www.konferenciaonline.org.ua

Міжнародна наукова інтернет-конференція

Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення

(випуск 39)

ISSN 2522-932X

11 червня 2019 р.

Тернопіль
2019

Міжнародна наукова інтернет-конференція "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення (випуск 39)" / Збірник тез доповідей: випуск 39 (м. Тернопіль, 11 червня 2019 р.). – Тернопіль. – 2019. – 140 с.

УДК 001 (063)
ББК 72я431

ISSN 2522-932X

Збірник тез доповідей підготовлено за матеріалами Міжнародної наукової інтернет-конференції (випуск 39) від 11 червня 2019 р.

Збірник матеріалів науково-практичної інтернет-конференції включаються до наукометричної бази даних "РІНЦ/RSCI".

Тексти матеріалів конференції подаються в авторській редакції. Відповідальність за точність, достовірність і зміст поданих матеріалів несуть автори.

Наша адреса: Оргкомітет МНІК "Конференція онлайн"
а/с 797, м. Тернопіль 46005
тел. моб. 068 366 0 525
e-mail: inetkonf@ukr.net

URL Інтернет-конференції: http://www.konferenciaonline.org.ua/

Всі права захищенні. При будь-якому використанні матеріалів конференції посилання на джерело є обов’язкове.

Секція 1. Інформаційні системи і технології

Аргунова О.В., Литовченко О.В.
Коледж інформаційних технологій та землевпорядкування
Національного авіаційного університету, Київ
Комісія природничих дисциплін, викладач

ВПЛИВ МЕРЕЖІ ІНТЕРНЕТ НА НАВЧАННЯ ФІЗИКИ

Всесвітня комп'ютерна мережа Інтернет створила нові можливості, яких не було у відокремлених комп'ютерів. Спостерігається її вплив на найрізноманітніші елементи системи навчання фізиці – від підручника та експерименту до створення та структури навчального предмету та діяльності викладача.

Передусім, мережа змінила навчальний текст. У студента з'явилася можливість вносити зміни безпосередньо в отриманий матеріал, тому що навчальний текст став одночасно робочим зошитом. Підручник перетворився в «мультимедійний конструктор», і цим дав можливість викладачу чи студенту ілюструвати текст. Іншими словами, підручник став не лише діалоговим (інтерактивним), а й динамічним.

Інтернет якісно змінює й навчальний експеримент. Після появи мікрокомп'ютерної лабораторії почалося її об'єднання з світовою мережею. Виник новий тип лабораторії, прикладом якої є лабораторія Чарльза в Празі, яка пропонує способи для запису, вимірювання та моделювання різних навчальних експериментів. Лабораторія допомагає навчальному експерименту і концентрує данні з світової мережі [2].

Інтернет оновлює віртуальний експеримент. Найкрупніші дослідні центри, обсерваторії, станції спостереження пропонують освіті отримані ними останні данні експериментів та спостережень. В якості приклада можна привести вузли Національного агентства по аеронавтиці та дослідженню космосу США, лабораторії ім. Е. Ферми, Європейського центра ядерних досліджень. Ці данні а також відеоінформація з дослідних центрів залучаються до навчального фізичного моделювання.

Світова мережа відкриває нові можливості і для віртуальних екскурсій. Більшість наукових музеїв створюють вузли для віддалених відвідувачів з відео експозиціями та цілими освітніми програмами (один з таких прикладів – науковий музей «Експлораторіум» в Сан-Франциско).

Відкриті освітні вузли при великих обсерваторіях, при дослідних фізичних центрах. З 1997 р. в США в межах проекту «Віртуальний клас», створеного НАСА, проводяться спільні заняття учнів різни шкіл та конференції вчених.

В підсумку, інформаційні ресурси світової мережі починають виконувати освітні функції. Це укріплює педагогічну функцію спеціально створених для цього систем освіти. Аналіз сучасних тенденцій розвитку освітніх матеріалів по фізиці показує, що Інтернет починає об'єднує всі існуючі освітні системи в

єдине фізичне освітнє середовище, наприклад, таке інтегроване співтовариство створено в Мерилендському університеті.

Такі співтовариства створюються професійними фізиками-дослідниками, дослідниками в області фізичної освіти, викладачами фізики, студентами. Наприклад, дослідники фізичної освіти різних країн Європи за допомогою мережі об'єдналися в міжнародну групу, подібну групу створили в США; викладачів фізики Англії об'єднує вузол Інституту фізики.

Використання ресурсів та технологій глобальної мережі Інтернет дає можливість здійснювати Інтернет-освіту та дистанційну освіту. Інтернет-освіта у відношенні до поняття дистанційна освіта є видовою відмінністю, більш строго регламентує техніко-технологічну специфіку навчання – використання мережі Інтернет (дистанційно можна навчатися не тільки через Інтернет, але і за допомогою локальних мереж, відеозв'язку тощо).

Дистанційне навчання – сукупність сучасних технологій, що забезпечують доставку інформації в інтерактивному режимі за допомогою використання ІКТ (інформаційно-комунікаційних технологій) від тих, хто навчає (викладачів, визначних постатей у певних галузях науки), до тих, хто навчається (студентів чи слухачів).

В Україні є декілька прикладів великих навчальних центрів з дистанційною формою навчання, що спеціалізуються на підготовці учнів та студентів. Одним з найбільших є Prometheus – проект масових відкритих онлайн-курсів [4]. Українська платформа онлайн-курсів Prometheus це пілотний проект зі змішаним типом навчання. Основна ідея в тому, що студенти будуть онлайн переглядати курси кращих світових університетів, а викладачі в університетах будуть приділяти більшу роль індивідуальній роботі, застосуванню та контролю знань.

Світова мережа, об'єднавши школи, вузи, дослідні центри, бібліотеки, музеї, дає можливість викладачам та студентам отримувати цінні освітні матеріали. Одночасно цей зв'язок допомагає вузам і дослідним центрам зрозуміти потреби школи.

Споживання інформації стає по суті діалоговим, таким чином створюється можливість індивідуально створювати навчальний процес, орієнтуючись на конкретного студента. Динамічний навчальний предмет відповідає стратегічному завданню створення унікальної освіти для кожного.

Література:

1. Бужиков Р. П. Дидактичний потенціал Інтернет-технологій в сучасній системі освіти/Р. П. Бужиков//Проблеми освіти: наук. збірник Ін-ту інновац. технологій і змісту освіти МОНМС України. –К., 2011. –Вип. 66. –Ч. ІІ., с. 41 –42.
2. Криштоф С. Д. Використання Інтернет-ресурсів для підвищення якості навчання дисциплін природно-математичного циклу в загальноосвітній школі. Наукові записки. Серія: Педагогіка. –2011. –No5. –134, 136 –137 с. [Електронний ресурс]. –Режим доступу: http://dspace.tnpu.edu.ua/bitstream/123456789/1671/1/Krushtof.pdf
3. Литвинова С. Г., Тебенко О. В. Хмарні технології. Соціальне середовище програмування TouchDevelop. Науково-методичний журнал «Комп'ютер у школі та сім'ї». –2013. –No5. –с. 26.
4. https://prometheus.org.ua/

Бенедицький В.Б, *старший викладач*
Волошинська М.П., *студентка 4-го курсу*
Коренівська О.Л., *к.т.н., доцент*
Державний університет «Житомирська політехніка», м. Житомир
Кафедра біомедичної інженерії та телекомунікацій

СИСТЕМА «РОЗУМНА ТЕПЛИЦЯ» НА АРДУІНО

Застосування інформаційних технологій та автоматизація процесів у сільскому господарстві в Україні впроваджується згідно «Концепції загальнодержавної цільової програми розвитку промисловості» схваленої постановою Кабінету Міністрів України №1174 від 28 липня 2003 року.

Метою роботи є розробка системи керування грибною теплицею для досягнення максимальної продуктивності та мінімізації затрат на виробництво.

Теплиці дозволяють вирощувати гриби круглий рік, а також наносять мінімальну шкоду природі. Грибне виробництво дозволяє збирати до 20 кг грибів з 3 кв.м. за місяць. Вирощування відбувається в ящиках, які для економії місця можна ставити на стелажі в 3 поверхи.

Перерахуємо основні елементи обладнання теплиці та вимоги до них:

1. **Опалювальна система**, переважного водяного типу, яку дуже легко підключити від опалення будинку за допомогою вентиля балансування. Потужність опалювального обладнання залежить від розмірів теплиці.
2. **Система вентиляції**. Гриби дуже люблять свіже повітря. Система вентиляції підтримує рівень вологості і температури та покращує обмін повітря. Також важливе правильне провітрювання теплиці: кватирки повинні розташовуватися у верхній частині парника, щоб не допустити холодні протяги і переохолодження рослин.
3. **Система зволоження та поливу**. Грибниці необхідно часто зволожувати і робити це методом розпилення води. Найчастіше використовують крапельний полив зі спеціальними розпилювачами.
4. **Освітлення.** Для вирощування глив досить буде і денного світла навіть взимку. Ці гриби не вимагають освітленості. Зовсім інша ситуація з розведенням в теплиці печериць, яким потрібно багато світла. Як правило, в грибних теплицях використовуються лампи денного світла з розсіяним освітленням, так як грибниці не переносять прямого попадання світлових променів. Кожна лампа має свою траєкторію, тому їх кількість визначається площею теплиці.
5. Для підтримки **оптимальної вологості** рекомендується регулярно провітрювати теплицю та встановити систему штучного зволоження. Підвищити вологість можна завдяки поливу доріжок з шланга, і розбризкування води – ця система монтується вгорі теплиці і розпорошує дрібні крапельки води по парники. Завдяки підвищенню вологості в повітрі, температура в парнику буде знижуватися.

Таким чином базовий проект саморобної «розумної» теплиці дозволяє автоматично виконувати наступні функції:

- контроль і регулювання температури всередині теплиці;
- моніторинг вологості повітря та зволоження ґрунту;
- підтримання та регулювання освітлення в теплиці;
- керування провітрюванням;
- керування вентиляцією теплиці.

Температура в теплиці повинна бути в межах від 16 °С до 25 °С, відносна вологість ґрунту від 65% до 85%, вологість повітря 50-60%, освітленість 6-10 тис. лк.

На рис. 1 наведено структурну схему розумної теплиці.

В якості сенсорів використано модулі DHT11 – сенсор температури і вологості повітря для Arduino, ECH$_2$O - 5TE – сенсор температури та вологості ґрунту (визначає об'ємний вміст води, температуру ґрунту та питому електропровідність і призначений для усіх типів ґрунтів), MH-Z19 – сенсор вуглекислого газу, цифровий сенсор освітленості GY-302.

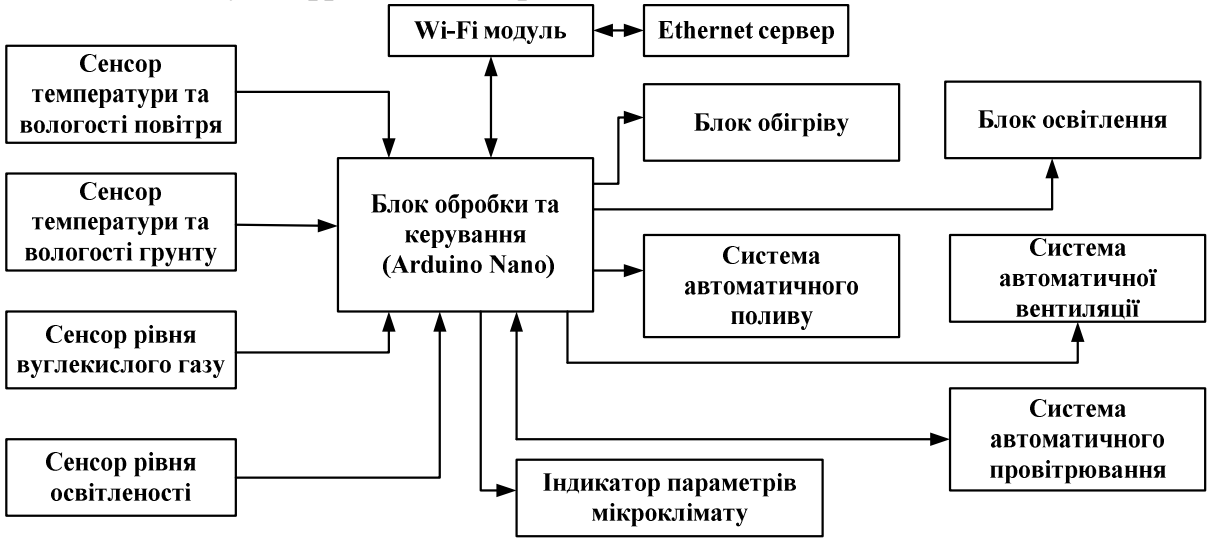


Рис. 1. – Структурна схема розумної теплиці

Основним модулем збору інформації та блоком керування є платформа Arduino Nano на мікроконтролері ATmega2560. Він відповідає за автономне управління мікрокліматом теплиці, а саме зчитує через модуль інтерфейсів показники сенсорів та згідно із внутрішньою програмою керує системами встановлення мікроклімату. Внутрішня програма аналізує межі необхідних параметрів мікроклімату та формує керівні сигнали. У разі наявності зв'язку із мобільним пристроєм управління мікроконтролер перевіряє, чи наявні зміни у параметрах управління теплицею та перезаписує конфігураційний файл. Керування модулем, встановлення параметрів мікроклімату та перегляд поточних даних можливе через мережу Інтернет на ПК або смартфоні. Це дозволяє контролювати та керувати параметрами мікроклімату теплиці на відстані.

Література:
1. ВНТ.П-СГіП–46-19-96. Тепличні і оранжерейні підприємства. Споруди захищеного ґрунту для фермерських господарств. – К.: Мінсільгосппрод України, 1996. – 68 с.
2. Офіційний сайт компанії Arduino. – [Електронний ресурс]. – Режим доступу: http:/arduino.ua/
3. Д. Блум. Изучаем Arduino. Инструменты и методы технического волшебства. / Блум. Д. — СПб.: БХВ-Петербург, 2015. – 336 с.

Богданенко М.О., *студент*
Київський Політехнічний Інститут ім. Ігоря Сікорського, м. Київ
Кафедра автоматизованих систем обробки інформації та управління, студент

ОПТИМІЗАЦІЯ ПРОЦЕСУ МАЙНІНГУ

У цій статі буде наведено спосіб оптимізації процесу майнінгу[1] для отримання збільшеного прибутку власниками майнінг-обладнання. Для розуміння, чому цей спосіб працює, розглянемо наступний список фактів:

1) Різні криптовалюти[2] мають різні ціни;
2) Різні криптовалюти мають різні складності мереж;
3) Ціни криптовалют змінюються у часі;
4) Складність мереж криптовалют змінюється у часі;
5) Не всі криптовалюти приносять прибуток при майнінгу;

З цих фактів можна зробити наступний висновок: «Якщо майнити у певний проміжок часу певну криптовалюту, міняючи її на більш вигідну коли це можливо, можна отримати більший прибуток, ніж якщо майнити одну й ту ж саму криптовалюту увесь доступний час».

Сама ідея зміни криптовалюти не нова, проте її реалізація стикається із складним інтерфейом програм-майнерів та необхідністю виконувати переналаштування на всіх компютерах. Нажаль, переналаштування майнера – процедура кропітка, оскільки майнери – це не програми зі зручним інтерфейсом. В основному майнери – це програми у вигляді консольного застосунку і кожна така програма має свій уніальний інтерфейс взаємодії. Через часозатратність процесу переналаштування власники нечасто виконують переналаштування, і втрачають можливість заробити чи хоча б не втратити гроші на окупівлі витрат на електроенергію.

Для вирішення цієї проблеми має сенс створити використовувати систему автоматизації майнінгу – спеціальну програмну систему, що дозволить автоматично переналаштовувати обладнання.

Висновок

Для покращення показнику прибутку від майнінгу має сенс часто проводити аналіз ринку та переналаштовувати обладання для отримання прибутку або, принаймні, для того щоб витрати на майнінг не перевищували прибуток. Саме підвищені витрати спричиняють відмову від майнінгу, тому

вирішення цієї проблеми – дуже важливий крок до покращення ситуації з криптовалютами.

Література:

1. Що таке майнінг [Електронний ресурс] – Режим доступу до ресурсу: https://prostocoin.com/blog/what-is-mining.
2. Що таке криптовалюта? [Електронний ресурс] – Режим доступу до ресурсу: https://www.radiosvoboda.org/a/details/28742278.html.

Богурський Д.О.
Національний технічний університет України
"Київський Політехнічний Інститут ім. Ігоря Сікорського", м. Київ
Кафедра автоматизованих систем обробки інформації та управління,
студент

ГІБРИДНИЙ АЛГОРИТМ МЕТОДУ ІТЕРАЦІЙ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧ СТІЙКОСТІ КОНСТРУКЦІЙ

Вступна частина

У наш час у сфері високопродуктивних обчислень широкого застосування набувають гібридні комп'ютери, архітектура яких поєднує багатоядерні процесори (CPU) з розподіленою пам'яттю та масивно – паралельні прискорювачі обчислень — графічні процесори (GPU).

Велика кількість наукових та практичних задач, зокрема, при дослідженні стійкості конструкцій, розрахунку динаміки напружено – деформованого стану об'єктів різної природи та ін., зводяться до розв'язання часткової алгебраїчної проблеми власних значень стрічкових симетричних додатно – означених матриць великої розмірності. Застосування гібридних комп'ютерів для розв'язування таких задач потребує створення алгоритмів, які враховують унікальні архітектурні та обчислювальні особливості цих комп'ютерів.

Гібридний алгоритм методу ітерацій для стрічкових матриць

Розглядається алгебраїчна проблема власних значень (АПВЗ) для стрічкових симетричних додатно – означених матриць:

$$Ax = \lambda Bx,\ A, B \in M^{n\times n},\ x \in Rn,\ \lambda \in R, \qquad (1)$$

де $M^{n\times n}$ — множина квадратних матриць порядку n. Метод ітерацій на підпросторі є узагальненням методу обернених ітерацій і полягає у побудові послідовності підпросторів $Et\ (t = 1, 2, ...)$, яка зводиться до підпростору E∞, що містить шукані власні вектори. В методі ітерацій на підпросторі на t-й ітерації обчислюється ортонормований базис підпростору Et, а також, якщо досягнута збіжність, то визначаються шукані власні пари. Детальний опис методу представлено в [1].

Розроблено гібридний алгоритм методу ітерацій на підпросторі для розв'язування АПВЗ задачі (1), за яким розпаралелювання обчислень на CPU здійснюється в середовищі MPI, а на GPU — за технологією CUDA.

Елементи (головної діагоналі та піддіагональні) стрічкових симетричних матриць A та B розподіляються між процесами CPU за одновимірною блочно – циклічної схемою [2], за якою процесу з логічним номером i розподіляються елементи рядків з номерами $js - s + 1, \ldots, js, js - s + 1 + ps, \ldots, js + ps, js - s + 1 + 2ps, \ldots$, а процесу з номером $i + 1$ — рядки $js + 1, \ldots, js + s, js + 1 + ps, \ldots, js + s + ps, js + 1 + 2ps, \ldots$, де p — кількість процесів, що використовуються, s — кількість рядків у блоці, а j — номер блоку. Цей розподіл для кожного процесу визначається номером першого рядка, елементи якого розподіляються даному процесу.

Зважаючи на те, що операції на GPU виконуються в рамках MPI процесу, розподіл даних аналогічний блочно – циклічній схемі.

Реалізація гібридного алгоритму методу ітерацій на підпросторі базується на факторизації (з використанням GPU) стрічкової матриці A плитковим гібридним алгоритмом LLT-розвинення [3].

На кожній ітерації виконуються наступні операції:

- розв'язування системи лінійних алгебраїчних рівнянь (розв'язується кожним процесом, використовуючи попередньо факторизовану матрицю)

$$AX_t = Y_{t-1}; \quad (2)$$

- обчислення проекції матриці A на підпростір Et (виконується на GPU)

$$A_t = X_t^T Y_{t-1} \equiv X_t^T A X_t; \quad (3)$$

- обчислення прямокутної матриці (виконується з використанням GPU)

$$W_t = BX_t; \quad (4)$$

- обчислення проекції матриці B на підпростір Et (виконується з використанням GPU)

$$B_t = X_t^T W_t \equiv X_t^T B X_t; \quad (5)$$

- розв'язування проблеми власних значень для проекцій (розв'язується кожним процесом незалежно)

$$A_t Z_t = B_t Z_t \Lambda_t; \quad (6)$$

- обчислення наступного наближення (за рахунок розподіленості даних між процесами, операції виконуються паралельно на CPU)

$$Y_t = W_t Z_t; \quad (7)$$

- перевірка умов закінчення ітераційного процесу, аналогічно до (6):

$$\frac{\left|\lambda_i^{(t)} + \lambda_i^{(t-1)}\right|}{\lambda_i^{(t)}} \leq \varepsilon \; (i = 1, 2, \ldots, r);$$

Якщо умова (8) виконується після t ітерацій, то наближеним розв'язком задачі вважається:

$$\lambda_i^* = \lambda_i^{(t+1)}, X^* = X_{t+1} Z_{t+1} \; (i = 1, 2, \ldots, r);$$

Тут, як і при перевірці умов закінчення ітераційного процесу, мається на увазі, що власні значення впорядковано за зростанням $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_r \ldots$

Результат роботи гібридного алгоритму — обчислені власні значення λі (розташовані у кожному процесі в порядку зростання) та розподілена між процесами матриця X відповідних власних векторів у відповідності до розподілу матриць A та B.

Висновки.

У роботі розглянуто гібридний алгоритм методу ітерацій на підпросторі для розв'язування часткової АПВЗ для симетричних додатно – означених стрічкових матриць на гібридних комп'ютерах. Створений алгоритм забезпечує високу швидкодію і масштабованість на комп'ютері гібридної архітектури. Програми, що реалізують розроблені гібридний алгоритми, увійшли до складу штатного програмного забезпечення комп'ютерів гібридної архітектури сімейства Інпарком-G та використовувалися для розв'язування практичних науково – технічних задач, зокрема для розрахунку задач стійкості композитних матеріалів при одноосьовому стисненні слоїв наповнювача.

Література:

1. Молчанов И. Н., Попов А. В, Химич А. Н. Алгоритм решения частичной проблемы собственных значений для больших профильных матриц. Кибернетика и системный анализ. 1992. № 2. С. 141–147.
2. Химич А. Н., Молчанов И. Н., Попов А.В., Чистякова Т.В., Яковлев М. Ф. Параллельные алгоритмы решения задач вычислительной математики. Киев: Наук. думка, 2008. 248 с.
3. Хіміч О. М., Баранов А. Ю. Гібридний алгоритм розв'язування лінійних систем зі стрічковими матрицями прямими методами. Компьютерная математика. 2013. Вып. 2. С. 80–87.
4. Декрет В. А., Зеленский В. С., Быстров В. М. Численное исследование устойчивости слоистого композитного материала при одноосном сжатии наполнителя. Киев: Наук. думка, 2008. 248 с.

Бойко О.Р., *к. т. н., доцент*
Мельник Т.В., *студент*
Вінницький національний технічний університет, м. Вінниця
Кафедра Автоматизації та інтелектуальних інформаційних технологій

АВТОМАТИЗОВАНА СИСТЕМА ПЛАНУВАННЯ ПАСАЖИРСЬКИХ ПЕРЕВЕЗЕНЬ

Рівень розвитку і якісного функціонування пасажирських перевезень значно впливає на внутрішньополітичну та соціальну стабільність суспільства, що ставить підвищені вимоги до цієї сфери здійснення виробничої діяльності. Основним видом міського пасажирського транспорту є автобусний транспорт. Він здатний щодня забезпечити необхідну мобільність громадян в повному

обсязі і з високою якістю, тому вдосконалення його роботи має пріоритетне значення.

Для таких завдань зазвичай досить шукати наближені рішення, які знаходяться досить швидко і досить точні для необхідних цілей. Зазвичай це досягається різними евристичними методами.

Задача маршрутизації транспорту (Vehicle Routing Problems, VRP) – завдання комбінаторної оптимізації, в яких для парку транспортних засобів, розташованих в одному або декількох депо, повинен бути визначений набір маршрутів до декількох віддалених точок-споживачів. Інтерес до VRP викликаний її практичною значущістю при значній складності.

Для вирішення задачі оптимізації було запропоновано використання модифікованого жадібного алгоритму, головною особливістю якого було обмеження області допустимих розв'язків. Було розроблено програмне забезпечення, що використовує даний алгоритм, а також реалізований алгоритм Кларка-Райта для проведення аналізу та порівняння з розробленим алгоритмом.

Для того щоб порівняти швидкодію модифікованого жадібного алгоритму та алгоритму Кларка-Райта було розраховано швидкість роботи даних алгоритмів використовуючи програмні засоби. За допомогою отриманих даних було побудовано графіки залежності часу роботи алгоритмів від кількості точок, що потрібно відвідати.

Як можна побачити на графіку, що на рисунку 1, алгоритм Кларка-Райта значно довше опрацьовує вхідні дані ніж розроблений модифікований жадібний алгоритм.

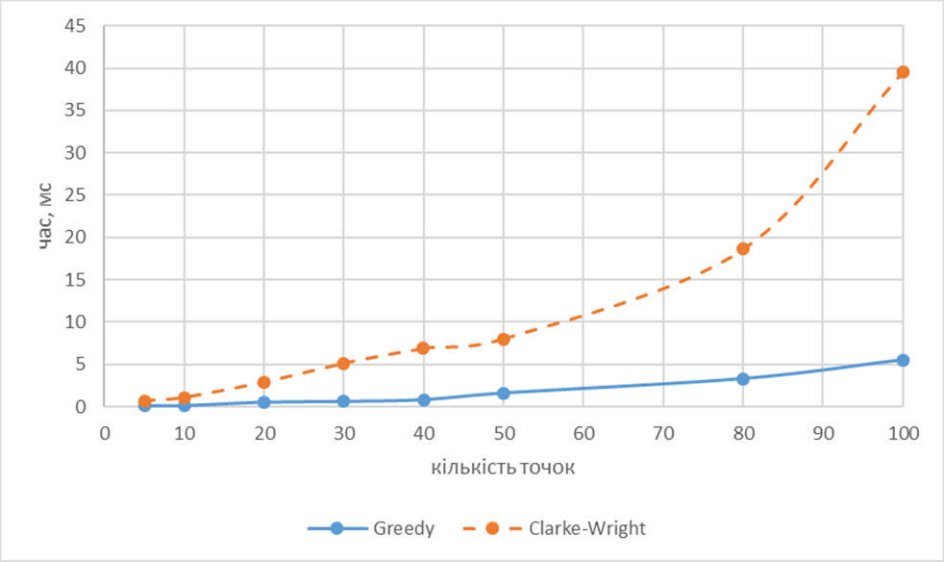


Рисунок 1 – Порівняння швидкодії

Час роботи розробленого методу на основі жадібного алгоритму має досить високу швидкість побудови маршрутів не лише через загальну простоту жадібних алгоритмів, але й завдяки внесеній модифікації у вигляді зменшення області допустимих розв'язків

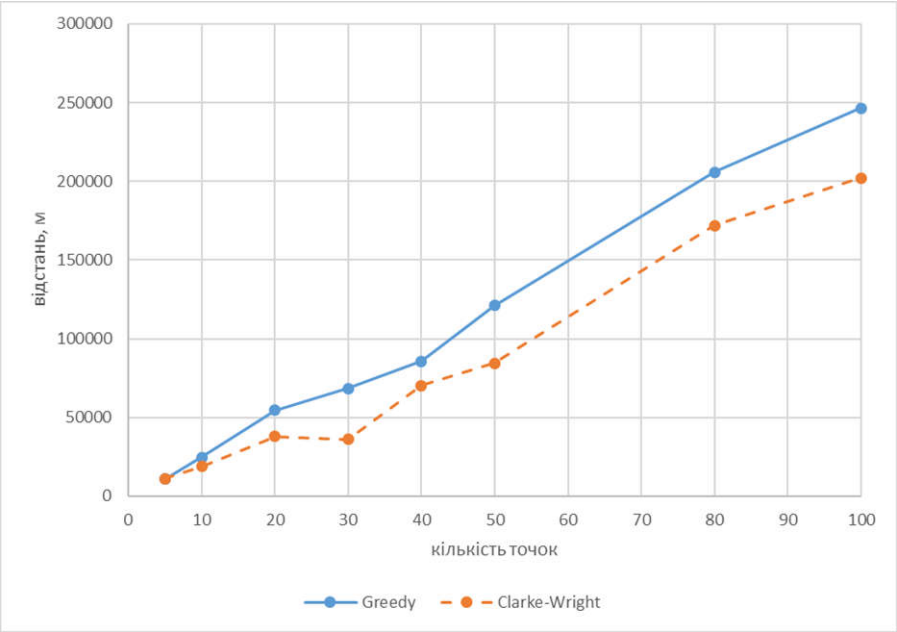


Рисунок 2 – Порівняння результатів роботи

На рисунку 2 зображене порівняння дистанції маршрутів, що були створені алгоритмами. Модифікований жадібний алгоритм не сильно поступається алгоритму Кларка-Райта в діапазоні до 40 адрес.

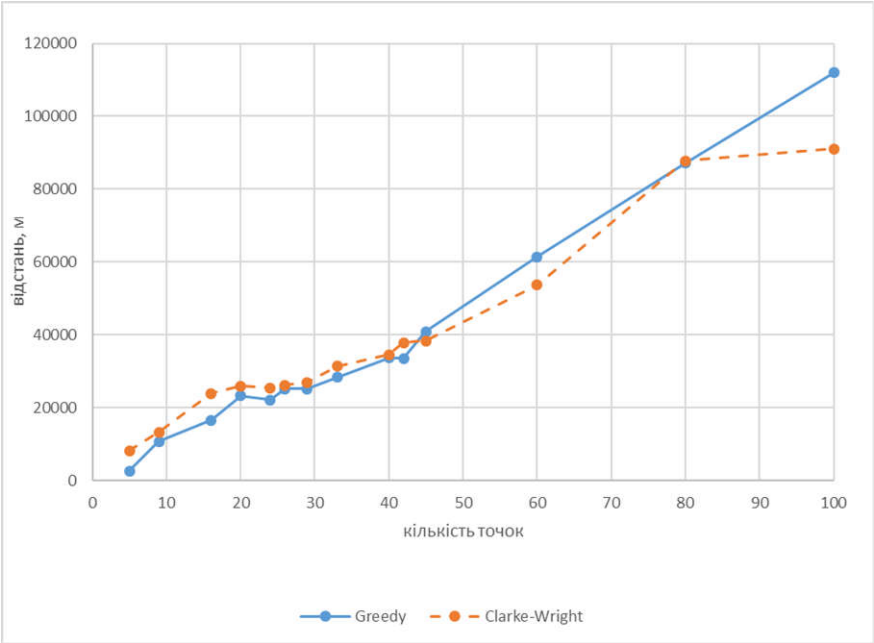


Рисунок 3 – Порівняння результатів без повернення в депо

На рисунку 3 зображено порівняння результатів роботи алгоритмів без повернення в депо. В даному випадку розроблений алгоритм має перевагу перед існуючим алгоритмом Кларка-Райта в діапазоні до 45 точок.

В підсумку можна зробити висновок, що розроблений алгоритм має значно кращу швидкодію в порівнянні з алгоритмом Кларка-Райта, а також не сильно поступається дистанцією маршрутів, а у випадку з не поверненням в депо навіть має незначну перевагу.

Література:
1. Golden, B. L. The vehicle routing problem: latest advances and new challenges / Golden, B. L., Raghavan, S., Wasil, E. A. – Springer Science & Business Media: 2008. – P. 589.
2. Solomon, M. M. Algorithms for the vehicle routing and scheduling problems with time window constraints / Taillard É. // Reprinted from Operations research Technical Journal. – 1987. – V.35 – P. 254-265.
3. Haimovich M., Bounds and heuristics for capacitated routing problems / M. Haimovich, A.H.G. Rinnooy Kan // Mathematics of Operations Research. — 1985. — № 10. — P. 527–542.

Бондаренко А.С., *студент*
Черкаський державний бізнес-коледж, м. Черкаси
Відділення інформаційних систем

OFFLINE FIRST АРХІТЕКТУРА ВЕБ-ДОДАТКУ

У сучасних реаліях досить просто уявити ситуацію коли користувач хоче перевірити недавній електронний лист, щоб згадати адресу, за якою повинен був піти, але поштовий клієнт відмовляється надати її, тому що немає доступу до Інтернету. Або після написання чудового нового посту для свого блогу в поїзді або літаку, бачити як додаток для блогів відмовляється його зберігати, і заряду батареї замало. [1]

Мобільні мережі просто не можуть забезпечити такий же рівень якості та стабільності, вже давно пора було почати адаптуватися до такої реальності. Це як розробляти додаток по стратегії Mobile First, але з урахуванням такого стану справ коли доступу до мережі може не бути – Offline First. Під автономним режимом мається на увазі не тільки спеціальний план дій на випадок виникнення помилок, а й фундаментально більш гнучкий і відмово стійкий шлях дизайну додатку. [2]

Вся робота в офлайн будується на використанні Service Worker. У нього є багато властивостей такі як контроль сторінок на яких він був викликаний, перехоплення запитів та їх модифікація, а також кешування ресурсів через спеціальний кеш API.

Також хочеться відзначити, що він працює в потоці, паралельному від основного, і відповідно, йому не доступний DOM, LocalStorage або об'єкт Window. Але зате він, виконуючи якісь дії, не блокує основний UI браузера.

Також варто відзначити, що робота з офлайн – це частина PWA (Progressive Web Applications). [3] PWA можна сприймати як сайт, побудований з використанням веб-технологій, але який взаємодіє з користувачем як нативний додаток. Досягнення в розробці браузерів, кеші і push-інтерфейси дозволили встановлювати додаток на домашній екран безпосередньо з браузера, отримувати спливаючі повідомлення і навіть працювати в автономному режимі. [4]

Отже, на сьогоднішній день уже існують всі необхідні технології які необхідні для створення максимально комфортних, доступних, легких та

швидких веб-додатків які можуть гарантувати стабільність роботи та блискавичну реакцію на дії користувача.

Але більшість розробників додатків, якими люди користуються кожного дня, вважають зайвим використовувати всі можливості які їм надає платформа у погоні за швидкістю розробки та тимчасовою вигодою у вигляді заробітної плати, забуваючи про зручність використання створених ними додатків іншими людьми.

Тому потрібно виконувати кожне завдання максимально якісно, ніби його результатами розробнику потрібно буде користуватися наступні декілька років при цьому використовуючи не самі сучасні пристрої. Але це в ідеалі, на практиці достатньо дотримуватися критеріїв якості які описані в стандарті ISO/IEC 25010:2011.

Література:

1. Huculak Ł. Offline-first application architecture [Електронний ресурс] / Łukasz Huculak – Режим доступу до ресурсу: https://codete.com/blog/offline-first-application-architecture/.
2. Lash D. Say Hello to Offline First [Електронний ресурс] / Dan Lash – Режим доступу до ресурсу: http://blog.hood.ie/2013/11/say-hello-to-offline-first/.
3. Бунин О. Как мы запустили offline-версию сайта RG.RU [Електронний ресурс] / Олег Бунин – Режим доступу до ресурсу: https://habr.com/ru/company/oleg-bunin/blog/348150/.
4. Демьяненко М. Что такое Progressive Web Apps и какие возможности они открывают для вашего бизнеса [Електронний ресурс] / Марина Демьяненко – Режим доступу до ресурсу: https://netpeak.net/ru/blog/chto-takoe-progressive-web-apps-i-kakie-vozmozhnosti-oni-otkryvayut-dlya-vashego-biznesa/.

Бондаренко Валерий Олегович, *аспирант*
Национальный университет «Запорожская политехника»
Кафедра программных средств
Субботин Сергей Александрович, *д-р техн. наук, профессор*
Национальный университет «Запорожская политехника»
Кафедра программных средств, заведующий кафедры
Олейник Андрей Александрович, *канд. техн. наук, доцент*
Национальный университет «Запорожская политехника»
Кафедра программных средств, доцент

ГЛУБОКИЕ СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ

Глубокое обучение недавно дало новую силу, которая позволяет создавать системы искусственного интеллекта (ИИ), которые были невозможны несколько лет назад. Сегодня ИИ - это взрывная технология, которая решает задачи, которые требуют огромного количества вычислительной мощности, выполняемой компьютерами [1]. С другой стороны, существуют проблемы, которые могут быть решены людьми, такие как распознавание изображений, распознавание слов или рассмотрение направления и препятствий на дорогах на

основе их интуиции и опыта. Это означает, что с массивно-параллельными системами обработки и внедрением сложных алгоритмов эти решения могут быть найдены намного быстрее и с более высокой точностью.

Есть два фундаментальных подхода в области ИИ. Первый подход основан на системах инженерии знаний, логическом программировании и логических рассуждениях. Второй подход охватывает микроскопические биологические модели [2]. Искусственные нейронные сети (ANN) и генетические алгоритмы являются основными примерами этого более позднего подхода. Наиболее заметными особенностями ANN являются: обширная сетка взаимосвязей простых блоков обработки, настройка параметров или весов сетки для выполнения задач или адаптации к окружающей среде в процессе обучения.

Глубокие сверточные нейронные сети (CNN) представляют собой специализированный вид ANN, которые используют свертку вместо общего умножения матриц, по крайней мере, на одном из их слоев. В отличие от простых нейронных сетей, CNN состоят из множества слоев. Такая особенность позволяет им компактно представлять сильно нелинейные и изменяющиеся функции. CNN включают в себя множество соединений, и архитектура, как правило, состоит из различных типов уровней, включая свертку, пул и полностью связанные уровни, и реализует форму регуляризации. Для изучения сложных функций CNN потребуется глубокая архитектура, которая состоят из большого числа нейронов и множества уровней скрытых вычислений нелинейности.

Изучение теоретических и некоторых практических применений CNN показало, что основным преимуществом использования CNN по отношению к стандартным нейронным сетям является уменьшенное количество параметров, приводит к меньшему шуму во время тренировочного процесса. Причина в том, что количество параметров зависит от ширины ядра. Чем шире ядро, тем больше количество параметров в модели. Также, CNN обычно нуждаются в тысячах или даже миллионах помеченных данных. Параметры модели становятся больше, если параметры затухания веса уменьшаются. Частота отсева должна быть уменьшена, чтобы избежать увеличения количества итераций. Скорость обучения должна быть оптимально настроена. Очень высокий или очень низкий показатель приведет к проблемам с оптимизацией и уменьшению эффективной емкости сети.

Литература:

1. Ivars Namatēvs Deep Convolutional Neural Networks: Structure, Feature Extraction and Training [Electronic resource] / Ivars Namatēvs –Access mode: https://www.researchgate.net/publication/322416654

2. Nash R. O'Shea K. An Introduction to Convolutional Neural Networks [Electronic resource] / R. Nash –Access mode: https://www.researchgate.net/publication/285164623

Ворона М.В.
Национальный университет кораблестроения имени адмирала Макарова, г. Николаев
Кафедра программного обеспечения автоматизированных систем, аспирант
Фарионова Т.А., *канд.техн.наук, доцент*
Национальный университет кораблестроения имени адмирала Макарова, г. Николаев
Кафедра программного обеспечения автоматизированных систем, доцент

МОДЕЛЬ РАСПОЗНАВАНИЯ ИМЕНОВАННЫХ ОБЪЕКТОВ ДЛЯ АНАЛИЗА И ИЗВЛЕЧЕНИЯ ИНФОРМАЦИИ ИЗ НЕСТРУКТУРИРОВАННЫХ РЕЗЮМЕ И ТЕКСТОВ ВАКАНСИЙ

Введение. По оценкам специалистов почти 80% мировых цифровых данных представлены в неструктурированной текстовой форме. Управление персоналом – одна из сфер человеческой деятельности, которая нуждается в разработке и использовании эффективных механизмов технологии анализа неструктурированной информации [1].

Текстовые данные о вакансиях и резюме представлены в различных формах, включая внутренние базы данных компаний, социальные сети, сайты о вакансиях и электронные доски объявлений о вакансиях в Интернете, которые содержат большой объем ценных, нераскрытых знаний и скрытых связей [2].

Не существует классических подходов к статистическому и математическому моделированию, которые могли бы дать хорошие результаты в извлечении чистой структурированной и конкретной информации из неструктурированного текста. Есть только один способ, который можно отнести к классической категории – использование специфических нейросетей, но для этого требуется огромное количество подготовленных наборов данных (текст, размеченный вручную) и значительные вычислительные мощности [3].

Альтернативный и менее затратный путь – это разработка модели, основанная на распознавании именованных объектов (Named Entity Recognition, NER) [3] для анализа и извлечения информации из неструктурированных резюме и текстов вакансий, что требует значительно меньший объём подготовленных маркированных данных.

Целью исследования является построение модели, которая обеспечит высокое качество извлечения форматированных данных из неформатированных текстов резюме и вакансий на основе небольшого набора размеченных объектов.

Модель распознавания именованных объектов для анализа и извлечения информации из неструктурированных текстов. Авторами предложена гибридная модель NER, которая включает:

- алгоритмы, основанные на правилах. В этом случае подход предполагает использование заранее подготовленных словарей, в виде структурированных наборов данных, чаще всего таблиц, которые описывают все возможные варианты использования той или иной лексической единицы в тексте;

- алгоритмы, основанные на статистике (с применением машинного обучения). Это предполагает статистический анализ текстов резюме и вакансий, где на основе вероятности появления той или иной лексической единицы в том или ином контексте вырабатываются рекомендации, куда «определить» ее в конкретном случае.

На рисунке 1 представлена структура гибридной модели NER. В качестве исходных данных для модели выступает набор данных в виде неформального текста. Также используется словарь, собранный из различных онлайн-источников. Модель поддерживает обнаружение как однословных, так и многословных NEs (именованных единиц).

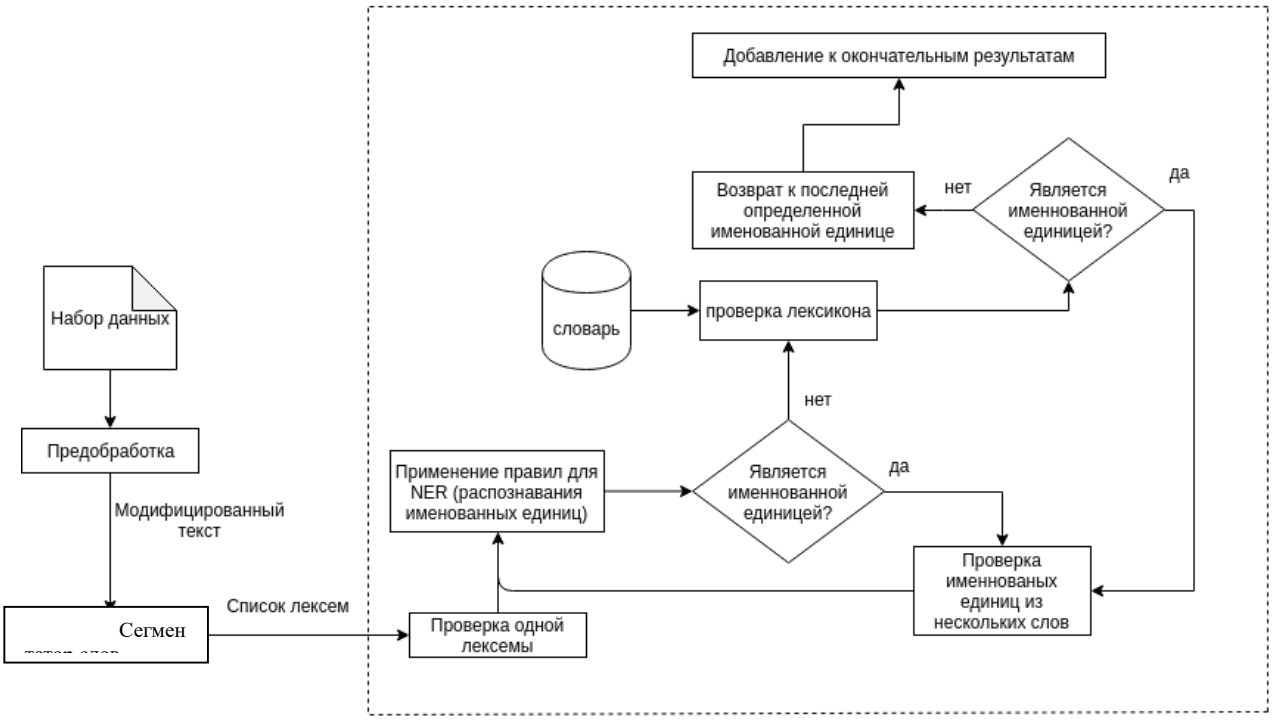


Рисунок 1 - Структура гибридной модели NER

Модель NER включает такие этапы: сбор словарей, предварительную обработку данных, проверку в одном или нескольких словах NE и извлечение правил NE.

Процесс классификации состоит из двух составляющих (рис.2). Первым шагом является "обучение работе с данными", на котором в качестве входных приводятся аннотированные данные. Затем, по результатам тренировок на основе анализа контурных векторов строится необходимая модель. Эта модель

используется для аннотации входных данных на втором этапе, где выполняется категоризация выборки входных данных и присвоения ей ярлыка в соответствии с моделью, построенной на первом этапе. Когда дается выборка входных данных, из нее извлекаются репрезентативные характеристики для формирования функционального вектора. Затем функциональный вектор переходит в модель ML (Machine Learning) для получения аннотации. Аннотация – это метка, присваиваемая образцу входных данных, предполагающая его прогнозируемый класс.

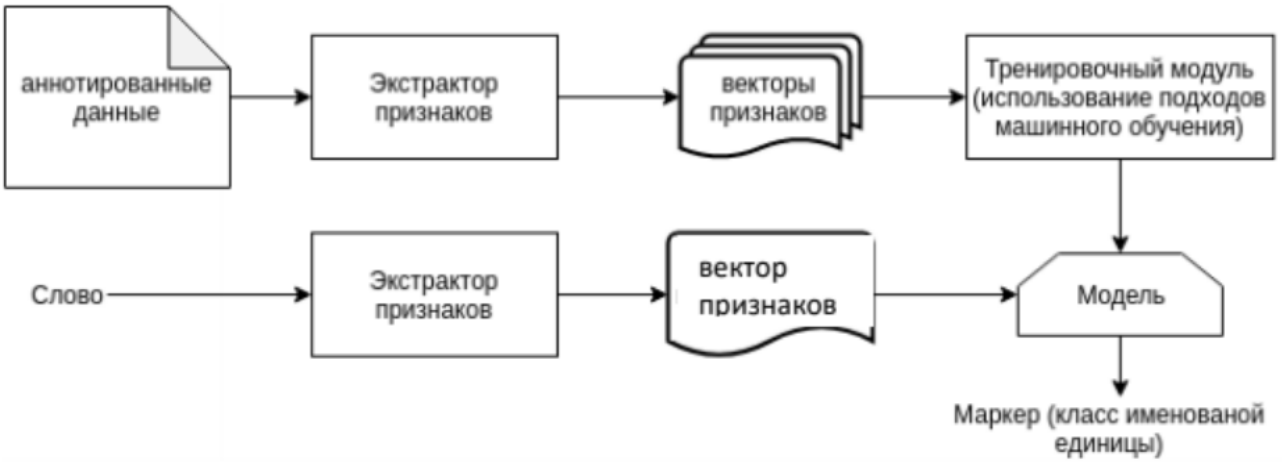


Рисунок 2 – Схема процесса классификации лексем

Выводы. В ходе апробации гибридной модели распознавания именованных объектов для анализа, извлечения информации из неструктурированных резюме и текстов вакансий было отмечено, что для достижения лучших результатов существенно необходимо:

- регулярно обновлять собранную лексику, а для каждого нового объекта определять свой собственный словарь.

- разработать больше правил, связанных с доменом, которые будут изменяться в зависимости от каждой именованной единицы.

Литература:

1. Романов Д.А. Зачем нам нужны технологии поиска и анализа неструктурированной информации? Как оценить экономический эффект? URL: https://publications.hse.ru/mirror/pubs/share/folder/b1ju87tuqn/direct/146157934 (дата обращения: 03.06.2019)

2. Text mining: алгоритмы извлечения семантики из текста URL: http://datareview.info/article/text-mining-algoritmyi-izvlecheniya-semantiki-iz-teksta (дата обращения: 03.06.2019)

3. S. Keretna, Chee Peng Lim and D. Creighton, "A Hybrid Model for Named Entity Recognition Using Unstructured Medical Text", in Proceedings of the IEEE 9th International System of Systems Engineering Conference, Adelaide, Australia, 9-13 June 2014.

Гаркуша Ю. А.
Університет митної справи та фінансів, м. Дніпро
студент
Рудянова Т. М., *к.ф.-м.н., доцент*
Університет митної справи та фінансів, м. Дніпро
кафедра прикладної математики та інформатики, доцент

ПОРІВНЯЛЬНИЙ АНАЛІЗ МОДЕЛЕЙ МАТЕМАТИЧНОГО ПРОГРАМУВАННЯ ДЛЯ ВИЗНАЧЕННЯ ВИДУ ЕФЕКТУ МАСШТАБУ

Особливе значення при розробці виробничої програми підприємства і її реалізації має оптимізація структури продукції, що випускається на основі аналізу купівельного попиту на конкретні види товарів і послуг і врахування обмежень за виробничими потужностями, матеріальними ресурсами і потребами на ринку.

В процесі виробництва завжди виникає питання, чи є план та обсяг виробництва, який використовується на даний момент, оптимальним? Відповісти на це питання можна за допомогою математичних методів.

Віддача за обсягом – це концепція, яка ілюструє зміни в обсязі виробництва при змінності обсягів залучення всіх ресурсів. Ефект масштабу відіграє важливу роль в ефективному функціонуванні певної галузі виробництва, тому доцільно дослідити його наявність та вид: додатній, від'ємний чи незмінний.

Незмінний (постійний) ефект масштабу виробництва пояснюється однорідністю змінних факторів і має місце, коли обсяг випуску продукту збільшується пропорційно росту витрат ресурсів. У цьому разі середня і гранична продуктивність факторів виробництва залишається незмінною, а сукупний галузевий випуск продукту буде однаковим незалежно від того, працюватиме одне велике підприємство чи замість нього буде створено кілька дрібних.

Додатній (зростаючий) ефект масштабу виробництва має місце, коли обсяг випуску продукції фірмою збільшується у пропорції, яка перевищує пропорцію зростання витрат ресурсів. Це характерно для тих виробництв, де можлива широка автоматизація виробничих процесів, застосування поточних і конвеєрних ліній, оптимізація розподілу праці, комплексна переробка базової сировини тощо.

Від'ємний (спадний) ефект масштабу виробництва виявляється, коли зростання обсягу випуску продукції відбувається у меншій пропорції, ніж збільшення витрат виробничих ресурсів. При від’ємному ефекті масштабу виробництва не вигідно збільшувати розміри підприємства. Причиною низької ефективності у цьому разі, як правило, є додаткові витрати, пов’язані з управлінням подібним підприємством, забезпеченням дотримання норм трудової і технологічної дисципліни, координації взаємодії підрозділів підприємства тощо.

Метою дослідження є виявлення наявності ефекту зміни втрат заліза від масштабу виробництва, залежності втрат заліза від обсягу виробництва, при зміні норм відповідних ресурсів. На основі статистичних даних металургійного заводу ПАТ "Дніпровський металургійний завод" про витрату матеріалів на виробництво чавуну, наведених в таблиці 1, складемо і порівняємо дві економіко-математичні моделі виробництва чавуну. Перша модель базується на первинних нормах споживання ресурсів і покаже оптимальний теоретичний обсяг виробництва, який дозволить мінімізувати відсоток втрат заліза, друга модель базується на обмеженнях, які вже враховують постійні відхилення та описує інший, коректований, обсяг виробництва.

Залежно від змісту домішок в чавуні, виділяють дві марки продукту :

1) Si 0,7%, Mn 0,27%, S 0,022% – чавун для виробництва напівспокійних марок сталі, повинен складати не менше 72,5% від виробництва загального об'єму. При цьому втрати заліза в шлакові відходи при виробництві цієї марки складають 6,5%.

2) Si до 1%, Mn 0,5%, S 0,07% – чавун для виробництва рейкової сталі, втрати заліза в шлакові відходи при виробництві цієї марки складають 8,2%.

Таблиця 1.

Витрати основних матеріалів на виробництво партії чавуну

Матеріали	Базова норма споживання, т	Альтернативна норма, т	Відхилення
Залізна руда	708	1079,7	371,7
Руда "Дружба"	4425	3770,1	-654,9
Кокс сухий	34898,5	34479,6	-418,9
Зварювальний шлак	944	831,9	-112,1
Залізофлюс	2419	0	-2419

Вважається недоцільним робити партії чавуну, які витрачають об'єм ресурсів менший за норму.

Основні матеріали для виробництва кожної марки чавуну наведені в таблиці

Таблиця 2.

Використані матеріали для виробництва чавуну

Матеріали	Марка 1 кг/т	Марка 2 кг/т
Залізна руда	12	18,3
Руда "Дружба"	75	63,9
Кокс сухий	591,5	584,4
Зварювальний шлак	16	14,1
Залізофлюс	41	0

Побудуємо математичні моделі задач лінійного програмування. Нехай x_1 – кількість тон чавуну першої марки, x_2 – кількість тон чавуну, другої марки, який планує виробляти підприємство. Цільова функція – втрати заліза при виробництві, які необхідно мінімізувати. Побудовані на підставі вихідних даних моделі мають вигляд:

Модель 1

$F = 0{,}065\,x_1 + 0{,}082x_2 \rightarrow \min$

$$\begin{cases} 0{,}012x_1 + 0{,}0183x_2 \geq 708 \\ 0{,}075x_1 + 0{,}0639x_2 \geq 4425 \\ 0{,}5915x_1 + 0{,}5844x_2 \geq 3489 \\ 0{,}016x_1 + 0{,}0141x_2 \geq 944 \\ 0{,}041x_1 \geq 2419\,; \\ 0{,}275x_1 \geq 0{,}725x_2\,; \\ x_1 \geq 0;\ x_2 \geq 0 \end{cases}$$

Модель 2

$Z = 0{,}065\,x_1 + 0{,}082x_2 \rightarrow \min$

$$\begin{cases} 0{,}012x_1 + 0{,}0183x_2 \geq 1079, \\ 0{,}075x_1 + 0{,}0639x_2 \geq 3770{,}1 \\ 0{,}5915x_1 + 0{,}5844x_2 \geq 3447{,}9 \\ 0{,}016x_1 + 0{,}0141x_2 \geq 831{,}9 \\ 0{,}041x_1 \geq 0\,; \\ 0{,}275x_1 \geq 0{,}725x_2\,; \\ x_1 \geq 0;\ x_2 \geq 0 \end{cases}$$

Вирішивши задачі лінійного програмування симплекс-методом отримуємо значення змінних і значення цільових функцій, що відповідають їм, для обох моделей:

модель 1: $x_1 = 59000$ т чавуну першої марки; $x_2 = 22379{,}310$ т чавуну другої марки; $F^* = 5670{,}103$ т заліза втраченого в шлакові відходи;

модель 2: $x_1 = 57002{,}18$ т чавуну першої марки; $x_2 = 21621{,}518$ т чавуну другої марки; $Z^* = 5478{,}106$ т заліза втраченого в шлакові відходи.

Порівнюючи показники моделей, можна зробити висновок, що відсоток втраченого в партії заліза не залежить від обсягу виробництва і обмежень норм витрат, оскільки:

$$\frac{\text{Обсяг втраченого заліза Моделі 1}}{\text{Загальний обсяг виготовленого чавуну Моделі 1}} = \frac{\text{Обсяг втраченого заліза Моделі 2}}{\text{Загальний обсяг виготовленого чавуну Моделі 2}};$$

$$\frac{5670{,}103}{81379{,}310} = \frac{5478{,}106}{78623{,}702} = 0{,}0696 = 6{,}96\%\ .$$

Таким чином, на основі проведеного дослідження, можна стверджувати, що має місце незмінний ефект масштабу виробництва, при якому обсяг браку продукту збільшується пропорційно зміні витрат ресурсів та дорівнює 6,96 % від загального обсягу. На підприємствах виробничої сфери потрібно постійно проводити аналіз структури виробництва, щоб розуміти, чи є теперішній обсяг оптимальним, це дозволяє дізнатися, коли потрібно збільшити або зменшити виробничі потужності для оптимізації структури продукції.

Список використаних джерел:

1. Моделювання комерційно-господарської діяльності підприємства: Навчальний посібник / Шерстенников Ю. В., Яковенко О. Г., Іванов Р. В., Рудянова Т. М. – Дніпро:

Дніпропетровський національний університет імені Олеся Гончара: Видавництво ПФ «Стандарт-Сервіс», 2018. – 130 с.
2. Мікроекономіка: навч. посібник. Рекомендовано Міністерством освіти і науки України / А.Є. Ачкасов, І.А. Островський, С.Б. Тимофієва; – Харк. нац. ун–т. міськ. госп-ва імені О.М. Бекетова. – Харків: ХНУМГ імені О. М. Бекетова, 2012. – 248 с.
3. Продукція та активи Дніпровського металургійного заводу. – Режим доступу: http://dmz-petrovka.dp.ua/index.php?page=prod .

Григорович В.В.
Національний технічний університет України
"Київський Політехнічний Інститут ім. Ігоря Сікорського", м. Київ
Кафедра автоматизованих систем обробки інформації та управління,
студент

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ЗАДАЧ ДИНАМІКИ ЛІНІЙНИХ СИСТЕМ З НЕПОВНО ВИЗНАЧЕННИМИ ПОЧАТКОВИМ ТА КРАЄВИМ СТАНОМ

Вступна частина

Останнім часом все частіше з'являються технологічні процесів, які формалізовані математично, але мають високий рівень небезпеки спостережень за ними для життя людини, що призводить до неповноти та некоректності математичної постановки задач досліджень цих процесів. В силу цього задачі дослідження таких процесів не можуть бути розв'язані строго математично або з використанням класичних чисельних методів[1]. Тому актуальними є підходи до математичного моделювання динаміки таких процесів на базі мінімально доступних спостережень за ними.

Підхід до побудови функції Гріна.

Започатковані та узагальнені методи дослідження неповно визначених за початково-крайовим станом розподілених просторово-часових систем дозволяють побудувати функцію стану та керуючі зовнішньо-динамічні збурення, які задовольняючи диференціальні моделі системи, за середньоквадратичним критерієм узгоджуються з початково-крайовим та бажаним (для задач керування) станами[2]. Розв'язки сформульованих там прямих та обернених задач побудовані без особливих обмежень на кількість та якість (дискретно, чи неперервно-визначені) початково-крайових спостережень. Обов'язковим однак при цьому є наявність функції Гріна досліджуваною процесу, побудованої для необмеженої просторово-часової області. Ґрунтуючись на теоретичних основах побудови та методах теорії функцій комплексної змінної нижче пропонується простий для реалізації алгоритм побудови функції Гріна для просторово зосередженої динамічної системи, визначеної на необмеженому часовому інтервалі. З її використанням розв'язуються задачі моделювання функції стану та керування ним для динамічних систем з неповно визначеним початково-кінцевим станом та дискретно (або неперервно) визначеним бажаним станом. Керуючи ми в

останньому випадку вважаються зовнішньо-динамічні та початково визначені збурення[3][4]

Розглянемо просторово зосереджену систему, динаміка якої при $t > 0$ (t – часовий параметр) визначається співвідношенням

$$L_n(\partial_t)y(t) = u(t), \quad (1)$$

$$L_r^0(\partial_t)\,y(t)\big|_{t=0} = Y_r^0\ (r = \overline{1, R_0}), \quad (2)$$

$$L_r^0(\partial_t)\,y(t)\big|_{t=\Gamma} = Y_r^0\ (r = \overline{1+R_0, R_0+R_\Gamma}) \quad (3)$$

в яких $y(t)$ – функція стану системи, $u(t)$ – зовнішньо динамічний збурюючий фактор, $L_n(\partial_t)$, $L_r^0(\partial_t)$ – лінійні диференціальні оператори порядку n та $n_r < n$ відповідно (∂_t – похідна по часу), а $Y_r^0\ (r = \overline{1, R})$ – задані значення. Зупинимося на особливостях розв'язання задачі (1), (2), (3) для випадку, коли $R < n$, а чисельно аналітичні підходи до побудови функції $y(t)$ не дають бажаного результату[5].

Будемо виходити з підходу, який дозволяє побудувати функцію $y(t)$ таку, щоб рівняння (1) виконувалося точно, а

$$\Phi = \sum_{r=1}^{R} (L_r^0(\partial_t)y(t)\big|_{t=h_r} - Y_r^0) \to \min_{y(t)}, \quad (4)$$

де $h_r = \begin{cases} 0, & r \in \overline{1, R_0} \\ \Gamma, & r \in \overline{R_0+1, R} \end{cases}$ коли $R = R_0 + R_\Gamma$

Розв'язок задачі (1)-(4) подамо співвідношенням

$$y(t) = y_\infty(t) + y_0(t)\ (t > 0), \quad (5)$$

в якому

$$y_\infty(t) = \int_0^{+\infty} G(t-t')u(t')dt', \quad (6)$$

$$y_0(t) = \int_{-\infty}^{0} G(t-t')u_0(t')dt', \quad (7)$$

$$G(t-t') = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \frac{1}{L_n(i\lambda)} e^{i\lambda(t-t')} d\lambda \quad (8)$$

(тут та надалі i – уявна одиниця), а $u_0(t)$ – функція, моделююча початкові та крайові умови (2), (3) так щоб

$$\Phi \to \min_{u_0(t)}. \quad (9)$$

Перед розв'язанням задачі (9) зупинимося на особливостях знаходження функції $G(t-t')$, для того співвідношення (8) запишемо у вигляді

$$G(t-t') = \frac{1}{2\pi i}\int_{-i\infty}^{+i\infty} \frac{1}{L_n(p)} e^{p(t-t')} dp. \quad (10)$$

Враховуючи, що $L_n(p)$ – є поліном n-го порядку, робимо висновок, що особливості підінтегральної функції визначаються його коренями $p_k^*\ (k = \overline{1, n})$.

За цих умов

$$G(t-t') = \sum_{k=1}^{n} \mathrm{Res}[\frac{1}{L_n(p)} e^{p(t-t')}, p_k], \qquad (11)$$

де $\mathrm{Res}[...]$ – інтегральний лишок, який визначається співвідношенням

$$\mathrm{Res}[...] = \frac{e^{p_k(t-t')}}{\partial_p L_n(p)\big|_{p=p_k}}, \qquad (12)$$

якщо p_k – простий полюс, або

$$\mathrm{Res}[...] = \frac{1}{(m_k-1)!} \lim_{p \to p_k} \left\{ \frac{d^{m_k-1}}{dp^{m_k-1}} \left[(p-p_k)^{m_k} \frac{e^{p(t-t')}}{L_n(p)} \right] \right\}, \qquad (13)$$

якщо p_k полюс кратності m_k.

Не важко бачити, що знаходження функції Гріна є важкою задачею. Для її вирішення варто скористатися таблицями з раніше знайденими явними представленнями функції Гріна.

Література:

1. Стоян В.А. Об одном подходе к исследованию начально-краевых задач матфизики // Проблемы управления и информатики.–1998– №1.С.79-86.
2. Скопецький В.В.,Стоян В.А., Кривонос Ю.Г. Математичне моделювання прямих та обернених задач динаміки систем з розподіленими параметрами. – К.:Наук. Думка, 2002. – 361 с.
3. Стоян В.А. До побудови функції Гріна для систем з розподіленими параметрами //Вычислительная и прикладная математика. – 1997. –Вып. 81. – С.108–111.
4. Скопецький В.В.,Стоян В.А., Зваридчук Б.В. Математичне моделювання динаміки розподілених просторово-часових процесів. - К.: Сталь, 2008. –316 с.
5. Стоян В.А. Моделювання та ідентифікація динаміки систем з розподіленими параметрами. – К.: ВПЦ “Київський університет”, 2004. – 184 с.

Данилюк В. М., *студент*
Івано-Франківський національний технічний університет нафти і газу, м. Івано-Франківськ
Корнута В. А., *доцент*
Івано-Франківський національний технічний університет нафти і газу, м. Івано-Франківськ
Кафедра інженерної та комп'ютерної графіки

ДОКУМЕНТНИЙ СУПРОВІД ЗАПРОВАДЖЕННЯ ПРОЕКТНО-ОРІЄНТОВАНОГО НАВЧАННЯ В УНІВЕРСИТЕТІ

Активізація євроінтеграційних процесів в Україні, охоплюючи практично всі сфери суспільного життя, включає й систему української вищої освіти, реформування концептуальних, структурних та організаційних засад які є серйозним викликом для українського суспільства. Слід зазначити, що повноцінне реформування освітньої системи має включати як активну викладацьку діяльність у сфері впровадження інноваційних підходів до організації навчального процесу, так і адміністративно-управлінські заходи.

В остатні роки світовий досвід підтверджує, що впровадження підходів спрямованих на розвиток у студентів здатності до самоспрямованого навчання, підготовка їх до роботи в сучасному світі є неможливою без впровадження практично-орієнтованих підходів [1]. Важливим аспектом даного підходу є переважання консультативної ролі викладача по відношенню до студентів, що працюють над певним проектом чи проблемою [4]. Таким чином, відбувається перехід від навчання, орієнтованого на викладача до навчання орієнтованого на студента (такого, що спрямовується студентом самостійно, виходячи із його освітніх потреб).

Сутність проектно-орієнтованого підходу до формування професійних компетентностей студентів полягає в тому, що в рамках єдиного навчально-науково-виховного процесу розробляються і реалізуються два типи інноваційних проектів: науково-виробничі та науково-освітні проекти. Практично проектно-орієнтований підхід дозволяє реалізувати «бінарні проекти», коли впровадження нових освітніх технологій, побудованих на розробці студентами науково-виробничих проектів, дозволяє забезпечити нову якість підготовки фахівців.

Організаційно-методична база впровадження проектно-орієнтованого підходу може бути забезпечена на основі чотирьох головних напрямів модернізації освітнього процесу [2]:

- доповнення класичної структури університету матричними компонентами;
- інтенсифікації процесу освоєння навчальних модулів за рахунок скорочення тривалості навчального циклу;
- розширення реальних можливостей вільного вибору студентами традиційних чи нових освітніх технологій в рамках передпроектної підготовки на молодших курсах;
- створення системи академічної селекції студентів та налагодження механізму виявлення потенціалу здатності до навчання та креативності при комплектуванні навчально-проектних груп.

Проектно-орієнтоване навчання студентів має проводитися на основі таких організаційно-розпорядчих документів, як:

- статут, який визначатиме основні правила, права та обов'язки у сфері цього типу навчання;
- індивідуальний навчальний план проектної підготовки, скоординованих з навчальними планами підготовки бакалаврів та магістрів різних спеціальностей;
- посадова інструкція, яка міститиме основні або типові завдання та обов'язки працівників;
- розроблені методичні вказівки. Прикладом можуть слугувати:

1. Markham T 2012 Project Based Learning (San Rafael, California: HeartIQ Press) 152.

2. Telizhenko O M, Luk’yanykhin V O and Baystryuchenko N O 2016 Project-oriented approach to the educational process Visnyk NTU “KhPI” 1 30-35 http://repository.kpi.kharkov.ua/handle/KhPI-Press/21627

Проектно-орієнтована діяльність нині є невід'ємною складовою підготовки студентів у багатьох країнах світу. Застосування проектно-орієнтованого навчання дозволяє хоча б частково долати критичний розрив між навчанням і реальною професійною діяльністю. Ефективність такого підходу підтверджується зростанням інтересу і мотивації студентів, а також дозволяє розвивати навички командної роботи, комунікативні навички та здатність самостійно формулювати задачі та шукати способи їх розв'язання.

Література:

1. Бушуєв С. Д. Ціннісний підхід у діяльності проектно-керуючих організацій. *Науковий вісник міжнародного гуманітарного університету.* 2010. Вип. 1. С. 12–20.
2. Павленко, О. О., Глівенко С. В. Використання проектного підходу при впровадженні навчальних інновацій. *Управління проектами та розвиток виробництва*. 2010. № 1. URL: http://www.pmdp.org.ua/images/Journal/-33/10poovni.pdf.
3. Луценко Г. В., Луценко Гр. В. Проектно орієнтоване навчання: точка зору українських викладачів Stem-дисциплін. *Science and Education a New Dimension. Pedagogy and Psychology.* 2018. № 6. URL: https://doi.org/10.31174/-SEND-PP2018-155VI65-08
4. Пєхота О. М., Піктенко А. З. Освітні технології: навч.-метод. посіб. / за ред. О. М. Пєхоти. К. : А.С.К., 2004. 256 с.

Дончик Т.О.
Національний університет кораблебудування імені адмірала Макарова, м. Миколаїв
Кафедра програмного забезпечення автоматизованих систем, аспірант

МЕХАНІЗМИ ОЦІНЮВАННЯ ТРУДОМІСТКОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Сучасні програмні продукти є складними технічними системами. Вони характеризуються наявністю великої множини компонентів, паралельного виконання, розподілених ресурсів та інших важливих аспектів складності [1], які значно ускладнюють оцінювання тривалості проектів розробки програмного забезпечення. Слід зазначити, що близько 70% проектів з розробки програмного забезпечення не виконуються в заданий термін або не вкладаються в запланований бюджет. Це свідчить про існування проблеми низької достовірності оцінювання тривалості та трудомісткості таких проектів, що призводить до фінансових втрат. Тому актуальними є питання щодо побудови та вдосконалення моделей тривалості та трудомісткості проектів з розробки програмного забезпечення (ПЗ).

Механізми оцінювання трудомісткості ПЗ використовуються для вирішення багатьох завдань, а саме розробки бюджету проекту, аналізу ступеня ризику та вибору компромісного рішення, тощо. При відсутності адекватної і достовірної оцінки вартості програмного продукту неможливо забезпечити чітке планування й управління відповідним проектом. [2]

Існуючі підходи оцінювання трудомісткості розробки програмного забезпечення (ПЗ) дуже різноманітні. Розглянемо найбільш розповсюджені з них.

Алгоритмічне моделювання передбачає аналіз статистичних даних про раніше виконані проекти, при цьому визначається залежність трудомісткості проекту від одного або декількох кількісних показників програмного продукту (зазвичай це розмір програмного коду). Такі моделі постійно удосконалюються з метою підвищення їх точності оцінювання. Найчастіше реалізованими та добре документованими є моделі Путнем (ступенева, аналітична), модель COCOMO (ступенева, емпірична) та її модифікації. [3]

– Модель Путнем є найбільш поширеною моделлю аналітичної групи. Вона створена для програмних проектів об'ємом більше 70 000 рядків коду та ґрунтується на твердженні, що витрати на розробку програмного забезпечення розподіляються згідно кривих Нордена-Рейлі, що представляють розподіл робочої сили у часі.

– SLIM (Software Life-cycle Model) є альтернативною реалізацію моделі Путнем. Обмеженням цієї моделі є те, що вона може бути використана, якщо передбачувані витрати на створення програмного забезпечення більше 20 людино-місяців.

– Найпопулярнішою серед алгоритмічних моделей є сімейство моделей COCOMO (Constructive Cost Model) та її модифікація COCOMO II, що використовують просту формулу регресії з параметрами, визначеними з даних, зібраних по ряду проектів. COCOMO являє собою три моделі у відповідності до фаз життєвого циклу програмного забезпечення: базова (Basic) застосовується на етапі напрацювання специфікацій, розширена (Intermediate) – після визначення конкретних вимог до програмного забезпечення, поглиблена (Advanced) застосовується після закінчення проектування програмного забезпечення. При цьому витрати праці на проект визначають у людино-місяцях і залежать від розміру коду програми та в складності. Основними відмінностями COCOMO II від COCOMO є використання для оцінювання складності вхідних даних у вигляді функціональних точок, оцінювання елементів повторного використання та інтеграції програмних продуктів, об’єктно-орієнтовані підходи до оцінки компонентів програмного забезпечення та ін. [2]

Експертне оцінювання в першу чергу застосовуються для проектів, що вирішують інноваційні завдання. На основі оцінок окремих експертів у відповідності до існуючих методик формується інтегрована оцінка. До загальних недоліків методів експертного оцінювання відносять моменти, пов’язані з впливом «людського» фактору. Відсутність у явному виді аналітичного обґрунтування виявлених проблем, складність у перевірці компетенції тієї або іншої особи через неформалізованість критеріїв об’єктивності представлених даних теж є актуальними.

Оцінка за аналогією. В основу цього підходу покладено порівнянні плануємого проекту з попередніми проектами, що мають подібні характеристики, використовуючи експертні дані або збережені дані про проект.

Слабкість моделі полягає в тому, що ступінь подібності нового проекту і попередніх, як правило, не дуже велика.

Закон Паркінсона. Критерієм для оцінювання витрат на проект є людські ресурси, а не цільове оцінювання самого програмного продукту.

Оцінка з метою виграти контракт або тендер. Витрати на проект визначаються наявністю тих коштів, які є у замовника, тому трудомісткість проекту залежить від бюджету замовника, а не від функціональних характеристик розробляємого продукту.

Описані механізми оцінювання розробки ПЗ застосовні, якщо сформульовано вимоги, які дають можливість визначити функціональні характеристики розроблюваної системи. Однак у багатьох проектах оцінка витрат проводиться тільки на основі попередніх вимог до програмної системи, що дає мінімум інформації для оцінювання вартості проекту, не завжди добре враховують реальний розподіл емпіричних даних, що знижує достовірність оцінювання тривалості цих проектів. Закон розподілу емпіричних даних тривалості проектів з розробки програмного забезпечення та трудомісткості цих проектів відрізняється від нормального закону розподілу. [4] Тому, як правило, вказані моделі не дозволяють впевнено виконувати оцінювання довірчих інтервалів та інтервалів прогнозування тривалості без застосування припущення про нормальність закону розподілу емпіричних даних [5].

Для підвищення достовірності оцінювання тривалості та трудомісткості проектів з розробки програмного забезпечення необхідно побудувати по перше ймовірнісну модель, яка буде більш адекватно апроксимувати розподіл емпіричних даних тривалості робіт в цих проектах; по друге, нелінійну регресійну модель тривалості проектів з розробки програмного забезпечення в залежності від трудомісткості цих проектів із використанням іншого нормалізуючого перетворення, яке дозволить звузити довірчий інтервал та інтервал прогнозування нелінійної регресії в порівнянні з перетворенням у вигляді десяткового логарифму, яке використовується при побудові моделей COCOMO.

Література:

1.Ройс, У. Управление проектами по созданию программного обеспечения. Унифицированный подход [Текст] / У. Ройс. – Москва : "Лори", 2002. –424 с.

2.Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник. – 2-е изд., перераб. и доп. - М.: Финансы и статистика, 2005. - 544 с: ил

3.Гороховатський В.О., Дубницький В.Ю., Кобилін А.М., Лукін В.О., Москаленко О.В. Визначення трудомісткості при розробленні програмних комплексів. Збірник наукових праць «Системи обробки інформації» випуск 2 (118), 2014. – с. 92-98

4.Oligny, S. An empirical assessment of project duration models in software engineering [Text] / S. Oligny, P. Bourque, A. Abran. // In Proc. 8th European Software Control and Metrics Conference ESCOM. – Berlin, 1997.

5.Pardoe, Iain. Applied regression modeling [Text] / Iain Pardoe. – Wiley, 2012. –325 p.

Жмакін А. О., *студент*
Чернівецький національний університет імені Юрія Федьковича

МЕТОДИ МАШИННОГО НАВЧАННЯ ПРИ СЕМАНТИЧНІЙ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ

Автоматичний аналіз зображень поступово впроваджується у різні області людської діяльності, зокрема, в медицину, біологію, поліграфію. Для більшості існуючих алгоритмів такого аналізу характерне використання машинного навчання. Машинні методи навчання застосовуються для автоматизації процесу сегментації зображень, що особливо актуально при великих обсягах даних і складності ручної сегментації самих даних.

Семантичній сегментації зображень притаманний широкий спектр застосувань – від визначення сцен зображення і встановлення зв'язків між об'єктами до автономного водіння транспорту. Класичним алгоритмам сегментації характерна розбивка зображення на однорідні ділянки за низькорівневими атрибутами (такі як колір, яскравість, текстура) за деякими мірами подібності [1-7]. Така розбивка часто призводить до проблем: присутності однорідності атрибутів, яка погано узгоджується з семантикою зображення; присутності ступеня однорідності ділянок, квантування яких проводиться зазвичай за допомогою порогів для певної міри. Кращі результати для окремих завдань (в порівнянні з фіксованою сегментацією) досягаються ієрархічної сегментацією, яка представляє собою пов'язаний набір розбиття зображення. Тобто ієрархічна сегментація зображення являє собою дерево (або ліс) фрагментів зображення (сегментів, суперпікселей), яка має властивості: фрагмент-нащадка (тобто фрагмент, відповідний дочірній вершині в дереві), який цілком міститься у фрагменті-родичі; об'єднання всіх фрагментів, які мають однакову глибину, та в результаті виходить вихідне зображення.

Задачею є розгляд методів машинного навчання при семантичній сегментації зображень.

У деяких задачах машинного навчання (наприклад, при тривимірній реконструкції сцен) виникає необхідність ефективного зіставлення виділених точок на парі зображень. У зв'язку з цим, кожна досліджувана точка разом з деякою площею описується спеціальним числовим вектором (дескриптором цієї точки), після чого, між точками зі «схожими» дескрипторами, встановлюється відповідність. Також за допомогою дескрипторів можна проводити класифікацію зображень [4]. Такий метод машинного навчання отримав назву SURF дескрипторів.

Дескриптор повинен мати наступні властивості: перцептуальну однорідність, стійкість, характерність. Один з методів побудови дескрипторів точок запропоновано в роботі [4]; вся процедура розбивається на два етапи: встановлення орієнтації точки і опис точки за допомогою суми відгуків вейвлета Хаара.

Одним з поширених способів подання інформації в задачах категоризації текстів є гістограма bag-of-words [2, 8], де текст призначений для опису

документів за допомогою частот входження певних слів. Кожне слово відповідає певному набору тем, в яких воно вживається найчастіше. Якщо врахувати такі відповідності та розглянути гістограми входження слів, то навчена система буде мати властивість вгадування тем досліджуваних текстів. Застосування такого підходу в області машинного навчання при сегментації зображень вперше було використано в [10], а в роботі [6] показана його застосовність до розпізнавання об'єктів. При цьому, як текстовий словник використовується словник характеристик, який заснований на безлічі всіх властивостей тренувальних зображень. Тобто для опису зображення в термінах гістограми bag-of-words необхідно задати алгоритм вилучення характеристик і функцію пошуку.

До методів машинного навчання при семантичній сегментації зображень належить метод опорних векторів (Support Vector Machines (SVM)), який використовується для класифікації та регресійного аналізу. Даний метод передбачає машинне навчання з учителем. Також до методів машинного навчання з учителем належать: випадковий ліс (Random Forest) [7] і логістична регресія (Logistic Regression) [3], при цьому вхідні розмірність (кількість вхідних ознак) – фіксована.

Бінарній задачі класифікації, яка вирішується за допомогою SVM, характерні етапи: формалізації, завдання основної форми, виконання м'якого відступу, виконання нелінійної класифікації, а також застосування SVM для декількох класів зображень. Для виконання процедури тренування SVM застосовуються кілька способів, наприклад, метод стохастичного градієнтного спуску [9]. У на основі експериментальних результатів показано, що випадковий ліс краще за інших справляється із завданням семантичної сегментації при машинному навчанні [5].

У цілому, принципи роботи класичних методів машинного навчання при семантичній сегментації зображень полягають: у фіксуванні кількості найближчих сусідів, ознаки яких будуть використовуватися для класифікації кожного пікселя зображення; у визначенні ознак пікселя, які будуть використовуватися, наприклад значення RGB; у складанні навчальної вибірки, в якій кожному пікселю (і його мітці класу), для якого потрібно передбачити клас буде відповідати вектор ознак, що складається з ознак сусідів пікселя; у визначенні вектора ознак для пікселів, які знаходяться на границі зображення, з необхідністю в доповненні до необхідної розмірності за допомогою заповнення значення; в навчанні однієї або декількох моделей (в залежності від підходу).

При збільшенні фіксування кількості найближчих сусідів класифікатор буде отримувати більше інформації про сусідів пікселя, проте, все менше відмінностей для нього буде між сусідніми пікселями. При цьому, розмір фіксування кількості найближчих сусідів при такому підході можна варіювати від невеликого до розміру всього зображення (в такому випадку величини розмірів всіх зображень розподілу повинні збігатися, або повинна бути фіксація конкретного розміру і, якщо саме зображення менше, то має бути доповнене заповнюючими символами, а якщо більше – виконується обрізка) [11].

Слід відмітити, що класичним методам машинного навчання характерна обчислювальна неефективність через необхідність проходження вікнами по зображенню і, відповідно, нижча точність семантичної сегментації.

Література:

1. Gould S., Rodgers J., Cohen D. et al. Multi-class segmentation with relative location prior // International Journal of Computer Vision. 2008. Vol. 80, no. 3. pp. 300-316.
2. Joachims T. Text categorization with support vector machines: Learning with many relevant features // Machine Learning: ECML-98. 1998. pp. 137-142.
3. Agresti Alan. Logistic regression. Wiley Online Library, 2002.
4. Bay H., Tuytelaars T., Gool L. J. V. SURF: Speeded Up Robust Features // ECCV (1) / Ed. by A. Leonardis, H. Bischof, A. Pinz. Vol. 3951 of Lecture Notes in Computer Science. Springer, 2006. pp. 404-417.
5. Boruvka O. On a minimal problem // Prace Morask´e Pridovedeck´e Spolecnosti. 1996. Vol. 3. p. 128.
6. Csurka G., Dance C., Fan L. et al. Visual categorization with bags of keypoints // Workshop on statistical learning in computer vision, ECCV / Citeseer. Vol. 1. 2004. P. 22.
7. Liaw Andy, Wiener Matthew et al. Classification and regression by randomForest // R news. 2002. Vol. 2, no. 3. P. 18-22.
8. Lodhi H., Saunders C., Shawe-Taylor J. et al. Text classification using string kernels // The Journal of Machine Learning Research. 2002. Vol. 2. pp. 419-444.
9. Shalev-Shwartz S., Singer Y., Srebro N. Pegasos: Primal estimated sub-gradient solver for svm // Proceedings of the 24th international conference on Machine learning / ACM. 2007. pp. 807-814.
10. Sivic J., Zisserman A. Video Google: A text retrieval approach to object matching in videos. 2003. P. 203.
11. Thoma Martin. A survey of semantic segmentation // arXiv preprint arXiv:1602.06541. 2016.

Коваль О.В., *студент*
Черкаський державний бізнес-коледж, м. Черкаси
Відділення інформаційних технологій, студент
Горобинський А.С., *студент*
Черкаський державний бізнес-коледж, м. Черкаси
Відділення інформаційних технологій, студент

ЗНАЙОМСТВО З ІГРОВИМ ДВИГУНОМ UNREAL ENGINE 4

Одним із найбільш популярних і технологічних ігрових двигунів на сьогоднішній день є Unreal Engine 4, який славиться фотореалістичною графікою. З ним варто познайомитись, адже це надзвичайний інструмент, що використовують не лише в ігровій індустрії, а і у мультиплікації, фільмах і архітектурних візуалізаціях.

Unreal Engine 4 – ігровий двигун, розроблений компанією Epic Games для компаній будь-якого рівня [1]. Він є повністю безкоштовним і крім того open source-ним. За час свого існування він обріс широким комюніті та безліччю проектів, таких, як Fortnite, Hellbalde, Smite, A Plague Tale та безліч інших.

Він може приємно здивувати відразу після першого ж запуску. В очі кидається мигаюча кнопка при натисканні на яку користувач бачить безліч

уроків, що допомагають освоїтись в програмі. Готових, офіційних навіть не потрібно нікуди йти і щось шукати. Все відразу в коробці [2]. І ця фраза чудово описує двигун. Редактор матеріалів, що дозволяє створювати шейдери будь-якої складності; система ландшафта та флори, що надає можливість легко та швидко створювати природні сцени; візуальний скиптинг, що дозволяє написати гру навіть не знаючи жодної мови програмування. Ні в якому іншому двигуні немає такої кількості готових інструментів як тут. Звичайно, для інших двигунів існують рішення, створенні користувачами, наприклад, система флори для того ж Unity. Але про існування цього розширення необхідно дізнатись, знайти його, встановити і крім того постає питання наскільки добре все це інтегроване з самим двигуном. Тут же все створено розробниками і пов'язане одне з іншим.

Взагалі, двигун надзвичайно простий в освоєнні. В інтернеті безліч уроків, а рішення будь-якої проблеми шукається дуже швидко, особливо на Answerhub [3], принаймні, при наявності знання англійської мови хоча б на рівні PreIntermediate. А якщо ні, то все одно можна знайти необхідну інформацію, просто це буде складніше. Так само як і в будь-якій іншій сфері.

Окрім безлічі інструментів багато приємних деталей також створено за користувачів та інкапсуювано. Наприклад, кожен колись бачив такий ефект, коли сонце проходить крізь пил у повітрі і з'являється чіткий сонячний промінь. Все, що потрібно щоб додати його на свою сцену – поставити галочку напроти параметра Light Shaft Bloom в настройках джерела освітлення та, за необхідності, змінити значення двох інших параметрів. Іншим прикладом є стрибок персонажа. Як він взагалі робиться? Потрібно кожен кадр відслідковувати чи натиснута клавіша, коли вона натиснута перевірити чи не знаходиться персонаж у повітрі, якщо ні, то додати вертикальний вектор переміщення, що кожен кадр буде зменшуватись на значення змінної, що відповідає за гравітацію. А все, що потрібно зробити в UE4 – викликати вбудовану функцію Jump. І, можливо, підкрутити пару параметрів, якщо потрібен незвичний результат, наприклад можливість керувати персонажем у повітрі під час падіння.

Може здатися, що UE4 – більше схожий на конструктор і вийти за межі вбудованих можливостей не можна. Але це не так. Ніхто не забороняє написати всю гру на C++ чи використовувати CPP/BP підхід, або, навіть, дописати двигун під себе. Навпаки, до цього навіть заохочують відкритістю вихідного коду.

UE4 використовують безліч компаній різного рівня по всьому світу. А враховуючи безкоштовність, широкий функціонал, що йде відразу в комплекті та простоту освоєння, він також чудово підійде для інді-студій та, навіть, соло-розробників.

Література:

1. Опис двигуна Unreal Engine 4 [Електронний ресурс] – Режим доступу до ресурсу: https://www.unrealengine.com/en-US/

2. Інструментарій двигуна Unreal Engine 4 [Електронний ресурс] – Режим доступу до ресурсу: https://www.unrealengine.com/en-US/features.

3. Answerhub [Електронний ресурс] – Режим доступу до ресурсу: https://answers.unrealengine.com/index.html

Кравченко А.С., *студент,*
Носовець О.К.,*канд. тех. наук, доцент*
Національний Технічний Університет України «Київський Політехнічний Інститут імені Ігоря Сікорського», м. Київ
Кафедра біомедичної кібернетики, студент

МЕТОДИ ДІАНОСТИКИ ТИПУ ПУХЛИН ГОЛОВНОГО МОЗКУ

Анотація. Проведено аналіз результатів обстеження магнітно-резонансної томографії у хворих з різним ступенем злоякісних гліом. Розроблені математичні моделі для диференціації гліом за ступенем злоякісності

Ключові слова: томографія, моделювання, гліоми.

1. Постановка проблеми.

У теперішній час кількість випадків через злоякістні пухлини головного мозку постійно зростає [1]. В сучасній нейрорадіології та нейрохірургії є актуальним питання вдосконалення, оцінки ефективності та оптимізації використання сучасних методик неінвазивної діагностики [2], [3] з використанням інформаційних технологій. Створення математичних моделей для визначення типу гліом за вхідними параметрами пацієнта, в подальшому може бути використано задля визначення типу новоутворень головного мозку та подальшого лікування.

2. Матеріали та методи

Робота заснована на результатах – обстеження магнітно-резонансної томографії (МРТ) з дифузійно-тензорною магнітно-резонансною томографією (ДТТ) та МР-трактографією 34 здорових осіб та 121 хворого на внутрішньомозкові пухлини в Державній установі «Інститут нейрохірургії ім. акад. А.П. Ромоданова НАМН України», отриманих у відділі нейрорадіології та радіонейрохірургії, відділенні радіонейрохірургії в період часу з 2012 року по 2016 рік.

Аналізувалися результати дифузійно-тензорної томографії, що дозволяє кількісно оцінити процеси, що відбуваються у білій та сірій речовині мозку, трактом виміру середнього коефіцієнта дифузії (ВКД) та фракційної анізотропії (ФА). Додатково проаналізовані показники асиметрія між параметрами дифузії між правим та лівим КСТ. хворих пацієнтів додатково фіксувались зміщення, здавлення, набряк волокон тракту.

3. Основна частина

Оскільки існує статистично значима різниця середніх значень показника ФА ($p<0,05$) в гліомах з II, III та IV типом та у здорових осіб, то можна побудувати математичні моделі (класифікатори) [4], що будуть диференціювати ступінь злоякісності (анаплазії) гліоми за показником ФА з певним ступенем достовірності.

Для побудови диференціальних класифікаторів використано розвідувальний статистичний аналіз на основі методу бінарної логістичної

регресії, який дав можливість отримати значення ймовірності віднесення гліоми до певного типу (ступеня анаплазії).

Включення параметра до моделі виконувалось шляхом покрокового включення Вальда, за допомогою якого з набору характеристик, використаних для моделювання, відібрані ті, що найбільше характеризують пухлину з певним ступенем злоякісності. Для покращення якості моделі в набір незалежних змінних були штучно введені параметри, що являють собою комбінації характеристик.

Математична модель за бінарною логістичною регресією має наступний вигляд:

$$p = \frac{1}{1+e^{-z}},$$

де $z = a_n \cdot x_n + a_{n-1} \cdot x_{n-1} + ... + a_0$,

p – вірогідність віднесення гліоми до певного типу;

e = 2,718 – експонента, основа натуральних логарифмів,

z - показник ступеню, що визначає рівень впливу врахованих характеристик;

$a_{1...n}$ – вагові коефіцієнти рівняння регресії;

$x_{1...n}$ – показники, що визначають ступінь анаплазії гліоми.

Значення p = 0,5 свідчить про рівну ймовірність настання або не настання події, в даному випадку віднесення гліоми до певного ступеню анаплазії. При значенні p<0,5 ймовірність віднесення гліоми до певного ступеню анаплазії незначна, при p>0,5 – навпаки.

1. Класифікатор, що диференціює гліому II ступеня анаплазії.

$$p = \frac{1}{1+e^{-z}},$$

де $z = 140{,}119 \cdot x_1 + 26{,}336 \cdot x_1 \cdot x_2 - 10{,}668 \cdot x_1 \cdot x_3 - 193{,}572 \cdot x_1^2 - 24{,}172$

x_1 - показник дифузії ФА;

x_2 - зміщення шляху;

x_3 - здавлення шляху.

Якість моделі оцінювалась за показниками функції подібності, за результатами якої розраховані значення показника хі-квадрат (63,141, p=0,001).

Аналіз прогностичної можливості побудованої моделі дозволив отримати показники чутливості − 84,3%, специфічності − 85,7%, точності – 84,9%.

2. Класифікатор, що диференціює гліому III ступеня анаплазії.

$$p = \frac{1}{1+e^{-z}},$$

де, $z = -34{,}509 \cdot x_1 - 3{,}950 \cdot x_1 \cdot x_2 - 4{,}311 \cdot x_3 - 0{,}353 \cdot x_4 + 55{,}295 \cdot x_1^2 + 4{,}630$

x_1 - показник дифузії ФА;

x_2 - зміщення шляху;

x_3 - здавлення шляху.

x_4 - набряк волокон шляху.

Якість моделі оцінювалась за показниками функції подібності, за результатами якої розраховані значення показника хі-квадрат (20,111, p=0,001).

Аналіз прогностичної можливості побудованої моделі дозволив отримати показники чутливості – 83,3%, специфічності 73,1%, точності – 80,2%.

3. Класифікатор, що диференціює гліому IV ступеня анаплазії.

$$p = \frac{1}{1+e^{-z}},$$

де $z = -11{,}732 \cdot x_1 \cdot x_2 + 7{,}176 \cdot x_1 \cdot x_3 + 2{,}614 \cdot x_4 - 56{,}520 \cdot x_1^2 + 1{,}976$

x_1 - показник дифузії ФА;

x_2 - зміщення шляху;

x_3 - руйнування волокон шляху;

x_4 - набряк волокон шляху.

Якість моделі оцінювалась за показниками функції подібності, за результатами якої розраховані значення показника хі-квадрат (38,230, p=0,001).

Аналіз прогностичної можливості побудованої моделі дозволив отримати показники чутливості – 85,2%, специфічності – 76%, точності – 82,6%.

Висновки. Виявлена статистично достовірна різниця середніх значень показника ФА у гліомах з різним ступенем злоякісності дала змогу розробити математичну модель для диференціації гліом за ступенем злоякісності G II, G III, G IV. Аналіз прогностичної можливості побудованої моделі для гліом з G II дозволив отримати показники чутливості − 84,3%, специфічності − 85,7%, точності – 84,9%; для гліом з G III показники чутливості – 83,3%, специфічності 73,1%, точності – 80,2%; для гліом з G IV показники чутливості – 85,2%, специфічності – 76%, точності – 82,6%.

Література:

1. L. G. Z.P. Fedorenko, «Byuleten Natsyonalnoho kantser-reyestru [Bulletin of the National Chancery Register],» Rak v Ukrayinii, 2015-2016, № 18, pp. 1-2, 2015.

2 F. A. M. B. R. Cordella, Intraoperative neurophysiological monitoring of the cortico-spinal tract in image-guided mini-invasiveneurosurgery, Clin. Neurophysiol, 2013.

3 C. Lakhmi, "Introduction," in Intelligent Computing, Communication and Devices, Springer, 2009, pp. 416-418.

4 K. K. P. N. P. B. Z. Li, «Diffusion tensor tractography of the arcuate fasciculus in patients with brain tumors: comparison between deterministic and probabilistic models,» J. Biomed. Sci. Eng., № 6, pp. 192-200, 2013.

Кузнецов Д.В.
Национальный технический университет Украины
«Киевский политехнический институт имени Игоря Сикорского», г. Киев
Кафедра технической кибернетики, студент

ИСПОЛЬЗОВАНИЕ JAVASCRIPT КАК ЦЕЛЕВОГО ЯЗЫКА КОМПИЛЯЦИИ

Последний опрос разработчиков Web-приложений «State of JavaScript» показал, что половина из них использовала компилируемый в JavaScript типизированный язык программирования TypeScript. В результате своего развития JavaScript оказался востребован для разработки сложных приложений, однако он не был спроектирован для построения сложных систем, и поэтому затраты на их поддержку постоянно и нелинейно росли, что и спровоцировало появление типизированного языка, который легко бы интегрировался с JavaScript, и при этом предоставлял бы возможности для статической проверки корректности программ.

При этом, существует огромное количество языков, компилируемых в JavaScript. Это могут быть как простые системы типов (Flow), или попытки «улучшить» язык (CoffeeScript), так и реализации существующих языков программирования для работы в браузере (ClojureScript – Clojure, Scala.js – Scala, Emscripten – C/C++). Существуют также и новые, по большей мере представители семейства ML, функциональные языки, направленные на избежание ошибок в работе программ при помощи развитой системы типов (Elm, Reason).

Более того, в последнее время был опробован подход разработки пользовательских интерфейсов без традиционных для этой области фреймворков. Это было сделано из соображений экономии размера конечного файла JavaScript (исключение кода самого фреймворка), так и для применения характерных для построения интерфейсов методик и парадигм, которые семантически слишком далеки от JavaScript. Ярким примером этого подхода служит Svelte – язык реактивного программирования.

Популярность такого подхода исторически вытекает из необходимости поддерживать множество Web-браузеров, которые, в свою очередь, реализовывали разные редакции стандарта ECMAScript. Для этой проблемы нашли элегантное решение: использовать старые, широко поддерживаемые, редакции стандарта как целевой язык для компиляции кода, написанного для более новых версий. Такие программы назвали транспайлерами (transpilers), и они используются повсеместно в современной Web-разработке.

Этот тренд обусловлен несколькими факторами: во-первых, несмотря на то что язык не подходит для написания больших приложений, гибкость JavaScript (слабая типизация, функции как объекты первого класса, широкая поддержка метапрограммирования) делают компиляцию в него достаточно простой. Во-вторых, JavaScript – одна из наиболее распространеных сред выполнения – он есть в каждом браузере. Наконец, в последнее время

наблюдается тенденция реализации Web-приложений как наименее затратного способа обеспечить кроссплатформенность, и все больше компаний выпускают свои продукты именно в таком формате. Современные браузеры – это невероятно сложные и продвинутые системы выполнения кода, которые при этом обеспечивают средства коммуникации с целевой платформой. В таком смысле, можно действительно провести аналогию между JavaScript и Assembly Language.

Список использованных источников:

1. JavaScript myth: JavaScript needs a standard bytecode [Электронный ресурс]. – Режим доступа: http://2ality.com/2012/01/bytecode-myth.html
2. JavaScript is Web Assembly Language and that's OK. [Электронный ресурс]. – Режим доступа: http://www.hanselman.com/blog/JavaScriptIsWebAssemblyLanguageAndThatsOK.aspx

Кузнецов Д.В.
Национальный технический университет Украины
«Киевский политехнический институт имени Игоря Сикорского», г. Киев
Кафедра технической кибернетики, студент

РАЗМЕТКА ДАННЫХ ДЛЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА: ОГРОМНЫЙ И МАЛОЗАМЕТНЫЙ РЫНОК

Мировая тенденция внедрения подходов и технологий, основанных на интеллектуальной обработке данных и искусственном интеллекте, уже очевидна всем: гиганты в сфере информационных технологий постоянно подчеркивают переориентирование направления их развития именно в эти отрасли, а постоянные изобретения, исследования и разработки в этой области, кажется, привлекают все внимание общества к себе. Однако, при таком пристальном внимании к результатам работы искусственного интеллекта, мы упускаем не менее важный элемент процесса работы этих технологий – то, как формируются исходные массивы данных.

Действительно, для того чтобы любой алгоритм машинного обучения смог точно предсказывать или распознавать хоть что-то, ему необходимы заранее подготовленные данные, которые «подскажут» ему характеристики или оценки данных, которые он принимает. В индустрии подобный процесс подготовки данных называется разметкой. В то время как возможна и вполне успешно применяется автоматизация обработки исходных данных, например, если эти данные по своей природе структурированы, в случае разметки наиболее близких к человеческому восприятию неструктурированных данных, автоматизация невозможна без применения искусственного интеллекта, для которого эти данные полагались изначально. Предприятия разорвали этот порочный круг с помощью низкооплачиваемой и низкоквалифицированной рабочей силы, которую публицисты назвали «современными издольщиками».

Помимо того, насколько изнурительной может показаться эта работа, она также несправедлива: работники, получающие оплату единоразово, создают данные, которые компании могут использовать многократно, из-за чего потенциальная прибавочная стоимость продуктов, основанных на данных, может быть колоссальной.

С экономической точки зрения, картина достаточно проста: создатели продуктов, основанных на данных, которые базируются в развитых странах, используют труд людей из бедных и малоразвитых стран для минимизации расходов на подобную работу. По сути, это своеобразный феномен: компании создают рабочие места для низкоквалифицированных работников, которых вытеснила из других отраслей автоматизация, которой эти компании добивались изначально.

Эксперты затрудняются оценивать этот рынок точно, однако очевидно одно: эта отрасль растет следом за информационными технологиями, которые ее породили, и прогнозируется, что к 2023 году разметка данных станет многомиллиардным рынком, который будет иметь значительное влияние на экономику слаборазвитых стран.

Список использованных источников:

1. The AI sharecroppers [Электронный ресурс]. – Режим доступа: https://www.axios.com/the-ai-sharecroppers-b316d333-ce00-47a1-afd5-219d6138461e.html
2. Data Engineering, Preparation, and Labeling for AI 2019 [Электронный ресурс]. – Режим доступа: https://www.cognilytica.com/2019/03/06/report-data-engineering-preparation-and-labeling-for-ai-2019
3. Mary L. Gray, Siddharth Suri. Ghost Work: How to Stop Silicon Valley from Building a New Global Underclass. Boston: Houghton Mifflin Harcourt, 2019. – 288p.

Кузнецов Д.В.
Национальный технический университет Украины
«Киевский политехнический институт имени Игоря Сикорского», г. Киев
Кафедра технической кибернетики, студент

ПРИМЕНЕНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В СОВРЕМЕННОЙ НАУКЕ

Современные эксперименты в физике и астрономии характеризуются невероятным количеством генерируемой информации, с которой люди уже не могут справиться. В некоторых из них ежедневно записываются терабайты данных – и поток будет только усиливаться. KSA (Square Kilometre Array), радиотелескоп, запуск которого планируется в 2020-х годах, будет ежегодно производить примерно столько же информации, сколько производит нынешний интернет.

В свете этого, многие ученые изучают применение технологий обработки данных в их прикладной области. С минимальным влиянием человека, системы

искусственного интеллекта, например, нейронные сети, могут эффективно обрабатывать огромные массивы данных, находя аномалии и выявляя закономерности, которые было бы невозможно обнаружить человеческим глазом.

Разумеется, компьютеры в науке применяются достаточно давно, а выявление закономерностей в данных – один из основных способов построения гипотез, однако некоторые исследователи говорят, что искусственный интеллект в корне изменит подход к научным исследованиям. Одним из таких подходов может стать генеративное моделирование, может помочь определить наиболее вероятное теоретическое обоснование изучаемых результатов экспериментов из всех возможных. Энтузиасты верят в инновационность этого моделирования и утверждают, что это станет «третьим способом познания вселенной».

Исторически, мы изучали природу, наблюдая за ней. Вторым успешно применяемым в исследованиях методом стала симуляция. И наблюдение, и симуляция позволяют нам строить гипотезы, которые могут быть экспериментально проверены с помощью дальнейших наблюдений. Генеративное моделирование отличается от обоих этих подходов.

В сущности, в процессе генеративного моделирования входные данные (обычно изображения) разбиваются их на базовые, абстрактные элементы, которые специалисты называют «пространством возможностей» данных. Затем, алгоритм использует полученные блоки для того, чтобы изучить, как их изменение влияет на данные и на основе наблюдений делаются выводы о возможных взаимосвязях и закономерностях.

Многие ученые настроены скептично и воспринимают подобные технологии просто как дополнительные инструменты для проведения исследований, однако большинство все же согласно с тем, что технологии искусственного интеллекта оказывают огромное влияние на науку, которое будет только расти. Некоторые ученые даже выражают обеспокоенность тем, что вскоре даже исследовательскую деятельность можно будет автоматизировать и научные исследования станут уделом компьютеров.

Список использованных источников:

1. Robot makes scientific discovery all by itself [Электронный ресурс]. – Режим доступа: https://www.wired.com/2009/04/robotscientist/

2. The AI robot chemist trying to find the origins of life on Earth [Электронный ресурс]. – Режим доступа: https://www.wired.co.uk/article/robot-chemist-life-on-earth

3. Artificial Intelligence Learns to Learn Entirely on Its Own [Электронный ресурс]. – Режим доступа: https://www.quantamagazine.org/artificial-intelligence-learns-to-learn-entirely-on-its-own-20171018/

Куперштейн Л.М., *к. т. н., доцент*
Лукічов В.В., *к. т. н., старший викладач*
Айвазян С.А., *студент*
Вінницький національний технічний університет, м. Вінниця
Кафедра захисту інформації

СИСТЕМА ДВОФАКТОРНОЇ АВТЕНТИФІКАЦІЇ НА ОСНОВІ ГОЛОСУ

У міру розвитку комп'ютерних мереж і розширення сфер автоматизації цінність інформації неухильно зростає. Популярність глобальної мережі Інтернет, з одного боку, відкриває величезні можливості для електронної взаємодії, але, з іншого боку, створює потребу в більш надійних засобах захисту корпоративної інформації. На сьогоднішній день для надійного захисту від несанкціонованого доступу використовується двофакторна автентифікація, яка забезпечує доступ до ресурсу зазвичай на основі паролю (1-й фактор) та ще одного унікального ідентифікатора (одноразовий пароль, апаратний токен, біометричний тощо) [1].

Пропонується система двофакторної автентифікації на основі символьного паролю та голосового токену (рис. 1).

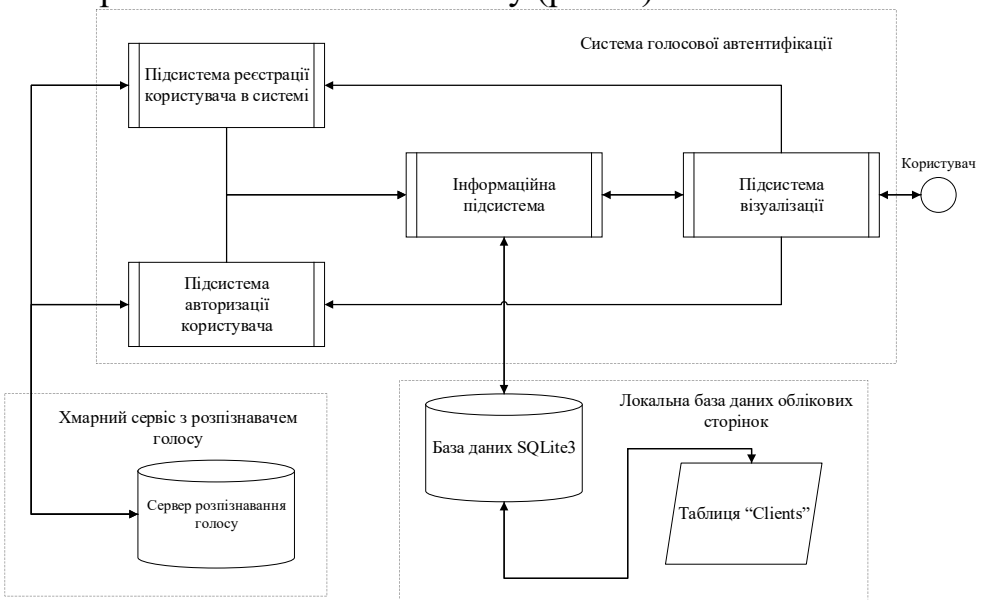


Рис. 1. Архітектура системи двофакторної автентифікації на основі голосу

Система двофакторної автентифікації на основі голосу складається з наступних підсистем:

– підсистема реєстрації користувача в системі – призначена для реєстрації користувачів в систему;

– підсистема візуалізації – призначена для візуального представлення результатів розрахунку;

– інформаційна підсистема – необхідна для зберігання даних на всіх етапах роботи підсистеми та забезпечення взаємодії між іншими підсистемами;

– підсистема авторизації користувача – виконує операції автентифікації користувача до облікового запису.

У системі реалізовано дві основні функції – створення облікового запису (рис. 2.а) та авторизація у систему (рис. 2.б).

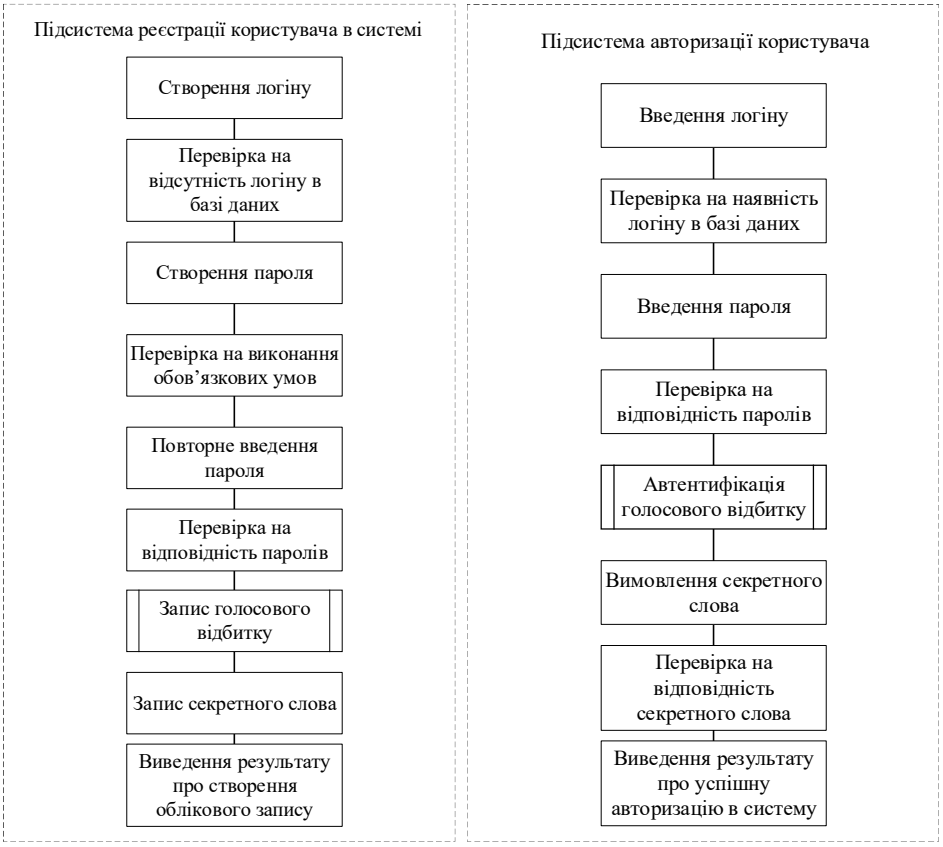


Рис. 2. Схема роботи підсистем:
а) реєстрації користувача в системі, б) авторизації користувача

При виконанні обох функцій, користувач повинен вводити свої дані, зокрема логін, пароль у вікна запиту, записувати «голосовий відбиток» та вимовити секретне слово для подальшої авторизації. Далі виконується шифрування локальних даних, їх передача у базу даних системи автентифікації, передача «голосового відбитку» до хмарного сервісу, що виконує завдання розпізнавання голосового повідомлення, виконання операції читання з бази даних для підтвердження усіх етапів автентифікації та надання гарантії щодо ідентифікації «голосового відбитку», створеного користувачем при авторизації.

При реалізації системи двофакторної автентифікації використовувалась мова програмування Python, а також інструментальні засоби – PyQt5 (GUI), PyAudio (робота із аудіо), SQLite3 (база даних), matplotlib (робота із математичними моделями), Speech Recognition (робота із хмарним сервісом ідентифікації голосового повідомлення). При тестуванні системи, а саме процесів створення облікового запису та авторизації у систему їх тривалість склала від 90 секунд до 3 хвилин, а достовірність автентифікації користувача – 85%.

Недоліком автентифікації по голосу являється низька точність методу, через різноманіття проявів голосу людини (хвороба, шумне довкілля, імітація чужого голосу, зміна тембру голосу) [3], що накладає жорсткі вимоги на використання системи.

Література:
1. Лужецький В. А., Кожухівський А. Д., Войтович О. П. Основи інформаційної безпеки: навчальний посібник – Вінниця: ВНТУ, 2013. – 221 с.
2. Біометричні технології: міфи та реальність. [Електронний ресурс]. – Режим доступу: URL, http://www.cprspb.ru/bibl/object/15.htm – Назва з екрану.
3. Шаров В. Біометричні методи комп'ютерної безпеки. [Електронний ресурс]. – Режим доступу: https://www.bytemag.ru/articles/detail.php?ID=6719 – Назва з екрану.

Лісовенко А.І., *к.т.н. асистент*
Жирома М.В., *студент*
Вінницький національний технічний університет, м. Вінниця
Кафедра автоматизації та інтелектуальних інформаційних технологій

ПРОГРАМНА РЕАЛІЗАЦІЯ ФОРМАЛЬНОЇ ТЕОРІЇ ЗАСОБАМИ NLTK PYTHON

Інтелектуальні лінгвістичні системи в наш час набувають все більшого поширення, що зумовлено швидким розвитком світової інформаційно-комунікаційної системи. Особливо це стосується засобів інтелектуального автоматизованого опрацювання природної мови. Накопичення значних текстових масивів у глобальних мережах, прогрес у галузі створення технологій опрацювання інформації спричинили зростання актуальності досліджень, спрямованих на автоматичний аналіз, опрацювання текстових даних. [1]

Актуальним завданням при роботі з неструктурованою текстовою інформацією стає знаходження відповіді на запитання користувача, поставлене природною мовою, до того ж, не позбавлене сенсу. Проте більшість існуючих систем надає відповідь або у вигляді переліку документів в яких найімовірніше знаходиться відповідь; або у вигляді цілих речень, які не дають точної відповіді; або у вигляді набору слів. Тому актуальною стає задача надання короткої відповіді. Для наочного відображення роботи система будуватиме граф проаналізованого речення. Маючи такий граф та питальні займенники цього розбору можна реалізувати задачу знаходження розгорнутої від питального займенника відповіді на питання.

Для формалізації пошуку розгорнутої від питального займенника відповіді на питання використаємо формальну теорію як прикладну теорію першого порядку Th та три додаткові теореми, які наведені нижче.

Теорема 1. < Терм >$\rightarrow$< АНФтерм >. Будь-який вираз у формі < Терм > можна перетворити у асоціативну нормальну форму .

Теорема 2. < АНФтерм >$\rightarrow$< АНФq >$\oplus$< АНФ? > $\oplus$ < АНФа > , де < АНФ?> – позначаємо <АНФω> = $x_i \setminus x_j \mid x_i\, x_j$ Al для зручності користування; < АНФа > – всі елементарні терми з < АНФтерм >, в яких символ x_j є першим, потім рекурсивно підставляється наступний символ за принципом пошуку у глибину, але, якщо в рекурсії знаходиться < АНФ? >= $x_i \setminus x_j$, то ця гілка пошуку переривається (символ x_i та всі наступні за ним не враховуються); < АНФq > –

всі інші окрім < АНФ? > ⊕ < АНФа > елементарні терми, що складають <АНФтерм > .

Теорема 3. < АНФа^j >→< АНФ$\mathbf{a}_1^j$ > ⊕ < АНФ$\mathbf{a}_2^j$ > ,де < АНФа^j > – піддерева елементарних термів, які відповідають умовам Теореми 2 та для яких символ j х є кореневим; < АНФ$\mathbf{a}_1^j$ > та < АНФ$\mathbf{a}_2^j$ > – елементарні терми, які відповідають принципу побудови < АНФа ^{j}> , але знайдені у двох різних термах <АНФтерм$_1$ > та < АНФтерм$_2$ >. Вербальне формулювання теореми 3: будь-яка конструкція відповіді на питання < АНФа^j > може бути побудована за допомогою операції об'єднання конструкцій ⊕ з елементарних термів <АНФ$\mathbf{a}_1^j$ > та < АНФ$\mathbf{a}_2^j$ >, для яких символ j х є кореневим та які знайдені у двох різних термах (реченнях тексту) < АНФтерм$_1$ > та < АНФтерм$_2$ > [2].

На рисунку 1 наведено приклад графічного розбору речення: «When Ukraine first became an independent country, what form of government did it adopt?».

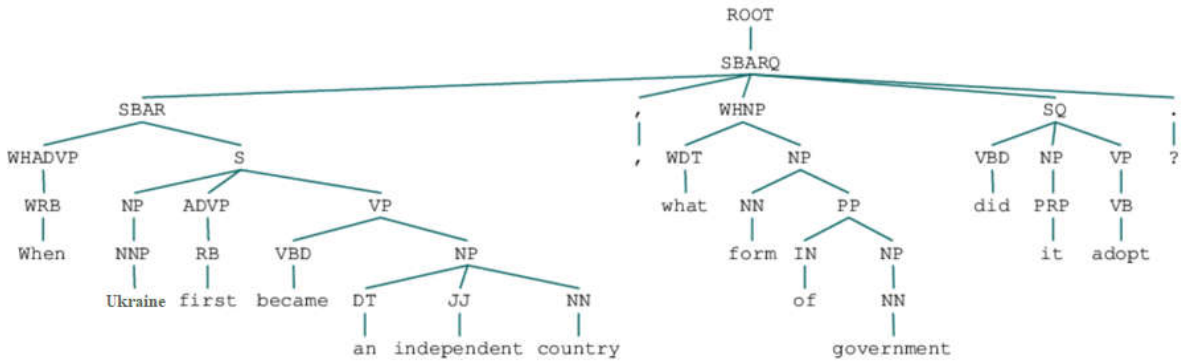


Рисунок 1 – приклад графічного розбору речення

Метою роботи є підвищення ефективності інтерактивної взаємодії користувача з системою типу «запитання-відповідь» (Question-Answering Systems) на основі формальної моделі пошуку відповіді. Також даній роботі було реалізовано здатність графового розбору речення для отримання розгорнутої від запитального займенника відповіді на питання користувача основаного на прикладній теорії першого порядку.

Література:

1. Шаров С.В. Сучасний стан розвитку інтелектуальних інформаційних систем / С.В. Шаров // Вісник Чернігівського національного педагогічного університету. Серія: Педагогічні науки. – №130. – 2015. – С. 114.

2. Лісовенко А.І. Інформаційна технологія підтримки функції «запитання-відповідь» на основі образного аналізу фахових текстів : дис. на здобуття ступеня канд. техн. наук : спец. 05.13.06 «Інформаційні технології» / Лісовенко Анна Ігорівна. – Вінниця , 2017. – 82-84 с.

Лютак І.З., *докт. техн. наук, професор*
Івано-Франківський національний технічний університет нафти і газу,
м. Івано-Франківськ
Кафедра інженерії програмного забезпечення, професор

ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ МЕТОДУ КОНТРОЛЮ НЕОДНОРІДНОСТЕЙ УЛЬТРАЗВУКОВИМ МЕТОДОМ

Обробка сигналів відіграє надзвичайно важливу роль в побудові пристроїв для ультразвукового неруйнівного контролю пружних властивостей металу стінок магістральних трубопроводів. Область застосування теорії обробки сигналів полягає у пошуку таких структур виміряних сигналів, що зв'язують параметри ультразвукових хвиль та об'єкта контролю.

Результатом досліджень є розроблений метод обробки сигналу від ультразвукової спрямованої кільцевої хвилі (УСКХ) з використанням вейлет аналізу, рис. 1.

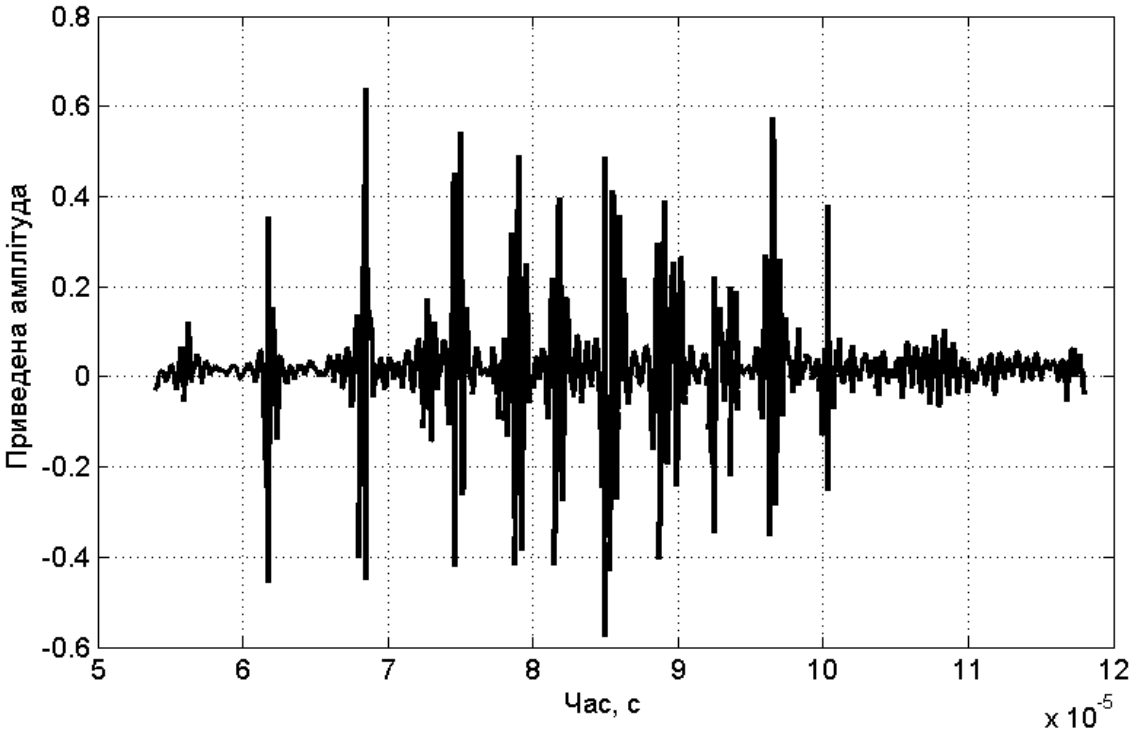


Рис. 1. Сигнал після обробки ультразвукової спрямованої кільцевої хвилі вейвлет-перетворенням. Вейвлет DB 9, рівень 5, коефіцієнт 5-го рівня = 1.4, 4-го рівня = 1.4, пороговий метод плавний

Вхідні дані:
- діаметр 1420 мм;
- товщина стінки виконавча 19.5 мм;
- марка сталі Х60;
- частота ультразвукових первинних перетворювачів 2.5 МГц.
- кількість усереднень сигналу 10.

- відстань між центрами чутливих елементів первинних перетворювачів 300 мм із врахуванням вигину віддалі поширення хвиль викликаного радіусом труби.

- швидкість ультразвуку в шарі акустичного контакту 1963 м/с.

Для декомпозиції сигналу УСКХ із метою аналізу його параметрів за допомогою вейвлет перетворень було розроблено дерево декомпозиції із порогом 2. Виявлено, що найбільше енергії сигналу зосереджено у деталізації третього рівня 79 %. Представлення розподілу коефіцієнтів деталізації сигналу УСКХ на площині показує щільність та їх місце в часі [1].

В результаті представленого аналізу можна оцінити вплив завад на сигнал УСКХ та визначити параметри завад, що показано як за енергією неінформативного сигналу, так і розподілу вейвлет коефіцієнтів на площині. Визначено, що інформативний сигнал є в коефіцієнтах деталізації третього рівня, інші рівні деталізації та апроксимації містять завади у відповідному енергетичному співвідношенні. При подальшому аналізі встановлено, що інформативний сигнал можна реконструювати за допомогою 40 коефіцієнтів апроксимації 5-го рівня та 3-х найбільших коефіцієнтах третього рівня деталізації.

Література:

1. Lyutak I. Wavelet analysis of ultrasonic guided waves in pipeline inspection / I. Lyutak // IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Sofia, Bulgaria.- 2005.- pp. 517-523.

Макаренко О.І., *кандидат економічних наук, доцент,*
Запорізький національний університет, м. Запоріжжя
Татосян Т.А.
Запорізький національний університет, м Запоріжжя

АНАЛІЗ ДИНАМІКИ ВИТРАТ НА ОХОРОНУ ПРИРОДНОГО НАВКОЛИШНЬОГО СЕРЕДОВИЩА У ЗАПОРІЗЬКІЙ ОБЛАСТІ

Запорізький регіон є одним з найбільших промислових центрів України з потужним індустріальним комплексом. У м. Запоріжжя зосереджені підприємства металургійної, машинобудівної, хімічної, переробної сфер економічної діяльності. Виробнича діяльність підприємств призводить до значної кількості викидів у навколишнє середовище, які спричиняють забруднення атмосферного повітря, водних і земельних ресурсів та негативно впливають на здоров'я населення. Проте, промисловий комплекс Запорізької області, що дістався у спадок від Радянського союзу, потребував та досі потребує оновлення. Заміна старих витратних, енергоємних технологій та устаткування потребує значних фінансових вкладень. Отже, моніторинг та аналіз динаміки еколого-економічних показників розвитку території з метою

розробки та впровадження природоохоронних заходів є важливим науково-практичним завданням.

За роки незалежності державні органи влади намагалися вирішити проблему фінансування заходів щодо охорони навколишнього середовища на територіях з високим рівнем забруднення. Одним із позитивних кроків державної підтримки було прийняття положення «Про порядок проведення еколого-економічного експерименту в містах Кривий Ріг, Дніпродзержинськ, Маріуполь і Запоріжжя» [1]. Джерелами фінансування визначалися кошти фондів охорони навколишнього природного середовища усіх рівнів, що отримано за рахунок надходження платежів від збору за забруднення навколишнього природного середовища, інші бюджетні кошти, в тому числі з Державного бюджету за державними програмами та власні кошти підприємств.

Основною метою еколого-економічного експерименту було виведення означених міст з екологічної кризи. Проте, з 2006 р. проведення цього експерименту припинено. Динаміку витрат на охорону природного навколишнього середовища у Запорізькій області представлено на рис. 1.

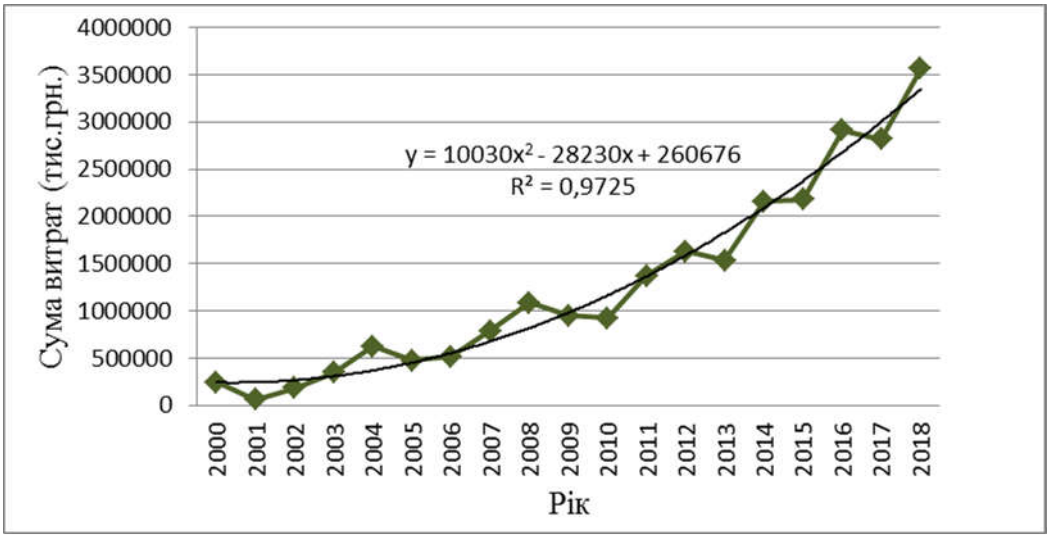


Рисунок 1 – Динаміка витрат на охорону природного навколишнього середовища у Запорізькій області, 2000-2018 р.

Джерело побудовано автором на основі [2,3].

Аналіз динаміки витрат свідчить про те, що витрати на охорону природного навколишнього середовища у Запорізькій області з 2000 р. по 2018 р. зросли у 14,6 разів. Візуальний аналіз динаміки витрат свідчить про наявність коливань, оскільки на рис. 1 представлено річні данні, то можна висунути гіпотезу про наявність певних циклічних коливань досліджуваного показника. Проте, така циклічність спостерігається лише до 2013 р., а починаючи з 2014 р. відмічаються стрибкоподібні зміни. Авторами було побудовано лінійну, експоненційну та параболічну трендові моделі та визначено, що параболічна модель (рис.1) найкраще описує тенденцію зміни

витрат. Встановлено, що якість моделі висока, оскільки коефіцієнт детермінації $R^2=0,97$, тобто параболічна модель тренду описує 97 % вихідних даних.

Аналіз динаміки складових витрат на охорону природного навколишнього середовища у Запорізькій області свідчить про те, що у 2000 р. поточні витрати перевищували обсяг капітальних інвестицій на охорону навколишнього природного середовища майже у 8 разів, а у 2018 р. поточні витрати перевищували капітальні інвестиції у 2,3. Динаміку структури витрат представлено на рис 2.

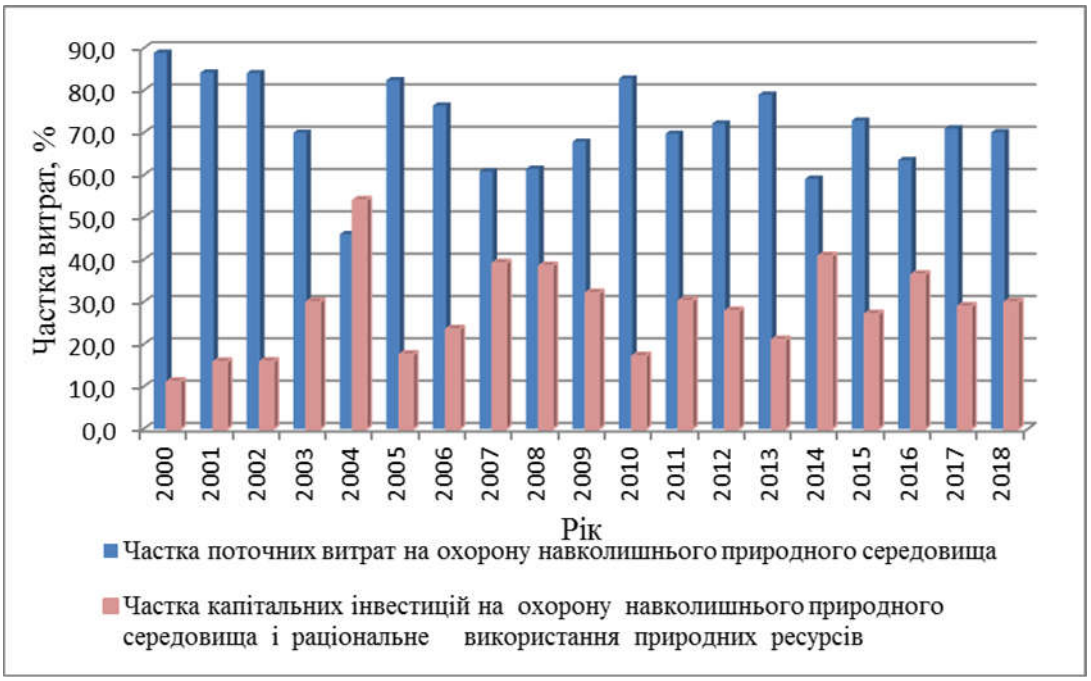


Рисунок 2 – Динаміка структури витрат на охорону природного навколишнього середовища у Запорізькій області, 2000-2018 р.

Джерело побудовано автором на основі [2,3].

Структура витрат постійно змінюється, так, у 2000 р. поточні витрати складали 88,9 %, а капітальні інвестиції – 11,1 %, а у 2018 р. 70,1 % та 29,9 % відповідно, при чому у 2004 р. спостерігалася кардинальна зміна структури (поточні витрати – 46 %, капітальні інвестиції – 54 %).

Перспективою подальших досліджень авторів є визначення впливу витрат на природоохоронні заходи на стан навколишнього середовища у Запорізькій області.

Література:

1. Постанова КМУ «Про проведення еколого-економічного експерименту в містах Кривий Ріг, Дніпродзержинськ, Маріуполь і Запоріжжя» від 28 квітня 1999 р. N 715 – Електронний ресурс. Режим доступу: https://zakon.rada.gov.ua/laws/show/715-99-%D0%BF

2. Навколишнє природне середовище. Регіональна статистика. – Електронний ресурс. Режим доступу: http://www.ukrstat.gov.ua/

3. Навколишнє середовище. – Електронний ресурс. Режим доступу: http://zp.ukrstat.gov.ua/index.php/statystychna-informatsiia

Марочканич О.Р.
Національний технічний університет України
“Київський Політехнічний Інститут ім. Ігоря Сікорського”, м. Київ
Кафедра автоматизованих систем обробки інформації та управління,
студент

ТЕХНОЛОГІЇ ОБЧИСЛЕНЬ ДЛЯ ВИЗНАЧЕННЯ МІЦНОСТІ БУДІВЕЛЬНИХ КОНСТРУКЦІЙ

Вступна частина

При проектуванні об’єктів різного призначення часто виникає необхідність в проведенні розрахунків міцності конструкції таких об’єктів або окремих її елементів. В першу чергу задачі розрахунку міцності конструкцій виникають у промисловому та цивільному будівництві – при розрахунках споруд в цілому або їх окремих конструктивних елементів, а також в різних галузях машинобудування – суднобудуванні, авіабудуванні, ракетобудуванні, моторобудуванні тощо. Зростаючі вимоги до якості проектних рішень, застосування нових конструктивних матеріалів викликає необхідність у розв’язуванні якісно нових задач. Зростає потреба в нових методах і підходах, пов’язаних з побудовою і дослідженням коректних комп’ютерних моделей, які адекватно відображають реальну поведінку конструкцій. Зростають вимоги до достовірності одержуваних комп’ютерних результатів. Ці чинники призводять до того, що істотно зростає обсяг оброблюваної інформації. Також великий обсяг інформації необхідно обробляти при виконанні розрахунків унікальних конструкцій. Тому виникає потреба у використанні високопродуктивних обчислень, що у свою чергу викликає зростання вимог до обчислювальних ресурсів. З іншого боку, на сучасному етапі розвитку обчислювальної техніки відбувається перехід від комп’ютерів з одним одноядерним процесором, можливості підвищення продуктивності яких вже вичерпано, до комп’ютерів з багатоядерними процесорами, яких теж може бути багато. Використання багатоядерних і багатопроцесорних комп’ютерних систем є необхідною передумовою створення технологій високопродуктивних обчислень. Але ефективність таких технологій визначається переважно відповідністю між програмноалгоритмічним забезпеченням і технічними засобами багатоядерного і багатопроцесорного комп’ютера. Адже найвищої продуктивності можна досягти, використовуючи алгоритми і програми з паралельною організацією обчислень. Тому виникає потреба як в адаптуванні для багатопроцесорних комп’ютерів раніше створеного програмного забезпечення комп’ютерів з одним одноядерним процесором, так і в створенні якісно нового програмного забезпечення, яке використовує паралельні алгоритми дослідження та розв’язування задач. Метою цієї роботи є розробка системи для визначення

міцності будівельних конструкцій, що за допомогою нейронної мережі, багатопроцесорних апаратів та алгоритмів буде визначати слабкі місця та можливі вдосконалення конструкцій. Так, для проведення розрахунків міцності конструкцій створено багато програмних засобів, як універсального призначення, так і орієнтованих на певну галузь (NASTRAN, MARC, ANSYS тощо).

Наприклад, для розрахунку міцності будівельних конструкцій розроблено вітчизняний програмний комплекс (ПК) ЛІРА [1]. Настроївши його інтерфейс для відповідної предметної області, ПК ЛІРА можна використовувати для розрахунків в інших галузях. На базі препроцесора і постпроцесора ПК ЛІРА та інтелектуальних паралельних програм бібліотеки Inparlib [2], [3] розроблено програмний комплекс, який реалізує технології високопродуктивних обчислень для дослідження і розв'язування задач розрахунку міцності будівельних конструкцій на комп'ютерах з паралельною організацією обчислень, зокрема, на інтелектуальних паралельних комп'ютерах Інпарком [4].

Основна ідея сучасного підходу – інтеграція в рамках єдиного програмного комплексу виконання наступних завдань:

- інтерактивна постановка задачі – створення моделі конструкції зі скінченними елементами;
- автоматичне формування даних дискретної задачі;
- автоматичне дослідження та розв'язування дискретної задачі;
- автоматична обробка результатів розв'язування дискретної задачі;
- інтерактивний аналіз та використання результатів розрахунку.

Паралельні обчислення варто використовувати для завдань, які виконуються автоматично, в першу чергу, для дослідження та розв'язку.

Математична постановка задачі

Теоретичною основою більшості програмних засобів для розрахунку міцності конструкцій є метод скінченних елементів (МСЕ), реалізований у формі зміщень. Вибір саме цієї форми пояснюється простотою її алгоритмізації і фізичної інтерпретації, наявністю єдиних методів побудови матриць жорсткості і векторів навантажень для різних типів скінченних елементів, можливістю врахування довільних граничних умов і складної геометрії конструкції, що розраховується [5].

Математично задачі розрахунку міцності конструкцій можуть бути поставлені у вигляді системи лінійних алгебраїчних рівнянь. Для розв'язування СЛАР на MIMD-комп'ютері можуть бути застосовані паралельні алгоритми для дослідження і розв'язування СЛАР з симетричними матрицями. Залежно від ширини і заповнення стрічки матриці використовуються різні види паралельних алгоритмів.

При розрахунку міцності конструкції важливо гарантувати достовірність отримуваних розв'язків. Коли йдеться про достовірність розв'язку, то мається на увазі, що досліджується достовірність наближених розв'язків математичних задач. Повна похибка наближеного розв'язку враховує вплив похибок вихідних даних, похибок дискретизації і формування даних відповідної дискретної задачі та похибок комп'ютерного розв'язку цієї дискретної задачі. Достовірність

розв’язку прикладних задач з наближеними вихідними даними гарантується використанням стійких до збурення вихідних даних математичних моделей конструкцій, що розраховуються.

Використання при дискретизації МСЕ теоретично обґрунтованих скінченних елементів, які задовольняють умовам збіжності і для яких отримано оцінки похибок розв’язку, є передумовою отримання достовірного наближеного розв’язку задачі. Для отримання достовірного розв’язку також важливо узгоджувати точність квадратурних формул, які використовуються при обчисленнях елементів і вектора навантажень, з точністю МСЕ, враховуючи при цьому поведінку підінтегральних функцій.

Висновки

В даній статті було розглянуто основні засади технологій високопродуктивних обчислень для визначення міцності будівельних конструкцій. Описано проблеми розв’язання задач розрахунку міцності будівельних конструкцій на паралельних багатоядерних та багатопроцесорних комп’ютерних системах та розглянуто основні засоби їх вирішення. Розглянуто основні програмні засоби для проведення розрахунків міцності конструкцій, такі як NASTRAN, MARC, ANSYS та вітчизняний аналог ПК ЛІРА. Сформована математична постановка задачі та можливості розрахунку точності та достовірності результатів.

Література:

1. [Електронний ресурс]. – Режим доступу : http://www.lira.com.ua.
2. Численное программное обеспечение интеллектуального MIMD-компьютера Инпарком / [Химич А.Н.,Молчанов И.Н., Мова В.И. и др.]. – Київ : Наукова думка, 2007. – 216 с.
3. Паралельні алгоритми розв’язання задач обчислювальної математики / [Хіміч О.М., Молчанов І.М., Попов О.В.]. – К. : Наукова думка, 2008. – 248 с.
4. [Електронний ресурс]. – Режим доступу : http://www.inparcom.com.
5. Баранов А.Ю., Я.Е. Слободян А.В. Попов А.Н. Химич Математическое моделирование прочности строительных конструкций на гибридных вычислительных системах // Международный научнотехнический журнал «Проблемы управления и информатики». – 2017. – № 4.– С. 68 – 81.1.

Мельник А. В., *студент*
Івано-Франківський національний технічний університет нафти й газу,
м. Івано-Франківськ
Кафедра автоматизації та комп’ютерно-інтегрованих технологій, студент

ПОРІВНЯННЯ ПІД І FUZZY РЕГУЛЯТОРІВ СИСТЕМИ АВТОМАТИЧНОГО КЕРУВАННЯ МІКРОКЛІМАТОМ ПРИМІЩЕННЯ

Застосування fuzzy-регуляторів для управління температурою або мікрокліматом в приміщенні широко висвітлювалось в зарубіжній науковій пресі. Одне з таких застосувань описано, наприклад, в роботі [1]. У таких

статтях часто наводиться порівняння результатів роботи розробленого авторами fuzzy-регулятора і традиційного ПІД-регулятора. При цьому fuzzy-регулятор, що включає всього 5 або 7 правил, які зв'язують одну вхідну і вихідну змінні, перевищує за якістю і точності регулювання результати, що отримані за допомогою традиційного ПІД-регулятора.

Автором була розроблена комп'ютерна модель системи автоматичного регулювання теплового режиму приміщення з fuzzy- і ПІД-регуляторами в системі Simulink пакету програм MATLAB. Блок-діаграма Simulink розробленої моделі показана на рис. 1.

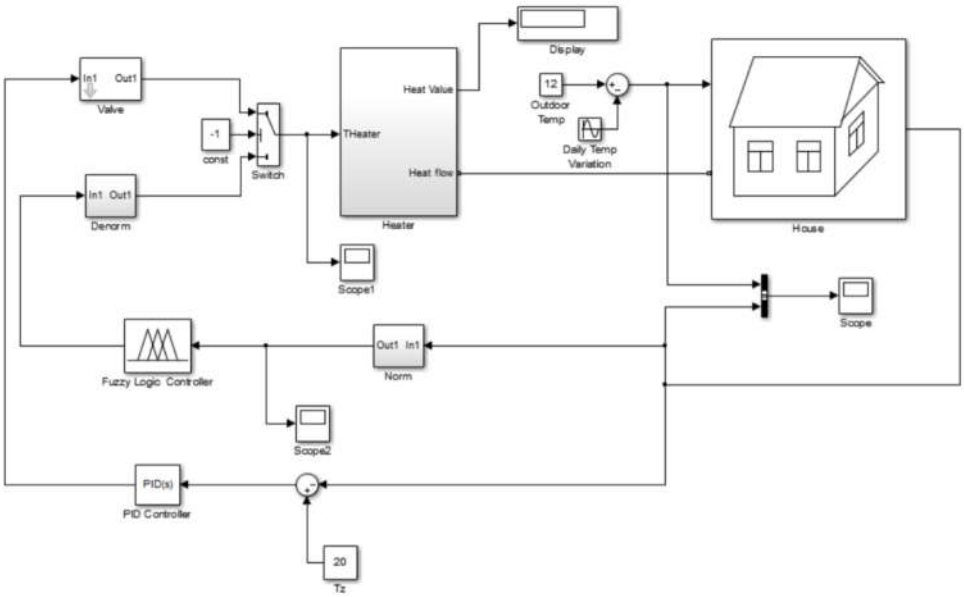


Рис.1. Блок-діаграма Simulink системи автоматичного регулювання теплового режиму приміщення з fuzzy- і ПІД-регулятором

У схемі на рис. 1 підсистема House містить блок-діаграму моделі теплового режиму будівлі, побудовану з використанням теплотехнічних компонентів Thermal бібліотеки Simscape. Основу моделі складають паралельні ланцюжки елементів, що представляють передачу тепла через огороджувальні конструкції (стіни, вікна, підлогу і стелю (дах)). Кожен ланцюжок включає в себе конвективний теплообмін між зовнішнім повітрям і огороджувальною поверхнею, передачу тепла теплопровідністю через огороджувальну поверхню і конвективний теплообмін між огороджувальної поверхнею і внутрішнім повітрям приміщення. Вихідним сигналом підсистеми є температура повітря всередині приміщення. Підсистема Heater моделює передачу тепла від нагрівального приладу до повітря всередині приміщення.

В якості одного з регуляторів використовується fuzzy-регулятор (блок Fuzzy Logic Controller). Даний fuzzy-регулятор використовує функції приналежності для вхідних і вихідних змінних і базу правил, подібні, описаним в роботі [1]. Нормування вхідного сигналу і денормування вихідного здійснюється в блоках Norm і Denorm відповідно. Альтернативним регулятором

є ПІД-регулятор (блок PID Controller). Завдання регулятора по температурі внутрішнього повітря дорівнює 20°C. Блок Switch дозволяє перемикати регулятори між собою.

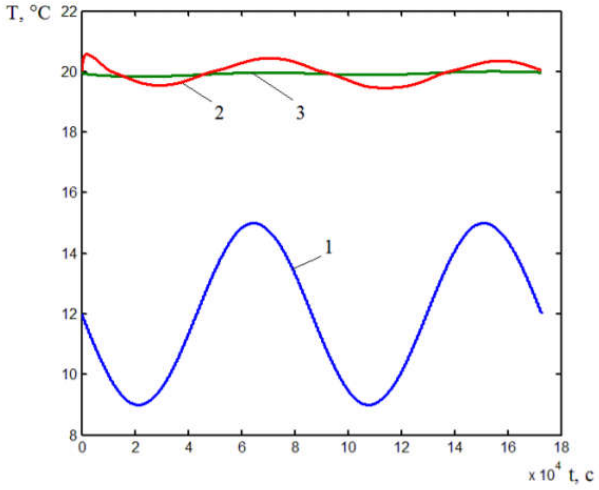


Рис. 2. Графіки зміни температури:
1 - зовнішнього повітря, 2 - всередині приміщення при fuzzy-регулюванні,
3 - всередині приміщення для ПІД-регулятора

Результати численних експериментів по моделі, що включають графіки зміни температури в приміщенні протягом сорока восьми годин (двох діб), показані на рис. 2. З рис. 2 видно, що точність підтримки температури всередині приміщення при fuzzy-регулюванні становить приблизно ±0.5°C, тоді як ПІД-регулятор дозволяє практично точно відпрацювати завдання. При зміні завдання, fuzzy-регулятор дає значно гіршу якість перехідних процесів ніж ПІД-регулятор, але кращу ніж для двохпозиційний (релейноий) регулятор.

Література:

1. Gouda M.M, Danaher S. Underwood C. P. Thermal comfort based fuzzy logic controller. Building Serv. Eng. 2001, 22(4), 237-253 pp

***Мельничук Володимир Ігорович**, студент*
Чернівецький національний університет імені Юрія Федьковича

ПРОБЛЕМИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ ПРИ МАШИННОМУ НАВЧАННІ

Машинне навчання є молодою областю комп’ютерних наук, яка пов’язана з пошуком існуючих прихованих закономірностей, що містяться в даних і завданнях, для яких не існує точного алгоритму рішення. Виникнення нових додатків методів машинного навчання відбувається постійно, при цьому відбувається зміна і самої області, в якій з’являються все нові методи, здатні повністю перевернути підходи у вирішенні завдань.

Більшість рекомендаційних систем передбачає застосування коллаборативної фільтрації (collaborative filtering) або контентної фільтрації (content-based filtering), чи в іншому випадку – застосування гібридних підходів [1]. Для коллаборативної фільтрації характерне вироблення рекомендацій, заснованих на моделі попередньої поведінки користувача [1, 2]. Таку модель можна побудувати, ґрунтуючись виключно на поведінці певного користувача або при обліку поведінки подібних користувачів, що мають подібні характеристики. У разі коллаборативної фільтрації, коли береться до уваги характер поведінки інших подібних користувачів, відбувається використання знань про групу (group knowledge) з метою вироблення певних рекомендацій на основі порівняння таких користувачів. В цілому рекомендації відбуваються в режимі автоматичної співпраці безлічі подібних користувачів з виділенням (метод фільтрації) таких користувачів, яким характерна демонстрація подібних переваг або шаблонів поведінки.

Контентній фільтрації характерне формування рекомендацій, що ґрунтуються на поведінці користувача [1, 3]. Наприклад, при цьому підході може бути використана ретроспективна інформація про перегляди (перелік блогів, які читаються користувачем і характеристика таких блогів). Сам контент можна визначити, використовуючи ручний режим або витягти автоматично за допомогою інших методів подібності.

Гібридним підходам, яким характерне поєднання коллаборативної і контентної фільтрації, властиво підвищення ефективності (і складності) рекомендаційних систем [1]. Об’єднання результатів коллаборативної і контентної фільтрації потенційно дозволяє підвищити точність рекомендації. Крім того, гібридний підхід може бути корисний, якщо застосування коллаборативної фільтрації починається при значній розрідженості даних. Гібридний підхід дозволяє спочатку зважувати результати згідно контентної фільтрації, а потім зміщувати ці ваги у напрямку до коллаборативної фільтрації (в міру «визрівання» доступного набору даних по конкретному користувачеві). Слід зазначити, що в рекомендаційних механізмах можуть бути використані самі різні алгоритми, багато з яких прийшли з області машинного навчання (яка, в свою чергу, являє собою підобласть штучного інтелекту), яка займається алгоритмами для навчання, прогнозування і прийняття рішень [3].

Рекомендаційним системам нового покоління характерне те, що інформація про користувача, яка використовується як навчальна інформація, як правило, витягується з усіх доступних джерел, в яких так чи інакше відбиваються, так звані, «сліди» його діяльності. В якості таких джерел зазвичай виступають соціальні мережі (Facebook, LinkedIn, Twitter тощо), історія відвіданих Web-сторінок браузера, історія покупок тощо. Крім того, в процес побудови профілю користувача часто втягується додаткова інформація, обсяги якої в сотні і тисячі разів перевершують початковий обсяг даних про нього. У вигляді джерел такої інформації зазвичай виступають хмарні ресурси і глобальні онтології, такі як WordNet, Wikipedia, DBpedia і ресурси Linked Data Web. Ці дані, як правило, є гетерогенними, мають різний ступінь достовірності і повноти, містять дані на природних мовах і іншу неструктуровану інформацію.

Слід зауважити, що в рекомендаційних системах третього покоління акцент робиться на семантичних моделях уявлення і використання всіх компонент знань про користувача і про продукти, які йому можуть бути запропоновані [3, 4]. Рекомендаційні системи третього покоління повинні виробляти рішення на основі семантичних моделей інтересів і переваг користувача, брати до уваги мотивацію і причини, які спонукають конкретного користувача робити той чи інший вибір. Такі системи повинні давати мотивовані пояснення з прийнятими рішеннями. Вони повинні враховувати психологічні та інші фактори, пов'язані до конкретних користувачів [2-4]. Тому технологія розробки рекомендаційних систем третього покоління акцентує увагу на побудові і використанні персоніфікованого профілю користувача рекомендаційної системи. Важливими в них є також і економічні чинники, пов'язані з практичним використанням рекомендаційних систем.

До проблем рекомендаційних систем при машинному навчанні слід віднести наступні. Так, за допомогою існуючих можливостей збору даних, які сьогодні надає Інтернет, відбулося суттєве спрощення використання «мудрості натовпу» на основі коллаборативної фільтрації. Але, при цьому, наявність величезної кількості доступних даних призводить до ускладнення реалізації такої можливості. Наприклад, моделювання поведінки багатьох користувачів цілком можливе, проте іншим користувачам характерна відсутність демонстрації типової поведінки. Тобто такі користувачі призводять до зміщення результатів, призначених для рекомендаційної системи з подальшим зниженням її ефективності при машинному навчанні. Крім того, ефективність знижується, коли окремі користувачі задіють рекомендаційну систему з метою підвищення переваг якогось одного продукту по відношенню до іншого продукту, наприклад, при відправці позитивного відгуку на один продукт і негативного відгуку про інші продукти. Для добре сформованої рекомендаційної системи це не буде проблемою, однак, з кожним днем, така система повинна ставати все більш точною і ефективною, тобто вдосконалюватися.

Іншою проблемою, яка властива великим рекомендаційним системам, є масштабованість. Традиційним алгоритмам характерна гарна робота з порівняно невеликими обсягами даних, однак з ростом цих наборів, отримання результатів на колишньому рівні якості за допомогою традиційних алгоритмів стає проблематичним. При оффлайновій обробці – це може не становити великої проблеми, проте для сценаріїв реального часу необхідні більш спеціалізовані підходи для машинного навчання.

Крім того, виникнення певних проблем в рекомендаційних системах вимагає дотримання конфіденційності. Рекомендаційним алгоритмам властиво розпізнавання таких закономірностей, про існування яких люди можуть навіть не підозрювати, що стає загрозою для конфіденційності.

На сьогоднішній день рекомендаційні системи активно застосовуються в найбільш популярних Web-сайтах, що мають різну соціальну і комерційну спрямованість. Безперечно, рекомендаційні системи приносять величезну користь власникам цих сайтів і їх користувачам, однак їх використанню можуть бути характерні і зворотні сторони. Тому конфіденційність рекомендаційних

систем повинна постійно забезпечуватися, що необхідно контролювати при машинному навчанні.

Вирішення існуючих проблем рекомендаційних систем можна досягти комплексним підходом при здійсненні машинного навчання, тобто при навчанні використовувати гібридний підхід: для одного випадку – віддавати перевагу коллаборативній фільтрації, а в іншому – контентній фільтрації за умови використання їх обох, але за відповідним призначенням.

Література:

1. Melville P. Content-Boosted Collaborative Filtering for Improved Recommendations / Melville P.,Mooney R., Nagarajan R. // National Conference on Artificial Intelligence : «AAAI-2002», 20-25 July 2016, Edmonton, Canada : materials. Edmonton, Canada : AAAI, 2002. P. 187- 192.

2. Han F., Liu H. Transition matrix estimation in high dimensional time series // Proceedings of the 30th International Conference on Machine Learning. USA. 2013. Vol. 28. pp. 172-180.

3. Tuzhilin A. Keynote presentation at International Conference on Data Mining (ICDM 2013) Dallas, Texas, December, 2012.

4. Witten I.H., Frank E., Hall M.A. Data Mining: Practical machine learning tools and techniques (3rd Edition). San Francisco. California. Morgan Kaufmann. 2011. 145 p.

Неруш Г.В., *студент*
Науковий керівник: ***Попенко В.Д.,*** *доцент*
Київський Політехнічний Інститут ім. Ігоря Сікорського, м. Київ
Кафедра автоматизованих систем обробки інформації та управління (АСКОІУ), студент

ВИКОРИСТАННЯ МОВИ ARCHIMATE ДЛЯ ПРОЕКТУВАННЯ АРХІТЕКТУРИ ПІДПРИЄМСТВА

В умовах активного розвитку та високої конкуренції підприємствам доводиться постійно вносити зміни в бізнес-моделі і удосконалювати свої бізнес-стратегії.

Інформаційні технології давно стали невід'ємною частиною українських компаній. Будь-які проблеми в IT боляче б'ють по бізнесу.[1] А з урахуванням того, що IT-проекти стають все складніше, а таких проектів все більше, потрібен додатковий рівень, на якому буде сплановано функціонування сфери IT в підприємстві

Без цього можна наламати дров і безцільно витратити час і ресурси. А в кризові часи це може бути фатально для компанії. Потрібні бачення і детальний план розвитку компанії, хоча б у частині IT. [2, 5]

Потрібна відповідь на питання «Як робити?»

Це - рівень Архітектури Підприємства.

Для досягнення узгодженості бізнесу та IT потрібен інтегрований підхід до всіх аспектів підприємства. Організаційна ефективність не отримується за допомогою локальних оптимізацій, але реалізується шляхом добре організованої взаємодії компонентів архітектури організації. При моделюванні корпоративної архітектури нам потрібна мова, яка знає про концепції корпоративної архітектури, і така мова є **- ArchiMate.** [6,7]

Мова **ArchiMate** містить 3 типи елементів:

- Елементи, які діють (активні елементи)
- Елементи, що відображають поведінку тих елементів, які діють (поведінкові елементи)
- Елементи, які не можуть діяти і на яких діє така поведінка (пасивні елементи)

Три типи елементів, пов'язаних відношеннями, можуть утворювати речення різного роду.

Структура граматики ArchiMate частково грунтується на шаблоні «суб'єкт-дієслово-об'єкт» природної мови. ArchiMate визначає конкретні концепції, специфічні для певного рівня архітектури.

У цьому контексті ми розрізняємо три основних рівні:

Рівень бізнесу - пропонує продукти та послуги зовнішнім замовникам, які реалізуються в організації бізнес-процесами (виконується бізнес-акторами або ролями).

Рівень застосувань - підтримує бізнес-рівень за допомогою доданків, які реалізуються компонентами програмного забезпечення.

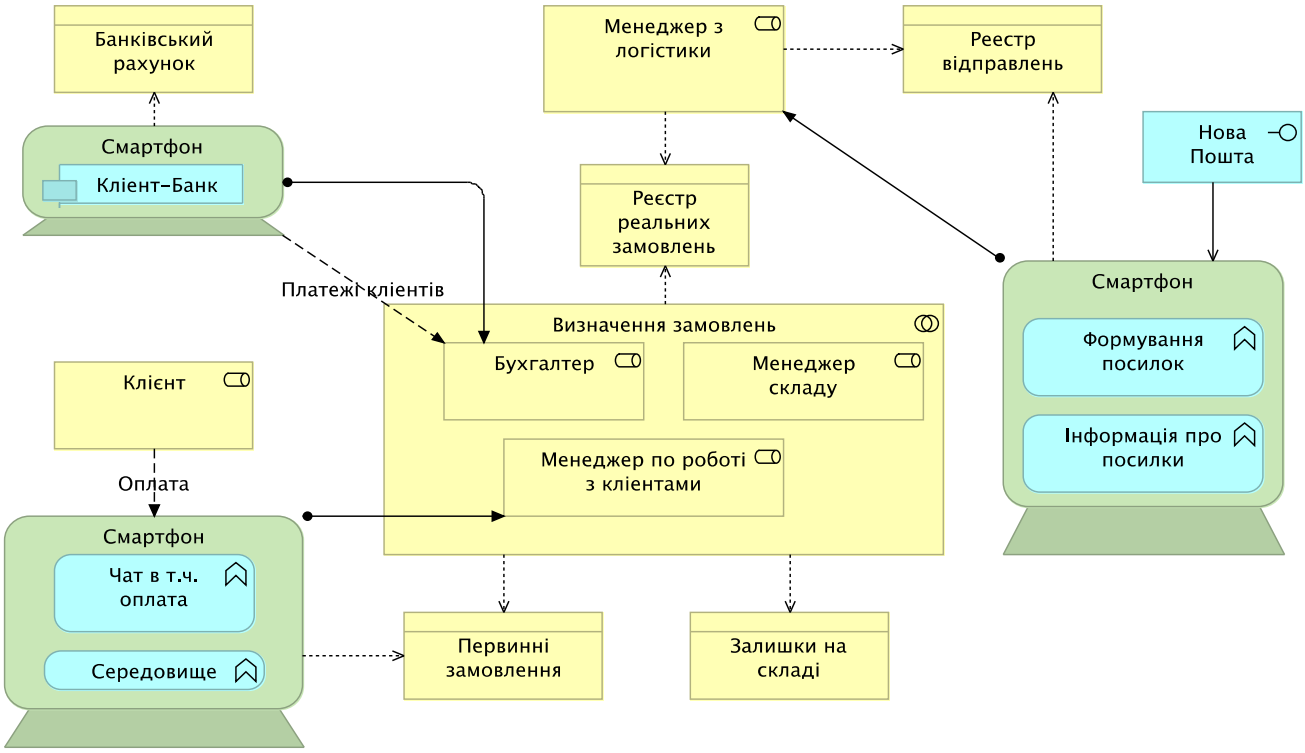
Технологічний рівень - пропонує інфраструктурні послуги (напр., обробка, зберігання та передача інформації), необхідні для запуску програм, виконуваних комп'ютерами, пристроями зв'язку та системним ПЗ.

Для того, щоб провести аналіз використання Archimate, розглянемо моделі підприємства, основна діяльність якого є: продаж товарів.

Перед цим підприємством стала проблема автоматизації основних бізнес-процесів, для оптимізації часових та фінансових витрат.

Була розроблена модель бізнесу “As-Is” (Як Є зараз). На ній розглянута робота на бізнес рівні, процесу продажу товарів. *Клієнт* починає взаємодію зі *Смартфоном* та передає інформацію *Менеджеру по роботі з клієнтами*, що включає в себе інформацію про оплату товару та побажання по замовленню. *Менеджер по роботі з клієнтами* утворює запит *Бухгалтеру* на перевірку оплати *Клієнтом*. *Бухгалтер* взаємодіє зі *Смартфоном* для доступу до *Банківського рахунку* потім передає цю інформацію *Менеджеру по роботі з клієнтами*. *Менеджер складу* бере оброблену інформацію та формує *Реєстр*

реальних замовлень потім передає його *Менеджеру з логістики*. *Менеджер з логістики* формує *Реєстр* відправлень і через *Смартфон Формує посилки* та отримує *Інформацію про посилки*. Оброблена інформація відправляється до *Нової пошти*.



Мал. 1 (Модель “As-Is” Як Є Зараз)

Для цього були взяти наступні елементи:

Бізнес Роль - визначається як іменоване специфічну поведінку бізнес-суб'єкта, який бере участь в даному контексті

Девайс - є фізичним IT-ресурсом, на якому системне програмне забезпечення та артефакти можуть зберігатися або розгортатися для виконання.

Функції програми - являє собою автоматизовану поведінку, яку може виконувати компонент програми.

Бізнес об’єкт - визначається як одиниця інформації, яка є актуальною з точки зору бізнесу.

Бізнес колаборація - визначається як конфігурація (можлива, тимчасова) двох або більше бізнес ролей, що призводить до специфічної колективної поведінки в конкретному контексті.

Інтерфейс програми - являє собою точку доступу, де Служби прикладних програм стають доступними для користувача, іншого компонента програми або вузла.

Використані наступні зв’язки:

Мал. 2 (Зв'язок доступу)

Зв’язок доступу - моделює доступ поведінкових концепцій до Бізнес або Об’єктів даних.

Мал. 3 (Зв'язок потік)

Зв'язок Потік - описує обмін або передачу інформації або значення між процесами, функціями, взаємодіями та подіями.

Мал. 4 (Зв'язок присвоєння)

Зв’язок присвоєння - пов'язує активні елементи (наприклад, бізнес-ролі або компоненти програми) з одиницями поведінки, які виконуються ними, або бізнес-акторами з діловими ролями, які виконуються ними.

Мал. 5 (Зв'язок обслуговування)

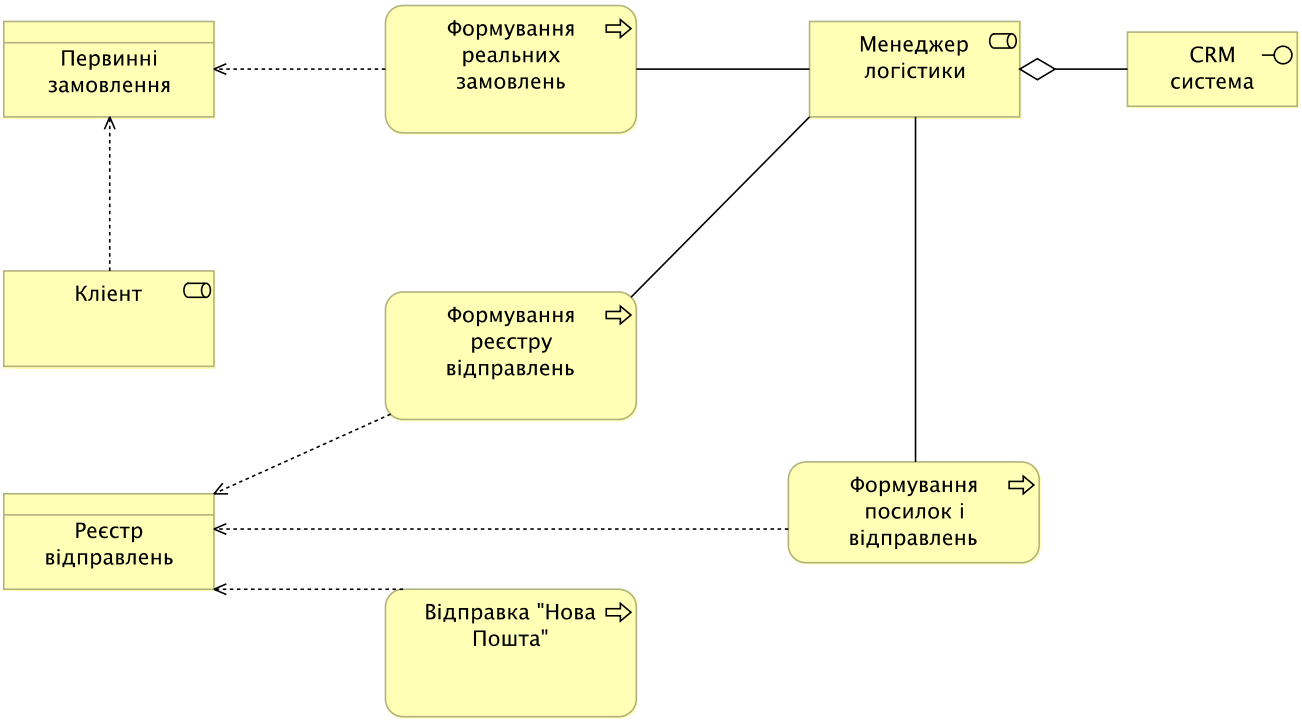
Зв’язок обслуговування - моделює які елемент забезпечує свою функціональність іншому елементу. [4]

Після аналізу даної моделі були виявлені її недоліки:

- Людський фактор у роботі
- Відсутність єдиної Системи для зберігання та обробки даних

Тому було прийнято рішення розглянути питання впровадження CRM системи та змоделювати роботу підприємства на бізнес рівні з новою Інформаційною Системою. *Менеджер з логістики* через *CRM систему* формує *Реальні замовляння* використовуючи інформацію *Первинних замовлень* які поступають від *Клієнта* в обробленому виді завдяки *CRM системі.*

Потім *Менеджер з логістики* через *CRM систему* виконує процеси: *Формування реєстру відправлень; Формування посилок і відправлень*. Вони утворюють *Реєстр відправлень* і інформація надсилається до *Нової Пошти.*



Мал. 6 (Модель «То-Ве» Як Буде)

Після аналізу нової моделі, можемо прийти к висновку що бізнес процесів стало менше. Велику долю роботи взяла на себе CRM система. Бізнес Ролі мають менше зв'язків. Як наслідок маємо наступні покращення:

- **Зменшення часо-витрат**

Всі данні знаходяться всередині єдиної системи, кожен співробітник має швидкий доступ до них.

- **Зниження фінансових витрат**

Частково усунено людський фактор. Обробка даних проходить автоматично без участі людини, що виключає шанс на помилку.

- **Підвищення мотивації співробітників**

Створено комфортні умови для роботи співробітників. Усунена проблема використання різноманітних засобів та інструментів для виконання одного єдиного процесу.

Таким чином, побудова архітектури підприємства за допомогою мови моделювання ArchiMate дозволяє систематизувати і формалізувати знання з урахуванням специфіки роботи, забезпечити управління змінами виробничих параметрів і врахувати потреби зацікавлених сторін. Це сприяє передачі досвіду, збереженню знань, підвищення ефективності роботи персоналу та управлінням людськими ресурсами.

Розроблена модель може бути застосована до більшості діючих підприємств які займаються товарним бізнесом, для автоматизації бізнес-процесів а як наслідок отримати конкурентоспроможне підприємство яке має всі шанси зайняти топові позиції в умовах сучасного ринку.

Як висновок можна привести Плюси и Мінуси мови Archimate

- **Плюси**

• Відкрита і безкоштовна
• Підтримується інструментами моделювання
• Узгоджений для моделювання на міжнародному рівні
• Охоплює багато предметних областей
• Розширюваний за допомогою View Point

- **Мінуси**

• Не дозволяє малювати BPMN, ERM.
• Мало кольорів для малювання моделей
• Не має достатнього поширення в Україні та бувших країн СНД.

Література:

1. Архитектура Предприятия. Как заставить ИТ работать на вашу компанию? / Коротков Андрей, 2013.
2. Элементы бизнес-слоя. Моделирование бизнес-архитектуры / Рубенчик А.В. 2016.
3. Исследование влияния изменения бизнес-модели на бизнес-архитектуру для ИТ компании / Я. В. Каблуков, П. М. Пашков / Инновации в жизнь. 2015.
4. ArchiMate® 3.0.1 Specification / The Open Group 2012-2017.
5. Маркетинг Инновации / Молчанов Н.Н. 2014.

6. ArchiMate® 2.0 – Understanding the Basics. A White Paper by: Gerben Wierda / February 2013 - Marc Lankhorst, Hans van Drunen. Enterprise Architecture. Режим електронного доступу: https://publications.opengroup.org/w130
7. Development and Modelling. Combining TOGAF and ArchiMate / 2007. Режим електронного доступу: http://api.ning.com/files/1vDSdWV6JAgpNXvLL9hU3l21jWMxztv9AzOoqrfwJLi-U79-UyYIIu1ll34yynga44xabZUmLaOdAiTUJx1I1K8btmziOt3N/Lankhorst.pdf

Панченко Б.В. *студент*
Одеський Національний Морський Університет, м. Одеса
Технічна кібернетика й інформаційні технології ім.
професора Р.В. Меркта, студент

ВИКОРИСТАННЯ ЕВОЛЮЦІЙНОГО ПІДХОДУ ДЛЯ ЗАДАЧ СКЛАДАННЯ РОЗКЛАДІВ

На сьогоднішній день більшість автоматизацій людської праці являются штучні нейронні мережі (далі ШНМ), їх використовують майже у всіх сферах людини, й цей темп лише набирає обертів.

Проте ШНМ сама по собі ні на нащо не здатна. Правильність та точність ШНМ напряму залежить від її навчання, де одним з основним елементом є дані розглядаємої системи, а інший - метод виявлення похибки результату розрахунку ШНМ. Зазвичай використовують градієнтний алгоритм зворотного поширення похибки, з деякою кількістю епох (кількість повторень).

Головним недоліком градієнтних методів є попадання в локальний мінімум. Є декілька рішень проблеми, в контексті навчання ШНМ, зміна або збільшення кількості даних, зміна структури (моделі) самой ШНМ. Проте можливо спробувати змінити сам алгоритм, наприклад на еволюційний, типовим представником является генетичний алгоритм (ГА). В них відсутня проблема з локальним мінімумом, проте вони мають кілька недоліків:

1. можлива різна точність навчання (при повторних навчань);
2. ГА потребує більше обчислювальних потужностей та часу (не завжди);
3. створення моделі ГА з існуючої моделі ШНМ.

Основною проблемою використання ГА полягає в відсутності бездоганного методу трансформації нейроної моделі в модель генетичних алгоритмів.

Головними складовими генетичного алгоритму є фенотип, яка складається з хромосом, які мають параметри характеристик фенотипу - геноми[1]. В математичній моделі фенотип можно розглядати як матрицю розміром кількість хромосом на кількість генотипів в хромосомі, при умові що хромосоми мають однаковий розмір.

Класична штучна нейронна мережа складається з шарів нейронів. Якщо у нас хромосоми можуть мати різний розмір то кожна хромосома відповідає кожному шару нейроній мережі. Якщо ні то можна представити всі нейрони як одну хромосому. Проте виграшу від такого підходу не буде[2].

Інший варіант, представити ген як шар нейронної мережі, інший ген як біаси слоя, ці два гена будут належати одній хромосомі, в кінцевому вигляді ми будемо мати фенотип з кількістю хромосом рівним кількістю слої, де ген це шар.

GENITOR (Рисунок 1). Існує кілька версій алгоритму GENITOR, базовий кодує ваги заданої (багаторівневої) топології в бітових рядках. Малюнок 2.2 ілюструє це. Біт індексу вказує, чи існує з'єднання взагалі, і біти кодування ваги є двійковим поданням значення ваги. Уитли повідомляє про 8-бітної кодуванні, «в діапазоні від -127 до +127, де 0 зустрічається двічі»[2].

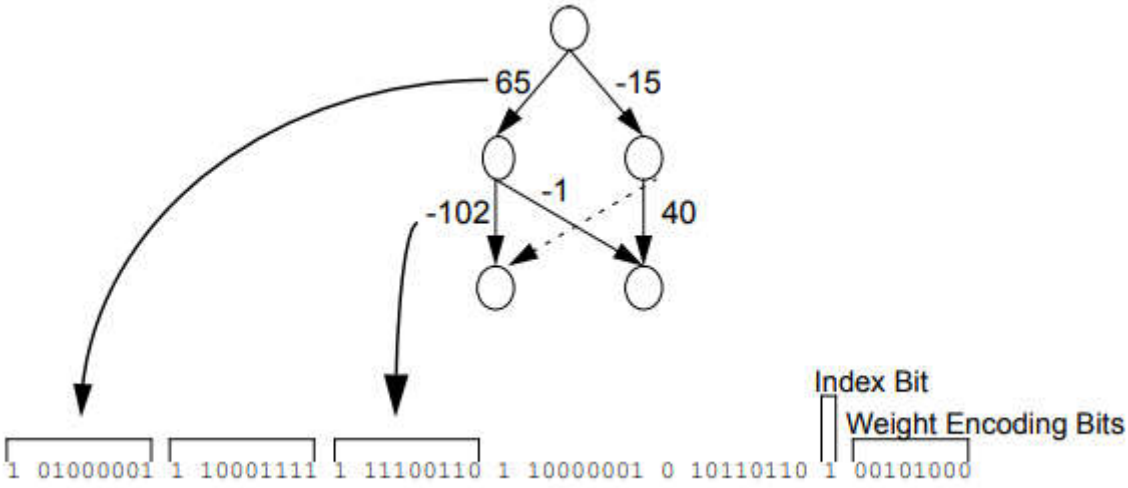


Рисунок 1 – Кодування двійкової маси згідно GENITOR.

Література:

1. David J. Montana. Training Feedforward Neural Networks Using Genetic Algorithms: https://www.ijcai.org/Proceedings/89-1/Papers/122.pdf.
2. Philipp Koehn. Combining Genetic Algorithmsand Neural Networks: The Encoding Problem. Електронний ресурс – Режим доступу : http://homepages.inf.ed.ac.uk/pkoehn/publications/gann94.pdf.

Рогоза А.В.
Національний технічний університет України
“Київський Політехнічний Інститут ім. Ігоря Сікорського”, м. Київ
Кафедра автоматизованих систем обробки інформації та управління,
студент

ШВИДКІ АЛГОРИТМИ ОБЧИСЛЮВАЛЬНИХ КРИПТОПРИМІТИВІВ

Вступна частина

У сучасному інформаційному суспільстві велику загрозу конфіденційності та цілісності інформації представляє кіберзлочинність. Зростання кількості кібератак та доступність програмно-технічних засобів для їх реалізації зумовлює

необхідність розробки сучасних засобів інформаційної безпеки громадян та держави в цілому.

Отже, актуальною задачею є створення та вдосконалення систем захисту інформації, зокрема алгоритмів криптографічного захисту, що в більшості випадків є базовим ядром таких систем. На даний час основним питанням, що підлягає вирішенню, є збільшення об'ємів інформації, що може оброблятися функціями криптографічного перетворення. Одним з підходів вирішення даної проблеми є застосування швидких алгоритмів обчислення криптопримітивів.

Постановка задачі

В основу покладені алгоритми виконання різних операцій над багаторозрядними числами. Для простоти викладення припустимо, що ми маємо справу з цілими числами. Розглянуті алгоритми можна легко розповсюдити на випадок чисел з плаваючою комою тощо.

Під n розрядним цілим числом будемо розуміти довільне ціле число, менше за b^n , де b — основа прийнятої позиційної системи, в якій ми представляємо числа; такі числа в цій системі записуються з використанням не більше ніж n розрядів.

Ці числа можна розглядати як числа, записані в системі числення за основою b, де b — розмір машинного слова. Наприклад, ціле число, яке займає 10 машинних слів в пам'яті ЕОМ, розмір слова якої дорівнює $b = 10^{10}$, має 100 десяткових (цифр; але ми будемо розглядати його як десятирозрядне число за основою 10^{10}. Це робиться шляхом групування бітів [1].

Далі багаторозрядні цілі числа будемо називати ще багатословними або s словними, оскільки кожне з них можна представити у вигляді масиву 16- бітових, 32-бітових або 64- бітових слів.

Алгоритми швидкого множення чисел

Як було зазначено у вступній частині, характерною особливістю розв'язання багатьох задач апроксимації функцій, моделювання фізичних, хімічних (біохімічних) процесів, аеродинаміки, гідродинаміки, захисту інформації є використання обчислень з багатократною точністю або над багаторозрядними числами. Це обумовлює актуальність створення ефективних алгоритмів виконання операцій над багаторозрядними числами для програмної реалізації на універсальних ЕОМ і для спеціалізованних апаратних та програмно-апаратних комплексів.

Алгоритми обчислення степеня за модулем виконуються операції піднесення до квадрату та множення за модулем на кожному кроці піднесення до степеня. Для обчислення $x^n \bmod M$ треба кожний раз застосовувати, програми множення за модулем [2].

Розглянемо алгоритм обчислення

$$R = a \cdot b \bmod M, \qquad (1)$$

де a, b, M – k-розрядні числа. Оскільки k часто більше ніж 256, то необхідно так будувати структури даних, щоб оперувати з дуже великими числами. Припускаючи, що довжина машинного слова дорівнює ω (як правило ω = 16 або 32), розіб'ємо k-бітове число на s слів так, що $(s-1)\omega < k \le s \cdot \omega$.

Проміжні результати можуть займати більше ніж *s* слів, і їх також треба зберегти.

Стандартний алгоритм

Нехай a і b два s-слівні числа, записані за основою *w*

$$a = (a_{s-1}a_{s-2}\ldots a_0) = \sum_{i=0}^{s-1} a_i w^i \quad (2)$$

$$b = (b_{s-1}b_{s-2}\ldots b_0) = \sum_{i=0}^{s-1} b_i w^i \quad (3)$$

де *a* і *b* – цифри з проміжку [0; *w*-1]. Взагалі *w* може бути будь-яким цілим числом.

Для використання на комп'ютері часто вибирається $w = 2^{\omega}$, ω – довжина машинного слова, наприклад, ω = 32. Стандартний алгоритм добутку *a* і *b* знаходить часткові добутки, підсумовує їх і отримує кінцевий 2s-слівний результат *t* [3].

Стандартний алгоритм по суті цифра за цифрою виконує описане множення і додавання. Для економії пам'яті використовується тільки одна змінна *t* для часткового добутку. Її початкове значення дорівнює 0; потім береться цифра множника *b,* множиться на *a* і підсумовується; для часткового добутку *t*. Наприкінці обчислень ця змінна для часткових добутків містить остаточний добуток $a \cdot b$.

Стандартний алгоритм обчислення $a \cdot b$ має наступний вигляд:

На вході: a, b

На виході: $t = a \cdot b$

Крок 0. Спочатку $t_i = 0$ для всіх $i = \overline{0,2s-1}$

Крок 1. Для $i = \overline{0,s-1}$

Крок 2. $C = 0$

Крок 3. Для $j = \overline{0,s-1}$

Крок 4. $S = t_{i+j} + a_j b_j + C$

Крок 5. $t_{i+j} = S$

Крок 6. $t_{i+s} = C$

Крок 7. Результат $(t_{2s-1}, t_{2s-2}, \ldots, t_0)$

Операція внутрішнього добутку, яку описано вище, потребує множення двох *w*-бітових чисел, потім додавання отриманого результату з *w*- бітовим переносом результату попереднього внутрішнього добутку і нарешті, додавання отриманого результату до поточного слова t_{i+j} часткового добутку. В результаті цих трьох дій отримуємо 2*w*-бітове число.

Оскільки крок внутрішнього добутку знаходиться в середині самого внутрішнього циклу, його слід виконувати як можна швидше. Аналіз крюків цього алгоритму показує, що загальне число кроків внутрішнього добутку дорівнює s^2. Так як s=k/ω, ω – постійне для конкретного комп'ютера, то стандартний алгоритм множення потребує $O(k^2)$ бітовий операцій для множення двох *k*-бітових чисел [4]. Цей алгоритм асимптотично повільніший, ніж алгоритм Карацуби та алгоритм, який використовує швидке перетворення Фур'є. Однак він простіший для застосування та має кращу швидкодію для малих чисел, ніж ці асимптотично більш швидкі алгоритми.

Висновок

В даній статті було розгянуто завдання та алгоритми швидкого обчислювання криптопримітивів. Було описано два підходи до множення багаторозрядних чисел. Було показано, що підхід множення чисел з використанням алгоритму швидкого перетворення Фур'є працює швидше стандартного підходу множення двох чисел, для багато розрядних чисел.

Завдяки описаним вище підходам отримуємо резерв для оптимізації алгоритмів криптографії, що базуються в основному на операціях додавання та множення, що і було продемонстровано на основі алгоритму описаного в статті.

Література:

1. Задірака В.К., Олексюк О.С. Комп'ютерна арифметика багаторозрядних чисел. //Київ - 2003
2. Карацуба А.А., Офман Ю.П. Умножение многорозрядных чисел на автоматах //ДАН СССР. — 1962. т.145. — С. 293-294.
3. Шенхаге А., Штрассен В. Быстрое умножение больших чисел // Кибернет. сб. — 1973. — вып. 10. — С. 87-98.
4. Cook S. A., Aanderaa S. O. On the minimum computation time of functions, Thesis, Harvard University, 1966. — P. 26-50.

Родіонов П.Ю., *к.е.н., ст. викл.*
Бишовець Н.М., *студент*
НТУУ «КПІ» імені Ігоря Сікорського, м Київ
Кафедра автоматизованих систем обробки інформації

ЗАСТОСУВАННЯ АФІННИХ ПЕРЕТВОРЕНЬ В КОМП'ЮТЕРНІЙ ГРАФІЦІ

Афінні перетворення широко застосовуються в машинній та комп'ютерній графіці. Найбільшого поширення, тим не менш, набули лише їхні три часткові випадки: зсув, поворот та масштабування.

Мета даної роботи полягає у необхідності розглянути можливості застосування афінних перетворень та визначити їх роль в комп'ютерній графіці.

Нехай у площині задана початкова система координат 0XY і деяка нова система координат $0_1X_1Y_1$. Тоді перетворення, які полягають у тому, щоб у відповідність точці P площини ставиться точка P_1, яка в новій системі має такі самі координати, що й точка P у початковій, називаються афінними (тобто, під перетворенням розуміють функцію, яка приймає точку або вектор і відображає її на іншу точку або вектор). Для точок перетворення формально записується у вигляді:

$$Q = T(P);$$

а для векторів – у вигляді:

$$v = R(u).$$

У чотиривимірному просторі ми працюємо з представленнями точок і векторів. Лінійне перетворення, яке трансформує представлення точки (або вектору) в інше представлення, завжди може бути записано в термінах двох представлень u і v за допомогою множення матриць:

$$v = Au,$$

де A - квадратна матриця.

Отже, лінійне перетворення можна розглядати з двох точок зору: як зміна подання або фрейму, що призводить до нового подання вершин, або як перетворення вершин в рамках того ж фрейму. Основні властивості афінних перетворень такі:

– множина точок, яка в початковій системі координат задовольняє деяке рівняння, переходить у множину точок, координати яких у новій системі задовольняють таке саме рівняння.

– відношення площ і об'ємів геометричних фігур зберігається;

– зберігається просте співвідношення трьох точок;

– існує таке єдине перетворення площини, що переводить трійку точок, які не належать одній прямій, у нову трійку точок так, що вони також не належать прямій;

– якщо початкова та нова системи координат є декартовими з однаковими одиничними відрізками по осях, то при перетвореннях зберігаються всі метричні властивості геометричних фігур.

Зсув, або плоскопаралельне зміщення (translation), - це операція, яка зміщує точки на фіксовану відстань уздовж заданого напрямку. Зрушення задається тільки вектором зміщення d, оскільки для всіх точок P на об'єкті справедливо співвідношення:

$$P' = P + d.$$

Необхідно звернути увагу на те, що таке визначення операції ніяк не пов'язане з системою відліку або фреймом подання. Зрушення має 3 ступені свободи, оскільки можна довільно задати три компоненти вектору зміщення. Задати поворот (rotation) дещо складніше, ніж зрушення, так як потрібно специфікувати більше параметрів. Існує три властивості цього перетворення, які зберігаються і в інших варіантах операції повороту:

– існує деяка точка, що називається фіксованою точкою перетворення, яка залишається нерухомою при застосуванні до неї цього перетворення;

– враховуючи, що двовимірний простір є окремим випадком тривимірного, можна поширити зроблені висновки і на три виміри. У правосторонній системі координат, позитивний напрямок осі z тривимірної системи координат буде направлено на спостерігача;

– двовимірне обертання на площині еквівалентно тривимірному обертанню навколо осі z. Всі точки на площині, паралельній xy, повертаються однаково, зберігаючи значення компонента z.

Поворот і зсув відносяться до групи так званих ізометричних перетворень або перетворень твердого тіла. Комбінація цих перетворень не може змінити

форми об'єкта, а змінює тільки його положення в просторі – позицію і орієнтацію. Отже, самі по собі повороти і зміщення не дають повного набору афінних перетворень.

Масштабування (scaling) являє собою анізометричне афінне перетворення. Масштабування збільшує або зменшує розміри об’єкта. Масштабуванням об’єктів називається розтягнення об’єктів вздовж відповідних осей координат відносно початку координат. Ця операція застосовується до кожної точки об’єкта, тому можна також говорити про масштабування точки. При цьому, звісно, мова не іде про зміну розмірів самої точки. Масштабування досягається множенням координат точок на деякі константи. В тому випадку, коли ці константи однакові, масштабування називають однорідним.

Таким чином, афінні перетворення у поєднанні з постійним розвитком можливостей обчислювальної техніки є невичерпним джерелом істотних просувань на шляху розвитку комп’ютерної графіки, її використання у наукових та інших дослідженнях.

Література:

1. https://web.posibnyky.vntu.edu.ua/fitki/8romanyuk_komp_grafika/zmg1/zmg/21.htm
2. http://stud24.ru/mathematic/afnn-peretvorennya-fgur/21364-60134-page5.html
3. http://www.berkut.mk.ua/download/pdf/cg_lab_3.pdf
4. https://studopedia.org/12-48541.html

Рубан Ю.І.

НТУУ «КПІ імені Ігоря Сікорського», Україна

Студент кафедри біомедичної кібернетики

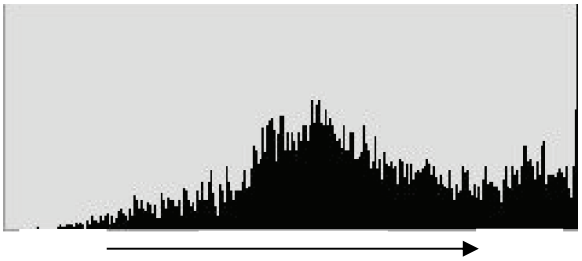
ВИЗНАЧЕННЯ СТУПЕНЮ ОПІКІВ ШКІРИ МЕТОДАМИ ЦИФРОВОЇ ОБРОБКИ ЗОБРАЖЕНЬ

Опіки шкіри є поширеним типом травматизму. Але немає чітких правил і вказівок щодо того, як виміряти ступінь опіків шкіри. У цій роботі запропоновано новий кількісний і простий метод для визначення ступінь опіків шкіри, використовуючи комп'ютерну технологію обробки зображень на основі розміру, форми і глибина опіків шкіри. Це має практичне значення для лікування шкірних опіків пацієнтів.

Технологія обробки комп'ютерних зображень є інтегрованою дисципліною. Швидкий розвиток сучасних цифрових технологій і мікроелектроніки надають сучасні технічні засоби для обробки комп'ютерних зображень. Обробка комп'ютерних зображень - це перетворення зображення в цифрову матрицю, яка розглядається з певним алгоритмом в комп'ютері [1].

Використовуючи мову програмування C++, було проведено першу процедуру обробки зображень опіків шкіри, що базується на аналізі гістограми зображення. Її функція полягає в тому, щоб візуалізувати гістограму опіків [2] зображення шкіри і знайти середнє та стандартне відхилення. З результатів,

зображених на рисунку 1 можна зробити висновок, що чим довше шкірні покрови були під дією вогню - тим темніше зображення.



яскравість

Рисунок 1. **Результати гістограмного аналізу**

На зображенні ураженої опіками шкіри, ми можемо з’ясувати, що колір уражених ділянок різний для різної тяжкості пошкоджень [3]. Таким чином, кольоровий аналіз також повинен бути врахований, щоб найкращим чином передбачити ступінь опіків шкіри [4]. Отже потрібно побудувати гістограму кольорів оригінальних кольорових зображень, які перетворюються з RGB колірного простору до HSV-простору. На гістограмі кольорів абсциса призначена для колірних значень, а поздовжні координати для числа пікселів:

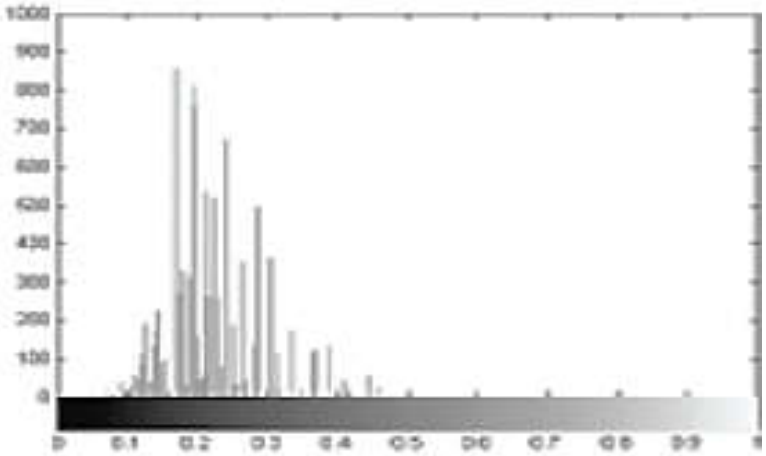


Рисунок 2. **Результати колірного аналізу**

Як видно з рисунку 2, якщо ступінь поверхневих опіків різна, на відповідній кольоровій гістограмі це, головним чином, відображається в розподілі значень кольору, а саме розподіл розсіювання кольору, який може бути позначений як стандартне відхилення кольорової гістограми.

Проти традиційної техніки оцінки опіків шкіри, що не є математично точною, було синтезовано новий метод визначення ступеня опіків шкіри на основі комп'ютерної технології обробки зображень.

За допомогою гістограмного аналізу сірошкального зображення отримуємо середнє значення коефіцієнту опіків зображення гістограми, стандартне відхилення кольорових карт [5], а також відсоток обпалених ділянок як індикатори опіків, що розподіляють показники зважених значень для

побудови моделі оцінки опіків шкіри. За цією моделлю можна обчислити значення опіків.

Список використаних джерел:

1. Cui Yi. Digital Image Processing Technology and Application [M]. Beijing: Electronics Industry Press 1997 1-17
2. Zhang Dapeng. Pattern Recognition and Image Processing parallel computer system design[M]. HARBIN: Harbin Institute of Technology Press 1998:32-45.
3. Paolo Gay, Remigio Bemuto, Pietro Piccarolo. Fruit color assessment for quality grading purposes. 2002 ASAE Annual Meeting, Paper No. 026097.
4. Erfan S Ahmad, John F Reid. Evaluation of color representations formaize images. Journal of agricultural Engineering Research, 1996,63:185-196.
5. Liao K, Paulsen M R, Reid J F. Real-time detection of color and surface defects of maize kemels using machine vision. Journal of agricultural Engineering Research,1994,59:263-271.

Сімонов А.Ю., *студент*
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського, м. Київ
Факультет інформатики і обчислювальної техніки, студент

СИСТЕМИ УПРАВЛІННЯ ОБЧИСЛЮВАНИМИ РЕСУРСАМИ

Кількість додатків та обчислювальних ресурсів невпинно зростає щодня. Через відносну простоту використання та більш дешевшу вартість ресурсів все більшу популярність набирають платформи хмарних обчислень. Але з зростанням кількості обчислюваних ресурсів виникає проблема ефективного управління та раціонального використання даних ресурсів. Забезпечити дану потребу допомагають оркестратори.

Оркестратор або система управління дозволяє в автоматичному режимі розгортати, масштабувати, координувати та керувати складними комп'ютерними системами та службами. Одним із напрямків що стрімко розвивається є системи управління контейнерами, так як на сьогодні використання контейнерів при розгортанні програмного забезпечення є одним із найбільш розповсюджених підходів. Згідно прогнозів компанії 451 Research, який був опублікований на початку 2017 року, світовий обсяг продажів контейнерів в найближчий час буде тільки зростати і до 2020 року темпи середньорічного зростання будуть складати приблизно 40% [1]. Віртуалізація на рівні операційної системи або контейнеризація – це метод віртуалізації, при якому ядро операційної системи підтримує декілька ізольованих примірників простору користувача. Дані примірники з точки зору користувача повністю ідентичні реальному серверові. Ядро операційної системи забезпечує повну ізоляцію контейнерів, тому програми з різних контейнерів не можуть впливати одна на одну. На сьогодні однією з найбільш розповсюджених технологій контейнеризації є Docker.

На разі найбільш розповсюдженою системою управління контейнерами є Kubernetes. Це відкрите програмне забезпечення яке було розроблено компанією Google для керування контейнерами. Kubernetes підтримує основні технології контейнеризації такі як Docker, а також є можливість підтримки апаратної віртуалізації. Система дозволяє запускати контейнери на великій кількості серверів, а також забезпечує сумісне розміщення і реплікацію великої кількості контейнерів. Kubernetes пропонує API високого рівня який надає можливість балансувати навантаження, групувати та розміщувати контейнери на різних серверах.

Система управління Docker Swarm також є достатньо розповсюдженою технологією в корпоративному середовищі. Компанія Docker була однією з перших яка запропоновувала своє рішення для керування кластерами контейнерів. Docker Swarm надає API інтерфейс який сумісний з Docker API, зо значно спрощує роботу з ним. Дана система складається з декількох серверів Docker, які поділяються на сервери що керують і ті що виконують різні процеси на кластері. Docker Swarm простіший у використанні але є менш надійний ніж Kubernetes тому є не кращим рішенням для управління високонавантаженими кластерами контейнерів.

Також однією з найбільш розповсюджених систем керування контейнерами є Apache Mesos. Дана система є новим ефективним рішенням керування серверною інфраструктурою. Якщо розглядати дану систему з точки зору традиційної віртуалізації, то вона відрізняється тим що замість розділення фізичних серверів на декілька віртуальних, Apache Mesos пропонує об’єднувати їх в одне ціле, а точніше в єдиний віртуальний ресурс. Розподілення ресурсів для виконання задач в системі Apache Mesos схоже на те як ядро операційної системи Linux розподіляє ресурси між локальними процесами. Якщо всі віртуальні ресурси об’єднати в один кластер то це дозволить підвищити ефективність використання ресурсів і паралельно може підвищити швидкість обробки задач. Кластер Apache Mesos також здатний створювати по новому окремі ресурси у випадку їх несправності, масштабувати ресурси вручну або автоматично при певних умовах та інше.

Отже на сьогодні існує достатня кількість систем управління обчислювальними ресурсами і кожна з них має свої переваги й недоліки. Різноманітні системи мають різне призначення і тому при проектуванні кластеру слід враховувати всі доступі характеристики.

Література:

1. https://www.itweek.ru/infrastructure/article/detail.php?ID=196419
2. https://docs.docker.com/engine/swarm/key-concepts/
3. https://habr.com/ru/post/308812/
4. https://habr.com/ru/post/258443/
5. https://habr.com/ru/company/d2cio/blog/349138/
6. https://ru.wikipedia.org/wiki/%D0%9E%D1%80%D0%BA%D0%B5%D1%81%D1%82%D1%80%D0%BE%D0%B2%D0%BA%D0%B0_(%D0%98%D0%A2)
7. https://ru.wikipedia.org/wiki/Kubernetes

Хомишин Д.О.
Харківський національний університет радіоелектроніки, м. Харків
Кафедра програмної інженерії

ДОСЛІДЖЕННЯ МЕТОДІВ КЛАСТЕРИЗАЦІЇ ДАНИХ ПРО ВИКОРИСТАННЯ РОЗУМНИХ ПАРКІНГІВ З МЕТОЮ ПРОГНОЗУВАННЯ ЇХ ЗАВАНТАЖЕННЯ

Дослідження даних про зайнятість паркінгів має велику актуальність, зважаючи на те, що завдяки цьому з'являється можливість отримувати актуальні данні пересування населення містом. Моя мета полягає в тому, щоб зробити ці дані корисними для таких користувачів як паркувальники, комерційні бізнеси, міська влада тощо.

Використання автоматизованих приладів для моніторингу на вуличних паркінгах стало популярним у кількох містах світу. У існуючих середовищах невеликі пристрої для зчитування зазвичай розміщуються в кожному місці для паркування. Це необхідно для моніторингу великих міських територій. Гарними прикладами серед багатьох інших є Лос-Анджелес, Сан-Франциско і Барселона. Перший шар цих складних систем складається з датчиків паркування на вулиці: дрібними бездротовими пристроями, що використовуються для моніторингу наявності транспортних засобів. Кожен датчик періодично прокидається, щоб перевірити стан зайнятості призначеної стоянки. Отримані дані надсилаються до серверу для подальшої обробки, управління віддаленим паркінгом та візуалізацією.

Основна мета цих систем полягає в підвищенні ефективності роботи громадської паркінгу, що досягається за рахунок збору інформації про зайнятість стоянки. Зібрані дані аналізуються та надаються відділу управління паркінгами міста через відповідні інформаційні панелі. Крім того, наявність інформації про паркування в режимі реального часу також дає змогу надавати нові послуги, забезпечуючи покращення роботи користувачам. Як приклад, системи паркування та інформації допомагають водіям більш ефективно знаходити місця для паркування, тим самим вирішуючи проблему довгого пошуку вільного місця.

Для досягнення поставленої мети застосуємо оригінальну техніку кластеризації за допомогою самоорганізуючих карт (SOM), які є самокерованими нейронними мережами, що здатні вивчати прототипи в багатовимірних векторних просторах.

SOM є нейро-обчислювальним алгоритмом, який перетворює високовимірні дані в одно- або двовимірний простір через нелінійний, конкурентний і без нагляду процес навчання. SOM відрізняється від інших штучних нейронних мереж, оскільки використовує функцію сусідства для збереження топологічних властивостей вхідного простору. Вона вивчається за

допомогою вхідних прикладів, а вхідний простір відображається в двовимірну решітку нейронів, зберігаючи властивість, що подібні вхідній структурі відображені сусідніми нейронами на карті.

Розглянемо одно- та двовимірні карти, які відповідно складаються з послідовностей M × 1 нейронів і решітки ℓ = M × M нейронів, з M> 1. Ці карти вибірково налаштовані на вхідні структури через безкваліфікований (також називається конкурентним) процес навчання. Коли навчання прогресує, ваги нейронів мають тенденцію ставати впорядкованими по відношенню один до одного таким чином, що над решіткою створюється значна система координат для різних функцій введення. Іншими словами, SOM створює топографічну карту вхідного простору даних, де просторові місця розташування або координати нейронів в решітці відповідають певній області або власній статистичній функції вхідних даних. Примітно, що це досягається без необхідності будь-яких попередніх знань щодо розподілу вхідних даних.

За допомогою $\mathcal{X} \subset \mathbb{R}K$ вказуємо множину ознак (вхід), а $xi \in \mathcal{X}$ - вхідний вектор ознак, пов'язаний з сенсором $i = 1, \ldots, N$, де $xi = [xi1,xi2,\cdots,xiK]T$. З N ми маємо на увазі число датчиків, і $|\mathcal{X}| = N$. Нехай $\mathcal{L}$ - решітка. Кожен нейрон з'єднаний з кожним компонентом вхідного вектора, як показано на рисунку 1. Зв'язки між вхідним вектором і нейронами зважені, так що j-й нейрон пов'язаний з синаптичним ваговим вектором wj ∈ ℝK, де $wj = [wi1,wi2,\cdots,wiK]T$.

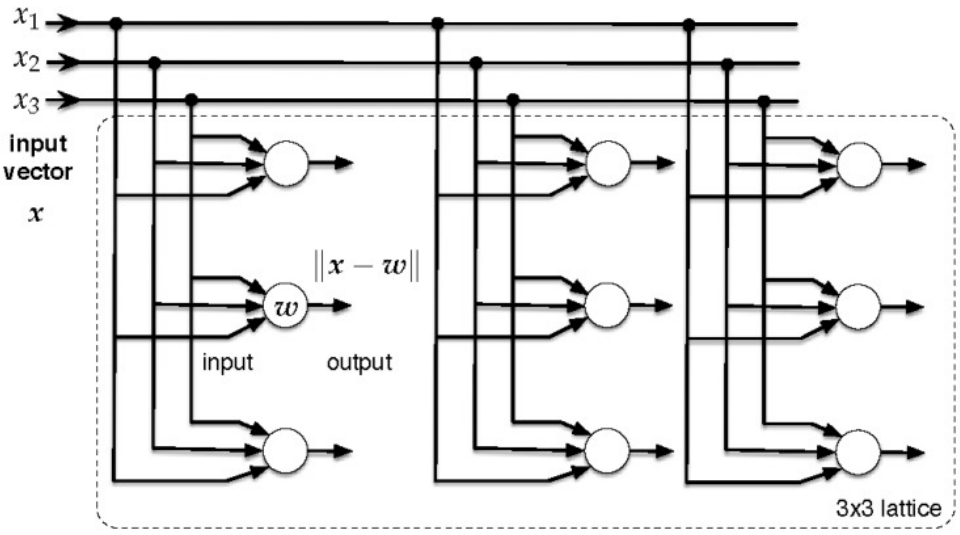


Рисунок 1 - Схема процесу навчання SOM

На рисунку 2 зображено схему роботи алгоритму SOM в загальному вигляді.

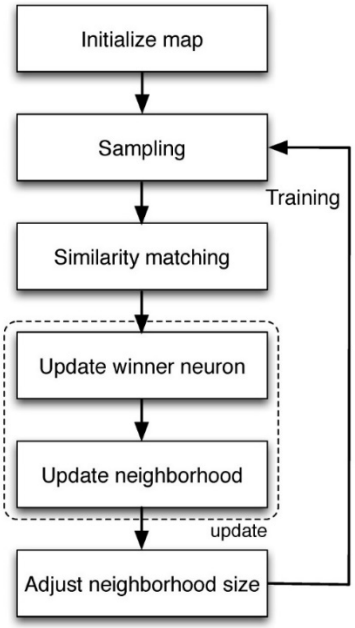


Рисунок 2 – Алгоритм SOM в загальному вигляді

Чемерис М.М., к.т.н., доцент
Черкаський національний університет імені Б. Хмельницького, м. Черкаси
Житков М.В., магістрант 2-го року навчання
Черкаський національний університет імені Б. Хмельницького, м. Черкаси

ПЕРЕДУМОВИ СТВОРЕННЯ WEB-СЕРВІСУ ОБЛІКУ ТА ПЛАНУВАННЯ ЗАХОДІВ МІСТА

В Україні протягом останніх кількох років успішно поширюється на місцевому рівні така форма громадської участі у вирішенні місцевих справ, як громадський бюджет. В бюджеті територіальної громади виділяється фіксована сума, яка витрачається на реалізацію проектів чи виконання певних робіт. Проекти готуються жителями відповідної громади, та визначаються на основі голосування за них теж жителями міста.

Отже процес керування бюджетом належить до класу неструктурованих або слабко структурованих; для керування бюджетом неможливо використати існуючі результати в області керування технічними системами, тому потрібні нові підходи, у тому числі засновані на нових інформаційних технологіях.

Реєструючись на web-сервісі, користувач має можливість запропонувати свій або проголосувати за один з наявних проектів, який на його думку в першу чергу необхідний для реалізації. Розроблюваний програмний продукт дозволить визначити шляхом прямого голосування користувачів найбільш важливі проекти для реалізації та покращення інфраструктури міста, а також об'єднати їх за допомогою інтелектуальних методів кластерного аналізу. Web - сервіс допоможе визначити актуальний проект за умови реалізації таких функцій, як авторизації та аутентифікації; подання проектів; опис характеристик проекту;

голосування за проекти; опрацювання результатів; дата оновлення та наповненість бази даних; моніторинг реалізації та завершення проектів.

Створення сервісу починається зі створення діаграми прецедентів (рис. 1), оскільки при моделюванні поведінки проектованої системи виникає необхідність не тільки уявити процес взаємодії, але і деталізувати особливості логічної складової виконуваних системою операцій.

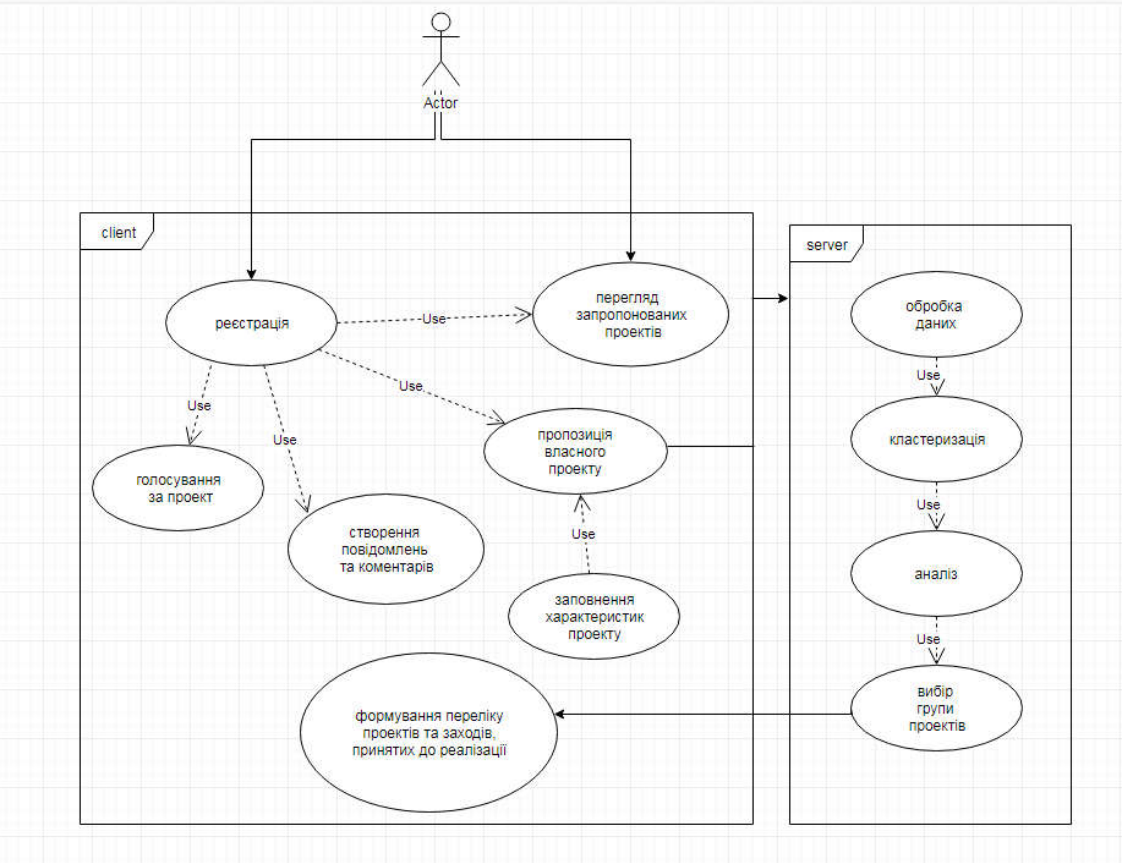


Рисунок 1. - Діаграма прецедентів для web-сервісу

Для застосування моделі визначення ступеня схожості різних заходів з використанням узагальнюючого критерію методом кластерного аналізу пропонується для кожного проекту вказувати такі характеристики, як користувачі продукту; мета; територія реалізації; тривалість; орієнтована вартість; джерела фінансування; потенційні учасники [1].

На основі цієї класифікації відбувається пошук схожості проектів за тими чи іншими параметрами. В результаті аналізу до реалізації приймається не один проект, який отримав найбільший рейтинг шляхом прямого голосування зацікавлених громадян, а портфель проектів, тобто група найбільш популярних заходів, які об'єднуються, за певними критеріями.

Література:

1. Осауленко И. А. Оптимизация портфеля региональных проектов методом кластерного анализа. Восточно-Европейский журнал передовых технологий. 2016. №2 (80). Т. 3. С. 40–49.

Юскович-Жуковська В.І., к. т. н., доцент
Соловей Л.Я., ст. викладач
ПВНЗ «Міжнародний економіко-гуманітарний університет імені академіка Степана Дем'янчука», м.Рівне
Кафедра інформаційних систем та обчислювальних методів

ПРОБЛЕМИ БЕЗПЕКИ ІНТЕРНЕТ-ІНФОРМАЦІЇ

Стрімкий розвиток Інтернет-технологій та їх широке використання в усіх сферах суспільного життя зумовили виникнення кіберзлочинності, тобто нового виду злочинності у сфері комп'ютерної інформації.

Суспільно небезпечні діяння у сфері кіберпростору та протидія їм являється актуальною темою для сучасних наукових досліджень. Серед науковців на сьогодні відсутня єдність поглядів на зміст та структуру цієї проблеми, що набуває міжнародного характеру. Так, експерти Ради Європи вважають, що практично неможливо встановити рівноцінний за ефективністю захист тієї інформації, яка зберігається або обробляється за допомогою комп'ютера, і тієї, що обробляється в більш традиційний спосіб (вручну). Ступінь захисту різниця у десять разів не на користь комп'ютерної інформації. Комп'ютерну інформацію значно простіше знищити, пошкодити, модифікувати, заволодіти нею тощо, ніж інформацією, яка зафіксована приміром на папері [1,с.20].

Оскільки в Інтернеті впроваджено технологію WWW (World Wide Web) – «Всесвітню павутину», яка є системою розподілених баз даних, що мають гіпертекстові зв'язки між документами, то постає проблема безпеки інформаційних сторінок державних органів, установ, комерційних організацій та підприємств в Інтернеті.

Так, через поширення в кінці червня 2017 року Інтернетом шкідливої програми вірусу «Петя», за повідомленням громадської організації "Громадянська кібероборона" – Україна недоотримала 0,4% ВВП за одну півгодинну кібератаку [2].

Не дивлячись на те, що розповсюдження та активне використання мережі Інтернет стало доступним в Україні у 1990-х роках, в даний час Інтернет-технології розвиваються надзвичайно інтенсивно. Якщо, спочатку, в 1991р. в Україні було два-три вузли, що надавали послуги електронної пошти (e-mail), то у 1993р. у Києві існувало вже шість основних Інтернет-сайти [3, с.31].

На сьогодні, за даними дослідження міжнародної організації "We are social", мережа Інтернет, як всесвітнє явище в Україні налічує понад 25,56 мільйонів активних користувачів [4]. Українські фірми-провайдери забезпечують гарантований розвиток Інтернету. Але до цього часу не існує чіткого визначення, чи є ця глобальна комп'ютерна мережа засобом зв'язку, чи засобом масової інформації, чи засобом комунікації чи чимось іншим.

Технічно Інтернет являє собою складну структуру, що об'єднує безліч локальних та глобальних мереж, які належать фірмам-провайдерам та пов'язані між собою різними лініями зв'язку. Тобто, Інтернет подібний до конструктора,

що складається з різноманітних мереж («павутини»), які постійно взаємодіють між собою, пересилаючи інформацію.

Проведені дослідження в галузі розвитку Інтернет-технологій свідчать, що однією з головних проблем інтенсивного розвитку ринку Інтернет-послуг є безпека інформації, що передається по мережі. Ринок Інтернет-сервіс провайдерів пропонує широкий асортимент різних видів доступу до «світової павутини». Наприклад, для користувачів з обмеженими фінансовими можливостями пропонується діал-ап-доступ. За ціновими параметрами такий вид підключення є найкращим вибором для користувачів, яких задовільняє невисока швидкість Інтернет-доступу.

Більш високу швидкість передачі даних забезпечують цифрові модеми та виділені лінії. Цей вид доступу гарантує високу швидкість і забезпечує потреби користувачів, обізнаних з сучасними інформаційними технологіями.

Бездротова мережа передачі даних базується на сотовій структурі, забезпечує оперативність, мобільність, можливість працювати на великих відстанях та ідеально підходить діловим користувачам.

Високошвидкісним та надійним являється супутниковий Інтернет, доступ до якого забезпечується за допомогою супутникових каналів зв'язку. Супутникові технології забезпечують надвисоку швидкість вхідного та вихідного трафіку.

В даний час найбільшої популярності набуває доступ до Інтернету на базі оптоволоконної лінії. Саме оптоволоконні мережі мають потужний потенціал і цій лінії найбільшу перевагу надають корпоративні користувачі.

Зі стрімким та різноманітним розвитком комп'ютерних мереж, радіоелектронної та супутникової техніки для Інтернету, цифрових носіїв інформації паралельно розвиваються та вдосконалюються підслуховуючі пристрої, кібершпигунство, організована комп'ютерна злочинність у сфері Інтернет-економіки, шахрайство в Інтернет-торгівлі, організоване комп'ютерне піратство, Інтернет-тероризм, заборонені електронні азартні ігри, комп'ютерне хуліганство, комп'ютерні віруси та ін..

Сучасний розвиток освіти, науки та техніки потребує оперативного доступу до світової інформації, тому непередбачені хакерські атаки можуть являти собою серйозну перешкоду в отриманні такої Інтернет-інформації.

Отже, із поширенням сфери Інтернет-технологій зростає і небезпека в комп'ютерних мережах. Провайдери не можуть заздалегідь забезпечити безпеку інформації, що передається мережею Інтернет ні технічно, ні апаратно, ні програмно. Юридично використання Інтернет в Україні регулюють міжнародні договори, ратифіковані Україною, національне законодавство та Укази Президента.

Незаконне втручання в роботу мережі Інтернет приводить до витоку інформації, її втрати, підробки, блокування, спотворення процесу обробки інформації, порушення встановленого порядку маршрутизації Інтернет-інформації.

Перехоплення Інтернет-інформації призводить до її збуту чи розповсюдження, копіювання, зміни або знищення. Такі дії мають суспільно

небезпечні наслідки та наносять значну шкоду правовласникам такої Інтернет-інформації.

Оскільки мережа Інтернет є відкритою, то роль безпеки інформації, якою обмінюються користувачі інформації, безпека всіх фінансових транзакцій, проведених через мережу та ін. набуває великого значення. Основними методами забезпечення безпеки в Інтернеті в даний час являються алгоритми шифрування (криптографія), електронний цифровий підпис та сертифікати. Брандмауєри спроможні виявляти порушників, блокувати їх спроби, проникнення у мережу, а також відслідковувати їх дії та місцезнаходження.

Телекомунікаційні мережі не захищені від вірусів. Найчастіше в них проникають файлові, завантажувальні й макровіруси, які бувають швидкодіючими, повільними та невидимками. Мета створення надійних серверів для забезпечення гарантованої безпеки електронної інформації та безпечного обслуговування в режимі онлайн ще не досягнута.

Таким чином, проблема безпеки Інтернет-інформації та її захисту в комп'ютерних мережах лишається до кінця не вирішеною та потребує активних наукових досліджень та технічних рішень.

Література:

1. Азаров Д.С. Злочини у сфері комп’ютерної інформації (кримінально-правове дослідження): Монографія.-К.: Атіка, 2007.-304с.
2. Електронний ресурс. Режим доступу до ресурсу: https://antikor.com.ua
3. Біленчук П.Д., Романюк Б.В., Цимбалюк В.С. та ін.. Компютерна злочинність. Навчальний посібник. - Київ: Атіка, 2002.-240с.
4. Електронний ресурс: https://wearesocial.com

Яшков И.О., канд. тех. наук, доцент Жгунова А.Е., ст.гр. АКТАКИТ-16-1
Харьковский национальный университет радиоэлектроники,
г. Харьков, Кафедра компьютерно-интегрированных технологий,
автоматизации и мехатроники

АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ МОДЕЛЕЙ В SOLIDWORKS

SolidWorks — программный комплекс САПР для автоматизации работ промышленного предприятия на этапах конструкторской и технологической подготовки производства. Обеспечивает разработку изделий любой степени сложности и назначения.

В основу SolidWorks положен модульный принцип построения, обеспечивающий масштабируемость системы для реализации сквозной автоматизированной поддержки процессов, связанных с изготовлением изделий, включая проектирование деталей и технологической оснастки, моделирование процессов механообработки и электроэрозии, генерацию управляющих программ для станков с ЧПУ. Базируясь на технологиях

гибридного параметрического моделирования, SolidWorks обеспечивает создание точной геометрической модели будущего изделия со всеми атрибутами, необходимыми для его изготовления (допуски на размеры, шероховатости, допуски формы и расположения поверхностей и др.). Также в пакет SolidWorks входят такие основные модули как:

- SolidWorks 3D Interconnect – программный модуль поддерживает форматы STEP, IGES и ACIS. Это означает, что программа может читать файловые документы из другого программного обеспечения.
- SolidWorks PDM – лучшая автоматизация процесса создания чертежей PDF.
- SolidWorks Simulation - алгоритм обнаружения сингулярности напряжения, метод управления перемещением.

Основой для работы в SolidWorks является проектирование изделий из листового материала. Оно включает этапы моделирования изделия, получения заготовки и разработки процесса. Наиболее продвинутым является подход пространственной модели. Этот объект может формироваться как твердое тело или описываться поверхностями. В первом случае применяются специальные функции, имеющиеся в современных геометрических САПР. Второй подход – использование аппарата поверхностного моделирования.

На рисунке 1.1 представлена простейшая модель, созданная в данной программе.

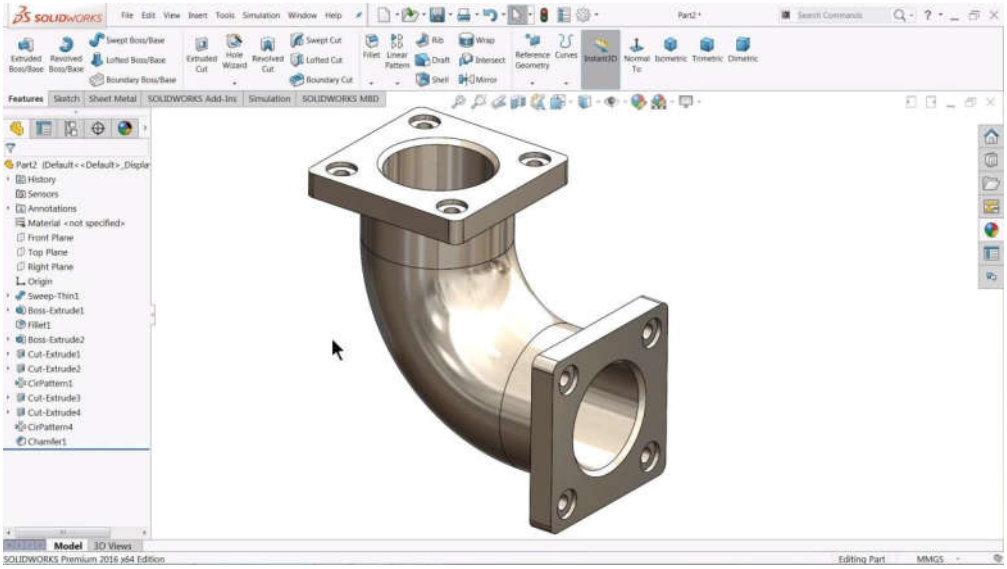


Рисунок 1.1 – Деталь в SolidWorks

Подводя итоги, можно сказать, что программа заслуженно считается одной из самых популярных систем автоматизированного проектирования, полностью удовлетворяющей потребности разнообразных промышленных организаций и частных лиц. Это мощный инструмент для комплексного проектирования изделий и компонентов любой сложности, в том числе и для промышленного дизайна. В среде 3D печати SolidWorks остается наиболее распространенным приложением для создания технических компонентов и твердотельного моделирования в целом.

Литература:

1. SolidWorks. Компьютерное моделирование в инжинерной практике / Алямовский А.А., Собачкин А.А. – СПб.: БХВ-Петербург, 2005. – 800 с.
2. Основы моделирования в SolidWorks / Зиновьев Д.В. – М. ДМК Пресс, 2017. – 240 с.

Voskoboinyk E.I.
Kharkiv National University of Radio Electronics, Kharkiv
Department of Computer-Integrated Technologies, Automation and Mechatronics, student
Hura Y.P.
Kharkiv National University of Radio Electronics, Kharkiv
Department of Computer-Integrated Technologies, Automation and Mechatronics, student
Kyrylenko O.A.
Kharkiv National University of Radio Electronics, Kharkiv
Department of Computer Science, Software Engineering , student

COMPUTER NETWORK REENGINEERING OF ROBOTIC SYSTEMS

One of the characteristics of the development of society at present is the widespread introduction of computerized and robotic technologies, which are increasingly used in all spheres of human activity: in everyday life, in manufacturing enterprises, in banking, military spheres, etc. However, the massive use of non-interconnected computer and robotic elements raises several problems associated with the preservation of information, sharing it. An effective solution to these problems at the present stage was the unification of such elements through computer networks.

Modern robotic systems have in their set interconnected automatic devices intended for performance of production and / or other operations and elements interacting with the external environment and between them. The main elements of such systems include computer networks that include computing devices, servers, routers, switches and other equipment that provides data exchange between elements of the network and the environment. They are usually created by wired or wireless technologies of tree topology, which combines the properties of two other topologies: tires and stars. Each of their higher-level knots is connected to nodes of the lower level by a star-shaped bond, forming a combination of stars.

Rapid changes in the external environment, computer, robotics and technologies lead to the need for reengineering both robotic systems and computer networks on which they operate. The processes of designing and reengineering computer networks as geographically distributed objects involve solving sets of interconnected tasks of structural, topological and parametric optimization [1].

$$MetaTask = \{Task_1\}, Task_1 = \{Task_i^1\}, i = 1, i_1, 1 = 1, n_1, \qquad (1.1)$$

where $Task_1$ – a set of synthesis tasks that relate to level l; n_1 – number of

levels of description of the TRS; i – number of the task; i_1 – the number of tasks to be solved at level l.

Each of the tasks at this stage can be represented in the form 1.2:

$$Task_i^1 = \ln_i^1 \to Out_i^1, i = 1, i_1, 1 = 1, n_1, \quad (1.2)$$

where $\ln_i^1, Out_i^1$ – the input and output data of the i-task of level l.

In this case, each of the selected tasks $Task_i^1, i = 1, i_1, 1 = 1, n_1$, generally, can be represented as a set of interconnected subtasks $Task_i^1 = \{Task_{ij}^1\}, j = 1, j_i$, where j_i - the number of subtasks of the task $Task_i^1$.

These tasks belong to the class of NP-complex and are solved under incomplete information on a set of functional and cost indicators (partial criteria). The choice of the best solution from the set of effective only in the simplest situations can be made by the person who makes the decision [2].

In the process of reengineering, designing a network structure, modeling in order to optimize the resulting structure and selecting an optimal implementation plan, taking into account the material resources allocated at this stage. Nowadays, the main communication devices of computer networks are switches, not routers or hubs, because switches are optimal in terms of their characteristics and prices.

In general reengineering is carried out in several stages. The first step is the diagnosis of network infrastructure, in which structure cabling system, placement of technical facilities, robotic network elements are defined. Next, the number and location of communication devices, the detection of problems and errors in the network topology are determined. Then - study the components of the network: servers, active and passive network equipment and workstations, namely checking the equipment for performance, correctness of settings and the ability to combine within the same network.

The next stage, based on the results of the network diagnostics, is an analytical note, which clearly shows: options for network upgrades that provide different levels of tasks, technical capacities and budgets, lists of required work and equipment, for each version of the upgrade.

Subsequently, installation works are carried out, namely the installation of new network equipment instead of outdated, installation of new cable segments of the network and the installation of additional necessary equipment.

The final stage is an adjustment of new network equipment, change the settings and configurations of network services, driver versions update for routers, setting access rights divisions.

Due to the complexity of the tasks to be performed during reengineering, the use of imitative simulation methods allows for reasonable management decisions and minimizing costs. In the general case, the life cycle of the ship's system can be described by the vector of development, according to formula 1.3:

$$V_s = (S_{c1}, S_{c2}, \ldots, S_{cq}), \quad (1.3)$$

where S_{c1} – the initial structure of the network; S_{cq} – the final structure of the network; q – the number of stages of the life cycle.

During the reengineering of the network structure, static and modified

fragments of the network are distinguished. Static fragments correspond to the optimality criterion at this stage and do not require redesign. A modified fragment requires some work. Proceeding from the above, the structure of the network can be represented as a combination of these fragments according to formula 1.4.

$$S_c = S_c^s \cup S_c^m \quad (1.4)$$

where S_c^s – static fragment; S_c^m – modified fragment.

Further, the synthesis of a computer network is reduced to the synthesis of the optimal structure of the modified fragment by the optimality criterion according to the formula 1.5.

$$E = \sum_{i=1}^{M} \frac{\vartheta_i}{C_{ki} + \sum_{j=1}^{N} l_j C_{kr}} \to \min, \quad (1.5)$$

where M – number of switches; ϑ_i – boolean variable, which is determined according to formula (1.6); C_{ki} – cost of the switch in the i-th fragment; C_{ij} – the cost of the cable; l_j – required length of the cable to connect the nodes in the j fragment;

$$\vartheta_i = \begin{Bmatrix} 1, if \sum_{i=1}^{N} \lambda_i \le P \\ 0, if \sum_{i=1}^{N} \lambda_i \ge P \end{Bmatrix}, \quad (1.6)$$

where N – number of subscribers of the switch; λ_i – load of the i-th port of the switch; P – bandwidth of the switch.

This model does not take into account all the necessary factors. So, when calculating the cost of nodes, the cost of installing the equipment and the cost of the various elements used during the installation of the equipment may not be taken into account. However, the practical use of the results can reduce the timing of the design; planning of development and reengineering of networks; and cost of their creation and operation, through general solving of problems to increase the quality of solutions and on this basis to improve the functional characteristics of computer networks robotic systems.

References:

1. Beskorovayny V.V. Systemological analysis of the problem of structural synthesis of territorially distributed systems // Automated control systems and automation devices. - 2002. - Issue 120. - P. 29-37.

2. Beskorovayny V.V. Synthesis of the logical scheme of system design of geographically distributed objects // Radioelektronika i informatika. - 2002. - No. 3. - P. 94 - 96.

3. Nesterenko S. Selection of the optimal plan for the implementation of corporate computer networks taking into account the dynamics of their development. /WITH. Nesterenko, A. Binkovsky, A. Cherny // Proceedings of the Odessa Polytechnic University. - 2005, No. 1 (41). - P. 66-68.

4. Olifer V.G., Olifer N.A., Computer networks. Principles, technologies, protocols. - 4th ed. - St. Petersburg: Peter, 2011. - P. 49-50.