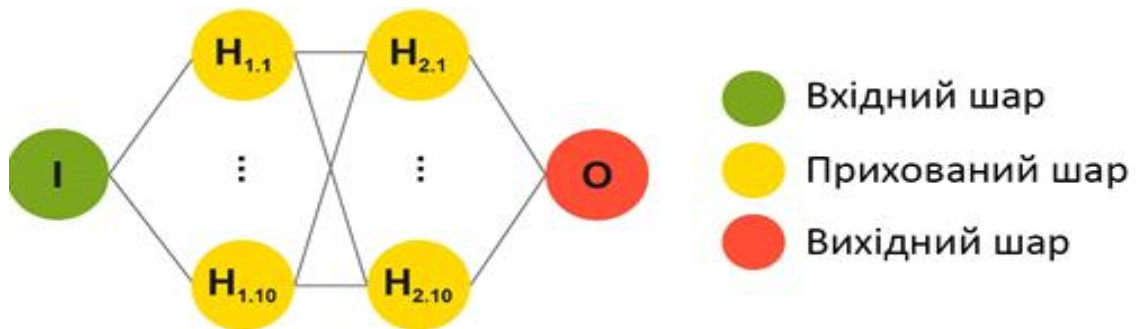


Рис. 3. Схема розробленої нейронної мережі

Для вибору типу мережі, яка буде використана для вирішення проблеми визначення заліза в коагулянті, розглянемо типи мереж більш детально.

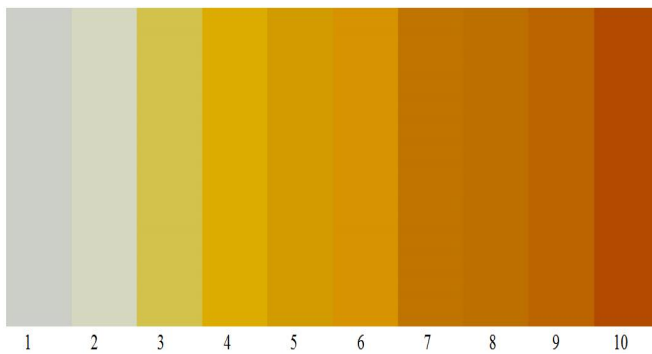


Рис. 4. Зміна кольору речовини з часом при зміні напруги

Для формування навчального та випробувального набору даних було проведено лабораторне дослідження різних зразків коагулянта, де визначали концентрацію заліза в коагулянті та параметри RGB для кожного з цих зразків (див. рис. 4).

В якості вхідних даних для нейронної мережі використовується параметр hue (одна з трьох основних характеристик кольору разом з насиченістю і яскравістю) з колірної моделі HSL. Вихідним результатом роботи нейронної мережі є кон-

центрація заліза в коагулянті.

В таблиці 1 наведені результати експериментального дослідження процесу, а саме: визначення концентрації загального та трьохвалентного заліза, сили струму та колірності речовини в різні моменти часу та зі зміною напруги.

Таблиця 1 -

Експериментальні дані

№	Час (хв)	Концентрація (мг / дц <sup>3</sup> )	Червоний	Зелений	Синій
1	6	0.8	204	207	200
2	12	1.1	214	215	193
3	18	3.7	211	194	75
4	24	4.5	220	173	54
5	30	6.3	210	155	49
6	36	6.9	215	147	41
7	42	9.1	193	116	38
8	48	9.6	189	111	33
9	54	11.8	188	100	27
10	60	12.6	178	74	11

Параметри RGB було конвертовано в параметри HSL, для визначення концентрації було вибрано параметр Hue.

RGB колірний простір переводиться в HSL з використанням наступних співвідношень:

$$H = \begin{cases} \text{undefined,} & \text{if } MAX = MIN \\ 60^\circ \times \frac{G - B}{MAX - MIN} + 0^\circ, & \text{if } MAX = R \text{ and } G \geq B \\ 60^\circ \times \frac{G - B}{MAX - MIN} + 360^\circ, & \text{if } MAX = R \text{ and } G < B \\ 60^\circ \times \frac{G - B}{MAX - MIN} + 120^\circ, & \text{if } MAX = G \\ 60^\circ \times \frac{G - B}{MAX - MIN} + 240^\circ, & \text{if } MAX = B \end{cases}, \quad (1)$$

$$S = \frac{MAX - MIN}{1 - |1 - (MAX + MIN)|}, \quad (2)$$

$$L = \frac{1}{2}(MAX + MIN), \quad (3)$$

де  $R, G, B$  - позначення кольору в колірній моделі RGB;  $MAX$  та  $MIN$  - максимальне та мінімальне значення параметрів  $R, G, B$ ;  $H$  - тон;  $S$  - насиченість;  $L$  - яскравість.

Далі було виконано апроксимацію експериментальних даних для визначення залежності значень  $Hue$  від часу та концентрації заліза в коагулянті від часу. На основі отриманих виразів було згенеровано 3600 точок  $Hue$  та концентрації заліза (див рис. 5. а, б).

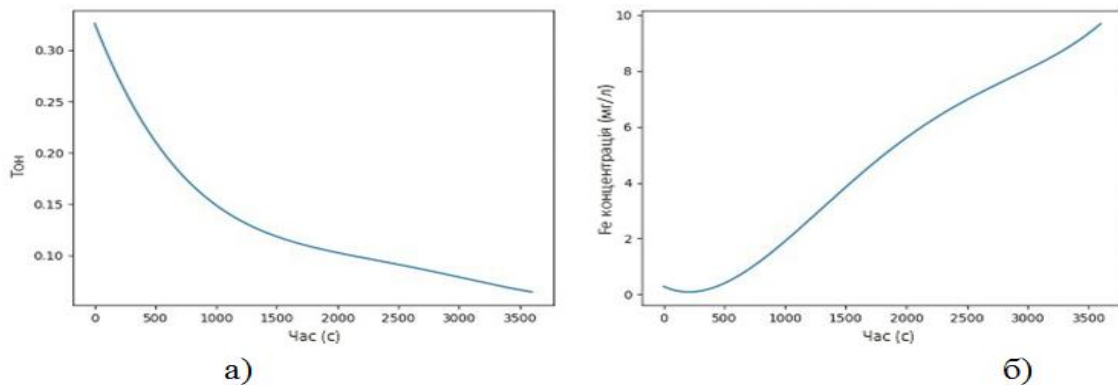


Рис. 5. Зміна а) значень  $Hue$  з часом; б) концентрації заліза з часом

З отриманих точок вибрано 3240 точок для навчання та 360 точок для тестування нейронної мережі. В таблиці 2 наведено вхідні та вихідні значення параметрів нейронної мережі.

Для навчання нейронної мережі в бібліотеці TensorFlow використовують функції-оптимізатори, які покращують навчання шляхом оновлення знань у мережі. Для того, щоб обрати найкращий оптимізатор для нейронної мережі, було розроблено та проаналізовано декілька нейронних мереж з різними оптимізаторами. Далі кожну нейронну мережу було натреновано за допомогою згенерованого датасету.

Було досліджено 8 оптимізаторів, доступних в бібліотеці TensorFlow. Як видно з таблиці 3, такі оптимізатори, як Ftrl та Adadelata, не впливають на навчання нейронної мережі. Інші ж

Таблиця 2 - Приклад вхідних та вихідних значень параметрів нейронної мережі

№	Hue	Концентрація
1	0.2110	0.4009
2	0.1969	0.7229
3	0.1073	4.8701
4	0.0649	9.5388

Дослідження оптимізаторів

Оптимізатор	SGD	RMSprop	Adam	Adadelata	Adagrad	Adamax	Nadam	Ftrl
Похибка	6.91	8.28	9.33	373.22	183.92	9.05	10.84	370.43

Таблиця 3 -

оптимізатори, наприклад, SGD та RMSprop, навпаки, навчають розроблену нейронну мережу. Найкраще себе показав оптимізатор SGD з похибкою 6.91 % на тестовому датасеті. Ця похибка обчислюється як середньоквадратичне відхилення результату роботи нейронної мережі від апроксимованих даних.

Як видно з рис. 6, оптимізатор SGD досягнув такого результату за 20 епох (ітерацій) навчання.

Проаналізувавши швидкість навчання нейронної мережі, для використання у програмному забезпеченні для визначення концентрації заліза в коагулянті було обрано нейронну мережу, навчену з допомогою оптимізатора SGD.

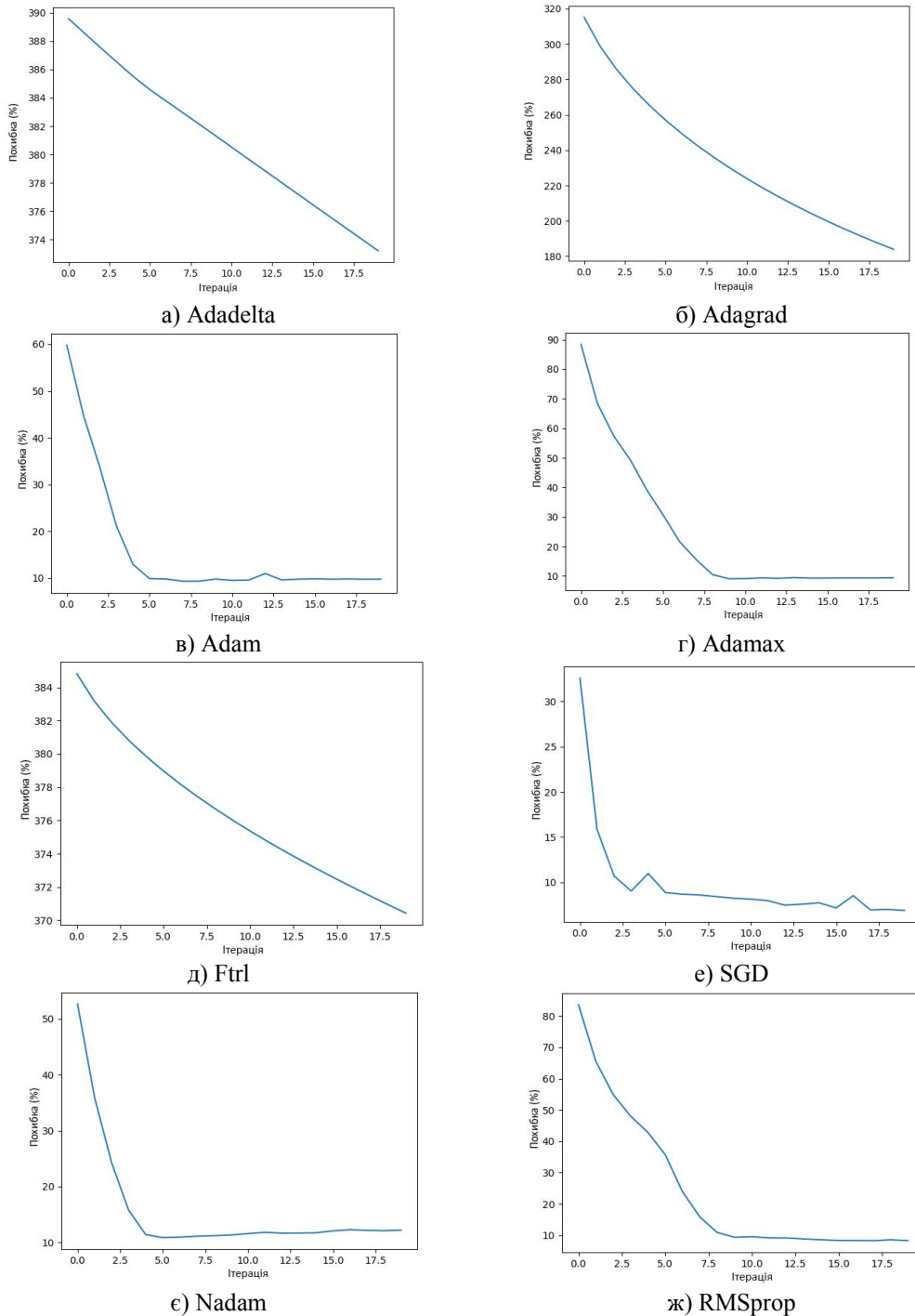


Рис. 6. Швидкість навчання нейронної мережі при різних оптимізаторах



## 5. Алгоритм роботи програми

Для зручного доступу до результатів вимірювання датчика та значень концентрації, визначених з допомогою нейронної мережі, було розроблено веб-додаток, в якості сервера для якого використано мікрокомп'ютер Raspberry Pi 4. Під'єднавши даний комп'ютер до мережі, за мережевою адресою даного пристрою можна потрапити на веб-панель даного пристрою (див. рис. 7).

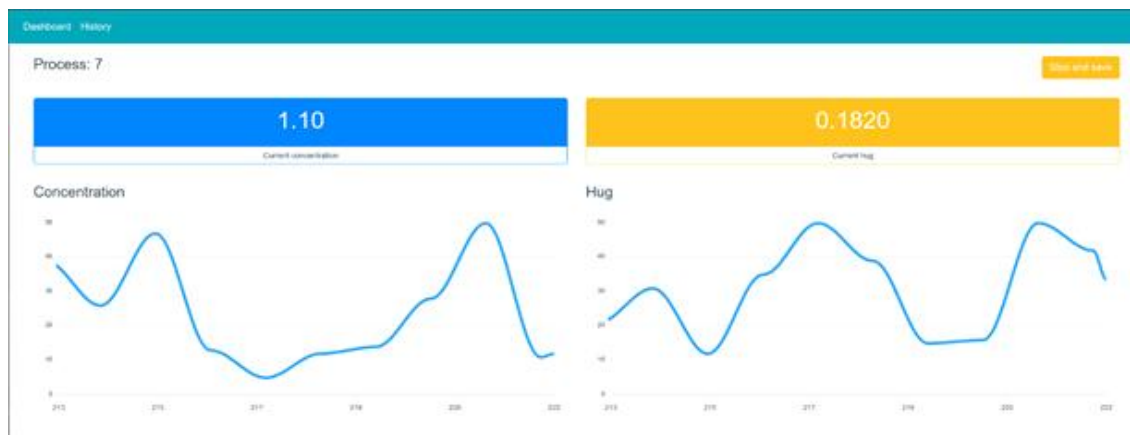


Рис. 7. Інтерфейс програми

З допомогою веб-інтерфейсу можна запустити процес вимірювання та побачити історію, а також поточні значення вимірювання.

Для розробки веб-інтерфейсу було використано JavaScript фреймворк Vue.js [10]. Основними критерієм вибору стали простота у використанні та значно менший розмір порівняно з аналогічними React.js та Angular. Для відображення даних користувачеві Vue.js використовує менеджер станів.

В ролі такого менеджера було використано Vuex. Стани в такому менеджері змінюються за допомогою мутацій, які, в свою чергу, викликаються з подій. Такий механізм дає можливість розділити код, який змінює дані та отримує їх із сервера.

Для реалізації комунікації із сервером (див. рис. 8.) використовуються дві бібліотеки: бібліотека axios для виконання асинхронних HTTP запитів та бібліотека socket.io для роботи із WebSocket. Серверна частина опитує датчик кольору за допомогою інтерфейсу введення-виведення General-purpose input/output. Потім дані з датчика обробляються нейронною мережею, за допомогою WebSockets відправляються на веб-панель та зберігаються в базі даних. В свою чергу, за допомогою HTTP протоколу отримуються дані з історії вимірювань та здійснюється керування процесами вимірювання.

Серверна частина програми відповідає за опитування датчика кольору, визначення концентрації заліза на основі отриманого кольору, обробку команд користувача, запис отриманих та розрахованих даних в базу даних.

Для розробки серверної частини було обрано Python-фреймворк Django [11].

З допомогою даного фреймворку було розроблено сервіс для визначення та збереження даних.

На його основі було також створено WebSocket-сервер, який дає можливість оновлювати дані на веб-панелі в реальному часі.

Розглянемо детальніше роботу серверної частини програми. Після натискання користувачем кнопки «Run» на веб-панелі на веб-сервер відправляється запит, який обробляється Django.

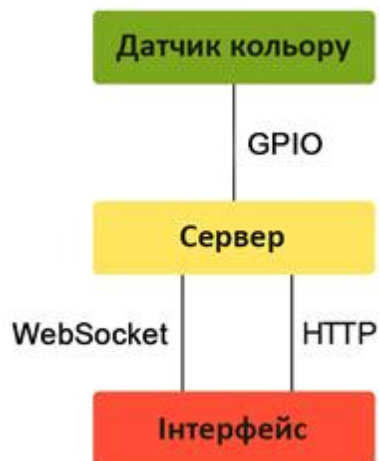


Рис. 8. Схема комунікації в системі

Під час цього відбувається валідація даних, які прийшли в запиті, перевірка наявності вже виконуваного процесу даного типу. Далі за умови відсутності помилок у даних та відсутності вже виконуваних процесів даного типу створюється запис в базі про новий запущений процес та відбувається запуск фоновому процесу, який опитує датчик та обробляє отримані дані.

Після запуску фоновому процесу відбувається його ініціалізація. Під час цього процес під'єднується до каналу WebSocket-сервера для передачі даних на веб-панель. Цей самий канал буде використаний для зупинки даного процесу після натиснення користувачем кнопки «Stop» на веб-панелі.



Рис. 9. Схема процесу

Після ініціалізації (див. рис. 9) запускається безкінечний цикл, за одну ітерацію якого з інтервалами в 1 секунду відбуваються такі дії: опитування датчика кольору; перетворення отриманих з датчика RGB параметрів кольору в параметри HSL; передача параметра hue на вхід нейронної мережі, де на його основі отримується значення концентрації заліза в коагулянті; збереження в базі даних та відправка на веб-панель з допомогою WebSocket-сервера значення параметрів з датчика кольору, конвертованих параметрів кольору та визначеної концентрації.

Цикл завершується після зупинки його користувачем, обробки сервером відповідного запиту, відправки повідомлення про завершення роботи у фоновий процес за допомогою WebSocket і оновлення запису в історії процесів після завершення його роботи.

## 6. Висновок

Розглянуто питання створення та застосування нейронних мереж для визначення концентрації заліза в коагулянті за його кольором. Було розроблено програмне забезпечення для визначення концентрації заліза в коагулянті у вигляді веб-додатку для відображення даних вимірювання в реальному часі та збереження історії вимірювань. Було розглянуто різні типи нейронних мереж, серед них було обрано нейронну мережу, яка найкраще підходить для вирішення задачі визначення концентрації заліза за кольором коагулянта. На основі аналізу оптимізаторів бібліотеки tenthoflow обрано оптимізатор SGD з похибкою 6.91 % на тестовому датасеті. Розроблено автоматизовану інформаційну систему визначення залізовмісного коагулянту на основі фотоколориметричного аналізу. Система містить комірку, через яку зі сталюю вит-

ратою пропускається досліджувана рідина, а також блок оброблення та збереження інформації, що дозволяє знизити участь людини в процесі вимірювань і забезпечити безперервність процесу вимірювань через відсутність потреби у відборі проби досліджуваного матеріалу, а також зменшити затратність процесу вимірювань.

**Список літератури:** 1. Masawat P., Harfield A., Srihirun N., Namwong A. Green Determination of Total Iron in Water by Digital Image Colorimetry // Analytical Letters. Volume 50, Issue 1. 2017. P. 173-185. 2. Sreenivasareddy Annem. Determination of Iron Content in Water // OPUS Open Portal to University Scholarship, Governors State University. Summer. 2017. P. 1-19. 3. Juan A. V. A. Barros, Fagner Moreira de Oliveira, Guilherme de O. Santos, Celio

Wisniewski, Pedro Orival Luccas. Digital Image Analysis for the Colorimetric Determination of Aluminum, Total Iron, Nitrite and Soluble Phosphorus in Waters // Analytical Letters. Volume 50. Issue 2. 2017. P. 414-430. 4. Firdaus M. L., Trinoveldi W. Abwi, F., Rahayu I., Rahmidar L., Warsito K. Determination of Chromium and Iron Using Digital Image-based Colorimetry // Procedia Environmental Sciences. Volume 20. 2014. P. 298 - 304. 5. Luka G. S., Nowak E., Kawchuk J., Hoorfar M., Najjaran H. Portable device for the detection of colorimetric assays // Royal Society Open Science. Volume 4. Issue 11. 2017. 171025. 6. Helfer G. A., Magnus V. S., Bock F. C., Teichmann A., Ferrao M. F., A. B. da Costa. PhotoMetrix: An Application for Univariate Calibration and Principal Components Analysis Using Colorimetry on Mobile Devices // Journal of the Brazilian Chemical Society. Volume 28. Issue 2. 2017. P. 328-335. 7. Multilayer perceptron: Theory and Applications / ed.: R. Vang-Mata. New York. Nova Science Publishers, 2020. 153 p. 8. Brownlee J. When to Use MLP, CNN, and RNN Neural Networks. 2019. URL: <https://machinelearningmastery.com/when-to-use-mlp-cnn-and-rnn-neural-networks>. 9. Bernico M. Deep Learning Quick Reference: Useful hacks for training and optimization deep neural networks with TensorFlow and Keras. Birmingham, UK: Packt Publishing, 2018. 10. Hanchett E., Listwon B. Vue.js in Action. Manning Publications, 2018. 304 c. 11. Holovaty A., Kaplan-Moss J. The Definitive Guide to Django. Web Development Done Right. NY, Apress, 2009. 536 p.

Надійшла до редколегії 21.05.2021

**Сафоник Андрій Петрович**, доктор технічних наук, професор, професор кафедри автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій Національного університету водного господарства та природокористування. Наукові інтереси: математичне та комп'ютерне моделювання природних, технологічних, техногенних процесів і систем та їх автоматизація. Адреса: Україна, 33018, м. Рівне, вул. Соборна 11, тел. +38 (050) 378 86 21.

**Грицюк Іванна Михайлівна**, асистент, асистент кафедри комп'ютерних наук та прикладної математики Національного університету водного господарства та природокористування. Наукові інтереси: математичне та комп'ютерне моделювання природних, технологічних, техногенних процесів і систем та їх автоматизація. Адреса: Україна, 33018, м. Рівне, вул. Соборна 11, тел. +38 (063) 345 46 11.

**Мищанчук Максим Миколайович**, студент. Наукові інтереси: математичне та комп'ютерне моделювання природних, технологічних, техногенних процесів і систем та їх автоматизація. Адреса: Україна, 33018, м. Рівне, вул. Соборна 11, тел. +38 (098) 083 90 80.

**Льків Ігор Васильович**, аспірант кафедри вищої математики Рівненського державного гуманітарного університету. Наукові інтереси: математичне та комп'ютерне моделювання природних, технологічних, техногенних процесів і систем. Адреса: Україна, 33017, м. Рівне, вул. Степана Бандери 12, тел. +38 (093) 487 26 89.

---

УДК 004.942

DOI: 10.30837/0135-1710.2021.177.043

*Д.Е.СИТНИКОВ, Ю.О.АНДРУСЕНКО*

## **РОЗРОБКА МЕТОДУ ФОРМУВАННЯ ЕФЕКТИВНОГО ПЛАНУ ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ**

---

Досліджено застосування моделі на основі згорткових нейронних мереж для задачі прогнозування поширення захворюваності COVID-19 в Україні. Серед сучасних моделей нейронних мереж для даної задачі обрана InceptionTime на основі архітектури Inception-v4. Дослідження реалізовано за допомогою високорівневої мови програмування Python. Проведено 30 експериментів з навчання моделі на вхідній вибірці даних з використанням різних ознак. Модель продемонструвала високу точність і швидкодію.

### **1. Вступ**

Пандемія раніше невідомого коронавірусу SARS-CoV-2, що викликає атипову пневмонію COVID-19, впливає на життя людей по всьому світу. У багатьох країнах введені карантинні обмеження, які істотно позначаються на всіх соціально-економічних сферах і світовій економіці в цілому.

Наслідком COVID-19 в Україні стало падіння ВВП і майже повна зупинка діяльності у сфері роздрібною торгівлі, готельного та ресторанного бізнесу, авіаперевезень. В регіонах, де існує пом'якшення карантинних обмежень, спостерігаються піки захворюваності. Через те, що вакцинація від COVID-19 в Україні розпочата нещодавно та довіра населення до вакцинації дуже низька, перспективи завершення карантину, а, відповідно, і нормалізації життя населення віддалені.

В таких умовах значно підвищується актуальність задач прогнозування фінансово-економічних, логістичних та навіть продовольчих показників, на які впливає поширення COVID-19 в Україні. Тому для прийняття рішення про посилення або пом'якшення карантинних обмежень необхідно прогнозувати поширення захворюваності по регіонах України в довгостроковій і короткостроковій перспективі з максимальною точністю.

Отже, розглянемо задачу прогнозування поширення COVID-19 в Україні в такій постановці. Вхідними даними є часові ряди показників захворюваності по регіонах: кількість підтверджених випадків, кількість смертей, кількість активних випадків, приріст підтверджених випадків і приріст активних випадків. Необхідно спрогнозувати приріст підтверджених випадків з мінімальною похибкою.

## 2. Основна частина

На сьогоднішній день існує значна кількість методів прогнозування часових рядів (time series, TS). Умовно їх можна розділити на статистичні, структурні і комбіновані [1].

Загальним недоліком статистичних моделей є складність вибору типу моделі і підбору її параметрів. При вирішенні задач прогнозування аналітику доводиться приймати рішення щодо таких характеристик часового ряду як тренд, сезонна і циклічна компоненти, робити припущення про модель часового ряду - адитивної, мультиплікативної тощо. Автоматичного способу виявлення трендів у часових рядах не існує. У той же час при вивченні кривої, що відображує результати спостережень, аналітику важко робити припущення щодо повторюваності форми кривої через рівні проміжки часу [2]. Все це істотно збільшує суб'єктивний внесок учасників процесу аналізу і прогнозування TS. Таким чином, результат аналізу і прогнозування TS залежить як від кваліфікації аналітика в предметній галузі, так і від його кваліфікації в методах аналізу.

Серед структурних моделей варто відзначити моделі на основі нейронних мереж (artificial neural networks, ANN). Здатність нейронної мережі до різнобічної обробки інформації впливає з її здатності до узагальнення і виділення прихованих залежностей між вхідними та вихідними даними [3]. Великою перевагою нейронних мереж є те, що вони здатні до навчання і узагальнення накопичених знань. Апарат ANN передбачає мінімальну участь аналітика в формуванні моделі часового ряду, оскільки здатність нейромережевих моделей до навчання дозволяє виявити приховані взаємозв'язки і закономірності між даними, а алгоритми навчання адаптують вагові коефіцієнти відповідно до структури даних, представлених для навчання [4].

Використання апарату ANN для прогнозування TS полягає в формуванні ANN певної структури, в її параметричному налаштуванні на основі поведінки досліджуваної системи в заздалегідь відомі моменти часу, в передбаченні майбутньої поведінки системи за її передісторією. Вибір архітектури нейронної мережі обумовлюється специфікою і складністю розв'язуваної задачі.

Для прогнозування TS широко застосовуються архітектури рекуррентної нейронної мережі (recurrent neural network, RNN) і згорткової нейронної мережі (convolutional neural network, CNN).

Ідея RNN полягає в послідовному використанні інформації, тобто виконується одна й та ж задача для кожного елемента послідовності, причому вихід залежить від попередніх обчислень. На відміну від інших алгоритмів машинного навчання, рекуррентні нейронні мережі з довгою короткочасною пам'яттю (long short-term memory, LSTM) здатні автоматично виявляти ознаки з тимчасових послідовностей, обробляти багатовимірні дані, а також виводити послідовності змінної довжини, завдяки чому їх можна використовувати для прогнозування [5].

Однією з найвідоміших архітектур є згорткова. CNN широко застосовується для ефективного розпізнавання зображень. Її особливість полягає в використанні згорткових і пулінгових шарів. Згортковий шар нейронної мережі являє собою застосування операції згортки до виходів з попереднього шару, де ваги ядра згортки є навчальними параметрами. Ядро являє собою фільтр, який ковзає по вихідній матриці і знаходить певні ознаки об'єктів. Операція пулінг на результатах згортки - перетворення цієї множини результатів в меншу кількість елементів. Типова CNN має кілька згорткових і пулінгових шарів, розташованих послідовно. Для класифікації і прогнозування часових рядів застосовуються одномірні CNN.

У багатьох сучасних публікаціях увага приділяється методам, для яких характерні висока точність і масштабованість. Одним з таких методів є InceptionTime, представлений у грудні 2020 року, який демонструє високу швидкість навчання для часових рядів за декількома ознаками.

Модель InceptionTime - ансамбль 5 моделей глибоких згорткових нейронних мереж, на основі архітектури Inception-v4. Кожна модель створюється шляхом каскадування декількох модулів Inception [6]. Основна ідея модуля Inception - одночасне застосування декількох фільтрів до вхідного часового ряду. Модуль включає фільтри різної довжини, які дозволяють мережі автоматично отримувати відповідні характеристики як з довгих, так і з коротких часових рядів [7]. Фактично, отримано мережі з фільтрами більшого розміру, ніж у тих, що будь-коли досліджувалися для задач комп'ютерного зору, безпосередньо використовуючи той факт, що часові ряди демонструють на один вимір менше, ніж зображення. За даними авторів, InceptionTime за рівнем точності відповідає HIVE-COTE та більш масштабована: модель може навчатись на 1500 часових рядах за одну годину, що повністю недоступно для HIVE-COTE [7].

Архітектура моделі InceptionTime представлена 2 блоками, кожен складається з 3 модулів Inception, розміщених один за одним. Вхід кожного блоку передається за допомогою лінійного з'єднання, щоб подати його на вхід наступного блоку, таким чином пом'якшуючи проблему зникнення градієнта, дозволяючи прямий потік градієнта [8]. Після блоків Inception застосовується шар глобального об'

'єднання (global average pooling, GAP), який усереднює вихідні багатовимірні часові ряди за весь часовий вимір. Завершує модель традиційний повністю зв'язаний шар softmax із кількістю нейронів, рівною кількості класів у наборі даних. Схематично архітектуру моделі InceptionTime представлено на рис. 1.

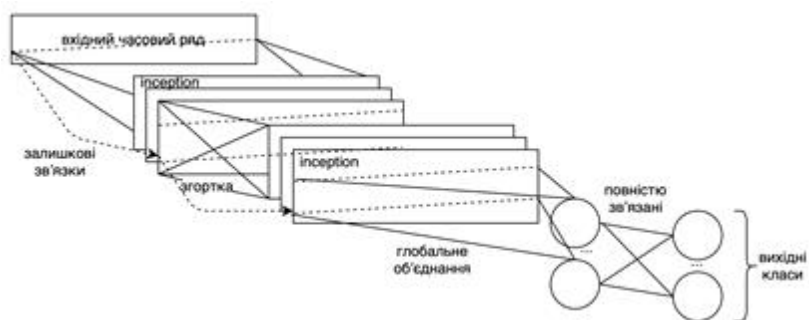


Рис. 1. Архітектура моделі InceptionTime

Зважаючи на переваги моделі InceptionTime, доцільно застосувати її для вирішення поставленої задачі та дослідити точність моделі в залежності від використовуваних ознак часового ряду.

Дослідження реалізовано за допомогою високорівневої мови програмування Python з використанням бібліотеки tsai, орієнтованої на сучасні методи класифікації, регресії та прогнозування часових рядів. Вхідні дані отримуємо з офіційного сайту РНБО за допомогою API. Нормалізація вхідних даних проведена в інтервалі від 0 до 1. Цільовий параметр прогнозування - приріст активних випадків. Точність прогнозування оцінюється середньою абсолютною похибкою (mean absolute error, MAE), середньою абсолютною похибкою у відсотках (mean absolute percentage error, MAPE), середньоквадратичною похибкою (mean squared error, MSE) та квадратного кореня з середньоквадратичною похибки (root-mean square error, RMSE).

В результаті проведено 30 експериментів навчання моделі з різним набором ознак: кількість активних випадків, приріст активних випадків, кількість підтверджених випадків, приріст підтверджених випадків, кількість смертельних випадків. Часове вікно - 56 днів, горизонт прогнозування - 14 днів. Час навчання та кількість епох навчання для кожного експерименту представлені на рис. 2.

Отже, навіть при максимальній кількості епох, час навчання не перевищує 6 хвилин 29 секунд, що

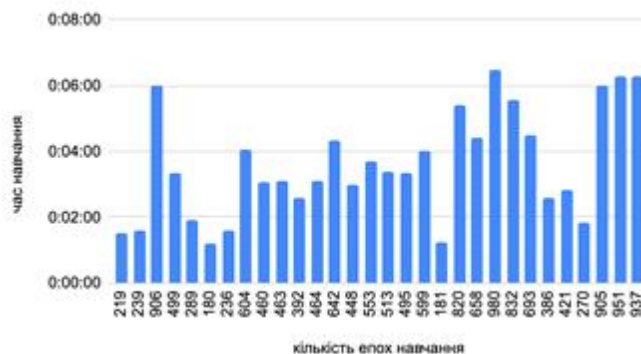


Рис. 2. Час та кількість епох навчання моделі

підтверджує швидкодню моделі InceptionTime.

Значення показників точності роботи моделі представлено на рис. 3-6.

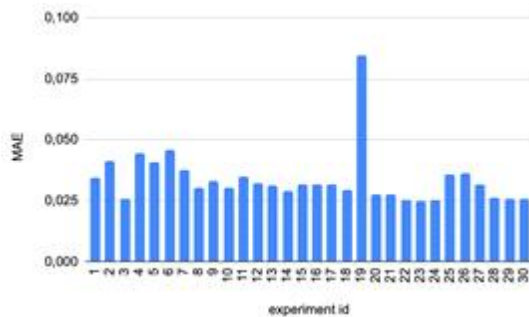


Рис. 3. Значення MAE для 30 експериментів

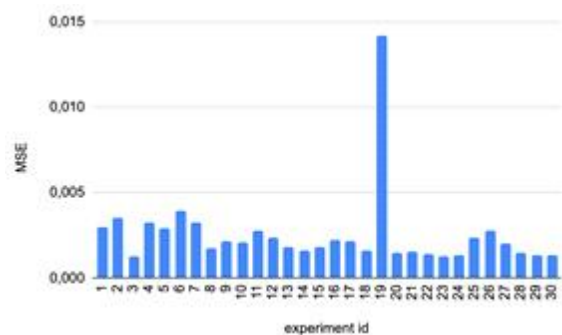


Рис. 4. Значення MSE для 30 експериментів

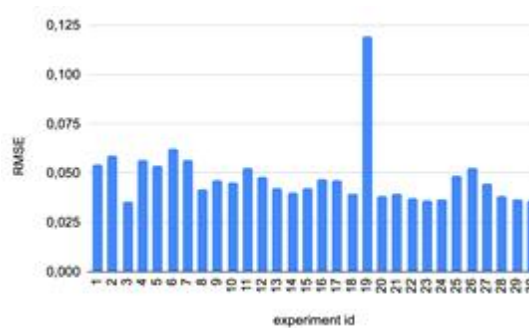


Рис. 5. Значення RMSE для 30 експериментів

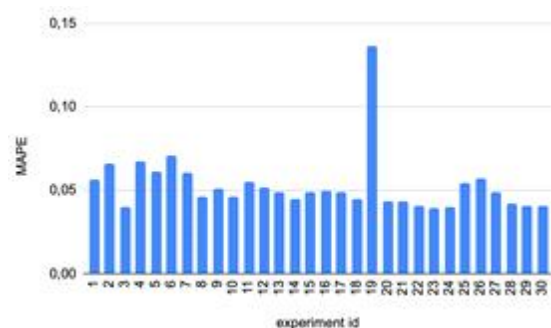


Рис. 6. Значення MAPE для 30 експериментів

Дослідження моделі InceptionTime демонструє стійкість результату, за винятком експерименту № 19. Параметри експерименту № 19 повторені у експериментах № 20 та № 21 для перевірки роботоспроможності моделі для даних ознак. Отримані результати відповідають загальній картині і вказують на технічну помилку навчання.

### 3. Висновки

Застосування моделі InceptionTime для задачі прогнозування поширення захворюваності на COVID-19 в Україні цілком можливе. Серед переваг даної моделі варто відмітити високу швидкість навчання та одночасну роботу з декількома часовими рядами, що можливо застосувати для прогнозування показників по кожному регіону окремо.

Отримані результати значень похибки демонструють високу точність роботи моделі для навчальної вибірки, де середньоквадратична похибка не перевищує 0,014, а середня абсолютна похибка - 13,6%.

Подальші дослідження мають на меті пошук найефективніших моделей на основі нейронних мереж та серед статистичних моделей. На разі розглядаються моделі CNN ResNet, FCNPlus, ResCNN, RNN\_FCPlus, LSTM\_FCPlus. Кращі моделі планується об'єднати в ансамбль для отримання оптимального прогнозу.

**Список літератури:** 1. Андрусенко Ю. О. Аналіз основних моделей прогнозування часових рядів. // Збірник наукових праць ХНУПС. 2020. №3(). С. 91-96. 2. Бокс Дж., Дженкінс Г. Аналіз временных рядов. Прогноз и управление: М. Мир, 1974. 406 с. 3. Bishop C.M. Neural Networks for Pattern Recognition. Oxford University Press, 2014. 482 p. 4. Хайкин С. Нейронные сети. Полный курс. М: Вильямс, 2006. 1104 с. 5. Wei, J., Nan, Z., Xiaoming, X. and Yanhe, X. Intelligent Deep Learning Method for Forecasting the Health Evolution Trend of Aero-Engine With Dispersion Entropy-Based Multi-Scale Series Aggregation and LSTM Neural Network. // IEEE Access, 2020. 34350-34361 pp. 6. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. // In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015. 1-9 pp. 7. Hassan Ismail Fawaz, Benjamin Lucas, Germain Forestier. InceptionTime: Finding AlexNet for Time Series Classification. // Data Mining and Knowledge Discovery. 2020. Vol. 34, 1936-1962

pp. 8. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. // In: IEEE Conference on Computer Vision and Pattern Recognition, 2016. 770-778 pp.

Надійшла до редколегії 14.05.2021

**Ситніков Дмитро Едуардович**, кандидат технічних наук, доцент, професор кафедри системотехніки ХНУРЕ. Наукові інтереси: Data Mining and Knowledge Discovery. Адреса: Україна, 61166, м. Харків, пр. Науки 14, тел. (057) 702 10 06.

**Андрусенко Юлія Олександрівна**, аспірантка кафедри електронних обчислювальних машин ХНУРЕ. Наукові інтереси: методи прогнозування часових рядів. Адреса: Україна, 61166, м. Харків, пр. Науки 14, тел. +38 (063) 407 06 09.

---

УДК 004.4

DOI: 10.30837/0135-1710.2021.177.047

*Н.С. КРАВЕЦЬ*

## **ВИЗНАЧЕННЯ ОБМЕЖЕНЬ РЕАЛІЗАЦІЇ ОБРОБКИ ПОДІЙ В ХМАРНОМУ ДОДАТКУ ЗА ДОПОМОГОЮ БЕЗСЕРВЕРНИХ ОБЧИСЛЕНЬ**

---

У статті розглядаються безсерверні обчислення як нова і перспективна парадигма для розгортання хмарних додатків, зокрема таких що мають подієво-орієнтовану архітектуру. Платформи FaaS провідних постачальників хмарних послуг є найсучаснішою реалізацією безсерверної моделі. На основі порівняння характеристик та переліку джерел оброблюваних подій платформ FaaS від Amazon і Azure визначено переваги та обмеження використання цієї моделі для реалізації високопродуктивних систем, що обробляють події в режимі реального часу.

### **1. Вступ**

У теперішній час розробникам все частіше доводиться вирішувати проблеми, пов'язані із обробкою в реальному часі потоку різноманітних подій, що надходять з різних джерел і вимагають різної обробки. Подібні проблеми покликана вирішувати подієво-орієнтована архітектура. Для вирішення проблем використовуються: асинхронна обробка подій, черги різних типів, механізм підписки на події тощо. Хмарні провайдери пропонують значну кількість рішень на базі своїх платформ, які також припускають використання безсерверних обчислень.

Безсерверні обчислення - це новий підхід, який концептуально ще більше дистанціює програмне забезпечення від інфраструктури, на якій воно виконується. Цей рівень абстракції істотно полегшує життя розробникам, дозволяючи зосередитися тільки на реалізації конкретних функцій. Хоча для реалізації безсерверних обчислень клієнта використовуються фізичні сервери, розробникам не потрібно думати про їх конфігурацію, характеристики та обслуговування. Компанія, що користується послугами безсерверного постачальника, по факту сплачує за використовувані ресурси і не повинна резервувати і оплачувати фіксовану пропускну здатність або кількість серверів, оскільки послуга автоматично масштабується.

У багатьох хмарних провайдерів, що пропонують безсерверні обчислення як послугу, є платформи Function-as-a-Service (FaaS), які дозволяють створювати прості функції, які незалежно запускаються при настанні якоїсь події і виконують одну задачу. За допомогою FaaS розробники можуть створювати модульну архітектуру, роблячи код більш масштабованим, не витрачаючи ресурси на підтримку бекенда. Основними перевагами безсерверних обчислень є низькі експлуатаційні витрати і ефективне управління та використання ресурсів. На теперішній час безсерверні обчислення пропонуються декількома постачальниками загальнодоступних хмарних сервісів. Безсерверні обчислення активно розвиваються, а їх реалізації від різних хмарних провайдерів мають істотні відмінності, аналіз яких є метою даної статті. AWS Microsoft і Azure на даний момент є лідерами ринку хмарних послуг, порівняємо можливості їх платформ FaaS з точки зору обробки подій.

## 2. Порівняння Azure Functions та AWS Lambda

Як правило, безсерверна технологія має наступні характеристики:

- відсутність керування серверами: не потрібно надавати або обслуговувати будь-які сервери, немає доступу до програмного забезпечення або середовища виконання для установки, обслуговування або адміністрування серверів;
- автоматичне масштабування: не потрібно турбуватися про масштабування додатка в разі різкого стрибка навантаження;
- більш «тонкий» білінг: до появи безсерверних хмарних сервісів в результаті автоматичного масштабування автоматично надавалися віртуальні машини, тобто резервувалися ресурси, які клієнт оплачував, навіть якщо вони простоювали; під час використання безсерверних обчислень оплата стягується на основі споживання ресурсів і часу, необхідного для виконання;
- виконання в результаті настання якоїсь події: зазвичай функція виконується після настання події, наприклад, завантаження нового файлу в сховище або спрацьовування таймера;
- висока доступність: висока доступність і відмовостійкість надаються безсерверним додаткам за замовчуванням.

Найчастіше функція як сервіс взаємодіє із масивом хмарних служб для реалізації наступних сценаріїв: створення веб-API, обробка переданих файлів, створення безсерверного робочого процесу, реагування на зміни бази даних, виконання запланованих завдань, створення надійних систем черги повідомлень, аналіз потоків даних Інтернету речей, обробка даних в реальному часі. Однак безсерверні платформи мають власну специфіку, яка залежить від виробника.

Наприклад, і AWS Lambda і Azure Functions [1,2] дозволяють використовувати крім базового набору будь-яку додаткову мову програмування за допомогою API Runtime, Custom handler або gRPC language worker. При цьому AWS Lambda включає в себе підтримку довільних бібліотек, проте існують певні обмеження на модель програмування, наприклад, максимальний час виконання, відсутність збереження стану.

У Azure Functions від вибору варіанту розміщення залежить реалізація автоматичного масштабування, максимально можлива кількість примірників; перелік ресурсів, доступних для кожного примірника програми-функції; підтримка розширених функцій, таких як підключення до віртуальної мережі Azure. При розгортанні функції AWS Lambda в залежності від профілю робочого навантаження потрібно визначити тільки максимальний розмір виділеної пам'яті, потужність процесора і вартість виконання функції пропорційні виділеній пам'яті. Щоб запустити безсерверну функцію в AWS, її потрібно розгорнути в сервісі AWS Lambda. Microsoft дотримується іншого підходу: поняття моделі програмування Azure Functions відокремлено від безсерверної операційної моделі.

AWS Lambda має просту модель програмування: функція отримує об'єкт JSON в якості вхідних даних і може повертати інший JSON в якості вихідних [2]. Тип події визначає схему тих об'єктів, які задокументовані і визначені в мові SDK. Функції Azure мають більш складну модель, засновану на тригерах та прив'язках. Тригер задає подію, яка прослуховується функцією. Функція може мати будь-яку кількість вхідних і вихідних прив'язок для вилучення та/або повернення даних під час обробки [1].

Безсерверні функції - це невеликі блоки коду, що виконують тільки одне завдання. Тому створення великих додатків і систем з цих невеликих шматочків, є нетривіальною проблемою, але деякі рішення існують. Безсерверні обчислення засновані на концепції контейнеризації. Кожного разу, коли функція запускається, система створює контейнер, в якому її можна запустити. У той час, як запуск функції може зайняти всього мікросекунди, запуск контейнера займає 1-2 секунди. Для багатьох додатків це досить швидко, в порівнянні зі створенням віртуальної машини, проте в деяких випадках подібний час затримки може виявитися критичним. І AWS, і Azure мають спеціальні сервіси для оркестровки робочих процесів. Часто функції використовуються в якості кроків в цих робочих процесах, що дозволяє їм залишатися незалежними, але при цьому вирішувати важливі завдання.

Більш докладне порівняння характеристик платформ FaaS від Amazon і Azure наведено в табл. 1.

Однією з важливих характеристик послуг FaaS є набір підтримуваних типів подій. У табл. 2 наведено джерела подій для AWS Lambda та Azure Functions.

Проблемою при обробці подій може стати некоректно сформована подія або збій одного з екземплярів функції. Для забезпечення надійності обробки повідомлень використовують-



Таблиця 1 -

## Порівняння характеристик AWS Lambda та Azure Functions

Характеристика	AWS Lambda	Azure Functions
Підтримувані мови програмування	C#, Java, Python, Node.js, Ruby, Go, API Runtime	C#, JavaScript (Express.js, Node.js), F#, Java, Python, TypeScript, <b>Custom handler/ gRPC language worker</b>
Автоматичне масштабування	Функції Lambda не зберігають стан, кількість оброблюваних запитів не обмежена	Визначається планом розміщення Consumption plan, Premium plan, and Dedicated (App Service) plan
Оркестратор	AWS Step Functions	<b>ASE, Kubernetes , Kubernetes c Azure Arc</b>
Тригери функцій	Підписка на події, тригери, Lambda API	Тригери і прив'язки
Інтеграція черг повідомлень	+	+
Інструмент моніторингу	Amazon Cloud Watch	Azure Application Insights
Підтримка CLI	+	+
Підтримка PowerShell	+	+
Підтримка шаблону ARM	+	+
Управління	Консоль AWS Lambda, AWS SDK	REST API, Visual Studio
Контекст виконання	У хмарі	Локально /у хмарі

ся черги. У Azure Functions при обробці повідомлення з черги функція може заблокувати некоректну подію, а також гарантує, що обробить подію хоча б один раз. У потоках подій (наприклад, в концентраторах подій Azure) блокування не використовуються. Ці служби налаштовані так, щоб забезпечити високу пропускну здатність, підтримувати кілька груп споживачів і можливість відтворення.

У AWS Lambda при безпосередньому виклику функції потрібно визначити стратегію обробки помилок: повторити спробу, відправити подію в чергу для налагодження або проігнорувати помилку. Якщо функція викликається побічно, можливі наступні сценарії: асинхронний виклик, зіставлення джерел подій. При асинхронному виклику AWS Lambda повторює помилковий запит двічі. Якщо у функції недостатньо можливостей для обробки всіх вхідних запитів, події можуть чекати в черзі протягом декількох годин або днів, щоб бути відправленими в функцію. Якщо використовується зіставлення джерел подій, які зчитуються з потоків, повторюється весь пакет елементів. Повторні помилки блокують обробку порушеного блоку доти, поки помилка не буде усунена або термін дії елементів не закінчиться.

З точки зору реалізації паралельної обробки подій, як AWS, так і служби Azure здатні здійснювати кілька виконань однієї і тієї ж функції одночасно.

Таким чином реалізація обробки подій у хмарі за допомогою безсерверних технологій має наступні переваги: відсутність необхідності керувати інфраструктурою; суттєве скорочення часу випуску додатка; «тонкий» білінг; висока продуктивність, доступність, відмовостійкість та масштабованість. До ризиків слід віднести обмеження на модель програмування; проблеми з оркестрацією, якщо додаток складається із великої кількості незалежних функцій; збільшення часу відгуку внаслідок так званого «холодного» запуску, якщо щільність потоку оброблюваних подій недостатньо велика; потреба у додаткових налаштуваннях задля надійності обробки подій, якщо щільність потоку оброблюваних подій досить велика.

### 3. Висновки

Для високопродуктивних систем, що обробляють події в режимі реального часу, головні характеристики - це час відгуку на запит і доступність, час відгуку (response time) - те, що бачить клієнт: крім фактичного часу обробки запиту (час обслуговування, service time), він включає затримки при передачі інформації по мережі і затримки повідомлень в черзі.

Таблиця 2 -

Джерела подій, що обробляються AWS Lambda та Azure Functions

Джерело оброблюваних подій	Azure Functions	AWS Lambda
Сховища даних	Сховище BLOB-об'єктів, Azure Cosmos DB	Amazon DynamoDB, Amazon Cognito, Amazon Simple Storage Service Batch, Amazon Simple Storage Service
Черги	Queue storage, Service Bus	Amazon Simple Queue Service
Середовище для створення мікросервісних додатків	Dapr	
Брокер подій	Event Grid, Event Hubs	Amazon CloudWatch Events
HTTP-запити та веб-перехоплювачі	HTTP тригери, API, веб-перехоплювачі, Azure SignalR	Amazon API Gateway
Інтернет речей	IoT Hub	AWS IoT, AWS IoT Events
Брокер повідомлень	RabbitMQ, Kafka	Amazon MQ (Apache ActiveMQ), Amazon Kinesis, Amazon Managed Streaming for Apache Kafka, self-managed Apache Kafka, Amazon Connect, Amazon Simple Notification Service, Amazon Simple Email Service
Таймер	Тригер таймера	Можливий запуск функцій за розкладом, використовуючи Amazon CloudWatch
Сервіси моніторингу та управління	Через Event Grid та Event Hubs	Elastic Load Balancing, Amazon CloudWatch, AWS CodeCommit, AWS CodePipeline
Сервіси із голосовим управлінням		Amazon Lex, Alexa
Сервіси доставки даних	Через Event Grid та Event Hubs	Amazon Kinesis Data Firehose, CloudFront
Сервіси роботи з інфраструктурою як з кодом		AWS Cloud Formation, AWS Config

Обмеженням безсерверних обчислень є так званий холодний запуск: запуск «холодних» функцій, тобто тих, які не запускалися протягом тривалого часу, займає додатковий час. Можна вирішувати цю проблему, періодично запускаючи функції, це гарантує, що вони залишаться «теплыми», але разом з тим це призводить до додаткових витрат і зводить нанівець одну із основних переваг безсерверної системи - «тонкий» білінг. Час запуску функції можна оптимізувати, але цього не достатньо [3]. Таким чином, проблеми, пов'язані із холодним запуском, означають, що висока продуктивність, яку пропонують безсерверні обчислення, досяжна тільки за певних умов. І AWS Lambda, і Azure дозволяють уникнути холодного запуску в преміальних і виділених планах.

Безсерверні обчислення стають новою і привабливою парадигмою для розгортання хмарних додатків, багато в чому завдяки недавньому переходу від архітектури корпоративних додатків до контейнерів і мікросервісів. Але найчастіше запускати весь додаток у безсерверному середовищі недоцільно або економічно не вигідно, тому ця модель передбачає можливість запускати в ній певні частини програми, яким постійно потрібна висока продуктивність і значні обчислювальні ресурси. FaaS - це більше розширення, яке працює разом із контейнерами або віртуальними машинами. Таким чином, потрібно розглядати

безсерверні рішення скоріше як спосіб доповнити різні типи архітектур додатків і підєво-орієнтовану архітектуру зокрема.

**Список літератури:** 1. *Sreeram P. K.* Azure Serverless Computing Cookbook: Build and monitor Azure applications hosted on serverless architecture using Azure functions. - Packt Publishing Ltd, 2020. 2. *Sbarski P., Kroonenburg S.* Serverless architectures on Aws: with examples using Aws Lambda. - Shelter Island : Manning Publications Company, 2017. - С. 376. 3. *McGrath G., Brenner P. R.* Serverless computing: Design, implementation, and performance //2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW). - IEEE, 2017. - С. 405-410.

Надійшла до редколегії 02.06.2021

**Кравець Наталя Сергіївна**, кандидат технічних наук, доцент, доцент кафедри програмної інженерії ХНУРЕ. Наукові інтереси: хмарні технології, паралельні обчислення. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. (057) 702 14 46.

УДК 004.62

DOI: 10.30837/0135-1710.2021.177.051

*М.С. ТИТОВСЬКОЇ, О.В. ХРЯПКІН*

## ДОСЛІДЖЕННЯ МЕТОДІВ МІГРАЦІЇ ДАНИХ В СИСТЕМАХ ЕЛЕКТРОННОЇ КОМЕРЦІЇ НА ПРИКЛАДІ CMS MAGENTO

Проведено аналіз особливостей та ризиків міграції даних на прикладі системи електронної комерції Magento. Визначено основні стратегії міграції даних. Для вибору найкращих методу та стратегії міграції даних проведено опитування експертів. За результатами опитування обґрунтовано вибір найкращих методу та стратегії міграції даних для систем електронної комерції різних масштабів.

### 1. Вступ

Система електронної комерції - це інформаційна система (ІС), яка інтегрує відповідне апаратне і програмне забезпечення для досягнення певних функціональних можливостей [1]. Серед таких можливостей можна виділити основні: прийом і оформлення замовлень по каталогах і прайс-листах через Інтернет на товари різних категорій, зберігання замовлень в єдиній базі даних, реєстрація користувачів, підтримка віддаленого адміністрування; обробка замовлень за стандартною схемою (обробка, поставка, звітно-фінансові документи).

Тема електронної комерції є популярною в Європі, що підтверджують статистичні дані eurostat [2]. Як показано на рис. 1, з 2010 року частка онлайн-продажів щорічно зростає серед покупців всіх вікових груп, що свідчить про існування величезного

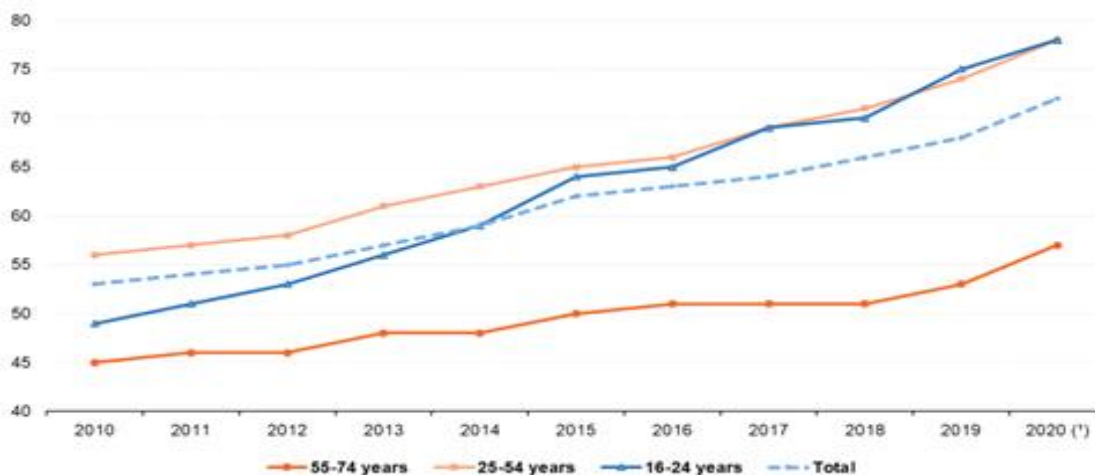


Рис.1. Змінювання частки онлайн-продажів серед користувачів Інтернету різного віку

потенціалу приросту й надалі. Неможливо також не зазначити позитивний вплив обмежень, викликаних наслідками пандемії Covid-19, на розвиток електронної комерції (e-commerce) в цілому [3].

Безперервне зростання частки інтернет-продажів та високий потенціал їх збільшення призводять до того, що все більше старих систем, використовуваних для реалізації електронної комерції, не відповідають сучасним вимогам відмовостійкості, безпеки та конфіденційності даних. Звідси перед власниками багатьох зростаючих інтернет-магазинів гостро стає питання міграції даних зі старої системи електронної комерції в більш сучасну, яка буде не тільки відповідати актуальним вимогам сьогодні, але й вчасно модернізуватися відповідно до тенденцій, що постійно змінюються.

## **2. Обрання системи електронної комерції**

Однією з таких систем є система управління контентом (CMS) Magento. На даний час вона вважається однією з найкращих та входить в п'ятірку найпопулярніших CMS для реалізації електронної комерції [7, 8]. Власники торговельного бізнесу обирають Magento як платформу для розгортання інтернет-магазину «з нуля», так і, все частіше, як систему для зміни використовуваної платформи на зразок WordPress, PrestaShop, OpenCart тощо. CMS Magento завоювала популярність завдяки наявності: безкоштовної версії; гарної документованості та стандартизованості; якісної підтримки та розвитку платформи; значної кількості необхідних функцій «з коробки», а також доступних платних та безкоштовних програмних модулів, призначених для розширення функціональності за замовчуванням; значної кількості кваліфікованих (навіть сертифікованих) фахівців, здатних змінити або розширити функціональність системи на вимогу замовника.

У даної CMS існує дві актуальні версії: 1.9 (далі Magento 1) та 2.4 (далі Magento 2). Обидві версії підтримуються, але Magento 1 більше не оновлюється, незважаючи на те, що вона є найбільш стабільною версією провідної платформи з відкритим вихідним кодом для створення високонавантажених e-commerce систем [9].

Виходячи з перелічених факторів, метою роботи є обґрунтування вибору методу та стратегії міграції даних для CMS на прикладі CMS Magento 2. Для досягнення цієї мети передбачається вирішення таких задач:

- дослідження методів та ризиків міграції даних;
- дослідження стратегій міграції даних;
- обґрунтування вибору методу та стратегії міграції даних для проектів CMS різних типів.

## **3. Методи міграції даних**

Міграція даних - це процес перенесення будь-яких даних з однієї системи в іншу. Існує три методи міграції даних на Magento 2: використання стандартного модуля міграції [10, 11], унікальна розробка модуля міграції [12], використання стороннього сервісу [13].

Стандартний модуль міграції був розроблений безпосередньо командою Magento через деякий час після впровадження другої версії та орієнтований на міграцію даних стандартних сутностей системи. Він також може бути додатково налаштований для перенесення змінених сутностей. Даний модуль не йде «з коробки», а за необхідністю може бути безкоштовно завантажений з офіційного сайту та встановлений додатково. Стандартний модуль не має графічного інтерфейсу - процес міграції запускається за допомогою консольних команд безпосередньо на сервері, де розташований проект. Налаштування системи для міграції методом стандартного модуля, як правило, виконує backend-розробник.

Унікальна розробка модуля міграції дозволяє отримати гнучкіше рішення, за допомогою якого можливо перенести дані з будь якої системи електронної комерції на Magento 2. Проте даний спосіб потребує чимало ресурсів на аналіз обох CMS, безпосередньо розробку та тестування модулів.

Використання стороннього сервісу, прикладом якого може слугувати Cart2Cart, передбачає отримання даних з системи-донора, їх обробку за принципом чорного ящика на боці сервісу та заповнення вже обробленими даними системи-реципієнта. Як правило, вартість міграції за допомогою стороннього сервісу залежить від об'єму даних, які необхідно перенести. Слід зауважити, що при використанні цього способу всі дані, що зберігаються у базі даних, в тому числі й персональні дані покупців, потрапляють до сторонньої організації, можливо, з іншої країни. У зв'язку з цим дуже складно проконтролювати дотримання умов конфіденційності інформації.

Також слід зазначити, що для методів з використанням стандартного модуля та стороннього сервісу може бути потрібна додаткова розробка, коли наявна функціональність не охоплює всі потреби при міграції.

Перенесення даних включає три основні кроки: витяг даних, перетворення даних, завантаження даних [4]. Оскільки для власників торгового бізнесу система електронної комерції є не тільки прямим засобом досягнення головної мети - прибутку, а ще й сховищем даних для аналітики та формування звітності, коректне збереження даних при міграції в системах даного типу критично важливе.

#### **4. Ризики в процесі міграції даних**

Зі збільшенням обсягу даних процес міграції став більш складним. Це робить перенесення даних набагато ризикованішою справою. Ці ризики пов'язані з наступними факторами [5].

**Ризик втрати даних.** Під час процесу міграції даних може статися їх втрата. Коли дані переносяться в нову систему або цільову систему, деякі дані можуть не перенестись із вихідної системи через різні помилки.

**Ризик простою.** Ризик тривалого простою виникає, коли процес міграції даних триває довше, ніж очікувалося. Під час процесу міграції вихідна система не працює, тому це створює потенційні ризики для організацій та зацікавлених сторін.

**Ризик втручання.** Ризик втручання є особливо проблематичним, коли декілька зацікавлених сторін одночасно використовують програму під час процесу міграції. Приклад ризику може виникнути, коли зацікавлена сторона блокує певну таблицю, що робить неможливим її використання для інших зацікавлених сторін.

**Ризик нестабільності програми.** Цільова програма або цільова система може бути нестабільною з ряду причин, включаючи неправильну розробку, неправильне кодування нового додатку або неправильне кодування вимог бізнесу в новій системі.

**Ризик невірної порадки виконання (оркестрації).** З ризиком оркестрації проблема виникає, коли процеси міграції даних не виконуються по порядку. Порядок міграції даних надзвичайно важливий, тим більше, що між різними бізнес-об'єктами існують різні залежності.

#### **5. Стратегії міграції даних**

Існують різні стратегії міграції даних на Magento 2: стратегія «великого вибуху», фазова, паралельна [4].

Стратегія «великого вибуху» - це стратегія перенесення всіх даних за одну операцію. Це може зайняти деякий час, але для користувачів існує єдиний момент часу, коли вони більше не можуть використовувати старі дані, а нова система запускається. Міграція великого вибуху, як правило, передбачає значні періоди підготовки та короткий час зупинки, протягом яких система недоступна. Процес міграції складається з чотирьох етапів: проектування, розробка та тестування, «великий вибух», тестування після міграції [4].

На етапі проектування планується обсяг і цілі проекту, аналізуються зразки даних та створюється план міграції даних. Виходячи з результатів етапу, буде запропонована архітектура.

На етапі розробки та тестування запропонована архітектура реалізується. На цьому етапі витрачається найбільше часу, розробляється та тестується інструмент міграції.

На етапі «великого вибуху» фактично переміщуються всі дані між системами. Зазвичай це вимагає зупинки вихідної та цільової систем для забезпечення узгодженості даних.

На етапі тестування, перевіряються всі дані, які були перенесені у цільову систему.

Фазова стратегія (ітеративна міграція) - це стратегія, при якій дані переносяться невеликими приростами з плином часу для кожного модуля. По мірі того, як кожна ітерація передається в цільову систему, помилки можуть бути усунені, і будь-яке необхідне повторне навчання користувача виконується невеликими порціями, замість того, щоб робити це одразу для всієї системи. Результат - менший ризик, ніж у випадку з міграцією «великим вибухом», але з більш тривалими часовими рамками переходу. Один з кращих способів забезпечити плавну міграцію у фазовому підході - використовувати реплікацію даних для кожної ітерації, яка є рішенням для аварійного відновлення і резервного копіювання [6].

Паралельна стратегія - це стратегія, при якій нова система встановлюється разом зі старою, і обидві працюють в тандемі під час переходу. Оновлення публікуються в обох системах до завершення перенесення. Після підтвердження того, що нова система працює належним чином,

стару вимикають. Паралельна міграція дещо зменшує ризик простою, оскільки старе середовище залишається функціональним, поки паралельне середовище створюється.

Перевага цього підходу полягає в тому, що поточна діяльність не порушується, і питання міграції можуть бути повністю вирішені до того, як цільова система почне цим займатися. Це найменш ризикована з трьох стратегій, оскільки у разі виникнення проблем із новою системою ви можете повернутися до попереднього стану системи [6].

### 6. Дослідження методів міграції в системах електронної комерції

Для визначення метода та стратегії міграції були залучені 32 експерта з трьох ІТ-компаній, які спеціалізуються на розробці програмних продуктів на базі системи електронної

Таблиця 1

Позначення	Визначення
m1	Стандартний модуль міграції
m2	Унікальна розробка модуля міграції
m3	Використання стороннього сервісу

Таблиця 2

Позначення	Визначення
s1	Стратегія «великий вибух»
s2	Фазова стратегія
s3	Паралельна стратегія

Таблиця 3

	s1	s2	s3
m1	37	7	0
m2	11	3	2
m3	7	1	0

методу міграції, але вони не були відображені в таблиці окремо, оскільки не є повноцінною унікальною розробкою, а також не можуть бути віднесені до інших стратегій через неподільність з основним методом, але й не можуть бути опущені, бо впливають на критерії вибору методу міграції.

Як відомо, в міграції значну роль відіграє обсяг даних, які треба перенести, тому було прийнято рішення розбити всі проекти на 3 групи: дрібні, середні й великі. Для цього було запропоновано виділити основні сутності й розбити проекти на групи, відштовхуючись від загальної кількості записів для цих сутностей, оскільки вони мають різну кількість записів в таблицях та не залежать одна від одної. Експерти виділили наступні основні сутності для міграції: категорії товарів, товари, клієнти, замовлення.

комерції Magento. Всі залучені експерти мають досвід web-розробки або управління проектами більше трьох років (рівень кваліфікації «middle», «senior») та досвід здійснення міграції даних з різних систем електронної комерції на Magento 2 від 1 до 4 разів; деякі з них працювали над міграцією сумісно.

Експертам було запропоновано обрати метод та стратегію, які вони використовували. Для зручності обробки результатів введені умовні позначення, наведені в таблицях 1, 2.

Результати опитування експертів наведені у таблиці 3.

Загальний досвід експертів - 68 міграцій. Серед них - 59 міграцій з системи Magento 1 до Magento 2 та 9 міграцій з інших платформ до Magento 2. Зі стандартним моду-

лем міграції за стратегією «великий вибух» було виконано 37 міграцій; зі стандартним модулем та за фазовою стратегією було виконано 7 міграцій. З використанням унікальної розробки модуля та стратегії «великий вибух» виконано 11 міграцій, за фазовою стратегією - 3 міграції, за паралельною стратегією - 2 міграції. Щодо використання стороннього сервісу, то за стратегією «великий вибух» було зроблено 7 міграцій та 1 міграція за фазовою стратегією.

Експерти зауважили, що 9 міграцій, серед них 3 у стандартному модулі міграції та 6 в унікальній розробці модуля, можуть бути віднесені до змішаного

Таким чином, за допомогою експертів всі проекти були розбиті на групи наступним чином: дрібні (загальна кількість записів менше 100000); середні (загальна кількість записів від 100000 до 1000000); великі (загальна кількість записів більше 1000000).

В загальну кількість записів входять також суміжні таблиці, без яких неможливе функціонування сутностей системи (EAV модель). Модель «сутність - атрибут - значення» (EAV) - це модель для зберігання значень атрибутів сутності в певному місці зберігання. Для зберігання Magento 2 підтримує MySQL-сумісні бази даних. У плоскій моделі значення атрибутів зберігаються в тій самій таблиці, що і сутності; для кожного атрибута в таблиці створюється окремий стовпець. У моделі EAV значення атрибутів зберігаються в окремій таблиці. Для кожного атрибута не створюється окремий стовпець, а для кожного значення атрибута сутності в таблиці EAV створюється новий рядок. Такі сутності уособлюють математичну концепцію розрідженої матриці [14].

Результат вибору експертів за критерієм загальної кількості записів можна побачити в таблиці 4 (результат вибору метода та стратегії для дрібних проектів), таблиці 5 (результат вибору метода та стратегії для середніх проектів) та таблиці 6 (результат вибору метода та стратегії для великих проектів).

Результати в таблиці 4 свідчать, що для міграції, де загальна кількість записів дрібна, більш за все використовувався стандартний модуль міграції та сторонній сервіс зі стратегією «великого вибуху».

За результатами в таблиці 5 можна сказати, що для міграції з середньої групи записів більш за все використовувались унікальна розробка, стандартний модуль міграції і сторонній сервіс зі стратегією «великого вибуху». Були також здійснені міграції з фазовою стратегією з використанням стандартного модуля міграції.

Щодо результатів в таблиці 6, то для великої групи записів використовувався стандартний модуль міграції зі стратегією «великого вибуху» та фазова стратегія. Для великого об'єму даних, експерти використовували також унікальні розробки, серед яких застосовувались стратегії: «великого вибуху», паралельна та фазова.

За результатами обраних варіантів експертами зазначено, що для всіх типів проектів частіше обирається варіант зі стратегією «великого вибуху» та стандартний модуль міграції, бо використання інших варіантів не підходить з економічної точки зору. Для великих проектів можливе використання фазової та паралельної стратегії, бо кошти, втрачені, поки проект буде вимкнений при стратегії «великого вибуху», можуть бути більшими, ніж вартість самої міграції даних.

## 7. Висновки

В статті розглянуто основні особливості систем електронної комерції. Відзначено, що з 2010 року кількість продажів в цих системах для покупців всіх вікових груп постійно зростає.

Були визначені причини необхідності міграції між системами: невідповідність старих систем електронної комерції сучасним вимогам відмовостійкості, безпеки та конфіденційності даних. В якості цільової системи для міграції даних обрана CMS Magento, яка входить в п'ятірку найпопулярніших CMS для реалізації електронної комерції. Описані її переваги та існуючі версії.

Розглянуті способи міграції між системами електронної комерції: використання стандартного модуля міграції, унікальна розробка модуля міграції, використання стороннього сервісу. Визначені кроки міграції, до яких входить витяг даних, перетворення даних, завантаження

Таблиця 4

	s1	s2	s3
m1	8	0	0
m2	1	1	0
m3	5	0	0

Таблиця 5

	s1	s2	s3
m1	9	3	0
m2	2	1	0
m3	2	1	0

Таблиця 6

	s1	s2	s3
m1	20	4	0
m2	8	1	2
m3	0	0	0

даних. Визначені фактори ризику міграції, які можуть виникнути при обробці великих баз даних: ризик простою, втрати даних, втручання, нестабільної програми, неввірного порядку виконання (оркестрації).

Проаналізовані існуючі стратегії міграції даних на Magento 2: стратегію «великого вибуху», фазову, паралельну.

Для дослідження методів міграції між системами електронної комерції була залучена група експертів. Мета опитування експертів: обрання методів та стратегій міграції даних, які вони використовували. Загальний досвід експертів складав 68 міграцій. Експерти виділили критерій оцінювання - загальну кількість записів у базі даних. В результаті опитування стало відомо, що для дрібних проєктів (менше 100000 записів) частіше за все обираються стратегія «великого вибуху» та стандартний модуль міграції або сторонній сервіс. У середніх проєктах обираються стратегії «великого вибуху» та фазова. Щодо методів, то обирались: стандартний модуль, унікальна розробка та сторонній сервіс. Зріст кількості міграцій проєктів з використанням фазової та паралельної стратегій зі збільшенням розміру проєкту обумовлений з економічної точки зору: витрати при використанні стратегії «великого вибуху» можуть бути більшими, ніж при обранні інших стратегій.

**Список літератури:** 1. *Савельєва Н.В.* Системы управления контентом // Открытые системы. СУБД. - 2004. №4. 2. *E-commerce statistics for individuals* [Електронний ресурс]. 2021. Режим доступу до ресурсу <https://ec.europa.eu/eurostat/statistics-explained/pdfscache/46776.pdf>. 3. *Ecommerce Europe. Impact of the Coronavirus on e-commerce survey results report* [Електронний ресурс] / Ecommerce Europe. 2021. Режим доступу до ресурсу: <https://www.ecommerce-europe.eu/wp-content/uploads/2021/01/Coronavirus-Survey-Report-January-2021.pdf>. 4. *Morris J.* Practical Data Migration / Johny Morris. - UK: BCS Learning & Development Limited, 2020. 266 с. 5. *Gawande S.* The Data Migration Process & the Potential Risks [Електронний ресурс] / Sandesh Gawande. 2015. Режим доступу до ресурсу: <https://icedq.com/data-migration/the-data-migration-process-and-the-potential-risks>. 6. *Telling K.* Dealing with data migration: how to choose the right fit for your organisation [Електронний ресурс] / Kerry Telling. 2018. Режим доступу до ресурсу: <https://www.itproportal.com/features/dealing-with-data-migration-how-to-choose-the-right-fit-for-your-organisation/>. 7. *Burets A.* Best eCommerce CMS in 2020 [Електронний ресурс] / Alex Burets. 2020. Режим доступу до ресурсу: <https://scand.com/company/blog/best-ecommerce-cms-in-2020/>. 8. *Khaneja D.* 10 Best eCommerce CMS in 2020 with Recommendations [Електронний ресурс] / Divya Khaneja. 2021. Режим доступу до ресурсу: <https://webscoot.io/blog/best-ecommerce-cms/>. 9. *eCommerce Platform, Best eCommerce Software for Selling Online, Magento* [Електронний ресурс]. 2018. Режим доступу до ресурсу: <https://magento.com/>. 10. *Fayyaz Khattak.* How To Migrate From Magento 1 To Magento 2 Without Breaking Anything On Your Store [Електронний ресурс] / Fayyaz Khattak. 2018. Режим доступу до ресурсу: <https://www.cloudways.com/blog/migrate-from-magento-1-to-magento-2/>. 11. *Surabhi S.* Magento Migration - From Magento 1 to 2 in 5 Simple Steps [Електронний ресурс] / Shukla Surabhi. - 2019. - Режим доступу до ресурсу: <https://www.netsolutions.com/insights/how-to-migrate-from-magento-1-to-magento-2/>. 12. *Фишбах Р.* Миграция данных при модернизации IT-среды предприятия / Р. Фишбах, И. Лахманн, М. Виннемут. // Рациональное Управление Предприятием. 2008. No4. 13. *Terebetska A.* Welcome to a Top of Value [Електронний ресурс] / Alina Terebetska. 2019. Режим доступу до ресурсу: <https://www.shopping-cart-migration.com/must-know-tips/magento-2-3-welcome-to-a-ton-of-value>. 14. *BelVG.* Magento 2 Certified Professional Developer Guide [Електронний ресурс] / BelVG. Режим доступу до ресурсу: <https://belvg.com/tutorial/magento-2-certified-professional-developer-guide-section-5>.

*Надійшла до редколегії 11.06.2021*

**Тіговської Микита Сергійович**, студент групи СПРМ-19-2 ХНУРЕ. Наукові інтереси: розробка програмного забезпечення, розробка архітектури систем. Адреса: Україна, м. Харків, 61166, пр. Науки, 14, тел. (057) 702 14 46.

**Хряпкін Олександр Володимирович**, кандидат технічних наук, доцент кафедри системотехніки ХНУРЕ. Наукові інтереси: технології проектування великомасштабних розподілених інформаційних систем, технології дистанційної освіти, моделі та методи автоматизованого проектування розподілених систем обробки даних, розпізнавання сигналів, бізнес аналітика. Адреса: Україна, м. Харків, 61166, пр. Науки, 14, тел. (057) 702 14 46.



*I.O. ЮР'ЄВ*

## РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ УПРАВЛІННЯ СИСТЕМОЮ НАДАННЯ ІТ-СЕРВІСІВ

Розглянуто основні особливості вирішення задачі визначення набору ІТ-сервісів, які відповідають заданому набору функціональних і нефункціональних обмежень. Розглянуто застосування для вирішення цієї задачі вдосконаленого методу вибору набору сервісів. Визначено основні особливості інформаційної технології управління системою надання ІТ-сервісів і виділені її основні стадії. Розроблено візуальні моделі інформаційної технології, основних функціональних задач і структур даних інструментального засобу, що реалізує дану технологію.

### 1. Вступ

Інформаційні системи (ІС), засновані на сервіс-орієнтованій архітектурі (СОА-ІС), за останні роки міцно зайняли значне місце серед ІС управління різними підприємствами. Однак досвід розробки та експлуатації СОА-ІС дозволяє виділити ряд науково-прикладних задач, які для СОА-ІС все ще не мають остаточного рішення. Серед цих задач особливо виділяються задачі вибору ІТ-сервісів, які відповідають висунутим до них вимогам. При цьому такі вимоги можуть бути як функціональними, так і нефункціональними, що ускладнює процедуру вибору ІТ-сервісу. Крім того, ці вимоги, як правило, виникають вже в ході експлуатації СОА-ІС за результатами аналізу "вузьких місць" існуючих ІТ-сервісів. Через це ІТ-сервіси, які знаходяться в складі експлуатованої СОА-ІС, можуть не повністю відповідати висунутим вимогам.

Подібні невідповідності між новими вимогами, які з'являються в ході експлуатації СОА-ІС, та ІТ-сервісами експлуатованої СОА-ІС призводять до необхідності вирішення задачі вибору з множини експлуатованих ІТ-сервісів такого ІТ-сервісу, який в максимально можливій мірі задовольняв би висунутим вимоги. Однак рішення цієї задачі в даний час серйозно утруднено практично повною відсутністю спеціалізованих інформаційних технологій (ІТ), що дозволяють автоматизувати дане рішення.

Необхідно відзначити, що теоретичним і прикладним дослідженням щодо рішення задачі вибору ІТ-сервісу, який в максимально можливій мірі задовольняв би висунутим вимогам, присвячений цілий ряд сучасних досліджень. Так, в [1] серед основних задач прийняття рішень щодо вибору ІТ-сервісів СОА-ІС виділяється задача визначення набору ІТ-сервісів, які відповідають заданому набору функціональних і нефункціональних обмежень.

Для її вирішення в [1] була розроблена модель якості ІТ-сервісів, здатна виражати як технічні, так і бізнес-аспекти якості, а також враховує точки зору програміста і кінцевого користувача. Дана модель була покладена в основу запропонованого в [1] узагальненого підходу до вирішення задачі. Однак застосування цього підходу викликає серйозні труднощі, оскільки він заснований на припущенні про існування апріорно заданої множини функціонально еквівалентних ІТ-сервісів. На практиці це припущення найчастіше не дотримується.

Серйозні недоліки притаманні також і дослідженням, присвяченим прикладним аспектам вирішення задачі визначення набору ІТ-сервісів, які відповідають заданому набору функціональних і нефункціональних обмежень. Найчастіше узагальнення прикладного досвіду вирішення даної задачі зводиться до формування наборів рекомендацій, використання яких в ряді випадків дозволило б адміністратору СОА-ІС обґрунтувати зроблений вибір. Приклади таких наборів рекомендацій наведені в [2, 3]. Однак подібний спосіб узагальнення прикладного досвіду вирішення задачі вибору ІТ-сервісів призводить до того, що вироблені рекомендації в ході застосування потребують адаптації до особливостей конкретної СОА-ІС.

Тому останнім часом значна увага приділяється дослідженню алгоритмів і процедур, що дозволяють автоматизувати рішення задач вибору ІТ-сервісів на основі сучасних наукових підходів. Однак точки зору дослідників на особливості цих алгоритмів і процедур сильно розрізняються. Так, до теперішнього часу як прикладні засоби вирішення задач вибору ІТ-сервісів пропонувалися, серед інших:

- а) алгоритми, що реалізують метод аналізу ієрархій [4];
- б) методи та алгоритми рішення задачі кластеризації і еволюційних обчислень [5];
- в) алгоритм глобальної оптимізації та динамічного перепланування [6];
- г) графові моделі варіантів композицій та алгоритм пошуку гармонії [7].

Однак всі ці алгоритми мають ряд недоліків, що ускладнюють їх застосування для вирішення задачі визначення набору ІТ-сервісів, які відповідають заданому набору функціональних і нефункціональних обмежень. Тому дослідження з розробки ІТ, що дозволяє автоматизувати рішення даної задачі, є актуальними.

## 2. Постановка задачі дослідження

Проведене дослідження засноване на представленні задачі вибору ІТ-сервісу для малих і середніх ІС із SOA як пошуку такого ІТ-сервісу  $ITs_j$ , який забезпечить реалізацію необхідних функціональних задач  $FZ = \{fz_j\}$ ,  $j=1, \dots, n$  відповідно до правил порівняння і вибору альтернатив  $R$ . Подібне представлення хоча і може трохи погіршити точність рішення задачі вибору ІТ-сервісу, але призводить до значного зниження обчислювальної складності способів вирішення задач за рахунок використання простіших в реалізації методів експертного оцінювання [8, 9].

Теоретичною базою дослідження є вдосконалений метод вибору набору сервісів для SOA-ІС. Даний метод описаний в [8] як сукупність наступних етапів.

Етап 1. Формування функціональних вимог до процесу або задачі прецеденту, для задоволення яких слід вибрати ІТ-сервіс.

Етап 2. Вибір з множини ІТ-сервісів  $ITs_{IS}$  підмножини ІТ-сервісів  $ITs_j^*$ , які відповідають функціональним вимогам до прецеденту  $n_j$ .

Етап 3. Уточнення множини критеріїв оцінювання ІТ-сервісів.

Етап 4. Вибір експертами множини правил порівняння  $R$ , на основі яких буде здійснюватися пошук ІТ-сервісу з підмножини  $ITs_j^*$ , який найкращим чином задовольняє заданому набору нефункціональних обмежень.

Етап 5. Оцінювання експертами ІТ-сервісів з підмножини  $ITs_j^*$  відповідно до прийнятого підмножиною правил  $R$ .

Етап 6. Формування результатів вибору ІТ-сервісу, який частково задовольняє заданому набору функціональних і нефункціональних обмежень. Завершення застосування методу.

Докладний зміст Етапів 1 і 2 розглянуто в [8]. Застосування методу аналізу ієрархій для реалізації Етапів 3-6 розглянуто в [9].

Метою даного дослідження є розробка ІТ управління системою надання ІТ-сервісів, що дозволила б автоматизувати рішення задачі визначення набору ІТ-сервісів, які відповідають заданому набору функціональних і нефункціональних обмежень на основі запропонованого в [8] вдосконаленого методу вибору набору сервісів.

Для досягнення даної мети в дослідженні передбачається рішення таких задач.

- визначення основних особливостей ІТ управління системою надання ІТ-сервісів;
- розробка основних візуальних моделей інструментальний засобу, що реалізує ІТ управління системою надання ІТ-сервісів.

## 3. Визначення основних особливостей інформаційної технології управління системою надання ІТ-сервісів

Рассмотрим основные типы шаблонов для построения логической сети.

Розроблювана ІТ управління системою надання ІТ-сервісів призначена для автоматизації процесу встановлення відповідності між запитами на зміну і ІТ-сервісами SOA-ІС. Це необхідно керівництву підприємства для того, щоб оцінити наслідки використання того чи іншого ІТ-сервісу і прийняти відповідне управлінське рішення. Тому пропонується виділити у цій ІТ такі стадії:

- Стадія 1 «Формування запиту на зміну функціональних вимог»;
- Стадія 2 «Пошук ІТ-сервісів в каталозі»;
- Стадія 3 «Оцінка і вибір ІТ-сервісів відповідно до функціональних вимог»;
- Стадія 4 «Оцінка впливу нового ІТ-сервісу на інфраструктуру підприємства».

Виконання Стадії 1 ІТ полягає в формуванні запиту на зміну (request for change, RFC) функціональних вимог. На основі функціональних вимог для RFC, отриманих з різних джерел, співробітник ІТ-служби формує новий RFC. Для цього він виділяє з набору різних текстових описів вимог множину функціональних вимог. Дані функціональні вимоги групуються в рамках RFC і реєструються шляхом заповнення відповідної форми.

Стадія 2 полягає в підборі ІТ-сервісів під функціональні задачі, визначені відповідно до RFC. Цей підбір здійснюється шляхом пошуку ІТ-сервісів, що реалізують необхідні функціональні задачі. Підготовчим етапом для цього є формування каталогу ІТ-сервісів. Після початку функціонування каталогу ІТ-сервісів з'являється можливість виконати підбір ІТ-сервісів під конкретний RFC. При цьому можливо прийняття проміжного управлінського рішення шляхом вибору однієї з наступних альтернатив:

- якщо в системі було знайдено кілька ІТ-сервісів під функціональну задачу, необхідно перейти на Стадію 4 для вибору одного з ІТ-сервісів;
- якщо в результаті пошуку не було знайдено жодного ІТ-сервісу, співробітнику ІТ-служби необхідно проаналізувати існуючі на ринку ІТ-сервіси, сформувані з них множину альтернативних ІТ-сервісів і перейти на Стадію 4;
- якщо в системі знайдений єдиний ІТ-сервіс під дані функціональні задачі, він повинен бути використаний для закриття даного RFC.

Стадія 3 використовується в ситуації, коли необхідно зробити вибір одного з декількох ІТ-сервісів. Для цього, використовуючи запропонований у [8] метод, з множини ІТ-сервісів, сформованих на стадії 2, і отриманих експертних оцінок програмно вибирається ІТ-сервіс, який найбільш задовольняє висунутим критеріям оцінки альтернатив. Після того, як в рамках системи надання ІТ-сервісів був визначений ІТ-сервіс, що задовольняє RFC, необхідно оцінити його вплив на існуючу інфраструктуру підприємства.

На Стадії 4 ІТ виконується оцінка стану експлуатованої інфраструктури підприємства на основі витрат і експертних оцінок. Значення отриманої оцінки зберігається в системі і використовується при формуванні рекомендацій щодо доцільності коригуючого впливу.

Спрощена IDEF0-модель ІТ управління системою надання ІТ-сервісів показана на рис. 1. У даній IDEF0-моделі не показано потоки механізмів, які описують інструментальний засіб, що реалізує заявлену ІТ.

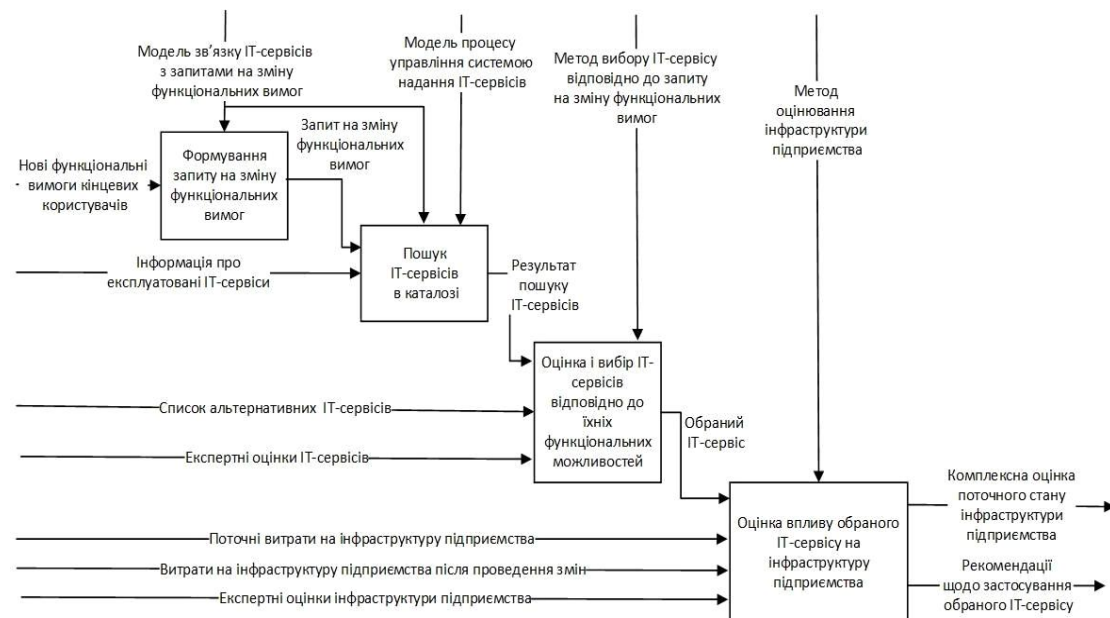


Рис. 1. Спрощена IDEF0-модель інформаційної технології управління системою надання ІТ-сервісів

Розроблювана ІТ призначена для автоматизації процесу встановлення відповідності між запитами на зміну і ІТ-сервісами інформаційної системи. Це необхідно керівництву підприє-

мства для того, щоб оцінити наслідки від використання того чи іншого ІТ-сервісу і прийняти відповідне управлінське рішення.

Основними користувачами ІТ управління системою надання ІТ-сервісів є співробітники ІТ-служби підприємства. В обов'язки співробітників ІТ-служби входить безпосередньо обробка заявок користувачів, формування RFC, управління ІТ-сервісами, а також оцінки впливу нового ІТ-сервісу на інфраструктуру підприємства.

Співробітники ІТ-служби передають результати роботи ІТ керівництву, яке безпосередньо приймає рішення про використання ІТ-сервісу. Представлена ІТ може бути використана на середніх і великих підприємствах у зв'язку з наявністю у таких підприємств великої кількості різних ІТ-сервісів і достатнього потоку RFC.

#### **4. Розробка основних візуальних моделей інструментального засобу, що реалізує інформаційну технологію управління системою надання ІТ-сервісів**

Для створення інструментального засобу, що повинен реалізувати запропоновану ІТ, пропонується згрупувати його основні функції у такі функціональні задачі:

- а) функціональна задача формування RFC;
- б) функціональна задача формування каталогу ІТ-сервісів;
- в) функціональна задача вибору ІТ-сервісів;
- г) функціональна задача оцінки інфраструктури підприємства.

Візуальну модель потоків даних функціональної задачі формування RFC наведено на рис. 2.

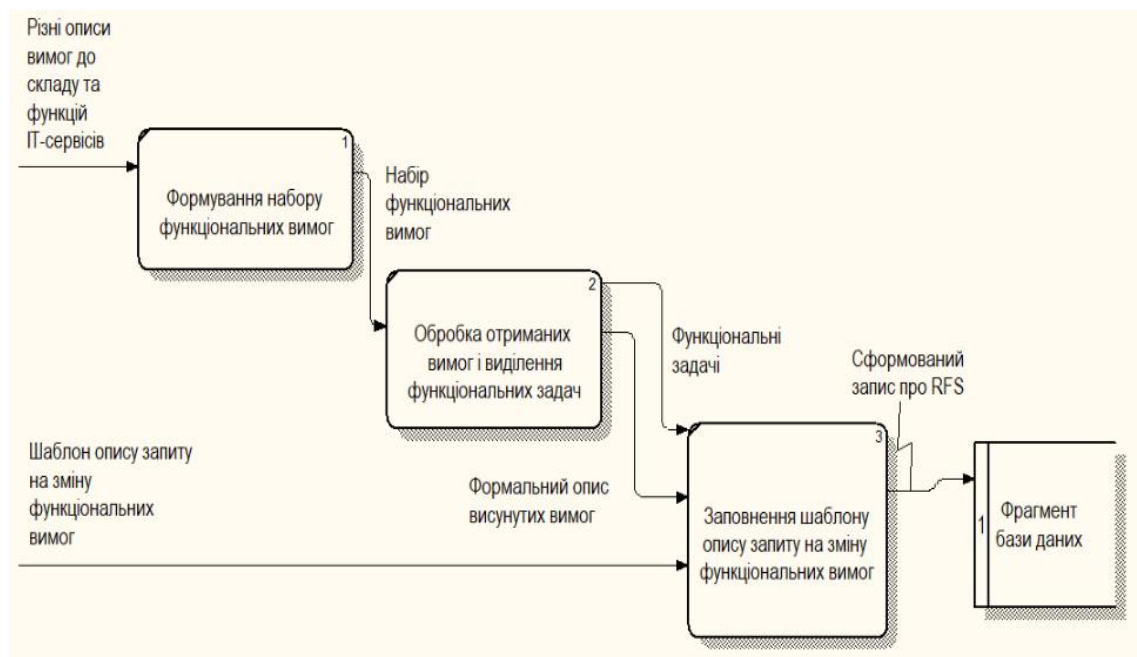


Рис. 2. Візуальна модель потоків даних функціональної задачі формування запитів на зміну

Візуальну модель потоків даних функціональної задачі формування каталогу ІТ-сервісів наведено на рис. 3.

Візуальну модель потоків даних функціональної задачі вибору ІТ-сервісів наведено на рис. 4.

Візуальну модель потоків даних функціональної задачі оцінки інфраструктури підприємства наведено на рис. 5.

Для розробки серверної частини програмного забезпечення інструментального засобу було використано мову PHP 7, а для розробки web-інтерфейсу - мови HTML, CSS, Javascript. Як СУБД була обрана MySQL. Фрагмент діаграми класів програмного забезпечення інструментального засобу представлений на рис. 6.

#### **5. Висновки і перспективи подальших досліджень**

Запропонована ІТ визначає основні стадії вирішення задачі визначення набору ІТ-сервісів, які відповідають заданому набору функціональних і нефункціональних обмежень на основі

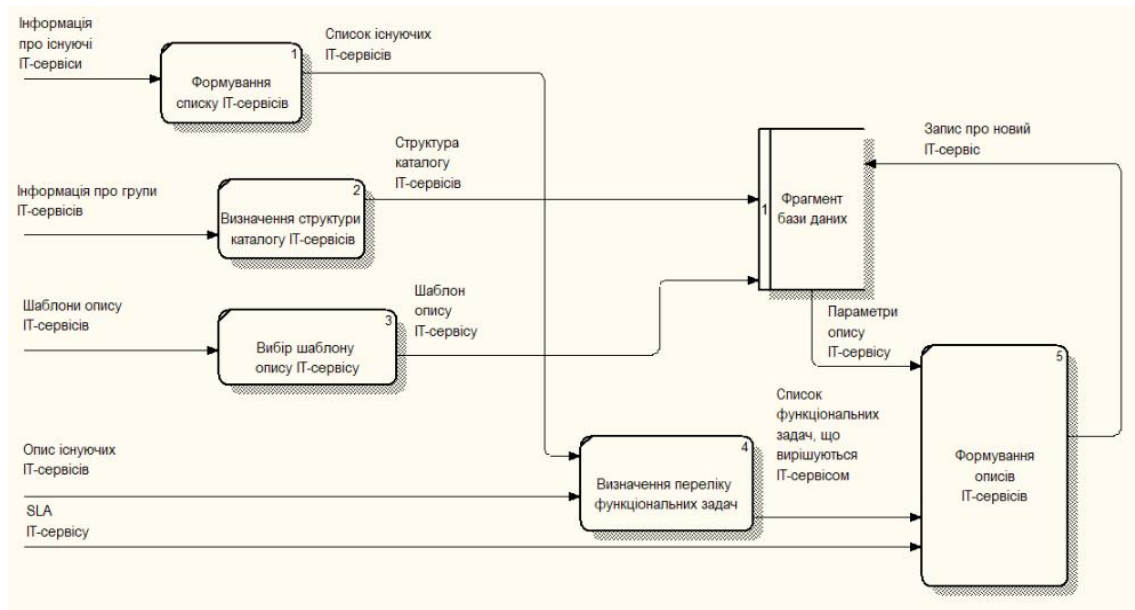


Рис. 3. Візуальна модель потоків даних функціональної задачі формування каталогу ІТ-сервісів

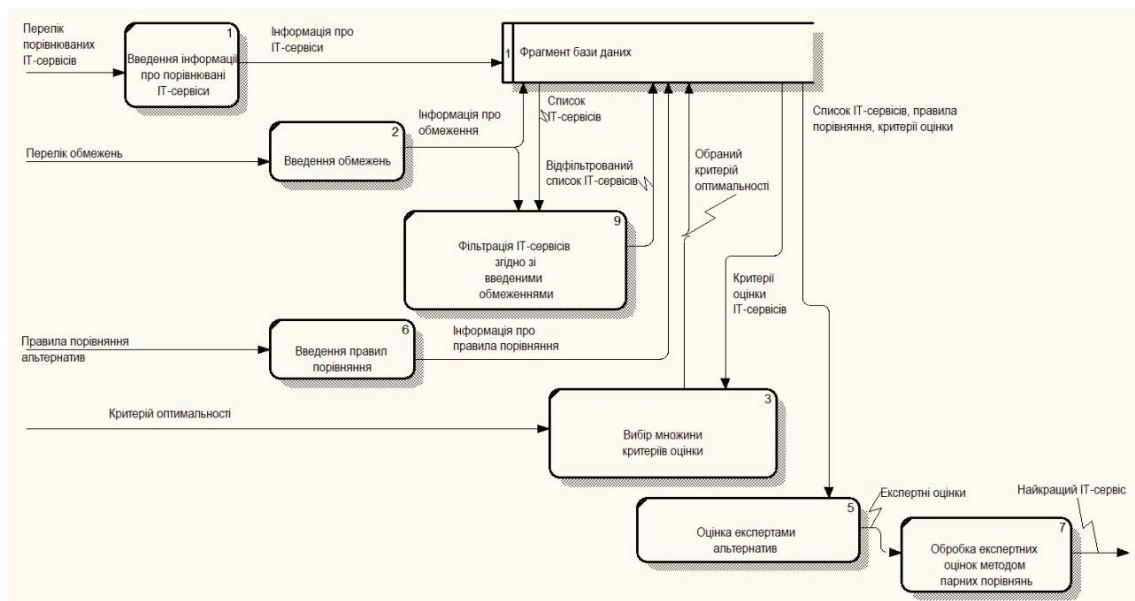


Рис. 4. Візуальна модель потоків даних функціональної задачі вибору ІТ-сервісів

запропонованого в [8] вдосконаленого методу вибору набору сервісів. Для формалізованого опису цих стадій та їх взаємозв'язків в ході виконання ІТ була розроблена IDEF0-модель.

Для реалізації запропонованої ІТ у вигляді спеціалізованого інструментального засобу були визначені основні функціональні задачі цього засобу. Була також розроблена діаграма класів, що описує основні структури даних цього інструментального засобу. Формалізовані описи функціональних задач і діаграми класів представлені у вигляді відповідних візуальних моделей.

Перспективою подальших досліджень, що дозволяють удосконалити отримані результати, є розробка та удосконалення алгоритмів обробки даних, які повинні забезпечити скорочення обчислювальних і тимчасових витрат на рішення задачі визначення набору ІТ-сервісів, що відповідають заданому набору функціональних і нефункціональних обмежень.

**Список літератури:** 1. *Reiff-Marganiec S., Tilly M.* (Eds.). Handbook of Research on Service-Oriented Systems and Non-Functional Properties: Future Directions. Hershey: IGI Global, 2012. 521 p. DOI: 10.4018/978-1-61350-432-1. 2. *Top 10 criteria for selecting a managed services provider. How cloud and managed*

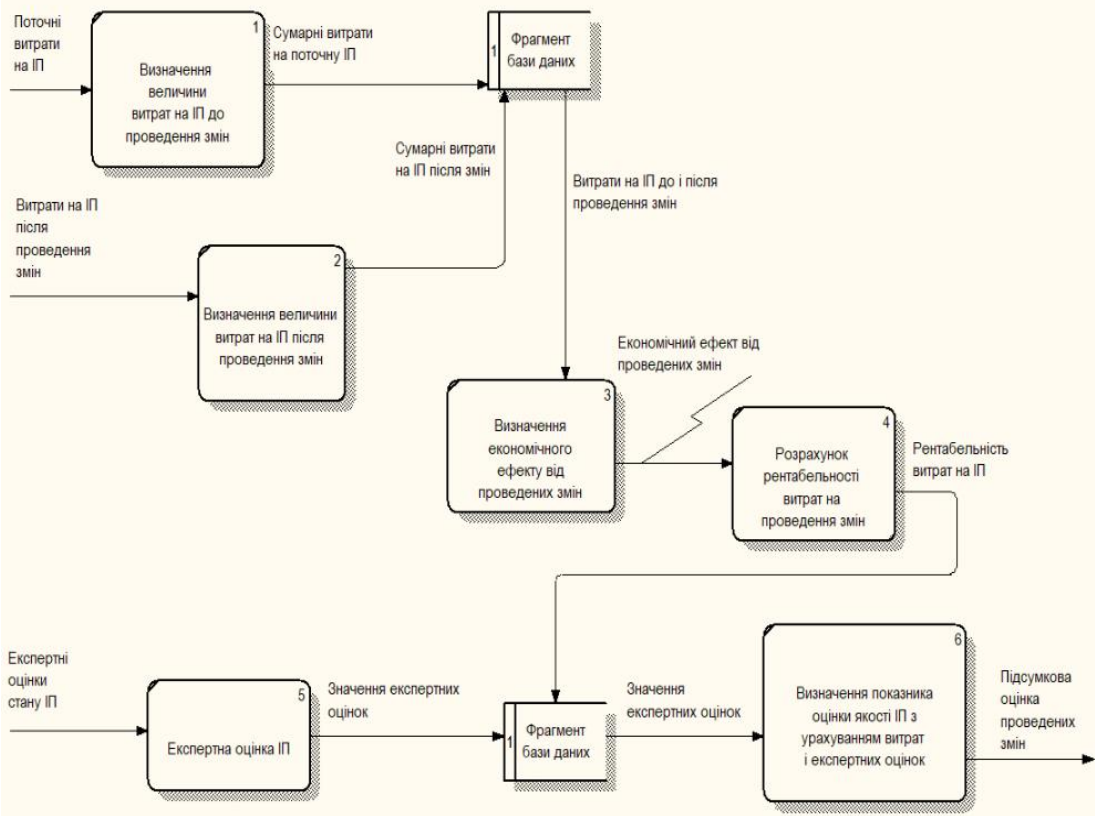


Рис. 5. Візуальна модель потоків даних функціональної задачі оцінки інфраструктури підприємства

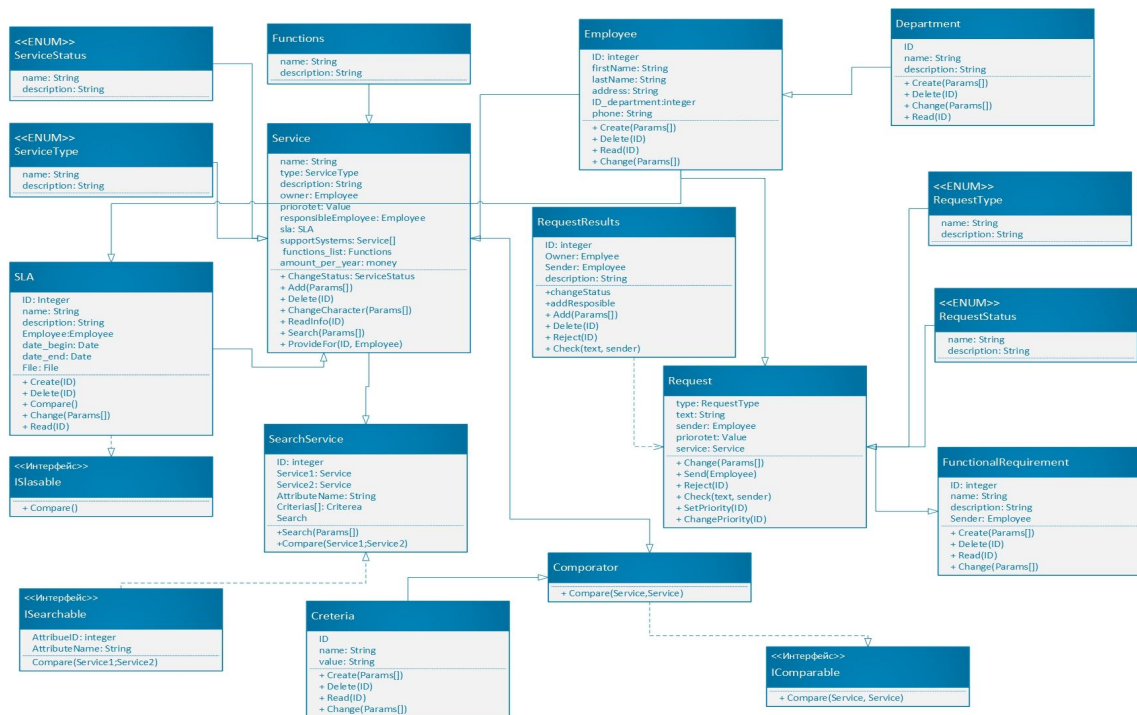


Рис. 5. Фрагмент діаграми класів програмного забезпечення інструментального засобу, що повинен реалізувати інформаційну технологію управління системою надання ІТ-сервісів

services help IT deliver business value [White Paper] / IBM Global Technology Services. IBM Corporation, 2017. URL: <https://www.ibm.com/downloads/cas/5GKMXAYZ> (дата звернення 12.02.2021 р.). 3. *Nokes C.* 9 Steps for Selecting the Right IT Services Provider // Сайт "ITSM tools". URL: <https://itsm.tools/9-steps-for-selecting-the-right-it-service-provider/> (дата звернення 12.02.2021 р.). 4. *Viriyasitavat W.* Multi-criteria selection for services selection in service workflow // *Journal of Industrial Information Integration*. 2016. Vol. 1. P. 20-25. DOI: 10.1016/j.jii.2016.03.003 5. *Wang C., Ma H., Chen G.* "Using EDA-Based Local Search to Improve the Performance of NSGA-II for Multiobjective Semantic Web Service Composition", In: Hartmann S., K?ng J., Chakravarthy S., Anderst-Kotsis G., Tjoa A., Khalil I. (eds) *Database and Expert Systems Applications. DEXA 2019. Lecture Notes in Computer Science*, vol 11707. Springer, Cham., 2019. P. 434-451. DOI: [https://doi.org/10.1007/978-3-030-27618-8\\_32](https://doi.org/10.1007/978-3-030-27618-8_32). 6. *Zhang N.* Service Discovery and Selection Based on Dynamic QoS in the Internet of Things // *Complexity*. 2021. Vol. 2021. Article ID 6642514. DOI: <https://doi.org/10.1155/2021/6642514>. 7. *Bekkouche A., Benslimane S.M., Huchard M., Tibermacine C., Hadjila F., Merzoug M.* QoS-Aware Optimal and Automated Semantic WebService Composition With User's Constraints // *Service Oriented Computing and Application*. 2017. Vol. 11. P. 183-201. DOI: 10.1007/s11761-017-0205-1. 8. *Petrychenko A., Levykin I., Iuriev I.* Improving a method for selecting information technology services // *Eastern-European Journal of Enterprise Technologies*. 2021. Vol. 2. No. 2 (110). P. 32-43. DOI: 10.15587/1729-4061.2021.2299. 9. *Левыкин В.М., Юрьев И.А.* Модель выбора набора ИТ-сервисов для конечных пользователей // *Вісник НТУ "ХПІ"*. Серія: Системний аналіз, управління та інформаційні технології. Х.: НТУ "ХПІ", 2016. №45 (1217). С. 78-84.

Надійшла до редколегії 17.06.2021

**Юр'єв Іван Олексійович**, старший викладач кафедри інформаційних управляючих систем ХНУРЕ. Наукові інтереси: методи, моделі, інформаційні технології управління інформаційними системами. Адреса: Україна, 61166, м. Харків, пр. Леніна, 14, тел. (057) 702 14 51.

---

УДК 004.652.4

DOI: 10.30837/0135-1710.2021.177.063

*Т. Г. БІЛОВА, В. М. ДЬОМІНА, І. О. ПОБІЖЕНКО*

## КОМПОЗИЦІЙНЕ ПРОЕКТУВАННЯ ТА РЕСТРУКТУРИЗАЦІЯ ГЛОБАЛЬНОЇ СХЕМИ МУЛЬТИБАЗИ ДАНИХ

---

Розглянуто особливості створення глобального шару над даними в розподілених базах даних. На основі аналізу варіантів представлення глобальних схем запропоновано поєднання централізованого та децентралізованого підходів. Визначено основні задачі композиційного проектування та реструктуризації глобальної схеми мультибази даних. Формалізовано процеси додавання та видалення локальних вузлів, різних варіантів оновлення з підтримкою обмежень цілісності.

### 1. Вступ

Проблема інтегрування великих обсягів даних визначає вимоги до представлення їх у вигляді, зручному для використання та перетворення. Користувачу розподіленої бази даних (РБД) надається інформація про дані у вигляді глобальної схеми. Вибір моделі подання такої схеми базується на аналізі вимог користувачів кінцевої системи та не має залежати від типу БД, набору прикладних програм, які створені або будуть створені, мов програмування, що використовуються, самої обчислювальної платформи тощо. Модель представлення даних, отримана в результаті концептуального проектування, є джерелом інформації для фази логічного проектування БД.

Побудова глобальної схеми на основі традиційної реляційної моделі дає можливість використовувати для побудови запитів апарат реляційної алгебри. Недоліком такого підходу є складність приведення моделей нереляційних баз даних до реляційної схеми.

Питання розробки загального підходу для представлення глобального рівня даних досліджуються в роботах багатьох науковців. Вибір графової моделі [1-3] дозволяє представити відношення між розподіленими даними у вигляді неорієнтованого графа, що визначає зв'язки між даними, розподіленими по різних вузлах системи. Такий підхід дозволяє створити єдиний інтерфейс доступу до усіх даних. Процес управління графовою моделлю здійснюється за допомогою операцій об'єднання, перетину та алгоритмів обходу у глибину [1]. Іншим підхо-

дом є рекурсивне представлення [4] у вигляді багаторівневої інтеграції гетерогенних джерел даних у єдиний інформаційний простір. Перевагою такого підходу є можливість динамічної зміни схеми даних та перспективне масштабування системи за рахунок приєднання нових моделей.

Недослідженими залишаються питання виникнення конфліктів в процесі проведення реструктуризації глобального рівня у разі зміни інформації на одному з вузлів. Актуальною є задача розробки єдиного підходу до проведення проектування та реструктуризації глобальної схеми в реальному часі з урахуванням обмежень цілісності та несуперечливості даних, що оброблюються, та забезпеченням безперервного доступу користувачів до розподіленої системи.

Метою дослідження є визначення представлення глобального шару даних в розподіленій системі таким чином, щоб забезпечити ефективний процес її подальшої реструктуризації.

В рамках поставленої мети потрібно вирішити наступні задачі: проаналізувати можливі варіанти представлення глобального рівню управління даними в РБД; формалізувати задачу проектування глобальної схеми РБД; визначити підтримку процесів управління глобальною схемою з урахуванням вимог цілісності та доступності інформації для користувачів; розробити діаграму станів процесу реструктуризації глобальної схеми.

## **2. Аналіз варіантів представлення глобальної схеми та постановка задачі проектування**

Розподілена система інтегрує різноманітні ресурси у єдине інформаційне середовище. Проектування такої системи приходиться виконувати методом «знизу до гори», тобто шукати механізми об'єднання вже існуючих баз даних таким чином, щоб представити користувачеві можливість доступу до ресурсів у той же спосіб, як і до централізованої системи. Класичне визначення за Дейтом [6] дванадцяти принципів побудови РБД потребує повної автономності кожного локального вузла, але на практиці досягнути цього майже неможливо.

Для забезпечення сумісної роботи в РБД використовуються наступні засоби:

- фрагментація - розбиття БД або таблиці на кілька частин і зберігання цих частин на різних вузлах РБД;
- реплікація - створення і зберігання копій одних і тих же даних на різних вузлах РБД;
- розподілені обмеження цілісності - обмеження, для перевірки виконання яких потрібно звернення до іншого вузла РБД;
- розподілені запити - запити на читання, які звертаються більш ніж до одного вузла РБД;
- розподілені транзакції - команди на зміну даних, які звертаються більш ніж до одного вузла РБД.

Одним з рішень при побудові розподілених систем є створення мультибази даних. Під мультибазою даних з глобальною схемою розуміється реалізація зовнішнього інтерфейсу доступу до множини локальних СУБД, тобто створення глобального рівню над локальними СУБД.

Мультибаза даних є розподіленою гетерогенною системою, до її характеристик слід віднести неоднорідність у представленні та використанні даних, різні схеми представлення даних та різна структура компонентів, різні інтерфейси та протоколи використання ресурсів та ін. Все це повинно бути приховано від користувача, який працює з розподіленою системою в той самий спосіб, що й з централізованою. Додатково може виникати потреба в обробці інших типів даних, що не є структурованими та надходять з різних джерел.

Проектування мультибазової РБД базується на способах та методах її використання, тобто на стратегії формування запитів, що, в свою чергу, визначає схеми фрагментації та реплікації даних. Вимоги до високої продуктивності розподілених баз даних потребують ефективних процедур оновлення схеми даних. В разі зростання кількості вузлів процес оновлення глобальної схеми може негативно вплинути на середній час виконання розподілених запитів.

Глобальна схема мультибази даних визначається як загальне представлення про дані на локальних вузлах та способи доступу до них. Для її підтримки необхідно реалізувати два етапи: проектування та реструктуризації.



Сформулюємо загальну задачу композиційного проектування глобальної схеми бази даних як процес визначення загальної концепції проектування, аналіз компонентів глобальної схеми та визначення і пошук варіантів оптимального розміщення компонентів.

Порівняльний аналіз різних видів представлення глобальної схеми подано у таблиці 1.

Таблиця 1-

Порівняльний аналіз варіантів представлення глобальної схеми

Глобальна схема	Переваги	Недоліки
Централізована – розміщення на спеціальному виділеному вузлі	– централізоване представлення даних; – швидкість оновлення; – єдині правила оновлення та підтримки цілісності даних	– порушення принципів локальної автономності; – покладання функцій управління на виділений центральний вузол
Реплікована – кожен вузол зберігає власну копію глобальної схеми	– швидкість виконання запитів; – підтримують гомогенні РБД; – єдині правила оновлення та підтримки цілісності даних	– необхідність підтримки усіх реплік в актуальному стані; – надмірні витрати пам'яті та часу на реструктуризацію
Секціонована – на кожному вузлі зберігаються схеми тільки тих вузлів, з яких може бути затребувана інформація	– децентралізація управління даними; – локальна автономія кожного з вузлів; – підтримують гетерогенні РБД	– складність процедур оновлення даних; – відсутність єдиного представлення даних
Секціонована з централізованою реплікацією – на кожному вузлі зберігається його локальна схема, а на центральному вузлі зберігаються репліки локальних схем	– поєднання централізованого та децентралізованого підходів; – підтримують гетерогенні РБД; – є можливість виконання розподілених запитів, які попередньо не були заплановані	– порушення принципів локальної автономності; – відсутність єдиного представлення даних; – покладання функцій управління на виділений центральний вузол

Як видно з табл. 1, кожне представлення має свої переваги та недоліки. Підвищити ефективність представлення глобальної схеми можна за рахунок знаходження компромісу між централізованим та децентралізованим розміщенням глобального шару даних.

Розглянемо постановку задачі управління глобальною схемою у випадку централізованого зберігання на виділеному вузлі з підтримкою для підвищення надійності реплікованих копій на  $n$  спеціально виділених «резервних» вузлах, число яких пропорційно кількості вузлів РБД.

В загальному вигляді проектування глобальної схеми  $S$  можна представити як ітераційний процес додавання нових вузлів, тобто як реструктуризацію. Тоді задачу додавання локальної схеми з нового вузла в глобальну схему можна представити наступним чином:

а) перевірка схеми на пустоту; якщо схема пуста, то схему локального вузла представляємо як глобальну;

б) якщо схема не пуста, то виконуємо процедуру додавання нового вузла.

Задача реструктуризації глобальної схеми полягає у підтримці основних операцій маніпулювання: оновлення; модифікації, розширення шляхом додавання нових локальних схем.

### 3. Методи проектування та реструктуризації глобальної схеми

Розглянемо глобальну схему бази даних як структуру, що визначає множину об'єктів - локальних схем БД та операцій над ними. Визначимо необхідні операції для визначеного варіанту представлення глобальної схеми.

Для формалізації процедури додавання нового вузла представимо управління процесом реструктуризації у вигляді трійки параметрів  $Z = \langle S, E, \Phi \rangle$ , де  $S$  - глобальна схема, що є результатом об'єднання множини логічних схем БД локальних вузлів;  $E$  - множина схем

сутностей для роботи з даними;  $\Phi$  - множина відображень схеми представлень локальних вузлів на схеми сутностей

$$\Phi = \{ \phi : S \rightarrow \Xi \mid \Xi \in E \}, \quad (1)$$

де  $\Xi$  - елемент множини відображень схеми представлень локальних вузлів на схеми сутностей,  $\phi \in \Phi$ .

Представлення (1) дозволяє інтегрувати різномірні локальні схеми в єдину, центральну схему для реалізації єдиного механізму управління даними, незалежно від моделей даних та схем локальних даних окремих вузлів.

Даний спосіб представлення дозволяє моделювати схему зв'язку зі схемами даних локальних вузлів, зводячи їх до все простіших елементів. Кожне відношення глобальної схеми даних представляється у вигляді набору трійок: набір атрибутів, клас розбиття домену та ідентифікатор локального вузла, на якому вони розміщені. Крім цього, глобальна схема повинна містити зв'язки між відношеннями. А відношення в свою чергу містять набір атрибутів, домени яких визначають розміщення даних в вузлах бази даних.

Дані будь-якого вузла поміщаються в атрибути трійок. Кожному локальному вузлу ставиться у відповідність набір зв'язків з іншими вузлами, набір обмежень цілісності і набір допустимих операцій. Кожен вузол позначимо  $S_{k,e}^i$

$$S_{k,e}^i = \left\langle \left\langle \left\langle a_j^i, d_k^i, s_{k,e}^i \right\rangle \mid l \in L^i, L^i < L, k \in K^i \right\rangle, \Lambda_j^i, P_m^i(s_{k,e}^i), O_q^i \right\rangle,$$

де  $j, m, q \in [1, \theta], \theta \in \mathbb{R}, N : [1, \theta] \rightarrow \mathbb{R} \triangleright \triangleleft \Omega$ .

Кожен вузол позначається трьома індексами:  $i$  - номер батьківського локального джерела даних (вузла);  $k$  - індекс відповідного класу розбиття домену на непересічні класи ( $K = \{K_1^i, K_2^i, K_3^i, \dots, K_k^i\}$ );  $e$  - грає роль унікального ключа. Індеси  $j, e, m, q$  - номери джерел локальних даних;  $R$  - максимальна кількість вузлів;  $a_j^i$  - атрибути елементів даних (призначені для зберігання даних);  $\Omega$  - множина перспективних локальних вузлів, з якими можуть бути встановлені зв'язки,  $L$  - максимальна кількість атрибутів.

Множину зв'язків даного вузла позначимо  $\Lambda_k^i, \Lambda_k^i = \left\langle \left\langle \left\langle s_{k,e}^i, \gamma \mid \gamma \in \Gamma, \mu \right\rangle \right\rangle \right\rangle$ , де  $\Gamma$  - множина атрибутів зв'язків;  $\mu$  - назва зв'язку; індекс  $e$  грає роль унікального ключа (у всіх вузлів в схемі даний індекс унікальний).

Множину предикатів, які задають обмеження цілісності на множині схем локальних вузлів, позначимо як  $P_j^i(S)$ . Множина  $S$  може містити схеми даних локальних вузлів  $S_r$ , тобто її можна представити як вектор  $S = \langle S_1, S_2, \dots, S_r \rangle, r \in \mathbb{R}$ .

Схема локального вузла  $s_{k,e}^i$  знаходиться в узгодженому стані, якщо параметри локальної схеми з локального вузла додані до глобальної схеми та задані все предикати з множини предикатів, які задають обмеження цілісності для даного вузла. Таким чином,  $s_{k,e}^i$  - в узгодженому стані, якщо  $\forall s : s \in S \cup \{s_{k,e}^i\}$  виконується умова  $\forall \phi : \phi \in P(s), \phi(s) = \text{true}$ , де  $s$  - довільна схема локального вузла;  $\phi(s)$  - деякий предикат,  $P(s)$  - множина предикатів, які задають обмеження цілісності для локального вузла  $s$ .

Обмеження на діапазон допустимих значень представляються як  $\Xi = a < A < b$ ,  $a, b \in \Xi$ , де  $A$  - атрибут локального вузла;  $\Xi$  - множина, в якій беруть значення атрибути локального вузла. Обмеження посилянь:  $\Xi(s^x, s^y) = \forall s^x \exists s^y (s^x \rightarrow s^y)$ . Агреговані обмеження:  $\Xi = \max(A) > a$ ,  $a \in \Xi$ , де  $A$  - атрибут локального вузла.

Множину операцій для даного локального вузла позначимо  $O_k^i$  :

$$O_k^i = \{ \varphi : S_{k,e}^i \rightarrow S_{k,e}^i \}, k \in K^i .$$

де  $s_{k,e}^i$  - множина схем даних для  $i$ -го локального вузла, що містить домен  $K^i$ . Всі операції з глобальною схемою даних можуть бути розділені на чотири типи: вибірка; оновлення; видалення; додавання.

Нехай  $s^x$  - схема локального вузла, що додається. Операція додавання для схеми локального вузла  $s_{k,e}^i$  має вигляд

$$\phi : S_k^i \rightarrow S_k^i, k \in K^i, S_k^i = S_k^i \cup \{s^x\} .$$

Нехай  $s^x$  - схема з локального вузла, що видаляється. Операція видалення для схеми локального вузла  $s_{k,e}^i$  має вигляд

$$\phi : S_k^i \rightarrow S_k^i, k \in K^i, S_k^i = S_k^i \setminus \{s^x\} .$$

Операція поновлення для схеми локального вузла  $s_{k,e}^i$  має вигляд

$$\phi : S_k^i \rightarrow S_k^i, k \in K^i, S_k^i = S_k^i .$$

У процесі оновлення у локальній схемі можуть змінюватися атрибути, обмеження цілісності, набір зв'язків, набір операцій. Операції додавання і видалення схем локальних вузлів, що знаходяться в узгодженому стані, зберігають узгодженим стан схеми локального вузла, для якого ці операції виконувалися.

Розглянемо два типи зв'язків між локальними схемами різних вузлів системи: структурні і асоціативні. Структурні зв'язки є неявними і обумовлені структурою схеми даних.

Асоціативні зв'язки дозволяють зв'язати схеми локальних вузлів, які не пов'язані безпосередніми структурними зв'язками. Асоціативні зв'язки з'єднують локальні вузли неявно, через проміжний вузол. Якщо розглядати структурні зв'язки як вертикальні, то асоціативні можна розглядати як горизонтальні.

Кожен зв'язок характеризується атрибутом, який характеризує дії, спрямовані на підтримку посилальної цілісності під час виконання операцій видалення над пов'язаними локальними вузлами. Атрибут може приймати одне з наступних значень:

- каскадне видалення: при видаленні схеми локального вузла  $s$ , на який посилається схема локального вузла  $t$ , схема локального вузла  $t$  також буде видаленою;

- заборона видалення: якщо при видаленні схеми локального вузла  $s$  існує схема локального вузла  $t$ , який посилається на локальний вузол  $s$ , то операція видалення не буде дозволена; щоб її здійснити, необхідно або спочатку видалити схему локального вузлу  $t$ , або перенаправити зв'язок від нього на інший вузол;

- видалення зв'язку: при видаленні схеми локального вузла даних  $s$  видаляється зв'язок, котрий посилається на нього від деякої схеми локального вузла  $t$ ; при цьому схема локального вузла  $t$  продовжує існувати незалежно.

Характер зв'язків визначає процедуру оновлення глобальної схеми, яка повинна бути виконана. Структурні зв'язки вимагають оновлення одного вузла, асоціативні зв'язки передбачають більш складне каскадне оновлення, яке передбачає формування глобального плану, що визначає порядок взаємодії вузлів.

В загальному вигляді реструктуризацію глобальної схеми можна визначити як циклічний безперервний процес, що містить наступні етапи:

- стан додавання нового вузла;
- стан моніторингу;
- стан оновлення, що охоплює один локальний вузол;
- стан каскадного оновлення, що охоплює декілька локальних вузлів;

- стан оновлення резервних копій глобальної схеми (ГС).

На рис. 1 представлена діаграма переходів станів процесу реструктуризації глобальної схеми.



Рис. 1. Діаграма переходів станів процесу реструктуризації глобальної схеми

#### 4. Висновки

Розроблений підхід до проектування та реструктуризації глобальної схеми дозволяє виконувати операції над метаданими з підтримкою цілісності та надійності роботи системи. Поєднання централізованого та децентралізованого представлення глобальної схеми надає можливість ефективно проводити процес реструктуризації в режимі обслуговування розподілених запитів. Розроблена діаграма станів реструктуризації відображує безперервний процес функціонування РБД з контролем актуального стану глобального шару опису даних.

Подальшого розвитку потребує вирішення проблеми децентралізації управління глобальною схемою. Питання управління частковою глобальною схемою, коли на кожному вузлі зберігаються схеми тільки тих вузлів, з яких може бути затребувана інформація, будуть розглядатися в наступних дослідженнях.

**Список літератури:** 1. *Афанасьев В.В., Лебедеко Е.В.* Графоаналитическая модель процесса формирования глобальной схемы мультибазы данных с учетом этапов ее реструктуризации // Информационные системы и технологии. 2014. № 1 (81). С 12-18. 2. *Білова Т.Г.* Проектування розподіленої бази даних системи надання електро-них адміністративних послуг // АСУ та прибори автоматики. Харків, 2019. Вип. 176. С. 49-54. № 175. С. 49-53. 3. *Лаврентьев К.А., Пономарчук Ю.В., Фалеева Е.В.* Применение ориентированных графов для построения схемы гетерогенной распределенной базы данных уровня предприятия // Фундаментальные исследования. 2015. № 6 (ч. 2). С. 257-262. 4. *Кашиников А, Лядова Л.* Интеграция гетерогенных источников данных на основе рекурсивной декомпозиции // International Journal "Information Technologies & Knowledge". 2011. Number 3, Vol.5. P. 274-284. 5. *Дейт К. Дж.* Введение в системы баз данных = An Introduction to Database Systems. 7-е изд. "Вильямс", 2001. 1072 с.

*Надійшла до редколегії 18.06.2021*

**Білова Тетяна Георгіївна**, кандидат технічних наук, доцент, доцент кафедри системотехніки ХНУРЕ. Наукові інтереси: моделювання бізнес-процесів, хмарні технології, розподілені бази даних. Адреса: Україна, м. Харків, пр. Науки 14, тел. (057) 702 14 51.

**Дьоміна Вікторія Михайлівна**, кандидат технічних наук, доцент, доцент кафедри інформаційних технологій, консалтинг і туризму Харківського національного аграрного університету ім. В. В. Докучаєва. Наукові інтереси: моделювання процесів стратегічного управління регіоном, математично-статистичні методи і моделі. Адреса: Україна, Харківська обл., селище Докучаєвське, уч. міс. ХНАУ, тел. (057) 709 03 01.

**Побіженко Ірина Олександрівна**, кандидат технічних наук, доцент, доцент кафедри інформаційних технологій Харківської державної академії культури. Наукові інтереси: моделювання бізнес-процесів, хмарні технології, розподілені бази даних, бізнес аналіз. Адреса: м. Харків, Бурсацький узвіз, 4, тел. (057) 731 32 82.

*С.Ф.ЧАЛИЙ, В.О. ЛЕЩИНСЬКИЙ, І.О. ЛЕЩИНСЬКА*

## **ПРЕДСТАВЛЕННЯ УЗГОДЖЕНИХ ЗНАНЬ З УРАХУВАННЯМ ЇХ ЛОГІЧНОЇ НЕСУПЕРЕЧЛИВОСТІ ДЛЯ ЗАДАЧІ ПОБУДОВИ ПОЯСНЕНЬ В ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ**

Розроблено узагальнений підхід до узгодження знань для побудови пояснень в інтелектуальних інформаційних системах. Підхід забезпечує можливість формування множини узгоджених пояснень, що відображають різні аспекти процесу прийняття рішення та отриманого інтелектуальною системою результату, що спрощує застосування цих результатів у предметній області користувача. Запропоновано модель представлення узгоджених знань з урахуванням їх логічної несуперечливості. Модель дає можливість перевірити узгодженість знань безпосередньо при формуванні пояснень, що створює умови для формування пояснень за принципом чорного ящика та дає можливість доповнити функціонуючі системи можливостями пояснень, а також адаптувати пояснення при зміні їх функціональності.

### **1. Вступ**

Потреба у ефективному використанні результатів роботи інформаційних систем при вирішенні погано формалізованих задач обумовлює важливість використання пояснень щодо процесу прийняття рішення в таких системах [1]. Формування пояснень є вельми актуальним для інтелектуальних інформаційних систем, які широко використовують методи машинного навчання, що не є «прозорими» для користувача. Непрозорість алгоритмів прийняття рішення призводить до виникнення проблеми практичного застосування отриманих у таких системах результатів. Дана проблема виникає внаслідок протиріччя між можливостями побудови ефективних рішень з використанням складних, «непрозорих» алгоритмів та обмеженими можливостями розуміння користувачем каузальних зв'язків, що визначають та обґрунтовують процес прийняття таких рішень. Вказане протиріччя знижує довіру користувача до отриманого рішення й ускладнює застосування отриманих результатів для вирішення практичних задач у предметній області користувача.

З метою вирішення вказаної проблеми і підвищення довіри користувача до результатів роботи інтелектуальних систем запропоновано концепцію інтеграції пояснень у процес роботи таких систем - XAI (Explainable Artificial Intelligence) [2]. Згідно з даною концепцією, пояснення представляє собою раціональне тлумачення результату роботи інтелектуальної системи та процесу його досягнення. Пояснення має обґрунтовувати коректність та відповідність рішення інформаційної системи завданням користувача. Для вирішення цієї задачі пояснення має відображати ключові причинно-наслідкові зв'язки, що привели до отриманого рішення. Сукупність таких каузальних залежностей дає можливість користувачеві не лише розуміти причини отриманого рішення з урахуванням властивостей вхідного набору даних, але й порівняти це рішення з його альтернативами з тим, щоб вибрати найкращий результат. Використання обмеженої кількості варіантів рішення з можливістю їх порівняння спрощує користувачеві вибір та обґрунтування отриманого результату [3].

Для побудови пояснень використовуються два альтернативних підходи: інтеграція пояснень у процес прийняття рішення на етапі проектування відповідної інформаційної системи або ж доповнення існуючої системи модулем пояснення [4-6]. В першому випадку використовуються ті ж множини вхідних даних та знань, що були використані для побудови рішення в інформаційній системі. Даний підхід дає можливість детально пояснити процес прийняття рішення, однак модуль пояснення потрібно передбачити при створенні інтелектуальної інформаційної системи. Необхідність удосконалення даного модулю може потребувати перебудови інформаційної системи в цілому. Другий підхід передбачає доповнення існуючих систем модулем пояснення. В даному випадку може бути використана підмножина вхідних даних, що використовується для прийняття рішення, а також додаткові дані, що відображають особливості предметної області та ключові події процесу прийняття рішення. Останній підхід забезпечує більшу гнучкість та можливість удосконалення пояснень у

порівнянні з першим підходом, однак при його використанні виникає проблема узгодження знань. Концепція узгодження знань була запропонована в роботах [7-9]. Властивості узгоджених знань з урахуванням ймовірнісного аспекту розглянуто в роботах [10, 11]. Підхід до узгодження пояснення із знаннями, що використовуються для прийняття рішення в інтелектуальній системі, запропоновано в [12]. Сутність узгодження знань полягає у визначенні підмножини таких знань, які б не мали протиріч та аномалій при використанні альтернативних варіантів опису предметної області. Ключова перевага узгодження знань полягає у можливості використовувати неповні та неточні знання. Це дає можливість сформулювати пояснення щодо результату роботи інформаційної системи, який відповідає знанням користувача щодо предметної області.

Однак існуючі підходи до узгодження знань орієнтовані на статичний, фіксований опис знань. В той же час при побудові пояснення необхідно враховувати можливості уточнення та поповнення знань, що використовуються при виконанні процесу прийняття рішення у інформаційній системі. Усунення даного недоліку потребує розробки та реалізації узагальненого підходу до узгодження знань при вирішенні задачі побудови пояснень в інтелектуальних інформаційних системах.

## **2. Мета і задачі дослідження**

Метою роботи є розробка моделі узгоджених знань щодо процесу прийняття рішення в інтелектуальній системі, яка враховувала б їх логічну несуперечливість та забезпечувала можливість формування пояснення процесу та результату роботи такої системи.

Досягнення мети дослідження потребує вирішення таких задач:

- розробка узагальненого підходу до узгодження знань для побудови пояснень в інтелектуальних інформаційних системах;
- розробка моделі представлення узгоджених знань з урахуванням їх логічної несуперечливості.

## **3. Розробка узагальненого підходу до узгодження знань для побудови пояснень в інтелектуальних інформаційних системах**

Визначимо ключові властивості узгоджених даних та знань, що використовуються для побудови пояснень, з урахуванням представленого у роботі [10] опису загальних властивостей структурованих знань. Згідно з даним підходом, система структурованих знань є узгодженою у тому випадку, коли кожне судження є наслідком решти знань, а також спричиняє інші знання у цій системі. Дане визначення охоплює не лише несуперечливість та повноту знань, але й можливість виводу із одних суджень інших в рамках єдиної системи знань, а також формування пояснень щодо отриманих в результаті виводу нових суджень. Базуючись на підході [10], узгодження знань для побудови пояснень розглянемо у аспектах логічної та ймовірнісної несуперечливості, в аспекті виводу на несуперечливих знаннях, а також в аспекті пояснень зв'язків між елементами єдиної системи узгоджених знань.

Перший аспект враховує неповноту знань про функціонування інтелектуальної системи, що потребує використання ймовірнісного підходу.

Другий аспект пов'язаний із формуванням нових узгоджених знань в результаті виводу. Процес виводу задає складні зв'язки між знаннями, тобто визначає цілісність системи узгоджених знань. Перші три характеристики узгоджених знань визначають їх формальну, логічну та ймовірнісну складову. Елементи системи знань можуть бути зв'язані, наприклад, каузальними або темпоральними відношеннями.

Такі залежності внаслідок неповноти знань можуть мати ймовірнісні характеристики [13]. Врахування цих характеристик дає можливість сформулювати систему знань, на якій забезпечується логічний та ймовірнісний вивід. Тобто така система знань буде послідовно зв'язною, що дає можливість виконати формальне узгодження знань. Однак формальна узгодженість не враховує семантику цих знань, що може привести до неможливості їх практичного застосування.

Для того, щоб врахувати семантику, необхідно пояснити зв'язки між елементами знань, тобто задати для цих знань додаткове відношення пояснення. Дане відношення забезпечує семантичну зв'язність системи узгоджених знань. Кожен елемент такої системи може бути зв'язаний з іншим відношенням пояснення.

Таким чином, узгоджені знання для побудови пояснень в інтелектуальній інформаційній системі мають характеризуватись такими властивостями:

- логічною несуперечливістю фактів та гіпотез щодо каузальних зв'язків між цими фактами у складі пояснення; каузальність може розглядатись у широкому сенсі, в тому числі з урахуванням ймовірнісних підходів до визначення причинно-наслідкових зв'язків;

- сумісністю фактів та гіпотез у ймовірнісному аспекті, що передбачає логічну несуперечливість елементів знань із найбільшою ймовірністю їх застосування при побудові пояснення;

- можливістю ймовірнісного виводу пояснення на узгоджених в логічному та ймовірнісному аспектах знаннях; результати такого виводу мають ймовірнісну оцінку, фінальне пояснення відбирається за найбільшим значенням ймовірності;

- відсутністю аномальних залежностей для знань, що використовуються при побудові пояснень; аномальність в даному випадку розглядається як неможливість отримати пояснення щодо існуючої закономірності.

Слід зазначити, що аномалії в знаннях можуть бути виявлені шляхом співставлення результатів ймовірнісного виводу для різних властивостей предметної області при використанні однакових вхідних даних. Аномалія виникає у випадку, якщо результати такого виводу приводять до взаємовиключних висновків. Наприклад, якщо результати першого виводу свідчать про популярність рекомендованого товару серед користувачів рекомендаційної системи, а результати другого - про недостатню надійність цього товару.

У відповідності до першої, другої та третьої властивостей, пояснення фактів та гіпотез може бути забезпечено шляхом ймовірнісного виводу на узгоджених знаннях. Такий вивід дає можливість порівняти найбільш ймовірні гіпотези, що враховують не лише поточні факти, але й базові знання з предметної області. Отримані гіпотези мають бути перевірені з урахуванням відношення пояснення. Тобто кожна з гіпотез має бути пояснена через іншу гіпотезу.

Таким чином, можливість пояснити результати як традиційного логічного, так і ймовірнісного виводу є невід'ємною властивістю узгоджених знань. Це означає, що узгодження знань є необхідною умовою для побудови модулю пояснень, що доповнює інтелектуальну систему. Узгодження знань дає можливість використати принцип чорного ящика при побудові пояснень. Згідно з цим принципом, для формування тлумачень можуть бути використані не лише вхідні дані та знання, які використовує інтелектуальна система при прийнятті рішень, а й додаткові знання про предметну область, які спрощують розуміння принципів та алгоритмів роботи цієї системи для користувача [14].

Структуру представлення узгоджених знань для побудови пояснень в інтелектуальних системах представлено на рис. 1. Згідно з розглянутими властивостями таких знань, погодження проводиться на двох рівнях: безпосередньо на рівні знань та на рівні пояснень щодо цих знань.

На першому рівні окремо узгоджуються факти даних, подій та результатів. При використанні даних вибирається така їх підмножина, для якої можуть бути погоджені факти виникнення цих даних. Для подій враховуються не лише каузальні, але й темпоральні залежності. Узгодженість результатів досягається шляхом співставлення фактів отримання результатів, які відображають різні властивості отриманого рішення. Таким чином, на першому рівні узгоджуються однотипні елементи знань.

На другому рівні виконується пояснення як однотипних елементів знань (факти подій процесу прийняття рішення та зв'язки між цими фактами), так і елементів із різних типів знань. Пояснення подій формується на базі як окремих залежностей, так і отриманої в результаті виводу послідовності таких залежностей.

Пояснення, згідно з запропонованим підходом, має дуальні властивості:

- відношення пояснення використовується для перевірки узгодженості знань щодо процесу прийняття рішення в інтелектуальній інформаційній системі;

- пояснення формується для користувача інтелектуальної системи з використанням узгоджених знань на основі вхідних даних такої системи, а також інформації про події, що відображають послідовність прийняття рішення в такій системі; узгодження в даному випадку грає роль обмеження при представленні пояснень користувачеві.

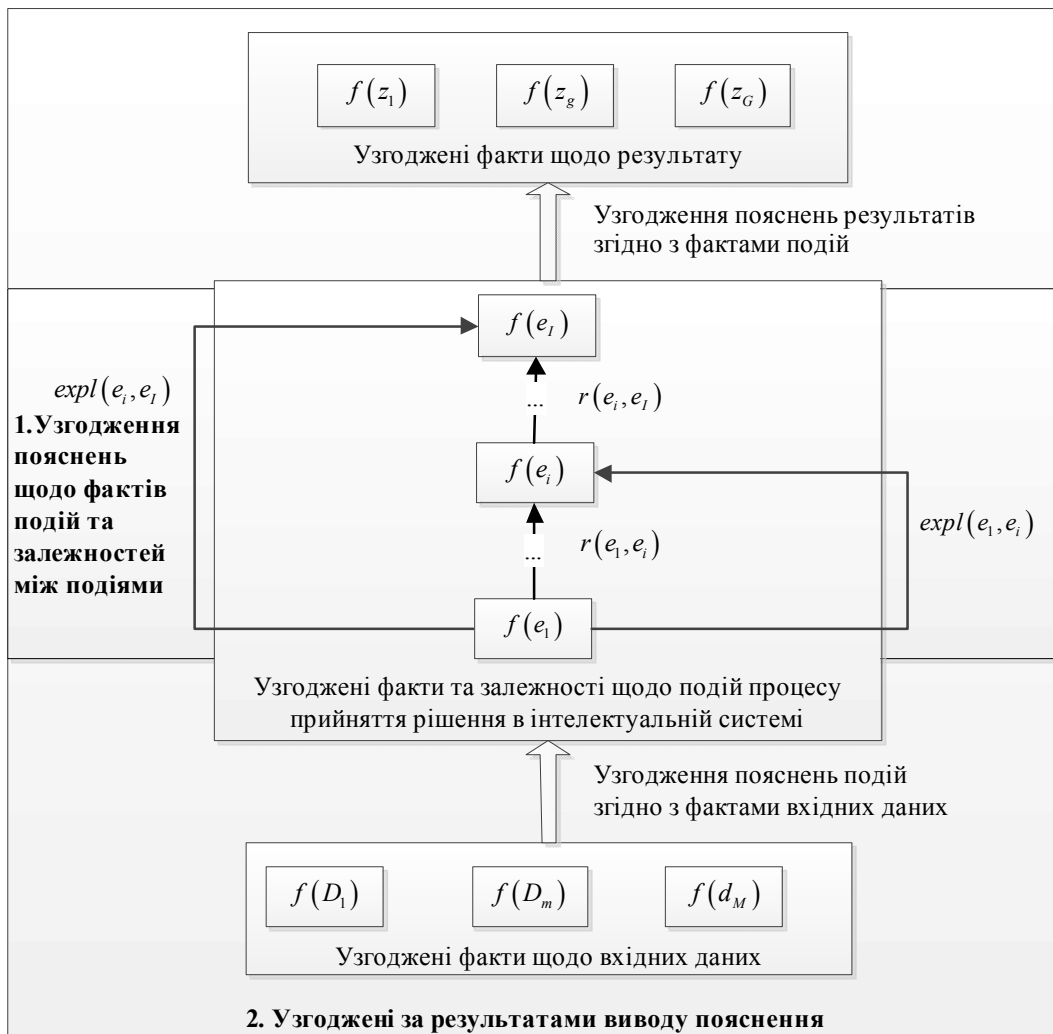


Рис. 1. Представлення узгоджених знань для побудови пояснень в інтелектуальних системах

Таким чином, узгодження знань використовуються як при формуванні цих знань, так і при побудові пояснення для користувача. Тому при постановці задачі побудови пояснення необхідно врахувати такі рівні узгодження знань:

- у логічному аспекті та в аспекті пояснення;
- у ймовірнісному аспекті та в аспекті пояснення.

У даному дослідженні головну увагу приділено логічному аспекту та аспекту пояснення у представленні узгоджених знань.

#### 4. Розробка моделі представлення узгоджених знань з урахуванням їх логічної несуперечливості

Постановка задачі побудови тлумачення полягає у знаходженні такого упорядкованого набору пояснень, який би задовольняв умові узгодженості вхідних даних, знань та поточного або кінцевого результату роботи інтелектуальної інформаційної системи безпосередньо в аспекті пояснень:

$$\begin{aligned}
 & \text{Знайти } Expl(D, E, Z) \\
 & \text{при обмеженнях} \\
 & (\forall i) Expl_i(D) \hat{=} Expl_i(E) \hat{=} Expl_i(Z) \\
 & (\forall i \forall k) Expl_i(D_m) \hat{=} Expl_k(D_m), \\
 & Expl_i(e_j, e_l) \hat{=} Expl_k(e_j, e_l), \\
 & Expl_i(z_g) \hat{=} Expl_k(z_g)
 \end{aligned} \tag{1}$$



де  $Expl(D, E, Z)$  - множина пояснень щодо вхідних даних, подій та рішення інтелектуальної інформаційної системи;  $Expl_i(D)$  - пояснення щодо вхідних даних;  $Expl_i(E)$  - пояснення щодо подій процесу прийняття рішення і інтелектуальній системі;  $Expl_i(Z)$  - пояснення щодо отриманого в результаті роботи інтелектуальної інформаційної системи рішення;  $Expl_i(D) \hat{=} Expl_i(E) \hat{=} Expl_i(Z)$  - обмеження узгодженості пояснень щодо даних, подій та результату за умови, що вони описують єдиний  $i$ -й процес прийняття рішення в інтелектуальній системі;  $Expl_i(D_m) \hat{=} Expl_k(D_m)$  - обмеження узгодженості різних пояснень щодо одного набору вхідних даних;  $Expl_i(e_j, e_l) \hat{=} Expl_k(e_j, e_l)$  - обмеження узгодженості різних пояснень щодо елементів одного й того ж процесу прийняття рішення в інформаційній системі;  $Expl_i(z_g) \hat{=} Expl_k(z_g)$  - обмеження узгодженості щодо одного із варіантів отриманого рішення.

Задача побудови пояснень у постановці (1) враховує логічну узгодженість знань. Такі знання відображають статичні та динамічні аспекти предметної області. Статичні властивості представляються у вигляді фактів, а динамічні - відношень між цими фактами.

Факти використання одного й того ж набору даних мають бути узгодженими між собою в сенсі логічної несуперечливості:

$$f_i(D_m) \hat{=} f_n(D_m) | \neg(f_i(D_m) \wedge f_n(D_m)) \quad (2)$$

де  $f_i(D_m), f_n(D_m)$  -  $i$ -й та  $n$ -й факти використання підмножини вхідних даних  $D_m$ .

Факти щодо подій процесу прийняття рішення в інтелектуальній системі узгоджуються не лише в сенсі логічної несуперечливості, а й з урахуванням порядку їх виникнення в часі:

$$f(e_j) \hat{=} f(e_l) | (f(e_j) \wedge f(e_l)) \wedge (e_j F^+ e_l) \vee (e_l F^+ e_j) \vee \exists Expl(e_j, e_l) \quad (3)$$

де  $f(e_j), f(e_l)$  -  $i$ -й та  $n$ -й факти щодо подій  $e_j$  та  $e_l$ ;  $e_j F^+ e_l$  - порядок подій у часі, що визначається темпоральним оператором  $F^+$ , тобто подія  $e_l$  обов'язково відбувається після події  $e_j$ , як безпосередньо за цією подією, так і після декількох проміжних подій;  $Expl(e_j, e_l)$  - пояснення послідовності подій.

Додаткове темпоральне узгодження фактів виникнення подій є необхідним внаслідок того, що послідовність отримання результату є лінійною у часі для користувача. Важливість лінійного представлення подій для отримання пояснення пов'язана з тим, що на кожному етапі деревоподібного процесу прийняття рішення існує порядок у часі для кожної пари подій.

В залежності від вхідних даних та попередніх подій у кожній точці вибору відбувається перехід до однієї гілки процесу прийняття рішення. Тому для кожного сформованого рішення користувачеві необхідно пояснити лінійну послідовність дій, яка представлена упорядкованою послідовністю подій та привела до запропонованого йому результату.

Узгодженість результатів роботи інтелектуальної системи визначається в аспектах їх несуперечливості або можливості пояснення їх суперечливості:

$$(\forall g \forall q) f(z_g) \hat{=} f(z_q) | \neg(f(z_g) \wedge f(z_q)) \vee \exists Expl(f(z_g), f(z_q)) \quad (4)$$

де  $f(z_g), f(z_q)$  - факти отримання результатів  $z_g, z_q \in Z$ ;  $Expl(f(z_g), f(z_q))$  - узгоджене пояснення щодо фактів отримання результатів роботи інтелектуальної системи.

Модель представлення узгоджених знань з урахуванням їх логічної несуперечливості або можливості пояснень виявленої суперечливості має вигляд:

$$M = \left\langle \begin{array}{l} D, E, Z : (\forall z_g \forall z_q) f(z_g) \hat{=} f(z_q), \\ (\forall e_j \forall e_l) f(e_j) \hat{=} f(e_l), \\ (\forall D_m \subseteq D) f_i(D_m) \hat{=} f_n(D_m) \end{array} \right\rangle \quad (5)$$

Побудова узгоджених знань згідно з представленою моделлю створює умови для автоматизованого формування пояснень за принципом чорного ящика.

### 5. Висновки і перспективи подальших досліджень

Розроблено узагальнений підхід до узгодження знань для побудови пояснень в інтелектуальних інформаційних системах. Підхід передбачає узгодження знань з урахуванням їх несуперечливості у логічному та ймовірнісному аспектах, а також за відсутності аномальних елементів знань. У практичному плані підхід забезпечує можливість побудови набору узгоджених пояснень, що відображають різні аспекти процесу прийняття рішення та отриманого інтелектуальною системою результату, що спрощує застосування цих результатів у предметній області користувача.

Запропоновано модель представлення узгоджених знань з урахуванням їх логічної несуперечливості. Модель призначена для побудови пояснень та містить несуперечливі вхідні дані, події процесу прийняття рішення, для яких задано порядок у часі, а також несуперечливі результати роботи інтелектуальної системи. Модель дає можливість перевірити узгодженість знань у процесі формування пояснень щодо процесу та результатів роботи інтелектуальної системи, що створює умови для формування пояснень за принципом чорного ящика та доповнення функціонуючих інтелектуальних систем можливостями пояснень.

Подальші дослідження у напрямку узгодження знань при побудові пояснень пов'язані із погодженням знань із урахуванням ймовірнісної несуперечливості, що дасть можливість використати неповні та неточні знання при побудові пояснень. Використання ймовірнісного представлення знань також дає можливість запропонувати користувачеві декілька варіантів пояснення з урахуванням різних аспектів предметної області та упорядкованих за ймовірнісним показником.

**Список літератури:** 1. *Miller T.* Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*. 2019. №267. P. 1-38. 2. *Zhang Y., Chen X.* Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*. 2020. № 14(1). P. 1-101. 3. *Левикін В.М., Чала О.В.* Підтримка прийняття рішень в інформаційно-управляючих системах з використанням темпоральної бази знань. *Науково-технічний журнал «Сучасні інформаційні системи»*, 2018 V. 2. №. 4. P. 101-107. 4. *Levykin V., Chala O.* Development of a method of probabilistic inference of sequences of business process activities to support business process management. *Eastern-European Journal of Enterprise Technologies*. 2018. № 5/3(95). P. 16-24. DOI: 10.15587/1729-4061.2018.142664. 5. *Phillips-Wren G.* Intelligent Systems to Support Human Decision Making. *Artificial Intelligence: Concepts, Methodologies, Tools, and Applications*. 2017. P. 3023-3036. <http://doi:10.4018/978-1-5225-1759-7.ch125>. 6. *Ribeiro M., Singh S., Guestrin C.* "Why should I trust you?": Explaining the predictions of any classifier. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. 2016. P. 97-101. DOI: 10.1145/2939672.2939778. 7. *Wang T., Lin Q.* Hybrid decision making: When interpretable models collaborate with black-box models. *Journal of Machine Learning Research*. 2019. № 1. P. 1-48. 8. *Thagard P., Verbeurgt K.* Coherence as constraint satisfaction. *Cognitive Science*. 1998. №. 22. P. 1-24. 9. *Thagard P.* Coherence, truth, and the development of scientific knowledge. *Philosophy of Science*. 2007. №. 74. P. 28-47. 10. *Thagard P.* Causal inference in legal decision making: Explanatory coherence vs. Bayesian networks. *Applied Artificial Intelligence*. 2004. №. 18. P. 231-249. 11. *Harwood W.* The Logic of Trust. PhD thesis, University of York. 2012. P. 245. 12. *Laurence B.J.* The Structure of Empirical Knowledge. Harvard University Press. 1985. P. 258. 13. *Chalyi S.F., Leshchynskiy V.O., Leshchynska I.O.* Explanation Model in an Intelligent Information System Based on the Concept of Knowledge Coherence. 2020. №1(3). P. 19-23. <https://doi.org/10.20998/2079-0023.2020.01.04>. 14. *Chalyi S., Leshchynskiy V., Leshchynska I.* Designing explanations in the recommender systems based on the principle of a black box // *Сучасні інформаційні системи*. 2019. Т. 3, № 2. С. 47-51.

*Надійшла до редколегії 29.06.2021*

**Чалий Сергій Федорович**, доктор технічних наук, професор, професор кафедри інформаційних управляючих систем ХНУРЕ. Наукові інтереси: розробка моделей, методів і технологій автоматизованого управління бізнес-процесами (в тому числі із змінною структурою) в умовах неконтрольованих зовнішніх збурень. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. (057) 702 14 51.

**Лещинський Володимир Олександрович**, кандидат технічних наук, доцент кафедри програмної інженерії ХНУРЕ. Наукові інтереси: проектування, аналіз та рефакторинг коду програмного забезпечення. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. +38 (095) 203 17 50, +38 (098) 232 10 07.

**Лещинська Ірина Олександрівна**, кандидат технічних наук, доцент кафедри програмної інженерії ХНУРЕ. Наукові інтереси: проектування, аналіз та рефакторинг коду програмного забезпечення. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. +38 (095) 203 17 50, +38 (098) 232 10 07.

*О.С. ЧАЛА*

## **ЗГОРТКОВА ПОДВІЙНА РАДІАЛЬНО-БАЗИСНА НЕЙРОННА МЕРЕЖА НА ОСНОВІ АКТИВАЦІЙНИХ ЯДЕРНИХ ФУНКЦІЙ СПЕЦІАЛЬНОГО ТИПУ**

Запропонована нейронна мережа, що має ядерні функції активації та призначена для вирішення задач розпізнавання та класифікації зображень в режимі онлайн. Модифікована ймовірнісна нейронна мережа використовується в якості автоенкодера і отримує на вхідний шар зображення, представлені у оригінальній матричній формі. На виходах нейрокомпресора, кількість яких визначається кількістю класів у наборі даних, з'являється вектор сигналів попередньої класифікації. Потім утворений вектор-сигнал надходить на вхідний шар радіально-базисної нейронної мережі, яка утворює розділяючі гіперповерхні між класами довільної складної форми. Система характеризується не тільки високою точністю класифікації, а й високою швидкістю навчання, що дозволяє обробляти потоки даних, які послідовно подаються в режимі онлайн.

### **1. Вступ**

Розпізнавання образів-зображень на сьогодні є однією з ключових задач інтелектуального аналізу даних, що вирішується у онлайн-режимі в рамках практичних завдань ідентифікації власників електронних пристроїв, знаходження правопорушників, розпізнавання об'єктів дорожньої інфраструктури безпілотними машинами, постановки попереднього діагнозу у медицині. Ця задача є досить складною у комп'ютерній реалізації через різноманітність форм кожного образу, тому для вирішення використовуються підходи, засновані на апараті обчислювального інтелекту [1-3].

На сьогодні найефективнішим апаратом для вирішення задачі розпізнавання образів-зображень є згорткові нейронні мережі (CNN), які є одним з найбільш передових різновидів глибоких нейронних мереж (DNN) [4-6]. Такі мережі забезпечують високу точність розпізнавання, проте їх використання стикається з цілим рядом проблем. По-перше, для навчання CNN потрібні великі обсяги навчальних виборок, які не завжди доступні при вирішенні практичних завдань. Використання передавального навчання далеко не завжди дозволяє вирішити цю проблему. По-друге, згорткові нейронні мережі містять, як правило, дуже велику кількість параметрів синаптичних ваг, та для свого навчання вимагають досить багато часу. Тому робота CNN в онлайн-режимі є практично неможливою. Згорткова нейронна мережа складається з двох секцій: нейрокомпресора, утвореного послідовністю шарів згортки та субдискретизації, та багатошарового перцептрона (MLP), який, власне, і вирішує задачу апроксимації-розпізнавання. Вхідне зображення поступає на вхід нейрокомпресора, що перетворює вхідне зображення у вигляді матриці у вектор відносно невисокої розмірності, який надходить на входи багатошарового перцептрона. Більшість витрат часу у згорткових нейронних мережах припадає на навчання багатошарових перцептронів.

Для того, щоб різко підвищити швидкодію процесу розпізнавання, можна використати переваги ймовірнісних нейронних мереж (PNN), основою яких є ідеї байєсівського висновку ядерної регресії, вікон Парзена та оцінки Надарая-Ватсона [7-9]. Налаштування параметрів таких мереж реалізується на основі лінивого навчання [10] за принципом «Нейрони в точках даних», тобто проходить дуже швидко, з використанням «миттєвих моделей» [11]. Ймовірнісні нейронні мережі, незважаючи на суттєві переваги у швидкості, а саме, здатність навчатися та вирішувати задачі розпізнавання майже миттєво, програють за точністю згортковим нейронним мережам.

Основною вимогою, що висувуються до MLP, є універсальні апроксимуючі властивості. Такі властивості мають не тільки багатошарові перцептрони, але й мережі радіально-базисних функцій (radial-basis function networks, RBFNs) [12-16], які є «близькими родичами» ймовірнісних нейронних мереж, що реалізують ядерну апроксимацію [9] і не поступаються за точністю MLP. В [16] була запропонована і досліджена CNN, в якій замість MLP

була використана RBFN. Така модифікована мережа є досить ефективною при вирішенні задачі розпізнавання образів, проте все ще має певні недоліки. По-перше, мережі радіально-базисних функцій страждають від «прокляття розмірності» коли кількість параметрів синаптичних ваг експоненційно зростає із зростанням розмірності вхідного вектора, що веде до формування вектора невисокої розмірності на виході останнього шару субдискретизації автоенкодера та відповідної втрати точності. По-друге, в такій мережі основний час витрачається на операції згортки та субдискретизації, а не власне на завдання апроксимації.

Подолати недоліки розглянутих мереж CNN та RBFN можна, використовуючи замість стандартного автоенкодера класичну ймовірнісну нейронну мережу. Остання повинна бути модифікована таким чином, щоб вхідний сигнал, який надходить до мережі, формувався не у вигляді вектора, а в формі зображення-матриці [17]. Варто зауважити, що класична ймовірнісна нейронна мережа також потерпає від «прокляття розмірності», тобто вона нарощує свою архітектуру у першому прихованому шарі патернів з надходженням нових спостережень з навчальної вибірки. Для того, щоб подолати цей недолік, можна замість стандартної RBFN застосувати, так звану мережу гіпербазисних функцій (HBFN) [18]. Ця мережа використовує замість рецепторних полів гіперсфер активаційних функцій RBFN, що засновано на гіпереліпсоїдах, які мають довільну орієнтацію осей. Введення додаткових контурів навчання параметрів цих гіпереліпсоїдів [19] дозволяє поліпшити апроксимаційні властивості мережі, зменшити кількість параметрів (подолання «прокляття розмірності») і захиститися від виникнення «дірок» у просторі ознак.

## 2. Архітектура подвійної радіально-базисної нейронної мережі з матричними входами

Запропонована нейронна мережа призначена для вирішення задачі розпізнавання зображень, і, подібно до традиційних глибоких згорткових нейронних мереж, також складається з двох секцій: автоенкодера у вигляді матричної ймовірнісної нейронної мережі і апроксиматора у вигляді радіально-базисної нейронної мережі з налаштовуваними рецепторними полями. При цьому архітектури обох секцій мережі досить близькі і засновані на використанні багатовимірних ядерних активаційних функцій [9]. Архітектура запропонованої мережі наведена на рис. 1.

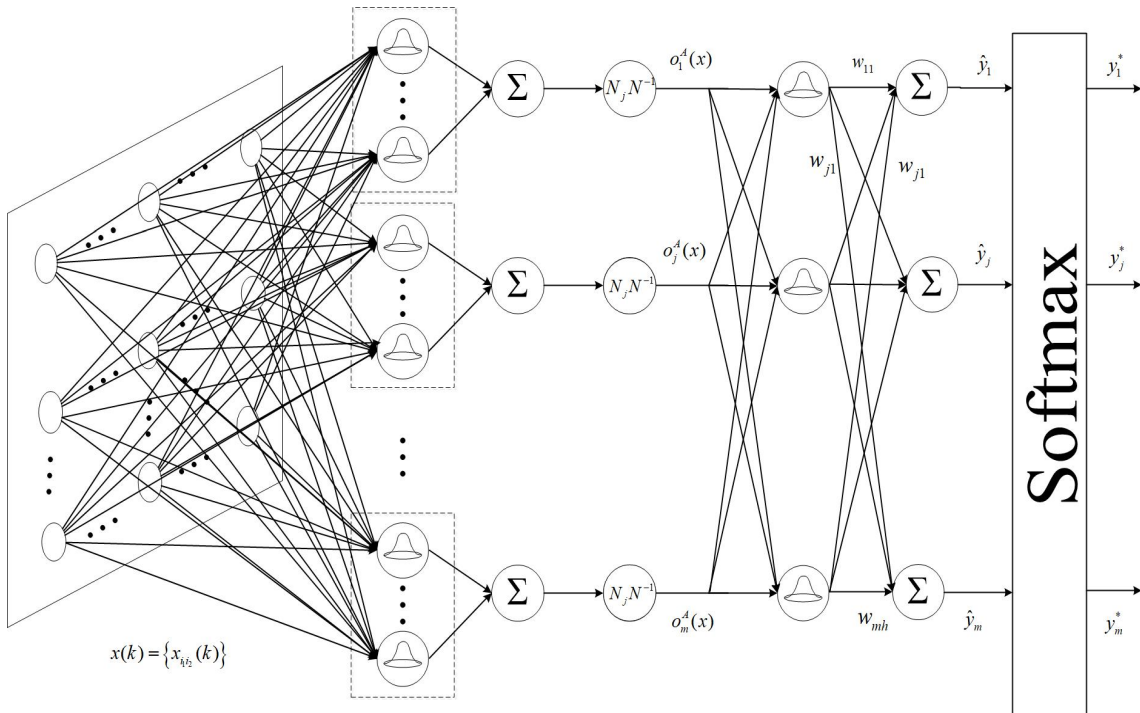


Рис. 1. Архітектура подвійної радіально-базисної нейронної мережі

Автоенкодер побудований на основі матричної ймовірнісної нейронної мережі [17] і містить три шари обробки інформації: шар образів, другий прихований шар, утворений  $m$  суматорами (тут  $m$  - кількість класів у вибірці) і третій (вихідний) шар корекції розподілу ймовірностей.

Навчальна вибірка представляє собою масив з  $N$  образів, кожен з яких являє собою  $(n_1 \times n_2)$  матрицю-зображення  $x(k) = \{x_{i_1 i_2}(k)\}$ , де  $k=1, 2, \dots, N$  - номер образу в навчальній вибірці. Також передбачається, що  $N_1$  образів у початковому масиві даних відноситься до першого класу  $Cl_1$ ,  $N_2$  - до другого  $Cl_2$  й нарешті,  $N_m$  -  $m$ -го класу  $Cl_m$ , тобто  $\sum_{j=1}^m N_j = N$ .

Кількість ядерних активаційних функцій (R-нейрони) у класичній PNN визначається обсягом вибірки  $N$ . Активаційні функції шару образів автоенкодера позначаються як

$$\varphi^A(x, c_{\tau_j}, \sigma_{\tau_j}^2), \quad (1)$$

де  $j$  - номер спостереження;  $\tau_j$  варіюється в інтервалі;  $c_{\tau_j} \in R^{n_1 \times n_2}$  - матриця-центр активаційної функції, яка визначається в процесі навчання;  $\sigma_{\tau_j}^2$  - параметр сферичного рецепторного поля відповідної дзвонуватої активаційної функції.

В якості активаційної функції була обрана ядерна функція В. Спанечнікова [20], модифікована для матричного випадку:

$$\varphi^A(x, c_{\tau_j}, \sigma_{\tau_j}^2) = 1 - \|x - c_{\tau_j}\|_{\sigma_{\tau_j}^2}^2, \quad (2)$$

де  $\|x - c_{\tau_j}\|_{\sigma_{\tau_j}^2}^2 = \text{Tr}(x - c_{\tau_j})(x - c_{\tau_j})^T$  - символ сліду матриці (матрична метрика Фробеніуса);  $\sigma_{\tau_j}^2$  - радіус рецепторного поля активаційної функції.

Другий прихований шар нейрокомпресора утворений  $m$  суматорами, кожен з яких відноситься до конкретного класу  $Cl_j$ . На виходах цих суматорів розраховуються парзенівські оцінки щільності розподілу з урахуванням виходів попереднього шару  $o_{\tau_j}^{[1]}(x)$  для спостереження  $x$ :

$$p_j(x) = o_j^{[2]}(x) = \sum_{\tau_j = N_1 + N_2 + \dots + N_{j-1} + 1}^{N_1 + N_2 + \dots + N_j} o_{\tau_j}^{[1]}(x). \quad (3)$$

У результаті роботи третього шару ці оцінки уточнюються з урахуванням значень емпіричних апіорних ймовірностей  $N_j N^{-1}$  і на виходах автоенкодера з'являються ймовірності відношення спостереження  $x$ , що класифікується до  $j$ -го класу:

$$o_j^A(x) = o_j^{[2]}(x) N_j N^{-1}. \quad (4)$$

Таким чином, на виході нейрокомпресора з'явиться  $m$ -вимірний сигнал  $o^A(x) = (o_1^A(x), \dots, o_j^A(x), \dots, o_m^A(x))^T$ , який є попередньою оцінкою результатів класифікації.

Нейронна мережа-апроксиматор побудована на основі гіпербазисної нейронної мережі (НВНН), що є модифікацією популярної мережі радіально-базисних функцій з рецепторними гіпереліпсоїдними полями, що мають довільну орієнтацію осей. При цьому передбачається, що в процесі навчання НВФН параметри цих гіпереліпсоїдів можуть налаштовуватися одночасно з синаптичними вагами.

На входи НВФН з навчальної вибірки надходить послідовність  $o^A(x(k)) = o^A(k) = (o_1^A(x), \dots, o_j^A(x), \dots, o_m^A(x))^T$ ,  $k=1, 2, \dots, N$ , яка передається на  $h$  гіпербазисних R-нейронів, на виходах яких формуються сигнали

$$\varphi_l^H(o^A(k), c_l, \Sigma_l^{-1}) = 1 - (c \sum^{-1} (o^A(k) - c_l)) = 1 - \|o^A(k) - c_l\|_{\Sigma_l^{-1}}^2, \quad l=1, 2, \dots, h, h \gg m, \quad (5)$$

де  $c_l \in R^m$  - векторний центр активаційної функції  $\varphi_l^H(\circ)$ ,  $\Sigma_l^{-1}$  - коваріаційна матриця, яка визначає форму, розмір та орієнтацію осей рецепторного поля відповідної активаційної функції.

Вихідні сигнали R-нейронів подаються на вихідний шар HBNN, утворений елементарними перцептронами Розенблата з активаційними функціями типу softmax. Таким чином, на виходах HBNN в цілому формуються сигнали

$$\hat{y}_j(k) = w_{j0} + \sum_{l=1}^h w_{jl} \varphi_l^H \left( \|o^A(k) - c_l\|_{\Sigma_l^{-1}}^2 \right) = \sum_{l=0}^h w_{jl} \varphi_l^H \left( \|o^A(k) - c_l\|_{\Sigma_l^{-1}}^2 \right), \varphi_0^H(\circ) \equiv 1, \quad (6)$$

$$y_j^*(k) = \text{softmax } \hat{y}_j(k) = \exp \hat{y}_j(k) \left( \sum_{p=1}^m \exp \hat{y}_p(k) \right)^{-1}. \quad (7)$$

Якщо сигнал на виході PNN визначає рівень ймовірності того, що образ, який класифікується, відноситься до конкретного класу, то сигнал на виході введеної HBNN задає рівні нечіткої належності цього спостереження до того ж класу.

### 3. Навчання подвійної радіально-базисної нейронної мережі з матричними входами

Навчання подвійної радіально-базисної нейронної мережі відбувається окремо для автоенкодера і апроксиматора, базуючись на різних принципах. Налаштування нейрокомпресора реалізується за допомогою лінійного навчання за принципом «Нейрони в точках даних» і практично миттєво. Тобто центр ядерної активаційної функції встановлюється в точці з координатами вхідного образу з певною класифікацією, забезпечуючи високу точність класифікації і високу швидкість навчання [11].

Налаштування гіпербазисної нейронної мережі реалізується на основі контрольованого навчання з «гарячим» кодуванням навчального сигналу, тобто елементи зовнішнього навчального сигналу  $y_j(k)$  можуть приймати тільки два значення: 1, якщо  $x(k)$  належить до конкретного класу, і 0 в іншому випадку.

В якості критерію навчання HBNN використовується стандартна кросентропія:

$$E = - \sum_{k=1}^N \sum_{j=1}^m y_j(k) \ln y_j^*(k) = - \sum_{k=1}^N \sum_{j=1}^m y_j(k) \ln \exp \hat{y}_j(k) \left( \sum_{p=1}^m \exp \hat{y}_p(k) \right)^{-1}. \quad (8)$$

Далі для скорочення запису введемо додаткові позначення:  $w_j = (w_{j0}, w_{j1}, \dots, w_{jl}, \dots, w_{jh})^T$ ,  $\varphi^H(o^A(k), c_l, \Sigma_l^{-1}) = (\varphi_1^H(o^A(k), c_l, \Sigma_l^{-1}), \dots, \varphi_l^H(o^A(k), c_l, \Sigma_l^{-1}), \dots, \varphi_h^H(o^A(k), c_h, \Sigma_l^{-1}))^T$ .

Градентна процедура навчання [19] для налаштування синаптичних ваг, центрів і матриць рецепторних полів має вигляд:

$$w_{jl}(k+1) = w_{jl}(k) + \eta_w(k+1) \left( y_j(k+1) - w_j^T(k) \varphi^H \left( \|o^A(k+1) - c(k)\|_{\Sigma^{-1}(k)}^2 \right) \right) \varphi_l^H \left( \|o^A(k+1) - c_l(k)\|_{\Sigma_l^{-1}(k)}^2 \right), \quad (9)$$

$$c_l(k+1) = c_l(k) - \eta_c(k+1) \left( y_j(k+1) - w_j^T(k+1) \varphi^H \left( \|o^A(k+1) - c_l(k)\|_{\Sigma^{-1}(k)}^2 \right) \right) w_{jl}(k+1) \cdot \left( \varphi^H \left( \|o^A(k+1) - c_l(k)\|_{\Sigma^{-1}(k)}^2 \right) \right)_{\Sigma_l^{-1}(k)} \left( o^A(k+1) - c_l(k) \right), \quad (10)$$

$$\Sigma_l^{-1}(k+1) = \Sigma_l^{-1}(k) \eta_{\Sigma}(k+1) \left( y_j(k+1) - w_j^T(k+1) \varphi^H \left( \|o^A(k+1) - c_l(k)\|_{\Sigma^{-1}(k)}^2 \right) \right) \cdot w_{jl}(k+1) \left( \varphi^H \left( \|o^A(k+1) - c_l(k)\|_{\Sigma^{-1}(k)}^2 \right) \right) \left( o^A(k+1) - c_l(k+1) \right) \left( o^A(k+1) - c_l(k+1) \right)^T, \quad (11)$$

де  $w_j$  й  $\varphi^H \left( \left\| o^A(k) - c_l \right\|_{\Sigma_l}^2 \right)$  -  $(h+1) \times 1$  вектори синаптичних ваг на  $j$ -му виході системи і сигналів на виходах R-нейронів HBNN відповідно;  $\eta_w(k+1)$ ,  $\eta_c(k+1)$ ,  $\eta_\Sigma(k+1)$  - параметри кроку навчання для налаштування змінних.

Таким чином, на відміну від традиційних CNN, де налаштовуються тільки синаптичні ваги нейронів, в запропонованій системі одночасно уточнюються параметри активаційних функцій, що забезпечують їй додаткову гнучкість і швидкодію.

#### 4. Висновки

Запропоновано нейромережеву систему, яка була розроблена для вирішення задачі класифікації образів-зображень за мінімальний час. На відміну від традиційних згорткових нейронних мереж, запропонована система використовує ядерні активаційні функції. Вона складається з двох частин - нейрокомпресора та системи розпізнавання. В якості нейрокомпресора використовується модифікована ймовірнісна нейронна мережа з матричними входами, а система розпізнавання базується на гіпербазисній нейронній мережі.

Особливість розробленої системи полягає в комбінованому навчанні двох незалежних блоків. Перший блок, що представлений автокодером, налаштовується за допомогою лінійного навчання, базуючись на концепції «Нейрони в точках даних». Система розпізнавання функціонує за парадигмою навчання з учителем та дозволяє налаштувати не тільки синаптичні ваги, але й характеристики функцій активації - центри та рецепторні поля. Завдяки такому підходу до навчання, система в цілому характеризується високою швидкістю та високою точністю. Ця система дозволяє з високою точністю вирішувати задачу розпізнавання образів-зображень за умов коротких та довгих вибірок та вирішувати задачу класифікації потоку даних. При цьому запропонована система не потерпає від «прокляття розмірності», тобто дозволяє класифікувати спостереження, що надходять послідовно у онлайн-режимі та мають форму матриць.

**Список літератури:** 1. *Mumford C.* Computational Intelligence//Springer Berlin Heidelberg. 2009. P. 732. DOI: 10.1007/978-3-642-01799-5. 2. *Kacprzyk J., Pedrycz W.* Springer Handbook of Computational Intelligence// Berlin Heidelberg: Springer, Verlag. 2015. P. 1634. DOI: 10.1007/978-3-662-43505-2. 3. *Kruse R., Borgelt C., Klawonn F., Moewes C., Steinbrecher M., Held P.* Computational Intelligence: A Methodological Introduction// Berlin: Springer-Verlag. 2013. P.492. DOI: 10.1007/978-1-4471-5013-8. 4. *Bengio Y., Le Cun Y., Hinton G.* Deep learning//Nature. №521(7553). 2015. P.436-444. 5. *Schmidhuber J.* Deep learning. Neural networks: An overview//Neural Networks. №61. 2015. P.85-117. 6. *Goodfellow I., Bengio Y., Courville A.* Deep Learning/MIT Press. 2016. P. 800. 7. *Specht D.F.* Probabilistic neural networks//Neural Network. №3. 1990. pp. 109-118. 8. *Specht D. F.* Probabilistic neural networks and polynomial ADALINE as complementary techniques to classification//IEEE Trans. on Neural Networks. №1. 1990. P. 111-121. 9. *Kung S.Y.* Kernel Methods and Machine Learning// Cambridge: University. 2014. P. 591. DOI: 10.1017/CBO9781139176224. 10. *Zahiriak D.R., Chapman R., Rogers S.K., Suter B.W., Kabriski M., Pyatti V.* Pattern recognition using radial basis function network//Aerospace Application of Artificial Intelligence, Proceedings, Dayton, Ohio. 1990. P. 249-260. 11. *Nelles O.* Nonlinear Systems Identification// Berlin: Springer. 2001. P.786. DOI: 10.1007/978-3-662-04323-3. 12. *Moody J., Darken C.J.* Fast learning in networks of locally tuned processing units//Neural Computation. №1. 1989. P. 281-294. 13. *Poggio T., Girosi F.* Networks for approximation and learning//Proceedings of the IEEE. №78(9). 1990. P. 1481-1497. 14. *Park J., Sandberg I. W.* Universal Approximation Using Radial-Basis-Function Networks//Neural Computation. №3(2). 1991. P. 246-257. 15. *Leonard J. A., Kramer M. A., Ungar L. H.* Using radial basis functions to approximate a function and its error bounds//IEEE Transactions on Neural Networks. №3(4). 1992. P. 624-627. 16. *Amirian M., Schwenker F.* Radial Basis Function Networks for Convolutional Neural Networks to Learn Similarity Distance Metric and Improve Interpretability//IEEE Access. №8. 2020. P. 123087-123097. 17. *Bodyanskiy Ye., Deineko A., Pliss I., Chala O., Nortsova A.* Matrix Fuzzy-Probabilistic Neural Network in Image Recognition Task//2020 IEEE Third International Conference on Data Stream Mining and Processing (DSMP). 2020. P. 33-36. 18. *Bodyanskiy Ye., Tyshchenko A., Deineko A.* An evolving radial basis neural network with adaptive learning of its parameters and architecture//Aut. Control Comp. Sci. №49. 2015. P. 255-260. 19. *Bodyanskiy Ye., Kolodyazhnyi V., Stephan A.* An Adaptive Learning Algorithm for a Neuro-fuzzy Network. Computational Intelligence. Theory and Applications. Fuzzy Days // Lecture Notes in Computer Science, Springer, Berlin, Heidelberg. №2206. 2001. P. 68-75. 20. *Epanechnikov V.A.* Non-Parametric Estimation of a Multivariate Probability Density// Theory Probab. №14(1). 1967. P. 153-158. DOI: 10.1137/1114019

Надійшла до редколегії 29.06.2021

**Чала Ольга Сергіївна**, молодший науковий співробітник ПНДІ АСУ ХНУРЕ. Наукові інтереси: нейронні мережі та інші засоби штучного інтелекту. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, тел. (057) 702 18 90.

*В.С. ЧУМАК, О.Г. АВРУНІН, Є.А. ЧУГУЙ, І.В. СВИД*

## **АНАЛІЗ ПРИНЦИПІВ ПОБУДОВИ ТЕЛЕМЕДИЧНИХ КОМПЛЕКСІВ ШИРОКОГО ПРИЗНАЧЕННЯ**

Враховуючи сучасні тенденції, у роботі наведено аналіз принципів побудови телемедичних систем, мереж та комплексів. Розроблено загальну структуру надання телемедичних послуг. Запропоновано структури реалізації портативного та мобільного телемедичного комплексу. Проаналізовано інфокомунікаційні системи та мережі стосовно впровадження в мобільних телемедичних комплексах. Обґрунтовано переваги застосування сенсорної мобільної натільної мережі стандарту IEEE 802.15.6 WBAN в мобільних телемедичних комплексах. Дослідження надійності сенсорної мережі проведено з використанням випадкового UDG-графа, що, у свою чергу, дозволяє визначити оптимальні параметри робочого циклу сенсорів.

### **1. Вступ**

Телемедицина - новий напрямок медичних послуг, який охоплює кілька різномірних областей - медицина, телекомунікації, інформаційні технології. У сучасному суспільстві внаслідок поточної епідеміологічної ситуації спричиненої COVID-19, телемедицина набула широкого розвитку та впровадження, як у світовому просторі, та, зокрема, і в Україні.

Ця сфера медичних послуг дозволяє пацієнту і лікарю заощадити час і сили, тому що спілкування відбувається онлайн. Це актуально, як для жителів мегаполісів, так і для людей, які живуть у сільській місцевості, і такий підхід забезпечує рівні можливості доступу до якісного медичного обслуговування для пацієнтів у всіх куточках країни. Крім того, телемедицина значно скорочує витрати на лікування, підвищує якість діагностики і реалізує можливість віддаленого моніторингу стану здоров'я, що, в свою чергу, є економічно ощадливим, як для пацієнта, так і для країни в цілому.

Існує декілька визначень терміну «телемедицина», зокрема, згідно з [1], «телемедицина - це напрям медицини, а саме, комплекс дій, технологій та заходів, що застосовуються при наданні медичної допомоги з використанням засобів дистанційного зв'язку у вигляді обміну електронними повідомленнями (у випадках, коли відстань є критичним чинником, наприклад, в сільській місцевості). Важливо: повідомленнями обмінюються не пацієнт і лікар, а лікарі між собою. Так лікарі первинної ланки мають можливість швидко отримати фахову консультацію вузькопрофільних колег».

У рамках політики Всесвітньої Організації Охорони Здоров'я (ВООЗ) в області телемедицини 1997 року було запропоновано наступне визначення. Телемедицина - це метод надання послуг з медичного обслуговування там, де відстань є критичним чинником. Послуги здійснюються медичними працівниками з використанням інформаційно-комунікаційних технологій з метою отримання інформації, необхідної для діагностики, лікування і профілактики захворювання [2].

Найпоширеніше визначення поняття «телемедицина» надано Американською асоціацією телемедицини: «Предмет телемедицини полягає в передаванні медичної інформації між віддаленими один від одного пунктами, де знаходяться пацієнти, лікарі, інші провайдери медичної допомоги, між окремими медичними закладами» [3].

### **2. Телемедицина в Україні**

На сьогодні телемедицина в Україні - це, в першу чергу, телемедичні консультації (консультації онлайн) сімейних лікарів, терапевтів або педіатрів, які працюють у віддалених районах фахівцями вищого рівня, наприклад, лікарями обласних лікарень [3].

Законопроект «Про телемедицину» (№ 10196 від 14.03.2012 р.) визначає її як комплекс організаційних, фінансових і технологічних заходів, що забезпечують надання дистанційної консультаційної медичної послуги, за якої пацієнт або лікар, що безпосередньо проводить обстеження та/або лікування пацієнта, отримує дистанційну консультацію іншого лікаря з використанням телекомунікацій.



На прохання Міністерства охорони здоров'я України та Міністерства регіонального розвитку, будівництва та житлово-комунального господарства України в межах проекту «Підтримка реформ та ефективного урядування в сфері охорони здоров'я в Україні», що фінансується Трастовим фондом за підтримки Швейцарської агенції з розвитку й співробітництва, Світовим банком було ініційоване дослідження. В рамках цього дослідження було проведено онлайн-анкетування 1108 лікарів первинної ланки, які працюють у населених пунктах з чисельністю населення до 100 тисяч осіб у п'яти областях України (Дніпропетровській - 245, Кіровоградській - 196, Полтавській - 237, Рівненській - 214, Харківській - 216). За результатами анкетування з'ясували, що лікар в середньому витрачає на відвідування пацієнтів на дому 2,5 години на день. Частина цього часу можна заощадити, якщо така послуга буде надаватися з застосуванням телемедицини технологій. Середня кількість випадків, коли у лікарів первинної ланки виникала потреба у телеконсультуванні, на місяць становила майже п'ять разів - тобто частіше, ніж раз на тиждень [1].

Головною метою використання телемедицини методів є забезпечення рівного доступу до медичних послуг належної якості, наприклад, при необхідності надання медичної допомоги пацієнту у випадках, коли відстань і час є критичними чинниками. Крім того, застосування телемедицини забезпечить формування цілісності медичної інформації про стан здоров'я пацієнтів та сприятиме створенню єдиного медичного простору в країні.

Найбільш прийнятними формами телеконсультування, за результатами дослідження, є: отримання висновку в електронному вигляді після перенаправлення пацієнта до вузького спеціаліста (54%); телеконсультування в режимі реального часу у присутності пацієнта (38%); отримання експертних порад на запит лікаря у режимі, відкладеному в часі (28%) [1]. Опитування також спростувало поширену тезу про те, що впровадження телемедицини є неможливим в Україні, особливо в сільській місцевості, через погане забезпечення комп'ютерами і недоступність Інтернету. Насправді, за свідченнями лікарів, 94% з них мають персональний комп'ютер на своєму робочому місці, а 90% мають доступ в Інтернет з комп'ютера на своєму робочому місці постійно або більшу частину часу [1].

Фахівці експертної групи МОЗ України, проаналізувавши результати дослідження, розробили «Методичні рекомендації з діагностики та лікування деяких захворювань при наданні телемедицини послуг для лікарів загальної практики - сімейної медицини» [4]. Цей документ містить детальні інструкції із застосування методів телемедицини для діагностики і лікування за напрямками, найбільш затребуваними лікарями первинної ланки, визначених дослідженням: пульмонологія та алергологія (бронхіальна астма, обструктивне захворювання легень); дерматологія (доброякісні і злоякісні новоутворення шкіри); кардіологія (гострий коронарний синдром); ендокринологія (цукровий діабет).

До основних телемедицини сервісів, що вже функціонують в Україні, можна віднести: телемедицину мережу Medinet [2, 5]; модуль «Телемедицина» РІМС «Медстар» [6]; проект «Телемедицина» [7, 8]; телемедицини сервіс для пацієнтів Doctor Online [9]; платформу для онлайн- зв'язку з лікарем Medikit [10]; телемедицини сервіс для пацієнтів «Поліклініка без черг» [11]; електронну систему охорони здоров'я eHealth [12].

Метою даної роботи є розробка узагальненої структурної схеми надання телемедицини послуг з урахуванням чинних нормативно-правових документів та стандартів. Згідно з цією метою, необхідно розробити структурну схему мобільного/портативного телемедицини комплексу, провести обґрунтування вибору мережі передачі даних у телемедицини комплексі та надати рекомендації щодо надійності запропонованої мережі.

### **3. Принципи побудови телемедицини систем та мереж**

Принципи побудови телемедицини систем та мереж, насамперед, визначаються інструментами телемедицини [4, 13, 14].

Основними інструментами телемедицини є:

- телемедицини консультування - надання медичної допомоги пацієнту лікарем із застосуванням телемедицини;

- телемедицини консиліум - надання медичної допомоги пацієнту із залученням декількох лікарів у складних випадках постановки діагнозу, лікування або проведення операцій, для передачі досвіду.

- телеметрія - сукупність технологій, що дають змогу проводити дистанційне вимірювання, збір і передачу інформації про показники діяльності (фізіологічні параметри) організму пацієнта;

- домашнє телеконсультування - процес спостереження за станом здоров'я пацієнта, який перебуває за межами закладу охорони здоров'я, із застосуванням телемедицини.

Ресурсне забезпечення телемедицини визначається такими особливостями [14]:

- технічні та інформаційно-програмні засоби телемедицини включають спеціалізоване програмне та програмно-апаратне забезпечення для реалізації функцій Порталу телемедицини; програмне та програмно-апаратне забезпечення відеозв'язку та аудіозв'язку; технічні прилади, оснащені засобами збереження і передачі даних та зображень у вигляді електронних повідомлень; засоби телеметрії; спеціалізоване програмне забезпечення для передавання, збереження та інтерпретації знімків і зображень, зафіксованих цифровими носіями, у тому числі у променевій діагностиці, обов'язково застосовує стандарт DICOM: ДСТУ ISO 17432:2009. Інформатика в охороні здоров'я. Повідомлення та пересилання даних. Web-доступ до файлових об'єктів системи DICOM (ISO 17432:2004, IDT);

- аудіо- та відеозв'язок самостійно використовуються при проведенні відеоконференцій,

відеосемінарів, відеолекцій та інших заходів, де передбачається візуальний контакт. Під час телемедичного консультування аудіо- та відеозв'язок може застосовуватись як додатковий інструмент комунікацій;

- під час надання медичної допомоги із застосуванням телемедицини забезпечується виконання вимог законодавства України щодо захисту персональних даних та дотримання норм етики та дентології надання медичної допомоги.

Враховуючи загальні принципи побудови телемедичних систем, комплексів та мереж затверджені в [14], запропоновано один з варіантів загальної структури надання телемедичних послуг (рис. 1).

Аналіз загальної структури надання телемедичних послуг показує, що вагоме місце у наведеній структурі займають інфокомунікаційні системи (мережі).

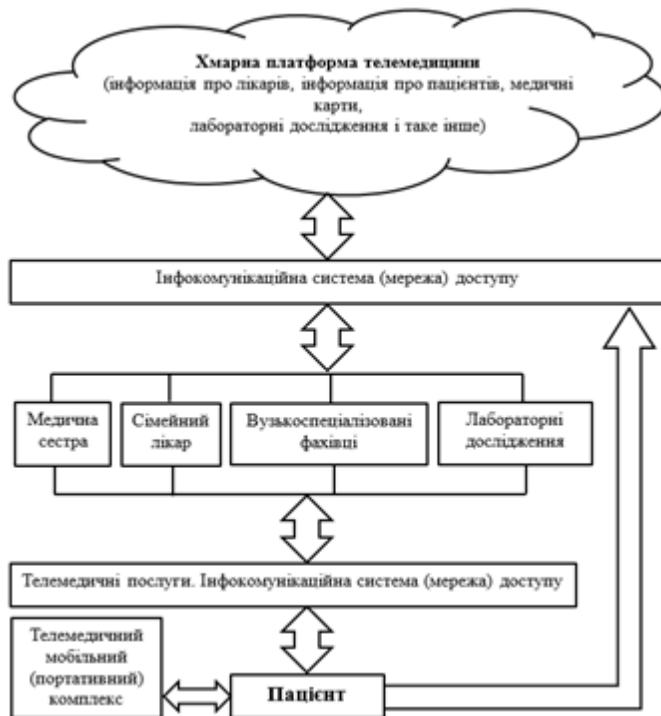


Рис. 1. Один з варіантів загальної структури надання телемедичних послуг

#### 4. Проектування структури телемедичного комплексу широкого призначення

Функціональне призначення телемедичного комплексу визначається відповідно до необхідних і достатніх вимог відстеження стану пацієнта, а для пацієнтів з хронічними захворюваннями і літніх людей це вкрай важливо. Крім того, телемедичний комплекс може використовуватися для відстеження стану спортсменів під час тренування, фітнес-тренування, реабілітації тощо.

Моніторинг можна класифікувати за метою (цільовим об'єктом) моніторингу, включаючи: моніторинг серця; гемодинамічний моніторинг; моніторинг дихання; неврологічний моніторинг; моніторинг рівня глюкози в крові; контроль температури тіла та інше.

Телемедичний комплекс може мати мобільну або портативну структуру. Запропоновані загальні структурні схеми побудови портативного/мобільного телемедичних комплексів наведено на рис. 2-3.

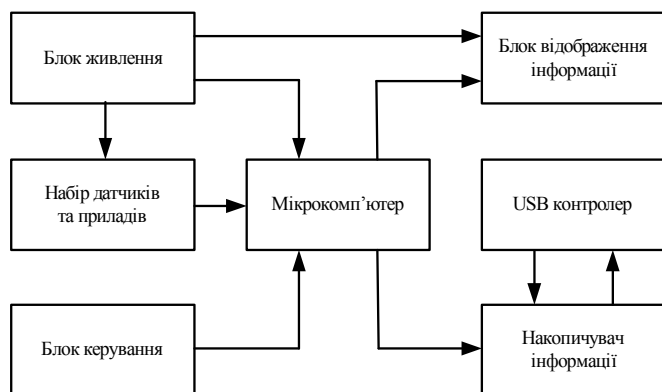


Рис. 2. Портативний телемедичний комплекс

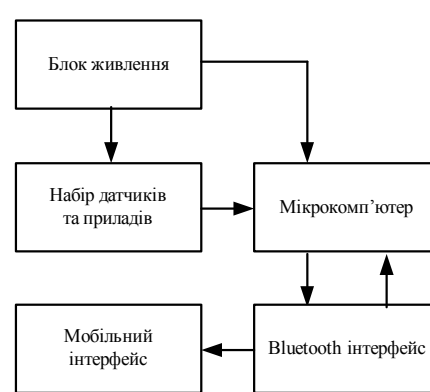


Рис. 3. Мобільний телемедичний комплекс

Все частіше у телемедичних комплексах і системах використовуються сенсорні мережі [19-22].

В наведеній структурі мобільного телемедичного комплексу (рис. 2) запропоновано використовувати сенсорну мобільну натільну мережу технології WBAN.

WBAN - це сенсорна мобільна натільна мережа стандарту IEEE 802.15.6, який було введено з метою забезпечити міжнародний стандарт малопотужного, малого радіусу дії й надзвичайно надійного бездротового зв'язку в межах навколишнього середовища людського тіла, підтримуючи великий діапазон швидкостей передачі даних для різних додатків. У цьому стандарті описано бездротовий зв'язок ближньої дії поблизу або всередині людського тіла (але не обмежуючись людьми). Він використовує існуючі промислово-наукові медичні (ISM) діапазони, а також діапазони частот, затверджені національними медичними та/або регулюючими органами. Необхідна підтримка якості обслуговування (QoS), надзвичайно низького енергоспоживання і швидкості передачі даних до 10 Мбіт/с при одночасному дотриманні суворих правил невтручання в разі потреби. У цьому стандарті розглядається вплив на портативні антени через присутність людини (в залежності від того, чи є ця людина чоловіком, жінкою, худю, повною тощо). Застосовується форма діаграми спрямованості для мінімізації питомого коефіцієнту поглинання (SAR) в організмі, а також зміни характеристик в результаті дій користувача. Загальна структура мережі WBAN наведена на рис. 4.

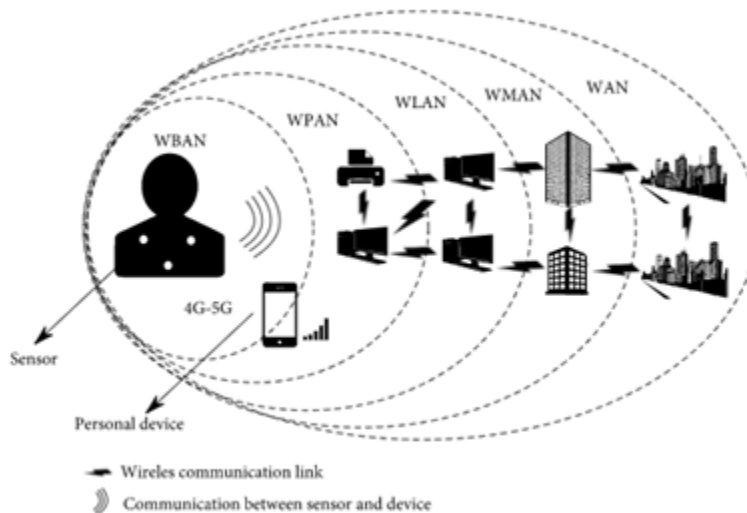


Рис. 4. Загальна структура мережі WBAN

Стандарт IEEE 802.15.6 націлений на забезпечення конфіденційності, аутентифікації, цілісності, захисту конфіденційності та захисту від відтворення. Всі вузли та концентратори повинні вибрати три рівня безпеки: незахищений зв'язок (рівень 0); аутентифікація, але без шифрування (рівень 1); а також аутентифікація і шифрування (рівень 2). У процесі зіставлення безпеки вузол і концентратор повинні спільно вибрати відповідний рівень безпеки. При одноадресній передачі активується попередньо розділений або новий загальний ключ (МК). Потім створюється парний тимчасовий ключ (РТК), який використовується тільки один раз на сеанс.

При багатоадресній передачі створюється тимчасовий ключ групи (ГТК), який використовується спільно з відповідною групою. Всі вузли та концентратори в WBAN повинні пройти певні етапи на рівні MAC перед обміном даними. Використовується процедура «асоціація безпеки» - процедура для ідентифікації вузла і концентратора один для одного; для встановлення нового загального ключа (МК), спільно використовуваного між ними, або для активації існуючого МК, попередньо спільно використовуваного між ними. Асоціація безпеки в стандарті IEEE 802.15.6 заснована на чотирьох протоколах узгодження ключів.

У стандарті IEEE 802.15.6 в якості носіїв інформації запропоновано використовувати імпульсні сигнали трьох типів: імпульси з лінійною частотною модуляцією (chirp pulses); хаотичні імпульси (chaotic pulses); короткі імпульси (short pulses shape).

При побудові сенсорних мереж стандарту IEEE 802.15.6, як і для подібних мереж стандартів IEEE 802.x, існує нагальна проблема - енергоефективність використовуваних датчиків, яка залежить від параметрів робочого циклу датчика, надійності мережі (ймовірності наявності можливості обміну даними між двома вузлами в визначений проміжок часу) і радіусу передачі повідомлення.

Запропоновано використовувати графові моделі для визначення надійності сенсорної мережі, і, як наслідок, визначення оптимальних параметрів робочого циклу сенсорів. Припустимо, що топологія мережі моделюється випадковим геометричним графом. Випадковий геометричний граф з параметрами  $m$  і  $r$  будується шляхом розміщення на площині (випадковим чином і незалежно один від одного)  $m$  вершин і додаванням ребер між тими вершинами, які знаходяться на відстані не більше  $r$  один від одного. Моделювання сенсорних мереж таким графом вважається більш правдоподібним, ніж випадковим графом. Серед геометричних графів виділяють важливий підклас - Unit Disk Graphs (UDG-графи) (рис. 5). У такому графі, розміщеному в Евклідовій площині, ребро між двома вершинами існує, якщо евклідова відстань між цими вершинами менше або дорівнює 1. Якщо вершини розподілені на площині випадковим чином і незалежно одна від одної, то такий UDG-граф називається випадковим UDG-графом [24].

Запропоновано використовувати випадковий UDG-граф  $G = (M, R)$ , де  $M$  - множина вершин,  $R$  - множина ребер, у якості моделі для аналізу сенсорної мережі. Припустимо, що ребра випадкового геометричного графу абсолютно надійні, оскільки у даному випадку, не розглядаються питання, пов'язані з затуханням сигналу, інтерференцією, екрануванням тощо. Надійність вершини графу (доступність сенсора) можна визначити за формулою:

$$p = 1 - S/T, \quad (1)$$

де  $S$  - середній час перебування сенсора в неактивному стані впродовж робочого циклу;  $T$  - період (довжина) робочого циклу.

Зазначимо, що надійність сенсора  $p$  є монотонно спадаючою функцією від радіуса передачі  $r$ :

$$r_1 > r_2 \Rightarrow p(r_1) < p(r_2). \quad (2)$$

Кількість ребер у графі ( $m$ ) має прямо пропорційну залежність від радіуса передачі.

$$r_1 > r_2 \Rightarrow m(r_1) > m(r_2). \quad (3)$$

В загальному виді задачу оптимізації надійності сенсорної мережі можна представити наступним чином [24]:

$$R(G(n, m(p), p(S))) \rightarrow \max(S), \quad (4)$$

де  $n$  - ваговий коефіцієнт вартості (надійності) вузла сенсорної мережі.

Використовуючи наведені вище властивості, отримаємо:

$$r = \arg \max_{r \in \Omega_r} R(G(n, m(r), p(r))), \quad (5)$$

де  $\Omega_r$  - множина допустимих відстаней для  $r$  між вузлами сенсорної мережі.

В деяких випадках доцільно обмежити максимальну кількість вершин у графі через специфіку протоколів MAC або інтерференцію.

Запропонований метод дослідження надійності з використанням випадкового UDG-графа задовольняє основним вимогам моделювання надійності мобільних мереж та, як наслідок, дозволяє визначити оптимальні параметри робочого циклу сенсорів.

## 5. Висновки

Телемедичні системи відповідають сучасним тенденціям розвитку надання медичних послуг та реалізують засобами телемедичних комплексів важливу функцію відстежування біологічних показників у режимі реального часу в різних сферах медицини та спорту, що вказує на актуальність задач розробки телемедичного комплексу автоматизованого контролю. Даний комплекс дозволить вимірювати ізометричну силу, напругу і розслаблення м'язів в режимі реального часу, з огляду на ритм дихання, артеріальний тиск і насичення організму киснем, а також може бути здатним вести запис, передавати і зберігати інформацію. У даному комплексі пропонується використовувати сенсорну мобільну натільну мережу стандарту IEEE 802.15.6 як досить надійний бездротовий зв'язок, що підтримує великий діапазон швидкостей передачі даних, забезпечує чотирирівневу аутентифікацію передачі даних та цілісність отриманої інформації.

**Список літератури:** 1. Як застосовувати телемедицину лікарям первинної ланки: методичні рекомендації [Електронний ресурс]. Режим доступу: <https://moz.gov.ua/article/for-medical-staff/jak-zastosovuvati-telemedicinu-likarjam-pervinnoi-lanki-metodichni-rekomendacii> 2. *Телемедична* платформа Medinet [Електронний ресурс]. Режим доступу: <https://medinet.com.ua/> 3. *Зленко С. М., Павлов С. В., Коваль Л. Г., Тимчик І. С.* Основи біомедичного радіоелектронного апаратуобудування. Вінниця : ВНТУ, 2011. - 133 с. 4. *Вороненко Ю.В., Орбіна Т.М., Моїсєєнко Р.І. та ін.* Методичні рекомендації з діагностики та лікування деяких захворювань при наданні телемедичних послуг (для лікарів загальної практики - сімейної медицини). НМАПО імені П.Л. Шупика, 2019. - 104 с. 5. *IT4Medicine* [Електронний ресурс]. Режим доступу: <https://it4medicine.com.ua/> 6. *MedStar* [Електронний ресурс]. Режим доступу: <https://medstar.ua/> 7. *Законопроект "Про телемедицину"* № 10196 від 14.03.2012 р. Закон України "Про телемедицину" [Електронний ресурс]. Режим доступу: [http://search.ligazakon.ua/l\\_doc2.nsf/link1/JF7V800A.html](http://search.ligazakon.ua/l_doc2.nsf/link1/JF7V800A.html) 8. *Наказ МОЗ України № 261 від 26.03.2010 р.* Про впровадження телемедицини в закладах охорони здоров'я. [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0261282-10#Text> 9. *Doctor Online* завжди з вами [Електронний ресурс]. Режим доступу: <https://kiyivstar.ua/uk/doctor-online> 10. *Medikit* [Електронний ресурс]. Режим доступу: <https://medikit.ua/> 11. *Поліклініка без черг* [Електронний ресурс]. Режим доступу: <https://pb4.com.ua/> 12. *eZdorovya* [Електронний ресурс]. Режим доступу: <https://ehealth.gov.ua/> 13. *Владзимирський А.В., Мозговой В.В., Попова Т.В.* Телемедицина в діяльності центрів первинної медико-санітарної допомоги: монографія. Донецьк: Вид-во "Ноулдж", 2013. - 185с. 14. *Наказ МОЗ України № 681 від 19.10.2015 р.* Про затвердження нормативних документів щодо застосування телемедицини у сфері охорони здоров'я. [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/z1400-15#Text> 15. *Воробієнко П.П., Нікітюк Л.А., Резніченко П.І.* Телекомунікаційні та інформаційні мережі. К.: САММІТ-Книга, 2010. 708 с. 16. *Скляр Бернгард.* Цифровая связь. Теоретические основы и практическое применение. 2-е изд. Пер. с англ. М.: Издательский дом "Вильямс", 2003. 1104 с. 17. *Фриман Р., Слепова Н. Н.* Волоконно-оптические системы связи. Пер. с англ. М.: Техносфера, 2003. 590 с. 18. *Олифер, В. Г., Олифер Н. А.* Компьютерные сети. Принципы, технологии, протоколы. СПб.: Питер, 2010. 944 с. 19. *Лихтциндер Б.Я., Киричек Р.В., Федотов Е.Д., Голубничая Е.Ю., Кочуров А.А.* Беспроводные сенсорные сети. Под общей редакцией Б. Я. Лихтциндера. 2020. 236 с. 20. *Гольдштейн, Б. С., Кучерявый А. Е.* Сети связи пост-NGN. СПб.: БХВ-Петербург, 2013. 160 с. 21. *Кучерявый, А. Е., Прокопьев А. В., Кучерявый Е. А.* Самоорганизующиеся сети. СПб.: Типография Любавич, 2011. 22. *Смельянов В.В., Свид І.В.* Системи стільникового рухомого радіозв'язку. Харків, ТОВ "Компанія СМІТ", 2011. 23. *Huson M. L., Sen A.* Broadcast scheduling algorithms for radionetworks / Military Communications Conf, IEEE MILCOM. 1995. V. 2. P. 647-651. 24. *Zhukovskii M. Raigorodskii A.* Random graphs: models and asymptotic characteristics, Russian Mathematical Surveys. 2015. V. 70. No. 1. P. 33-81.

*Надійшла до редколегії 30.06.2021*

**Чумак Валерія Сергіївна**, магістрант кафедри біомедичної інженерії, лаборант кафедри мікропроцесорних технологій і систем ХНУРЕ. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, e-mail: [valeriia.chumak@nure.ua](mailto:valeriia.chumak@nure.ua), тел. +38 (095) 014 14 16.

**Аврунін Олег Григорович**, доктор технічних наук, професор, завідувач кафедри біомедичної інженерії ХНУРЕ. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, e-mail: [oleh.avrunin@nure.ua](mailto:oleh.avrunin@nure.ua), тел. (057) 702 13 64.

**Чугуй Євген Анатолійович**, старший викладач кафедри біомедичної інженерії ХНУРЕ. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, e-mail: [yevhen.chuhui@nure.ua](mailto:yevhen.chuhui@nure.ua), тел. +38 (096) 762 65 20.

**Свид Ірина Вікторівна**, кандидат технічних наук, доцент, завідувач кафедри мікропроцесорних технологій і систем ХНУРЕ. Адреса: Україна, 61166, м. Харків, пр. Науки, 14, e-mail: [iryna.svyd@nure.ua](mailto:iryna.svyd@nure.ua), тел. (057) 702 02 29.

## РЕФЕРАТИ

---

УДК 005.01

**Вдосконалення методології управління ІТ-послугами підприємства** / О.В. Петриченко // АСУ та прилади автоматики. 2021. № 177. С. 4-12.

В статті розглянуто питання вдосконалення існуючої методології управління послугами з метою автоматизації робіт з управління експлуатацією веб-базованих інформаційних систем управління підприємством. Виявлено основне протиріччя, що утруднює використання існуючої методології управління послугами для управління експлуатацією веб-базованих інформаційних систем. Вдосконалено існуючу методологію управління послугами шляхом введення нових концептів. Розроблено концепцію управління експлуатацією веб-базованої інформаційної системи управління підприємством. Створено візуальну модель онтології управління ІТ-послугами веб-базованої інформаційної системи. Отримано категорні моделі, які встановлюють взаємодію основних описів вдосконаленої методології управління ІТ-послугами підприємства на різних стадіях життєвого циклу управління інформаційної системи.

Табл. 1. Іл. 1. Бібліогр.: 9 назв.

UDC 005.01

**Improving enterprise IT-service management methodology** // A. Petrichenko // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag. 2021. № 177. P. 4-12.

The article addresses the issues of improving the existing service management methodology in order to automate the operation management of web-based enterprise management information systems. A major contradiction has been identified that makes it difficult to use the existing service management methodology to manage the operation of web-based information systems. The existing service management methodology has been improved by introducing new concepts. The concept of management of operation of web-based information system of enterprise management has been developed. A visual model of ontology of management of IT-services of a web-based information system has been created. There are categorical models that establish the interaction of the main descriptions of the improved methodology for managing enterprise IT services at different stages of the information system management lifecycle.

Tab. 1. Fig. 1. Ref.: 9 items.

УДК 005.01

**Дослідження структурних рішень при створенні розподіленого віртуального виробництва** / Ю.Л. Прончаков // АСУ та прилади автоматики. 2021. № 177. С. 13-17.

Поставлено і вирішено науково-прикладну задачу синтезу архітектури розподіленої віртуальної виробничої системи. Актуальність теми дослідження пов'язана зі створенням віртуальних виробництв, що відповідає концепції виробничого розвитку Industry 4.0. Метою дослідження є обґрунтування раціонального складу і топології віртуальної виробничої системи, орієнтованої на інноваційний портфель замовлень. Дослідження складається з декількох етапів: формування архітектури віртуального виробництва; моделювання виробничого циклу; дослідження вузьких місць в розподіленому віртуальному виробництві. Для оцінки розміщення технологічного обладнання використовуються показники у вигляді витрат, часу і ризиків. Проведено моделювання виробничого циклу виконання замовлень з урахуванням часових затримок на виконання операцій в технологічному обладнанні. Для моделювання вузьких місць досліджуються черги, які виникають при виконанні замовлень. Використано математичні методи: методи цілочисельного (булевого) програмування; імітаційне подієве моделювання; агентне моделювання.

Іл. 1. Бібліогр.: 7

UDC 005.01

**Research of structural solutions in the creation of the distributed virtual production** / Yu.L. Pronchakov // ACS and automation devices. 2021. № 177. P. 13-17.

The scientific and applied problem of synthesizing the architecture of the distributed virtual production system has been stated and solved. The relevance of the research topic is associated with the creation of virtual industries that corresponds to the concept of production development Industry 4.0. The aim of the study is to substantiate the rational composition and topology of the virtual production system focused on the innovative portfolio of orders. The research consists of several stages: develop the architecture of virtual production; simulate the production cycle; research the bottlenecks in the distributed virtual production. In order to assess the technological equipment location the indicators in the form of costs, time and risks are used. The simulation of the production cycle of order fulfillment taking into account the time delays for the execution of operations in the technological equipment has been performed. In order to simulate the bottlenecks the queues that appear during the execution of orders are investigated. Mathematical methods used: methods of integer (boolean) programming; simulation event modeling; agent-based modeling.

Fig. 1. Ref.: 7 items.

---

УДК 004.93

**Розпізнавання об'єктів у відеопотоці/** Є.В. Губаренко, М.С. Губаренко, М.В. Антонюк // АСУ та прилади автоматики, 2021. Вип. 177. С. 18-28.

Проаналізовані етапи процесу розпізнавання об'єктів у відеопотоці, виділені проблеми, які виникають на кожному з етапів. Запропоновано удосконалення методу еквіваріантного детектора розпізнавання об'єктів, що швидко рухаються. Створена та пройшла навчання нейронна мережа для розпізнавання об'єктів у відеопотоці. Описані рекомендації по вдосконаленню алгоритму пошуку локальних ознак при навчанні нейронної мережі. Проведено експеримент, який демонструє роботу нейронної мережі при роботі із зображеннями у різних умовах. Наведені результати розпізнавання композитних сцен.

Табл. 6. Іл. 3. Бібліогр. 19 назв.

UDC 004.93

**Image pre-processing in the video stream/** Ye.V. Hubarenko, M.S. Hubarenko, M.V. Antoniuk // Management Information System and Devices. All-Ukr.Sci.Interdep.Mag. 2021. N. 177. P. 18-28.

The stages of the process of object recognition in the video stream are analyzed, the problems that arise at each of the stages are highlighted. It is proposed to improve the method of an equivalent detector for recognizing fast-moving objects. A neural network for object recognition in a video stream has been created and trained. The recommendations on improvement of algorithm of search of local signs at training of a neural network are described. An experiment was performed to demonstrate the operation of a neural network when working with images under different conditions. The results of composite scene recognition are presented.

Tab. 6. Fig. 3. Ref.: 19 items.

---

УДК 004.42

**AWS Step Functions как неотъемлемая составляющая архитектуры системы с последовательным выполнением безсерверных функций/** О.П. Говдерчак, П.Е. Ситнікова// АСУ та прилади автоматики, 2021. Вип. 177. С. 29-35.

В статті проаналізовано використання AWS Step Functions для розробки архітектури системи з послідовним виконанням безсерверних функцій. Було розглянуто основні переваги AWS Step Functions, до яких відноситься швидка обробка даних, графічний інтерфейс переходу між станами, налаштування та контроль незалежних безсерверних функцій та його недоліки, основним з яких є складність конфігурування AWS Step Functions через використання специфічної мови Амазону, яка є власністю AWS, тому міграція з одного постачальника хмарних послуг до іншого може бути складною або неможливою. Також було розглянуто проблеми послідовної роботи безсерверних функцій без використання AWS Step Functions, найважливішою з яких являється відсутність загального стану та контролю вхідних-вихідних параметрів кожної з функцій. Досліджено ціноутворення використання AWS Step Functions.

Табл. 1. Іл. 1. Бібліогр. 9 назв.

UDC 004.42

**AWS Step Functions as an integral part of the system architecture with sequential execution of serverless functions/** O. Hovderchak, P. Sytnikova // Management Information System and Devices. All-Ukr.Sci.Interdep.Mag. 2021. N. 177. P. 29-35.

It is proposed to use AWS Step Functions to develop a system architecture with sequential execution of serverless functions. The main advantages of AWS Step Functions were considered, which include fast data processing, graphical interface of state transition, configuration and control of independent serverless functions and its disadvantages, the main of which is the difficulty of configuring AWS Step Functions due to the use of Amazon-specific AWS, so migrating from one cloud service provider to another may be difficult or impossible. Problems of sequential operation of serverless functions without the use of AWS Step Functions were also considered, the most important of which is the lack of general state and control of input-output parameters of each of the functions. The pricing of AWS Step Functions was also discovered.

Tab. 1. Fig. 1. Ref.: 9 items.

---

УДК 004.048:004.89:

**Інформаційна технологія визначення заліза в коагулянті на основі нейронної мережі/** А.П. Сафоник, І.М. Грицюк, М.М. Міщанчук, І.В. Ільків// АСУ та прилади автоматики, 2021. Вип. 177. С. 35-43.

Для визначення концентрації заліза в коагулянті за його кольором на базі нейронної мережі було розглянуто питання щодо побудови інтелектуальної системи та проаналізовано різні типи нейронних мереж. В ході аналізу обрано найбільш підходящу архітектуру нейронної мережі для вирішення задачі визначення концентрації заліза в коагулянті. Для вирішення поставленої задачі був описаний процес проектування архітектури, проведений аналіз методів навчання, підготовки даних для проведення навчання нейронної мережі, що в подальшому дасть змогу визначити концентрацію заліза в коагулянті

за його колірністю. Розроблено структурну схему функціонування нейронної мережі, яка складається із вхідного, прихованих та вихідного шарів, описано активаційні функції. Проведений аналіз точності навчання нейронної мережі, за допомогою порівняння даних, отриманих з використанням різних оптимізаторів, із використанням бібліотеки tensorflow. На основі отриманих даних аналізу був розроблений веб-додаток, що може бути використаний в якості складової інформаційно-аналітичної системи автоматизованого керування технологічним процесом електрокоагуляційної очистки стічних вод.

Табл. 3. Іл. 9. Бібліогр. 11 назв.

UDC 004.048:004.89:

**Information technology of determination of iron in coagulants based on neural network**/ A.P. Safonyk, I.M. Hrytsiuk, M.M. Mishchanchuk, I.V. Ilkiv// Management Information System and Devices. All-Ukr.Sci.Interdep.Mag. 2021. N. 177. P. 35-43.

To determine the iron concentration in the coagulant by its color on the basis of the neural network, the question of building an intelligent system was considered and different types of neural networks were analyzed. During the analysis, the most suitable neural network architecture was selected to solve the problem of determining the concentration of iron in the coagulant. To solve this problem, the architecture design process was described, the analysis of training methods, data preparation for neural network training was carried out, which will further allow determining the iron concentration in the coagulant by its chromaticity. The neural network functioning structural scheme, which input consists, hidden and output layers, is developed, activation functions are described. Accuracy analysis of neural network learning was performed by comparing data obtained using different optimizers using the tensorflow library. Based on the obtained analysis data, a web application was developed that can be used as a component of the automated control information-analytical system of the electrocoagulation treatment technological process.

Tab. 3. Fig. 9. Ref.: 11 items.

---

УДК 004.942

**Застосування моделі на основі згорткових нейронних мереж для задачі прогнозування COVID-19 в Україні** / Д.Е. Ситніков, Ю.О. Андрусенко // АСУ та прилади автоматики. 2021. № 177. С. 43-47.

В роботі розглянуто задачу прогнозування поширення COVID-19 в Україні. Актуальність цієї задачі зростає з кожним днем. Серед сучасних моделей для прогнозування часових рядів на основі згорткових нейронних мереж обрана модель InceptionTime. Для неї характерна висока точність і масштабованість. Дослідження реалізовано на мові програмування високого рівня Python. Результати представлені значеннями середньої абсолютної похибки, середньої квадратичної похибки, середньої абсолютної похибки в процентах по кожному експерименту. Модель InceptionTime демонструє високу точність та швидкодію для даної задачі.

Іл. 6. Бібліогр.: 8 назв.

UDC 004.942

**CNN-based application for Covid-19 forecasting in Ukraine** / D.Sytников, Y.Andrusenko // Management Information System and Devices. 2021. N 177. P. 43-47.

The article deals with the problem of predicting the COVID-19 spreading in Ukraine. This task becomes more relevant each day. From all the modern models of CNN-based time series forecasting, the InceptionTime model was chosen. Its advantages are high accuracy and scalability. The research is implemented using the Python high-level programming language. The results are presented as the mean absolute error, mean squared error, root-mean square error, mean absolute percentage error for each experiments. The InceptionTime model has proved its high accuracy for this task.

Fig. 6. Ref.: 8 items.

---

УДК 004.4

**Обробка подій в хмарному додатку за допомогою безсерверних обчислень** / Н.С. Кравець // АСУ та прилади автоматики. 2021. № 177. С. 47-51.

Безсерверні обчислення використовуються при розробці програмного забезпечення для сфер ритейлу, фінансів, медіа, медицини, соціальних мереж, обробки потокових даних в режимі реального часу. Розробники, що використовують безсерверні обчислення, можуть отримати економію витрат і масштабованість без необхідності володіти високим рівнем знань в області хмарних обчислень. Крім того використання безсерверної парадигми скорочує час випуску програми. Даний формат добре підходить для реалізації додатків з подієво-орієнтованою архітектурою. При обробці подій в режимі реального часу засобами FaaS проблемою може стати «холодний» старт, забезпечення надійності обробки повідомлень. Таким чином, потрібно розглядати безсерверні рішення скоріше як спосіб доповнити різні типи архітектур додатків і подієво-орієнтовану архітектуру зокрема.

Табл. 2. Іл. Бібліогр.: 3 назв.



UDC 004.4

**Processing Events in a Cloud Application Using Serverless Computing/** N.S. Kravets // Management Information System and Devices. 2021. N 177. P. 47-51.

Serverless computing is used in the development of software for the areas of retail, finance, media, medicine, social networks, and processing of streaming data in real time. Developers using serverless computing can achieve cost savings and scalability without having to have a high level of knowledge in the field of cloud computing. In addition, using the serverless paradigm reduces the application release time. This format is well suited for implementing applications with an event-oriented architecture. When processing events in real time by means of FaaS, the problem may be a «cold» start, ensuring the reliability of message processing. Thus, serverless solutions should be considered more as a way to complement different types of application architectures and event-oriented architectures in particular.

Tab. 2. Ref.: 3 items.

---

УДК 004.62

**Дослідження методів міграції даних в системах електронної комерції на прикладі CMS Magento /** М.С. Тітовської, О.В. Хряпкін // АСУ та прилади автоматики. 2021. № 177. С. 51-56.

Були визначені причини необхідності міграції між системами. В якості цільової системи для міграції даних обрана CMS Magento, описані її переваги та існуючі версії. Розглянуті способи міграції між системами електронної комерції, визначені кроки міграції та фактори ризику міграції, які можуть виникнути при обробці великих баз даних. Проаналізовані існуючі стратегії міграції даних на Magento 2. За результатами опитування експертів виділено критерій оцінювання та визначена найбільш ефективна стратегія і засіб міграції даних.

Табл. 6. Іл. 1. Бібліогр.: 14 назв.

---

UDC 004.62

**Research of data migration methods in e-commerce systems using the example of CMS Magento /** M. Titovskoi, O. Khriapkin // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag. 2021. № 177. P. 51-56.

The reasons for the need for migration between systems were identified. CMS Magento was chosen as the target system for data migration, its advantages and existing versions are also described. Methods of migration between e-commerce systems are considered, migration steps and migration risk factors that can arise when processing large databases are defined. The existing strategies for data migration to Magento 2 are analyzed. Based on the results of the expert survey, the evaluation criteria were identified and the most effective strategy and means of data migration were identified.

Tab. 6. Fig. 1. Ref.: 14 items.

---

УДК 004.94:005.4

**Розробка інформаційної технології управління системою надання ІТ-сервісів /** І.О. Юр'єв // АСУ та прилади автоматики. 2021. № 177. С. 57-63.

Розглянуто основні особливості вирішення задачі визначення набору ІТ-сервісів, які відповідають заданому набору функціональних і нефункціональних обмежень. Розглянуто застосування для вирішення цієї задачі вдосконаленого методу вибору набору сервісів. Визначено основні особливості інформаційної технології управління системою надання ІТ-сервісів і виділені її основні стадії. Розроблено візуальні моделі інформаційної технології, основних функціональних задач і структур даних інструментального засобу, що реалізує дану технологію.

Іл. 5. Бібліогр.: 9 назв.

---

UDC 004.94:005.4

**Developing IT Service Management Information Technology /** I.A. Iuriev // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag. 2021. № 177. P. 57-63.

The main features of solving the problem of determining a set of IT services satisfying a given set of functional and non-functional limitations are considered. An improved method of selecting a set of services is considered to solve this problem. The main features of IT services system management information technology are defined and its main stages are highlighted. Visual models of information technology, basic functional tasks and data structures of the tool implementing this technology have been developed.

Fig. 5. Ref.: 9 items.

---

УДК 004.652.4

**Композиційне проектування та реструктуризація глобальної схеми мультибази даних /** Т.Г. Білова, В.М. Дьоміна, І.О. Побіженко // АСУ та прилади автоматики. 2021. № 177. С. 63-68.

Визначено особливості композиційного проектування та реструктуризації глобальної схеми мультибази даних. Розглянуто різні типи представлення глобальної схеми, виявлено їх переваги та недоліки.

Сформульована задача композиційного проектування «знизу до гори» як процес послідовного додавання нових вузлів. Описані процеси видалення вузлів та оновлення, що охоплює один або декілька вузлів. Розроблено механізми управління процесом реструктуризації глобальної схеми.

Табл. 1. Ил. 1. Бібліогр.: 5 назв.

UDC 004.652

**Composite design and restructuring of the global multi-database schema** / Т. Bilova, V. Dyomina, I. Pobizhenko // Management Information System and Devices. 2021. N 177. P.63-68.

The features of compositional design and restructuring of the global multi-database schema are determined. Various types of representation of the global scheme are considered, their advantages and disadvantages are revealed. The problem of compositional design «from the bottom up» is formulated as a process of sequential addition of new nodes. Node deletion and update processes that span one or more nodes are described. Mechanisms for managing the process of restructuring the global scheme are developed.

Tab. 1. Fig. 1. Ref.: 5 items.

---

УДК 004.891.3

**Представлення узгоджених знань з урахуванням їх логічної несуперечливості для задачі побудови пояснень в інтелектуальних системах** / С.Ф. Чалий, В.О. Лещинський, І.О. Лещинська // АСУ та прилади автоматики. 2021. № 177. С. 69-74.

Метою дослідження є розробка моделі узгоджених знань щодо процесу прийняття рішення в інтелектуальних системах, яка враховувала б їх логічну несуперечливість та забезпечувала можливість формування пояснення процесу та результату роботи таких систем. Об'єктом дослідження є процеси побудови пояснень в інтелектуальних системах. Задачі дослідження: розробка узагальненого підходу до узгодження знань для побудови пояснень в інтелектуальних інформаційних системах; розробка моделі представлення узгоджених знань з урахуванням їх логічної несуперечливості. Отримано наступні результати. Розроблено узагальнений підхід до узгодження знань для побудови пояснень в інтелектуальних інформаційних системах. Підхід забезпечує можливість формування множини узгоджених пояснень, що відображають різні аспекти процесу прийняття рішення та отриманого інтелектуальною системою результату, що спрощує застосування цих результатів у предметній області користувача. Запропоновано модель представлення узгоджених знань з урахуванням їх логічної несуперечливості. Модель дає можливість перевірити узгодженість знань безпосередньо при формуванні пояснень, що створює умови для формування пояснень за принципом чорного ящика та дає можливість доповнити функціонуючі системи можливостями пояснень, а також адаптувати пояснення при зміні їх функціональності.

Л. 1. Бібліогр. : 14 назв.

UDC 004.891.3

**Representation of coherence knowledge considering their logical inconsistency for the problem of constructing explanations in intellectual systems** / S.F. Chalyi, V.O. Leshchynskyi, I.O. Leshchynska // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag. 2021. № 177. P. 69-74.

The aim of the study is to develop a model of consistent knowledge about the decision-making process in intelligent systems, which would consider their logical consistency and provide an opportunity to explain the process and outcome of such systems. The object of study is the process of constructing explanations in intelligent systems. Research objectives: development of a generalized approach to the harmonization of knowledge to build explanations in intelligent information systems; development of a model for the presentation of consistent knowledge, considering their logical consistency. The following results were obtained. A generalized approach to the harmonization of knowledge to build explanations in intelligent information systems has been developed. The approach provides the possibility of forming a set of agreed explanations that reflect different aspects of the decision-making process and the result obtained by the intelligent system, which simplifies the application of these results in the subject area of the user. The model of representation of the coordinated knowledge considering their logical consistency is offered. The model makes it possible to check the consistency of knowledge directly in the formation of explanations, which creates conditions for the formation of explanations on the principle of the black box and allows to supplement functioning systems with explanations, as well as to adapt explanations when changing their functionality.

Fig. 1. Ref.: 14 items.

---

УДК 005.01

**Згорткова подвійна радіально-базисна нейронна мережа на основі активаційних ядерних функцій спеціального типу** / О.С. Чала // АСУ та прилади автоматики. 2021. № 177. С. 75-79.

У статті запропонована згорткова подвійна нейронна мережа, що має ядерні функції активації та призначена для вирішення задач розпізнавання та класифікації зображень в режимі онлайн. Ця мережа

містить дві блоки, де перший це нейрокомпесор, що за суттю є модифікованою ймовірнісною нейронною мережею і отримує на вхідний шар зображення, представлені у первинній матричній формі. На виходах автоенкодера, кількість яких визначається кількістю класів у наборі даних, з'являється вектор сигналів попередньої класифікації. Отриманий вектор-сигнал надходить на вхід другого блоку - радіально-базисної нейронної мережі, яка утворює розділяючі гіперповерхні між класами довільної складної форми.

Відмінною рисою розробленої системи є комбіноване навчання наведених двох блоків, де налаштування першого відбувається за допомогою лінивого навчання, базуючись на концепції «Нейрони в точках даних», а система розпізнавання - за парадигмою навчання з учителем та у результаті налаштовуються синаптичні ваги й центри та рецепторні поля функцій активації. Завдяки такому підходу до навчання, система характеризується не тільки високою точністю класифікації, а також високою швидкістю навчання, що дозволяє обробляти потоки даних, які послідовно подаються в режимі онлайн.

Л. 1. Бібліогр. : 20 назв.

UDC 005.01

**Convolutional Double Radial-basis Neural Network Based on Special Type of Kernel Functions** // OS Chala // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag. 2021. № 177. P. 75-79.

In the article, the convolutional double neural network that has kernel activation function was proposed. It is designed to solve image recognition-classification task in online mode. The neural network has two blocks, where the first one is autoencoder, and as a matter of fact is a modified probabilistic neural network and receives on its input images in their initial matrix form. The output is represented as a vector-signal of the pre-classification, where the length of the vector is defined by the number of classes in a dataset. Then this vector is fed to the input layer of the second block - radial-basis neural network that forms separating hypersurfaces between arbitrary complex form classes.

The distinctive feature of the proposed system is the combined learning of both blocks of the system, where the first one is tuned by the paradigm of lazy learning, basing on the concept «Neurons at datapoints», and the recognition system follow by the paradigm of supervised learning where synaptic weighs, centers and receptive fields of activation functions are tuned. Following this learning approach, the system is characterised by high accuracy and learning speed, which allows classification of data streams as the sequence of images that are fed in the system in online mode.

Fig. 1. Ref.: 20 items.

УДК 004.77:621.398:614.2

**Аналіз принципів побудови телемедичних комплексів широкого призначення** / В.С. Чумак, О.Г. Аврунін, Є.А. Чугуй, І.В. Свид // АСУ та прилади автоматики. 2021. № 177. С. 80-85.

Розглянуто принципи побудови існуючих телемедичних комплексів та систем. Розроблено загальну структурну схему надання телемедичних послуг. Запропоновано загальну структурну схему побудови портативного та мобільного телемедичних комплексів. Обґрунтовано вибір та переваги застосування в телемедичних комплексах сенсорної мобільної натільної мережі стандарту IEEE 802.15.6. Запропоновано метод дослідження надійності сенсорної мережі та визначення оптимальних параметрів робочого циклу сенсорів, з використанням випадкового UDG-графа.

Л. 5. Бібліогр.: 23 назв.

UDC 004.77:621.398:614.2

**Analysis of the principles of construction of telemedicine complexes of general purpose** / V.S. Chumak, O.G. Avrunin, Y.A. Chuhui, I.V. Svyd // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag. 2021. N 177. P. 80-85.

The principles of construction of existing telemedicine complexes and systems are considered. A general structural scheme for the provision of telemedicine services has been developed. The general structural scheme of construction of portable and mobile telemedicine complexes is offered. The choice and advantages of application of IEEE 802.15.6 standard mobile body network in telemedicine complexes are substantiated. A method for studying the reliability of the sensor network and determining the optimal parameters of the operating cycle of sensors, using a random UDG-graph.

Fig. 5. Ref.: 23 items.

**ПРАВИЛА**  
**оформления рукописей для авторов научно-технического сборника**  
**"АСУ и приборы автоматики"**

Формат страницы — А4 (210x297мм), поля: сверху, справа, слева, снизу – 30 мм. Редактор: Pagemaker 6.0, 6,5 (можно, но нежелательно Word), гарнитура Times New Roman Суг, кегль – 11 пунктов, межстрочное расстояние — 110 %, табуляция — 5 мм.

Объем рукописи – до 10 с. (языки: русский, украинский, английский). Содержание должно отражать актуальность исследования, постановку задачи, цель, сущность, научные и практические результаты, сравнение с лучшими аналогами, выводы.

Структура рукописи: заголовок, аннотация, текст, литература, реферат на украинском и английском языках, сведения об авторах.

**ОБРАЗЕЦ ОФОРМЛЕНИЯ**

---

УДК 519.713

*И.О. ФАМИЛИЯ*

**НАЗВАНИЕ РУКОПИСИ**

---

Аннотация (абзац 5-10 строк, кегль 10) помещается в начале статьи и содержит информацию о результатах описанных исследований.

Основной текст можно разделять на 2 и более подразделов с заголовками, выделенными полужирным шрифтом, пронумерованными арабскими цифрами, как показано в следующей строке.

**1. Название раздела**

Рисунки и таблицы (черно-белые, контрастные) помещаются в текст после первой ссылки в виде *переносимых объектов* и отдельно нумеруются, при наличии более одного рисунка (таблицы), арабскими цифрами. Рисунок содержит подрисуючную центрированную подпись (текстовая строка, расположенная вне рисунка, кегль 10) под иллюстрацией, как показано на рис. 1.

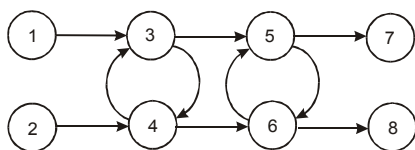


Рис. 1. Граф с контурами

Табличный заголовок располагается справа над таблицей, что иллюстрируется табл.1. Редакторы: CorelDraw, Table Editor и др.

Формулы нумеруются при наличии ссылок на них в рукописи. Рекомендуемый кегль формульного набора: обычный (переменная) – 11 пунктов, крупный индекс – 8, мелкий индекс (над- и подиндекс) – 8, крупный символ (основной) – 12, мелкий (индексный) математический символ – 10:

Таблица 1

Ш ar i	1	2	3	4	5	6
$\Phi_1(1,3)$	1	2	2	4	6	1
$\Phi_1(1,4)$	1	2	2	4	4	8

$$F_{i+j} = \sum_{i=1}^{b^k} F_j^i - \prod_{j=1}^{1+h^2} P_{R_{j+i}} + F^{j-1} + X^{\Sigma n^k} \quad (1)$$

Формат переменных (желательно не курсивом – без наклона) в тексте и формулах должен быть идентичным. В тексте над- и подиндексы составляют 70 % от кегля, которые рекомендуется опускать (поднимать) на 17 (33) % относительно основной строки.

Список литературы (включает опубликованные источники, на которые имеются ссылки в тексте, заключенные в квадратные скобки) печатается без отступа, кегль 9 пунктов.

Образец окончания текста рукописи (литература, сведения об авторах, реферат) представлен ниже.

**Список литературы:** 1. *Фамилия И.О.* Название книги. Город: Издательство, 1900. 000 с. 2. *Название сборника / Под ред. И.О. Фамилия.* Город: Издательство, 1900. 000 с. 3. *Фамилия И.О.* Название статьи / Название журнала. Название серии. 2000. Т. 00, № 00. С. 00-00 .

*Поступила в редколлегию 00.00.00*

**Фамилия, имя, отчество**, ученая степень, звание, должность и место работы. Научные интересы. Адрес, контактный телефон.

Рефераты на украинском и английском языках. Текст аннотации не должен дублировать реферат.

---

УДК 000.000.00

**Назва статті українською мовою / Ініціали. Прізвище // АСУ та прилади автоматики.** 2000. Вип. 00. С. 000-000.

Текст реферату.

Табл. 00. Іл. 00. Бібліогр.: 00 назв.

UDC 000.000.00

**Title of paper / Initials. Surname // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag.** 2000. N 00. P. 000-000.

Text.

Tab. 00. Fig. 00. Ref.: 00 items.

#### Представление материалов

Рукопись, реферат, сведения об авторах — в одном файле, *поименованном фамилией первого автора*. Твердая копия материалов – для граждан Украины — в одном экземпляре: рукопись, подписанная авторами, рефераты, акт экспертизы, внешняя рецензия, подписанная доктором наук, заявление на имя главного редактора со сведениями об авторах.

Адрес редакции: Украина, 61166, Харьков, просп. Науки, 14, ХНУРЭ, комната 254, тел. (057) 702 14 51.

E-mail: [misd@nure.ua](mailto:misd@nure.ua).

Тематика статей, публикуемых в сборнике:

- Компьютерная инженерия
- Математическое моделирование
- Оптимизация и процессы управления
- Автоматизация проектирования и диагностика
- Информационные интеллектуальные системы
- Проектирование интегральных схем и микросистем
- Компьютерные технологии в образовании

Відповідальний випусковий В.М. Левикін  
Редактор О.Є. Неумивакіна  
Комп'ютерна верстка М.В. Євланов, О.Є. Неумивакіна  
Дизайн обкладинки номера за участі Є. Чех

Підп. до друку 30.06.2021. Формат 60x841/8. Умов. друк. арк. .  
Обл.-вид. арк. 10,5. Тираж 300 прим.  
Зам. № . Ціна договірна.

---

Харківський національний університет радіоелектроніки (ХНУРЕ).  
Україна, 61166, Харків, просп. Науки, 14.

---

Оригінал-макет підготовлено у редакційно-видавничому відділі ХНУРЕ.  
Україна, 61166, Харків, просп. Науки, 14.

---

Збірник віддруковано в ТОВ «ДРУКАРНЯ МАДРИД»  
61024, м. Харків, вул. Гуданова, 18  
Тел.: (057) 756 53 25

Свідоцтво суб'єкта видавничої справи  
Серія ДК № 4399 від 27.08.2012 р.  
[www.madrid.in.ua](http://www.madrid.in.ua) e-mail: [info@madrid.in.ua](mailto:info@madrid.in.ua)