

БИОНИКА ИНТЕЛЛЕКТА

ИНФОРМАЦИЯ, ЯЗЫК, ИНТЕЛЛЕКТ

№ 2 (93)

2019

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Основан в октябре 1967 г.

Учредитель и издатель

Харьковский национальный университет радиозлектроники

Периодичность издания – 2 раза в год



Научно-технический журнал
«БИОНИКА ИНТЕЛЛЕКТА»

ISSN 0555-2656

Основан Харьковским национальным университетом
радиоэлектроники в 1967 году

Реферирование и индексирование:

Google Scholar



INDEX  COPERNICUS
I N T E R N A T I O N A L



Журнал включен в список научных специализированных изданий Украины
по техническим и физико-математическим наукам
согласно приказа Министерства образования и науки Украины № 820 от 11.07.2016



Є.В. Бодянський¹, А.Ю. Шафроненко², І.М. Климова³

¹Доктор технических наук, профессор кафедры Искусственного интеллекта

Харьковский национальный университет радиоэлектроники
yevgeniy.bodyanskiy@nure.ua ORCID 0000-0001-5418-2143

²Кандидат технических наук, доцент кафедры Информатики
Харьковский национальный университет радиоэлектроники
alina.shafronenko@nure.ua ORCID 0000-0002-8040-0279

³Ассистент кафедры Системотехники
Харьковский национальный университет радиоэлектроники
iryana.klymova@nure.ua ORCID 0000-0003-0455-6180

ОНЛАЙН ДОСТОВІРНА НЕЧІТКА КЛАСТЕРІЗАЦІЯ ДАНИХ З ВИКОРИСТАННЯМ ФУНКЦІЇ НАЛЕЖНОСТІ СПЕЦІАЛЬНОГО ТИПУ

Предложен онлайн метод достоверной нечеткой кластеризации, предназначенный для анализа данных, последовательно поступающих на обработку. Особенностью развиваемого подхода является использование функции принадлежности специального вида, описываемой функции плотности распределения Коши. Собственно процедура уточнения центроидов кластеров является по сути правилом самообучения «Победитель получает больше» (WTM), в котором функция соседства порождается введенной функцией принадлежности.

НЕЧЕТКАЯ КЛАСТЕРИЗАЦИЯ, ОБРАБОТКА, ГРАДИЕНТНАЯ ОПТИМИЗАЦИЯ, ФУНКЦИЯ ПРИНАДЛЕЖНОСТИ.

Запропоновано онлайн метод достовірної нечіткої кластеризації, призначений для аналізу даних, послідовно надходять на обробку. Особливістю розвиваючого підходу є використання функції належності спеціального виду, описуваної функції щільності розподілу Коші. Власне процедура уточнення центроїд кластерів є по суті правилом самонавчання «Переможець отримує більше» (WTM), в якому функція сусідства породжується введеною функцією приналежності.

НЕЧІТКА КЛАСТЕРИЗАЦІЯ, ОБРОБКА, ГРАДІЄНТНА ОПТИМІЗАЦІЯ, ФУНКЦІЯ НАЛЕЖНОСТІ.

An online method of reliable fuzzy clustering is proposed, designed to analyze data sequentially received for processing. A feature of the developed approach is the use of the membership function of a special kind described by the density function of the Cauchy distribution. The actual procedure for clarifying the centroids of clusters is essentially a self-learning rule “The Winner Takes More” (WTM), in which the neighborhood function is generated by the introduced membership function.

FUZZY CLUSTERING, PROCESSING, GRADIENT OPTIMIZATION, PROCESSING, ACCESSORIES FUNCTION.

Вступ

Задача кластеризації (класифікації в режимі самонавчання) багатовимірних даних є важливою частиною інтелектуального аналізу даних (Data Mining), в рамках якої склався ряд напрямків і підходів [1, 2]. Один з таких напрямків утворюють методи нечіткої (фаззі-) кластеризації, в основі яких лежить припущення що класи-кластери, які формуються, взаємно перетинаються так, що кожен вектор-спостереження з різними рівнями належності-ймовірності-можливості може належати одночасно до декількох класів одночасно.

Тут найбільш широкого поширення набули алгоритми ймовірнісної нечіткої кластеризації і, перш за все, метод нечітких с-середніх (FCM) [3, 4]. Цей підхід обмежується ймовірнісними обмеженнями на рівні належності так, що «забруднені» збуреннями і викидами спостереження можуть бути віднесені до різних класів з практично однаковими рівнями належності.

У зв'язку з цим в [5] був запропонований можливісний підхід до нечіткої кластеризації (PCM) більш стійкий до шумів і збурень. Разом з тим PCM-алгоритми

страждають від, так званої, проблеми співпадіння, коли в процесі обробки інформації деякі кластери починають зливатися один з одним, що в результаті веде до невірної оцінки кількості сформованих кластерів.

Цих недоліків позбавлені алгоритми достовірної нечіткої кластеризації [6-8], засновані на апараті теорії достовірності [9]. В рамках цього підходу в процесі розрахунків оцінюються не тільки рівні нечіткої належності, але і рівні довіри, що засновані на мірі належності спеціального виду [10]. Результати експериментів показали [7, 8], що достовірний підхід забезпечує більш високу якість кластеризації в порівнянні з ймовірнісними і можливісними методами.

Вихідною інформацією для рішення задачі нечіткої кластеризації є масив n -вимірних векторів спостережень $X = \{x_1, x_2, \dots, x_N\} \subset R^n$, $x(k) \in X$, $k = 1, 2, \dots, N$, який повинен бути розбитий на m класів-кластерів з деяким рівнем належності-можливості-достовірності $U_q(k)$ k -го вектора x_k - q му кластеру ($1 < m < N, 1 \leq q \leq m$). Необхідно також відзначити, що вихідні дані попередньо передоброблені так, що $-1 \leq x_{ki} \leq 1$ ($1 \leq i \leq n$), де x_{ki} i -та компонента вектора x_k .

Таким чином, завдання кластеризації вирішується в пакетному режимі, коли весь масив даних обробляється багаторазово на основі почергового оцінювання [8]. Якщо ж дані надходять на обробку в вигляді потоку або утворюють надвеликі масиви, пакетний режим не дозволяє ефективно вирішити цю задачу.

У цій ситуації найбільш ефективними є рекурентні процедури нечіткої кластеризації, що дозволяють вирішувати задачу в online режимі, уточнюючи результати в міру надходження кожного нового спостереження. Так, в [11,12] були запропоновані рекурентні варіанти FCM, які є по суті градієнтними процедурами оптимізації прийнятої цільової функції, а в [13,14] були введені рекурентні модифікації РСМ, що призначені для послідовної обробки даних.

У зв'язку з цим є доцільною розробка рекурентної модифікації методу достовірної нечіткої кластеризації, що дозволяє уточнювати характеристики кластерів у міру надходження кожного нового спостереження.

1. Рекурентний метод достовірної нечіткої кластеризації (RCCM)

Найбільш популярний метод ймовірнісної нечіткої кластеризації пов'язаний з мінімізацією цільової функції [4]

$$E(U_q(k), w_q) = \sum_{k=1}^N \sum_{q=1}^m U_q^\beta(k) D^2(x_k, w_q) \quad (1)$$

за обмежень $\sum_{q=1}^m U_q(k) = 1, 0 < U_q(k) < N$. Вирішуючи задачу нелінійного програмування за допомогою методу невизначених множників Лагранжа, приходимо до відомого результату

$$U_q^{(\tau+1)}(k) = \left(D^2(x_k, w_q^{(\tau)}) \right)^{\frac{1}{1-\beta}} \left(\sum_{l=1}^m \left(D^2(x_k, w_l^{(\tau)}) \right)^{\frac{1}{1-\beta}} \right)^{-1}, \quad (2)$$

$$w_q^{(\tau+1)} = \sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^\beta x_k \left(\sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^\beta \right)^{-1}, \quad (3)$$

де $U_q(k)$ -рівень належності векторного спостереження x_k q -му кластеру $Cl_q (1 \leq q \leq m)$, w_q — прототип-центроїд q -го кластера, $\beta > 1$ фаззифікатор, що визначає «розмитість» границь між класами, $D(x_k, w_q)$ — відстань між x і w_q в прийнятій метриці, $\tau = 0, 1, 2, \dots$ — індекс епохи обробки інформації в режимі попереминого оцінювання. При цьому процес обчислень триває до виконання умови

$$w - w \leq \varepsilon \forall 1 \leq q \leq m,$$

де ε — наперед заданий поріг точності обчислень.

В разі $\beta = 2$ і евклідової метрики

$$D^2(x_k, w_q) = \|x_k - w_q\|_2^2,$$

приходимо до популярного алгоритму нечітких с-середніх (FCM) [13] виду

$$U_q^{(\tau+1)}(k) = \left\| x_k - w_q^{(\tau)} \right\|^{-2} \left(\sum_{l=1}^m \left\| x_k - w_l^{(\tau)} \right\|^{-2} \right)^{-1}, \quad (4)$$

$$w_q^{(\tau+1)} = \sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^2 x_k \left(\sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^2 \right)^{-1}. \quad (5)$$

Якщо дані надходять на обробку послідовно в онлайн режимі, задача нелінійного програмування може бути вирішена за допомогою алгоритму Ерроу-Гурвіца-Удзави, що є за суттю градієнтною процедурою пошуку сідлової точки функції Лагранжа на основі критерія (1) з обмеженнями на суму належностей.

При цьому співвідношення (2), (3) можуть бути переписані у формі

$$\begin{cases} U_q(k+1) = \left(D^2(x_{k+1}, w_q^{(k)}) \right)^{\frac{1}{1-\beta}} \left(\sum_{l=1}^m \left(x_{k+1}, w_l^{(k)} \right)^{\frac{1}{1-\beta}} \right)^{-1}, \\ w_q(k+1) = w(k) + \eta(k+1) U_q^\beta(k+1) (x_{k+1} - w_q(k)), \end{cases} \quad (6)$$

(тут $\eta(k)$ — параметр кроку навчання, а (4), (5)

$$\begin{cases} U(k+1) = \|x - w(k)\| \left(\|x - w(k)\| \right), \\ w(k+1) = w + \eta(k+1) U(k+1) (x - w(k)), \end{cases} \quad (7)$$

що є узагальненням рекурентних процедур Парка-Деггера [11] і Чанга-Лі [12].

Можливісні алгоритми нечіткої кластеризації засновані на мінімізації цільової функції [5]

$$E(U(k), w, \mu) = U(k) D(x, w) + \mu (1 - U(k)), \quad (8)$$

де $\mu_q \geq 0$ визначає відстань, на якій рівень належності приймає значення 0,5, тобто. $U_q(k) = 0$, якщо $D^2(x_k, w_q) = \mu$.

Мінімізація критерія (8) дозволяє отримати аналітичний розв'язок у вигляді

$$U(k) = \left(1 + \left(\frac{D(x, w)}{\mu} \right) \right)^{-1}, \quad (9)$$

$$w_q^{(\tau+1)} = \sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^\beta x_k \left(\sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^\beta \right)^{-1}, \quad (10)$$

$$\mu_q^{(\tau+1)} = \sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^\beta D^2(x_k, w_q^{(\tau+1)}) \left(\sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^\beta \right)^{-1}, \quad (11)$$

який у квадратичному випадку набуває форму

$$U_q^{(\tau+1)}(k) = \left(1 + \frac{\|x_k - w_q^{(\tau)}\|^2}{\mu_q^{(\tau)}} \right)^{-1}, \quad (12)$$

$$w_q^{(\tau+1)} = \sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^2 x_k \left(\sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^2 \right)^{-1}, \quad (13)$$

$$\mu_q^{(\tau+1)} = \sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^2 \|x_k - w_q^{(\tau+1)}\|^2 \left(\sum_{k=1}^N \left(U_q^{(\tau+1)}(k) \right)^2 \right)^{-1}. \quad (14)$$

Онлайн версії (9)-(14) при цьому мають вигляд [13,14]

$$\begin{cases} U_q(k+1) = \left(1 + \left(\frac{D^2(x_{k+1}, w_q(k))}{\mu_q(k)} \right)^{\frac{1}{\beta-1}} \right)^{-1}, \\ w_q(k+1) = w_q(k) + \eta(k+1) U_q^\beta(k+1) (x_{k+1} - w_q(k)), \\ \mu_q(k+1) = \sum_{p=1}^{k+1} U_q^\beta(p) D^2(x_p, w_q(k+1)) \left(\sum_{p=1}^{k+1} U_q^\beta(p) \right)^{-1} \end{cases} \quad (15)$$

і (при $\beta = 2$)

$$\begin{cases} U_q(k+1) = \left(1 + \frac{\|x_{k+1} - w_q(k)\|^2}{\mu_q(k)} \right)^{-1}, \\ w_q(k+1) = w_q(k) + \eta(k+1) U_q^2(k+1) (x_{k+1} - w_q(k)), \\ \mu_q(k+1) = \sum_{p=1}^{k+1} U_q^2(p) \|x_p - w_q(k+1)\|^2 \left(\sum_{p=1}^{k+1} U_q^2(p) \right)^{-1}. \end{cases} \quad (16)$$

Достовірна нечітка кластеризація пов'язана з мінімізацією цільової функції

$$E(Cr_q(k), w_q) = \sum_{k=1}^N \sum_{q=1}^m Cr_q^\beta(k) D^2(x_k, w_q) \quad (17)$$

за обмежень

$$\begin{aligned} 0 \leq Cr_q(k) \leq 1 \forall q, k; \sup Cr_q(k) \geq 0,5 \forall k; \\ Cr_q(k) + \sup Cr_l(k) = 1 \end{aligned}$$

для будь-яких q і k , для яких $Cr_q(k) \geq 0,5$. Тут $Cr_q(k)$ – достовірність того, що спостереження x_k належить кластеру Cl_q . При цьому рівень достовірності розраховується на основі функції приналежності [15]

$$U(k) = \varphi(D(x, w)), \quad (18)$$

що задовольняє умовам:

- $\varphi_q(\cdot)$ монотонно зменшується на інтервалі $[0, \infty]$,
- $\varphi_q(0) = 1$,
- $\varphi_q(\infty) \rightarrow 0$.

Нескладно помітити, що функція (18) є за суттю мірою подібності, заснованій на відстані [16].

В якості такої функції в [15] було запропоновано використовувати вираз

$$U_q(k) = \left(1 + D^2(x_k, w_q) \right)^{-1}, \quad (19)$$

що є звичайною дзвонуватою функцією належності, яка використовується в системах нечіткого висновування.

Цікаво зауважити, що вираз (2) може бути перепи-саний у формі

$$\begin{aligned} U_q(k) &= \left(D^2(x_k, w_q(k)) \right)^{\frac{1}{1-\beta}} \left(\sum_{l=1}^m \left(D^2(x_k, w_l(k)) \right)^{\frac{1}{1-\beta}} \right)^{-1} = \\ &= \left(D^2(x_k, w_q(k)) \right)^{\frac{1}{1-\beta}} \left(D^2(x_k, w_q(k)) \right)^{\frac{1}{1-\beta}} + \sum_{l \neq q}^m \left(D^2(x_k, w_l(k)) \right)^{\frac{1}{1-\beta}} \right)^{-1} = (20) \end{aligned}$$

$$= \left(1 + \left(D^2(x_k, w_q(k)) \right)^{\frac{1}{1-\beta}} \sum_{l \neq q}^m \left(D^2(x_k, w_l(k)) \right)^{\frac{1}{1-\beta}} \right)^{-1},$$

а для евклідової метрики і $\beta = 2$ приймає вид функції щільності розподілу Коші з параметром ширини σ_q^2 [16]

$$U_q(k) = \left(1 + \frac{\|x_k - w_q(k)\|^2}{\sigma_q^2} \right)^{-1}, \quad (21)$$

$$\sigma_q^2 = \left(\sum_{l=1}^m \|x_k - w_l(k)\|^2 \right)^{-1}. \quad (22)$$

Нескладно бачити, що функція належності (19) є окремим випадком (21) при $\sigma_q^2 = 1$.

Остаточо пакетний алгоритм достовірної нечіткої кластеризації може бути записаний у формі [7,8]:

$$U_q^{(\tau+1)}(k) = \left(1 + D^2(x_k, w_q^{(\tau)}) \right)^{-1}, \quad (23)$$

$$U_q^{*(\tau+1)}(k) = U_q^{(\tau+1)}(k) \left(\sup U_l^{(\tau+1)}(k) \right)^{-1}, \quad (24)$$

$$Cr_q^{(\tau+1)}(k) = \frac{1}{2} \left(U_q^{*(\tau+1)}(k) + 1 - \sup_{l \neq q} U_l^*(k) \right), \quad (25)$$

$$w_q^{(\tau+1)} = \sum_{k=1}^N \left(Cr_q^{(\tau+1)}(k) \right)^\beta x_k \left(\sum_{k=1}^N \left(Cr_q^{(\tau+1)}(k) \right)^\beta \right)^{-1} \quad (26)$$

На підставі (17), (21)-(26) можна записати онлайн версію алгоритма достовірної нечіткої кластеризації у вигляді

$$\begin{cases} \sigma_q^2(k+1) = \left(\sum_{l \neq q}^m \|x_{k+1} - w_l(k)\|^2 \right)^{-1}, \\ U_q(k+1) = \left(1 + \frac{\|x_{k+1} - w_q(k)\|^2}{\sigma_q^2(k+1)} \right)^{-1}, \\ U_{(k+1)}^* = U_q(k+1) \left(\sup U_l(k+1) \right)^{-1}, \\ Cr_q(k+1) = \frac{1}{2} \left(U_q^*(k+1) + 1 - \sup_{l \neq q} U_l^*(k+1) \right), \\ w_q(k+1) = w_q(k) + \eta(k+1) Cr_q^\beta(k+1) (x_{k+1} - w_q(k)). \end{cases} \quad (27)$$

Як видно, з обчислювальної точки зору online алгоритм достовірної нечіткої кластеризації не складні-ше рекурентних версій FCM і PCM, зберігаючи при цьому переваги достовірного підходу.

2. Результати обчислювального експерименту

Щоб перевірити ефективність розроблених методів, а також довести їх переваги перед аналогами, експериментальні дослідження були проведені за допомогою двох різних баз даних. Також проведено порівняльний аналіз якості кластеризації даних щодо основних характеристик оцінок якості, таких як: коефіцієнт розподілу (ПК), що визначає «перекриття» між групами точок; індекс розділу (SC), що кількісно визначає співвідношення суми компактності та розділеності кластерів; індекс C_i і Бені (XB), що вимірюють співвідношення загальної мінливості всередині кластерів та їх поділу.

Таблиця 1

Оцінка якості методів нечіткої кластеризації за допомогою першого набору даних

Методи кластеризації даних	Перший набір даних		
	PC	SC	XB
Fuzzy C-means (FCM)	0.50	1.62	0.19
Gustafson-Kessel	0.27	1.66	1.62
Gath-Geva	0.25	1.54	1.35
Адаптивна ймовірнісна нечітка кластеризація	0.25	1.44	0.01
Адаптивна нечітка можлива кластеризація даних	0.26	1.22	0.01
Адаптивна нечітка достовірнісна кластеризація даних	0.21	1.13	0.01

Таблиця 2

Оцінка якості методів нечіткої кластеризації за допомогою другого набору даних

Методи кластеризації даних	Другий набір даних		
	PC	SC	XB
Fuzzy C-means (FCM)	0.48	1.60	0.19
Gustafson-Kessel	0.26	1.64	1.62
Gath-Geva	0.26	1.50	1.35
Адаптивна ймовірнісна нечітка кластеризація	0.25	1.42	0.01
Адаптивна нечітка можлива кластеризація даних	0.37	1.11	0.18
Адаптивна нечітка достовірнісна кластеризація даних	0.23	1.22	0.01

Порівняно з відомими методами запропонований підхід до онлайн кластеризації даних демонструє дещо надійніші результати.

Висновок

Розглянуто задачу нечіткої кластеризації на основі ймовірнісного, можливісного і достовірного підходів на основі пакетного і online режимів надходження і обробки інформації. Введена рекурентна версія достовірного алгоритму, що є за суттю процедурою градієнтної оптимізації прийнятого критерія нечіткої достовірної кластеризації.

Список літератури

[1] Xu R., Wunsch D.C. II. Clustering— Hoboken, N.J.: John Wiley & Sons, Inc., 2009.
 [2] Aggarwal C.C. Data Mining: Text Book. Springer, 2015.

[3] Bezdek J.C. Pattern Recognition with Fuzzy Objective Function Algorithms. — N.Y.: — Plenum Press, 1981.
 [4] Höppner F., Klawonn F., Kruse R., Runkler T. Fuzzy Clustering Analysis: Methods for Classification, Data Analysis and Image Recognition.-Chichester: John Wiley & Sons, 1999. — 289 p.
 [5] R. Krishnapuram, J.M. Keller. A possibilistic approach to clustering. Fuzzy Systems, 1993, 1, № 2, P. 98–110.
 [6] Chintalapudi K. K. and M. kam, “A noise resistant fuzzy c-means algorithm for clustering,” IEEE conference on Fuzzy Systems Proceedings, vol. 2, May 1998, pp. 1458–1463.
 [7] Zhou J., Wang Q., Hung C.-C., Yi X. Credibilistic clustering: the model and algorithms. Int.J. of Uncertainty, Fuzziness and Knowledge-Based Systems- 2015-23-№4 — P. 545–564.
 [8] Zhou, J., Wang, Q., Hung, C. C. Credibilistic clustering algorithms via alternating cluster estimation.- J. Intell. Manuf.-2017-28 — P. 727–738.
 [9] Liu, B., & Liu, Y. Expected value of fuzzy variable and fuzzy expected value models. IEEE Transactions on Fuzzy Systems, - 2002-10-№4 — P. 445–450.
 [10] Liu, B. A survey of credibility theory. Fuzzy Optimization and Decision Making-2006-5-№4 — P. 387–408.
 [11] D.C. Park, I. Dagher. Gradient based fuzzy c-means (GB-FCM) algorithm. Proc. IEEE Int. Conf. on Neural Networks, 1984, P. 1626–1631.
 [12] F.L. Chung, T. Lee. Fuzzy competitive learning. Neural Networks, 1994, 7, №3, P. 539–552.
 [13] Bodyanskiy Ye, Kolodyazhnyi V., Stephan A. Recurcive fuzzy clustering algorithms. —Proc 10th East West Fuzzy Coll. 2002, -Zittau-Görlitz, HS, 2002 — P. 276–283.
 [14] Bodyanskiy, Ye. Computational intelligence techniques for data analysis / Ye. Bodyanskiy // Lecture Notes in Informatics.-Bonn: V.p. 72GI, 2005. — P. 15–36.
 [15] Zhou, J., & Hung, C.-C. (2007). A generalized approach to possibilistic clustering algorithms. Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems. — 2007 — 15. — P.117–138.
 [16] Young F.W., Hamer R.M. Theory and Applications of Multi-dimensional Scaling-Hillsdale, N.J.: Erlbaum, 1994.
 [17] Hu Zh., Bodyanskiy Ye, Tyshchenko O., Shafronenko A. Fuzzy clustering of incomplete data by means of similarity measures- Proc.2019 IEEE 2nd Ukr. Conf. on Electrical and Computer Engineering (UKRCON), Track 6.-Lviv, Ukraine, 2019. — P. 149–152.
 [18] Bezdek J.C. A convergence theorem for the fuzzy ISODATA clustering algorithms. — IEEE Trans. Pattern Anal. Mach. Intell. - 1980 - 2. — P. 1–8.
 Bodyanskiy Ye, Gorshkov Ye, Kokshenev I., Kolodyazhnyi V. Outlier resistant recursive fuzzy clustering algorithms. Ed. By B. Reusch «Computational Intelligence Theory and Applications» - Advances in Soft Computing-Vol.38.-Berlia Heidelberg, Springer Verlag, 2006 — P. 647–652.
 [19] Bodyanskiy Ye, Gorshkov Ye, Kokshenev I., Kolodyazhnyi V. Robust recursive fuzzy clustering algorithms- Proc. 12th East West Fuzzy Coll 2005 - Zittau-Görlitz, FH,2005 — P. 301–308.
 [20] Bodyanskiy Ye, Shafronenko A., Mashtalir S., Online robust fuzzy clustering of data with omissions using similarity measure of special type - Lecture Notes in Computational Intelligence and Decision Making-Cham: Springer, 2020 — P. 637–646

Надійшла до редколегії 4.09.2019

УДК 004.852

DOI 10.30837/bi.2019.2(93).04

О.Г. Руденко¹, О.О. Безсонов², Н.М. Сердюк³, К.О. Олійник⁴, О.С. Романюк⁴

¹Доктор технічних наук, завідувач комп'ютерних інтелектуальних технологій та систем,
Харківський національний університет радіоелектроніки,
oleh.rudenko@nure.ua, ORCID iD: 0000-0003-0859-2015

²Доктор технічних наук, професор кафедри комп'ютерних інтелектуальних технологій та систем,
Харківський національний університет радіоелектроніки,
oleksandr.bezsonov@nure.ua, ORCID iD: 0000-0001-6104-4275

³Кандидат технічних наук, доцент кафедри комп'ютерних інтелектуальних технологій та систем,
Харківський національний університет радіоелектроніки,
nataliya.serdyuk@nure.ua, ORCID iD: 0000-0002-0107-4365

⁴Аспірант кафедри інформатики, Харківський національний університет радіоелектроніки

РОБАСТНА ІДЕНТИФІКАЦІЯ ОБ'ЄКТІВ ЗА НАЯВНІСТЮ НЕГАУСІВСЬКИХ ЗАВАД

Розглянуто задачу ідентифікації параметрів лінійного об'єкта за наявністю негаусівських завад на основі мінімізації комбінованого функціоналу, який поєднує властивості МНК та МНМ. Визначено умови збіжності градієнтного алгоритму ідентифікації в середньому і середньоквадратичному. Отримано аналітичні оцінки неасимптотичних та асимптотичних значень помилки оцінювання параметрів і точності ідентифікації. Показано, що ці значення помилки оцінювання і точності ідентифікації залежать від вибору параметра змішування.

КОМБІНОВАНИЙ КРИТЕРІЙ, ГРАДІЄНТНИЙ АЛГОРИТМ, ПАРАМЕТР ЗВАЖУВАННЯ, РЕКУРЕНТНА ПРОЦЕДУРА, АСИМПТОТИЧНА ОЦІНКА, ТОЧНІСТЬ ІДЕНТИФІКАЦІЇ.

Рассмотрена задача идентификации параметров линейного объекта при наличии негауссовских помех на основе минимизации комбинированного функционала, который сочетает свойства МНК и МНМ. Определены условия сходимости градиентного алгоритма идентификации в среднем и средноквадратичном. Получены аналитические оценки неасимптотических и асимптотических значений ошибки оценивания параметров и точности идентификации. Показано, что эти значения ошибки оценивания и точности идентификации зависят от выбора параметра смешивания.

КОМБИНИРОВАННЫЙ КРИТЕРИЙ, ГРАДИЕНТНЫЙ АЛГОРИТМ, ПАРАМЕТР ВЗВЕШИВАНИЯ, РЕКУРРЕНТНАЯ ПРОЦЕДУРА, АСИМПТОТИЧЕСКАЯ ОЦЕНКА, ТОЧНОСТЬ ИДЕНТИФИКАЦИИ.

The problem of identifying the parameters of a linear object in the presence of non-Gaussian interference based on minimizing the combined functional that combines the properties of MLS and MSM is considered. The conditions for the convergence of the gradient identification algorithm in the mean and mean-square are determined. Analytical estimates of non-asymptotic and asymptotic values of the parameter estimation error and identification accuracy are obtained. It is shown that these values of estimation error and identification accuracy depend on the choice of the mixing parameter.

COMBINED CRITERION, GRADIENT ALGORITHM, WEIGHTS PARAMETER, RECURSIVE PROCEDURE, ASYMPTOTIC ESTIMATION, ACCURACY OF IDENTIFICATION.

Вступ

Задача ідентифікації не тільки становить інтерес сама по собі, але і є складовою частиною загальної задачі оптимізації. Багато задач управління, прогнозування, розпізнавання образів тощо пов'язані з побудовою моделі виду:

$$y(k) = \theta^{*T} x(k) + \xi(k), \quad (1)$$

де $y(k)$ – спостережуваний вихідний сигнал; $x(k) = (x_1(k), x_2(k), \dots, x_N(k))^T$ – вектор вхідних сигналів $N \times 1$; $\theta^* = (\theta_1^*, \theta_2^*, \dots, \theta_N^*)^T$ – вектор шуканих параметрів $N \times 1$; $\xi(k)$ – завада, і зводяться до мінімізації деякого наперед обраного функціоналу якості (критерію ідентифікації).

Найбільш широко використовуваний на практиці квадратичний функціонал призводить до різних алгоритмів ідентифікації, що дозволяє отримати оцінки шуканого вектора θ^* при нормальних розподілах завади, тобто $\xi(k) \sim N(0, \sigma_\xi^2)$.

Більшість існуючих в даний час методів ідентифікації засновано на використанні жорстких і важко перевіряємих умов, пов'язаних з гіпотезою нормальності закону розподілу завади і таких, що обґрунтовуються посиланнями на центральну граничну теорему. Як відомо, нормальним законом щільності розподілу описуються завади, присутні в вимірах, що проводяться при абсолютній стабільності умов вимірювання, законом Лапласа, що має довші «хвости» – завади, що виникають при максимальній нестабільності умов. Відповідно алгоритми ідентифікації в разі гаусовських завад засновані на методі найменших квадратів (МНК), а в разі завад, розподілених за законом Лапласа, – на методі найменших модулів (МНМ). Обидва ці методи є оптимальними в своїх умовах і рішення, які отримуються за їх допомогою, можуть сильно відрізнитися. Крім того, у зв'язку з тим, що на практиці ці крайні випадки реалізуються надзвичайно рідко, ні закон Гауса, ні закон Лапласа, як правило, не виконуються.

Якщо інформація про приналежність завади ξ деякому певному класу розподілів є відомою, то шляхом мінімізації оптимального критерію, що представляє собою узятий з оберненим знаком логарифм функції розподілу завади, може бути отримана оцінка максимальної правдоподібності (М-оцінка). Якщо ж такої інформації немає, то для оцінювання шуканого вектора параметрів θ слід застосувати будь-який неквадратичний критерій, що забезпечує робастність одержуваної оцінки. Одним з таких критеріїв є модульний критерій, що приводить до знакового алгоритму.

Застосування цього критерію в задачі ідентифікації об'єкта при наявності імпульсних завод розглядалося в [1-4]. Так в [1] вивчалася ефективність афінного проєкційного знакового алгоритму, в [2] застосовувався афінний проєкційний знаковий алгоритм зі змінним коефіцієнтом посилення. Необхідно відзначити, що знакові алгоритми, забезпечуючи робастність одержуваних оцінок, мають низьку швидкість збіжності. Тому з метою прискорення процесу оцінювання в [3] пропонувався і досліджувався нормалізований знаковий алгоритм ідентифікації. В [4] вивчається простий в реалізації алгоритм, який використовує для корекції розміру кроку середньоквадратичну помилку і розрахункову потужність перешкоди. Аналогічно в [5] розглянуто алгоритм Качмажа зі змінним коефіцієнтом посилення, що залежить від квадрата взаємної кореляції між квадратом вихідної помилки і виходом адаптивної моделі, і показана його ефективність при вирішенні деяких завдань шумозаглушення.

Існує досить велика кількість функціоналів, які забезпечують отримання робастних М-оцінок. Найбільш поширеними є комбіновані функціонали, запропоновані Хьюбером [6,7] і Хемпелем [8,9]. Вони складаються з квадратичного функціоналу, що забезпечує оптимальність оцінок для гаусівського розподілу, і модульного, що дозволяє отримати більш робастну до розподілів з важкими «хвостами» (викидами) оцінку.

Однак ефективність одержуваних робастних оцінок істотно залежить від численних параметрів, що використовуються в критеріях і які обираються на основі досвіду дослідника.

Іншим підходом до отримання робастних оцінок, позбавлених зазначеного недоліку, є використання комбінованого критерію.

Комбінований критерій, заснований на поєднанні квадратичного і модульного критеріїв, що забезпечує робастність одержуваних оцінок, був запропонований в [10]. У роботах [11-13] даний критерій застосовувався для розв'язання задачі ідентифікації за наявністю імпульсних завод.

В роботі [14] був запропонований критерій найменшого четвертого ступеня, властивості якого вивчалися в роботах [15-19].

Комбінований критерій оцінювання для прискорення процесу ідентифікації, що використовує поєднання квадратичного критерію і критерію четвертого ступеня, запропонованого і дослідженого в [20], було розвинуто в роботі [21], метою якої було прискорення процесу ідентифікації. Властивості адаптивного алгоритма мінімізації такого комбінованого критерія вивчалися в роботах [22-23].

Як свідчить аналіз робіт, присвячених робастній ідентифікації об'єктів керування, застосування комбінованого критерія є досить ефективним та значно простішим за використання традиційних критеріїв. У зв'язку з цим представляється вельми актуальним розробка підходу до робастного оцінювання параметрів з використанням деякого комбінованого функціоналу, що дозволяє поєднувати переваги МНК і МНМ.

1. Дослідження збіжності робастної процедури ідентифікації

Для забезпечення робастних властивостей одержуваних оцінок досить ефективним є застосування комбінованого функціоналу навчання [31,32]

$$F[e(k)] = \frac{1}{4} \lambda e^4(k) + (1 - \lambda) |e(k)|, \quad (2)$$

де $e(k) = y(k) - \hat{y}(k) + \xi(k) = y(k) - \hat{\theta}^T(k-1)x(k) + \xi(k)$, $\hat{y}(k)$ – вихідний сигнал моделі; $\hat{\theta}(k-1) = (\hat{\theta}_1(k-1), \hat{\theta}_2(k-1), \dots, \hat{\theta}_N(k-1))^T$ – вектор оцінюваних параметрів $N \times 1$; $\lambda \in [0,1]$ – параметр змішування.

При використанні комбінованого критерію (2) градієнтна процедура мінімізації має вигляд

$$\theta(k) = \theta(k-1) + \gamma(k) [\lambda e^3(k) + (1 - \lambda) \text{sign } e(k)] x(k), \quad (3)$$

де γ – деякий параметр, що впливає на швидкість збіжності алгоритму.

Дана процедура поєднує властивості МНК з властивостями МНМ, тому що при $\lambda = 1$ з (3) маємо алгоритм МНК, а при $\lambda = 0$ – алгоритм МНМ, і дозволяє боротися з негаусівськими заводами. Варіюючи параметр λ , можна змінювати властивості алгоритма.

Введемо в розгляд помилку оцінювання

$$\tilde{\theta}(k) = \theta(k) - \hat{\theta}(k), \quad (4)$$

що дозволяє записати вираз для $e(k)$ в такий спосіб:

$$e(k) = \tilde{\theta}^T(k-1)x(k) + \xi(k) = e_a(k) + \xi(k), \quad (5)$$

де $e_a(k) = \tilde{\theta}^T(k-1)x(k)$ - апіорна помилка ідентифікації.

У зв'язку з тим, що передбачається $\xi(k) \sim N(0, \sigma_\xi^2)$, маємо

$$M\{e^2(k)\} = \sigma_\xi^2 + \sigma_x^2 M\{\|\tilde{\theta}(k)\|^2\}, \quad (6)$$

де $M\{\bullet\}$ – символ математичного очікування; $\|\bullet\|$ – евклідова норма.

Розглянемо питання збіжності процедури (3) за відсутності завод. З цією метою введемо функцію Ляпунова $\|\tilde{\theta}(k)\|^2$.

Запишемо алгоритм (3) щодо помилок ідентифікації $\tilde{\theta}(i)$

$$\tilde{\theta}(k) = \tilde{\theta}(k-1) - \gamma [\lambda e^3(k) + (1-\lambda) \text{sign } e(k)] x(k). \quad (7)$$

Помноживши обидві частини (7) зліва на $\tilde{\theta}^T(k)$, з урахуванням того, що $e(k) = \tilde{\theta}^T(k-1)x(k)$, маємо

$$\begin{aligned} \|\tilde{\theta}(k)\|^2 &= \|\tilde{\theta}(k-1)\|^2 - 2\gamma\lambda e^4(k) - 2\gamma(1-\lambda)e(k)\text{sign } e(k) - \\ &- \gamma^2\lambda^2 e^6(k)\|x(k)\|^2 + 2\gamma^2\lambda(1-\lambda)e^3(k)\text{sign } e(k)\|x(k)\|^2 + \\ &+ \gamma^2(1-\lambda)^2\|x(k)\|^2. \end{aligned} \quad (8)$$

З (8) видно, що при $\gamma > 0$, приріст функції Ляпунова

$$\Delta V(k) = \|\tilde{\theta}(k)\|^2 - \|\tilde{\theta}(k-1)\|^2$$

буде негативним, якщо

$$\begin{aligned} &(\lambda e^4(k) + 2(1-\lambda)|e(k)|) > \\ &> \gamma(\lambda e^3(k) + (1-\lambda)|e(k)|)^2 \|x(k)\|^2. \end{aligned} \quad (9)$$

Таким чином, умова збіжності процедури (3) буде виконуватися, якщо параметр γ задовольняє нерівності

$$0 < \gamma < \frac{2e(k)}{(\lambda e^3(k) + (1-\lambda)\text{sign } e(k))\|x(k)\|^2}. \quad (10)$$

Оптимальне значення параметра γ визначаємо з рівняння, одержуваного шляхом диференціювання (8) по γ і прирівнювання похідної нулю. Таким чином,

$$\gamma^{\text{опт}} = \frac{e(k)}{(\lambda e^3(k) + (1-\lambda)\text{sign } e(k))\|x(k)\|^2}. \quad (11)$$

Дослідимо статистичні властивості процедури навчання (3) при наявності перешкод виміру, тобто

$$y^*(k) = \theta^{*T} x(k) + \xi(k), \quad \xi(k) \sim N(0, \sigma^2).$$

Припустимо, що завада не корельована з корисними сигналами. Записавши (3) щодо помилок навчання, маємо (7).

Розглянемо математичне сподівання $M\{\tilde{\theta}(k)\}$. Усреднюючи обидві частини (7), отримуємо

$$\begin{aligned} M\{\tilde{\theta}(k)\} &= M\{\tilde{\theta}(k-1) - 2\gamma\lambda(\tilde{\theta}^T(k-1)x(k) + \xi(k))x(k) - \\ &- \gamma(1-\lambda)\text{sign}(\tilde{\theta}^T(k-1)x(k))x(k)\}. \end{aligned} \quad (12)$$

Легко бачити, що

$$\begin{aligned} M\{\tilde{\theta}^T(k-1)x(k)x(k)\} &= M\{x(k)x^T(k)\tilde{\theta}(k-1)\} = \\ &= M\{x(k)x^T(k)\}M\{\tilde{\theta}(k-1)\} = R_{xx}M\{\tilde{\theta}(k-1)\}, \end{aligned} \quad (13)$$

де R_{xx} – кореляційна матриця вхідного сигналу. Детально розглянемо вираз

$$M\{x(k)\text{sign}(\tilde{\theta}^T(k-1)x(k))x(k)\}.$$

У випадку, якщо сигнал $x(k) \sim N(0, \sigma_x^2)$, маємо

$$\begin{aligned} &M\{x(k)\text{sign}(\tilde{\theta}^T(k-1)x(k))\} = \\ &M\{M\{x(k)\text{sign}(\tilde{\theta}^T(k-1)x(k))\}\} = \\ &= M\left\{\sqrt{\frac{2}{\pi}} \frac{1}{\sigma_e} M\{x(k)(\theta^T(k-1)x(k))\}\right\} = \\ &= M\left\{\sqrt{\frac{2}{\pi\sigma_e^2}} M\{x(k)x^T(k)\theta(k-1)\}\right\} = \\ &= \sqrt{\frac{2}{\pi\sigma_e^2}} M\{x(k)x^T(k)\theta(k-1)\} = \sqrt{\frac{2}{\pi\sigma_e^2}} R_{xx}M\{\theta(k-1)\}, \end{aligned} \quad (14)$$

де σ_e – середньоквадратичне значення помилки $e(k)$.

Вираз (9) отримано з використанням теореми Прайса [24], згідно з якою для двох випадкових гаусовських величин x та y з нульовими математичними сподіваннями справедливо

$$M\{x \text{sign } y\} = \sqrt{\frac{2}{\pi}} \frac{1}{\sigma_y} M\{xy\},$$

де σ_y – середньоквадратичне значення y .

З урахуванням властивостей завади

$$M\{\xi(k)x(k)\} = 0$$

і виразів (8), (9) маємо

$$M\{\tilde{\theta}(k)\} = \{I - 3\gamma\lambda\sigma_e^2 R_{xx} - \gamma(1-\lambda)\beta R_{xx}\}M\{\tilde{\theta}(k-1)\}, \quad (15)$$

звідки випливає, що процедура (3) буде збігатися в середньому, якщо параметр γ задовольняє нерівності

$$0 < \gamma < \frac{2}{(3\lambda\sigma_e^2 + (1-\lambda)\beta) \text{tr} R_{xx}}. \quad (16)$$

Тут $\beta = \sqrt{\frac{2}{\pi\sigma_e^2}}$; $\text{tr} R_{xx}$ – слід матриці R_{xx} .

З огляду на властивості вхідних сигналів, можемо записати умову (16) наступним чином:

$$0 < \gamma < \frac{2}{(3\sigma_e^2\lambda + (1-\lambda)\beta)N\sigma_x^2}. \quad (17)$$

Для дослідження збіжності алгоритму в середньоквадратичному розглянемо функцію Ляпунова $M\{\|\tilde{\theta}(k)\|^2\}$.

Помноживши обидві частини (7) зліва на $\tilde{\theta}^T(k)$, отримаємо

$$\begin{aligned} \|\tilde{\theta}(k)\|^2 &= \|\tilde{\theta}(k-1)\|^2 - 2\gamma\lambda e^3(k)(\tilde{\theta}^T(k-1)x(k)) - \\ &- 2\gamma(1-\lambda)\tilde{\theta}^T(k-1)x(k)\text{sign } e(k) + \gamma^2\lambda^2 e^6(k)\|x(k)\|^2 + \\ &+ 2\gamma^2\lambda(1-\lambda)e^3(k)\text{sign } e(k)\|x(k)\|^2 + \gamma^2(1-\lambda)^2\|x(k)\|^2. \end{aligned}$$

В даний вираз входять такі величини $e^3(k)$ та $e^6(k)$, які відповідно дорівнюють

$$e^3(k) = (e_a(k) + \xi(k))(e_a^2(k) + 2e_a(k)\xi(k) + \xi^2(k));$$

$$e^6(k) = (e_a(k) + \xi(k))^6 = e_a^6(k) + 5e_a^5(k)\xi(k) + 6e_a^4(k)\xi^2(k) + 4e_a^3(k)\xi^3(k) + 20e_a^2(k)\xi^4(k) + 15e_a(k)\xi^5(k) + \xi^6(k). \quad (18)$$

Взявши від обох частин (8) математичне сподівання, маємо

$$\begin{aligned}
 M\left\{\|\tilde{\theta}(k)\|^2\right\} &= M\left\{\|\tilde{\theta}(k-1)\|^2\right\} - 2\gamma\lambda M\left\{e^3(k)e_a(k)\right\} - \\
 &- 2\gamma(1-\lambda)M\left\{e_a(k)\text{sign}e(k)\right\} + \gamma^2\lambda^2 M\left\{e^6(k)\|x(k)\|^2\right\} + \\
 &+ 2\gamma^2\lambda(1-\lambda)M\left\{e^3(k)\text{sign}e(k)\|x(k)\|^2\right\} + \\
 &+ \gamma^2(1-\lambda)^2 M\left\{\|x(k)\|^2\right\}. \quad (19)
 \end{aligned}$$

Розглянемо кожний доданок в правій частині (19) з урахуванням (18) і статистичних властивостей сигналів і завод. У нашому випадку

$$\begin{aligned}
 M\left\{e^3(k)e_a(k)\right\} &= M\left\{\left(\tilde{\theta}^T(k-1)x(k)\right)^4\right\} \approx \\
 &\approx 3\sigma_x^4\left(M\left\{\|\tilde{\theta}(k-1)\|^2\right\}\right)^2; \\
 M\left\{\left(\tilde{\theta}^T(k-1)x(k)\right)^2\right\} M\left\{\xi^4(k)\right\} &\approx \\
 &\approx \sigma_x^2 M\left\{\xi^4(k)\right\} M\left\{\|\tilde{\theta}(k-1)\|^2\right\}; \\
 M\left\{\left(\tilde{\theta}^T(k-1)x(k)\right)^6\|x(k)\|^2\right\} &\approx \\
 &\approx (15N+90)\sigma_x^8\left(M\left\{\|\tilde{\theta}(k-1)\|^2\right\}\right)^3; \\
 M\left\{\left(\tilde{\theta}^T(k-1)x(k)\right)^4\|x(k)\|^2\right\} &= \\
 &= (3N+2)\sigma_x^4 M\left\{\xi^4(k)\right\} M\left\{\|\tilde{\theta}(k-1)\|^2\right\}; \\
 M\left\{\left(\tilde{\theta}^T(k-1)x(k)\right)^2\|x(k)\|^2\right\} &= \\
 &= (N+2)\sigma_x^4 M\left\{\xi^4(k)\right\} M\left\{\|\tilde{\theta}(k-1)\|^2\right\}; \\
 M\left\{\left(\tilde{\theta}^T(k-1)x(k)\right)^2\right\} &= \sigma_x^2 M\left\{\|\tilde{\theta}(k-1)\|^2\right\}; \\
 M\left\{\|x(k)\|^2 \xi^6\right\} &= M\left\{\|x(k)\|^2\right\} M\left\{\xi^6\right\} = N\sigma_x^2 M\left\{\xi^6\right\}; \\
 M\left\{\|x(k)\|^2\right\} &= N\sigma_x^2.
 \end{aligned}$$

Вираз $M\left\{\tilde{\theta}^T(k-1)x(k)\text{sign}\left(\tilde{\theta}^T(k-1)x(k)\right)\right\}$ для проаналізуємо за аналогією з (9)

$$\begin{aligned}
 &M\left\{\tilde{\theta}^T(k-1)x(k)\text{sign}\left(\tilde{\theta}^T(k-1)x(k)\right)\right\} = \\
 &= M\left\{M\left\{\tilde{\theta}^T(k-1)x(k)\text{sign}\left(\tilde{\theta}^T(k-1)x(k)\right)|\tilde{\theta}(k-1)\right\}\right\} = \\
 &= M\left\{\tilde{\theta}^T(k-1)\sqrt{\frac{2}{\pi\sigma_e}} M\left\{x(k)x^T(k)\tilde{\theta}(k-1)|\tilde{\theta}(k-1)\right\}\right\} \approx \\
 &\approx M\left\{\tilde{\theta}^T(k-1)\beta\tilde{\theta}(k-1)\right\} R_{xx} = \\
 &= \beta tr R_{xx} M\left\{\|\tilde{\theta}(k-1)\|^2\right\} = \beta\sigma_x^2 M\left\{\|\tilde{\theta}(k-1)\|^2\right\}. \quad (20)
 \end{aligned}$$

Аналогічно отримуємо

$$\begin{aligned}
 M\left\{\tilde{\theta}^T(k-1)x(k)\text{sign}\left(\tilde{\theta}^T(k-1)x(k)\right)\|x(k)\|^2\right\} &= \\
 &= 3\beta\sigma_x^4 M\left\{\|\tilde{\theta}(k-1)\|^2\right\}. \quad (21)
 \end{aligned}$$

При обчисленні $M\left\{\|\tilde{\theta}(k)\|^2\right\}$ враховано статистичні властивості завади, тобто

$$M(\xi(k)) = M(\xi^3(k)) = M(\xi^5(k)) = 0.$$

Підстановка отриманих виразів в (19) і нескладні перетворення дають

$$\begin{aligned}
 M\left\{\|\tilde{\theta}(k)\|^2\right\} &= \\
 &(1-6\gamma\lambda\sigma_x^2\sigma_\xi^2 + 15(N+2)\gamma^2\lambda^2\sigma_x^4 M\left\{\xi^4(k)\right\} - 2\gamma(1-\lambda)\beta\sigma_x^2) \times \\
 &\times M\left\{\|\tilde{\theta}(k)\|^2\right\} + \\
 &+ (15(3N+12)\gamma^2\lambda^2\sigma_x^6\sigma_\xi^2 + 6\gamma^2\lambda(1-\lambda)\sigma_x^4\beta - 6\gamma\lambda\sigma_x^4) \times \\
 &\times \left(M\left\{\|\tilde{\theta}(k-1)\|^2\right\}\right)^2 + \\
 &+ (15N+90)\gamma^2\lambda^2\sigma_x^8 \left(M\left\{\|\tilde{\theta}(k-1)\|^2\right\}\right)^3 + \\
 &+ \gamma^2\sigma_x^2 N \left(\lambda^2 M\left\{\xi^6(k)\right\} + (1-\lambda)^2\right). \quad (22)
 \end{aligned}$$

Якщо алгоритм збігається, величина $M\left\{\|\tilde{\theta}(k)\|^2\right\}$ буде малою. Тому для аналізу сталого стану вираз (22) можна спростити, нехтуючи величинами

$$\left(M\left\{\|\tilde{\theta}(k-1)\|^2\right\}\right)^2 \text{ та } \left(M\left\{\|\tilde{\theta}(k-1)\|^2\right\}\right)^3$$

і обмежуючись розглядом величини

$$\begin{aligned}
 M\left\{\|\tilde{\theta}(k)\|^2\right\} &= (1-6\gamma\lambda\sigma_x^2\sigma_\xi^2 + 15(N+2)\gamma^2\lambda^2\sigma_x^4 M\left\{\xi^4(k)\right\} - \\
 &- 2\gamma(1-\lambda)\beta\sigma_x^2) M\left\{\|\tilde{\theta}(k)\|^2\right\}. \quad (23)
 \end{aligned}$$

З (23) випливає, що процедура (3) буде збігатися в середньоквадратичному (приріст функції Ляпунова буде негативним) за виконання умови

$$M\left\{\|\tilde{\theta}(\infty)\|^2\right\} = \frac{\gamma N \left[\lambda^2 M\left\{\xi^6\right\} + (1-\lambda)^2\right]}{15\gamma(N+2)M\left\{\xi^4\right\} - 6\lambda\sigma_\xi^2 - 2(1-\lambda)\beta}, \quad (24)$$

тобто якщо параметр γ задовольняє нерівності

$$0 < \gamma < \frac{2(3\lambda\sigma_\xi^2 + (1-\lambda)\beta)}{15(N+2)\lambda^2\sigma_x^2 M\left\{\xi^4(k)\right\}}.$$

Оптимальне значення цього параметра, що забезпечує максимальну швидкість збіжності алгоритму, яке отримується шляхом вирішення рівняння

$$\frac{\partial M\left\{\|\tilde{\theta}(k)\|^2\right\}}{\partial \gamma} = 0,$$

матиме такий вигляд:

$$\gamma^{opt} = \frac{3\lambda\sigma_\xi^2 + (1-\lambda)\beta}{15(N+2)\lambda^2 M\left\{\xi^4(k)\right\}}.$$

Зі співвідношення (23) можна отримати вираз для асимптотичної помилки оцінювання

$$M\left\{\|\tilde{\theta}(\infty)\|^2\right\} = \frac{\gamma N \left[\lambda^2 M\left\{\xi^6\right\} + (1-\lambda)^2\right]}{15\gamma(N+2)M\left\{\xi^4\right\} - 6\lambda\sigma_\xi^2 - 2(1-\lambda)\beta}, \quad (25)$$

тобто для забезпечення $\lim_{k \rightarrow \infty} M \left\{ \|\tilde{\theta}(\infty)\|^2 \right\} = 0$ параметр γ повинен обиратися змінним і з ростом k прагнути до нуля, тобто задовольняти умовам Дворецького [25].

Підстановка (24) в (6) дає вираз для асимптотичної помилки ідентифікації

$$M \left\{ e^2(\infty) \right\} = \sigma_{\xi}^2 + \frac{\sigma_x^2 \gamma N \left[\lambda^2 M \left\{ \xi^6 \right\} + (1 - \lambda)^2 \right]}{15 \gamma (N + 2) M \left\{ \xi^4 \right\} - 6 \lambda \sigma_{\xi}^2 - 2(1 - \lambda) \beta}.$$

2. Моделювання

Для експериментального дослідження можливостей алгоритму (7) проводилася ідентифікація лінійного об'єкта, описуваного рівнянням (1) з

$$\theta^* = (1, 2, 4, 8, 16, 32, 64, 128, 256)^T.$$

При тестуванні робастності алгоритмів в вихідний сигнал об'єкта додавався незалежний шум с розподілом Релея ($\gamma = 1.0$). Гістограма такої завади показана на рис. 1. Результати моделювання представлені на рис. 2-3. На рис. 2 показано графік налаштування параметрів моделі, а на рис. 3 – помилки ідентифікації.

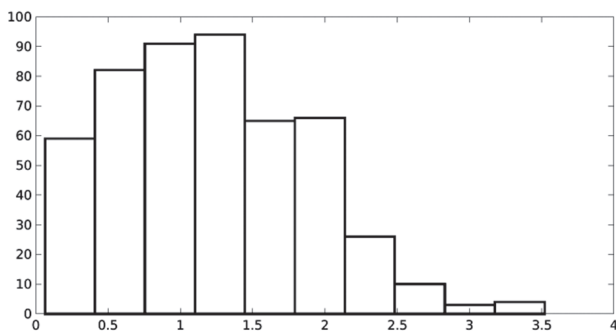


Рис. 1

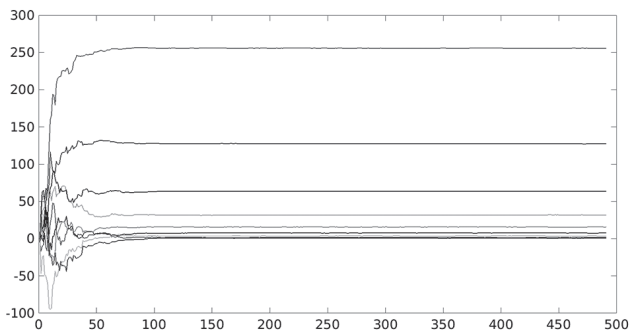


Рис. 2

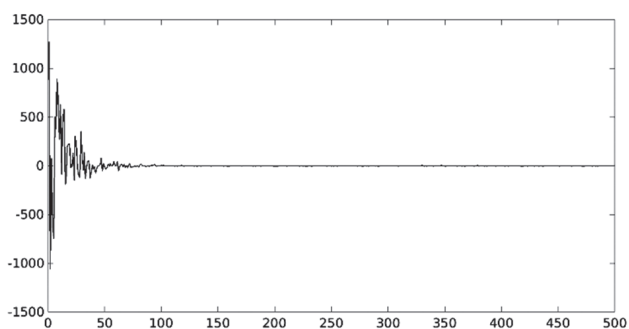


Рис. 3

Висновки

В роботі був розглянутий новий метод ідентифікації лінійних об'єктів за умов негаусівських завад із застосуванням комбінованого функціоналу, який поєднує властивості МНК та МНМ. Визначено умови збіжності градієнтного алгоритму ідентифікації в середньому і середньоквадратичному. Отримано аналітичні оцінки неасимптотичних та асимптотичних значень помилки оцінювання параметрів і точності ідентифікації. Показано, що ці значення помилки оцінювання і точності ідентифікації залежать від вибору параметра змішування. Отримані оцінки є досить загальними і залежать від статистичних характеристик корисних сигналів і завад. Тому для їх практичного застосування слід скористатися оцінками цих параметрів.

Слід зазначити, що отримані в даній роботі оцінки дозволяють досліднику при вирішенні практичних завдань попередньо оцінити можливості досліджуваного алгоритму і ефективність його застосування.

Список літератури:

- [1] Shao T. An affine projection sign algorithm robust against impulsive interferences / T. Shao, Y.R. Zheng, J. Benesty // IEEE Signal Process. Lett. -2010- 17(4).-P. 327–330
- [2] Shin J. Variable step-size affine projection sign algorithm / J. Shin, J. Yoo, P. Park // Electronics Lett.- 2012.- 48(9). - P. 483–485
- [3] Lu L. A novel normalized sign algorithm for system identification under impulsive noise interference / L. Lu, H. Zhao, K. Li, B. Chen //Int. J. of Electronics and Communications.- 2015.- 69(11).-P. 1590–1598.
- [4] Huang H.J. A new variable step-size NLMS algorithm and its performance analysis/ H. Huang, J. Lee // IEEE Trans. Signal Process. -2012.-60(4).-P. 2055–2060.
- [5] Casco-Sánchez F.M. A New Variable Step-Size NLMS Algorithm and its Performance Evaluation in Echo Cancelling Applications / F.M. Casco-Sánchez, R.C. Medina-Ramírez, M. López-Guerrero //J. of Applied Research and Technology.- 2011.-9.-3.-P. 302-313
- [6] Huber P. Robust methods of estimation of regression coefficients / P. Huber // Statistics.- 1977.- 8.-1.-P. 41–53.
- [7] Хьюбер П. Робастність в статистиці / П. Хьюбер – М.: Мир, 1984. – 304 с.
- [8] Hampel F.R. The influence curve and its role in robust estimation / F.R. Hampel // J. Amer. Statist. Assoc. – 1974. – June. – 69. – P. 383-393.
- [9] Hampel F.R. Robust Statistics. The Approach Based on Influence Function. / F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, W.A. Stahel.— N.Y.: John Wiley and Sons, 1986. – 526 p.
- [10] Chambers J. A Robust Mixed-Norm Adaptive Filter / J. Chambers, A. Avlonitis // IEEE Signal Processing Letters. – 1997. – 4. – 2. –P. 46–48.
- [11] Papoulis E.V. A Normalized Robust Mixed-Norm Adaptive Algorithm for System Identification / E.V. Papoulis, T. Stathaki // IEEE Signal Processing Letters.- 2004. – 11. – 1. –P. 56–59

- [12] *Arenas-Garcia J.* Adaptive combination of normalised filters for robust system identification. / J. Arenas-Garcia, A.R. Figueiras-Vidal // *Electronics Lett.* – 2005. – 41(15). – P. 874–875.
- [13] *Rudenko O.* Robust identification of non-stationary objects with nongaussian interference / O. Rudenko, O. Bezsonov, V. Lebediev, N. Serdiuk // *Eastern-European Journal of Enterprise Technologies*, 2019. – № 5/4 (101). – P.44–52.
- [14] *Walach, E.* The least mean fourth (LMF) adaptive algorithm and its family / E. Walach, D. Widrow // *IEEE Trans.* – 1984. – IT–30. – P. 275–283
- [15] *Bershad N.* Mean-square stability of the normalized least mean fourth algorithm for white Gaussian Inputs / N. Bershad, J.C.M. Bermudez // *Digit. Signal Process.* – 2011. – 21(6). – P.694–700.
- [16] *Eweda E.* New insights into the normalization of the least mean fourth algorithm / E. Eweda, A. Zerguine // *Signal Image Video Process.* – 2013. – 7(2). – P.255–262..
- [17] *Eweda E.* Global stabilization of the least mean fourth algorithm / E. Eweda // *IEEE Trans. Signal Process.* – 2012. – 60(3). – P. 1473–1477.
- [18] *Eweda E.* Stochastic analysis of a stable normalized least mean fourth algorithm for adaptive noise canceling with a white Gaussian reference / E. Eweda, N. Bershad // *IEEE Trans. Signal Process.* – 2012. – 60(12). P. 6235–6244.
- [19] *Hübscher P.I.* A mean-square stability analysis of the least mean fourth adaptive algorithm / P.I. Hübscher, J.C.M. Bermudez, V.H. Nascimento // *IEEE Trans. Signal Process.* – 2007. – 55(8). – P. 4018–4028
- [20] *Chambers J.* Least mean mixed-norm adaptive filtering / J. Chambers, O. Tanrikulu, A.G. Constantinides // *Electronics letters.* – 1994. – 30. – 19. – P. 1574–1575.
- [21] *Rakesh P.* Modified least-mean mixed-norm algorithms for adaptive spars system identification under impulsive noise environment / P. Rakesh, T.K. Kumar, F. Albu // 42 Int. Conf. on Telecommunications and Signal Processing (TSP), Budapesht, July, 2019. – 1. – P. 557–561
- [22] *Zerguine A.* A variable-parameter normalized mixed-norm (VPNMN) adaptive algorithm / A. Zerguine // *EURASIP Journal on Advances in Signal Processing* 2012, 2012:55.- 13 p. <http://asp.eurasipjournals.com/content/2012/1/55>
- [23] *Zerguine A.* A time-varying normalized mixed-norm LMS-LMF algorithm / A. Zerguine // 11 th European Signal Proc. Conf., Toulouse, France CNUM, 3-6 Sep 2002. – 4 p. <https://www.eurasip.org/Proceedings/Eusipco/2002/articles/paper394.pdf>
- [24] *Price R.* A useful theorem for nonlinear devices having Gaussian inputs / R. Price // *IREN Trans. Inform Theory.* – 1958. – 4. – P. 69–72. doi: 10.1109/TIT.1958.1057444
- [25] Вазан М. Стохастическая аппроксимация / М. Вазан. – М.: Мир, 1972. – 295 с.

Надійшла до редколегії 20.09.2019

УДК 004.681

DOI 10.30837/bi.2019.2(93).05

С.Г. Удовенко¹, Д.В. Келембет², В.А. Затхей³

¹ Доктор технічних наук, завідувач кафедри інформатики та комп'ютерної техніки,
Харківський національний економічний університет ім. С. Кузнеця,
serhiy.udovenko@hneu.net; ORCID id: 0000-0001-5945-8647

² Аспірант кафедри електронних обчислювальних машин,
Харківський національний університет радіоелектроніки,
kelembet@gmail.com; ORCID id: 0000-0002-5853-4166

³ Кандидат технічних наук, доцент кафедри інформатики та комп'ютерної техніки,
Харківський національний економічний університет ім. С. Кузнеця,
zathey_va@ukr.net; ORCID id: 0000-0003-4426-7789

КЕРУВАННЯ НЕЛІНІЙНИМИ ТЕХНОЛОГІЧНИМИ ОБ'ЄКТАМИ З ВИКОРИСТАННЯМ НЕЧІТКИХ PDC-РЕГУЛЯТОРІВ

Досліджено особливості побудови алгоритмів PDC - керування нелінійними об'єктами з використанням дискретних локальних ТС-моделей. Запропоновано схеми PDC – регуляторів з компенсацією статистичних помилок за рахунок введення в структуру системи керування цифрових інтеграторів і наступного обчислення коефіцієнтів регулятора для кожної з афінних підмоделей. Наведено результати моделювання розглянутої схеми для керування нелінійними технологічними процесами. Визначено перспективи використання запропонованого підходу.

PDC-КЕРУВАННЯ, НЕЧІТКИЙ РЕГУЛЯТОР, КОМПЕНСАЦІЯ СТАТИЧНИХ ПОМИЛОК

Исследованы особенности построения алгоритмов PDC - управления нелинейными объектами с использованием дискретных локальных ТС-моделей. Предложены схемы PDC - регуляторов с компенсацией статистических ошибок за счет введения в структуру системы управления цифровых интеграторов и последующего вычисления коэффициентов регулятора для каждой из аффинных подмоделей. Приведены результаты моделирования рассматриваемой схемы для управления нелинейными процессами. Определены перспективы использования предложенного подхода.

PDC-УПРАВЛЕНИЕ, НЕЧЕТКИЙ РЕГУЛЯТОР, КОМПЕНСАЦИЯ СТАТИЧЕСКИХ ОШИБОК

The features of the construction of PDC algorithms - control of nonlinear objects using discrete local TS-models are investigated. Schemes of PDC - controllers with compensation of static errors due to the introduction of digital integrators into the control system structure and the subsequent calculation of controller coefficients for each of the affine submodels are proposed. The simulation results of the considered scheme for controlling nonlinear processes are presented. The prospects of using the proposed approach are determined.

PDC-CONTROL, FUZZY REGULATOR, STATIC ERROR COMPENSATION

Вступ

Вирішення проблеми створення цифрових регуляторів для складних багатопараметричних технологічних об'єктів (в енергетиці, хімічній технології тощо) може бути засноване на описі динаміки цих об'єктів нечіткими моделями, в яких залежності між вхідними, вихідними змінними і змінними станів представляються за допомогою нечітких правил [1]. Завдання нечіткої ідентифікації та нечіткого керування пов'язані з необхідністю застосування різних методів апроксимації нечітких систем [2, 3].

Такі методи можуть бути, зокрема, засновані на застосуванні нечітких моделей типу Такагі-Сугено (ТС [4]). У порівнянні з іншими підходами до моделювання керування процесів нечітке моделювання типу ТС не вимагає попередніх знань рівнянь динаміки системи, а ґрунтується тільки на реєстрованих даних. Декомпозиція нелінійної динаміки системи за допомогою локальних підмоделей може служити альтернативою вирішення проблем моделювання та управління для нелінійних об'єктів відносно простим

способом. Після моделювання здійснюється синтез нечіткого ТС- керування на основі отриманої моделі.

Нечітке ТС-керування отримало розвиток в системах керування нелінійними об'єктами різного функціонального призначення, в тому числі мобільними роботами та неперервними технологічними агрегатами [5, 6]. Зокрема, перспективним є використання нечітких регуляторів з ТС-моделлю для керування газоперекачувальними агрегатами (ГПА) компресорних станцій газотранспортної системи [7].

У контурах керування з ТС-моделями широко застосовуються нечіткі регулятори типу PDC (регулятори з паралельною розподіленою компенсацією – parallel distributed compensation) і квадратичний синтез [8]. Зокрема, інтерес представляє підхід до субоптимального лінійного квадратичного керування зі зворотним зв'язком за станом, заснований на використанні афінної ТС-моделі в дискретному часі [9].

У даній роботі досліджується завдання розробки алгоритмів PDC-керування нелінійними об'єктами керування (ОК) з використанням дискретних

локальних ТС-моделей. Зокрема, одним з таких завдань є побудова схем PDC-регуляторів з компенсацією статичних помилок за рахунок введення в структуру системи керування цифрових інтеграторів і наступного обчислення коефіцієнтів регулятора для кожної з афінних підмоделей. Прикладним застосуванням таких регуляторів може бути оптимізація технологічних режимів роботи ГПА.

1. Принцип побудови субоптимальних нечітких PDC-регуляторів

Концепція PDC-керування передбачає побудову лінійного закону керування зі зворотним зв'язком за станом для кожної локальної моделі нечіткої системи. Синтез закону керування полягає у визначенні для кожної з локальних моделей матриць коефіцієнтів, наприклад, використовуючи LMI-оптимізацію (за допомогою лінійних матричних нерівностей (linear matrix inequalities – LMI)) або оптимізацію за квадратичним критерієм з використанням лінійних квадратичних регуляторів (LQR) [10]. У даній роботі основна увага приділена другому підходу. Мінімізація квадратичного критерію є одним з методів, зручним для визначення структури керування зі зворотним зв'язком за станом для лінійних багатовимірних систем. Використовуючи цей підхід, можна поширити класичні результати оптимального керування для локальних моделей загальної афінної ТС-моделі.

Як вже було зазначено, застосування нечітких ТС-моделей є одним з напрямків рішення загальної проблеми моделювання нелінійної системи з застосуванням декомпозиції загальної моделі на сукупність більш простих (локальних) лінійних моделей. Такий підхід дає ефективні засоби представлення невизначеності, пов'язаної із завданням декомпозиції, що дозволяють врахувати зв'язки між лінійними підсистемами і об'єднати знання різних типів. В цьому випадку нечітку ідентифікацію можна розглядати як задачу декомпозиції системи.

Розглянемо метод субоптимального PDC-керування зі зворотним зв'язком за станом, де закон керування визначається тільки для кожної з лінійних підмоделей. На рис. 1 наведено загальну схему нечіткого PDC-керування.

Дослідимо особливості побудови законів LQR-керування для дискретних афінних ТС-моделей, що становлять основу PDC-схеми, наведеної на рис. 1.

Розглянемо модель в просторі станів, яка відповідає нелінійній системі в дискретному часі наступного виду (для спрощення позначення $X(k)$ замінено на x_k):

$$x_{k+1} = f_k(x_k, u_k), \quad (1)$$

де $k = 0, 1, \dots, N$ – дискретний час; x_k – вектор стану в момент k ; u_k – вектор керування в момент k ; x_0 – початкові умови.

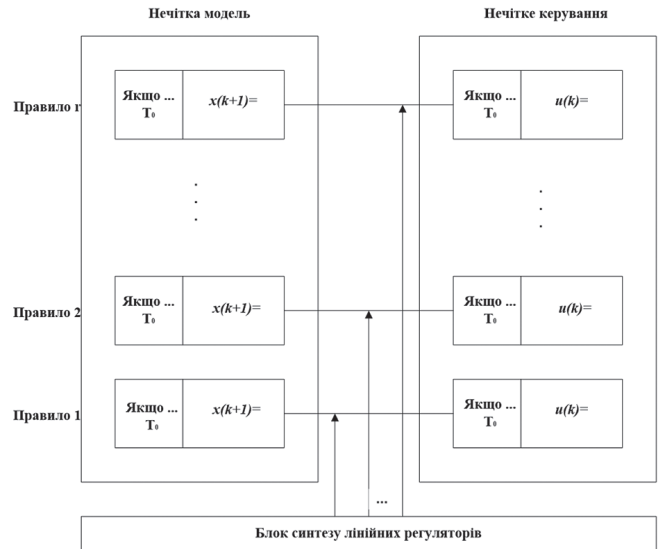


Рис. 1. Загальна схема нечіткого PDC-керування

Припустимо, що система (1) апроксимується нечіткою афінною ТС-моделлю, що складається з логічно пов'язаних дискретних лінійних підмоделей виду:

$$x_{k+1} = A_k x_k + B_k u_k + d_k; \quad (2)$$

$$y_k = C_k x_k,$$

де u_k – вектор виходу; A_k, B_k, C_k – матричні коефіцієнти; d_k – векторний коефіцієнт афінної моделі.

Визначення оптимального керування по лінійним підмоделям з мінімізацією квадратичного критерію може здійснюватися за допомогою різних підходів (наприклад, динамічне програмування і принцип максимуму Понтрягіна, засновані на варіаційних методах [19]). Далі ми приділимо увагу другому підходу. У цьому контексті важливим завданням синтезу LQR-регулятора є знаходження закону керування, що дозволяє для лінійної афінної підсистеми, що описується рівнянням (2) з початковим станом, мінімізувати наступний квадратичний критерій

$$J = 0.5 x_N^T S_N x_N + 0.5 \sum_{k=0}^{N-1} (x_k^T Q_k x_k + u_k^T R_k u_k), \quad (3)$$

де S_k, Q_k – симетричні позитивно напіввизначені матриці; R_k – симетрична позитивно визначена матриця.

Матриця S_k зважає кінцевий стан, Q_k – еволюцію станів і R_k – енергію керування. В процесі оптимізації критерій має сходиться, а підсистема (2) повинна бути стійкою. Метою оптимізації є отримання послідовності керувань u_0, u_1, \dots, u_{N-1} , яка мінімізує (3). В цьому випадку гамільтоніан задається таким чином:

$$H(x_k, u_k, \lambda_{k+1}) = \frac{1}{2} (x_k^T Q_k x_k + u_k^T R_k u_k) + \lambda_{k+1}^T (A_k x_k + B_k u_k + d_k). \quad (4)$$

Відповідно до теорії оптимального керування, заснованої на варіаційному численні, розглянемо такі рівняння:

Рівняння станів:

$$x_{k+1} = \frac{\partial H}{\partial \lambda_{k+1}} = A_k x_k + B_k u_k + d_k; \quad (5)$$

Рівняння коефіцієнтів:

$$\lambda_k^* = \frac{\partial H}{\partial x_k} = Q_k x_k^* + A_k^T \lambda_{k+1}^*; \quad (6)$$

Рівняння стаціонарності:

$$\frac{\partial H}{\partial u_k} = R_k u_k^* + B_k^T \lambda_{k+1}^*. \quad (7)$$

Згідно з (7), закон оптимального керування в розімкнутому контурі набуде вигляду:

$$u_k^* = -R_k^{-1} B_k^T \lambda_{k+1}^*. \quad (8)$$

Відзначимо, що можливість інверсії матриці R_k обумовлюється її позитивною визначеністю. Підставляючи u_k^* в рівняння станів (5), отримуємо:

$$x_{k+1}^* = A_k x_k^* - B_k R_k^{-1} B_k^T \lambda_{k+1}^* + d_k. \quad (9)$$

Для випадку, коли кінцевий стан є вільним, відповідна умова контуру визначається наступним чином:

$$\lambda_N = \frac{\partial}{\partial x_N} (0.5 x_N^T S_N x_N) = S_N x_N. \quad (10)$$

Для отримання закону оптимального керування в замкнутому контурі необхідно представити функцію λ_{k+1}^* в оптимальному керуванні (8) термами функції станів. Кінцева умова при цьому характеризується присутністю афінного терма в рівнянні стану (5), який будемо задавати наступним чином:

$$\lambda_k^* = P_k x_k^* + q_k, \quad (11)$$

де матриця P_k і вектор q_k мають бути заздалегідь визначені.

Підставляючи (11) в рівняння станів (9), з урахуванням (6) отримуємо:

$$P_k x_k^* + q_k = Q_k x_k^* + A_k^T (P_{k+1} x_{k+1}^* + q_{k+1}); \quad (12)$$

$$x_{k+1}^* = A_k x_k^* - B_k R_k^{-1} B_k^T (P_{k+1} x_{k+1}^* + q_{k+1}) + d_k. \quad (13)$$

З (13) випливає, що:

$$x_{k+1}^* = \left(I + B_k R_k^{-1} B_k^T P_{k+1} \right)^{-1} \times \left(A_k x_k^* - B_k R_k^{-1} B_k^T q_{k+1} + d_k \right), \quad (14)$$

де I – одинична матриця.

Підставляючи (14) в (12), отримуємо:

$$P_k x_k^* + q_k = Q_k x_k^* + A_k^T P_{k+1} \times \left(\left(I + B_k R_k^{-1} B_k^T P_{k+1} \right)^{-1} \left(A_k x_k^* - B_k R_k^{-1} B_k^T q_{k+1} + d_k \right) \right) + A_k^T q_{k+1}. \quad (15)$$

Здійснимо верифікацію рівняння (15) (для всіх x_k) для двох наступних відносин:

$$P_k = Q_k + A_k^T P_{k+1} \left(I + B_k R_k^{-1} B_k^T P_{k+1} \right)^{-1} A_k; \quad (16)$$

$$q_k = A_k^T P_{k+1} \left(I + B_k R_k^{-1} B_k^T P_{k+1} \right)^{-1} \left(A_k^T q_{k+1} + d_k \right). \quad (17)$$

Вираз (16) відповідає матричному різницевою рівнянню Ріккати (DRE – Difference Riccati Equation).

Кінцева умова рішення (16) отримується відповідно до (10) і (11) у вигляді:

$$P_N = S_N; \quad q_N = 0. \quad (18)$$

У рівнянні (16) терм P_{k+1} знаходиться в правій частині, а $-P_k$ в лівій частині, що вимагає реалізації ітеративного рішення зі зростанням індексу часу з урахуванням фінальної умови (18). Аналогічна процедура використовується і для розрахунку q_k . Оскільки рекурсивні відносини визначені в інверсному часі, то практична реалізація керування, що здійснюється за виразами (8) і (11), вимагає перерахунку матриці P_k і вектора q_k для всіх моментів часу розглянутого горизонту. Це викликає необхідність проведення обчислень і запам'ятовування всіх результатів перед кожною зміною задавального впливу. Очевидно, що це є істотним обмеженням для використання комп'ютерів в контурі керування.

Таким чином, доцільно для більшості практичних технічних додатків розглядати нескінченний горизонт оптимізації (або, принаймні, дуже великий по відношенню до шкали часу даних фізичних процесів). Критерій мінімізації при цьому матиме такий вигляд:

$$J = 0.5 \sum_{k=0}^{\infty} \left(x_k^T Q_k x_k + u_k^T R_k u_k \right). \quad (19)$$

У разі стаціонарної системи ($A_k = A$, $B_k = B$, $d_k = d$, $C_k = C$) і при виконанні вимог до матриць Q та R можна показати, що рекурентні рівняння (16) і (17) сходяться до єдиних обмежених рішень P та q відповідно, якщо підмодель є керованою. В цьому випадку отримуємо:

$$P = Q + A^T P \left(I + B R^{-1} B^T P \right)^{-1} A. \quad (20)$$

Використовуючи лему про матричну інверсію:

$$\left(M_1 + M_2 M_3 M_4 \right)^{-1} M_1^{-1} M_1^{-1} M_2 \times \left(M_3^{-1} + M_4 M_1^{-1} M_2 \right)^{-1} M_4 M_1^{-1}, \quad (21)$$

з $M_1 = I$; $M_2 = B$; $M_3 = R^{-1}$; $M_4 = B^T P$ отримуємо алгебраїчне дискретне рівняння Ріккати (DARE – Discrete Algebraic Riccati Equation):

$$A^T P A - P - \left(A^T P B \right) \left(R + B^T P B \right)^{-1} \left(B^T P A \right) + Q = 0. \quad (22)$$

Для вектора q , відповідно до (17), після рішення рівняння (22) отримуємо:

$$q = \left(I + A^T P \left(I + B R^{-1} B^T P \right)^{-1} B R^{-1} A^T \right)^{-1} \times \left(A^T P \left(I + B R^{-1} B^T P \right)^{-1} \right). \quad (23)$$

Це рівняння можна представити у вигляді:

$$q = (G + FE)^{-1} Fd, \quad (24)$$

де $E = B R^{-1} B^T$; $F = A^T P (I + EP)^{-1}$; $G = I - A^T$.

Згідно з оптимальним керуванням (8) і рівнянням (11) отримуємо:

$$u_k^* = -R^{-1}B^T(Px_{k+1}^* + q) = -R^{-1}B^T(P(Ax_k^* + Bu_k^* + d) + (G + FE)^{-1}Fd), \quad (25)$$

що призводить до наступного оптимального керування у замкнутому контурі:

$$u_k^* = -(Kx_k + Nd), \quad (26)$$

де

$$K = (R + B^T PB)^{-1} B^T PA;$$

$$N = (R + B^T PB)^{-1} B^T (P + (G + FE)^{-1} F).$$

З огляду на, що нечіткий регулятор PDC використовує ті ж нечіткі набори, що і модель Такагі-Сугено, в ньому застосовуються ті ж функції належності. Для нечіткої афінної дискретної моделі ТС реалізація нечіткого регулятора PDC полягає у визначенні для $i=1,2,\dots,r$ правил моделі, коефіцієнтів зворотного зв'язку K_i і складових в частині висновків N_i відповідно до (26), використовуючи такі правила R^i :

$$\text{«Якщо } z_1(k) \in M_1^i \text{ і } \dots \text{ і } z_p(k) \in M_p^i, \quad (27)$$

то: $u(k) = -(K_i x(k) + N_i d)$ », де p – кількість правил.

Вихід нечіткого регулятора формується таким чином:

$$u(k) = -\sum_{i=1}^r h_i(z(k))(K_i x(k) + N_i d), \quad (28)$$

де

$$h_i(z(k)) = \frac{w_i(z(k))}{\sum_{i=1}^r w_i(z(k))}.$$

Для всіх k мають виконуватися такі умови активації функцій належності:

$$\sum_{i=1}^r h_i(z(k)) = 1; w_i(z(k)) \geq 0. \quad (29)$$

Загальна схема керування з регулятором типу PDC для лінійних афінних підмоделей наведена на рис. 2.

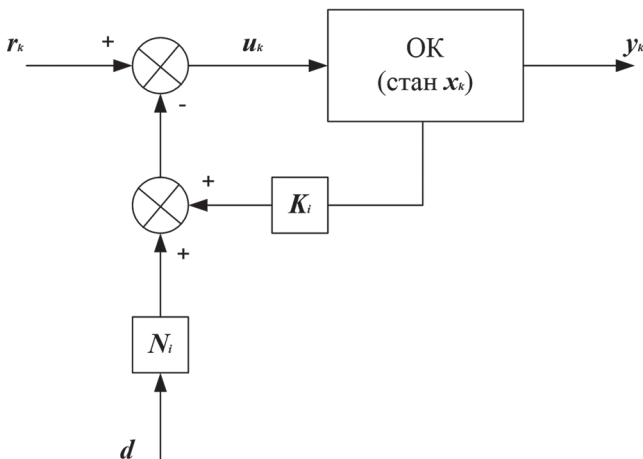


Рис. 2. Схема нечіткого PDC-керування за станом для лінійних підмоделей

З огляду на властивості матриць S , Q та R можна стверджувати, що рішення рівняння Ріккати є симетричним.

Треба також зазначити, що вираз (16) є узагальненням алгоритму оптимального керування за

квадратичним критерієм (LQR) для лінійних афінних систем виду (2). Дійсно, в разі $d=0$ отримуємо стандартне рівняння алгебри Ріккати для дискретного часу.

2. Системи PDC-керування з компенсацією статичних помилок

Отримані результати дозволяють розрахувати закон оптимального керування для кожної з локальних афінних моделей ТС. Глобальний закон керування типу PDC можна отримати шляхом дефазифікації нечіткої моделі. В цьому випадку не завжди можна гарантувати оптимальність результатів глобального керування. До сих пір розглядався загальний випадок керування для лінійних афінних моделей незалежно від типу регулятора. Недоліком багатьох структур керування є наявність статичних помилок у разі ненульових задавальних вхідних значень. Це дуже небажано для практичних застосувань, для яких задавальні впливи мають ефективно відпрацьовуватися. Розглянемо в першу чергу випадок компенсації помилок шляхом відповідного вибору статичного коефіцієнта і введенням в систему цифрових інтеграторів. Розглянемо алгоритм керування, що дозволяє компенсувати статичні помилки при постійних значеннях збурюючих сигналів, який передбачає використання відповідного вибору статичних коефіцієнтів для підмоделей.

З урахуванням оптимального керування (26), додаючи задавальний вплив, отримуємо:

$$u_k = Lr_k - (Kx_k + Nd), \quad (30)$$

де r_k – вхідний задавальний сигнал; L – матриця компенсуючих коефіцієнтів; N – матриця, яку асоціюють з вектором взаємозв'язків d .

Підставляючи (30) в рівняння динаміки (2), а потім визначаючи Z -перетворення для нульових початкових умов, отримуємо наступне операторне рівняння для виходу підмоделі:

$$Y(z) = C(zI - A + BK)^{-1} BLr(z) + Cz^{-1}d. \quad (31)$$

Очевидно, що для отримання нульової статичної помилки при постійному значенні задавального сигналу треба компенсувати складову, пов'язану з впливом d . Для цього додамо матричну складову V в рівняння керування (30):

$$u_k = Lr_k - (Kx_k + (N + V)d). \quad (32)$$

У цьому випадку застосування Z -перетворення для нульових початкових умов призводить до наступного операторного рівняння для виходу підмоделі:

$$Y(z) = C(zI - A + BK)^{-1} (BLr(z) - (BN + BV - I)d). \quad (33)$$

Для постійного значення $r_k = r$ важливим завданням керування є отримання нульової статичної помилки:

$$\lim_{t \rightarrow \infty} y_k = \lim_{z \rightarrow 1} (z-1)Y(z) = r. \quad (34)$$

Відповідно до (33) та (34) отримуємо:

$$\lim_{t \rightarrow \infty} y_k = r = \lim_{z \rightarrow 1} (z-1)C(zI - A + BK)^{-1} \left(BL \frac{rz}{z-1} - \left(BN + BV - I \frac{dz}{z-1} \right) \right), \quad (35)$$

звідки випливає, що:

$$r = C(I - A + BK)^{-1} (BLr - (BN + BV - I)d). \quad (36)$$

Для отримання нульової статичної помилки еволюція (36) вимагає виконання наступних умов:

$$L = (MB)^{-1}; \quad M(BN + BV - I)d = 0, \quad (37)$$

де $M = C(I - A + BK)^{-1}$.

З урахуванням розмірностей матриць A , B , C , K матриця MB може бути інвертована лише за умови, що підмоделі мають однакову розмірність входів та виходів, тобто $p = m$, і що результуюча матриця не є особливою. Оскільки розглянута структура забезпечує стійкість контуру підмоделі, то коректність цієї умови передбачає можливість виконання обчислень з використанням матричних рівнянь (37).

З другої умови (37) випливає, що:

$$V = LM - N. \quad (38)$$

Підставляючи (38) в (32), отримуємо наступну модифікацію закону керування:

$$u_k = Lr_k - (Kx_k + Td), \quad (39)$$

де $T = LM = (MB)^{-1}M$.

Для нечіткої афінної дискретної ТС-моделі синтез нечіткого PDC-регулятора з компенсуючим статичним коефіцієнтом полягає у визначенні правил моделі; коефіцієнтів компенсації; коефіцієнтів зворотного зв'язку і коефіцієнтів для подальшого формування висновків відповідно до (39), використовуючи правила такого вигляду:

$$\text{«Якщо } z_1(k) \in M_1^i \text{ і } \dots \text{ і } z_p(k) \in M_p^i, \quad (40)$$

$$\text{то: } u(k) = L_i r(k) - (K_i x(k) + T_i d)\text{»},$$

де p – кількість правил.

Таким чином, остаточний закон керування, що формується досліджуванним нечітким регулятором, має наступний вигляд:

$$u(k) = \sum_{i=1}^r h_i(z(k)) (L_i r(k) - (K_i x(k) + T_i d)), \quad (41)$$

$$\text{де } h_i(z(k)) = \frac{w_i(z(k))}{\sum_{i=1}^r w_i(z(k))}.$$

З урахуванням рівняння (41) загальна структура розподіленого PDC-керування з компенсуючим статичним коефіцієнтом для лінійних афінних підмоделей може бути представлена схемою, що наведено на рис. 3.

Рівняння (38) дозволяє розраховувати закон керування, що компенсує статичні помилки тільки для постійних значень задавальних сигналів. Однак при

наявності параметричних змін у ТС-моделі виникає необхідність компенсації помилок для змінних значень задаючих сигналів, що вимагає ускладнення структури керування.

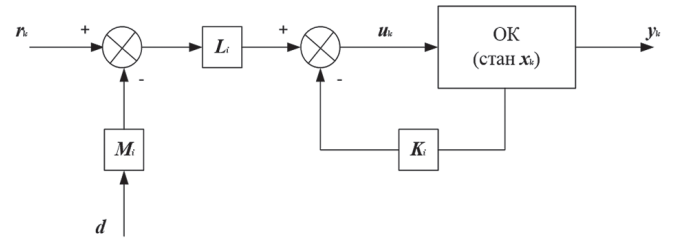


Рис. 3. Схема PDC-керування з компенсуючим статичним коефіцієнтом для лінійних нечітких підсистем

Підвищити ефективність (з точки зору компенсації помилок для змінюваних значень задавальних сигналів) PDC-керування можна введенням в структуру системи цифрових інтеграторів і наступним обчисленням коефіцієнтів регулятора для кожної з афінних підмоделей. Використовуючи для цифрового інтегрування метод прямокутників, отримуємо рівняння такого вигляду:

$$v_{k+1} = v_k + T_e \cdot e_k = v_k + T_e \cdot (r_k - y_k) \quad (42)$$

де T_e – період дискретизації; e_k – вектор помилок між задавальним сигналом r_k і вимірюванням виходу y_k .

Якщо система керування містить p входів і m виходів, то в разі $p \leq m$ кожен додатковий вихід будемо асоціювати з одним інтегратором, щоб число задавальних впливів відповідало числу вимірюваних виходів. З урахуванням вектора v_k з (42) для кожної з підмоделей отримуємо розширені рівняння стану:

$$\begin{bmatrix} x_{k+1} \\ v_{k+1} \end{bmatrix} = \begin{bmatrix} A_{[n \times n]} & 0_{[n \times p]} \\ \bar{C}_{[p \times n]} & I_{[p \times p]} \end{bmatrix} \begin{bmatrix} x_k \\ v_k \end{bmatrix} + \begin{bmatrix} B_{[n \times m]} \\ 0_{[p \times m]} \end{bmatrix} u_k + \begin{bmatrix} d \\ \bar{r}_k \end{bmatrix};$$

$$[y_k] = \begin{bmatrix} C_{[p \times n]} & 0_{[p \times p]} \end{bmatrix} \begin{bmatrix} x_k \\ v_k \end{bmatrix}, \quad (43)$$

$$\text{де } \bar{C} = -T_e \cdot C; \quad \bar{r}_k = T_e \cdot r_k.$$

У рівняннях (43) індекси матриць відповідають їх розмірностям. Представимо ці рівняння в наступному вигляді:

$$\tilde{x}_{k+1} = \tilde{A} \tilde{x}_k + \tilde{B} u_k + \tilde{d}; \quad \tilde{y}_k = \tilde{C} \tilde{x}_k, \quad (44)$$

де матриці \tilde{A} , \tilde{B} , \tilde{d} , \tilde{C} та вектори \tilde{x}_k , \tilde{y}_k мають таку структуру:

$$\left\{ \begin{array}{l} \tilde{A} = \begin{bmatrix} A & 0 \\ \bar{C} & I \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad \tilde{d} = \begin{bmatrix} d \\ \bar{r}_k \end{bmatrix}, \quad \tilde{C} = [C \quad 0] \\ \tilde{x}_k = \begin{bmatrix} x_k \\ v_k \end{bmatrix}, \quad \tilde{y}_k = [y_k] \end{array} \right.$$

Завдання керування полягає в мінімізації квадратичного функціоналу такого вигляду:

$$\tilde{J} = 0.5 \sum_{k=0}^{\infty} (\tilde{x}_k^T \tilde{Q} \tilde{x}_k + u_k^T R u_k), \quad (45)$$

де \tilde{Q} , – симетрична позитивно напіввизначена матриця; R – симетрична позитивно визначена матриця.

Очевидно, що критерій (45) має бути збіжним, щоб підсистема (44) була стійкою. При цьому структура матриці \tilde{Q} має наступний вигляд:

$$\tilde{Q} = \begin{bmatrix} Q_{[n \times n]} & 0_{[n \times p]} \\ 0_{[p \times n]} & Q_v_{[p \times p]} \end{bmatrix}. \quad (46)$$

З урахуванням (46) критерій \tilde{J} представимо в такому вигляді:

$$\tilde{J} = 0.5 \sum_{k=0}^{\infty} (x_k^T Q x_k + v_k^T Q_v v_k + u_k^T R u_k). \quad (47)$$

Таким чином, для розширеної системи (44) можна розрахувати оптимальне керування u_k^* , що мінімізує квадратичний критерій (47) відповідно до процедури, подібної раніше розглянутій процедурі (26). Для кожної з підсистем (в замкнутому циклі) оптимальне керування визначається наступним чином:

$$u_k^* = -(\tilde{K} \tilde{x}_k + \tilde{N} \tilde{d}) = \tilde{L} \tilde{r}_k - (\tilde{K} x_k + \tilde{K}_v v_k + \tilde{N} d). \quad (48)$$

Відзначимо, що в рівнянні (48) матриці \tilde{K} і \tilde{N} не відповідають однозначно матрицям K і N , отриманим в рівнянні (26) для синтезу системи керування без додаткових інтеграторів. З іншого боку, матриці \tilde{K} і \tilde{N} , а також розширений вектор стану і вектор задавальних впливів з урахуванням взаємозв'язків декомпонуються відповідно наступним чином: $\tilde{K} = [\tilde{K} \quad \tilde{K}_v]$; $\tilde{N} = [\tilde{N} \quad -\tilde{L}]$; $\tilde{x}_k = [x_k \quad v_k]^T$; $\tilde{d} = [d \quad \tilde{r}_k]^T$.

Для розрахунку оптимального керування u_k^* , що компенсує статичні помилки (між задавальним сигналом і вимірюваним виходом) для кожної з підмоделей, описуваних розширеним рівнянням стану (44), пропонується наступний алгоритм:

1. Визначаємо матрицю рішення рівняння Ріккати (DARE):

$$\tilde{A}^T \tilde{P} \tilde{A} - \tilde{P} - (\tilde{A}^T \tilde{P} \tilde{B}) \times \times (R + \tilde{B}^T \tilde{P} \tilde{B})^{-1} (\tilde{B}^T \tilde{P} \tilde{A}) + \tilde{Q} = 0. \quad (49)$$

2. Визначаємо матриці \tilde{E} , \tilde{F} та \tilde{G} :

$$\begin{cases} \tilde{E} = \tilde{B} R^{-1} \tilde{B}^T \\ \tilde{F} = \tilde{A}^T \tilde{P} (I + \tilde{E} \tilde{P})^{-1} = \tilde{A}^T (\tilde{P}^{-1} + \tilde{E})^{-1} \\ \tilde{G} = I - \tilde{A}^T \tilde{P} \end{cases} \quad (50)$$

3. З урахуванням (49) і (50) отримуємо закон керування:

$$u_k^* = -(\tilde{K} \tilde{x}_k + \tilde{N} \tilde{d}), \quad (51)$$

де
$$\begin{cases} \tilde{K} = (R + \tilde{B}^T \tilde{P} \tilde{B})^{-1} \tilde{B}^T \tilde{P} \tilde{A} \\ \tilde{N} = (R + \tilde{B}^T \tilde{P} \tilde{B})^{-1} \tilde{B}^T (\tilde{P} + (\tilde{G} + \tilde{F} \tilde{E})^{-1} \tilde{F}) \end{cases}$$

Беручи до уваги (48), можна стверджувати, що отримання оптимального керування для афінних дискретних підмоделей полягає у визначенні зворотного зв'язку за станом $-\tilde{K} x_k$, компенсації складової взаємозв'язків $-\tilde{N} d$, складової $\tilde{L} r_k$, пов'язаної

з задавальним сигналом та інтегруючим впливом $-\tilde{K}_v v_k$, що компенсує неузгодженість між заданою дією і вимірюваним виходом. За аналогією з класичним цифровим ПІ-регулятором отриманий результат може бути представлений як закон багатовимірного ПІ-керування з компенсацією взаємозв'язків між підсистемами. Для нечіткої афінної дискретної ТС-моделі реалізація нечіткого PDC-регулятора з інтегруючою складовою полягає у визначенні (для $i=1,2,\dots,r$) правил моделі коефіцієнтів \tilde{K}_i і \tilde{N}_i для розширеної системи (44) з подальшим формуванням висновків відповідно до (51), використовуючи правила такого вигляду:

$$\text{«Якщо } z_1(k) \in M_1^i \text{ і } \dots \text{ і } z_p(k) \in M_p^i, \quad (52)$$

$$\text{то: } u(k) = -(\tilde{K}_i \tilde{x}(k) + \tilde{N}_i \tilde{d})\text{»},$$

де p – кількість правил.

Таким чином, остаточний закон керування, що формується досліджуваным нечітким регулятором з інтегруванням, має наступний вигляд:

$$u(k) = -\sum_{i=1}^r h_i(z(k)) (\tilde{K}_i \tilde{x}(k) + \tilde{N}_i \tilde{d}), \quad (53)$$

де
$$h_i(z(k)) = \frac{w_i(z(k))}{\sum_{i=1}^r w_i(z(k))}$$

З урахуванням рівняння (48) загальна структура розподіленого PDC-керування з додатковою інтегруючою складовою для лінійних афінних підмоделей може бути представлена схемою, наведеною на рис.4.

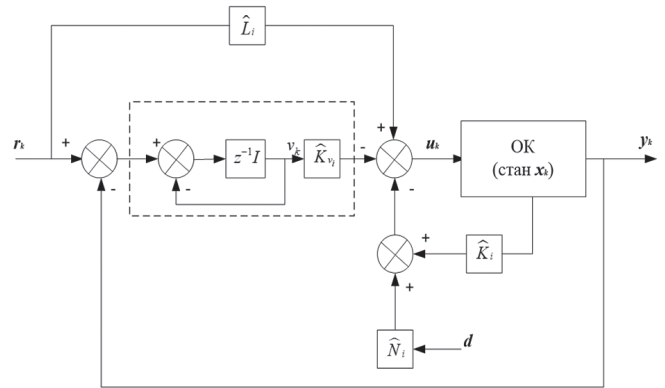


Рис. 4. Схема PDC-керування з додатковим інтегратором

3. Визначення вагових матриць нечіткої системи PDC-керування

Очевидно, що значення матриць \tilde{K}_i і \tilde{N}_i в законі керування (51) залежать від вибору матриць матриці \tilde{Q} і R , які задаються априорі. Щоб зважувати розширений стан \tilde{x}_k і вхід u_k , ці дві матриці мають бути позитивно визначеними.

Щоб спростити перевірку позитивності \tilde{Q} і R , доцільно вибирати їх з класу діагональних матриць з позитивними елементами головної діагоналі:

$$Q = \text{diag}(q_1, q_2, \dots, q_n); \quad R = \text{diag}(r_1, r_2, \dots, r_m). \quad (54)$$

При цьому: чим більше q_i , тим швидше сходиться до нуля відповідна змінна \tilde{x}_i ; чим більше r_j , тим менше амплітуда відповідного керування u_j .

Якщо i -й елемент j -й елемент приймають відповідно максимально допустимі значення $\max(\tilde{x}_i)$ та $\max(u_j)$, то вибір q_i та r_j можна здійснювати наступним чином:

$$q_i = (1/\max(\tilde{x}_i))^2; r_j = (1/\max(u_j))^2. \quad (55)$$

Слід зазначити, що даний метод вибору структури керування передбачає, що зміни задають впливів здійснюються стрибкоподібно. Якщо закон керування використовується для здійснення автоматичного регулювання (з програмною зміною значень задавальних впливів), настройка системи істотно ускладнюється. Налаштування матриць \tilde{Q} і R , що входять в квадратичний критерій, має враховувати компроміс між амплітудами керувань і загальною якістю контуру керування. Наприклад, підвищення значень коефіцієнтів матриці R по відношенню до значень коефіцієнтів матриці \tilde{Q} штрафувє входи, але в той же час це призводить до ослаблення зважування помилок. Крім того, компенсація за можливі збурення може стати дуже повільною по відношенню до потенційних поліпшень показників якості обраної структури керування.

Для забезпечення ефективного функціонування системи в розглянутих випадках можна ввести в структуру керування різницеву модель, входом якої є вектор задавальних впливів r_k , а вихід відображає бажану еволюцію зміни вектора y_k керуваного процесу. Це дозволить розділити завдання вибору елементів контуру зворотного зв'язку і завдання забезпечення якості регулювання.

Цікавою перспективою продовження проведених досліджень є також аналіз стійкості для системи, що використовує нечітке керування по дискретній ТС-моделі.

4. Моделювання нечіткого цифрового регулятора для PDC-керування ГПА

Для моделювання системи нечіткого керування ГПА КС з використанням ТС-моделі використовувалися можливості пакета для моделювання динамічних систем Simulink і пакета синтезу нечітких систем управління Fuzzy Logic Toolbox.

Для різних діапазонів вихідного тиску (або ступеня стиснення газу) динамічні властивості ГПА можуть змінюватися (через нелінійність статичної залежності по каналу «швидкість обертів вала ГПА - вихідний тиск»). Згідно з алгоритмом нечіткої кластеризації, наведеним в [7], було проведено поділ загального діапазону зміни вихідного тиску на 5 інтервалів, що дозволило після вибору відповідних функцій належності сформувати 5 лінійних підсистем. Правила

визначення виходів для кожної з цих підсистем наведені в табл. 1.

Таблиця 1

Таблиця виходів локальних підсистем нечіткої моделі ГПА

№ підсистеми	Вихід $y(k+1)$
1	$0,30y(k) + 0,99u(k) + 1,603$
2	$-0,258y(k) + 0,91u(k) + 1,078$
3	$-0,574y(k) + 0,97u(k) + 1,052$
4	$0,193y(k) + 0,905u(k) + 0,453$
5	$0,158y(k) + 0,9u(k) + 0,511$

На основі отриманої моделі та розглянутих вище алгоритмів PDC-керування було побудовано закон керування вихідним тиском ГПА типу (53).

Результати моделювання роботи системи при стрибкоподібній зміні величини задавального сигналу від 8,0 МПа до 8,5 МПа і подальшому її зниженні до 8,0 МПа підтвердив переваги запропонованої схеми нечіткого керування в порівнянні з ПІД-керуванням. Зміна режиму роботи ГПА здійснюється на верхньому рівні автоматизованої системи керування технологічними процесами компресорної станції.

Тестування нечіткого PDC-регулятора з додатковим інтегратором свідчить про можливість ефективної компенсації статичних помилок керування вихідними параметрами ГПА. Крім того, додаткові можливості підвищення якості цього керування може дати експериментальне налаштування матричних коефіцієнтів \tilde{Q} і R квадратичного критерію (45).

Висновки

У статті досліджено завдання розробки алгоритмів PDC-керування нелінійними об'єктами за станом з використанням дискретних локальних ТС-моделей. Розглянуто особливості синтезу систем PDC-керування. Запропоновано підхід до побудови схем PDC-регуляторів з компенсацією статичних помилок за рахунок введення в структуру системи керування цифрових інтеграторів і наступного обчислення коефіцієнтів регулятора для кожної з локальних моделей.

Наведено рекомендації по визначенню вагових коефіцієнтів квадратичного критерію керування.

Результати моделювання системи нечіткого PDC-керування газоперекачувальними агрегатами компресорних станцій з використанням ТС-моделі свідчать про переваги запропонованих нечітких регуляторів в порівнянні з ПІД-регуляторами.

Перспективними напрямками продовження проведених досліджень є аналіз стійкості для системи, що використовує нечітке PDC-керування, а також дослідження еквівалентності (на динамічному рівні) між нечіткою ТС-моделлю і нелінійною керуваною системою, що апроксимується.

Список літератури:

- [1] *Lozynskyy A.* Synthesis of fuzzy logic controller of nonlinear dynamic system with variable parameters / A. Lozynskyy, L. Demkiv // *Computational problems of electrical engineering* Vol. 6, No. 2, 2016, P. 91–98.
- [2] *Штовба С.Д.* Введение в теорию нечетких множеств и нечеткую логику / С.Д. Штовба // [Електронний ресурс]: – Режим доступу: <http://matlab.exponenta.ru/fuzzylogic/book1/index.php>
- [3] *Gorokhovatskyi V.A.* Employment of Intelligent Technologies in Multiparametric Control Systems / V.A. Gorokhovatskyi, A.A. Zamula // *Telecommunications and Radio Engineering*. – 2016, Vol. 75, No 19. – P. 1775–1785.
- [4] *Takagi T.* Fuzzy identification of systems and its applications to modeling and control / T. Takagi, M. Sugeno // *IEEE Trans. Syst., Man and Cyber* – 1985 – V. 15. – P. 116–132.
- [5] *G. S. Rao* Implementation of fuzzy logic controller in SHPF&TCR for power quality/ G. S. Rao and Y. R. Reddy: in 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), 2016, P. 2960–2964.
- [6] *Замула А. А.* Нечеткое моделирование при проектировании структуры интеллектуальной системы управления / А. А. Замула // *Системы обработки информации*. – 2016. – Вып. 1. – С. 81-85.
- [7] *Удовенко С.Г.* Керування агрегатами компресорної станції з використанням нечітких інверсних моделей / С.Г. Удовенко, Д.В. Келембет // *Біоніка інтелекту*. – 2018. – Вып. 2 (91). – С. 25-30.
- [8] *Milad G.* Design of fuzzy parallel distributed compensation controller for magnetic levitation system / G. Milad, A. Zohreh, D. Sanai, H. Masoud // *Journal of Electrical and Electronics Engineering (IOSR - JEEE)* – 2017. – Vol. 12 (4) –P. 20 – 28.
- [9] *Morteza S.* Performance-oriented parallel distributed compensation / S. Morteza, A. Markazi // *Journal of the Franklin Institute*. – 2010. – V. 348 (7). – P. 1231–1244.
- [10] *Shayanfar H.A.* Takagi-Sugeno fuzzy parallel distribution compensation based three-area LFC design / H.A. Shayanfar, H. Shayeghi, A. Jalili // *International Journal on “Technical and Physical Problems Problems of Engineering”*. – 2011. – V. 3 (8). – P. 55–64.

Надійшла до редколегії 22.10.2019



К.Э. Петров¹, Т.С. Чайникова², И.В. Кобзев³, В.Г. Демчук⁴

¹Доктор технических наук, профессор кафедры искусственного интеллекта, Харьковский национальный университет радиоэлектроники, kostiantyn.petrov@nure.ua, ORCID iD: 0000-0003-1973-711X

²Менеджер по контролю качества, Компания Certent, г. Киев, chaynikova@gmail.com, ORCID iD: 0000-0002-8923-8829

³Кандидат технических наук, доцент кафедры информационных технологий и систем управления, Харьковский региональный институт государственного управления Национальной академии государственного управления при Президенте Украины, ikobzev12@gmail.com, ORCID iD: 0000-0002-7182-5814

⁴Аспирант кафедры электронных вычислительных машин, Харьковский национальный университет радиоэлектроники, vadym.demchuk@nure.ua, ORCID iD: 0000-0003-3700-2344

КОМПАРАТОРНАЯ ИДЕНТИФИКАЦИЯ МОДЕЛИ МНОГОФАКТОРНОГО ОЦЕНИВАНИЯ АЛЬТЕРНАТИВ С ИСПОЛЬЗОВАНИЕМ МЕТОДА БЭГГИНГА

Предложен один из подходов к решению задачи построения математической модели многофакторного оценивания альтернатив, который базируется на идеях теории компараторной идентификации. Рассмотрены методы ее структурной и параметрической идентификации. Структура модели определяется в рамках полинома Колмогорова-Габора, что позволяет синтезировать как традиционные аддитивные и мультипликативные, так и модели многофакторного оценивания альтернатив произвольной сложности. Для повышения точности модели предлагается использовать ансамбль моделей, агрегирование которых осуществляется на основе метода бэггинга. Приведены результаты компьютерного моделирования, подтверждающие эффективность описанного подхода. Предложенный подход открывает перспективы повышения эффективности и объективности идентификации моделей экспертного оценивания альтернатив.

МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ, ФУНКЦИЯ ПОЛЕЗНОСТИ, ОТНОШЕНИЕ ПОРЯДКА, РАНЖИРОВАНИЕ АЛЬТЕРНАТИВ, АНСАМБЛЬ МОДЕЛЕЙ

Запропоновано один з підходів до розв'язання задачі побудови математичної моделі багатфакторного оцінювання альтернатив, який базується на ідеях теорії компараторної ідентифікації. Розглянуто методи її структурної і параметричної ідентифікації. Структура моделі визначається в рамках поліному Колмогорова-Габора, що дозволяє синтезувати як традиційні адитивні і мультиплікативні, так і моделі багатфакторного оцінювання альтернатив довільної складності. Для підвищення точності моделі пропонується використовувати ансамбль моделей, агрегування яких здійснюється на основі методу бегінга. Наведено результати комп'ютерного моделювання, що підтверджують ефективність описаного підходу. Запропонований підхід відкриває перспективи підвищення ефективності та об'єктивності ідентифікації моделей експертного оцінювання альтернатив.

БАГАТОКРИТЕРІАЛЬНА ОПТИМІЗАЦІЯ, ФУНКЦІЯ КОРИСНОСТІ, ВІДНОШЕННЯ ПОРЯДКУ, РАНЖУВАННЯ АЛЬТЕРНАТИВ, АНСАМБЛЬ МОДЕЛЕЙ

One of the approaches to solving the problem of constructing a mathematical model of multifactor estimation of alternatives, which is based on the ideas of the comparative identification theory, is proposed. The methods of its structural and parametric identification are considered. A model structure is determined within the fragments of Kolmogorov-Gabor polynomial, that allows to synthesize both traditional additive and multiplicative and models of multifactor estimations of alternatives, which can have of any complication. To improve the accuracy of the model, it is proposed to use an ensemble of models, the aggregation of which is carried out on the basis of the bagging method. The results of computer modeling confirming the effectiveness of the described approach are presented. The proposed approach opens up prospects for improving the efficiency and objectivity of the identification of models for expert estimation of alternatives.

MULTICRITERIA OPTIMIZATION, FUNCTION OF UTILITY, RELATIONSHIP OF ORDER, RANKING OF ALTERNATIVES, ENSEMBLE OF MODELS

Введение

Процесс принятия решений можно определить как особый вид интеллектуальной человеческой деятельности, состоящий в выборе одного или нескольких вариантов решений (альтернатив) из некоторого имеющегося множества [1]. Одним из ключевых этапов этого процесса является оценивание альтернатив экспертом, т. к. именно от него зависит эффективность принятого решения.

Альтернативы описываются в форме множеств разнородных частных характеристик, проанализировав которые эксперт должен представить свое суждение о наиболее предпочтительной или предпочтительных из них для решения имеющейся проблемы.

Основная гипотеза теории рационального поведения заключается в том, что индивидум выбирает альтернативу, которая дает наилучший результат. Теория же полезности предполагает существование

некоторой обобщенной оценки альтернативы, на основе которой и происходит этот выбор.

Таким образом, процесс оценивания, это не что иное, как присвоение альтернативам некоторых качественных или количественных оценок с целью их ранжирования по степени предпочтительности для эксперта.

Формализация процесса получения таких оценок сталкивается с огромными трудностями, так как в этом случае и объектом и субъектом исследования является эксперт, что существенно влияет на результат принятия решения. И все же, несмотря на целый ряд недостатков, неточностей и субъективизма, информация, полученная от экспертов, является практически единственным источником исходных данных.

Для получения любой информации от экспертов необходимо провести с ними серию активных или пассивных экспериментов. Этот этап присутствует в большинстве подходов, применяемых в настоящее время для получения многофакторных оценок альтернатив.

Ранжирование допустимого множества и выбор экстремального, т. е. наилучшего с точки зрения качества, единственного решения так или иначе сводится к решению задачи многокритериальной оптимизации.

Существующие в настоящее время подходы к решению такого рода задач, например, такие как метод анализа иерархий [2] и его обобщение — анализ сетей [3], методы ELECTRE [4], PROMETHEE [5], TOPSIS [6], STEM [7], методы последовательной оптимизации [7, 8], функционально-стоимостной анализ [7], метод формирования обобщенного скалярного критерия (свертки) [9] и т. п. требуют от эксперта в явном виде предоставить информацию об относительной важности частных критериев, характеризующих альтернативы или о значениях их «весовых» коэффициентов.

В отличие от этих методов, в работе предлагается один из подходов к определению относительных многофакторных оценок альтернатив в случае, когда информация, полученная от экспертов, имеет нечисловой характер, т. е. имеются данные только о наиболее предпочтительной альтернативе или установлено отношение частичного порядка на множестве альтернатив, который базируется на применении метода компараторной идентификации [1].

Для повышения точности решения задачи определения скалярных многофакторных оценок альтернатив, вместо использования одиночной модели оценивания предлагается формировать ансамбль моделей на основе применения метода бэггинга (сокр. от **bootstrap aggregating**) [10].

1. Постановка задачи

Пусть имеется некоторое конечное множество альтернатив $X = \{x_1, x_2, \dots, x_n\}$. Каждая из них может быть описана кортежем разнородных частных характеристик (факторов) $K(x_i) = \langle k_1(x_i), k_2(x_i), \dots, k_m(x_i) \rangle$,

$i = \overline{1, n}$, которые могут быть измерены в количественных шкалах.

Эти альтернативы (варианты действий) предъявляются эксперту, который оценивает их с точки зрения полезности для последующего выбора наиболее предпочтительных альтернатив при принятии решения в каждой конкретной ситуации.

Такое оценивание возможно осуществить в рамках теории полезности, основная гипотеза которой заключается в том, что предполагается существование некоторой скалярной многофакторной оценки обобщенной полезности $P(x_i)$, $i = \overline{1, n}$ для каждой из альтернатив $x_i \in X$. Эту оценку (функцию полезности) можно представить в таком виде:

$$P(x_i) = F[A, K(x_i)], \quad x_i \in X, \quad i = \overline{1, n}, \quad (1)$$

где $A = \langle a_1, a_2, \dots, a_i \rangle$ — кортеж значений коэффициентов, выражающих относительную важность («вес») для эксперта частных факторов $k_j(x_i)$, $j = \overline{1, m}$, которые характеризуют альтернативы $x_i \in X$.

Задача состоит в структурной, т. е. в определении вида оператора F и параметрической идентификации — нахождении значений кортежа A математической модели многофакторного оценивания (1) на основании информации только о наиболее предпочтительной альтернативе либо их частичного упорядочения, полученного в ходе анализа экспертом множества имеющихся альтернатив.

Решение этой задачи позволит провести ранжирование альтернатив исходя из их значимости для эксперта с учетом того, что по определению для функции полезности $P(x_i)$ выполняются следующие соотношения:

$$x_i \succ x_j \Leftrightarrow P(x_i) > P(x_j);$$

$$\text{и} \quad x_i \sim x_j \Leftrightarrow P(x_i) = P(x_j), \quad \forall x_i, x_j \in X, \quad i \neq j. \quad (2)$$

2. Структурная и параметрическая идентификация модели многофакторного оценивания

Без потери общности рассмотрим ситуацию когда эксперт на основании анализа представленных альтернатив произвел их упорядочение по степени предпочтительности. В общем случае, исходя из (2) получим систему из $n-1$ ограничений, где n — количество альтернатив x_i .

Например, для случая $X = \{x_1, x_2, \dots, x_4\}$ и установленного отношения порядка $x_2 \succ x_3 \sim x_1 \succ x_4$ получим такую систему ограничений:

$$\begin{cases} P(x_2) > P(x_3) \\ P(x_3) = P(x_1) \\ P(x_1) > P(x_4) \end{cases} \Leftrightarrow \begin{cases} P(x_2) - P(x_3) > 0 \\ P(x_3) - P(x_1) = 0 \\ P(x_1) - P(x_4) > 0 \end{cases}. \quad (3)$$

В качестве универсальной функции полезности $P(x_i)$ в работе предлагается использовать полином Колмогорова-Габора [11], который в принятых выше обозначениях имеет вид:

$$P(x_i) = a_0 + \sum_{j=1}^m a_j k_j(x_i) + \sum_{j=1}^m \sum_{q=1}^m a_{jq} k_j(x_i) k_q(x_i) + \sum_{j=1}^m \sum_{q=1}^m \sum_{r=1}^m a_{jqr} k_j(x_i) k_q(x_i) k_r(x_i) + \dots \quad (4)$$

Выбор именно полинома такого вида обусловлен следующими причинами:

– как показано в работе Колмогорова [11] полином (4) позволяет точно аппроксимировать любую функцию многих переменных;

– полином содержит в своем составе как аддитивные так и мультипликативные линейные по характеризующим факторам $K(x_i)$ составляющие и позволяет формировать на их основе полиномы любого вида;

– частные полиномы, синтезируемые на основе (4), имеют традиционную форму (аддитивную, мультипликативную и т. п.), а составляющие имеют ясную интерпретацию, как "вклад" тех или иных факторов $K(x_i)$ или их комплексов в обобщенную многофакторную оценку альтернативы.

Кроме того, следует отметить еще одно важное свойство полинома Колмогорова-Габора (4): любой полином, синтезированный как фрагмент (4) является линейным по параметрам A , но в общем случае существенно нелинейным по множеству факторов $K(x_i)$, т. е. по входу.

Для целей нашего исследования выберем фрагмент полинома (4), в котором примем $a_0 = 0$ (при нулевых значениях характеристик $k_j(x_i)$ "полезность" любой альтернативы равна нулю) и ограничимся учетом членов только первого порядка. Как показали компьютерные эксперименты [1], использование такой простейшей аддитивной модели является в данном случае оправданным из-за ограниченного объема информации полученной от эксперта в процессе выбора альтернатив. Однако это не исключает использования более сложных моделей с членами второго, третьего и более высоких порядков, учитывающих взаимовлияние частных характеристик, в случае, если удастся получить больше информации от эксперта. Вопросы структурной и параметрической идентификации таких моделей на основе использования генетических алгоритмов и метода группового учета аргументов подробно рассмотрены в [12].

Таким образом, значения a_j и $P(x_i)$ будем определять в рамках фрагмента полинома (4), вида

$$P(x_i) = \sum_{j=1}^m a_j k_j^H(x_i), \quad (5)$$

где $K^H(x_i) = \langle k_1^H(x_i), k_2^H(x_i), \dots, k_m^H(x_i) \rangle$ – нормированные значения частных характеристик альтернатив; a_j – безразмерные коэффициенты относительной важности нормированных частных характеристик $k_j^H(x_i)$, которые удовлетворяют условиям:

$$a_j \in [0, 1], \quad j = \overline{1, m}; \quad \sum_{j=1}^m a_j = 1. \quad (6)$$

Нормирование частных характеристик необходимо, так как, в общем случае, они имеют различные размерность, интервал изменений и направление доминирования. Это нормирование можно осуществить по формуле:

$$k_j^H(x_i) = \left(\frac{k_j(x_i) - k_j^-(x_i)}{k_j^+(x_i) - k_j^-(x_i)} \right)^{\alpha_j}, \quad j = \overline{1, m}, \quad i = \overline{1, n}, \quad (7)$$

где $k_j(x_i)$ – действительное (абсолютное) значение j -й частной характеристики; $k_j^-(x_i)$ и $k_j^+(x_i)$ – соответственно ее "наихудшее" и "наилучшее" значения в зависимости от направления доминирования $k_j(x_i)$; α_j – коэффициент нелинейности ($\alpha_j > 0$).

Тогда с учетом (2), (5) и (7) для случая $x_r \succ x_s$ можно записать:

$$P(x_r) > P(x_s) \Leftrightarrow P(x_r) - P(x_s) > 0 \quad \forall x_r, x_s \in X, \quad r \neq s, \quad \text{т. е.}$$

$$P(x_r) - P(x_s) \equiv \left(\sum_{j=1}^m a_j k_j^H(x_r) \right) - \left(\sum_{j=1}^m a_j k_j^H(x_s) \right) > 0 \quad (8)$$

или

$$\sum_{j=1}^m a_j [k_j^H(x_r) - k_j^H(x_s)] > 0, \quad (9)$$

где $a_j, j = \overline{1, m}$ удовлетворяют условиям (6).

Таким образом, полученная от эксперта информация в виде отношения порядка на множестве альтернатив X (например, (3)), может быть формализована в виде системы линейных ограничений вида (9), которая в свою очередь определяет выпуклый многогранник на гиперплоскости (6) любая точка которого является допустимым решением. Следовательно, задача определения коэффициентов относительной важности частных характеристик a_j является некорректной по Тихонову [13], так как в исходном виде не имеет единственного решения. Для получения такого решения исходную задачу необходимо регуляризовать [13] путем добавления в нее некоторого дополнительного условия.

В связи с отсутствием информации, позволяющей выдвинуть объективную гипотезу, определяющую вид регуляризирующей функции, в качестве рабочей примем эвристическую гипотезу, что точечная оценка параметров модели многофакторного оценивания должна находиться в центральной области многогранника допустимых значений коэффициентов, определяемых системой ограничений (6) и (9).

Для регуляризации задачи, в качестве единственного решения предлагается выбрать чебышевскую точку [14] для системы линейных ограничений (6) и (9). Экспериментальная проверка показала, что данное решение обладает высокой устойчивостью к изменению границ многогранника.

Чебышевская точка, по сути, представляет собой точку, находящуюся внутри области допустимых значений Ω кортежа A (для совместной системы

линейных ограничений). Эта точка равно уклонена на некоторую величину $L < 0$ от $n+1$, гиперплоскостей, ограничивающий n -мерный симплекс и уклоняется не более чем на L от остальных $m-n-1$ гиперплоскостей, где n – число переменных, а m – количество ограничений.

Поскольку все ограничения вида (6), (9), входящие в систему, являются линейными, задачу определения чебышевской точки можно свести к задаче линейного программирования (ЛП) [14].

Для примера (3) формальная запись такой задачи ЛП будет выглядеть следующим образом:

$$\begin{cases}
 f(a_1, a_2, \dots, a_m, L) \equiv L \rightarrow \min \\
 \sum_{j=1}^m a_j [k_j^H(x_2) - k_j^H(x_3)] + L \geq 0 \\
 \sum_{j=1}^m a_j [k_j^H(x_3) - k_j^H(x_1)] + L \geq 0 \\
 \sum_{j=1}^m a_j [-k_j^H(x_3) + k_j^H(x_1)] + L \geq 0 \\
 \sum_{j=1}^m a_j [k_j^H(x_1) - k_j^H(x_4)] + L \geq 0 \\
 a_j + L \geq 0, j = \overline{1, m} \\
 \sum_{j=1}^m a_j = 1
 \end{cases} \quad (10)$$

В системе линейных ограничений (10) знаки «>» изменены на «≥» и равенство $P(x_3) - P(x_1) = 0$ представлено в виде двух неравенств $P(x_3) - P(x_1) \geq 0$ и $-P(x_3) + P(x_1) \geq 0$. Замена знаков ограничений не влияет на решение так как оно лежит внутри многогранника описываемого системой (10), а не на его границах.

В результате решения задачи (10), например симплекс-методом, получим точечные значения кортежа «весовых» коэффициентов частных характеристик A , а также относительных многофакторных оценок альтернатив $P(x_i)$, $i = \overline{1, n}$, вычисленных на основе A . Это позволяет провести ранжирование альтернатив исходя из полученных значений $P(x_i)$, $i = \overline{1, n}$.

3. Использование бэггинга для повышения точности модели многофакторного оценивания альтернатив

Точность модели является одним из важнейших ее свойств. Повысить точность модели можно при помощи построения ансамбля моделей, т. е. набора моделей, «работающих» над решением одной и той же задачи. В частности, использование метода построения ансамбля моделей основанного на применении бэггинга дает существенно более точный результат, чем применение одиночной модели на имеющемся наборе данных.

Рассмотрим метод бэггинга более подробно. Подобно тому, как усреднение нескольких наблюдений снижает оценку дисперсии данных, так и

возможное снижение дисперсии прогноза можно достичь путем извлечения большого количества порций данных (обучающих выборок) из генеральной совокупности, построения предсказательной модели по каждой обучающей выборке и последующее усреднение полученных предсказаний. Если вместо отдельных обучающих выборок (которых, как правило, всегда не хватает) применить процедуру бутстрепа (bootstrap) [15], которая заключается в многократной генерации выборок методом Монте-Карло на базе имеющейся выборки и на основе сгенерированных псевдовыборок построить N моделей регрессии $f_i(x)$, $i = \overline{1, N}$, то средний коллективный прогноз $f_{cp}(x) = (f_1(x) + f_2(x) + \dots + f_N(x))/N$ будет обладать более низкой дисперсией. Эта процедура и называется бэггингом [15].

Для повышения точности модели оценивания предлагается использовать следующий алгоритм на основе использования метода бэггинга.

1. Пусть имеется множество $X = \{x_1, x_2, \dots, x_n\}$ состоящее из n альтернатив и задано отношение порядка на этом множестве. Из множества X случайным образом отбирается несколько подмножеств $X_1 \in X$, $X_2 \in X$, ..., $X_p \in X$, количество элементов в которых $m < n$ (общее количество этих подмножеств равно C_n^m). На каждом из подмножеств $X_q \in X$, $q = \overline{1, p}$ сохраняется отношение порядка альтернатив установленного для исходного множества X .

2. На основе каждого подмножества $X_q \in X$, $q = \overline{1, p}$ строится математическая модель многофакторного оценивания альтернатив (1) в соответствии с подходом описанным выше. В результате получаем значения функций полезности (выходы модели) $P_q(x_i)$, для каждой альтернативы $x_i \in X$, $i = \overline{1, n}$ по каждой из моделей построенной исходя из $X_q \in X$, $q = \overline{1, p}$. Таким образом получаем ансамбль моделей.

3. Находим среднее значение функции полезности $P_{cp}(x_i)$ для каждой альтернативы $x_i \in X$, $i = \overline{1, n}$ по всем моделям соответствующим $X_q \in X$, $q = \overline{1, p}$ (метод бэггинга) по формуле:

$$P_{cp}(x_i) = (P_1(x_i) + P_2(x_i) + \dots + P_p(x_i))/p, i = \overline{1, n}. \quad (11)$$

4. Проводим ранжирование альтернатив в соответствии с полученными средними значениями их функций полезности $P_{cp}(x_i)$, $i = \overline{1, n}$.

Ожидается, что результат ранжирования полученный путем агрегации моделей (построения ансамбля моделей) будет намного точнее результата ранжирования по одиночной модели при тех же исходных данных.

4. Имитационный эксперимент

Для иллюстрации работоспособности предлагаемого метода рассмотрим следующий абстрактный пример.

Пусть имеется десять альтернатив, каждая из которых характеризуется семью частными характеристиками.

Сгенерируем значения нормализованных частных характеристик $K^H(x_i) = \langle k_j^H(x_i) \rangle, j = \overline{1,7}$ альтернатив $X = \{x_i\}, i = \overline{1,10}$ с помощью датчика случайных чисел. Все эти данные представлены в таблице 1.

Также случайным образом сгенерируем порядок следования альтернатив по убыванию их полезности для эксперта.

Пусть получено следующее отношение линейного порядка:

$$x_9 \succ x_{10} \succ x_4 \succ x_7 \succ x_5 \succ x_3 \succ x_1 \succ x_6 \succ x_2 \succ x_8. \quad (12)$$

В качестве функции полезности альтернатив $x_i, i = \overline{1,10}$ выберем полином вида:

$$P(x_i) = a_1 k_1^H(x_i) + a_2 k_2^H(x_i) + \dots + a_7 k_7^H(x_i), \quad (13)$$

где $a_j, j = \overline{1,7}$ удовлетворяют условиям (6).

В соответствии с (2), (12) и (13) получим следующую систему линейных неравенств:

$$\begin{cases} P(x_9) - P(x_{10}) \geq 0 \\ P(x_{10}) - P(x_4) \geq 0 \\ P(x_4) - P(x_7) \geq 0 \\ P(x_7) - P(x_5) \geq 0 \\ P(x_5) - P(x_3) \geq 0 \\ P(x_3) - P(x_1) \geq 0 \\ P(x_1) - P(x_6) \geq 0 \\ P(x_6) - P(x_2) \geq 0 \\ P(x_2) - P(x_8) \geq 0 \end{cases} \quad (14)$$

Убедимся, что существует такой полином вида (13), который «аппроксимирует» отношение порядка (12) установленного на множестве альтернатив $X = \{x_i\}, i = \overline{1,10}$.

Для этого находим чебышевскую точку для системы ограничений (14), т. е. значения $A^* = \langle a_j^* \rangle, j = \overline{1,7}$ в соответствии с методом описанным выше (см. (10)). Все вычисления проводились с использованием программного средства Mathcad v.14.0.

Были получены следующие значения: $A^* = \langle 0,10; 0,19; 0,14; 0,16; 0,17; 0,13; 0,11 \rangle$ и $L = -0,02$.

Далее, в соответствии с найденными значениями $\langle a_j^* \rangle, j = \overline{1,7}$, по формуле (13) вычислим модельные значения функции полезности для каждой из альтернатив $x_i \in X$, т. е. $P^*(x_i), i = \overline{1,10}$. Таким образом, мы получили относительные обобщенные скалярные оценки альтернатив (столбец $P^*(x_i)$ в табл. 1), на основании которых можно произвести их ранжирование по убыванию полезности (столбец R^* в табл. 1).

Следовательно, полученный полином $P^*(x_i)$ со значениями «весовых» коэффициентов A^* , который «аппроксимирует» отношение порядка (12) существует.

Разделим множество альтернатив $X = \{x_i\}, i = \overline{1,10}$ на обучающее и проверочное. Пусть обучающее подмножество содержит альтернативы $x_1, x_5, x_6, x_7, x_8, x_9, x_{10}$, а проверочное – x_2, x_3, x_4 .

В этом случае в соответствии с (12) имеем следующую информацию для проверочного подмножества:

$$Z0: x_9 \succ x_{10} \succ x_7 \succ x_5 \succ x_1 \succ x_6 \succ x_8. \quad (15)$$

На основе этой информации определяем значения коэффициентов полинома $P_0(x_i)$, который «аппроксимирует» отношение порядка (15) аналогично методу описанному выше. Значения функции полезности для каждой из альтернатив $x_i \in X, i = \overline{1,10}$, в том числе и для проверочного подмножества, представлены в таблице 2 (столбец $P_0(x_i)$), а их соответствующие ранги – в таблице 3 (столбец R_0).

Для оценки точности полученного ранжирования альтернатив по отношению к исходному R^* используем коэффициент ранговой корреляции Спирмена [15]:

$$\rho_t = 1 - \frac{6 \cdot \sum_{i=1}^n (R_i^* - R_{ti})^2}{n \cdot (n^2 - 1)}, \quad (16)$$

где R_i^* и R_{ti} – ранги (места) в исходном отношении порядка альтернатив и в соответствующих установленных модельных отношениях порядка $Z0, Z1, \dots, Zt$; n – количество альтернатив $x_i \in X$.

Таким образом, анализируя полученное в соответствии с отношением порядка $Z0$ (15) ранжирование

Таблица 1

Значения нормализованных частных характеристик альтернатив

	$k_1^H(x_i)$	$k_2^H(x_i)$	$k_3^H(x_i)$	$k_4^H(x_i)$	$k_5^H(x_i)$	$k_6^H(x_i)$	$k_7^H(x_i)$	$P^*(x_i)$	R^*
x_1	0,00	0,29	0,82	0,16	0,46	1,00	0,62	0,4719	7
x_2	0,18	0,71	0,64	0,32	0,25	0,51	0,24	0,4289	9
x_3	0,49	0,00	0,70	0,15	0,79	0,63	1,00	0,4972	6
x_4	0,39	0,60	0,62	0,58	0,48	0,46	0,83	0,5653	3
x_5	0,13	0,84	0,00	0,73	0,73	0,42	0,51	0,5242	5
x_6	0,52	0,50	0,94	0,18	0,58	0,37	0,00	0,4541	8
x_7	0,68	0,59	0,79	1,00	0,00	0,53	0,24	0,546	4
x_8	0,35	1,00	0,08	0,21	0,57	0,00	0,25	0,3942	10
x_9	1,00	0,53	0,81	0,62	0,77	0,40	0,42	0,6424	1
x_{10}	0,60	0,45	1,00	0,00	1,00	0,48	0,64	0,5883	2

альтернатив R_0 можно заметить значительное расхождение с исходным ранжированием R^* (см табл. 3 столбцы R^* и R_0). Об этом говорит и значение коэффициента Спирмена $\rho_0 = 0,87$ (см. табл. 3 строка ρ_i столбца R_0).

Воспользуемся методом бэггинга для повышения точности предлагаемой модели многофакторного оценивания.

Как было отмечено выше основная идея бэггинга – усреднить некоторое множество «шумных» и несмещённых моделей для создания модели с низким расхождением в рамках аппроксимации.

Сгенерируем случайным образом на основе отношения порядка (15) шесть неполных отношений, содержащих только пять альтернатив (общее количество таких отношений равно $C_7^5 = 21$):

$$Z1: x_9 \succ x_7 \succ x_1 \succ x_6 \succ x_8;$$

$$Z2: x_{10} \succ x_7 \succ x_5 \succ x_1 \succ x_8;$$

$$Z3: x_7 \succ x_5 \succ x_1 \succ x_6 \succ x_8;$$

$$Z4: x_9 \succ x_{10} \succ x_7 \succ x_5 \succ x_6;$$

$$Z5: x_9 \succ x_{10} \succ x_7 \succ x_5 \succ x_1;$$

$$Z6: x_{10} \succ x_7 \succ x_5 \succ x_1 \succ x_6.$$

Затем, с помощью метода описанного выше, проведем параметрическую идентификацию моделей многофакторного оценивания для каждого из отношений $Z1, Z2, \dots, Z6$, вычислим значения относительных полезностей альтернатив исходя из полученных моделей (см. табл. 2 столбцы $P_1(x_i), P_2(x_i), \dots, P_6(x_i)$) и проведем их ранжирование в порядке убывания этих значений (см. табл. 3 столбцы, R_1, R_2, \dots, R_6). В строке ρ_i таблицы 3 приведены соответствующие значения коэффициентов ранговой корреляции для исходного порядка альтернатив R^* и соответствующих ранжирований R_1, R_2, \dots, R_6 .

Далее усредняем полученные значения относительных полезностей альтернатив по формуле

$$P_{cp}(x_i) = \frac{\sum_{q=1}^6 P_q(x_i)}{6}$$

и проводим ранжирование альтернатив исходя из этих усредненных величин.

Полученные результаты представлены в соответствующих столбцах $P_{cp}(x_i)$ таблицы 2 и R_{cp} табл. 3.

Таблица 3

Значения рангов альтернатив

	R^*	R_0	R_1	R_2	R_3	R_4	R_5	R_6	R_{cp}
x_1	7	7	6	8	5	7	10	7	7
x_2	9	9	9	9	8	10	9	10	9
x_3	6	2	2	5	6	2	5	2	5
x_4	3	3	3	4	3	4	4	3	3
x_5	5	6	7	7	4	6	7	6	6
x_6	8	8	8	6	9	8	6	8	8
x_7	4	5	5	3	1	5	3	5	4
x_8	10	10	10	10	10	9	8	9	10
x_9	1	1	1	1	2	1	1	1	1
x_{10}	2	4	4	2	7	3	2	4	2
ρ_i	-	0,87	0,84	0,93	0,75	0,87	0,86	0,86	0,99

Таким образом, исходя из полученных результатов, можно сделать вывод, что ансамбль более простых моделей с использованием метода бэггинга дает существенное улучшение результатов по сравнению с более «полной» и сложной моделью многофакторного оценивания построенной на основе отношения (15).

Заключение

Результаты компьютерного имитационного моделирования показали, что при использовании одинаковой информации о предпочтительности альтернатив для эксперта метод с использованием ансамбля моделей на основе бэггинга более эффективен при построении модели многофакторного оценивания

Таблица 2

Значения функций полезности альтернатив

	$P_0(x_i)$	$P_1(x_i)$	$P_2(x_i)$	$P_3(x_i)$	$P_4(x_i)$	$P_5(x_i)$	$P_6(x_i)$	$P_{cp}(x_i)$
x_1	0,4236	0,4715	0,4135	0,4531	0,4076	0,3557	0,3888	0,4150
x_2	0,3357	0,3691	0,3649	0,3958	0,3262	0,3605	0,3058	0,3537
x_3	0,5966	0,5677	0,5384	0,4467	0,6108	0,5143	0,6187	0,5494
x_4	0,5936	0,5612	0,5396	0,5578	0,5858	0,5205	0,5945	0,5599
x_5	0,4807	0,4475	0,4818	0,5384	0,438	0,4411	0,4500	0,4661
x_6	0,3670	0,3865	0,4825	0,3656	0,3596	0,4741	0,3288	0,3995
x_7	0,5297	0,5533	0,5472	0,663	0,5168	0,5365	0,5158	0,5554
x_8	0,3183	0,2989	0,3464	0,276	0,3296	0,4075	0,3218	0,3300
x_9	0,6594	0,6425	0,7168	0,6147	0,6682	0,7329	0,6646	0,6733
x_{10}	0,5778	0,5593	0,6219	0,4281	0,5896	0,6203	0,5702	0,5649

чем метод прямого построения модели, учитывающий всю информацию об отношении порядка установленного на множестве альтернатив. Сравнение отношений порядка полученных с помощью этих моделей — R_0 и R_{cp} с исходным R^* показывает, что в первом случае количество инверсий рангов альтернатив $x_i \in X$, $i = \overline{1, 10}$ равно четырём ($\rho_0 = 0,87$), а во втором — двум ($\rho_{cp} = 0,99$).

Следует также отметить, что использование в ансамбле малого числа моделей ухудшает итоговый результат. В частности, для данного примера при использовании четырех и пяти моделей $\rho = 0,93$, а для трех — $\rho = 0,87$.

При построении ансамбля состоящего из моделей, учитывающих отношения предпочтения из трех альтернатив (общее количество таких отношений для приведенного примера равно $C_7^3 = 35$), аналогичный результат ($\rho = 0,99$) был достигнут при использовании восьми случайно отобранных моделей. Для ансамбля из семи моделей $\rho = 0,98$, для шести — $\rho = 0,96$.

В перспективе необходимо провести исследование зависимости точности итоговой модели от количества участвующих в ансамбле моделей и их «простоты» с целью выработки возможных рекомендаций относительно выбора этих двух параметров или их адаптивной настройки в зависимости от имеющихся исходных данных, т. е. количества альтернатив и факторов, которые их характеризуют.

Предложенный в работе метод целесообразно использовать в случае когда объем исходных данных невелик, что чаще всего характеризует ситуации экспертного оценивания или принятия решений. В этом случае применение аппарата искусственных нейронных сетей для решения такого рода задач будет неэффективным из-за существенной ограниченности количества исходных данных, которые могут быть использованы для их обучения.

Также представленный подход можно использовать для выработки некоторого согласованного мнения экспертов, т. к. ансамбль моделей можно строить и для отношений порядка, которые получены от экспертов, различной длины («полноты»), т. е. один эксперт — одна модель в ансамбле, а результат работы ансамбля — их согласованное мнение (ранжирование альтернатив в соответствии с их усредненной полезностью).

Список литературы:

- [1] *Петров К.Э.* Компараторная структурно-параметрическая идентификация моделей скалярного многофакторного оценивания / К.Э. Петров, В.В. Крючковский. — Херсон: Олди-плюс, 2009. — 294 с.
- [2] *Саати Т.* Принятие решений. Метод анализа иерархий / Т. Саати. — М.: Радио и связь, 1993. — 278 с.
- [3] *Саати Т.Л.* Принятие решений при зависимостях и обратных связях. Аналитические сети / Т. Л. Саати. — М.: Книжный дом «ЛИБРОКОМ», 2011. — 360 с.
- [4] *Figueira J., Mousseau V., Roy B.* ELECTRE methods / J. Figueira, S. Greco, M. Ehrgott (eds.), Multiple Criteria Decision Analysis: The State of the Art Surveys. — New York: Springer Science+Business Media, Inc., 2005. — P. 133–162.
- [5] *Corrente S.* Multiple criteria hierarchy process with ELECTRE and PROMETHEE / S. Corrente, S. Greco, R. Slowinski // Omega. — 2013. — Vol. 41, Issue 5. — P. 820–846.
- [6] *Bilbao-Terol A.* Using TOPSIS for assessing the sustainability of government bond funds / A. Bilbao-Terol, M. Arenas-Parrá, V. Canal-Fernandez, J. Antomil-Ibias. // Omega. — 2014. — 49. — P. 1–17.
- [7] *Подиновский В.В.* Парето-оптимальные решения многокритериальных задач / В.В. Подиновский, В.Д. Ногин. — М.: Наука, 1982. — 254 с.
- [8] *Ларичев О.И.* Теория и методы принятия решений, а также хроника событий в волшебной стране / О.И. Ларичев. — М.: Логос, 2000. — 294 с.
- [9] *Кини Р.Л.* Принятие решений при многих критериях: предпочтения и замещения / Р.Л. Кини, Х. Райфа. — М.: Радио и связь, 1981. — 560 с.
- [10] *Breiman L.* Bagging predictors / L. Breiman // Machine Learning. — 1996. — Vol. 24, Issue 2. — P. 123–140.
- [11] *Колмогоров А.Н.* О представлении непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одного переменного и сложения / А.Н. Колмогоров // Доклады АН СССР. — 1957. — Т. 5(114). — С. 953–956.
- [12] *Овезгельдыев А.О.* Построение модели индивидуального многофакторного оценивания с применением элементов МГУА и генетических алгоритмов / А.О. Овезгельдыев, К.Э. Петров // Кибернетика и системный анализ. — 2007. — №1. — С. 151–159.
- [13] *Тихонов А.Н.* Методы решения некорректных задач / А.Н. Тихонов, В.Я. Арсенин. — М.: Наука, 1986. — 288 с.
- [14] *Зуховицкий С.И.* Линейное и выпуклое программирование / С.И. Зуховицкий, Л.И. Авдеева. — М.: Наука, 1967. — 460 с.
- [15] *Брюс П.* Практическая статистика для специалистов Data Science / П. Брюс, Э. Брюс. — С.-Пб.: БХВ-Петербург, 2016. — 304 с.

Поступила в редколлегию 16.09.2019



І. В. Гребеннік¹, О. С. Чорна²

¹доктор технічних наук, професор, зав. каф. Системотехніки Харківського
Національного університету радіоелектроніки, Україна,
igor.grebennik@nure.ua, ORCID iD: 0000-0003-3716-9638

²асистент каф. Системотехніки Харківського Національного університету радіоелектроніки,
Україна, olha.chorna@nure.ua, ORCID iD: 0000-0001-6745-8137

ЦИКЛІЧНІ ПЕРЕСТАНОВКИ В МЕТОДАХ КОМБІНАТОРНОЇ ОПТИМІЗАЦІЇ НА ОСНОВІ ЦИКЛІЧНИХ ТРАНСФЕРІВ

Дана стаття присвячена стратегії розв'язання задач комбінаторної оптимізації, оснований на циклічних трансферах та властивостях множини циклічних перестановок. Циклічні трансфери – це один із відомих методів пошуку в околі, що дозволяє отримувати наближене рішення задач комбінаторної оптимізації. Існує загальний метод для пошуку в околі за допомогою циклічних трансферів. Він заснований на пошуку циклічного трансфера у допоміжному графі. Але особливості задачі, що розв'язується, деколи не дозволяють побудувати допоміжний граф та скористатися вже існуючими методами пошуку. У такому випадку ми пропонуємо використовувати множину циклічних перестановок та її властивості для побудови циклічних трансферів. Для демонстрації застосування, описаної у статті стратегії розв'язку задач комбінаторної оптимізації, було розв'язано задачу маршрутизації транспорту. Обчислювальні експерименти та їх результати наведені у статті.

КОМБІНАТОРНА ОПТИМІЗАЦІЯ, ЦИКЛІЧНІ ПЕРЕСТАНОВКИ, ЦИКЛІЧНІ ТРАНСФЕРИ, ЗАДАЧА МАРШРУТИЗАЦІЇ, КЛАСТЕРИЗАЦІЯ

Данная статья посвящена стратегии решения задач комбинаторной оптимизации, основанной на циклических трансферах и свойствах множества циклических перестановок. Циклические трансферы – это один из известных методов поиска в окрестности, позволяющий получать приближенное решение задач комбинаторной оптимизации. Известна общая схема поиска в окрестности с помощью циклических трансферов. Она основана на поиске циклического трансфера в вспомогательном графе. Но особенности решаемых задач иногда не позволяют построить вспомогательный граф и воспользоваться уже имеющимися методами поиска. В таком случае предлагается использовать множество циклических перестановок и его свойства для построения циклических трансферов. Для демонстрации стратегии решения задач комбинаторной оптимизации, описанной в статье, было произведено решение задачи маршрутизации транспорта. Вычислительные эксперименты и их результаты приведены в статье.

КОМБІНАТОРНА ОПТИМІЗАЦІЯ, ЦИКЛІЧЕСКІ ПЕРЕСТАНОВКИ, ЦИКЛІЧЕСКІ ТРАНСФЕРИ, ЗАДАЧА МАРШРУТИЗАЦІЇ, КЛАСТЕРИЗАЦІЯ

Permutations sets are very often considered in theoretical and applied research in the field of combinatorics and combinatorial optimization. By now many properties of permutations have been investigated, in particular those associated with the cyclic structure of permutations. To solve optimization problems on combinatorial sets, a number of methods have been developed and successfully used. Among them, there are methods for finding exact, approximate, and heuristic solutions. Among the search methods for an approximate solution, search methods in the neighborhood are widely used. Among the combinatorial search methods in the neighborhood, it is necessary to distinguish a group of methods that uses cyclic transfers. A general search scheme using cyclic transfers is known. It is based on the search for a cyclic transfer in the auxiliary graph. But the features of the tasks being solved sometimes do not allow constructing an auxiliary graph and using the existing search methods. In such cases, an alternative approach is proposed. This is a general strategy for solving problems using cyclic transfers, which is a modification of the strategy of Thompson and Orlin. We propose the construction of cyclic transfers based not on the use of an auxiliary graph, but based on the set of cyclic permutations and its properties to obtain cyclic transfers. To demonstrate the strategy for solving combinatorial optimization problems described in the article, one of the vehicle routing problems was solved. Computational experiments were conducted to demonstrate the effectiveness of the proposed strategy. Their results are given in the article.

КОМБІНАТОРНА ОПТИМІЗАЦІЯ, ЦИКЛІЧЕСКІ ПЕРЕСТАНОВКИ, ЦИКЛІЧЕСКІ ТРАНСФЕРИ, ЗАДАЧА МАРШРУТИЗАЦІЇ, КЛАСТЕРИЗАЦІЯ

Вступ

При аналізі багатьох наукових і прикладних задач, що мають дискретну структуру, як засіб математичного моделювання виступають комбінаторні множини. Найбільш поширеною серед них є множина перестановок і її різні підмножини. Додаткові обмеження на параметри в таких задачах дозволяють виділити наступні відомі в літературі класи множин перестановок [1–6]: перестановки різних елементів, перестановки з повтореннями, перестановки з n елементів, k з яких

різні, циклічні перестановки, перестановки кортежів, композиції перестановок, перестановки містять (що не містять) шаблон (pattern), поліперестановки та інші.

Для вирішення задач оптимізації на комбінаторних множинах розроблені та успішно застосовуються багато методів. Серед них є методи для знаходження точних, наближених та евристичних рішень [1–4]. Доцільність застосування різних методів завжди спирається на особливості даної задачі, її обмеження та

вхідні данні. Основні методи знаходження наближеного рішення можуть бути розділені на наступні групи: методи з використанням точних методів, жадібні алгоритми, методи випадкового пошуку та методи пошуку в околі. Методи рішення, які є модифікаціями точних методів, як правило, засновані на методі гілок та меж, методі послідовного аналізу варіантів або на методах відсікання [1].

Серед методів пошуку наближеного рішення широкого застосування набули методи пошуку в околі – це великий клас удосконалюючих алгоритмів, який базується на пошуку кращого рішення в деякій визначеній області навколо стартового рішення задачі комбінаторної оптимізації [7].

Серед комбінаторних методів пошуку в околі необхідно виділити групу методів, що використовують циклічні трансфери [8–13]. Теорія циклічних трансферів є основою загальної стратегії пошуку, подібної до локальних методів пошуку, як-то Simulated Annealing або Tabu Search. Методи на основі циклічних трансферів можуть використовуватись як самостійні методи пошуку у широкому околі, так і як допоміжні кроки інших алгоритмів. Існує ряд робіт, у яких описано застосування методів з використанням циклічних трансферів до широкого спектру задач комбінаторної оптимізації, включаючи узагальнену та квадратичну задачу про призначення [10; 11], проблеми маршрутизації транспортних засобів [9], задачу про мінімальне кістякове дерево [12], а також задачу формування розкладу [13].

Необхідно зауважити, що застосування циклічних трансферів ефективне для вирішення задач комбінаторної оптимізації з певними загальними особливостями. Ці задачі зазвичай характеризуються неможливістю знайти точний розв'язок задачі у наявних обчислювальних можливостях, та застосуванням кластеризації вхідних даних у методах розв'язання. Застосування циклічних трансферів у рамках вирішення задач комбінаторної оптимізації є окремою NP-важкою задачею. Метою даної роботи є розробка стратегії розв'язання задач комбінаторної оптимізації, яка використовує циклічні трансфери, побудовані на основі циклічних перестановок.

1. Теорія циклічних трансферів

Дамо огляд основних, необхідних нам результатів, з теорії циклічних трансферів, згідно робіт Thompson, Orlin та Thompson, Psaraftis [8; 9]. При вирішенні задач комбінаторної оптимізації за допомогою циклічних трансферів розв'язання відбувається у дві стадії. Перша стадія – це стартове розбиття комбінаторної множини на кластери, що не перетинаються, одним із відомих методів кластеризації та формування стартового рішення задачі на даному розбитті одним із відомих методів. Це рішення може бути наближеним, або

евристичним. Стартове розбиття, наприклад, може залежати від обмежень задачі, як то, кількість транспортних засобів у задачах маршрутизації, кількість груп, потоків або курсів у задачах формування розкладу. Ефективність стартового рішення оцінюється за допомогою цільової функції задачі комбінаторної оптимізації. В рамках теорії циклічних трансферів в якості показника ефективності розглядається вартість рішення, яка мінімізується.

Друга стадія – застосування циклічних трансферів з метою покращення стартового рішення. Циклічний трансфер – це циклічна перестановка деяких обраних елементів між кластерами. Вибір елементів кластерів, які будуть елементами циклічної перестановки, та формування циклічного трансферу – нетривіальна задача, для вирішення якої можуть застосовуватись різні методи. Різниця між ефективністю стартового рішення та рішення, отриманого після застосування циклічного трансфера, називається вартістю циклічного трансфера. Якщо нове рішення має меншу вартість, то трансфер, за допомогою якого воно було отримано, називають циклічним трансфером від'ємної вартості. Застосування знайденого циклічного трансферу від'ємної вартості до стартового рішення відповідає переміщенню елементів між кластерами, завдяки чому утворюється більш ефективне рішення.

Thompson і Orlin розробили загальний метод для пошуку в околі за допомогою циклічних трансферів. Він заснований на заміні пошуку циклічного трансфера від'ємної вартості на пошук допустимого циклу від'ємної вартості у допоміжному графі. Поняття допоміжного графа було введено в Thompson і Orlin [8] і далі розглянуто Thompson і Psaraftis [9]. Допоміжний граф формується з елементів вхідних даних, які утворюють вершини графа. Ребра допоміжного графу з'єднують тільки вершини, у яких відповідні елементи, належать різним кластерам у початковому розбитті. А вартість кожного ребра (i, j) , це те, наскільки зміниться вартість всього рішення якщо із відповідного кластера видалити елемент i та додати до кластера елемент j . Важливим у всіх випадках є те, що використання допустимого циклу з допоміжного графа відповідає переходу до сусіднього рішення від поточного рішення. Вартість циклічного трансферу, оснований на допустимому циклі, точно дорівнює вартості відповідного циклу у графі. Для циклів, які не є допустимими, відповідність вартості циклу у графі вартості циклічного трансферу не гарантується або відповідне переміщення елементів може бути неможливим.

У багатьох випадках неможливо точно оцінити вартість ребер у допоміжному графі, і, як наслідок, неможливо точно оцінити вартість циклів у графі та відповідних їм циклічних трансферів. Допоміжний граф,

який був запропонований Thompson і Orlin, можливо побудувати тільки тоді, коли є можливість точно оцінити наскільки зміниться вартість цільової функції деякого рішення вихідної задачі при таких двох діях, застосованих до цього рішення. Перша дія – це вилучення з будь-якого кластеру будь-якого елементу, друга дія – це додавання до будь-якого кластеру будь-якого елементу. Без точної оцінки вартості цих дій, неможливо розрахувати вартість ребер у допоміжному графі. Без вартості ребер неможлива побудова допоміжного графу та пошук циклу від'ємної вартості на ньому. Без допустимого циклу від'ємної вартості у допоміжному графі, якщо дотримуватись методу, описаного Thompson і Orlin, неможливо знайти циклічний трансфер від'ємної вартості.

2. Ідея альтернативного формування циклічних трансферів

У випадках, описаних вище, коли знаходження циклічного трансферу від'ємної вартості на основі допоміжного графа неможливе, пропонується альтернативний підхід. Це загальна стратегія розв'язання задач з використанням циклічних трансферів, яка є модифікацією стратегії Thompson і Orlin [8]. В ній побудова циклічних трансферів заснована не на використанні допоміжного графа, а на застосуванні множини циклічних перестановок та її властивостей для отримання циклічних трансферів від'ємної вартості. Циклічні трансфери від'ємної вартості, побудовані на базі циклічних перестановок, дають можливість знаходження кращого рішення у широкому околі стартового рішення.

В рамках реалізації цієї стратегії, з обраних кластерів вибирають по одному елементу, запропонованому для переміщення в інші кластери. Відбір елементів з кластерів відбувається на підставі попереднього аналізу даних, наприклад, за допомогою евристичних правил, що залежать від предметної галузі. Відібрані елементи є породжуючими для множини циклічних перестановок. У цій множині кожна циклічна перестановка відповідає одному циклічному трансферу. Побудована таким чином множина циклічних перестановок описує всі можливі циклічні трансфери вибраних елементів. Отримані таким чином циклічні трансфери застосовуються до стартових кластерів. Після застосування кожної циклічної перестановки ми отримуємо нове рішення вихідної задачі в околі стартового рішення. Вихідна задача вирішується на нових кластерах та оцінюється, чи змінилась вартість рішення порівняно зі стартовим та у який бік, збільшення чи зменшення. Таким чином буде отримана низка нових рішень вихідної задачі у широкому околі стартового рішення. Кількість цих рішень відповідає потужності множини циклічних перестановок. З усіх отриманих нових рішень обираємо найкраще та робимо крок методу циклічних трансферів. Тобто обране нове рішення стає стартовим для наступного кроку.

3. Постановка задачі та загальна стратегія розв'язання

Виходячи з мети даної роботи, сформуємо стратегію розв'язання задач комбінаторної оптимізації, для яких доречна кластеризація вхідних даних, за допомогою циклічних трансферів, у випадках, коли неможливо побудувати допоміжний граф, для поліпшення стартового рішення. Замість допоміжного графа пропонується використання циклічних перестановок та їх властивостей, для опису і формалізації переміщення елементів між кластерами. Множину циклічних перестановок, та відповідні циклічні трансфери, пропонується формувати за допомогою результатів досліджень комбінаторної множини циклічних перестановок, описаних у роботах [14; 15].

Розглянемо задачу комбінаторної оптимізації:

$$F(x) \rightarrow \min_{x \in D \subset E}, \quad (1)$$

де E – комбінаторна множина у деякому комбінаторному просторі Y , $x = (x_1, x_2, \dots, x_k) \in E$, $F: E \rightarrow R^1$, а множина D формується за допомогою системи обмежень на змінні $x \in E$. Для розв'язання задачі (1) застосуємо стратегію, яка використовує циклічні трансфери. Множину E можна представити у вигляді кластерів, що не перетинаються $E = \bigcup E^i$, $E^i \in E$, $i = 1, 2, \dots, n$, за деякою ознакою. Для будь-якого розбиття на кластери множини E , виходячи з певних міркувань, можна сформувати множину $A = \{a_1, a_2, \dots, a_n\}$, таку що $a_i \in E^i$, $i \in J_n = \{1, 2, \dots, n\}$. Ця множина буде множиною породжуючих елементів для множини циклічних перестановок $P_n^C(A)$, $p_i \in P_n^C$, $p_i = (a_{i_1}, a_{i_2}, \dots, a_{i_n})$, $\{i_1, i_2, \dots, i_n\}$ – упорядкування індексів J_n .

На першому етапі розв'язання задачі формується стартове розбиття $E^0 = [E_1^0, E_2^0, \dots, E_m^0]$, на кластери, що не перетинаються, за деякою ознакою. Нехай отримано допустиме стартове рішення задачі x^0 за умови стартового розбиття E^0 деяким методом комбінаторної оптимізації, ефективність стартового рішення дорівнює $F(x^0 | E^0)$.

Другим етапом відбувається формування множини $A^0 = \{a_1, a_2, \dots, a_n\}$ породжуючих елементів для множини циклічних перестановок $P_n^C(A)$, такої що $a_i \in E_i^0$, $i \in J_n$. Вибір елементів множини A^0 залежить від особливостей задачі, що розв'язується. Деякі евристичні прийоми для обрання елементів множини A^0 пропонуються та обговорюються нижче. Кожен елемент множини $p_i \in P^C(A^0)$ – це один циклічний трансфер, що описує, які елементи будуть переміщені між якими кластерами у стартовому розбитті. Після застосування циклічної перестановки $p_i \in P_n^C$ до початкового розбиття E^0 , отримуємо розбиття $E^j(p_j)$, та відповідну задачу комбінаторної оптимізації:

$$F(x | E^j(p_j)) \rightarrow \min_{x \in D \subset E}. \quad (2)$$

Розв'язуємо задачу (2) за умови розбиття $E^j(p_j)$ обраним методом комбінаторної оптимізації. Процес

зміни початкового розбиття множини E на кластери за допомогою циклічних трансферів та вирішення задачі на альтернативних розбиттях може бути повторювано багаторазово. Розглянемо задачу пошуку циклічної перестановки з множини $P^C(A^0)$, такої що відповідний циклічний трансфер дає можливість отримати найкраще рішення вихідної задачі, з усіх розбиттів $E^j(p_j)$, $j \in J_n$.

Для цього сформулюємо наступну задачу комбінаторної оптимізації на множині циклічних перестановок:

$$F(x|E^j(p)) \rightarrow \min_{p \in P^C(A)} . \quad (3)$$

Розв'язання задачі (3) залежить від особливостей та властивостей задачі (1), області її застосування, та від властивостей множини циклічних перестановок, які були досліджені у роботах [14; 15]. Для випадку, коли функція $F(x|E^j(p))$ лінійна, у роботах [16; 17] описано та реалізовано методи пошуку точного, наближеного та евристичного розв'язку задачі (3). Для розв'язання задачі (3) з лінійною функцією цілі пропонується використовувати метод гілок та меж та його модифікації, випадковий пошук, та його модифікацію на основі властивостей циклічних перестановок.

Запропонована модифікація стратегії Thompson і Orlin є досить загальною, особливості її імплементації для конкретної задачі комбінаторної оптимізації залежать від властивостей задачі, що розв'язується. В класі задач комбінаторної оптимізації на кластерних даних, розглянемо клас задач маршрутизації транспорту для демонстрації ефективності стратегії поліпшення рішення задач за допомогою циклічних трансферів.

4. Задачі маршрутизації транспорту

Розглянемо клас задач маршрутизації транспорту (vehicle routing problems, VRP). Це широкий клас задач оптимізації, який грає ключову роль в транспортній логістиці. Задачам маршрутизації транспорту присвячено багато книг, статей та монографій, зокрема, [11; 18–22].

В таких задачах для заданого парку транспортних засобів, що знаходяться в одному або декількох депо, треба побудувати маршрути та доставити вантажі до споживачів з урахуванням різних обмежень: на вантажопідйомність транспорту, на часові вікна (time windows), в які клієнти можуть обслуговуватися, тощо.

В реальних задачах виникає багато додаткових обмежень, яким відповідають різні підкласи задач VRP [23–26], серед яких capacitated VRP (CVRP), VRP with Time Windows (VRPTW), Multiple Depot VRP (MDVRP), Split Delivery VRP (SDVRP), Periodic VRP (PVRP), Stochastic VRP (SVRP), VRP with Satellite Facilities (VRPSF), та багато інших.

Описані класи задач Vehicle Routing Problem стосуються різних практичних ситуацій, але зосереджуються на спільній проблемі – ефективному використанні

парку транспортних засобів, що повинні розвозити вантажі.

Одним з класів задач маршрутизації транспорту є задачі вивозу і доставки (Vehicle Routing Problems with Pickups and Deliveries, VRPDP, або Pickup and Delivery Problem, PDP) [20; 27], де транспортні засоби обслуговують замовлення клієнтів (transportation requests), в кожному з яких треба забрати вантаж (або декілька вантажів) в пункті завантаження (pickup) та перевезти його у відповідний пункт доставки (delivery). Задача PDP виникає в багатьох контекстах, таких як міські кур'єрські послуги, системи транспортування від дверей до дверей та послуги таксі.

Важливим для реальних задач класом обмежень є обмеження на завантаження транспортного засобу (loading constraints). В найпростіших моделях враховується вантажопідйомність кожного транспортного засобу і маса кожного вантажу.

В реальних задачах також дуже важливо забезпечувати відсутність блокування вантажу в пункті доставки: вантаж має стояти так, щоб його можна було легко дістати, і не повинен бути заблокований іншими вантажами. Щоб врахувати такі вимоги, можна накласти обмеження на порядок завантаження і розвантаження, описавши вантажне відділення транспортного засобу як стек або як буфер.

Розміри можуть бути задані двома вимірами (2D loading constraints), коли задається довжина і ширина вантажів та вантажних відсіків транспортних засобів, а їх висота вважається однаковою, або трьома вимірами (3D loading constraints). При цьому тримірні обмеження завантаження (3D loading constraints) можуть бути описані за допомогою математичного апарату Ф-функцій, для формального описання умови взаємного перетину n -мірних об'єктів [27–29].

У роботі [29] наведено рішення задачі Pickup and Delivery Problem в постановці, що одночасно враховує послідовність завантаження тримірних контейнерів та порядок відвідування пунктів вивозу і доставки, виключаючи можливість блокування одних контейнерів іншими при розвантаженні та забезпечуючи стійкість завантаження. Запропонована дворівнева стратегія розв'язання досліджуваної задачі. Верхній рівень – вирішення задачі кластеризації пунктів вивозу і доставки на задану кількість кластерів k за допомогою методу k -середніх [30]. Це дозволяє поставити у відповідність кожному автомобілю множини пунктів, які він має відвідати. Нижній рівень – побудова маршруту найменшої вартості для кожного автомобіля за допомогою евристики променевого пошуку на етапі розширення рекурсивного дерева. Розроблене програмне забезпечення для отримання точного та евристичного розв'язків досліджуваної задачі.

Особливістю реалізації описаної дворівневої стратегії у роботі [27] є вибір в якості методу кластеризації простого, але не завжди ефективного методу

k-середніх, який має деякі недоліки, наприклад, невизначеність вибору початкових центрів кластерів та як наслідок не завжди оптимально сформовані кластери. Тому джерелом покращення цієї стратегії є використання інших методів та алгоритмів кластеризації, або покращення кластеризації, отриманої за допомогою методу k-середніх.

5. Підвищення ефективності розв'язку задачі PDP за допомогою циклічних трансферів

Застосуємо для досліджуваної задачі PDP запропоновану стратегію розв'язання задач комбінаторної оптимізації за допомогою циклічних трансферів. Для цього, після кластеризації множини пунктів pickup and delivery за допомогою методу k-середніх, пропонується вибирати з кожного кластера по одній парі пунктів pickup and delivery які будуть формувати породжуючу множину циклічних перестановок.

Сформована таким чином множина циклічних перестановок $P_n^C(A)$ буде джерелом циклічних трансферів, які застосовуються до стартового розбиття. Таким чином зі стартового розбиття E^0 буде отримано $(n-1)!$ альтернативних розбиттів, або «сусідів». Для цих $(n-1)!$ альтернативних розбиттів розв'язуємо задачу (3), та обираємо найкраще розбиття $E^*(p^*)$. Задачу PDP розв'язуємо на розбитті $E^*(p^*)$ та оцінюємо вартість розв'язку.

Для вибору пар вершин pickup and delivery з кожного кластера, було випробувано дві евристики. Першою була евристика з урахуванням відстані між вершинами pickup and delivery та центром кластера, до якого вони належать. Тобто елементом породжуючої множини ставала та пара pickup and delivery в кластері, для якої сума відстаней від центру кластера до точки pickup та від центру кластера до точки delivery була найбільша. Але ця евристика не надала суттєвого зменшення вартості функції цілі після застосування циклічних трансферів. Другою була випробувана евристика з урахуванням відстані між вершинами pickup and delivery з різних кластерів. Тобто парою для циклічного трансфера буде та пара вершин у кластері відстань від якої до будь-якої вершини pickup and delivery з іншого кластеру буде мінімальна. Завдяки цій евристичі вдалось отримати кращі результати. Нижче наведені результати обчислювальних експериментів з використанням другої евристики.

6. Результати проведених експериментів

Описана вище стратегія вирішення задачі маршрутизації була реалізована програмно. Для проведення експериментів з покращення рішення задачі PDP за допомогою циклічних трансферів, у програмний комплекс з роботи [29] додано можливість зміни стартового розбиття задачі PDP. Також програмно реалізовані наступні кроки: вибір елементів, які будуть переміщені між кластерами, формування з них

породжуючої множини для множини циклічних перестановок, застосування циклічних трансферів, відповідних елементам множини циклічних перестановок, вирішення задачі PDP на альтернативних розбиттях, пошук найкращого значення функції цілі з отриманих. Для отримання породжуючої множини циклічних перестановок з елементів кластерів було випробувано дві евристики. Проведені обчислювальні експерименти, які підтвердили доцільність використання однієї з цих евристик для вирішення поставленої задачі.

Проведені експерименти для 3, 4 та 5 кластерів. Кількість пар pickup and delivery змінювалась від 15 до 50. Для невеликої кількості вершин та для 3 та 4 кластерів ширина пошукового променя була 5%. Коли, у зв'язку зі збільшенням розмірності задачі, витрати часу на її вирішення зросли до неприйнятних показників, ширину пошукового променя було зменшено до 1%.

Нижче наведені результати проведених обчислювальних експериментів. Позначення у таблиці: N – розмірність задачі, тобто кількість пар pickup-delivery, SR (success rate) – відсоток задач, для яких вдалось зменшити значення функції цілі, в результаті одноразового застосування циклічних трансферів, AVG (Cost) – середнє отримане значення функції цілі, для усіх вирішених задач даної розмірності, після застосування циклічного трансферу, AVG(Benefit) – середнє значення, наскільки вдалось зменшити значення функції цілі, Benefit % – AVG(Benefit) у відсотковому співвідношенні до AVG(Cost), RWB – ширина пошукового променя.

Таблиця 1

Результати для 3 кластерів

N	SR	AVG(Cost)	AVG(Benefit)	Benefit %
15	45%	5 084,549	306,270	6%
20	40%	6095,300	529,092	8,6%
30	55%	8057,091	583,622	7,2%
40	60%	8885,765	434,511	4,8%

Таблиця 2

Результати для 4 кластерів

N	SR	RWB	AVG(Cost)	AVG(Benefit)	Benefit %
20	35%	5%	6 582,775	514,715	7,8%
30	45%	5%	8 396,427	657,150	7,8%
40	55%	1%	9 637,011	687,679	7,1%
50	70%	1%	11 218,095	809,446	7,2%

Таблиця 3

Результати для 5 кластерів

N	SR	RWB	AVG(Cost)	AVG(Benefit)	Benefit %
40	60%	1%	10 540,754	887,174	8,4%
50	70%	1%	11 905,301	1 067,554	9%

В задачах розмірності від 15 до 50 вдалось отримати зменшення вартості доставки вантажів від 5 до 9%. Для задач розмірності більше 30 використання ширини пошукового променя, більше 1% виявилось неефективним з точки зору обчислювальних можливостей.

Необхідно відзначити, що з ростом розмірності задачі та кількості кластерів, вигода, тобто зменшення вартості функції цілі, та відсоток задач, для яких вдалось зменшити значення функції цілі зростають. Таким чином дана стратегія поліпшення рішення задач комбінаторної оптимізації є перспективною, та потребує подальшого розвитку.

Висновки

У цій роботі розроблена модифікація стратегії застосування циклічних трансферів з метою покращення розв'язку задач комбінаторної оптимізації на кластерних даних для випадків, коли неможливо побудувати допоміжний граф. В задачах зазначеного класу запропоновано використовувати множину циклічних перестановок та властивості цієї множини для формування циклічних трансферів та знаходження найкращого циклічного трансферу від'ємної вартості.

Запропонована альтернативна стратегія формування циклічних трансферів від'ємної вартості що спирається на властивості множини циклічних перестановок. Описані евристичні правила для вибору елементів, які будуть переміщені між кластерами.

Запропонована стратегія реалізована на прикладі задач маршрутизації PDP. До існуючого розв'язку задачі маршрутизації PDP з роботи [29] додана можливість покращити вартість функції цілі за допомогою циклічних трансферів сформованих на основі множини циклічних перестановок. Пошук найкращого значення функції цілі задачі PDP на альтернативних розбиттях, було реалізовано програмно.

Були проведені обчислювальні експерименти, які показали ефективність запропонованої стратегії. Було отримане зменшення вартості доставки вантажів.

Список літератури:

- [1] Сергієнко І. В. Математичні моделі і методи розв'язання задач дискретної оптимізації // К.: Наук. думка, 1988. – С. 472.
- [2] Стоян Ю.Г., Яковлев С.В. Математичні моделі і оптимізаційні методи геометричного проектування // К.: Наук. думка, 1986. – С. 268.
- [3] Sergienko I. V., Hulianytskyi L. F., Sirenko S.I. Classification of applied methods of combinatorial optimization // Cybernetics and Systems Analysis. – 2009. – № 5(45). – P. 732–741.
- [4] Papadimitriou C.H., Steiglitz K. Combinatorial Optimization: Algorithms and Complexity // Prentice-Hall, Englewood Cliffs, NJ, 1982 – P. 528.
- [5] Стоян Ю.Г., Ємець О.О. Теорія і методи евклідової комбінаторної оптимізації // Київ.: Ін-т систем. досліджень освіти, 1993. – С. 188.
- [6] Сергієнко І.В., Шило В.П. Сучасні підходи до розв'язання складних задач дискретної оптимізації // Проблеми управління і інформатики. – 2016. – № 1. – С. 32-40.
- [7] Ahuja R.K., Ergun ., Orlin J.B., Punnen A.P. A survey of very large-scale neighborhood search techniques // Discrete Applied Mathematics. – 2002. – № 1–3(123). – P.75–102.
- [8] Thompson P.M., Orlin J.B. The theory of cyclic transfers: working paper // Massachusetts Institute of Technology, Operations Research Center, 1989 – P. 37.
- [9] Thompson P.M., Psarafitis H.N. Cyclic transfer algorithms for multivehicle routing and scheduling problems // Operations Research. – 1993. – № 5(41). – P. 935-946.
- [10] Yagiura M., Ibaraki T., Glover F.W. An ejection chain approach for the generalized assignment problem // INFORMS Journal on Computing. – 2004. – № 2(16). – P. 133-151.
- [11] Rego C., James T.L., Glover F.W. An ejection chain algorithm for the quadratic assignment problem // Networks. – 2010. – № 3(56). – P. 188-206.
- [12] Ahuja R.K., Orlin J.B., Sharma D. Multi-exchange neighborhood structures for the capacitated minimum spanning tree // Mathematical Programming. – 2001. – № 1(91). – P. 71–97.
- [13] Post G., Ahmadi S., Geertsema F. Cyclic transfers in school timetabling // OR Spectrum. – 2012. – № 1(34). – P. 133–154.
- [14] Grebennik I.V., Chorna O.S. Influence of Certain Transpositions on the Cyclic Structure of Permutations // Cybernetics and Systems Analysis. – 2015. – № 6(51). – P. 947–955.
- [15] Grebennik I.V., Чорна О.С. Спеціальні транспозиції елементів перестановок і властивості композицій // Кібернетика і системний аналіз. – 2017. – № 1(53). – С.79–90.
- [16] Grebennik I. V., Литвиненко А. С., Тімова О. С. Оптимізація лінійної функції на множині циклічних перестановок // Біоніка інтелекту. – 2012.– № 2 (67). – С. 8–12.
- [17] Grebennik I., Baranov A., Chorna O., Gorbacheva E. Optimization of linear functions on a cyclic permutation based on the random search // An International Quarterly Journal on Economics of Technology and Modelling Processes . – 2016. – № 3(5). – P. 211–216.
- [18] Kumar S., Panneerselvam R. A survey on the vehicle routing problem and its variants // Intelligent Information Management. – 2012.– № 3(4). – P. 66–74.
- [19] Pillac V., Gendreau M., Guéret C., Medaglia A. L. A review of dynamic vehicle routing problems // European Journal of Operational Research. – 2013. – № 1(225). – P. 1–11.
- [20] Toth P., Vigo D. Vehicle routing: problems, methods, and applications // Society for Industrial and Applied Mathematics, 2001 – P. 481.
- [21] Панишев А. В. Моделі і методи оптимізації замкнених маршрутів на транспортній мережі: монографія // ЖГТУ, Житомир, 2014 – С. 316.
- [22] Cordeau J., Laporte G., Ropke S. Recent models and algorithms for one-to-one pickup and delivery problems // The vehicle routing problem: latest advances and new challenges. – 2008. – № 43. – P. 327–357.
- [23] Gendreau M., Laporte G., Potvin J. Metaheuristics for the capacitated VRP // The vehicle routing problem. Society for Industrial and Applied Mathematics. – 2001. – P. 129–154.
- [24] Belhaiza S., M'Hallah R., Brahim G.B. A new Hybrid Genetic Variable Neighborhood search heuristic for the Vehicle Routing Problem with Multiple Time Windows // 2017 IEEE Congress on Evolutionary Computation (CEC). – 2017. – P. 1319–1326.
- [25] Yao B., Hu P., Zhang M., Wang S. Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem // SIMULATION: Transactions of The Society for Modeling and Simulation International. – 2013. – № 6(89). – P. 762–770.
- [26] Gendreau M., Jabali O., Rei W. Stochastic vehicle routing problems // Vehicle routing: problems, methods, and applications. – 2014. – P. 213–239.
- [27] Grebennik I. V., Pankratov A. V., Chugay A. M., Baranov A. V. Packing n-dimensional parallelepipeds with the feasibility of changing their orthogonal orientation in an n-dimensional parallelepiped // Cybernetics and Systems Analysis. – 2010. – № 5(46). – P. 793–802.
- [28] Stoyan Y. G., Gil N. I., Scheithauer G., Pankratov A., Magdalena I. Packing of convex polytopes into a parallelepiped // Optimization. – 2005. – № 2(54). – P. 215–235.
- [29] Dupas R., Grebennik I., Lytvynenko O., Baranov O. An Heuristic Approach to Solving the one-to-one Pickup and Delivery Problem with Three-dimensional Loading Constraints // International Journal of Information Technology and Computer Science Information Technology and Computer Science(IJITCS). – 2017. – № 10(9). – P. 1–12.
- [30] Fabregas A. C., Gerardo B. D., Tanguilig III B. T. Enhanced Initial Centroids for K-means Algorithm // International Journal of Information Technology and Computer Science(IJITCS). – 2017. – № 1(9). – P. 26–33.

Надійшла до редколегії 4.10.2019

УДК 519.62

DOI 10.30837/bi.2019.2(93).08

Р.Т. Гасько¹, Н.Б. Шаховська²¹ НУ «ЛП», м. Львів, Україна, roman.t.hasko@lpnu.ua;² НУ «ЛП», м. Львів, Україна, nataliya.b.shakhovska@lpnu.ua

НАВЧАЛЬНА РОБОТОТЕХНІЧНА ПЛАТФОРМА ДЛЯ FOG/EDGE/AIOT

У статті описано навчальну трьох-рівневу робототехнічну платформу для використання в навчальному процесі. Пропонована платформа є органічною частиною сучасної спіралі розвитку інформаційних процесів та систем і дозволяє покращити якість навчального процесу, збільшити його орієнтацію на отримання практичного досвіду та віддзеркалює актуальний стан синергії AI через Edge/Fog/Cloud з IoT.

ХМАРНІ ОБЧИСЛЕННЯ, ТУМАННІ ОБЧИСЛЕННЯ, ЗМІШАНЕ НАВЧАННЯ

The article describes a three-level training robotics platform for use in the learning process. The proposed platform is an organic part of the modern spiral of development of information processes and systems and allows to improve the quality of the educational process, to increase its orientation towards practical experience and to reflect the current state of AI synergy through Edge / Fog / Cloud with IoT.

CLOUD COMPUTING, FOG COMPUTING, BLENDED LEARNING

В статье описано учебную трех-уровневую робототехническую платформу для использования в учебном процессе. Предлагаемая платформа является органической частью современной спирали развития информационных процессов и систем и позволяет улучшить качество учебного процесса, увеличить его ориентацию на получение практического опыта и отображает актуальное состояние синергии AI через Edge / Fog / Cloud с IoT.

ОБЛАЧНЫЕ ВЫЧИСЛЕНИЯ, ТУМАННЫЕ ИСЧИСЛЕНИЯ, СМЕШАННОЕ ОБУЧЕНИЕ

Вступ

Навчальний процес підготовки інженера в галузі інформаційних технологій вимагає сучасного програмно-апаратного забезпечення. Розвиток IT, як і інші еволюційні процеси, можна охарактеризувати спіралеподібним розвитком. Періодично йде повернення до подібних рішень, але на вищому технологічному рівні. Прикладом є розвиток мейнфреймів (ЕС ЕВМ та ІВМ) з розвинутою системою терміналів для роботи операторів. Обчислювальні можливості терміналів поступово зростали та еволюціонували в “розумні” термінали, настала ера персональних комп’ютерів (ПК). Автономні ПК стали об’єднувати в мережі, з розвитком яких Інтернет став самодостатнім явищем. Відповідно, домінування архітектура “клієнт-сервер” можна вважати в певній мірі відповідником “термінали-мейнфрейм”, перенесеним на вищий технологічний рівень. Поступово окремі сервери стали об’єднувати в кластери та розвинулась система розподілених “хмарних” обчислень (Cloud), а Інтернет почав також об’єднувати величезне число периферійних пристроїв – Інтернет Речей (IoT, Internet of Things). Постійно зростаючі обсяги інформації, зокрема відеопотоки з високою роздільною здатністю від мережі різноманітних камер спостереження, та, відповідно, необхідність їх опрацьовувати в режимі, наближеному до реального часу, зумовили черговий виток спіралі розвитку з делегування частини обчислювальних можливостей з рівня Cloud безпосередньо на периферійні пристрої, виникнення Edge/Fog computing [1, 2] та виникнення “розумного” IoT – AIoT як результату синергії IoT та штучного інтелекту (AI).

Паралельно йде швидкий еволюційний розвиток принципово нової для людської цивілізації концепції

– створення роботів, наділених певним штучним інтелектом, орієнтованим переважно у вузьких сферах, як орієнтування та автономне пересування різними теренами, виконання специфічних технологічних функцій та інші види діяльності, що не вимагають високого рівня власного інтелекту.

Черговим етапом розвитку очікується поява “повноцінного” штучного інтелекту — Сильного Штучного Інтелекту, відомого як Strong AI в англійських джерелах [3], або full AI [4] або Artificial general intelligence (AGI) [5].

Метою статті є опис принципів змішаного навчання з використанням робототехнічної платформи та штучного інтелекту в Інтернеті речей.

1. Опис навчальної апаратної платформи

Для підготовки спеціалістів у системі вищої освіти пропонується використовувати спеціалізовану навчальну платформу, що включає програмно-апаратне забезпечення для послідовного вивчення навчальних курсів пов’язаних загальним напрямом штучний інтелект (AI), таких як Інтернет Речей, Хмарні Обчислення, Робототехніка тощо. Описане рішення дозволяє будувати як комплексні лабораторні роботи, так і повноцінні закінчені проекти в розрізі концепції tripled learning [6], а також виконувати науково-дослідні проекти. Описувана платформа є модульна та містить кілька рівнів складності для оптимальної конфігурації відповідно до навчальних програм конкретного курсу.

Перший рівень представляє собою універсальну автономну рухому платформу T-Vot на основі популярного навчального мікроконтролера Arduino з сенсорами відстані, руху та кольору.

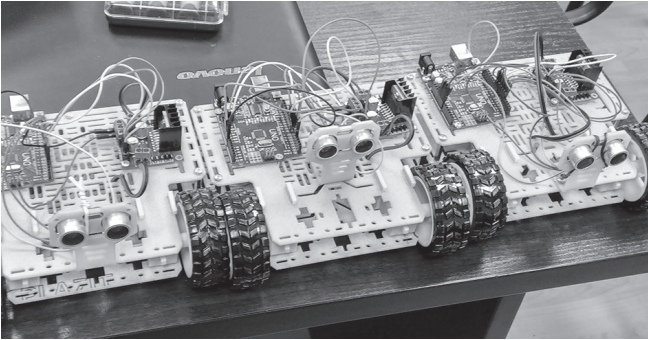


Рис. 1. Зовнішній вигляд навчальної робототехнічної платформи T-Vot першого рівня з Arduino та ультразвуковим дальноміром

Також передбачено можливість використання потужнішого мікроконтролера з лінійки STM32. Зазначений рівень вже успішно впроваджено в навчальний процес протягом трьох років вивчення в курсі “Алгоритмізація та Програмування. Частина 1” першого року навчання кафедри Систем Штучного Інтелекту Н університету “Львівська політехніка”.

Другий рівень конструктивно розміщується зверху першого та відповідає за більш інтелектуальні можливості, як розпізнавання зображень, аналіз об’єктів та комунікація з Cloud. Технічно цей рівень представляє собою мікрокомп’ютер Raspberry Pi 3/4 з відеокамерою та OpenCV/OpenVINO. Головним класом задач є машинне навчання, тобто обчислення попередньо натренованих та оптимізованих нейронних мереж, як правило з використанням Keras/TensorFlow. Оскільки обчислювальних ресурсів Raspberry Pi не вистачає для повноцінної підтримки задач штучного інтелекту, зокрема при переході від хмарних (Cloud) до периферійних (Edge/Fog Computing) обчислень, наступним компонентом цього рівня є окремий спеціалізований USB-модуль з орієнтованою на нейромережі архітектурою. Приклади такого підходу – Myriad™ X VPU від Intel [7] – третє покоління та найдосконаліший VPU від Movidius™, компанії Intel або Google Edge Coral Accelerator [8]. Myriad™ X VPU від Intel вперше в своєму класі має Neural Compute Engine – спеціалізований апаратний прискорювач для глибоких нейронних мереж. Поєднання 16 потужних ядер SHAVE з інтелектуальною пам’яттю робить Myriad X лідером в галузі практичного використання глибоких нейронних мереж та програм комп’ютерного зору. Google Edge Coral прискорювач для машинного навчання – Edge TPU ASIC, розроблена Google. Забезпечує високоєфективне виконання машинного навчання (ML) для моделей TensorFlow Lite.

Третій рівень є розширення описаної дворівневої платформи для задач робототехніки та передбачає як виконання лабораторних робіт у відповідному курсі, так і можливості для дослідницької роботи в галузях робототехніки, штучного інтелекту та Інтернету Речей. Програмною основою цього рівня є відкритий

фреймворк Robot Operating System (ROS/ROS2), який представляє набір бібліотек програмного забезпечення та інструментів для програмування робототехнічних систем різного рівня складності і містить широкий спектр застосунків – від драйверів до найсучасніших алгоритмів та потужних інструментів для розробників. Додатково до програмного забезпечення цей рівень передбачає використання різних актуаторів та давачів, що є типовими в задачах робототехніки. Також однією з переваг ROS є можливість як попередньої симуляції робототехнічних систем у віртуальному середовищі Gazebo так і виконання на реальному апаратному забезпеченні.

2. Від Cloud до Edge/Fog/AIoT

Сучасні технології штучного інтелекту та IoT тісно пов’язані і взаємодоповнюють. Класична топологія IoT-рішень передбачає аналітичну обробку інформації на стороні Cloud і завдяки аналізу даних, отриманих з багатьох джерел інформації, отримується синергетичний ефект Штучний інтелект в Інтернеті Речей (AIoT). Недоліком, як згадувалось вище, сьогоденних IoT-рішень є власне аналіз даних лише на хмарному рівні. Збір величезної кількості даних від тисяч або й мільйонів периферійних пристроїв вимагає не лише постійного росту обчислювальних ресурсів власне Cloud, але й швидкісних та широкопasmових каналів зв’язку. Оскільки ефективність виконання нейромережами задач машинного навчання залежить не в останню чергу від якості отриманих даних, то це теж спричиняє ріст обсягу передачі даних.

Одним зі шляхів вирішення зазначеної проблеми є впровадження штучного інтелекту на якомога більш ранніх етапах передачі інформації, застосовуючи концепцію переходу від Cloud computing в сторону Туманних обчислень (Fog computing) [9]. Тобто периферійні (Edge) пристрої стають “розумними” та здатними самостійно, без пересилання у Cloud, виконувати широкий спектр типових задач штучного інтелекту, зокрема машинного навчання та глибокого навчання (ML/DL) на власному високоєфективному апаратному забезпеченні, орієнтованому на нейронні мережі. Результати надсилаються у Cloud вже не у формі “сирцевих” необроблених даних, а як структуровані мета-дані. Наприклад, замість безперервного відеопотоку у високій роздільній здатності, на Cloud надсилається лише результати детектованих подій чи об’єктів з відповідним описом, причому цей процес носить не безперервний характер, а лише в міру виявлення певних змін в спостережуваному просторі, тобто час від часу. Очевидно, що такий підхід дозволяє суттєво зменшити вимоги до каналів передачі даних, хмарних обчислювальних ресурсів та забезпечує ряд додаткових переваг, як можливості покращення периферійних пристроїв AIoT без внесення змін в

архітектуру системи в цілому. Одночасно, вивільнені хмарні ресурси дозволять більш ефективно опрацювати структуровані мета-дані та приймати рішення.

Оскільки технології AIoT/Edge/Fog знаходяться в активному розвитку та становленні, існують різні підходи як до архітектури в цілому, так і до розділу на окремі логічно-структурні шари. Одним з найбільш відповідних можна прийняти умовний розділ по вертикалі від периферійних пристроїв через “розумний туман” (Fog) до хмарного рівня. Інші автори вважають такий підхід дещо спрощеним і використовують представлення в двох вимірах з деталізацією та складною системою взаємних зв’язків.

За даними Mukherjee та співавторів [10], Азам і Хух [11] та Muntjir та інших [12], архітектура Edge/Fog обчислення складається з шести шарів:

- 1. фізичного та віртуалізації:** віртуальні давачі та їх віртуальна мережа; фізичні давачі, пристрої, їх бездротова мережа;
- 2. моніторинги:** активності, потужності, ресурсів, реакцій (відповідей) та сервісів;
- 3. попередньої обробки даних:** аналіз, фільтрування, реконструкція та відновлення, очистка;
- 4. рівня тимчасового зберігання:** поширення даних, реплікація та де-дуплікація; віртуалізація простору сховищ та пристроїв (NAS, FC, ISCSI тощо);
- 5. безпеки:** кодування/декодування, приватність, цілісність;
- 6. транспорт:** завантаження підготовлених та безпечних даних у хмарні сервіси.

Фізичний і віртуалізаційний шар включає різні типи вузлів, такі як фізичні вузли, віртуальні вузли та віртуальні сенсорні мережі. Ці вузли керуються та підтримуються відповідно до їх видів та вимог щодо обслуговування.

На рівні моніторингу контролюється використання ресурсів, наявність датчиків і моніторинг Edge/Fog вузлів та мережевих елементів. У ньому відстежуються всі завдання, що виконуються вузлами даного шару, моніторинг того, як контролюється споживання енергії та часу обчислень, який вузол виконує яке завдання, в який час і що від нього буде потрібно в наступний момент. Продуктивність та статус усіх застосунків та служб, розгорнутих в інфраструктурі, в значній мірі залежить від ефективності моніторингу.

Шар попередньої обробки виконує завдання управління даними. Зібрані дані аналізуються, фільтруються та очищаються від непотрібної інформації та шумів. Попередньо оброблені дані зберігаються у рівні тимчасового зберігання. Після передачі підготовлених даних у хмару, їх більше не потрібно зберігати локально і можна видалити з тимчасового сховища.

У рівні захисту вступає в дію шифрування / дешифрування даних. Крім того, цілісність даних запобігає

сторонньому втручанню та застосовуються заходи для захисту даних від підробки.

Нарешті, у транспортному шарі, попередньо оброблені дані завантажуються у хмару для подальшого аналізу. Для ефективного використання енергії до хмари завантажуються лише частина зібраних даних. Іншими словами, пристрій шлюзу, що підключає IoT до хмари, обробляє дані перед відправленням їх до хмари. Цей тип шлюзу ще називають розумним шлюзом. Дані, зібрані з давачів у мережі та пристроїв IoT передаються через розумні шлюзи до хмари. Отримані дані хмара зберігає та використовує для створення відповідних служб та сервісів користувачам. Враховуючи обмеженість ресурсів, протоколи зв’язку для Edge/Fog обчислень повинні бути ефективними та легкими в налаштуванні.

Загальну архітектуру сучасних Edge/Fog/Cloud можна представити [13] як на рис. 2

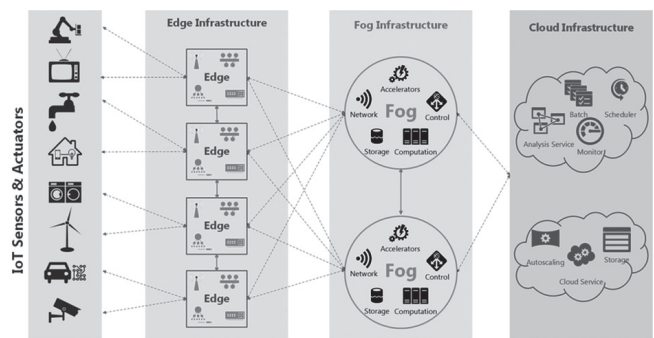


Рис. 2. Горизонтальна edge-fog-cloud архітектура

Інша, вертикально орієнтована, архітектура, що представлена на рис. 3, запропонована в праці [14], яка описує Edge/Fog/Cloud з точки зору розподілених обчислень Інтернету Речей.

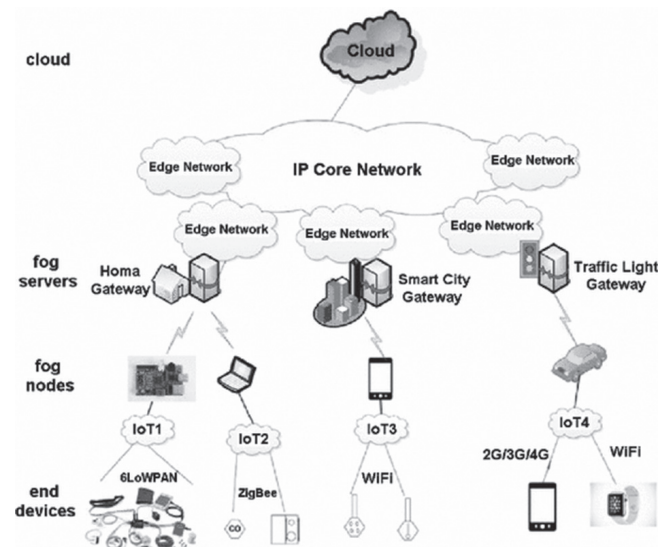


Рис. 3. Вертикальна edge-fog-cloud архітектура

Отже, використання мікросервісного підходу з віддаленим препроцесингом даних (на рівні edge) дає змогу значно знизити використання обчислювальних та часових ресурсів, а також відмовитись від

централізованого зберігання усього масиву даних, що значно наближує опрацювання даних до онлайн режиму.

3. Імплементация концепції tripled learning

Сучасна цивілізація знаходиться в постійному пошуку ефективних способів навчання, зокрема в умовах швидкого прогресу. Поряд з традиційними або «класичними» методами, такими як лекції та практичні роботи, існує також його онлайн-версія як електронне навчання та їх поєднання – змішане навчання (blended learning). В роботі [6] вперше впроваджено термін blended learning або – потрійне навчання як поєднання трьох різних взаємодоповнюючих підходів:

1. Офлайн або «традиційне» навчання, що включає лекції, практичні та/або лабораторні роботи згідно «класичного» підходу.
2. Самостійне навчання в Інтернеті на одному або декількох попередньо обраних вільних онлайн курсах (MOOC), підібраних лектором відповідно до навчальної тематики.
3. Командна робота над власними проектами з можливістю залучення сторонніх менторів та експертів для оцінювання.

Усі три складові є сильно пов'язані і проходять одночасно в рамках конкретного навчального курсу. Відповідно до запропонованої структури потрійного навчання, яке вже практикується в окремих курсах, сторонніми незалежними експертами відмічено зростання ефективності навчання та пробудження творчого потенціалу серед студентів. Протягом трьох років ми успішно впроваджували запропоноване потрійне навчання в рамках курсу «Алгоритмізація та програмування. Частина перша» для студентів першого курсу спеціальності «Системи Штучного Інтелекту». Цей курс створений для вивчення основ програмування мовами C/C++ із прикладами використання різних алгоритмів. Також оглядово даються основи різних базових понять та концепцій ІТ, як операційні системи, мережі, Інтернет, клієнт-серверні технології, елементи сучасної веб-розробки, процес управління проектами, хмарні обчислення, Інтернет речей, штучний інтелект, робототехніка тощо. Головний акцент приділяється вивченню програмування мовами C/C++.

В якості другої компоненти, для самостійного вивчення, було проведено експертний аналіз цілої низки різних онлайн курсів, для вибору найкращого згідно тематики цього курсу. Результатом став міжнародно визнаний онлайн-курс «Вступ до інформатики. CS50» від Гарвардського університету.

Третя частина – командна робота над різними проектами протягом семестру. Як результат, три роки поспіль, з більш ніж 120 студентів-першокурсників стабільно формуються від 30 до 40 команд, що спільно працюють над самостійно обраними темами, під

керівництвом лектора розробляють проекти протягом семестру, представляють їх на проміжному захисті та завершують презентацією перед незалежною групою експертів, запрошених з різних ІТ-компаній.

Ця практика дозволила розширити концепцію tripled learning також на наступні навчальні роки. Теперешні проекти передбачають можливості розвитку на наступних курсах та менторську підтримку з боку старшокурсників.

Наступним є впровадження аналогічного підходу для рівня професійної підготовки в рамках навчальних курсів з хмарних технологій, Інтернету Речей, робототехніки зі спільним об'єднуючим підходом машинного навчання в рамках штучного інтелекту [15].

Пропонована робототехнічна платформа забезпечує технічні можливості для якісного впровадження концепції tripled learning у наскрізному ракурсі, тобто з поступовим перманентним розвитком та розбудовою проектів від простішого до складнішого в міру навчання, а також продовження студентських проектів до рівня науково-дослідних досліджень. Отже, на першому курсі студенти використовують готову робототехнічну систему із вказаними давачами а потужностями для програмування вузького класу задач – руху по траєкторії, виходу з лабіринту тощо. на другому курсі, залежно від типу проекту, студенти додають додаткові давачі, а також формують фронт-енд та бек-енд частину керування робототехнічною частиною. На третьому курсі під час вивчення дисциплін «Основи штучного інтелекту» та «Машинне навчання» студенти мають змогу додати до вказаної платформи Raspberry Pi 3/4 з відеокамерою та OpenCV/OpenVINO, використати Muguad™ X VPU з натренованою нейромережею. Прикладом такого проекту є рухома робототехнічна платформа для розпізнавання переміщень об'єкта в закритому просторі. І на четвертому курсі під час вивчення курсів «Технології хмарних обчислень» та «Вступ до робототехніки», студенти реалізують АІоТ-частину.

Висновки

У статті описано навчальну трьох-рівневу робототехнічну платформу для використання в навчальному процесі при виконанні лабораторних робіт, розробленні курсових проектів та в науково-дослідному процесі. Істотно новим в роботі є створення універсального модульного рішення зі взаємодоповнюючих компонентів, можливість застосування на різних рівнях кваліфікації та модифікувати залежно до актуальних потреб. При цьому пропонована платформа є органічною частиною сучасної спіралі розвитку інформаційних процесів та систем і дозволяє покращити якість навчального процесу, збільшити його орієнтацію на отримання практичного досвіду та віддзеркалює актуальний стан синергії АІ через Edge/Fog/Cloud з ІоТ.

Список літератури:

- [1] *Bonomi, F., Milito, R., Zhu, J., and Addepalli, S.* Fog Computing and its Role in the Internet of Things. // Proc of MCC (2012), pp. 13-16.
- [2] What Comes After the Cloud? How About the Fog? // IEEE Spectrum: Technology, Engineering, and Science News. Retrieved 2017-04-07.
- [3] *Kurzweil, Ray.* Long Live AI // Forbes, 2019. https://www.forbes.com/home/free_forbes/2005/0815/030.html
- [4] *George John.* The Age of Artificial Intelligence // TEDxLondonBusinessSchool – 2013. tedxtalks.ted.com/video/The-Age-of-Artificial-Intelligence
- [5] *Muehlhauser, Luke.* What is AGI? // Machine Intelligence Research Institute. – 2014. <https://intelligence.org/2013/08/11/what-is-agi/>
- [6] *Roman Hasko, Nataliya Shakhovska.* Triple learning: conception and first steps // 14th International Conference on ICT in Education, Research and Industrial Applications, Integration, Harmonization and Knowledge Transfer – 2018 – <http://ceur-ws.org/Vol-2105/10000481.pdf> pp.481-484
- [7] Intel® Movidius™ Myriad™ X VPU // movidius.com/myriadx
- [8] *Liam Tung.* Google's Raspberry Pi-like Coral: AI board with TPU is ready for business // zdnet.com/article/googles-raspberry-pi-like-coral-ai-developer-board-with-tpu-is-ready-for-business/
- [9] *Dov Nimratz.* IoT architecture. // weekly-geekly.github.io/articles/455377/
- [10] *Mukherjee, M.; Shu, L.; Wang, D.* Survey of Fog Computing: Fundamental, Network Applications, and Research Challenges. IEEE Commun. Surv. Tutor. 2018, pp. 1826 – 1857
- [11] *Aazam, M.; Huh, E.N.* Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT. Proc. Int. Conf. Adv. Inf. Netw. Appl. AINA 2015, 2015, pp. 687–694.
- [12] *Muntjir, M.; Rahul, M.; Alhumyani, H.A.* An Analysis of Internet of Things (IoT): Novel Architectures, Modern Applications, Security Aspects and Future Scope with Latest Case Studies. Int. J. Eng. Res. Technol. 2017, 6, pp. 422–447.
- [13] *Cao, Hung & Wachowicz, M. & Renso, Chiara & Carlini, Emanuele.* (2019). Analytics Everywhere: generating insights from the Internet of Things. IEEE Access. 7. 71749-71769. 10.1109/ACCESS.2019.2919514.
- [14] *Escamilla-Ambrosio P.J., Rodríguez-Mota A., Aguirre-Anaya E., Acosta-Bermejo R., Salinas-Rosales M.* (2018) Distributing Computing in the Internet of Things: Cloud, Fog and Edge Computing Overview. // In: Maldonado Y., Trujillo L., Schütze O., Riccardi A., Vasile M. (eds) NEO 2016. Studies in Computational Intelligence, vol 731. Springer, Cham
- [15] *Четвериков Г.Г.* Концепція уніфікації інформаційно-інтелектуальних технологій в систем мовлення [Текст] / Г.Г. Четвериков, М.Ф. Бондаренко, З.Д. Коноплянко // Научно-технический журнал «Бионика интеллекта». – 2011. – № 3 (77). – С. 156–162

Надійшла до редколегії 29.11.2019

УДК 004.75

DOI 10.30837/bi.2019.2(93).09

Л.Е. Чала¹, Д.І. Карпенко², О.О. Берестовий³

¹ Кандидат технічних наук, доцент кафедри штучного інтелекту,
Харківський національний університет радіоелектроніки,
larysa.chala@nure.ua, ORCID iD: 0000-0002-9890-4790

² Студент кафедри штучного інтелекту, Харківський національний університет радіоелектроніки,
daniil.karpenko@nure.ua

³ Студент кафедри штучного інтелекту, Харківський національний університет радіоелектроніки,
oleksandr.berestovyi@nure.ua

РОЗРОБКА ІНТЕРАКТИВНОГО МУЗЕЙНОГО ДОДАТКУ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЇ ДОПОВНЕНОЇ РЕАЛЬНОСТІ

Наведено результати розробки інтерактивного музейного додатку з використанням технології доповненої реальності, що базується на інструментах середовища Vuforia та графічного ядра Unity. Додаток дає можливість здійснювати аналіз та розпізнавання реальних об'єктів (наприклад, музейних експонатів), які раніше були додані в сервіси Vuforia для реєстрації зображень. Додаток протестовано для низки різноманітних мобільних пристроїв на платформах iOS та Android. Визначено перспективи використання запропонованого додатку.

ДОПОВНЕНА РЕАЛЬНІСТЬ, ІНТЕРАКТИВНИЙ ДОДАТОК, VUFORIA, FAST-ДЕТЕКТОР, ЯКІСНЕ НАВЧАННЯ

Приведены результаты разработки интерактивного музейного приложения с использованием технологии дополненной реальности, основанной на инструментах среды Vuforia и графического ядра Unity. Приложение дает возможность осуществлять анализ и распознавание реальных объектов (например, музейных экспонатов), которые ранее были добавлены в сервисы Vuforia для регистрации изображений. Приложение протестировано для ряда различных мобильных устройств на платформах iOS и Android. Определены перспективы использования предложенного приложения.

ДОПОЛНЕННАЯ РЕАЛЬНОСТЬ, ИНТЕРАКТИВНОЕ ПРИЛОЖЕНИЕ, VUFORIA, FAST-ДЕТЕКТОР, КАЧЕСТВЕННОЕ ОБРАЗОВАНИЕ

The results of the development of an interactive museum application using augmented reality technology based on tools from the Vuforia environment and the Unity graphic core are presented. The application makes it possible to analyze and recognize real objects (for example, museum exhibits) that were previously added to Vuforia services for registering images. The application has been tested for different mobile devices on iOS and Android platforms. The prospects for using the proposed application are determined.

AUGMENTED REALITY, INTERACTIVE APPLICATION, VUFORIA, FAST-DETECTOR, QUALITY EDUCATION

Вступ

Технології доповненої реальності (AR – augmented reality) – це технології, що дозволяють доповнювати зображення реальних об'єктів різними об'єктами комп'ютерної графіки, а також поєднувати зображення, отримані від різних джерел [1]. Доповнена реальність, на відміну від віртуальної, лише вносить окремі штучні елементи в сприйняття світу реального. Доповнена реальність додає до відчуттів користувачів з реального світу уявні об'єкти, зазвичай допоміжно-інформативного характеру. AR – технології мають забезпечувати додаткам, що створюються на їх основі такі властивості як: комбінування реального та віртуального світу; інтерактивність; тривимірне представлення об'єктів [2].

До найбільш перспективних галузей використання технологій доповненої реальності, в першу чергу, слід віднести: створення тренажерних систем (в дистанційному освітньому процесі та в промисловості); проектування та дизайн; картографія і туристична галузь; використання QR кодів тощо [3–7].

Наприклад, у туристичній галузі доповнена реальність отримала актуальність в зв'язку з широким

розповсюдженням мобільних пристроїв. Так, подібні системи можуть ідентифікувати навколишні об'єкти (зокрема об'єкти з музейних колекцій), дозволяючи користувачу оперативно орієнтуватися в навколишньому просторі. У зв'язку з цим актуальною є задача створення додатків, які дозволяли б отримати додаткову інформацію про музейний об'єкт після наведення на нього камери телефону або іншого мобільного пристрою.

Такі додатки мають здійснювати функцію автоматичної ідентифікації зображень в відеопотоці з можливістю подальшої прив'язки до них інтерактивних об'єктів.

Перспективним слід вважати використання для реалізації цієї функції платформи Vuforia [8, 9], що є програмним забезпеченням для мобільних пристроїв, яке дозволяє створювати додатки доповненої реальності. Воно використовує технологію комп'ютерного зору для того, щоб розпізнавати і відстежувати плоскі зображення і прості 3D-об'єкти в режимі реального часу. Ця можливість реєстрації зображень дозволяє визначати розташування й орієнтації віртуальних

об'єктів, таких як 3D-моделі, в реальному світі, коли вони розглядаються через камеру мобільного пристрою. Положення і орієнтація віртуального об'єкта відстежується в реальному часі, так що точка зору глядача на об'єкт співвідноситься з їх точкою зору на зображення, що створює ілюзію належності віртуального об'єкту до реальної сцени. Vuforia надає можливість використовувати бібліотеки OpenCV (ComputerVision), які обгорнуті в зручний інтерфейс для розробників на C++, Java, Objective-C і .Net через інтеграцію з міжплатформним середовищем розробки комп'ютерних ігор Unity. Це дає нам можливість використовувати системи когнітивного розвитку, які пов'язані з аналізом реальних образів.

У даній роботі досліджується завдання розробки інтерактивного музейного додатку з використанням технології доповненої реальності, що базується на інструментах середовища Vuforia та дає можливість аналізувати і розпізнавати реальні об'єкти, які раніше були додані в один із сервісів Vuforia для реєстрації зображень (маркерів). В результаті система надає користувачам дані про відповідність реального об'єкта і маркера, внаслідок чого виникає можливість зміни віртуального оточення реальних об'єктів. Наприклад, користувач може завантажити в веб сервіс зображення якої-небудь картини і використовуючи розроблений додаток, визначити збіг картини з частиною реального світу, що потрапила в фокус камери, з маркером, який розташований в сервісі, і в разі необхідності здійснити певні маніпуляції в віртуальному світі, наприклад, вивести опис про цю картину на екран.

1. Принципи побудови доповненої реальності

Можна виділити два головних принципи побудови доповненої реальності: на основі маркера і на основі координат місця розташування користувача.

Безмаркерні технології часто застосовуються в мобільних пристроях і будуються за допомогою спеціальних датчиків: акселерометр, гіроскоп, магнетометр, GPS-приймач. Розглянемо завдання побудови доповненої реальності за допомогою маркерів і алгоритмів комп'ютерного зору.

Під маркером будемо розуміти об'єкт, розташований в навколишньому просторі, який визначається та аналізується спеціальним програмним забезпеченням для подальшого відтворення віртуальних об'єктів. На основі інформації про стан маркера в просторі програма може досить точно спроектувати на нього віртуальний об'єкт, від чого буде досягнуто ефект його фізичної присутності в навколишньому просторі. Використовуючи додаткові графічні фільтри і високоякісні моделі, віртуальний об'єкт може стати важко відмінним від інших елементів інтер'єру або екстер'єру. Як маркери можуть використовуватися аркуш паперу з деяким спеціальним зображенням,

геометричні фігури, тривимірні об'єкти і навіть очі та обличчя людей.

Технології комп'ютерного зору є основоположними для розвитку технологій доповненої реальності і перш за все в галузі використання маркерів. Основний напрямок даної дисципліни – це аналіз і обробка статичних зображень та відеопотоків. Алгоритми комп'ютерного зору дозволяють виділяти ключові особливості на зображенні (кути, межі області), проводити пошук фігур і об'єктів в реальному часі, виконувати 3D реконструкцію за кількома фотографіями тощо. Приклади об'єктів, що можуть використовуватися як маркери, наведено на рис. 1.

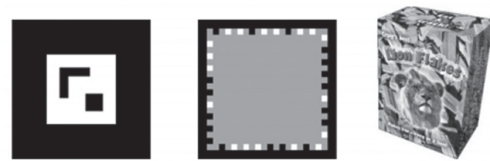


Рис. 1. Приклади маркерів

В області доповненої реальності алгоритми комп'ютерного зору використовуються для пошуку в відеопотоці спеціальних маркерів. Залежно від завдання маркерами можуть бути визначені як спеціально сформовані зображення, так і особи людей. Після знаходження маркера в відеопотоці і обчислення його розташування з'являється можливість побудови матриці проєкції і позиціонування віртуальних моделей. За допомогою цих моделей можна накласти віртуальний об'єкт таким чином, що буде досягнутий ефект присутності. Основна складність полягає в тому, щоб знайти маркер, визначити його місце розташування в кадрі і спроектувати відповідним чином віртуальну модель.

За останнє десятиліття була створена теоретична база в сфері обробки зображень та пошуку в ньому різних об'єктів [10]. Перш за все це стосується методів template matching, генетичних алгоритмів і feature detection. З точки зору побудови доповненої реальності часто використовується feature detection.

Методи feature detection, що найчастіше застосовуються в системах комп'ютерного зору, націлені на обчислення абстракції зображення і виділення на ньому ключових особливостей (наприклад, ізольованих точок, а також кривих або пов'язаних областей). того, що таке ключова особливість зображення. Кожен з конкретних алгоритмів, які реалізують концепцію feature detection, може мати свої специфічні ключові особливості зображення (кути, межі, області тощо).

Згідно з завданням даної роботи будемо для пошуку маркерів виконувати пошук і порівняння зображень по ключових точках, що є певними ділянками фрагменту, характерного для заданого зображення. Приклад такої ключові точки на зображенні наведено на рис. 2.

Для знаходження ключових точок для ідентифікації зображення в відеопотоці використовуються три складові:

- детектор (здійснює пошук ключових точок на зображенні безпосередньо (feature detector));
- дескриптор (здійснює опис знайдених ключових точок, оцінюючи їх позиції через опис навколишніх областей);
- матчер (здійснює побудову відповідностей між двома наборами точок).



Рис. 2. Приклади ключових точок на зображенні

Спочатку за допомогою детектора здійснюється пошук ключових точок шаблонного (шуканого) зображення, які потім описуються за допомогою дескриптора. Дана інформація зберігається в окремий файл (або базу даних), щоб не виконувати цей процес повторно. При обробці відеопотоку з метою пошуку заданого шаблону описаний процес виконується для кожного кадру (за винятком збереження даних). Для встановлення відповідності між ключовими точками і дескрипторами застосовується матчер. Приклад встановлення відповідності точок наведено на рис. 3.



Рис. 3. Встановлення відповідностей між точками шаблону та тестового зображення

Природно припустити, що різні алгоритми працюють з різною швидкістю та ефективністю. В умовах їх застосування для побудови доповненої реальності

необхідно використовувати тільки ті, які показують високу швидкість роботи при досить хорошій якості відстеження позицій ключових точок. В іншому випадку ми можемо отримати помітні відставання у відеоданих, що знімаються. Для підвищення швидкості роботи алгоритмів роботи feature points detection застосовуються різні способи фільтрації точок, щоб мінімізувати їх число і відсіяти невдалі поєднання. Таким чином можна домогтися підвищення як швидкості роботи, так і якості трекінгу маркерів.

2. Алгоритми пошуку особливих точок зображення та створення їх дескрипторів

Як було зазначено, для реалізації технології доповненої реальності використовуються маркери, які допомагають визначити положення камери в просторі. Це обмежує ефективність її використання, так як поперше, маркери повинні бути постійно в кадрі, а подруге, їх необхідно спочатку підготувати.

Працювати відразу з усіма точками зображення для мобільних платформ досить складно, тому будемо виділяти окремі особливі точки на зображенні і стежити за їх переміщеннями. Для цього необхідно попередньо здійснити процес препроцесінгу маркерів.

Для реалізації поставлених в даній роботі завдань доцільним є застосування методу QCAR, що послідовно здійснює пошук особливих точок зображення та створення їх дескрипторів, інваріантних до масштабу і обертання [10]. Така інваріантність означає, що опис ключової точки буде однаковим, навіть якщо зразок змінить розмір і буде повернений. Крім того, сам пошук ключових точок теж має бути інваріантним, щоб повернений об'єкт сцени мав той же набір ключових точок, як і зразок.

В базовому алгоритмі QCAR з використанням FAST-детектора (Features from Accelerated Test) розглядається коло з 16 пікселів навколо точки кандидата. У цьому алгоритмі будується дуга кола для першого квадранта, а координати точок кола для інших квадрантів задаються симетрично. На кожному кроці алгоритму розглядаються три пікселя, з яких вибирається найбільш придатний шляхом порівняння відстаней від центру до обраного пікселя з радіусом кола. Далі вибирається точка зображення p з яскравістю I_p , для якої буде вирішуватися, чи є вона особливою чи ні. Нехай I_p – яскравість точки. Для цього вибирається значення порога t та розглядається коло з 16 пікселів навколо обраної точки. Точка p вважається кутом, якщо серед 16-ти пікселів кола існує n пікселів, кожен з яких яскравіше, ніж $I_p + t$, або n пікселів, кожен з яких темніше, ніж $I_p - t$ (n зазвичай вибирається рівним 12). Відомо, що найменше значення n , при якому точки починають стабільно виявлятися, дорівнює 9.

Висока швидкість роботи даного алгоритму обумовлена тим, що спочатку перевіряється інтенсивність

лише чотирьох точок з кола, під номерами 1,5, 9 і 13. Якщо хоча б для трьох з них виконується зазначена вище умова нерівності, тоді проводиться перевірка решти 12-ти пікселів. В іншому випадку вибирається наступна точка і алгоритм повторюється для неї. Однак ефективність даного алгоритму залежить від порядку обробки зображення і розподілу пікселів (поблизу деякого околу може виявитися кілька особливих точок). В [10] було запропоновано використання машинного навчання для виправлення цього недоліку, згідно з чим було отримано модифікований алгоритм технології QCAR:

- вибирається набір зображень для навчання;
- алгоритм FAST застосовується для пошуку особливих точок на кожному із зображень;
- для кожної особливої точки зберігається 16 пікселів навколо неї у вигляді вектора. Після виконання даного кроку для кожного зображення всі отримані вектори об'єднуються в один вектор P ;
- кожен з 16-ти пікселів S_p відносять до одного з трьох станів згідно з наступними умовами:

$$S_{p \rightarrow x} = \begin{cases} d, I_{p \rightarrow x} \leq I_p - t \text{ (темніше)} \\ s, I_p - t < I_{p \rightarrow x} < I_p + t \text{ (схожий)} \\ b, I_p + t \leq I_{p \rightarrow x} \text{ (світліше)} \end{cases} \quad (1)$$

- в залежності від станів, наведених вище, P розділяється на три підмножини P_d, P_s, P_b ;
- в розгляд вводиться булева змінна K_p , яка дорівнює 1, якщо p – кут та 0 у зворотному випадку;
- використовується алгоритм ID3 (класифікатор дерев рішень) [11,12]. Він вибирає x , який дає більше інформації про те, чи є піксель кутом чи ні, обчисленої за допомогою ентропії K_p ;
- ця процедура повторюється рекурсивно, до тих пір, поки ентропія K_p не дорівнюватиме нулю;
- створене таким чином дерево рішень використовується для більш швидкого детектування в інших зображеннях.

Для того, щоб уникнути знаходження декількох особливих точок поблизу деякого околу, в розроблюваному додатку будемо використовувати процедуру пошуку локальних максимумів, що полягає в реалізації таких операцій:

- обчислюється значення функції V для кожної зі знайдених особливих точок (V – сума різниць інтенсивностей кожного з 16-ти навколишніх пікселів);
- розглядаються дві сусідні особливої точки і порівнюються їх значення V ;
- точка з меншим значенням V виключається з розгляду.

3. Обчислення координат камери і положення особливої точки

Графічна інтерпретація процедури визначення координат наведена на рис. 4. Ідея цієї процедури

полягає в тому, що при зміні кута, з якого направлено камеру користувача, відповідним чином змінюється розмір проекції.

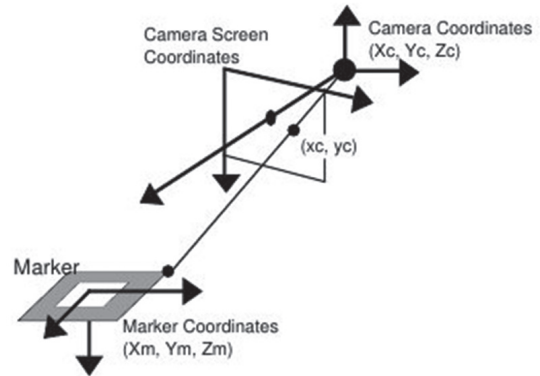


Рис. 4. Принцип визначення координат

Для визначення положення камери в просторі за особливою точкою спочатку здійснюється ініціалізація по двом сусіднім кадрам та обчислюються координати камери в цих кадрах і положення особливої точки. Далі, по відомим тривимірним координатам точок можна обчислювати положення камери в кожному наступному кадрі відеопотоку. Для більш стабільної роботи в процес спостереження додаються нові точки та складається карту простору в зоні спостереження камери. Для початку переведемо тривимірні координати особливої точки в екранні. Для цього використовується наступна формула:

$$\begin{pmatrix} screen[i]_x \\ screen[i]_z \end{pmatrix} = mProj * mWorld * World[i], 1, \quad (2)$$

де $World[i]$ – особливої точки в світових координатах, невідомих до ініціалізації;

$screen[i]$ – компоненти x та z – координати особливої точки на зображенні;

z – глибина зображення відносно камери;

$mProj$ – матриця проекції, що має вигляд:

$$mProj = \begin{bmatrix} pf_x & 0 & -pc_x \\ 0 & pf_y & -pc_y \\ 0 & 0 & 1 \end{bmatrix},$$

pf – фокальна відстань камери в пікселях;

pc – оптичний центр камери в пікселях (центр зображення);

$mWorld$ – матриця, що описує трансформацію точок, розміром 3 на 4 (світова матриця з якої прибрати останній рядок (0 0 0 1)).

У матриці $mWorld$ міститься інформація про переміщення і повороті камери на кожному кадрі.

Екранні координати вже будуть використовуватися на кожному кадрі відео потоку.

Спростимо формулу (2), позбувшись від матриці $mProj$ шляхом введення коефіцієнту $c[i]$ наступного вигляду:

$$c[i] = mProj^{-1} * (screen[i]_x, screen[i]_z, 1). \quad (3)$$

З урахуванням (3) отримуємо:

$$c[i] * (screen[i]_z = mWorld * (world[i], 1)). \quad (4)$$

Ініціалізація відбувається за двома кадрами. Позначимо їх як A і B . Точки $c[i]$ на кадрах A і B позначимо як $cA[i]$ і $cB[i]$

Таким чином з'являються матриці $mWorldA$ (одична матриця) та $mWorldB$, яка обчислюється за допомогою точок, розташованих на одній площині.

Для цих матриць є дійсним таке рівняння:

$$\begin{pmatrix} cB_x * screenB_z \\ cB_y * screenB_z \\ screenB_z \end{pmatrix} = \begin{pmatrix} H_{00} * cA_x + H_{01} * cA_y + H_{02} \\ H_{10} * cA_x + H_{11} * cA_y + H_{12} \\ H_{20} * cA_x + H_{21} * cA_y + H_{22} \end{pmatrix}$$

Це справедливо тільки для точок, розташованих на одній площині. В загальному випадку, у циклі з заздалегідь заданою кількістю ітерацій виконуються наступні дії: випадковим чином вибираються чотири пари точок кадрів A і B ; знаходиться матриця H ; визначається, скільки точок дають помилку менше заданого значення; вибирається H на тій ітерації, для якої було отримано більше точок.

З матриці H необхідно отримати матрицю переходу положення кадру A до кадру B (матрицю $mMotion$). Для цього виконується сингулярне розкладання матриці H згідно з наступними перетвореннями:

$$s = [U] * [V]; \quad (5)$$

$$w1 = |W_{00}|; w2 = |W_{11}|; w3 = |W_{22}|; \quad (6)$$

$$x2 = 0; x3 = \sqrt{(w2^2 - w3^2) / (w1^2 - w3^2)}. \quad (7)$$

Використовуючи перетворення (5) – (7), можна отримати матрицю повороту $R[i]$:

$$R[i] = U * \begin{bmatrix} \cos A & 0 & -\sin A * e3[i] \\ 0 & 1 & 0 \\ \sin A & 0 & \cos A * e3[i] \end{bmatrix} * V^T * e3[i],$$

де

$$\sin A = (w1 - w3 * e3[i]) * x1 * x3 * e1[i] * \frac{e2[i]}{w2};$$

$$\cos A = (w3 * x1^2 + w1 * x3^2 * e3[i]) * \frac{1}{w2};$$

$$e1 = [1, -1, 1, -1, 1, -1, 1, -1];$$

$$e2 = [1, 1, -1, -1, 1, 1, -1, -1];$$

$$e3 = [1, 1, 1, 1, -1, -1, -1, -1];$$

Крім того, необхідно визначити вектор зміщення:

$$t[i] = U * (((w1 - w3 * e3[i]) * x1 * e1[i]), 0, (w3 - w1 * e3[i]) * x3 * e2[i])).$$

Отримані значення $R[i]$ та $t[i]$ дозволяють побудувати 8 можливих варіантів матриці $mMotion$:

$$mMotion[i] = \begin{bmatrix} R[i]_{00} & R[i]_{01} & R[i]_{02} & t[i]_x \\ R[i]_{10} & R[i]_{11} & R[i]_{12} & t[i]_y \\ R[i]_{20} & R[i]_{21} & R[i]_{22} & t[i]_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

Можна показати, що:

$$mWorld_B = mMotion[i] * mWorld_A. \quad (9)$$

Залишається вибрати одну з восьми матриць $mMotion[i]$. Якщо промені виходять з точок першого і другого кадру, то вони повинні перетинатися, причому перед камерою, як в першому, так і в другому випадку. Тому слід підрахувати кількість точок перетину перед камерою в першому і в другому кадрі, використовуючи отримані $mMotion[i]$, та відкинути варіанти, при яких кількість точок буде меншою.

В результаті вибирається та з матриць, яка дає менше помилок. Таким чином, використовуючи отримані матриці $mWorldA$ та $mWorldB$, можна знайти світові координати точок по проєкціях.

Після проведених дій можна обчислити положення камери використовуючи відомі світові координати точок. Для цього використовуємо ітеративний алгоритм, початковим наближенням якого є попередній результат визначення. На кожній ітерації вводимо точки $b[i]$:

$$b[i] = mWorld[i] * (World[i], 1). \quad (10)$$

Ці точки мають послідовно наближатися до визначених вище точок $c[i]$ з відповідною помилкою:

$$e[i] = \left(c[i]_x - \frac{b[i]_x}{b[i]_z}, c[i]_y - \frac{b[i]_y}{b[i]_z} \right). \quad (11)$$

Далі підраховується вектор помилки і вирішується система рівнянь методом найменших квадратів з остаточною отриманням шуканих координат.

4. Опис розробки інтерактивного музейного додатку та його структури

В розробленому інтерактивному музейному додатку для підтримки AR – технологій з забезпеченням функції трекінгу було використано засоби середовища Vuforia. Модуль Vuforia підтримує різні 2D і 3D типи мішеней, включаючи маркерні Image Target, що дозволяє використовувати вкладені функції для реалізації плавного треку маркерів, які можуть бути реалізовані у вигляді деяких реальних графічних зображень музейних колекцій. Наступним кроком став вибір графічного рушія, який дозволив би перенести додаток на мобільну платформу, оскільки відеопотік має транслюватися безпосередньо через відеокамеру мобільного пристрою.

Інструментом для розробки додатків, що працюють під операційною системою Windows, обрано мову C # 7.0, яка легко інтегрується в графічне ядро Unity. З використанням цієї мови було реалізовано модуль візуалізації особливих точок зображення за методом FAST технології QCAR. Приклад такої реалізації наведено на рис. 5.

У даній реалізації алгоритм ID3 використовується для оптимізації порядку, в якому аналізуються пікселі, що дозволило побудувати найбільш ефективний

детектор особливостей. Для отримання проміжних результатів розроблено допоміжний алгоритм, який виводить початкові зображення і зображення, що пройшли препроцесінг (з візуалізованими на них особливими точками).



Рис. 5. Приклад візуалізації особливих точок за методом FAST

Після успішної інтеграції бази даних аналізованих зображень в проект Unity необхідно, використовуючи модуль Vuforia, співвіднести маркери і відповідні їм віртуальні об'єкти. У розробленому додатку як віртуальні об'єкти розглядаються тривимірні тексти з координатами, що відповідають локальним координатам маркера. Можливості інтерактивного характеру для кожного віртуального об'єкта реалізуються за допомогою виведення додаткової інформації при натисканні на ідентифікований в відеопотоці маркер.

Основними перевагами Unity є наявність візуальної середовища розробки, міжплатформенної підтримки і модульної системи компонентів. До недоліків зазвичай відносять появу складнощів при роботі з багатокомпонентними схемами і труднощі при підключенні зовнішніх бібліотек, однак для розроблюваного додатку ця умова не є критичною.

Головний інтерфейс розробки музейного інтерактивного додатку поділено на 5 зон:

- зона 1, де знаходяться всі фізичні об'єкти, з якими в подальшому працює додаток;
- зона 2 (сцена), на якій здійснюється відображення усіх фізичних об'єктів;
- зона 3 (файлова структура);

– зона 4, через яку можна працювати з файлами, які знаходяться в зоні 3;

– зона 4, що містить усі характеристики об'єктів, які обрано в зоні 1.

Коренева структура розробленого додатку має 3 основних директорії, а саме: Modules, StreamingAssets та Mechanisms.

У директорії Modules знаходяться три основних Third-Party модулі: Demigant, TextMesh Pro, Vuforia.

Demigant – модуль для взаємодії функціонального пересування, контролю на налаштування ротації та змінення розмірів об'єктів у реальному часі. Цей модуль необхідний для полегшення роботи при написанні механізмів візуалізації.

TextMesh Pro – допоміжний модуль для роботи зі шрифтами та UI текстовими компонентами, що має можливість роботи в мультимовному режимі. У додатку він використовується для візуалізації описів ідентифікованих картин.

Vuforia – допоміжний модуль для реалізації взаємодії трекінгу та додаткових даними.

У директорії StreamingAssets знаходиться база даних маркерів, що дають можливість ідентифікувати зображення.

Директорія Mechanisms містить наступні механізми: Bootstrap; DescriptionManager; MuseumManager; TrackingSystem; UIController.

Для розробки бази даних зображень та описів до них був реалізований модуль ScriptableObject DescriptionDataStorage, що серіалізує лист класу ArtDescription, який містить назву зображення, спрайт зображення, рік створення та код трекінгу. Разом з тим реалізовано механізм GetArtByCode, що здійснює пошук необхідних даних по трекінговому коду для серіалізованого об'єкту.

Для доступу до бази даних використовується менеджер DescriptionManager, що представлений статичним класом та має в диспозиції набір методів доступу до бази даних.

Для видачі інформації про ідентифіковане зображення та меню основної та додаткової інформації у додатку використано універсальний контролер UI.

Важливим кроком реалізації додатку є обробка подій детекції зображень та їх виходу з зони пошуку. Для реалізації цих функцій використано клас компонентів CustomTrackableEventHandler, що є спадкоємцем класів MonoBehaviour та інтерфейсу ITrackableEventHandler. При ініціалізації цього компоненту проходить реєстрація підписок TrackableEventHandler компоненту та ініціалізація TextMesh, тобто генерація об'єкту з компонентом тексту за заданим набором параметрів відображення до кожного із об'єктів трекінгу. Даний алгоритм необхідний для того, щоб при ідентифікації об'єкту у відеопотоці на ідентифікований об'єкт можна було зпроєціювати його назву.

При реєстрації події трекінгу маркеру викликається метод OnTrackingFound, що в свою чергу викликає метод UpdateTextMesh, який виводить на ідентифіковане зображення необхідний текст та викликає метод OnItemTracked класу MuseumManager. При загубленні зображення у зоні пошуку викликається метод OnTrackingLost, що відключає текст на зображенні та викликає метод OnItemLost, який відключає процес отримання додаткової інформації на екрані.

Для реалізації основних модулів додатку необхідно заповнити базу даних зображеннями та дескрипторами для них з застосуванням механізмів Vuforia.

Додаток передбачає можливість роботи з хмарною базою даних. Для реалізації в додатку цієї можливості використано клас CloudRecoBehaviour.

5. Апробація інтерактивного музейного додатку

Додаток був протестований для низки різноманітних мобільних пристроїв на платформах iOS та Android. Результати тестування показали, що додаток працює стабільно і не викликає системних помилок, що призводять до закриття чи збоїв.

Реалізований продукт відповідає усім вимогам, що були поставлені у розрізі аналізу області та маркетингового огляду: стабільна робота на різноманітних пристроях; кросплатформеність; можливість роботи без підключення до інтернету; безперервна робота у режимі відеопотоку; можливість отримувати назву зображення при фокусі на ньому; можливість отримати базову додаткову інформацію про зображення, не виходячи з додатку; можливість отримати розширену інформацію про додаток з виходом у браузер.

Базовий інтерфейс додатку є відеопотоком камери на пристрої та інтерактивною кнопкою у активному стані та кнопкою-підказкою у пасивному. У пасивному стані (стані пошуку) текст на кнопці підказує користувачу, що треба знайти картину. Приклад візуалізації інтерфейсу у цьому стані представлено на рис. 6.



Рис. 6. Інтерфейс додатку у стані пошуку

Коли у рамках відеопотоку з'явиться зображення, яке можливо ідентифікувати, то кнопка змінить свій стан та стане інтерактивною, внаслідок чого текст на ній зміниться на Get Info, а під ідентифікованим зображенням з'явиться його назва. Приклад візуалізації цього стану представлено на рис. 7.

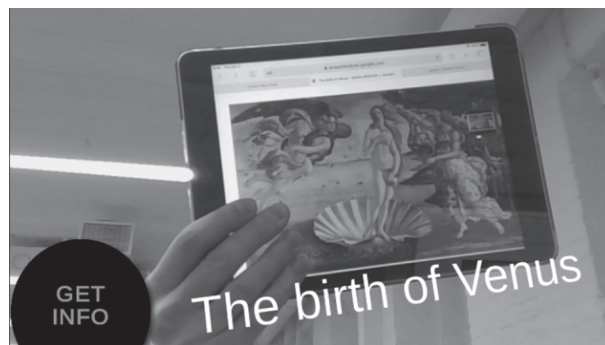


Рис. 7. Інтерфейс додатку в активному стані

Якщо користувач натисне на кнопку Get Info, то у додатку запуститься вікно додаткової інформації, що містить у собі зображення у кращій якості, назву картини, прізвище автора та роки її створення. Візуалізація меню додаткової інформації представлена на рис. 8.

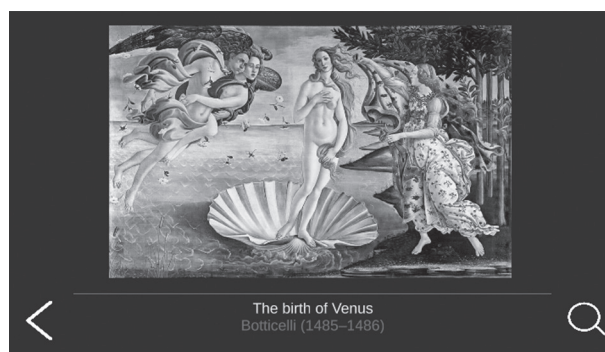


Рис. 8. Меню додаткової інформації про картину

На цьому меню також присутні кнопка виходу та кнопка пошуку інформації в інтернеті. При натисканні на кнопку виходу додаток повернеться до стану обробки відеопотоку. При натисканні на кнопку пошуку інформації в інтернеті відкриється вікно браузеру із запитом про дану картину.

Оскільки розробка додатку зорієнтована на мобільні платформи, що передбачає необхідність взаємодії додатку з відеопотоком та рендерінгом, важливо дотримуватись високої загальної швидкодії продукту та високої швидкодії його роботи з відеопам'ятю.

Тестування швидкодії додатку по CPU та аналіз навантаження на відеопроцесор підтвердили відсутність проблем зі швидкістю при використанні додатку в музейних приміщеннях в реальному режимі часу.

Крім того, тестування дозволило отримати оцінки мінімальних затрат фізичної пам'яті, необхідних для стабільної роботи додатку, що реалізується на різних на мобільних платформах.

Висновки

У статті досліджено завдання розробки інтерактивного музейного додатку з використанням технології доповненої реальності, що базується на інструментах середовища Vuforia та графічного ядра Unity. Додаток дає можливість здійснювати аналізу та розпізнавання

реальних об'єктів (наприклад, музейних експонатів), які раніше були додані в один із сервісів Vuforia для реєстрації зображень (маркерів). В результаті система надає користувачам дані про відповідність реального об'єкта і маркера, внаслідок виникає можливість зміни віртуального оточення реальних об'єктів. Положення і орієнтація віртуального об'єкта відстежується в реальному часі, так що точка зору користувача на об'єкт співвідноситься з його точкою зору на зображення, що створює ілюзію належності віртуального об'єкту до реальної сцени. Наприклад, користувач може завантажити в веб сервіс зображення картини і використовуючи розроблений додаток, визначити збіг картини з частиною реального світу, що потрапила в фокус камери, з маркером, який розташований в сервісі, і в разі необхідності здійснити певні маніпуляції в віртуальному світі, наприклад, вивести опис про цю картину на екран. Результати експериментального тестування підтверджують відсутність проблем зі швидкодією при використанні додатку в музейних приміщеннях в реальному режимі часу.

Перспективним розвитком додатку є подальше дослідження напряму розвитку його функціональних можливостей з урахуванням додаткових метаданих щодо зображень об'єктів та створення відповідних програмних модулів. Додаток можна використовувати під час навчання людей с обмеженими можливостями, розробивши модуль, який відтворюватиме звук.

Список літератури:

- [1] Website Architecture Planning - Augmented Reality (AR) [Електронний ресурс]. – URL: <https://dynamapper.com/blog/15-augmentedreality-ar>. – Назва з екрана.
- [2] Доповнена реальність у мобільному додатку Smartum! [Електронний ресурс]. – Режим доступу: <https://smartum.com.ua/news/dopolnennaya-real-nost-v-mobilnomprilozhenii/>. – Назва з екрана.
- [3] Tim Pedure. (2017). Applications of Augmented Reality [Електронний ресурс]. URL: <https://www.lifewire.com/applications-of-augmented-reality-2495561>. – Назва з екрана.
- [4] V. Interrante, T. Höllerer and A. Lécuyer, «Virtual and Augmented Reality,» in IEEE Computer Graphics and Applications, vol. 38, no. 2, pp. 28-30, Mar./Apr. 2018. doi: 10.1109/MCG.2018.021951630 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8336832&isnumber=8336820>. – Назва з екрана.
- [5] Qingfeng Zhang, Weilong Chu, Changhong Ji, Chengyuan Ke, Yamei Li. (2015). An implementation of generic augmented reality in mobile devices [Електронний ресурс]. URL: <http://ieeexplore.ieee.org/libaccess.sjlibrary.org/document/7065112/>. – Назва з екрана.
- [6] Модло Є. О. Використання технології доповненої реальності у мобільно орієнтованому середовищі навчання ВНЗ / Є. О. Модло, Ю. В. Єчкало, С. О. Семеріков, В. В. Ткачук. [Електронний ресурс]. – URL: <https://phm.cuspu.edu.ua/ojs/index.php/NZPMMFTO/article/viewFile/1115/1094>. – Назва з екрана.
- [7] Слюсар В.І. Концепція стандартизації тактичних засобів доповненої реальності / В.І. Слюсар //Тези доповідей Міжнародної науково-технічної конференції «Перспективи розвитку озброєння та військової техніки Сухопутних військ». –Львів: Національна академія сухопутних військ імені гетьмана Петра Сагайдачного.– 2018 – С. 63 – 64.
- [8] Чубукова О.Ю. Інноваційні технології доповненої реальності для викладання дисциплін у вищих навчальних закладах України / О.Ю. Чубукова, І.В. Пономаренко // Проблеми інноваційно- інвестиційного розвитку. – 2018. – № 16 –С. 20 – 27.
- [9] Берестовий О.О. Системи когнітивного розвитку з використанням платформи Vuforia / О.О. Берестовий, Д.І. Карпенко, С.Г. Удовенко // Матеріали 5 Міжнародної науково-практичної конференції «Обчислювальний інтелект» – 15-21.04.19 р. Україна . – Ужгород. – 2019. – С. 165-166.
- [10] Mikolajczyk K. An Affine Invariant Interest Point Detector / K. Mikolajczyk, C. Schmid // European Conference on Computer Vision, – 2002. – P. 128 –142.
- [11] Ronald T. Azuma. A Survey of Augmented Reality / T. Azuma Ronald // [Електронний ресурс]. – URL: <https://www.cs.unc.edu/~azuma/ARpresence>
- [12] Четвериков Г.Г. Многозначные структуры (анализ, сравнение, синтез, обобщение). Ч.1: Учебное пособие.– К.: ИСМО, 1997.– 192 с.

Надійшла до редколегії 3.12.2019

**И.В. Левыкин**

Кандидат технических наук, профессор кафедры «Медиасистем и технологий»,
Харьковский национальный университет радиоэлектроники,
ihor.levykin@nure.ua, ORCID – <https://orcid.org/0000-0001-8086-237X>

ОБОБЩЁННЫЙ АЛГОРИТМ И ПРОГРАММНАЯ ПЛАТФОРМА ПОЛУЧЕНИЯ РЕШЕНИЯ ПО ПРИОРИТЕТАМ ЗАПУСКА БИЗНЕС-ПРОЦЕССОВ

Особенностью нерегулярных сквозных бизнес-процессов является то, что в некоторых конкурентных точках их количество не только может изменяться, но и в их число может быть включён новый бизнес-процесс. С учётом этих особенностей возникают проблемы планирования, учёта, контроля, анализа и регулирования. Для их решения применяются методы и средства оперативно-календарного планирования. Однако возникают проблемы при управлении такими бизнес-процессами, конкурирующими за общие ресурсы. Для их решения в статье предлагается обобщённый алгоритм нахождения решения по установлению приоритетов запуска сквозных бизнес-процессов с участием ЛПП в каждой конкурентной точке. Описана разработанная программная платформа <СОВЕТЧИК ЛПП> обеспечивающая в режиме just in time, автоматическую корректировку полученных вариантов решений.

БИЗНЕС-ПРОЦЕСС, ПРИОРИТЕТ, КОНКУРЕНТНАЯ ТОЧКА, МОДЕЛЬ ПОТОКОВ РАБОТ

Особливістю нерегулярних наскрізних бізнес-процесів є те, що в деяких конкурентних точках їх кількість не тільки може змінюватися, але і в їх число може бути включений новий бізнес-процес. З урахуванням цих особливостей виникають проблеми планування, обліку, контролю, аналізу і регулювання. Для їх вирішення застосовують методи і засоби оперативно-календарного планування. Однак виникають проблеми при управлінні такими бізнес-процесами конкуруючими за загальні ресурси. Для їх вирішення в статті пропонується узагальнений алгоритм знаходження рішення щодо встановлення пріоритетів запуску наскрізних бізнес-процесів за участю ЛПП в кожній конкурентній точці. Описано розроблену програмна платформа <ПОРАДНИК ЛПП> забезпечує в режимі just in time, автоматичне коректування отриманих варіантів рішень.

БИЗНЕС-ПРОЦЕСС, ПРИОРИТЕТ, КОНКУРЕНТНА ТОЧКА, МОДЕЛЬ ПОТОКІВ РОБІТ

A feature of irregular end-to-end business processes is that at some competitive points their number can not only change, but a new business process can be included in their number. Given these features, problems arise in planning, accounting, control, analysis and regulation. To solve them, methods and tools of operational calendar planning are used. However, problems arise when managing such business processes competing for shared resources. To solve them, the article proposes a generalized algorithm for finding a solution to establish priorities for launching end-to-end business processes with the participation of decision makers at each competitive point. The developed software platform <LPR ADVISOR> is described which provides in the just in time mode the automatic correction of the obtained solutions.

BUSINESS PROCESS, PRIORITY, COMPETITIVE POINT, WORKFLOW MODEL

Введение

Сложность описания последовательности выполнения нерегулярных, взаимосвязанных, конкурирующих за общие ресурсы сквозных бизнес-процессов (БП) связана с факторами, определяющими как номенклатуру таких процессов, так и накладываемые на них условия выполнения, которые в ходе реализации жизненного цикла производственного процесса могут быть изменены. Сквозные БП представляются в виде последовательностей действий их выполнения с учетом доступа к соответствующим ресурсам (конкуренции за ресурсы между процессами, которые выполняются в одно и то же время), что необходимо учитывать при достижении целей каждого из этих процессов. Такие задачи относятся к классу задач оперативно-календарного планирования, базирующиеся на основе математической теории расписаний и характеризуются сложностью разработки моделей их решения. Однако, использование методов линейного программирования не позволяет учитывать динамику дискретных производственных процессов, изменение состояния производства, а также требует

детального его описания, что приводит к увеличению сложности разрабатываемых моделей. При этом, возникает проблема выбора такой последовательности действий выполнения сквозных бизнес-процессов, которая может быть реализована с использованием конкретных ресурсов на всём производственном цикле. Причём для всех заказов, выполняемых одновременно, устанавливаются конкретные последовательности, реализуемые с использованием общих ресурсов. В рамках математического моделирования процесса решения этой задачи, в отличие от теории расписаний, такую последовательность можно представить моделью потоков работ, описывающей реализацию жизненного цикла сквозного бизнес-процесса [1, 2, 3, 4].

1. Модель нахождения решения определения приоритетов запуска сквозных бизнес-процессов.

Проблема управления сквозными бизнес-процессами связана с многими факторами, влияющими на реализацию всех этапов их жизненного цикла [5, 6]. При этом необходимо учитывать ряд коммерционных

составляющих таких процессов [7, 8]. Одной из них является минимизация времени доступа к ресурсам каждого из этих процессов с условием обязательного их выполнения, в соответствии с установленными темпаральными ограничениями, регламентируемые соответствующими договорами с заказчиками [9]. Так как конкуренция среди группы бизнес-процессов возникает в конкурентных точках, то необходимо найти требуемое решение не только в каждой из них, но и с учётом выполнения процессов во всех последующих точках.

Получение такого решения представим следующей моделью:

$$M = (T_i^{set}, t_i^{left}, t_{li}^i, t_{li}^h, B_{ls}^i, t_j^{wt}), \quad (1)$$

где T_i^{set} — время выполнения i -го заказа по договору; t_i^{left} — оставшееся время выполнения i -го заказа; t_{lj}^h — нормативное время выполнения j -ого действия l -го бизнес-процесса; t_{lj}^i — время выполнения j -ого действия l -го бизнес-процесса i -го заказа; B_{ls}^i — порядок/приоритет прохождения l -го бизнес-процесса i -го заказа в s -й конкурентной точке; t_j^{wt} — время ожидания j -ого действия.

Особенностью нерегулярных сквозных бизнес-процессов является то, что в некоторых конкурентных точках их количество не только может изменяться, но и в их число может быть включён новый бизнес-процесс. В связи с этим необходимо для каждого бизнес-процесса найти такую последовательность порядка их прохождения по конкурентным точкам, которая удовлетворяла бы заданным условиям. Получение требуемых решений осуществляется несколькими прогонами с использованием программного средства <СОВЕТЧИК ЛПР> с включением в этот процесс лица принимающего решение (ЛПР).

2. Описание обобщённого алгоритма нахождения требуемого решения

Для реализации приведенной модели (1), предлагается обобщённый алгоритм нахождения вариантов <Решений> (управляющих воздействий) в виде установления приоритетов доступа к общим ресурсам не регулярных бизнес-процессов в соответствующих конкурентных точках. Нахождения требуемого <Решения> осуществляется по критериям: оставшееся время выполнения бизнес-процессов/заказов и времени запаздывания их выполнения. Полученное <Решение> означает выполнение заказов в установленные по договорам сроки.

Входной информацией такого алгоритма являются данные о событиях бизнес-процессов отображаемых в соответствующем логге событий, фиксируемом информационной системой.

Выходной информацией алгоритма являются данные о приоритетах доступа к общим ресурсам бизнес-процессов в соответствующих конкурентных точках.

Описание обобщённого алгоритма определения приоритетов доступа к общим ресурсам конкурирующих бизнес-процессов, представим следующими этапами:

Этап 1. Ввод параметров по каждому сквозному бизнес-процессу из документов о заказах (время начала и окончания выполнения заказа, нормативное время выполнения действия бизнес-процесса t_{ij}^h и время ожидания t_j^{wt} его доступа к ресурсам в конкурентных точках).

Этап 2. Вычислить длительность выполнения действий каждого заказа t_{ij}^i в конкурентных точках в виде суммы: времени ожидания t_j^{wt} и нормативного времени выполнения действия t_{ij}^h .

Этап 3. Определить оставшееся время выполнения каждого бизнес-процесса, вычитанием из времени выполнения заказа по договору текущего времени выполнения заказа до соответствующей конкурентной точки $t_{ij}^{left} = T_{ij}^{set} - t_{ij}^i$.

Этап 4. Рассчитать суммарное оставшееся время выполнения заказов по соответствующим порядком в каждой конкурентной точке $T_j = \sum_{i=1}^3 t_{ij}^{left}$.

Этап 5. Установить по наибольшему суммарному оставшемуся времени выполнения, порядок запуска заказов B_{ls}^i в l -ой конкурентной точке.

Этап 6. Вычислить отклонения наибольшего значений суммарного оставшегося времени от текущего по всем порядкам di , $di = T_j^{Smax} - T_j^S$, определяющие насколько изменятся эти значения при изменении приоритетов запуска B_{ls}^i на последующих конкурентных точках.

Этап 7. Проверить наличие отрицательных значений суммарного оставшегося времени выполнения бизнес-процессов в последней конкурентной точке.

Этап 8. Провести корректировку последовательности порядков запуска заказа ЛПР для ликвидации этих отрицательных значений, повышением его приоритета в соответствующих конкурентных точках на 1-цу (поднять заказ на 1-ин уровень вверх).

Этап 9. Проверить наличие отрицательных значений суммарного оставшегося времени выполнения бизнес-процессов в последней конкурентной точке на 2-м прогоне.

Этап 10. Рассчитать значения времени максимального запаздывания выполнения БП, которые определяются суммой значений оставшегося времени соответствующего заказа в соответствии со всеми вариантами установления приоритетов в последней конкурентной точке, исходя из результатов полученных после первого прогона.

Этап 11. Проверить наличие отрицательных значений суммарного оставшегося времени выполнения бизнес-процессов в последней конкурентной точке на 2-м прогоне.

Этап 12. Определить порядки запуска заказов с фиксированием 2-х конкурентных точек на 3-ем прогоне. Для этого, исходя из данных о запаздывании, определяется какой из заказов по приоритету необходимо передвигать вверх.

Этап 13. Пересчитать последовательности порядков в оставшихся конкурентных точках, для определения порядка имеющего наибольшее оставшееся время выполнения.

Этап 14. Провести корректировку последовательностей порядков ЛПР после третьего прогона с фиксацией одной точки с последующими изменениями в другой и в остальных точках, по данным об оставшемся общем времени выполнения заказов.

Этап 15. Провести запуск четвёртого прогона, на котором определяется возможность нахождения лучшего результата, по критерию значения большего общего оставшееся время, чем найденное после прогона на третьем этапе с фиксацией трёх конкурентных точек. Каждый порядок, который был выбран при выполнении 3-го прогона, уже имел определённые значения d_i . Последовательно установленные порядки со своими значениями d_i определили общее суммарное значение ds , $ds = d_I + d_{II}$ по результатам 3-го прогона. Исходя из полученных сопоставлений между оставшемся времени выполнения t_{ij}^{left} и общим суммарным значением ds , осуществляется поиск таких вариантов порядков на каждом этапе прогонов, при которых суммарное значение ds будет меньше лучшего результата после прогона на третьем этапе. Затем ЛПР перебираются все варианты запуска, удовлетворяющие заданным условиям с поиском таких порядков, у которых суммарное значение $d1$, $d2$ и $d3$ будет меньше ds , полученного после 3-го прогона. Дальнейшее нахождение лучшего варианта с фиксацией следующей четвёртой точки невозможно в связи с тем, что не выполняется условие меньше или равное ds , полученное после фиксации трёх точек. При фиксации четвёртой точки это значения будет только увеличиваться, что свидетельствует о получении худшего решения.

3. Имитационный эксперимент

Апробация предлагаемого алгоритма получения требуемого Решения по определению приоритетов запуска заказов/бизнес-процессов проведена на примере одной группы, состоящей из 3-х заказов с семью конкурентными точками, с использованием разработанного программного средства <СОВЕТЧИК ЛПР>. При этом рассматриваются все возможные комбинации запуска бизнес-процессов в каждой из 7-и конкурентных точек с 6-ю возможными вариантами.

Для реализации задачи нахождения требуемого Решения осуществляется ввод исходных данных (время начала и окончания выполнения заказа,

нормативное время выполнения действия бизнес-процесса t_{ij}^h и время ожидания t_j^{wt} его доступа к ресурсам) из документов о-заказах (договоров, технологических карт), ресурсах за какие они будут конкурировать (оборудование, материалы и т.д.) и в каких конкурентных точках n по каждому БП. Значения нормативного времени выполнения бизнес – процессов по всем порядкам их запуска в 1-ой конкурентной точке представлены в таблице 1.

Таблица 1

Значения нормативного времени выполнения бизнес – процессов по всем порядкам их запуска в 1-ой конкурентной точке

Порядок	123	132	231	213	312	321
Первый	4	4	8	8	10	10
Второй	8	10	10	4	4	8
Третий	10	8	4	10	8	4

Расчёты времени выполнения бизнес-процессов по всем порядкам и их общее время выполнения в 1-ой конкурентной точке представлены в таблице 2.

Таблица 2

Значения времени выполнения бизнес-процессов в 1-ой конкурентной точке

Порядок	123	132	231	213	312	321
Первый	4	4	8	8	10	10
Второй	12	14	18	12	14	18
Третий	22	22	22	22	22	22
Итого	38	40	48	42	46	50

Оставшееся время выполнения каждого БП по всем порядкам, реализуется вычитанием из времени выполнения каждого заказа по договору T_i^{set} , нормативного времени выполнения t_{ij}^h из таблицы 2. Например, для первого порядка при значениях времени выполнения заказов 120,110,100, получено оставшееся время выполнения каждого БП равное $120-4=116, 110-12=98, 100-22=78$. Аналогично такие расчёты, проведенные по остальным порядкам, представлены в таблице 3.

Таблица 3

Оставшееся время выполнения БП по всем порядкам в 1-ой конкурентной точке

Порядок	123	132	231	213	312	321
Первый	116	116	102	102	90	90
Второй	98	86	82	108	106	92
Третий	78	88	98	78	88	98
Итого	292	290	282	288	284	280
d_i	0	2	10	4	8	12

Также рассчитываются значения суммарного оставшегося времени выполнения 3-х заказов в соответствии с порядком их прохождения, а также параметр d_i , определяющий отклонения от наибольшего суммарного оставшегося времени выполнения заказов T_j^{Smx} его последующего значения по всем порядкам (таблица 3). По параметру d фиксируется

изменение оставшегося времени при корректировке приоритетов запуска на последующих конкурентных точках. Кроме того, проводится расчёт суммарного оставшегося времени выполнения заказов $T_j = \sum_{i=1}^3 t_{ij}^{left}$

по i -у заказу и j -у порядку. В таблице 4 представлены итоговые данные по оставшемуся времени выполнения 3-х БП после прохождения последней 7-ой конкурентной точки.

Таблица 4

Оставшееся время выполнения БП по всем порядкам на 7-ой конкурентной точке

Порядок	123	132	231	213	312	321	1	-99
Первый	-10	-10	14	14	42	42	2	21
Второй	2	30	38	-14	-19	5	3	204
Третий	26	-7	-23	26	-7	-23		
Всего	18	13	29	26	16	24		

Так как в таблице 4 имеются отрицательные значения оставшегося времени выполнения заказов, то для их устранения устанавливаются приоритеты запуска заказа с максимальным значением оставшегося времени T_j^{Smx} с повышением приоритета его выполнения в соответствующих конкурентных точках на 1-цу (поднять заказ на 1-н уровень вверх по всем конкурентным точкам);

Если требуемое <Решение> по всем конкурентным точкам не получено, то на 2-ом прогоне ЛПР необходимо, с использованием программного средства <СОВЕТЧИК ЛПР>, перейти к корректировке приоритетов запуска бизнес-процессов по критерию максимального времени невыполнения/запаздывания бизнес-процессов. Для этого ЛПР определяет тот бизнес-процесс, у которого по всем вариантам значения времени невыполнения по заказу максимально. Как видно из таблицы 4, для 1-го бизнес-процесса по всем комбинациям значения оставшегося времени выполнения по этому критерию соответствуют первому приоритету (-10,-10,-23,-14,-1,-23=-99). Соответственно для 2-го и 3-го бизнес-процессов эти величины будут равны 21 и 204. Для устранения этого отклонения, для бизнес-процесса с самым большим невыполнением этого ограничения, ЛПР увеличивает приоритет его запуска на один уровень вверх. В тех конкурентных точках, в которых 1-ый бизнес-процесс имеет 1-ый приоритет запуска, существующий порядок не изменяется. Соответственно, как и при первом прогоне определяются новые порядки для каждой последующей конкурентной точки. При этом, чем меньше оставшееся время или оно имеет отрицательное значение (запаздывание), тем приоритет у данного заказа должен быть выше. Возможные варианты определения приоритетов представлены на рисунке 1. На рисунке 1 в правой части представлена таблица, а на левой находятся кнопки, являющиеся элементами интерфейса программной платформы.

Все таб.	Сброс	Очистка	№	Точка	d	порядок
Таб 1_1	Кон т.1.1		1	точка 1	0	0
Таб 1_2		Кон т.1.2	2	точка 1	0	0
Таб 2_1	Кон т.2.1		3	точка 2	8	213
Таб 2_2		Кон т.2.2	4	точка 2	2	312
Таб 3_1	Кон т.3.1		5	точка 3	6	213
Таб 3_2		Кон т.3.2	6	точка 3	2	312
Таб 4_1	Кон т.4.1		7	точка 4	10	213
Таб 4_2		Кон т.4.2	8	точка 4	4	312
Таб 5_1	Кон т.5.1		9	точка 5	10	213
Таб 5_2		Кон т.5.2	10	точка 5	6	312
Таб 6_1	Кон т.6.1		11	точка 6	11	123
Таб 6_2		Кон т.6.2	12	точка 6	4	132

Рис. 1. Возможные варианты определения приоритетов

В столбце «время», показаны порядки, удовлетворяющие условию выполнения всех заказов. Наличие значение «-» означает, что данный порядок не удовлетворяет сформулированному условию.

Для получения всех результатов и последующего его анализа последовательного изменения порядка в одной конкурентной точке по всем конкурентным точкам, на втором прогоне ЛПР в платформе использует кнопку «Все таб.» (рис. 2). Данная кнопка позволяет запустить проведение необходимых расчётов и заполнить все результирующие таблицы с соответствующими данными о результатах выполнения 3-х заказов по всем конкурентным точкам. Кнопка «Очистка» предназначена для очистки введенных порядков во всех контрольных точках и возвращение установленных при первом проходе начальных порядков. Кнопка «Сброс» позволяет запустить очистку данных о результатах выполнения заказов по всем конкурентным точкам в таблицах.

На третьем прогоне фиксируется один из порядков и последовательно, после каждого фиксирования порядка в остальных конкурентных точках, программой осуществляется пересчёт во всех остальных точках. ЛПР находит тот порядок, у которого значение d минимально. Аналогично для исключения отрицательных значений оставшегося времени выполнения по всем порядкам осуществляется поднятия 1-го заказа на следующий уровень. При этом ЛПР имеет возможность получать набор вариантов порядков по каждой фиксированной точки, представленный на рис. 2.

Из рис. 2 видно, что ЛПР имеет возможность выбрать тот вариант, который имеет наибольшее оставшееся время и удовлетворял бы или больше соответствовал порядку в сложившихся условиях выполнения заказов.

Возможность получения лучшего результата, может быть проверена с фиксацией трёх конкурентных точек при запуске 4-о прогона по общему суммарному значению ds , полученному по результатам 3-го

прогона. Сопоставляя t_i^{left} и общее суммарное значение ds , проводится поиск порядков, при которых суммарное значение ds должно быть меньше лучшего результата после прогона на третьем этапе. ЛПР перебираются все варианты запуска, удовлетворяющие заданным условиям. Для этого последовательно в начале фиксируется первая конкурентная точка с минимальным значения di , а затем вторая и третья конкурентные точки с этим же значением. При этом после 4-х прогонов с фиксацией 1-й, 2-х, 3-х и 4-х конкурентных точек выбирается та последовательность, у которой оставшееся время большее.

Все таб.	Сброс	Очистка	№	Точка	d	Порядок	Время
Таб 1_1	Кон т.1.1		1	точка 1	0	0	-
Таб 1_2	Кон т.1.2		2	точка 1	0	0	-
Таб 2_1	Кон т.2.1		3	точка 2	8	213	-
Таб 2_2	Кон т.2.2		4	точка 2	2	312	-
Таб 3_1	Кон т.3.1		5	точка 3	6	213	-
Таб 3_2	Кон т.3.2		6	точка 3	2	312	-
Таб 4_1	Кон т.4.1		7	точка 4	10	213	-
Таб 4_2	Кон т.4.2		8	точка 4	4	312	18
Таб 5_1	Кон т.5.1		9	точка 5	10	213	-
Таб 5_2	Кон т.5.2		10	точка 5	6	312	-
Таб 6_1	Кон т.6.1		11	точка 6	11	123	-
Таб 6_2	Кон т.6.2		12	точка 6	4	132	-
							18

Рис. 2. Данные после фиксации всех конкурентных точек

Практическая реализация прогонов с использованием разработанной платформы на каждом этапе осуществляется: с помощью поиска необходимых значений параметров, соответствующих вычислений, проверок выполнения условий накладываемых ЛПР и реализация соответствующих макросов.

Примером поиска нужного значения является нахождения значений максимального оставшегося времени в каждой конкурентной точки. Например, результат поиска порядка с максимальным значением оставшегося времени в диапазоне T9:Y9, диапазон «всего» с занесения порядка 123 в ячейку R11, которому соответствует это максимальное время, представлен на рис. 3.

=ВПР(Q11:Q14;S5:Y9;ПОИСКПОЗ(МАКС(S9:Y9);S9:Y9,0);ЛОЖЬ)																
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y			
1 конкур. точка							3	100								
Послед	123	132	231	213	312	321	Послед	123	132	231	213	312	321			
первый	4	4	8	8	10	10	первый	116	116	102	102	90	90			
второй	12	14	18	12	14	18	второй	98	86	82	108	106	92			
третий	22	22	22	22	22	22	третий	78	88	98	78	88	98			
всего	38	40	46	42	46	50	всего	292	290	282	288	284	280			
							d1	0	2	10	4	8	12			
							123									
							Упоря	Послед								
							116	первый								
							98	второй								
							78	третий								
							78									
2 конкур. точка																

Рис. 3. Результаты поиска порядка с максимальным значением оставшегося времени

Примером выполнения условий, является выполнение установленного порядка запуска в конкурентной точке со значениями оставшегося времени в

диапазоне Y22:Y24, при его фиксации в виде неизменяемой последовательности по порядку их расположения в технологической карте по номеру заказа в диапазоне P22:P24, который остаётся одним и тем же в каждой конкурентной точке. Это необходимо для предоставления информации об оставшемся времени выполнения по каждому заказу в каждой конкурентной точке и правильном сопоставлении определённого значения своему заказу, рис. 4.

=ЕСЛИ(Y21=T15;T16;ЕСЛИ(Y21=U15;U16;ЕСЛИ(Y21=V15;V18;ЕСЛИ(Y21=W15;W17;ЕСЛИ(Y21=X15;X17;ЕСЛИ(Y21=Y15;Y18))))))

P22																
=ЕСЛИ(Y21=T15;T16;ЕСЛИ(Y21=U15;U16;ЕСЛИ(Y21=V15;V18;ЕСЛИ(Y21=W15;W17;ЕСЛИ(Y21=X15;X17;ЕСЛИ(Y21=Y15;Y18))))))																
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y			
15		123	132	231	213	312	321	Послед	123	132	231	213	312	321		
16		10	10	8	8	5	5	первый	106	106	90	90	73	73		
17		18	16	13	18	15	13	второй	80	63	65	98	101	85		
18		23	23	23	23	23	23	третий	55	75	93	55	75	93		
19		51	48	44	49	43	41	всего	241	244	248	243	249	251		
20							d2		10	7	3	8	2	0		
21							Упоря	Послед	321					312		
22							101	первый	73					73		
23							75	второй	85					101		
24		3 конкур. точка					73	третий	93					75		

Рис. 4. Размещение значений оставшегося времени по номеру заказа

Наибольшее число вложенных макросов в количестве 252 было назначено на кнопку Фикс кнопка. На некоторые кнопки было назначено не один, а несколько вложенных макросов, которые обеспечивают перезапуск платформы с последующим сохранением полученного результата. Ещё одним условием, является занесение в поле «порядок» всех возможных вариантов порядков, с учётом установленного порядка и номера передвигаемого заказа на 1-н приоритет вверх.

В результате получено требуемое Решение, состоящее из последовательностей порядков прохождения всех заказов в каждой конкурентной точке в соответствии с их приоритетами выполнения сквозных бизнес-процессов.

Заключение

Предложен алгоритм, позволяющий с участием ЛПР осуществлять выбор порядков запуска нерегулярных, взаимосвязанных, конкурирующих за общие ресурсы бизнес-процессов исходя из условий и ограничений, которые невозможно было предвидеть при оперативном-календарном планировании. В процессе получения требуемого <Решения>, ЛПР принимает непосредственное участие посредством изменения любого приоритета (фиксирование порядка) запуска бизнес-процессов в соответствующей конкурентной точке. Разработана программная платформа <СОВЕТЧИК ЛПР> обеспечивающая в режиме just in time, автоматическую корректировку полученных вариантов решений, исходя из изменившихся предыдущих или появившихся новых условий запуска бизнес-процессов, получая соответствующие результаты вычислений с учётом текущих

условий для последующего анализа и выбора лучшего варианта. Особенностью программной платформы <СОВЕТЧИК ЛПР> является то, что оно позволяет последовательно на каждом этапе определять следующую лучшую последовательность порядков.

Список литературы:

- [1] *Charles Plesums* An Introduction to Workflow// Workflow Handbook.-2002.- http://www.wfmc.org/information/introduction_to_workflow02.pdf.
- [2] *Carol Prior*. Workflow and Process Management// Workflow Handbook.-2003.http://www.wfmc.org/information/Workflow_and_Process_Management.pdf.
- [3] *W.M.P. van der Aalst and S. Jablonski*. Dealing with Workflow Change: Identification of Issues and Solutions. *International Journal of Computer Systems, Science, and Engineering*, 15(5):267–276, 2000.
- [4] *W.M.P. Van der Aalst and T. Basten*. Inheritance of Workflows: An Approach to Tackling. *Problems Related to Change. Theoretical Computer Science*, 270(1-2):125-203, 2002.
- [5] *W.M.P. van der Aalst, J. Desel, and A. Oberweis*, editors. *Business Process Management: Models, Techniques, and Empirical Studies*, volume 1806 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, 2000.
- [6] *A. Agostini and G. De Michelis*. Improving Flexibility of Workflow Management Systems. In *W.M.P. van der Aalst, J. Desel, and A. Oberweis*, editors, *Business Process Management: Models, Techniques, and Empirical Studies*, volume 1806 of *Lecture Notes in Computer Science*, pages 218–234. Springer-Verlag, Berlin, 2000.
- [7] *W.M.P. Van der Aalst and P.J.S. Berens*. Beyond Workflow Management: Product-Driven Case Handling. In *S. Ellis, T. Rodden, and I. Zigurs*, editors, *International ACM SIG-GROUP Conference on Supporting Group Work (GROUP 2001)*, pages 42–51. ACM Press, New York, 2001.
- [8] *Casati and G. Pozzi*. Modeling Exceptional Behaviors in Commercial Workflow Management Systems. In *Proceedings Fourth IFCIS International Conference on Cooperative Information Systems (CoopIS 99)*, IEEE Computer Society, Brussels, 1999.
- [9] *Chalyi S., Levykin I., Petrychenko A., Bogatov I.* (2018). Causality-based model checking in business process management tasks. *IEEE 9th International Conference on Dependable Systems, Services and Technologies DESSERT'2018*. doi: 10.1109/DESSERT.2018.8409176

Поступила в редколлегию 10.10.2019



О.О. Мазурова¹, Т. О. Гордієнко²

¹Кандидат технічних наук, доцент кафедри Програмної інженерії,
Харківський національний університет радіоелектроніки,
oksana.mazurova@nure.ua, ORSID ID: 0000-0003-3715-3476

²Магістр, напрям «Програмна інженерія»,
Харківський національний університет радіоелектроніки,
trsrdrnk@gmail.com, ORSID ID: 0000-0002-9563-7076

ДОСЛІДЖЕННЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ ДЛЯ ПОШУКУ ОПТИМАЛЬНИХ ШЛЯХІВ В СИСТЕМІ ПРОВЕДЕННЯ МЕТРОМАРАФОНІВ

Робота присвячена дослідженню генетичних алгоритмів на прикладі пошуку оптимальних шляхів для підтримки проведення метромарафонів в системах типу «Transit Challenge» або «Subway Challenge». На основі правил проведення метромарафонів та задачі комівояжера сформульована проблема станцій — пошуку оптимального шляху, який би дозволив відвідати всі станції метрополітену за найкоротший час. На базі теорії графів розроблена математична модель системи метромарафону. Для вирішення проблеми станцій розроблено генетичний алгоритм: обрано спосіб представлення геному, правила мутації популяцій і схрещування геномів. На основі експериментального дослідження генетичного алгоритму підібрано найбільш ефективні параметри та розроблено рекомендації стосовно вирішення проблеми станцій для метромарафонів різної розмірності.

ГЕНЕТИЧНИЙ АЛГОРИТМ, ГРАФ, ЗАДАЧА КОМІВОЯЖЕРА, МАТЕМАТИЧНА МОДЕЛЬ, МЕТРОМАРАФОН, ПОШУК ОПТИМАЛЬНОГО ШЛЯХУ, ПРОБЛЕМА СТАНЦІЙ, СХРЕЩУВАННЯ ГЕНОМІВ.

Работа посвящена исследованию генетических алгоритмов на примере поиска оптимальных путей для поддержки проведения метромарафонов в системах типа «Transit Challenge» или «Subway Challenge». На основе правил проведения метромарафонов и задачи коммивояжера сформулирована проблема станций - поиск оптимального пути, который бы позволил посетить все станции метрополитена за кратчайшее время. На базе теории графов разработана математическая модель системы метромарафона. Для решения проблемы станций разработан генетический алгоритм: выбран способ представления генома, правила мутации популяций и скрещивания геномов.

ГЕНЕТИЧЕСКИЙ АЛГОРИТМ, ГРАФ, ЗАДАЧА КОММИВОЯЖЕРА, МАТЕМАТИЧЕСКАЯ МОДЕЛЬ, МЕТРОМАРАФОН, ПОИСК ОПТИМАЛЬНОГО ПУТИ, ПРОБЛЕМА СТАНЦИЙ, СКРЕЩИВАНИЕ ГЕНОМА.

The work is devoted to the study of genetic algorithms on the example of the search for the best ways to support “Transit challenge” or “Subway Challenge” system. Based on the rules of “Transit challenge” and the tasks of the salesman, the problem of stations was formulated - the search for the best way to visit all subway stations in the shortest possible time. Based on the graph theory, a mathematical model of the metrophone system was developed. To solve the station problem, a genetic algorithm has been developed: the method of genome representation, population mutation and genome crossing rules have been chosen. On the basis of the experimental study of the genetic algorithm the most effective parameters were selected and recommendations on the solution of the station problem for “Transit challenge” of different dimensions were developed.

GENETIC ALGORITHM, GRAPHS, TRAVELING SALESMAN PROBLEM, MATHEMATICAL MODEL, UNDERGROUND CHALLENGE, FINDING THE OPTIMAL PATH, STATION PROBLEM, GENOME CROSSING.

Вступ

Метро — це досить популярний засіб пересування у великих містах. З недавніх пір метро стало визивати навіть певний спортивний інтерес. Так у багатьох країнах світу серед мандрівників та просто зацікавлених у такого роду змаганнях популярним став сценарій використання метро в рамках проведення метромарафонів.

Це змагання називають по-різному: «Transit Challenge» чи «Subway Challenge». В країнах СНГ воно більш відомо як метромарафон. Суть метромарафону проста: відвідати всі станції метрополітену за найкоротший час. Існує навіть сайт rapidtransitchallenge.com, де зафіксовані правила та рекорди цього змагання.

Перед тим як почати змагання учасники заздалегідь планують маршрут та час, який буде затрачено.

Складність цього процесу в тому, що чим більший метрополітен, тим більше різних маршрутів існує. А врахування навіть таких незначних деталей, як інтервали між електропоїздами в різний час, час зупинки на станціях та завантаженість, може дати вигреш у часі під час таких змагань.

В Україні є два великих метрополітени — Харківський та Київський, які мають 30 та 52 станції відповідно. Але в Україні поки що такі змагання не проводилися, відсутні також публікації стосовно результатів побудови відповідних маршрутів. Отже, існує практична задача створення системи «Transit Challenge» для проведення українських метромарафонів, та відповідно наукова проблема побудови найкращого маршруту для них.

1. Аналіз основних досліджень

З метою пошуку шляхів метро краще за все моделюють у вигляді графа, де вершини графу – станції, а перегони – ребра. Методів рішення такого роду задач в теорії графів існує багато. В випадку проведення метромарафону, де потрібно знайти маршрут, який включає всі станції, задача зводиться до задачі комівояжера [4]. Вона має досить високу складність обчислення, якщо знаходити її точний розв'язок. Тому часто використовують евристичні методи, які хоч і не дозволяють знайти оптимальний маршрут, але дають гарний компроміс між оптимальністю маршруту та затраченим часом на його пошук.

Деякі з цих евристичних методів намагаються скоротити вже побудований маршрут, видаляючи з маршруту групи ребер та замінюючи їх іншими, що дозволяє отримати новий маршрут. Інші методи комбінують пошук глобальних та локальних розв'язків.

Такі методи як, мурашиний алгоритм, генетичні алгоритми та нейронні мережі використовують ідеї природних процесів та показують досить високу ефективність [1, 2].

2. Постановка задачі

Отже, на основі теорії графів необхідно розробити математичну модель системи метромарафону, яка буде охоплювати всю предметну область разом з існуючими правилами, а також розробити генетичний алгоритм пошуку маршруту для участі в метромарафоні. Також необхідно спланувати та провести експериментальне дослідження алгоритму, виробити рекомендації стосовно параметрів алгоритму та його використання на моделях метромарафону різної розмірності.

3. Розробка математичної моделі системи метромарафону

При моделюванні систему метрополітену доцільно представити у вигляді графів, де вершина – це станція, а дуга – це перехід чи перегон між станціям. Середній час, необхідний для переходу з однієї вершини до іншої, моделюється як вага дуги. Оскільки метрополітен – це дуже налагоджена система, яка працює по чіткому графіку, то такий середній час наближається до фактичного часу здійснення кожного перегону метрополітену.

З пошуком оптимального шляху по всім станціям метрополітену тісно пов'язана проблема комівояжера, так звана (Traveling Salesman Problem, TSP), а також її узагальнена версія Generalized Traveling Salesman Problem (GTSP). В них враховується, що потрібно відвідати всі вершини графа. А в нашому випадку ще й не має ніякого значення, яка лінія та напрямок використовуються для того, щоб відвідати станцію [3].

Якщо представити транспортну мережу у вигляді певного графу, то отримаємо наступну структуру:

- скінченний набір станцій, що позначимо за S .
- скінченний набір ліній метрополітену L , де кожна лінія $L \in L$ – це впорядкований набір станцій.

Припустимо, що кожна лінія працює в обох напрямках і що кожна лінія складається щонайменше з двох станцій. Отже, тепер систему можна змодельовати як орієнтований граф $G = (V, A)$, де V – множина вершин, які зв'язані між собою множиною ребер A . Цей граф має такі властивості:

- для кожної лінії $L \in L$ і всіх станцій $s (s \in L)$ завжди є початкова вершина $\vec{v}(s, L) \in V$;
- для кожної лінії $L \in L$ і всіх станцій $s (s \in L)$ завжди є кінцева вершина $\bar{v}(s, L) \in V$.

Визначимо множину всіх вершин $s \in S$, що належать станції, як:

$$V(s) = \cup_{L \in L, s \in L} \{ \vec{v}(s, L), \bar{v}(s, L) \}.$$

Позначимо дуги, що відповідають сегментам, які утворюють перегін між двома станціями, як $(\vec{v}(s, L), \bar{v}(s', L))$, де s' наслідуює s для всіх $s \in L$. Аналогічно позначимо дугу $(\bar{v}(s, L), \vec{v}(s'', L))$, де навпаки s наслідуює s'' . Позначимо як R набір дуг, що з'єднує вершини тільки однієї лінії та напрямку.

Для моделювання переходів додамо дуги (u, v) для кожної можливої пари вершин у кожному $V(s)$. Такі дуги дозволять змінювати напрямок.

Кожна дуга $a \in A$ має певну вагу $t(a) > 0$. Якщо a належить до R , то будемо вважати, що це час в дорозі між станціями, а якщо $a \notin R$, то це відповідає часу, необхідному на перехід до іншої станції. Отриманий граф назвемо метрографом (графом метрополітену) від L та S .

На основі такої моделі метромарафону можемо сформулювати проблему станцій та проблему перегонів. В рамках предметної області більш цікава проблема станцій, яка полягає в тому, що потрібно відвідати всі вершини змодельованого графу, саме вона задовольняє умовам, при яких має значення відвідування кожної станції, а не проїзд по всім перегонам на лінії.

Користуючись моделлю метромарафону сформулюємо проблему станцій наступним чином: нехай $G = (V, A)$ є графом метрополітену з вагами дуг $t(a)$, де $t(a) > 0$ та $t(a) \in Q$, тоді необхідно знайти найдешевший маршрут T , де кожній станції $s \in S$ принаймні один вузол в V_s належить T .

Проблема станцій дуже тісно пов'язана з асиметричним випадком задачі комівояжера і є NP-повною. Математична модель такої задачі буде виглядати таким чином:

$$\min \sum_{a \in A} t_a x_a,$$

де a – певна дуга, що $a \in A$, $t(a)$ – функція розрахунку ваги ребра a , $x(a)$ – значення з множини $\{1, 0\}$, при якому враховується чи належить дане ребро до оптимального маршруту чи ні.

Також за умовами проблеми станцій ми повинні додати до математичної моделі певні обмеження. Обмеження

$$\sum_{a \in \delta^+(v)} x_a - \sum_{a \in \delta^-(v)} x_a = 0 \quad \forall v \in V$$

гарантує, що рішення складається тільки з набору підмаршрутів. Де $\delta^+(v)$ – набір дуг без вершини v , а $\delta^-(v)$ – набір дуг з вершиною v .

Завдяки другому обмеженню

$$\sum_{a \in \delta^+(V_s)} x_a \geq 1 \quad \forall s \in S$$

кожна станція відвідується принаймні один раз в деякому підмаршруті, де S – це множина станцій.

Обмеження

$$\sum_{a \in \delta^+(C)} x_a + \sum_{a \in \delta^-(C)} x_a \geq 2 \quad \forall C \subset V, C$$

$$x_a \in \mathbb{N}_{\geq 0} \quad \forall a \in A$$

гарантує, що кожна станція з'єднана з усіма іншими станціями. Тобто кожен елемент множини станцій повинен мати хоча б по одному зв'язку в кожному з напрямків.

Розглянемо варіанти функції t , яка відповідає за розрахунок ваги кожної з дуг:

– для розрахунку часу на перегоні між двома станціями використовуємо такі залежності:

$$t = K + F, \quad K \in \mathbb{N}_{\geq 0}, \quad F \in \mathbb{N}_{\geq 0},$$

де K – час, який витрачає потяг на подолання перегону між станціями у хвиликах, F – інтервал руху поїздів в певний час доби (у хвиликах);

– для розрахунку на перехід між двома лініями:

$$t = J + F, \quad J \in \mathbb{N}_{\geq 0},$$

де J – це час на перехід між станціями; слід зазначити, що він може відрізнятись в залежності від того, в якому напрямку необхідно перейти зі станції на станцію.

4. Розробка генетичного алгоритму для пошуку оптимального маршруту

Задача комівояжера є транс обчислювальною та відноситься до класу NP-повних задач. Для Харківського метрополітену кількість можливих рішень складає 2.6525286×10^{32} при кількості станцій 30, а для Київського це число набагато більше – 8.0658175×10^{67} на 52 станціях. Якщо уявити, що комп'ютер здатний аналізувати по 1×10^{23} рішення в секунду, то на розрахунок оптимального шляху знадобиться щонайменше 84 роки.

Ідея генетичних алгоритмів вперше була представлена у 1960 році як техніка оптимізації, яка наближає результат до оптимального рішення [5]. Отже ідея використати генетичні алгоритми для рішення задачі комівояжера є цілком перспективною.

Для застосування генетичного алгоритму потрібно описати основні параметри та функції, які буде застосовано в процесі його виконання. До основних

складових алгоритму належать: популяція, особина, геном та хромосома. Після визначення їх представлення потрібно сформулювати основні функції генетичного алгоритму: функцію відбору, пристосованості, схрещування та функції мутації.

Кожне рішення проблеми метромарфону можна представити як вектор P , що є набором всіх станцій в порядку їх слідування в маршруті. Тоді в алгоритмі вектор P буде грати роль геному або, інакше кажучи, набору хромосом $\{P_1 \dots P_n\}$. Але таке рішення приховує декілька неприємних моментів. По-перше, при такому підході ми отримуємо вектори різної довжини, які буде дуже складно схрещувати між собою, а по-друге, багато векторів буде невалідними. В результаті кожен з геномів потрібно буде перевіряти на валідність, що збільшить складність алгоритму.

Кращим рішенням буде взяти на роль геному перестановку множини станцій метрополітену. Для Харківського метрополітену це перестановка 30. Але при такому підході поряд може опинитися пара станцій між якими немає прямого маршруту. Для вирішення цієї проблеми було вирішено перейти до повного графу. Для цього потрібно буде розрахувати найкоротші маршрути між кожною парою станцій. Складність такого розрахунку висока $O(n^3)$, але для Харківського метрополітену це всього лише 27000 варіантів.

Функцією пристосованості має бути цільова функція, яку можна застосувати в ролі порівняльного показника якості. В даній проблемі ним є час, який буде затрачено на подолання даного маршруту.

Нехай $G_i = \{g_{ik}\}$ ($k = \overline{1, n}$) – хромосомі i -ї особини, де g_{ik} значення k -го гена на i -й особині, а n – кількість генів в хромосомі. Тоді можемо сформулювати умову пристосованості хромосоми:

$$f_i = f(G_i) < f_j = f(G_j),$$

а саме: якщо значення функції пристосованості для хромосоми G_i буде меншим за значення функції для хромосоми G_j , то таку хромосому можна вважати більш пристосованою.

Для розробленого алгоритму було обрано наступні функції мутації, які правильно сходяться між собою та дають змогу вибиратися з локальних оптимумів:

– випадкове перетасування випадкового інтервалу генома; ця мутація є швидкою, але не дуже акуратною; вирішено використовувати її на ранніх етапах еволюції, оскільки в фіналі дуже низька ймовірність поліпшення рішення;

– дзеркальне відображення випадкового інтервалу генома; ця мутація буде використовуватися для виходу з локального оптимуму;

– перестановка K пар випадкових значень в геномі місцями; ця досить непогана, але повільна мутація буде ефективно використовуватися ближче до завершення еволюції;

схрещування двох окремих геномів; в цій мутації будемо обирати точку на геномі, розрізатимемо по ній обидва геноми та мінятимемо їх місцями.

В результаті на кожній ітерації ми застосуємо ті чи інші мутації і формуємо з їх нащадків нове покоління, додаючи до них декілька випадкових маршрутів (схему алгоритму див. на рис. 1).

5. Дослідження генетичного алгоритму

Для дослідження розробленого генетичного алгоритму використано наступні графи:

- граф на основі маршрутної схеми Харківського метрополітену (30 вершин);
- граф на основі маршрутної схеми Київського метрополітену (52 вершини);

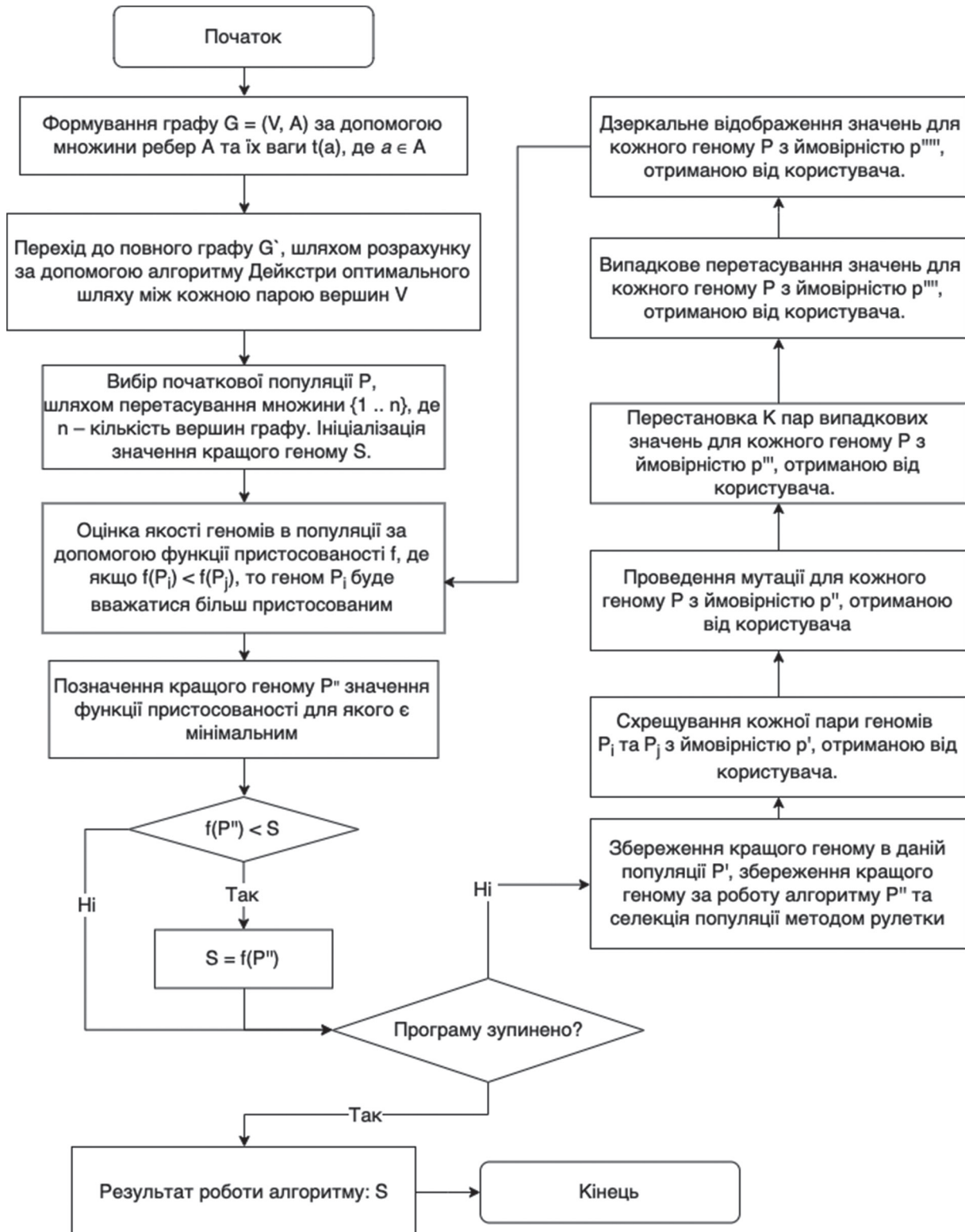


Рис. 1. Схема розробленого генетичного алгоритму

– граф, який імітує транспортну схему з 10 переходами та 50 вершинами, що дозволив дослідити алгоритм в складніших умовах.

Для коректного порівняння ефективності потрібно було обрати час для експериментальної роботи алгоритму. Для цього було проведено кілька пробних запусків алгоритму при різних параметрах на графі Харківського метрополітену. На графіку, наведеному на рисунку 2, точками зображено значення функції пристосованості та час, за який вони були знайдені. З графіку видно, що після 60 секунд алгоритм втрачає темп покращення результату та демонструє стагнацію.

Під час дослідження вирішено змінювати наступні параметри алгоритму:

- кількість особин в популяції – g ;
- ймовірність схрещування геномів – p' ;
- ймовірність проведення мутації – p'' ;
- ймовірність застосування мутації перестановки декількох пар (p'''), випадкового перетасування (p'''') та дзеркального відображення (p'''''').

Алгоритм було протестовано на 5 наборах даних по 3 рази на кожному графі. Кожен тест тривав одну хвилину. В першому наборі даних ймовірність мутації встановлено на 95%. Цей набір має на меті перевірити залежність результату від кількості мутацій. Другий набір даних більш стандартний для такого роду задач

– невелика ймовірність мутацій (5%) та велика схрещування (95%). Третій набір має на меті перевірити роботу з популяцією в 50 особин, тобто більше чим в 2 рази більшою, чим в інших наборах. Четвертий набір базується на зміні ймовірності мутацій і схрещування в процесі роботи програми. Збільшення ймовірності мутації з перестановкою пар та зменшення з випадковим перетасуванням, а також збільшення кількості схрещувань. У п'ятому наборі стратегія зміни параметрів повністю протилежна попередньому пункту. Набори параметрів для проведення експерименту наведено в таблиці 1.

Кращі результати дослідження (вимірюються в хвилину проходження оптимального маршруту) на всіх трьох графах наведено в таблиці 2.

Таблиця 1

Ймовірність/№	1	2	3	4	5
Мутації взагалі	95%	5%	10%	10%	10%
Мутації з перестан. пар	10%	5%	10%	5→30%	30→5%
Мутації з перетасуванням	10%	5%	10%	30→5%	5→30%
Мутації дзерк. відображення	5%	5%	5%	5%	5%
Схрещування	1%	98%	80%	10→80%	80→10%
К-сть особин в популяції, шт	20	20	50	20	20

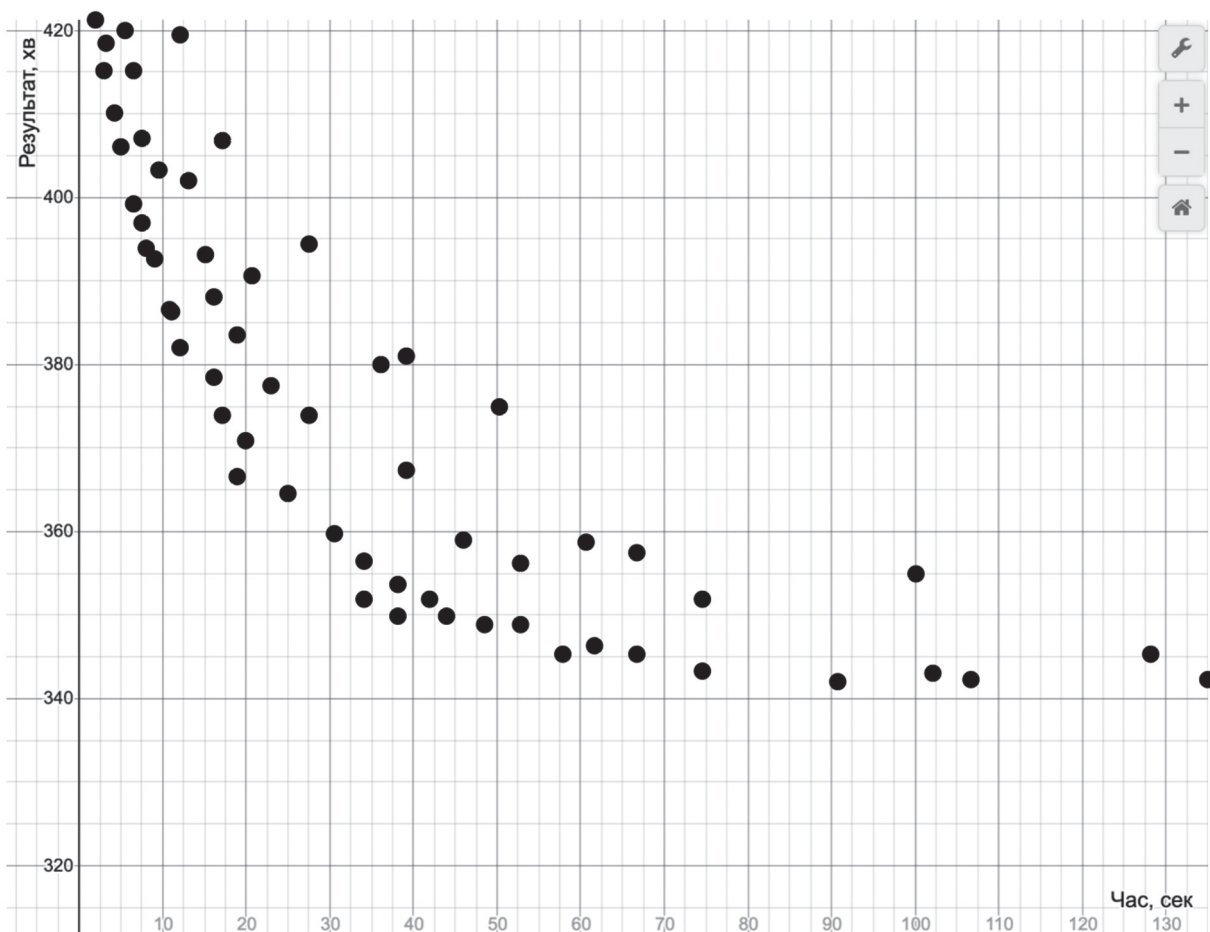


Рис. 1. Схема розробленого генетичного алгоритму

Таблиця 2

Граф	Експеримент номер				
	1	2	3	4	5
Харківський	314	252	262	242	258
Київський	572	415	409	409	421
Штучний	657	430	445	414	433

Аналізуючи на порівнюючи отримані дані, можна зробити висновок, що найоптимальнішою виявився набір параметрів номер 4, як і було прогнозовано. Набір даних №1, де ймовірність мутації була найвищою, показав найгірший результат. Це пояснюється тим, що особини дуже швидко мутували, не встигаючи обмінятися між собою генами.

Другою частиною експерименту є порівняння результату звичайної роботи алгоритму з початковими параметрами та роботи з ручним корегуванням параметрів. Очікувалось невелике зменшення фінального результату при ручному корегуванні параметрів. Тестування було проведено зі штучним графом, що має 50 вершин та 10 переходів. Кращі результати другого етапу наведено в таблиці 3.

Отже, зміна параметрів хоч і не суттєво, але впливає на фінальний результат. В результаті в кожному експерименті був знайдений більш оптимальний маршрут.

Таблиця 3

Вид тестування	Експеримент номер	
	1	2
Без корегування	427 хв.	421 хв.
З корегуванням	410 хв.	409 хв.

Отже, під час вирішенні проблеми метромарафону за допомогою генетичних алгоритмів не рекомендовано задавати високу ймовірність мутацій або змінювати її за межі 10 відсотків. Мутація з випадковою перестановкою краще працює на початку роботи алгоритму. Після швидкої зміни значень варто спробувати зменшити її ймовірність та збільшити ймовірність мутації з перестановкою пар. Якщо значення довго перебуває в локальному оптимумі, варто збільшити вірогідність мутації з дзеркальним відображенням або ймовірність схрещування.

Висновки та перспективи

В роботі на основі теорії графів запропоновано математичну модель системи метромарафону та сформульовано проблему станцій як проблему пошуку оптимального маршруту для метромарафону, який би охопив всі станції метрополітену. Для зниження складності вирішення цієї NP-повною задачі було запропоновано пошук оптимального маршруту за допомогою генетичних алгоритмів.

В розробленому алгоритмі в якості геному було взято перестановку по кількості станцій метрополітену та перейдено до повного графу, сформульовано правила схрещування геномів між собою, що дозволяють вибиратися з локальних оптимумів.

Результати експериментального дослідження дозволили знайти оптимальні значення для параметрів алгоритму, що дозволяє знайти найкращий маршрут в метрополітені за більш-менш реальний час, не застосовуючи повний перебір варіантів.

Список літератури:

- [1] *Murat Albayrak and Novruz Allahverdi*, Development a new mutation operator to solve the traveling salesman problem by aid of genetic algorithms. *Expert Systems with Applications*, vol. 38, no. 3, 2011 – pp. 1313–1320.
- [2] *Chakroborty, P., and Dwivedi, T.* Optimal route network design for transit system using genetic algorithms. *Optimization and Engineering*. 34(1), 200 с. — 2003.
- [3] *Dijkstra, E.W.* A note on two problems in connexion with graphs. *Numerische Mathematik*. 1, 471 с. — 1959.
- [4] *Fan L. and Mumford C. L.* A Metaheuristic Approach to the Urban Transit Routing Problem. *Journal of Heuristic*. 16, 372 с. — 2010..
- [5] *Ngamchai, S. and Lovell, D.* Optimal time transfer in bus transit route network design using a genetic algorithm, *Journal of Transportation Engineering*. 129 (5), 510–521 с. — 2003.
- [6] *Baaj, M. H., and Mahmassani, H. S.* An AI-based approach for transit route system planning and design. *Journal of Advanced Transportation*. 25(2), 287 с. — 1990.

Надійшла до редколегії 4.11.2019



А.О. Функендорф¹, В.О. Бортнікова², І.В. Груздо³, Ю.Ю. Черепанова⁴

¹ ХНУРЕ, м. Харків, Україна, anastasiia.funkendorf@nure.ua;

² Укснаб ЧАО, м. Харків, Україна, viktoriiia.bortnikova@nure.ua

³ ХНУРЕ, м. Харків, Україна, irina.guzdo@nure.ua

⁴ ХНУРЕ, м. Харків, Україна, yulia.cherepanova@nure.ua

МАТЕМАТИЧНА МОДЕЛЬ ПРОЦЕСУ ОТРИМАННЯ НЕРОЗ'ЄМНИХ З'ЄДНАНЬ У КОНСТРУКЦІЯХ МЕМС МЕТОДОМ УЛЬТРАЗВУКОВОГО ЗВАРЮВАННЯ

Запропоновано підхід до адаптації процесу отримання нероз'ємних з'єднань методом ультразвукового зварювання в рамках технології пакування МЕМС на рівні пластин. Представлено конструкційне рішення запропонованого підходу, а також математична модель процесу його.

МЕМС, УЛЬТРАЗВУКОВЕ ЗВАРЮВАННЯ, WLP ТЕХНОЛОГІЯ, МАТЕМАТИЧНА МОДЕЛЬ

Предложен подход к адаптации процесса получения неразъемных соединений методом ультразвуковой сварки в рамках технологии упаковки МЭМС на уровне пластин. Представлено конструктивное решение предложенного подхода, а также математическая модель процесса его реализации.

МЭМС, УЛЬТРАЗВУКОВАЯ СВАРКА, WLP ТЕХНОЛОГИЯ, МАТЕМАТИЧЕСКАЯ МОДЕЛЬ

An approach to the adaptation of the fixed connections by ultrasonic welding in the framework of MEMS packaging technology at the plate level obtaining process is proposed. The structural solution of the proposed approach is presented, as well as the mathematical model of the implementation process.

MEMS, ULTRASONIC WELDING, WLP TECHNOLOGY, MATHEMATICAL MODEL

Вступ

Процес корпусування мікроелектромеханічних систем (МЕМС) є одним з найбільш важливих етапів їх виробництва. У більшості випадків однією з обов'язкових умов для його реалізації є досягнення вакуумізації внутрішніх порожнин (наприклад, у МЕМС ємнісних датчиках тиску). Це може бути досягнуто при упакуванні МЕМС в дискретні корпуси, що є досить складним і дорогим підходом. Аналогічним рішенням є корпусування МЕМС на рівні пластини (WLP – Wafer-Level Packaging), що дозволяє значно спростити і здешевити технологічний процес їх виробництва [1].

В рамках вимог до герметизації корпусів реалізація цього процесу досягається за рахунок використання технології термічного скріплення, яка вимагає температури значно вищої, ніж допустима для окремих МЕМС робоча температура. Можливість виникнення додаткових механічних напружень на межі між матеріалами окремих пластин внаслідок термічної деформації, може спричинити виникнення критичних похибок у роботі МЕМС пристроїв [2].

Ультразвукове (УЗ) зварювання в МЕМС на даний момент знаходить широке застосування переважно в рамках технології фліп-чіп, проте воно також може бути застосована і до WLP. Цей метод отримання нероз'ємних з'єднань має ряд переваг, а саме відсутність тривалого теплового нагріву поверхонь та сусідніх елементів, можливість використання для різних типів матеріалів зварювальних елементів і проміжних шарів, низькі вимоги до попереднього очищення поверхонь, що зварюються, а також достатню простоту автоматизації. Застосування даного методу

отримання нероз'ємних з'єднань в рамках технологій WLP є складним наукомістким завданням, що вимагає розробки нових підходів та математичних моделей для його реалізації з урахуванням високих вимог до герметизації корпусів МЕМС.

1. Адаптація методу ультразвукового зварювання для його застосування в рамках технології пакування МЕМС на рівні пластин

WLP метод заснований на з'єднанні пластин зі сформованими функціональними структурами з пластиною-кришкою. Даний процес дозволяє істотно спростити і здешевити технологію, оскільки робить можливою корпусування одночасно декількох пристроїв, розташованих на одній пластині (рис. 1) [1, 3].



Рис. 1. WLP метод пакування МЕМС датчиків

На даний момент існує безліч підходів для корпусування МЕМС на рівні пластин. Більшість з них засновані на отриманні нероз'ємних з'єднань між пластинами з проміжним шаром, або ж без нього. В якості проміжних шарів можуть виступати різні легкоплавкі метали і сплави, полімери, скло керамічні припої, геттери.

Застосування УЗ зварювання на етапі корпусування в рамках технології WLP, а також базовий похід без додаткового нагріву елементів був запропонований і обґрунтований в роботі [2]. Відповідно, адаптація даного методу отримання нероз'ємного з'єднання зі збереженням вимог до герметизації може бути досягнута за рахунок зварювання пластин в закритому боксі з відкачуванням повітря. Сам процес відведення повітря пропонується реалізувати через трубку, приєднану до боксу, з додатковим підключенням до неї відповідного обладнання, що дозволяє регулювати необхідний рівень тиску. Ущільнювальний елемент в рамках запропонованого підходу повинен бути виконаним з щільного, еластичного і герметичного матеріалу. Це дозволить забезпечувати необхідний рівень герметизації ємності, в якій проходить зварювання пластин. Візуально конструкційне рішення адаптації УЗ зварювання в рамках WLP технології, засноване на використанні поздовжньої коливальні системи [4, 5], представлено на рис. 2.

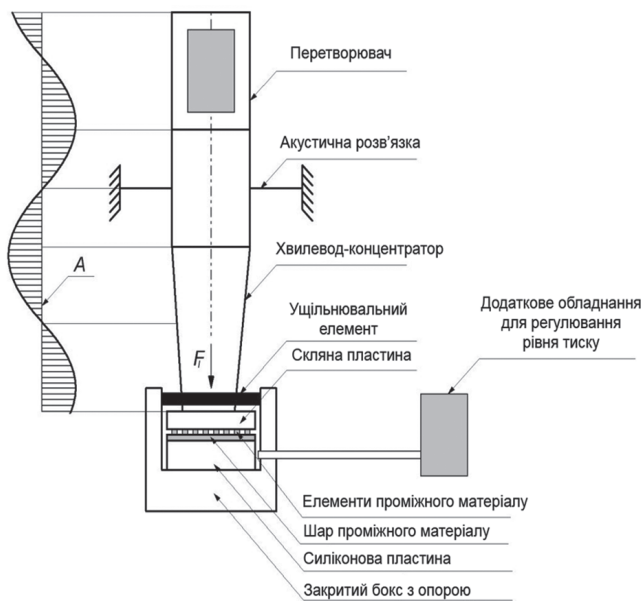


Рис. 2. Конструкційне рішення запропонованого підходу

У конструкціях такого типу важливим є розташування поверхонь, що зварюються, і площини опори до виводу хвильовода-концентратора, що забезпечує ортогональність спрямованості поширення УЗ хвиль щодо зварювальних поверхонь і відповідне напрямку вектора сили зварювального зусилля F_l [6].

Також, для ефективності запропонованої адаптації, додатковою умовою є наявність проміжного шару між пластинами, виконаного з твердого матеріалу (алюміній). Цей шар конструктивно забезпечує доступ до внутрішньої порожнини MEMS пристрою, що є необхідним для виконання умови забезпечення її герметизації. У процесі зварювання елементи проміжного шару поступово рівномірно розподіляються

між пластинами, тим самим зменшуючи зазор між ними і закриваючи внутрішню порожнину пристрою.

Дослідження запропонованого підходу вимагає проведення ряду підготовчих етапів, які включають в себе розробку відповідної моделі технологічного процесу в рамках його автоматизації. Так, структурно-параметричну модель зазначеного процесу в узагальненому вигляді можна представити таким чином:

$$M = \begin{cases} X = \{X_1, X_2, X_3, \dots, X_n\}; \\ U = \{U_1, U_2, U_3, \dots, U_k\}; \\ Y, \end{cases} \quad (1)$$

де M – модель процесу УЗ зварювання, n – кількість вхідних параметрів X , що впливають на процес (основних технологічних параметрів), k – кількість випадкових параметрів U , що впливають на процес (додаткових параметрів), Y – параметр, за яким оцінюється якість отриманого з'єднання, (згідно до роботи [5] зусиллям на розрив).

Виходячи з цього, в рамках запропонованого підходу до адаптації, можна виділити п'ять основних технологічних параметрів процесу корпусування MEMS на рівні пластин УЗ зварюванням: зусилля навантаження зварювального інструменту F_l , амплітуда його коливань A , вихідна потужність УЗ генератора P_g , час зварювання t і тиск P , необхідний для досягнення герметизації корпусу. Тоді, вплив зазначених параметрів на якість отриманого зварювального з'єднання може бути описано моделлю:

$$Y = f(F_l, A, P_g, t_{US}, P). \quad (2)$$

В результаті апроксимації представленої моделі лінійною функцією модель адаптованого для WLP процесу УЗ зварювання набуде вигляду:

$$Y = a_0 + a_1 F_l + a_2 A + a_3 P_g + a_4 t_{US} + a_5 P. \quad (3)$$

де $a_0, a_1, a_2, a_3, a_4, a_5$ – питомі параметри моделі, що дозволяють прогнозувати значення і поведінку системи в цілому, з урахуванням впливу випадкових параметрів $U(a_0)$.

Для адекватності моделі також необхідно враховувати ефекти взаємодії між її основними технологічними параметрами. Використання методу активного планування в рамках даного дослідження надає змогу варіювати ряд факторів і отримувати одночасно кількісні оцінки для всіх ефектів. При цьому, на відміну від класичного регресійного аналізу, надає змогу уникнути кореляції між коефіцієнтами рівняння регресії. Таким чином, при статичному підході математична модель процесу отримання нероз'ємних з'єднань методом УЗ зварювання може бути представлена в загальному вигляді відрізком ряду Тейлора, в який розкладається невідома функція:

$$\begin{aligned}
 Y = f(F_l, A, P_g, t, P) = & b_0 + \\
 & + b_1 F_l + b_2 A + b_3 P_g + b_4 t + b_5 P + \\
 & + b_{12} F_l A + b_{13} F_l P_g + b_{14} F_l t + b_{15} F_l P + \\
 & + b_{23} A P_g + b_{24} A t + b_{25} A P + b_{34} P_g t + \\
 & + b_{35} P_g P + b_{45} t P + b_{123} F_l A P_g + \\
 & + b_{124} F_l A t + b_{125} F_l A P + b_{234} A P_g t + \\
 & + b_{235} A P_g P + b_{345} P_g t P + b_{341} A P_g F_l + \\
 & + b_{351} P_g P F_l + b_{451} t P F_l + b_{452} t P A + \\
 & + b_{1234} F_l A P_g t + b_{1235} F_l A P_g P + \\
 & + b_{2345} A P_g t P + b_{3451} P_g t P F_l + \\
 & + b_{4512} t P F_l A + b_{12345} F_l A P_g t P
 \end{aligned}
 \tag{4}$$

де b_0 – вільний член, що дозволяє враховувати вплив випадкових параметрів U , $b_1 - b_5$ – лінійні ефекти, $b_{12} - b_{45}$ – ефекти парної взаємодії, $b_{123} - b_{452}$ – ефекти потрійної взаємодії, $b_{1234} - b_{4512}$ – ефекти четверної взаємодії, b_{12345} – ефект п'ятірної взаємодії.

Базуючись на раніше проведених дослідженнях в даній області [2, 7, 8] можна визначити граничні значення оптимальних технологічних параметрів, для побудови моделі (табл. 1). При цьому, граничні значення тиску в боксі планується отримати в ході подальшого натурного експерименту, а значення амплітуди коливальної системи в реальних пристроях, як правило, характеризується їх конструкцією. Тому, в рамках зменшення розмірності задачі, для визначення значення було прийнято рішення про побудову математичної моделі процесу з урахуванням трьох основних регульованих технологічних параметрів, граничні значення яких наведені в табл. 1.

Таблиця 1

Значення регульованих параметрів

Параметр	Діапазон значень
P_g	30~50 Вт
t	2.5~5 м
F_l	20,8~40,1 МПа

2. Розробка математичної моделі адаптації УЗ зварювання в рамках WLP технології корпусування МЕМС

На основі отриманих в ході моделювання [9] і зазначених в табл. 1 даних, згідно планування повного факторного експерименту (ПФЕ) [10-12], була складена матриця ПФЕ, на основі якої отримано наступні математичні моделі.

Розроблена математична модель адаптації УЗ зварювання в рамках WLP технології корпусування МЕМС без урахування ефектів взаємодії, що ґрунтується на матриці планування (табл. 2), має вигляд:

$$Y = 206,49 - 1,11P_g - 4,8t - 0,61F_l. \tag{5}$$

Коефіцієнт детермінації для цієї моделі становив 0,58, що є досить низьким показником, який свідчить про недостатню адекватність моделі і потребує додаткової побудови моделі з урахуванням ефектів взаємодії.

Співвідношення реальних и отриманих в ході моделювання даних для цієї моделі показано на рис. 3.

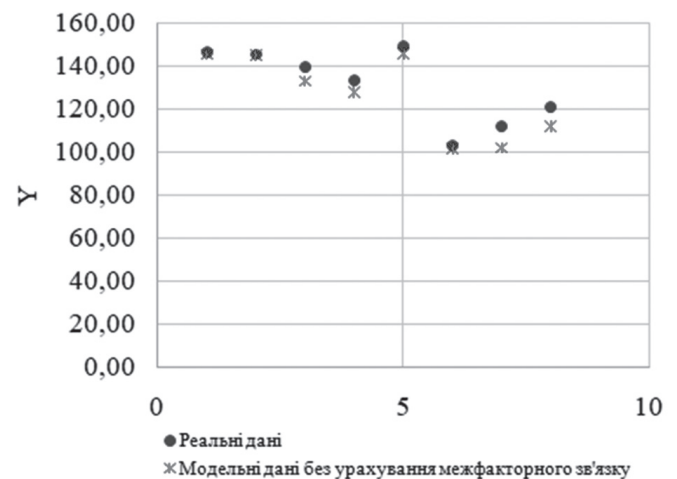


Рис. 3. Порівняння експериментальних та модельних даних

Отримана математична модель, заснована на розширеній матриці планування (табл. 2) з урахуванням важливості коефіцієнтів має вигляд:

Таблиця 2

Матриця планування

P_g	t	P	Y	a_1	a_2	a_3	b
30,00	2,90	20,80	146,54	-1,11	-4,8	-0,61	206,49
30,00	2,90	40,10	145,45	-1,11	-4,8	-0,61	206,49
30,00	5,00	20,80	139,42	-1,11	-4,8	-0,61	206,49
30,00	5,00	40,10	133,16	-1,11	-4,8	-0,61	206,49
50,00	2,90	20,80	149,27	-1,11	-4,8	-0,61	206,49
50,00	2,90	40,10	103,13	-1,11	-4,8	-0,61	206,49
50,00	5,00	20,80	112,26	-1,11	-4,8	-0,61	206,49
50,00	5,00	40,10	121,02	-1,11	-4,8	-0,61	206,49

$$Y = 9,1P_g + 65,86t + 10,27F_l - 2,3P_g t - 0,33P_g F_l - 2,35tF_l + 0,074P_g tF_l - 118,1. \quad (6)$$

Порівняння експериментальних і отриманих в ході моделювання [9] даних з урахуванням ефектів взаємодії між основними технологічними параметрами представлено на рис. 4.

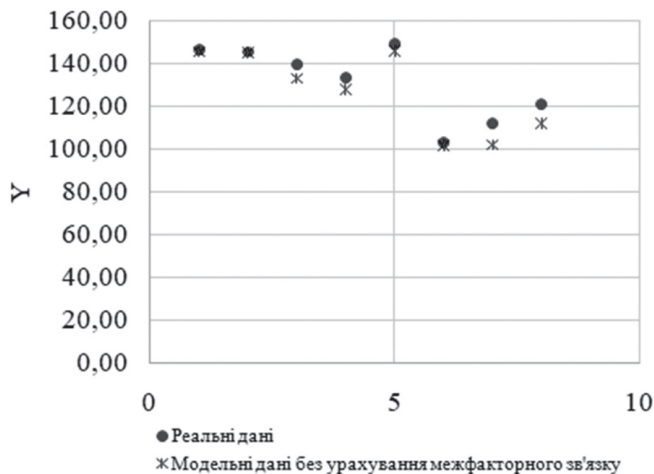


Рис. 4. Порівняння експериментальних та модельних даних з урахуванням ефектів взаємодії між основними параметрами

Коефіцієнт детермінації для розширеної моделі склав 0,98. Це є досить високим показником і говорить про мінімізацію помилки між експериментальними і розрахунковими даними.

Висновки

Запропонована адаптація методу ультразвукового зварювання для його застосування в рамках технології пакування МЕМС на рівні пластин. Представлено його конструктивне рішення, розроблене на основі поздовжньої коливальні системи.

Розроблена структурно-параметрична модель відповідного технологічного процесу, із застосуванням якої було виділено п'ять основних технологічних параметрів: зусилля навантаження зварювального інструменту, амплітуда його коливань, вихідна потужність УЗ генератора, час зварювання і тиск, необхідний для досягнення герметизації корпусу.

На основі отриманих в ході попередніх досліджень результатів та імітаційного моделювання була розроблена математична модель, що відображає функціональні взаємозв'язки між основними технологічними параметрами процесу і результирующим показником якості УЗ зварних з'єднань – зусилля на розрив.

Таблиця 3

Розширена матриця планування

P_g	t	P	$P_g \cdot t$	$P_g \cdot x_3$	$t \cdot P$	$P_g \cdot t \cdot P$	Y
30,00	2,90	20,80	87,00	624,00	60,32	1809,60	146,54
30,00	2,90	40,10	87,00	1203,00	116,29	3488,70	145,45
30,00	5,00	20,80	150,00	624,00	104,00	3120,00	139,42
30,00	5,00	40,10	150,00	1203,00	200,50	6015,00	133,16
50,00	2,90	20,80	145,00	1040,00	60,32	3016,00	149,27
50,00	2,90	40,10	145,00	2005,00	116,29	5814,50	103,13
50,00	5,00	20,80	250,00	1040,00	104,00	5200,00	112,26
50,00	5,00	40,10	250,00	2005,00	200,50	10025,00	121,02
a_1	a_2	a_3	a_4	a_5	a_6	a_7	b
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1
9,1	65,86	10,27	-2,3	-0,33	-2,35	0,074	-118,1

Представлена модель дозволяє визначати необхідні значення основних технологічних параметрів процесу, а також стає основою для подальшої реалізації запропонованого рішення адаптації метода УЗ зварювання в рамках WLP технології корпусування МЕМС.

Список літератури:

- [1] *E. Esashi*, "Wafer level packaging of MEMS", 2009 International Solid-State Sensors, Actuators and Microsystems Conference (TRANSDUCERS 2009), Denver, Colorado, 21-25 June, 2009, pp. 9-16.
- [2] *P.I. Eswaran, S. Malarvizhi* "MEMS Capacitive Pressure Sensors: A Review on Recent Development and Prospective", International Journal of Engineering and Technology (IJET), Vol 5 No 3 Jun-Jul 2013, pp. 2734-2746.
- [3] *Ephraim Suhir, C. P. Wong, Y. C. Lee*, "Micro- and Opto-Electronic Materials and Structures: Physics, Mechanics, Design, Reliability, Packaging",
- [4] *З.Ю. Готра*, "Технологія електронної техніки: навч. посібник у 2 т." – Львів: Видавництво Національного університету "Львівська політехніка", 2010. — 884 с.
- [5] *G. Harman*, "Wire Bonding in Microelectronics: Materials, Processes, Reliability, and Yield", 3rd ed., McGraw Hill, New York, 2009.
- [6] *И.Ш. Невлюдов, А.В. Пономарева, А.О. Функендорф, К.Л. Хрусталева*, "Математическая модель технологического процесса получения неразъемных соединений в конструкциях роботов модульного типа". Доклады БГУИР. 2018, №4. — с. 51-57.
- [7] *V. M. Borshchov, O. M. Listratenko, M. A. Protsenko*, "Innovative microelectronic technologies for high-energy physics experiments", VFunctional materials, vol. 24, pp. 143-153, 2017.
- [8] *S. Y. Kang, P. M. Williams, T. S. McLaren, and Y. C. Lee*, "Studies of thermosonic bonding for flip-chip assembly," Materials Chemistry Phys., vol. 42, pp. 31-37, 1995
- [9] *A. Funkendorf, V. Bortnikova, S. Maksymova, M. Melnyk*, "Mathematical Model of Adapted Ultrasonic Bonding Process for MEMS Packaging," IEEE XV-th International Conference Perspective Technologies and Methods in MEMS Design (MEMSTECH), Lviv-Polyana, Svalyava, (Zakarpattya), Ukraine, May 22-26, 2019. — P. 79-82.
- [10] *J.O. Rawlings*, "Applied Regression Analysis". Pacic Grove, CA: Wadsworth & Brooks/Cole, 1988.
- [11] *E. Harrell Jr.*, "Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis" (Springer Series in Statistics), Springer, 2015.
- [12] *G. Ciaburro* "Regression Analysis with R: Design and develop statistical nodes to identify unique relationships within data at scale", Packt Publishing, 2018.

Надійшла до редколегії 19.12.2019



Konarieva I.¹, Pydorenko D.², Turuta O.³

¹Master student of the University of Computense, Madrid (UCM), Spain,
iulikona@ucm.es, ORCID ID: 0000-0001-9266-9877

² Master student of the Department of Software Engineering,
Kharkiv National University of Radio Electronics, Kharkiv, Ukraine,
daria.pydorenko@nure.ua, ORCID ID: 0000-0003-0232-4634

³ PhD in Computer Science, Associate Professor of the Department of Software Engineering,
Kharkiv National University of Radio Electronics, Kharkiv, Ukraine,
oleksii.turuta@nure.ua, ORCID ID: 0000-0002-0970-8617

A SURVEY OF METHODS OF TEXT-TO-IMAGE TRANSLATION

The given work considers the existing methods of text compression (finding keywords or creating summary) using RAKE, Lex Rank, Luhn, LSA, Text Rank algorithms; image generation; text-to-image and image-to-image translation including GANs (generative adversarial networks). Different types of GANs were described such as StyleGAN, GauGAN, Pix2Pix, CycleGAN, BigGAN, AttnGAN. This work aims to show ways to create illustrations for the text. First, key information should be obtained from the text. Second, this key information should be transformed into images. There were proposed several ways to transform keywords to images: generating images or selecting them from a dataset with further transforming like generating new images based on selected or combining selected images e.g. with applying style from one image to another. Based on results, possibilities for further improving the quality of image generation were also planned: combining image generation with selecting images from a dataset, limiting topics of image generation.

IMAGE GENERATION, TEXT KEYWORDS, IMAGE-TO-IMAGE TRANSLATION, TEXT-TO-IMAGE TRANSLATION, TEXT COMPRESSION

1. State of the Art

Online publications are currently very common. Their advantage is simplicity and accessibility for all people. A publication of their works is no longer unattainable for most people. Access to the Internet solves almost every possible problem.

One of the components of a successful, attention-grabbing publication is the illustrations that accompany it. Not everyone can create its unique illustration or buy the rights to someone else's picture. Images in the public domain often repeat and quickly become boring. How, then, can you create an illustration without art skills?

One of the options may be the generation of images by software.

There are neural networks such as GAN. GAN is a generative-competitive network, which is based on a combination of two neural networks, one of which generates candidates, and the other tries to distinguish the right candidates from the wrong ones.

There are a lot of types of GAN, and each of them does its job.

For example, the face generation is very popular now. There is even a website that creates faces of non-existing people. This creating process is based on style-based GAN. The architecture of StyleGAN leads to an automatically learned, unsupervised separation of high-level attributes (e.g., pose and identity when trained on human faces) and stochastic variation in the generated images (e.g., freckles, hair), and it enables intuitive, scale-specific control of the synthesis [1].

One of the implementations of StyleGAN architecture is provided by NVIDIA labs.

Also, NVIDIA presented GauGAN network in March 2019 and provided an interactive app that generates realistic landscape images from the layout users draw. GauGAN allows user control over both semantic and style of created images [2].

Generally, there are a lot of ways to translate image to image, e.g. Pix2Pix and CycleGAN.

Pix2Pix trains to translate images basing on pairs of images {A,B}, where A and B are two different depictions of the same underlying scene. This architecture has examples of translating labels to facades, changing day to night, edges to photo, or black-white image to colorful [3].

CycleGAN is an unpaired version of Pix2Pix architecture. It can translate an image from a source domain X to a target domain Y in the absence of paired examples. This architecture is used for changing weather or season on the photo, or for applying a style of a specific artist [4].

Also, there is a neural style algorithm that can apply a style of one image to another [5].

There are a lot of GANs that can generate images basing on specific labels. They are called conditional GANs.

One of improving such GANs is BigGAN. It creates very realistic images for the specific class. It can be trained with complex datasets such as ImageNet, for example. One of the observed failures of partially-trained BigGAN models is a class leakage, where images from one class contain properties of another. Resulted images can be translated from one class to another [6].

Another improving of conditional GAN is AttnGAN. AttnGAN allows changing the generated image in a multi-stage manner according to the change of individual words in the text description [7].

AttnGAN uses a variational autoencoder. Autoencoders are neural networks that copy their inputs to their outputs. They work by compressing the input into a latent-space representation and then reconstructing the output from this representation. This kind of network is composed of two parts: encoder and decoder.

Variational autoencoders (unlike vanilla autoencoders) allow creating data similar to existing data or even changing it in a specific direction. Variational autoencoders are trained to extend their latent space generating possible input and feed it to the decoder to generate new data samples [8].

Based on variational autoencoders, different GANs can be built.

Also, there is a SinGAN, an unconditional generative model that can be learned from a single natural image. It can generate high quality, diverse samples that carry the same visual content as the image.

SinGAN contains a pyramid of fully convolutional GANs, each responsible for learning the patch distribution at a different scale of the image. This allows generating new realistic samples of arbitrary size and aspect ratio, that have significant variability, yet maintain both the global structure and the fine textures of the training image. In contrast to other single image GAN schemes, SinGAN approach is not limited to texture images, and is not conditional (i.e. it generates samples from noise) [9].

2. Problem Statement and Proposed Solution

There are a lot of ways to generate images basing on input images or text description. But how can this information be received from a big text?

So, the problem of generating illustrations from a text can be divided into two parts. The first one is text compression to appropriate data for input. The second one is translating text to image or selecting images from a dataset.

Consider each of the steps in more detail.

2.1 Text Compression

There are various ways to work with text: generating annotations, finding key phrases, or choosing certain words or word combinations. One of them is Rapid Automatic Keyword Extraction (RAKE) algorithm. RAKE is an algorithm to automatically extract keywords from documents [10].

Another way is natural language processing (NLP), which provides a lot of abilities to understand what text is about. For example, tokenization (the process of splitting text to words, sentences, paragraphs or other types of tokens) is one of the basic components of almost any NLP task, and it can be the first step to prepare a text for processing.

Extractive text summarization techniques perform summarization by picking portions of texts and constructing a summary, unlike abstractive techniques that conceptualize a summary and paraphrases.

There are several unsupervised graphical-based text summarizers such as Lex Rank or Text Rank.

In original TextRank the weights of an edge between two sentences is the percentage of words appearing in both of them.

Lex Rank uses IDF-modified Cosine as the similarity measure between two sentences. This similarity is used as a weight of the graph edge between two sentences. Lex Rank also incorporates an intelligent post-processing step which makes sure that the top sentences chosen for the summary are not too similar to each other.

Luhn is an algorithm that scores sentences based on the frequency of the most frequent (significant) words. It ranks sentences for summarization extracts by considering significant words and the linear distance between these words due to non-significant words.

LSA works by projecting the data into a lower-dimensional space without any significant loss of information. One way to interpret this spatial decomposition operation is that singular vectors can capture and represent word combination patterns that are recurring in the corpus. The magnitude of the singular value indicates the importance of the pattern in a document [11].

Also, there is Parts-Of-Speech tagging (POS tagging) and is also known as word classes or lexical categories. The task of POS-tagging is to labeling words of a sentence with their appropriate Parts-Of-Speech (Nouns, Pronouns, Verbs, Adjectives, etc.) [12].

For example, only nouns can be selected from the text, because nouns characterize the text most of all. Or a combination of a noun and a verb can be chosen. These word combinations show what actors are in the texts (nouns) and what actions they perform (verbs) – that is almost the main idea of the text.

2.2 Image Generation

So, images should be generated based on the keywords received in one of the ways described in paragraph 2.1.

GAN networks can be used for image generation, for example, AttnGAN.

Also based on a combination of nouns and verbs (as described in paragraph 2.1) images of people can be generated. There are many ways to generate faces, but it is better to generate the whole person who performs a specific action (so have the specific pose).

There is Pose Guided Person Image Generation network that allows synthesizing person images in arbitrary poses, based on an image of that person and a novel pose. A generation framework PG2 utilizes the pose information explicitly and consists of two key stages: pose integration and image refinement. It allows simultaneously transferring the appearance of a person from a given pose to a desired pose and keep important appearance details of the identity. Each stage is focusing on one aspect. For the first stage, several model variants can be used and, for the second stage, conditional DCGAN is used to fill in more appearance details [13-15].

Another way is not to generate an image (because the result can be unpredictable), but to find specific images by labels in datasets.

The dataset should have not only images but labels, which describe each picture. Images can be selected using these labels. Image labels should match input keywords as much as possible. An example of a suitable dataset is ImageNet.

A full process of generating images for the text is shown in Fig. 1.

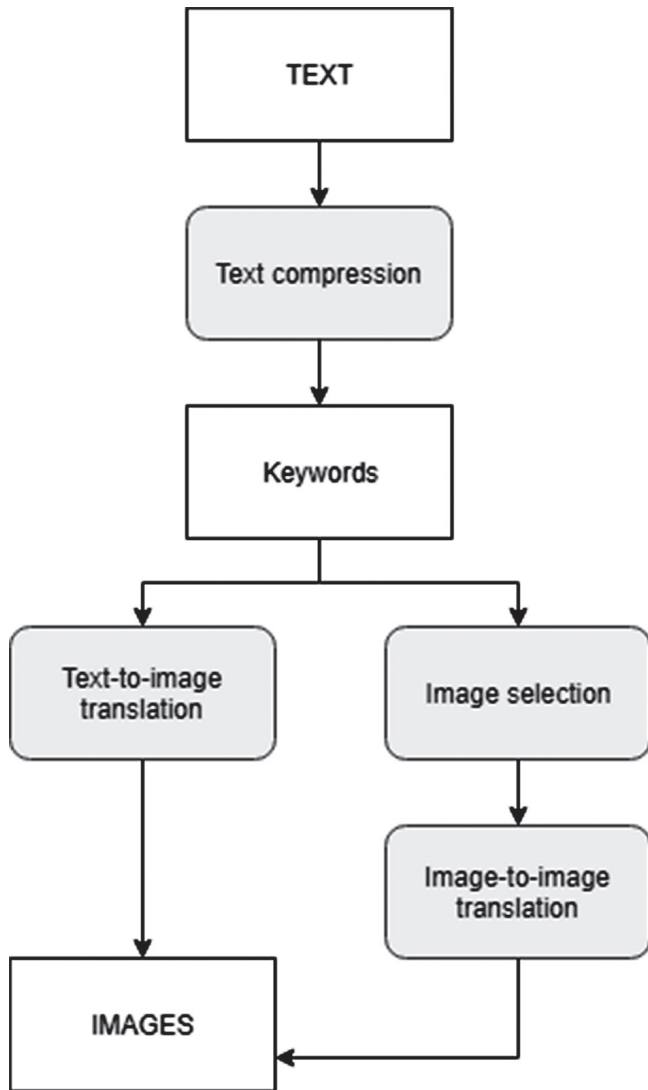


Fig. 1. Steps of image generation for the text

3. Conclusion

First, several ways to compress text (find keywords or create summary) were considered.

Examples are created for the story “The Last Leaf” by O. Henry and the novel “The Old Man and the Sea” by Ernest Hemingway. These texts were chosen because they are short and have a lot of descriptions to generate illustrations. The obtained text compressions are shown in Table 1 and Table 2.

Table 1

Results of Text Compression Methods (The Last Leaf)		
	The Last Leaf	Settings & Metrics
Rake	de world mit der foolishness dot poor leetle miss yohnsy elegant horseshow riding trousers useless woollen shoulder scarf pen-and-ink drawing sue found behrman smelling strongly red eyes plainly streaming especial mastiff-in-waiting lone ivy leaf clinging wide-open eyes staring small dutch window-panes three-story brick sue	12 keywords min characters = 3, max words = 5, min frequency = 1 <hr/> 5 (both other texts) and 2 common key phrases
Multi Rake	michael angelo’s mooses beard curling de world mit der foolishness dot poor leetle miss yohnsy elegant horseshow riding trousers useless woollen shoulder scarf red eyes plainly streaming sue found behrman smelling strongly lone ivy leaf clinging brick house twenty feet busy doctor invited sue ravager strode boldly feet trod slowly	12 keywords min characters = 3, max words = 7, min frequency = 1 <hr/> 5 (both other texts) and 3 common key phrases
Rake NLTK	dot poor leetle miss yohnsy miss yohnsy shall lie sick brush without getting near enough de world mit der foolishness art people soon came prowling brick house twenty feet away useless woollen shoulder scarf elegant horseshow riding trousers allow dot silly pusiness sue found behrman smelling strongly last one said johnsy brick wall one ivy leaf	12 keywords <hr/> 5 (both other texts) and 1 common key phrases
Lex Rank	«It is the last one,» said Johnsy. And then they found a lantern, still lighted, and a ladder that had been dragged from its place, and some scattered brushes, and a palette with green and yellow colours mixed on it, and - look out the window, dear, at the last ivy leaf on the wall.	2 sentences
Luhn	You may bring a me a little broth now, and some milk with a little port in it, and - no; bring me a hand-mirror first, and then pack some pillows about me, and I will sit up and watch you cook.»	2 sentences
LSA	And then they found a lantern, still lighted, and a ladder that had been dragged from its place, and some scattered brushes, and a palette with green and yellow colours mixed on it, and - look out the window, dear, at the last ivy leaf on the wall.	2 sentences
Text Rank	In a little district west of Washington Square the streets have run crazy and broken themselves into small strips called «places.» «Johnsy, dear,» said Sue, bending over her, «will you promise me to keep your eyes closed, and not look out the window until I am done working?»	2 sentences

Table 2

Results of Text Compression Methods
(The Old Man and the Sea)

	The Old Man and the Sea	Settings & Metrics
Rake	portuguese man-of-war floating dose white tipped wide pectoral fins portuguese men-of-war long deadly purple filaments trailing purple pectoral fins set wide two-decker metal container projecting green sticks dip sharply ordinary pyramid-shaped teeth man-of-war bird thrusting all-swallowing jaws high dorsal fin knifing great erect tail slicing	12 keywords min characters = 3, max words = 5, min frequency = 1 7 common key phrases
Multi Rake	white tipped wide pectoral fins long deadly purple filaments trailing purple pectoral fins set wide portuguese man-of-war floating dose high dorsal fin knifing projecting green sticks dip sharply great erect tail slicing he'll weigh ten pounds small delicate dark terns great long white spine shark's yellow cat-like eyes small tuna's shivering pull	12 keywords min characters = 3, max words = 7, min frequency = 1 7 common key phrases
Rake NLTK	good luck old man good luck know others better que va beg keep warm old man know sleep well old man	12 keywords 0 common key phrases
Rake NLTK	sorry qua va get four fresh ones one white cumulus built like friendly piles old man saw flying fish spurt old man said happily head let pedrico chop hands well old man day ends let us hope	
Lex Rank	He cannot know that it is only one man against him, nor that it is an old man. But there was not much of it.	2 sentences
Luhn	After that he began to dream of the long yellow beach and he saw the first of the lions come down onto it in the early dark and then the other lions came and he rested his chin on the wood of the bows where the ship lay anchored with the evening off-shore breeze and he waited to see if there would be more lions and he was happy. He took all his pain and what was left of his strength and his long gone pride and he put it against the fish's agony and the fish came over onto his side and swam gently on his side, his bill almost touching the planking of the skiff and started to pass the boat, long, deep, wide, silver and barred with purple and interminable in the water.	2 sentences

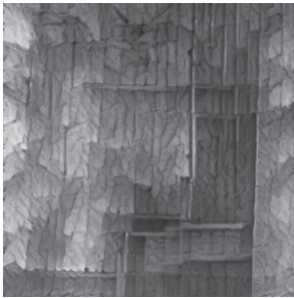

	The Old Man and the Sea	Settings & Metrics
LSA	How would you like to see me bring one in that dressed out over a thousand pounds?» «I'll get the cast net and go for sardines. No matter what passes I must gut the dolphin so he does not spoil and eat some of him to be strong.	2 sentences
Text Rank	He saw the phosphorescence of the Gulf weed in the water as he rowed over the part of the ocean that the fishermen called the great well because there was a sudden deep of seven hundred fathoms where all sorts of fish congregated because of the swirl the current made against the steep walls of the floor of the ocean. After that he began to dream of the long yellow beach and he saw the first of the lions come down onto it in the early dark and then the other lions came and he rested his chin on the wood of the bows where the ship lay anchored with the evening off-shore breeze and he waited to see if there would be more lions and he was happy.	2 sentences

Second, AttnGAN was used to generate images from resulted keywords.

Examples were created based on results from the previous step, so the same texts were used. The obtained images are shown in Table 3 and Table 4.

Table 3

Results of Image Generation (The Last Leaf)

	The Last Leaf
brick wall one ivy leaf	
lone ivy leaf clinging	
In a little district west of Washington Square the streets have run crazy and broken themselves into small strips called «places.»	

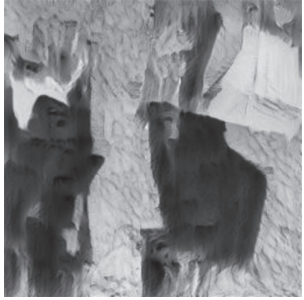
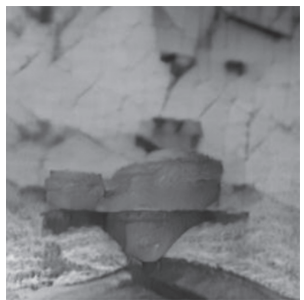



<p>And then they found a lantern, still lighted, and a ladder that had been dragged from its place, and some scattered brushes, and a palette with green and yellow colours mixed on it, and - look out the window, dear, at the last ivy leaf on the wall.</p>	
---	---

Table 4

Results of Image Generation (The Old Man and the Sea)

The Old Man and the Sea	
small tuna's shivering pull	
purple pectoral fins set wide	
long yellow beach	
<p>After that he began to dream of the long yellow beach and he saw the first of the lions come down onto it in the early dark and then the other lions came and he rested his chin on the wood of the bows where the ship lay anchored with the evening off-shore breeze and he waited to see if there would be more lions and he was happy.</p>	

As part of further researches, the freedom of GAN image generation can be combined with an easy and exact selection of pictures. If the GAN receives a specific template for creating an image, the resulting images will be more

logical and realistic. It is necessary to think where this template can be obtained and how it can be used in the GAN.

GAN freedom should be limited. The ability to generate any image complicates the training of the network and its further use. Perhaps for each topic (type of image) networks can be trained separately with some specific parameters and additional restrictions or conditions.

Also, existing images can be translated using other images, as the StyleGAN mentioned in paragraph 1 does, for example.

A combination of a text-to-image translation and an image-to-image translation can be a way to provide better results of an image from text generation.

References

- [1] Karras T., Laine S., Aila T. A Style-Based Generator Architecture for Generative Adversarial Networks // IEEE Conference on Computer Vision and Pattern Recognition (CVPR) – 2019. – P. 4401-4410.
- [2] Park T., Liu M.-Y., Wang T.-C., Zhu J.-Y. Semantic Image Synthesis with Spatially-Adaptive Normalization // IEEE Conference on Computer Vision and Pattern Recognition (CVPR) – 2019. – P. 2337-2346.
- [3] Isola P., Zhu J.-Y., Zhou T., Efros A. A. Image-to-Image Translation with Conditional Adversarial Networks // IEEE Conference on Computer Vision and Pattern Recognition (CVPR) – 2017. – P. 1125-1134.
- [4] Zhu J.-Y., Park T., Isola P., Efros A. A. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks // IEEE International Conference on Computer Vision (ICCV) – 2017. – P. 2223-2232.
- [5] Gatys L. A., Ecker A. S., Bethge M. A Neural Algorithm of Artistic Style // arXiv e-prints, arXiv:1508.06576v2 – 2015. – P. 1.
- [6] Brock A., Donahue J., Simonyan K. Large Scale GAN Training for High Fidelity Natural Image Synthesis // arXiv e-prints, arXiv:1809.11096v2 – 2019. – P. 8.
- [7] Xu T., Zhang P., Huang Q., Zhang H., Gan Z., Huang X., He X. AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks // IEEE Conference on Computer Vision and Pattern Recognition (CVPR) – 2018. – P. 1316-1324.
- [8] Shafkat I. Intuitively Understanding Variational Autoencoders // Medium – 2018. – URL: <https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf>
- [9] Shaham T. R., Dekel T., Michaeli T. SinGAN: Learning a Generative Model from a Single Natural Image // IEEE International Conference on Computer Vision (ICCV) – 2019. – P. 4570-4580.
- [10] Rose, S., Engel, D., Cramer, N., & Cowley, W. Automatic Keyword Extraction from Individual Documents // M. W. Berry & J. Kogan (Eds.), Text Mining: Theory and Applications – John Wiley & Sons, Hoboken, 2010. – P. 1-20.
- [11] Pranay M., Aman G., Aayush Y. Text Summarization in Python: Extractive vs. Abstractive techniques revisited // Rare Technologies. – 2017. – URL: <https://rare-technologies.com/text-summarization-in-python-extractive-vs-abstractive-techniques-revisited/>
- [12] Naskar A. Extract Custom Keywords using NLTK POS tagger in python // ThinkInfi. – 2018. – URL: <https://www.thinkinfi.com/2018/10/extract-custom-entity-using-nltk-pos.html>
- [13] Ma L., Jia X., Sun Q., Schiele B., Tuytelaars T., Gool L. V. Pose Guided Person Image Generation // IEEE Conference on Computer Vision and Pattern Recognition (CVPR) – 2019. – P. 2337-2346.
- [14] Bondarenko M., Konoplyanko Z., Chetverikov G. Analiz problemy sozdaniya novich technicheskich sredstv dlya realizataii lingvisticheskogo interfeisan // Proc.of the 10th International Conference KDS–2003, Varna, Bulgaria, – 2003. – P. 3–15.
- [15] Bondarenko M.F., Konoplyanko Z..D., Chetverikov G.G. Theory fundamentals of multipli-valued structures and coding in artificial intelligence systems.iKharkiv: Factor-druk. 2003.– 336 p.

The article was delivered to your editorial staff on the 20.11.2019

ПРАВИЛА оформлення рукописів для авторів науково-технічного журналу «БІОНІКА ІНТЕЛЕКТУ»

Науково-технічний журнал «Біоніка інтелекту» приймає до друку написані спеціально для нього оригінальні рукописи, які раніше ніде не друкувались. Структура рукопису повинна бути такою: індекс УДК, відомості про авторів, заголовок, анотації (на трьох мовах), ключові слова, вступ, основний текст статті, висновки, список використаної літератури, резюме.

Відповідно до Постанови ВАК України від 15.01.2003 №7-05/1 (Бюлетень ВАК, №1, 2003, с. 2), стаття повинна мати такі необхідні елементи: постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями; аналіз останніх досліджень і публікацій і виділення не вирішених раніше частин загальної проблеми в даній області; формулювання цілей та завдань дослідження; виклад основного матеріалу досліджень з повним обґрунтуванням отриманих наукових результатів; висновки з даного дослідження та перспективи подальших досліджень у даному напрямку.

Статті мають бути виконані в редакторі Microsoft Word. Формат сторінки – А4 (210×297 мм), поля: верхнє – 25 мм, нижнє – 20 мм, лівє, правє – 17 мм. Кількість колонок – 2, з інтервалом між ними 5 мм, основний шрифт Times New Roman, кегль основного тексту – 10 пунктів, міжрядковий інтервал – множник (1,1), абзацний відступ – 6 мм. Обсяг рукопису – від 6 до 12 сторінок (мови: українська, англійська, російська та мовою оригінала).

УДК друкується з першого рядка, без відступів, вирівнювання по лівому краю.

ПІБ автора (-ів), назва статті, назва та адреса учбового закладу необхідно надати повністю російською, українською та англійською мовами.

Назва статті друкується прописними літерами; шрифт прямий, напівжирний, кегль 12.

Назви розділів нумерують арабськими цифрами, виділяють жирним шрифтом. Відступи для назви статті, ініціалів та прізвищ авторів, відомостей про авторів, назв розділів, вступу та висновків, списку літератури: зверху – 6 пт, знизу – 3 пт.

Анотації (мовою статті, абзац 6–12 рядків, кегль 9) розміщують на початку статті, в ній має бути розміщена інформація про очікувані результати описаних досліджень (на трьох мовах).

Ключові слова (4–10 слів з тексту статті, які з точки зору інформаційного пошуку несуть змістовне навантаження) наводять мовою рукопису, через кому в називному відмінку, кегль 9.

Рисунки та таблиці (чорно-білі, контрастні) розміщуються у тексті після першого посилання у вигляді окремих об'єктів і нумерують арабськими цифрами наскрізною нумерацією за наявності більше ніж одного об'єкта. Невеликі схеми, що складаються з 3–4 елементів виконують, використовуючи вставку об'єкта Рисунок Microsoft Word. Більш складні виконують у графічних редакторах у вигляді чорно-білих графічних файлів форматів .tif, .jpg, .wmf, .cdr із розділенням 300 dpi. Рисунки мають міститися у текстовому файлі й обов'язково подаватися

окремими файлами з відповідними назвами (наприклад, рис1.jpg).

Усі елементи рисунка, включаючи написи, повинні бути згруповані. Усі написи в рисунках і таблицях мають бути виконані шрифтом Times New Roman, кегль у рисунках – 10, у таблицях – 9.

Рисунок повинен мати центрований підпис (поза рисунком), шрифт 9, відступи зверху і знизу по 6 пт. Ширина рисунка має відповідати ширині колонки (або ширині сторінки).

Формули, символи, змінні повинні бути набрані в редакторі формул **MathType**. Формули розміщують посередині рядка й нумерують за наявності посилань на них у рукописі. Шрифт – Times New Roman. Висота змінної – 10 пунктів, великих і малих індексів – 8 пт, основний математичний символ – 12 (10) пт. Змінні, позначені латинськими літерами, набирають курсивом, грецькі літери, скорочення російських слів і цифри – прямим написанням. Змінні, які є в тексті, також набирають у редакторі формул.

Список літератури вміщує опубліковані джерела, на які є посилання в тексті, укладені у квадратні дужки, друкують без абзацного відступу, кегль 9 пт, відступ зверху – 6 пт.

Після списку літератури з відступом зверху 6 пт зазначають *дату подання статті до редколегії*. Число та місяць задають двозначними числами через крапку. Розмір шрифта – 9 пт, курсив, вирівнювання по правому краю.

Резюме (Times New Roman, кегль – 10 пунктів,) подають англійською мовою: обсяг резюме до 2000 знаків (бажаний переклад). *Структура резюме: Background, Materials and methods, Results, Conclusion.*

Разом із рукописом (на аркушах білого паперу формату А4 щільністю 80-90 г/м², надрукований на лазерному принтері) необхідно подати такі документи:

1. Заяву, яку повинні підписати всі автори.
2. Акт експертизи про можливість опублікування матеріалів у відкритому друці (якщо потрібно).
3. Рецензію, підписану доктором чи кандидатом наук.
4. Відомості про авторів.
5. Електронний варіант рукопису, резюме та відомостей про авторів на e-mail: bionics@nure.ua.
6. Зробити оплату публікації.

Необхідно також зазначити один з наступних тематичних розділів, якому відповідає рукопис:

1. Теоретичні основи інформатики та кібернетики. Теорія інтелекту.
2. Математичне моделювання. Системний аналіз. Прийняття рішень.
3. Інтелектуальна обробка інформації. Розпізнавання образів.
4. Інформаційні технології та програмно-технічні ком-плекси.
5. Структурна, прикладна та математична лінгвістика.
6. Дискусійні повідомлення.

СОДЕРЖАНИЕ

ОБЪЕКТНОЕ МОДЕЛИРОВАНИЕ. МЕТОДОЛОГИЯ ИНФОРМАТИЗАЦИИ

<i>Бодянский С.В., Шафроненко А.Ю., Климова И.М.</i> Онлайн достовірна нечітка кластеризація даних з використанням функції належності спеціального типу	3
<i>Руденко О.Г., Безсонов О.О., Сердюк Н.М., Олійник К.О., Романюк О.С.</i> Робастна ідентифікація об'єктів за наявністю негаусівських завад	7
<i>Удовенко С.Г., Келембет Д.В., Затхей В.А.</i> Керування нелінійними технологічними об'єктами з використанням нечітких РДС-регуляторів	13

ОНТОЛОГИЧЕСКИЕ СИСТЕМЫ. КОМПАРАТОРНАЯ ИДЕНТИФИКАЦИЯ. МЕТОДЫ КОМБИНАТОРНОЙ ОПТИМИЗАЦИИ

<i>Петров К.Э., Чайникова Т.С., Кобзев И.В., Демчук В.Г.</i> Компараторная идентификация модели многофакторного оценивания альтернатив с использованием метода Бэггинга	21
<i>Гребенник И.В., Чорна О.С.</i> Циклічні перестановки в методах комбінаторної оптимізації на основі циклічних трансферів	28
<i>Гасько Р.Т., Шаховська Н.Б.</i> Навчальна робототехнічна платформа для FOG\EDGE\AIoT	34
<i>Чала Л.Е., Карпенко Д.І., Берестовий О.О.</i> Розробка інтерактивного музейного додатку з використанням технології доповненої реальності	39

МАТЕМАТИЧЕСКИЕ МОДЕЛИ. СИСТЕМНЫЙ АНАЛИЗ. ПРИНЯТИЕ РЕШЕНИЙ

<i>Левыкин И.В.</i> Обобщенный алгоритм и программная платформа получения решения по приоритетам запуска бизнес-процессов.....	47
<i>Мазурова О.О., Гордієнко Т.О.</i> Дослідження генетичних алгоритмів для пошуку оптимальних шляхів в системі проведення метромарафонів.....	53
<i>Функендорф А.О., Бортнікова В.О., Груздо І.В., Черепанова Ю.Ю.</i> Математична модель процесу отримання нероз'ємних з'єднань у конструкціях МЕМС методом ультразвукового зварювання.....	59
<i>Rydorenko D., Turuta O.</i> A survey of methods of text-to-image translation	64

ПРАВИЛА оформлення рукописів для авторів науково-технічного журналу «БІОНИКА ІНТЕЛЕКТУ»	69
--	-----------

Наукове видання

БІОНІКА ІНТЕЛЕКТУ
інформація, мова, інтелект

Науково-технічний журнал

№ 2 (93)

2019

Головний редактор — *Г. Г. Четвериков*
Відповідальний редактор — *І. Д. Вечірська*

Комп'ютерна верстка — *О. Б. Ісаєва*

Рекомендовано Вченою Радою
Харківського національного університету радіоелектроніки
(протокол № 11/11 от 26.12.2019)

Адреса редакції:

Україна, 61166, Харків-166, просп. Науки, 14,
Харківський національний університет радіоелектроніки, к. 127
тел. 702-14-77, факс 702-10-13,
e-mail: bionics@nure.ua

Підписано до друку 27.12.2019. Формат $60 \times 84 \frac{1}{8}$. Друк ризографічний.
Папір офсетний. Гарнітура Newton. Умов. друк. арк. 15,4. Обл.-вид. арк. 15,0.
Тираж 100 прим.

Віддруковано в редакційно-видавничому відділі ХНУРЕ
61166, Харків, просп. Науки, 14.