

## ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

# ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

КАФЕДРА ЕЛЕКТРОННИХ ОБЧИСЛЮВАЛЬНИХ МАШИН

## КВАЛІФІКАЦІЙНА РОБОТА

# «Метод енергозбереження в безпроводних сенсорних мережах»

Студентка гр. СПМ-23-5

Гармаш В.С.

Керівник

проф. Міхаль О.П.

Харків 2025

## *Мета та завдання кваліфікаційної роботи*

**Мета кваліфікаційної роботи:** розробка та дослідження методу енергозбереження у безпроводних сенсорних мережах на основі використання алгоритмів машинного навчання, зокрема штучних нейронних мереж типу карт Кохонена та їх модифікацій, для забезпечення максимально ефективного використання енергетичних ресурсів вузлів, оптимізації процесів передачі та обробки даних, а також збільшення періоду автономної роботи та надійності функціонування сенсорних мереж у змінних умовах експлуатації.

**Завдання:**

- ❖ провести аналіз сучасного стану досліджень у сфері енергозбереження в безпроводних сенсорних мережах та виявити основні фактори, що впливають на енергетичну ефективність вузлів;
- ❖ розглянути можливості використання методів машинного навчання для вирішення задач оптимізації енергоспоживання в WSN, з акцентом на нейронні мережі самоорганізації типу карт Кохонена;
- ❖ дослідити модифікації процесу навчання карт Кохонена, орієнтовані на підвищення енергоефективності при обмежених ресурсах сенсорних вузлів;
- ❖ розробити метод енергозбереження, який базується на динамічному кластеруванні вузлів із використанням модифікованої карти Кохонена та адаптивному управлінні режимами їх роботи;
- ❖ реалізувати запропонований метод у вигляді програмної моделі в середовищі Google Colab із використанням Python, бібліотеки MiniSom та засобів візуалізації;
- ❖ провести експериментальне дослідження ефективності розробленого методу на штучно згенерованих даних з оцінкою показників енергоспоживання, розподілу навантаження та тривалості автономної роботи мережі.

**Об'єкт дослідження:** процеси енергоспоживання та функціонування безпроводних сенсорних мереж.

# Особливості функціонування та сфери застосування безпроводних сенсорних мереж

| ОСОБЛИВОСТІ ФУНКЦІОНУВАННЯ  | СФЕРИ ЗАСТОСУВАННЯ   |
|---|--|
| <p><b>Самоорганізація</b></p> <p>Автоматичне встановлення зв'язків між вузлами без централізованого управління та ручного налаштування топології мережі</p> | <p><b>Промислове виробництво</b></p> <p>Моніторинг технологічних процесів, контроль якості продукції, прогнозування вітросо обладнання, автоматизація виробничих ліній</p> |
| <p><b>Енергоефективність</b></p> <p>Оптимізація алгоритмів для мінімізації споживання енергії та максимізації терміну автономної роботи системи</p>         | <p><b>Сільське господарство</b></p> <p>Точне землеробство, моніторинг вологості ґрунту, автоматизовані іригаційні системи, контроль мікроклімату теплиць</p>               |
| <p><b>Масштабованість</b></p> <p>Здатність систем ефективно функціонувати при збільшенні кількості сенсорних вузлів без деградації продуктивності</p>       | <p><b>Медицина та охорона здоров'я</b></p> <p>Біомедичний моніторинг пацієнтів, телемедицина, системи раннього попередження, моніторинг життєвих показників</p>            |
| <p><b>Відмовостійкість</b></p> <p>Забезпечення безперервності роботи мережі при виході з ладу окремих вузлів за рахунок резервування шляхів</p>             | <p><b>Екологічний моніторинг</b></p> <p>Контроль якості повітря та води, моніторинг радіаційного фону, спостереження за кліматичними змінами</p>                           |
| <p><b>Адаптивність</b></p> <p>Динамічне пристосування до змін у навколишньому середовищі, топології мережі та умов передачі даних</p>                       | <p><b>Розумні міста та будівлі</b></p> <p>Автоматизація інженерних систем, управління освітленням, системи безпеки, оптимізація енергоспоживання</p>                       |

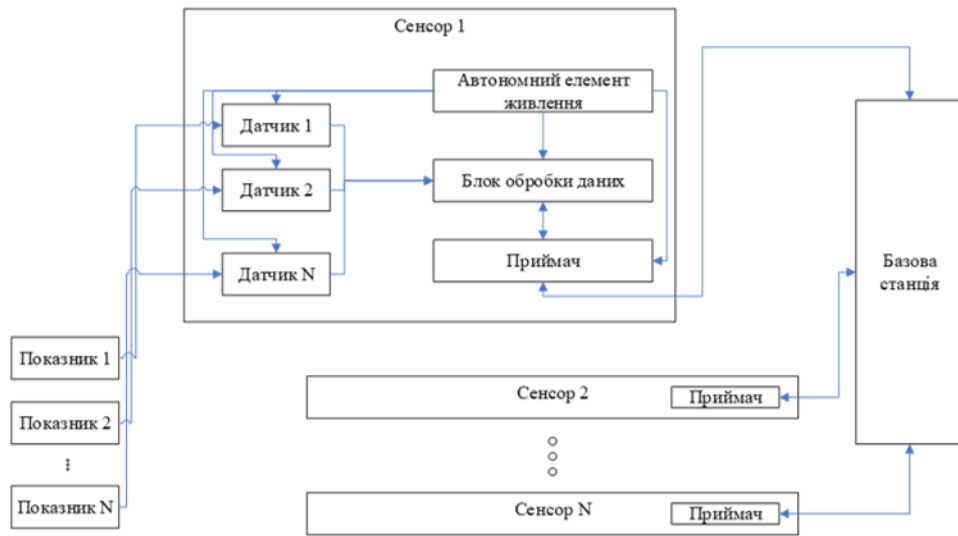
3

# Проблеми енергоживлення в безпроводних сенсорних мережах

|  |   |
|--|---|
| <p><b>ОБМЕЖЕНА ЄМНІСТЬ БАТАРЕЇ</b> <span style="color: red;">Високий вплив</span></p> <p>Сенсорні вузли живляться від батарей з обмеженою ємністю. Заміна батарей у віддалених або недоступних місцях може бути складною або дорогою, що створює фундаментальне обмеження для довготривалої роботи мережі.</p> <p><b>Основні фактори впливу:</b></p> <ul style="list-style-type: none"> <li>• Типова ємність літійових батарей: 1000-3000 мАг</li> <li>• Час роботи без заряду: 6-24 місяці</li> <li>• Вартість заміни в недоступних місцях</li> <li>• Державні ємності і часом та температурою</li> </ul> | <p><b>ЕНЕРГОСПОЖИВАННЯ ПРИ ПЕРЕДАЧІ</b> <span style="color: red;">Високий вплив</span></p> <p>Передача даних споживає найбільше енергії в сенсорних вузлах. Дані відстані та передачі обумовлюють енергетичні витрати на зв'язок, що може швидко виснажити батарею вузла.</p> <p><b>Енергетичні характеристики:</b></p> <ul style="list-style-type: none"> <li>• Передача: 50-100 мВт енергетичності</li> <li>• Прийм.: 30-60 мВт енергетичності</li> <li>• Залежність від квадрата відстані</li> <li>• Втрата через безпроводне поширення</li> </ul> |
| <p><b>НЕЕФЕКТИВНЕ МАРШРУТИЗУВАННЯ</b> <span style="color: orange;">Середній вплив</span></p> <p>Погово спростковані алгоритми маршрутизування можуть призвести до надлишкового виснаження енергії окремих вузлів через нерівномірний розподіл навантаження в мережі.</p> <p><b>Проблеми традиційних підходів:</b></p> <ul style="list-style-type: none"> <li>• Перевантаження вузла біля базової станції</li> <li>• Відсутність балансування навантаження</li> <li>• Неправильне використання енергії</li> <li>• Статичні маршрути без адаптивності</li> </ul>   | <p><b>РЕЖИМ ОЧІКУВАННЯ</b> <span style="color: orange;">Середній вплив</span></p> <p>Навіть в режимі очікування вузли споживають енергію для підтримки готовності до прийому сигналу. Неefективне управління режимами роботи призводить до надлишкового витрат енергії.</p> <p><b>Енергоспоживання в режимі очікування:</b></p> <ul style="list-style-type: none"> <li>• Активний режим: 100% споживання</li> <li>• Режим очікування: 10-30% споживання</li> <li>• Глибокий сон: 1-5% споживання</li> <li>• Час пробудження: 1-10 мс</li> </ul>       |
| <p><b>ВПЛИВ НАВКОЛИШНЬОГО СЕРЕДОВИЩА</b> <span style="color: orange;">Середній вплив</span></p> <p>Екстремальні температури можуть знизити ефективність батарей та збільшити енергоспоживання компонентів. Температурні коливання впливають на всі аспекти роботи сенсорних вузлів.</p> <p><b>Температурні ефекти:</b></p> <ul style="list-style-type: none"> <li>• Зниження ємності при низькій температурі</li> <li>• Збільшення опору при нагріванні</li> <li>• Температурна компенсація сенсорів</li> <li>• Термічний шум у радіокомунікації</li> </ul>  | <p><b>ОБРОБКА ДАНИХ</b> <span style="color: green;">Низький вплив</span></p> <p>Складні обчислення на вузлах для обробки сенсорних даних також споживають певну кількість енергії, особливо при використанні алгоритмів машинного навчання або криптографії.</p> <p><b>Обчислювальні витрати:</b></p> <ul style="list-style-type: none"> <li>• Фільтрація та агломерація сигналів</li> <li>• Стиснення даних перед передачею</li> <li>• Криптографічне шифрування</li> <li>• Локальна агрегація даних</li> </ul>                                      |

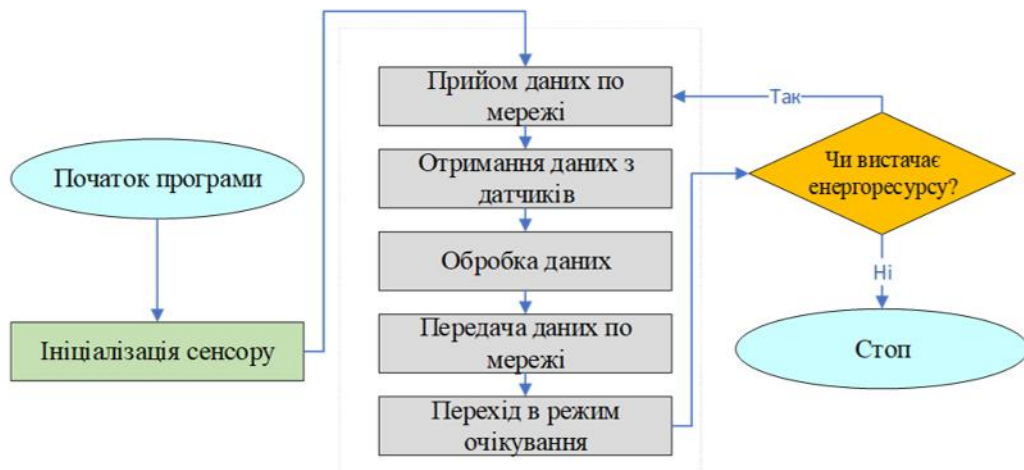
4

## Аналіз енергоживлення вузла бездротової сенсорної мережі



5

## Алгоритм роботи сенсора WSN



6

# Алгоритми кластеризації, класифікації та прогнозування

### АЛГОРИТМИ КЛАСТЕРИЗАЦІЇ

**LEACH (Low-Energy Adaptive Clustering Hierarchy)**  
Розподілений протокол кластеризації з динамічною системою кластерів для покращення енергозбереження.

**HEED (Hybrid Energy-Efficient Distributed)**  
Спрощений алгоритм, що враховує різноманітні потреби та вартість комунікаційних ресурсів.

**К-середнє для WSN**  
Адаптивний алгоритм в середовищі з динамічною топологією розподіленою на енергетично обмежені вузли.

**PEGASIS (Pervasive Efficient Gathering)**  
Покращений протокол маршрутизації та збору даних, що враховує енергетичні обмеження.

### АЛГОРИТМИ КЛАСИФІКАЦІЇ

**Naive Bayes для WSN**  
Спростований алгоритм класифікації з використанням байєвських мереж для розподілення даних.

**SVM (Support Vector Machine)**  
Універсальний алгоритм для класифікації даних з високим рівнем точності.

**Decision Trees**  
Деревні структури з оптимальними для класифікації характеристиками.

**Neural Networks (глибокі)**  
Мережі з обчислювальною здатністю вибирати найкращі для класифікації ознаки даних.

### АЛГОРИТМИ ПРОГНОЗУВАННЯ

**ARIMA (AutoRegressive Integrated Moving Average)**  
Статистична модель для прогнозування часових рядів на основі минулих значень.

**Kalman Filter**  
Оптимальний фільтр для оцінки стану динамічної системи з урахуванням шуму та невизначеності.

**Exponential Smoothing**  
Метод прогнозування, що використовує експоненціальне згладжування для акценту на більш актуальних даних.

**LSTM (глибокий)**  
Адаптивний алгоритм прогнозування на основі глибоких нейронних мереж для прогнозування часових рядів.

| ПОРІВНЯЛЬНИЙ АНАЛІЗ ЕФЕКТИВНОСТІ |               |                  |                          |             |                 |             |                        |
|----------------------------------|---------------|------------------|--------------------------|-------------|-----------------|-------------|------------------------|
| Алгоритм                         | Категорія     | Енергозбереження | Обчислювальна складність | Точність    | Масштабованість | Затримка    | Продуктивність для WSN |
| LEACH                            | Кластеризація | Висока           | O(n)                     | Середня     | Висока          | Низька      | Висока                 |
| HEED                             | Кластеризація | Дуже висока      | O(n log n)               | Висока      | Висока          | Середня     | Висока                 |
| К-середнє                        | Кластеризація | Середня          | O(n^2)                   | Висока      | Середня         | Висока      | Добра                  |
| Naive Bayes                      | Класифікація  | Дуже висока      | O(n)                     | Середня     | Висока          | Дуже висока | Висока                 |
| SVM                              | Класифікація  | Висока           | O(n^2)                   | Дуже висока | Низька          | Висока      | Залежна                |
| Decision Trees                   | Класифікація  | Висока           | O(n log n)               | Висока      | Середня         | Низька      | Добра                  |
| ARIMA                            | Прогнозування | Середня          | O(n^2)                   | Висока      | Середня         | Середня     | Добра                  |
| Kalman Filter                    | Прогнозування | Висока           | O(n)                     | Дуже висока | Висока          | Дуже висока | Висока                 |
| Exp. Smoothing                   | Прогнозування | Дуже висока      | O(1)                     | Середня     | Дуже висока     | Дуже висока | Висока                 |

7

# Застосування карт Кохонена для оптимізації WSN

### СТРУКТУРА SOM

X (координата)    Y (координата)    E (енергія)    D (діаметр)    C (тип вузла)

↓ Ваги зв'язків ↓

|     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|
| N11 | N12 | N13 | N14 | N15 | N16 |
| N21 | N22 | N23 | N24 | N25 | N26 |
| N31 | N32 | N33 | N34 | N35 | N36 |
| N41 | N42 | N43 | N44 | N45 | N46 |
| N51 | N52 | N53 | N54 | N55 | N56 |
| N61 | N62 | N63 | N64 | N65 | N66 |

6×6 SOM сітка  
BMU - Best Matching Unit (переможець)  
Кольори показують кластери

Важкий вектор вхідних даних

### ЗАСТОСУВАННЯ У WSN

**Рівні енергії вузлів**

- Високоенергетичні (>70%)
- Середньоенергетичні (30-70%)
- Низькоенергетичні (<30%)
- Головні вузли кластерів

**Кластери SOM**

- Кластер 1 (високоенергетичний)
- Кластер 2 (обчислювальний)
- Кластер 3 (периферійний)

8

# Метод динамічної кластеризації з адаптованим алгоритмом навчання карти Кохонена для енергооптимізації WSN

## КОНЦЕПЦІЯ МЕТОДУ

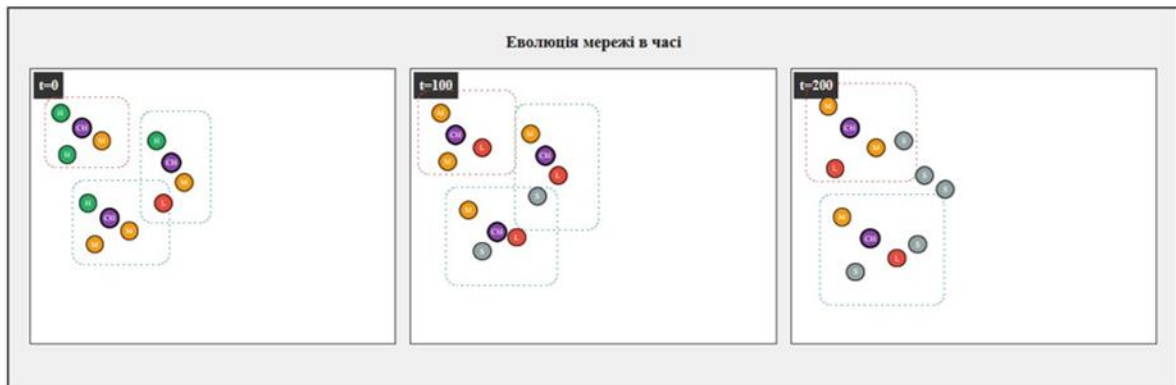
Побудова енергозалежної карти Кохонена, яка оновлюється в онлайн-режимі та приймає рішення щодо:

- Формування кластерів вузлів
- Вибору енергооптимальних кластерних центрів
- Визначення ефективних маршрутів передачі
- Адаптації режимів роботи вузлів

### Ключові особливості:

- Динамічно адаптований радіус сусідства
- Енергозалежне оновлення ваг
- Інкрементальне онлайн-навчання
- Цілісне проєктування вузлів

## Еволюція мережі в часі



9

## Етапи методу

### Етапи реалізації методу

#### Первинне навчання SOM на основі стану вузлів

Система отримує входні дані з вузлів: рівень заряду, частота передач, кількість сусідів, індекс зв'язку.

#### Кластеризація вузлів і вибір центрів кластерів

Формується множина кластерів, у межах яких обираються головні вузли.

#### Адаптивна маршрутизація

Формується таблиця маршрутів, яка оновлюється з урахуванням зміни умов SOM

#### Періодичне перенавчання SOM

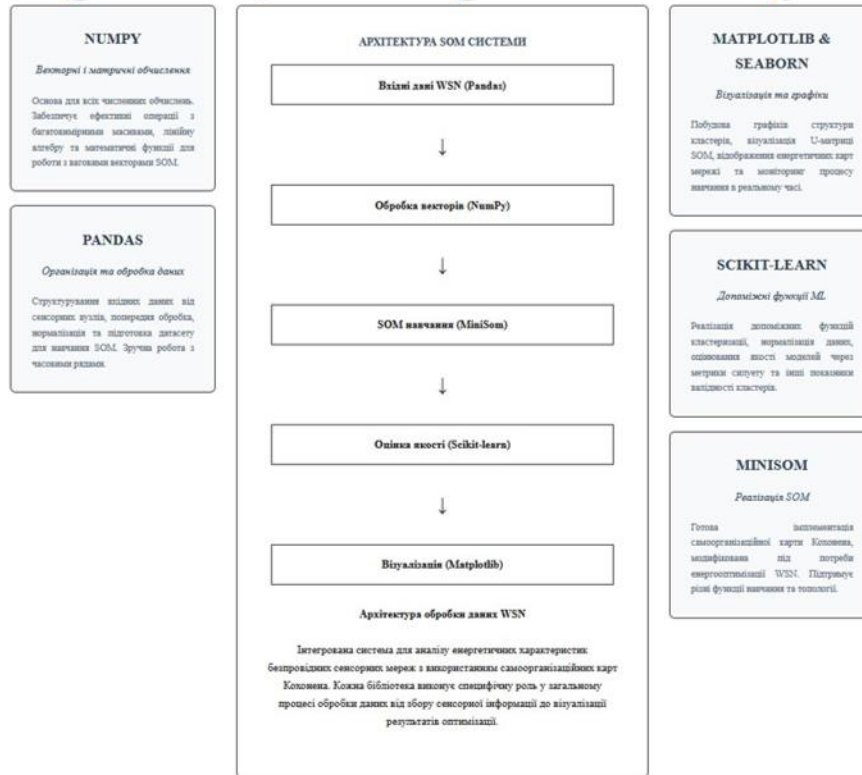
Перенавчання карти SOM з використанням інкрементального підходу.

#### Управління енергетичними режимами вузлів

Визначаються вузли, які можуть бути переведені в сплячий або напівактивний режим без втрати цілісності даних.

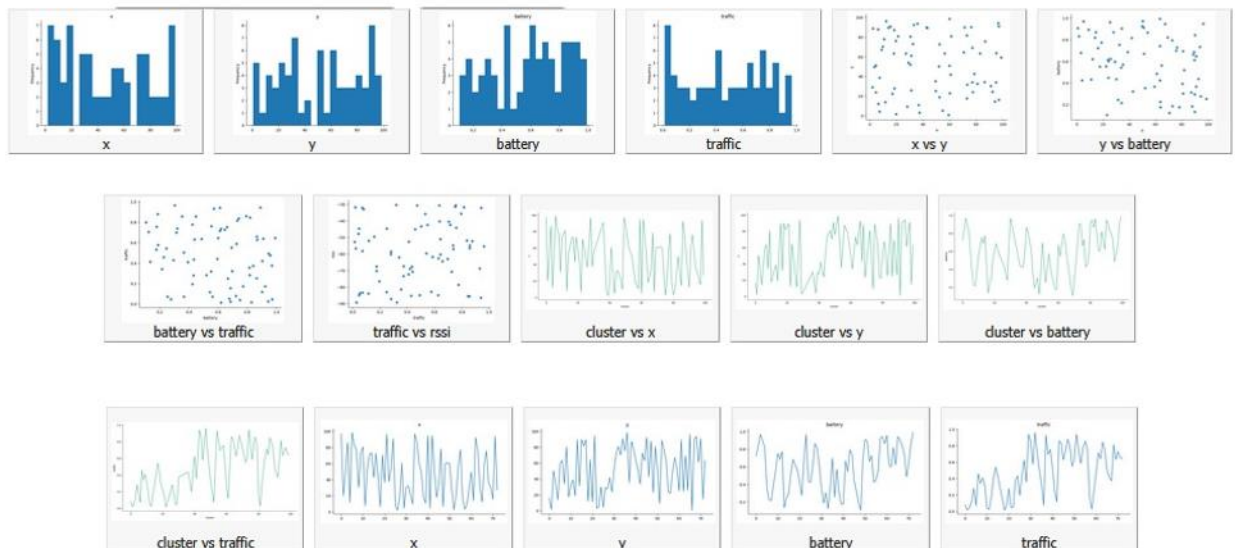
10

# Технологічний стек для реалізації самоорганізаційних карт Кохонена у WSN



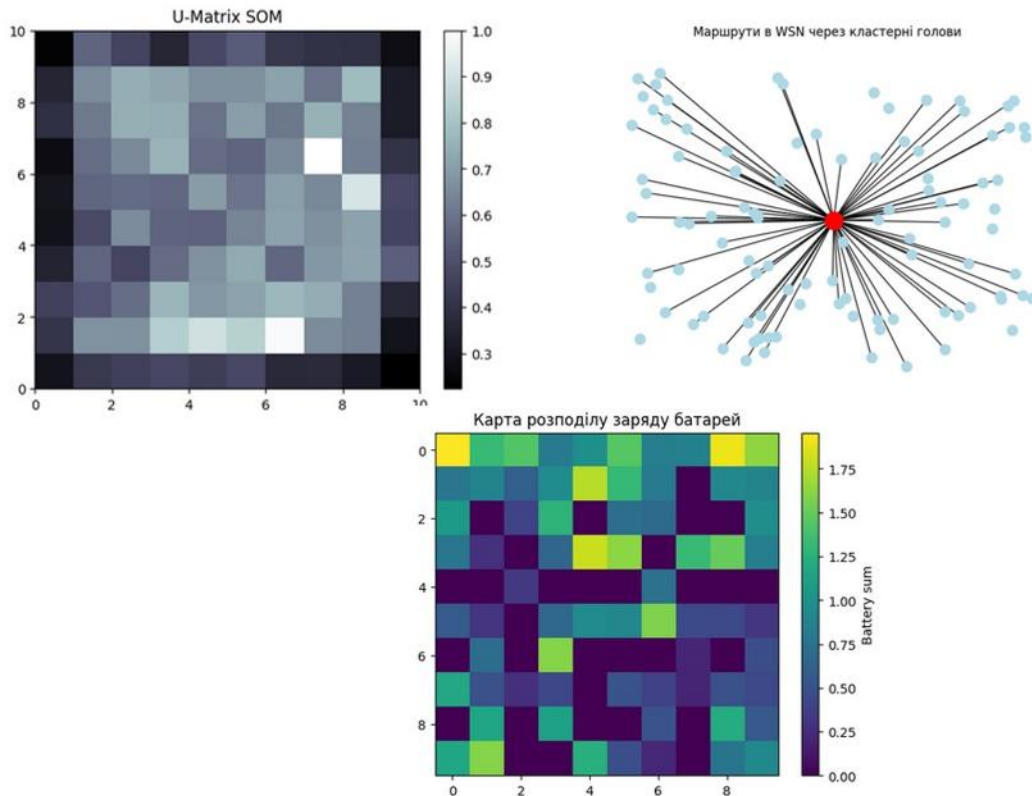
11

## Результати роботи



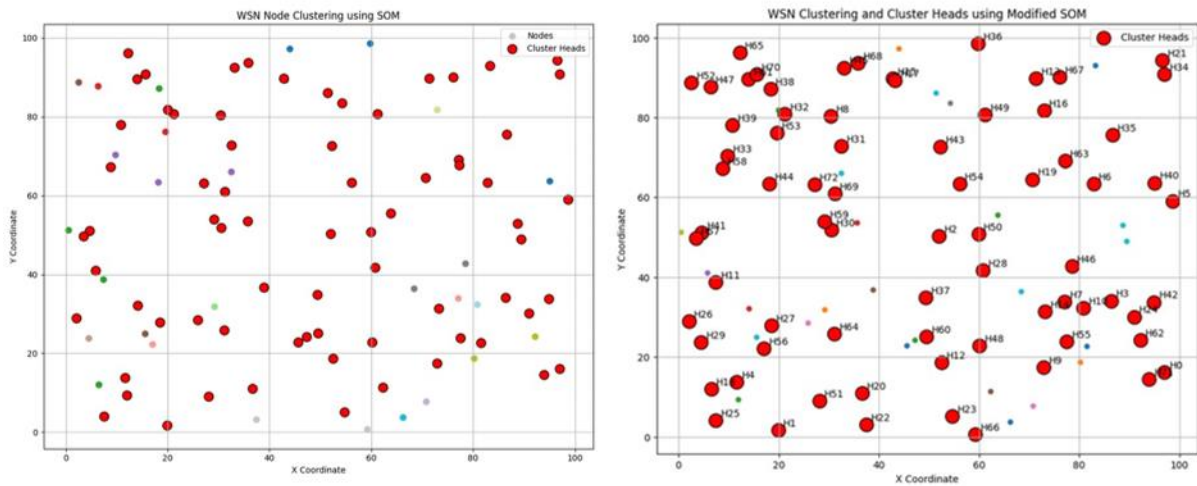
12

## Результати роботи



13

## Результати роботи



14

## Апробація результатів дослідження

### ENERGY-SAVING METHOD IN WIRELESS SENSOR NETWORKS

**Abstract. Relevance.** The importance of energy efficiency in Wireless Sensor Networks is steadily increasing due to the widespread adoption of these technologies across various sectors of modern society – from environmental monitoring and intelligent resource management systems to healthcare, industry, and military applications. WSNs are characterized by many autonomous nodes that are expected to operate for extended periods without battery replacement or recharging. This poses a significant challenge for researchers and developers to find effective energy-saving methods that can substantially extend the autonomous operating time of the nodes and ensure the stable and reliable functioning of the entire network. Of relevance is the development of energy-saving methods that leverage modern machine learning technologies, as traditional approaches to energy management no longer fully address the complex challenges posed by the scale and dynamics of modern sensor systems. Machine learning enables adaptive network operation management, considering changing external conditions and internal parameters, thus ensuring more rational energy resource utilization. In this context, Artificial Neural Networks – especially Kohonen neural maps and their modifications – play a particularly important role. Data clustering and classification using Kohonen maps allows efficient solutions to energy-saving tasks through optimized routing and load balancing among sensor nodes. A distinctive feature of Kohonen maps is their ability to self-organize and adapt, enabling the automatic creation of an optimal network structure based on minimizing energy consumption. As a result, such neural networks are actively used for clustering sensor nodes, forming node groups based on energy characteristics, as well as for predicting and detecting inefficient operational states of the nodes. Modifications of Kohonen maps, such as hierarchical or hybrid neural networks, significantly enhance accuracy and efficiency in complex and dynamic conditions. For instance, hierarchical Kohonen maps can be applied for multilevel clustering of the network, enabling the optimization of energy expenditures by creating structures with varying levels of node activity. Other modifications include the integration of Kohonen maps with other machine learning methods, such as recurrent and convolutional neural networks, to account for the spatiotemporal characteristics of data in sensor networks. **The object of research** is the processes of energy consumption and functioning of the Wireless Sensor Networks. **Purpose of the article** is the development and investigation of an energy-saving method in Wireless Sensor Networks based on the use of machine learning algorithms, particularly artificial neural networks of the Kohonen map type and their modifications, in order to ensure the most efficient utilization of node energy resources, optimize data transmission and processing processes, and increase the duration of autonomous operation and the reliability of sensor network functioning under variable operating conditions. **Research results.** An energy-saving method based on modified Kohonen maps has been developed. The proposed approach involves multilevel clustering of nodes considering their energy parameters, adaptive selection of transmission routes, regulation of node activity modes, and online retraining of the map. **Conclusion.** The energy consumption of nodes depends not only on the hardware configuration but also on the method of data exchange organization, the chosen network topology, transmission frequency, and environmental conditions. The issue of energy depletion in individual nodes is critical, as it can lead to network fragmentation or complete network failure. Therefore, there is a need for dynamic, intelligent management that considers both local and global characteristics of the network.

**Keywords:** wireless sensor network, energy consumption, Kohonen map, machine learning, clustering, adaptive routing, nodes.

#### Introduction

Modern wireless sensor networks are one of the key infrastructure components across various fields of human activity, such as environmental monitoring, industrial process control, healthcare, smart homes and cities, agriculture, and military applications. The widespread adoption of such networks is driven by their capability for continuous data collection, transmission, and analysis from many spatially distributed sensor nodes. At the same time, the growing number of sensor devices, the increasing volume of processed data, and the expansion of geographic deployment areas significantly raise the demand for energy autonomy of these systems.

One of the most critical challenges faced by wireless sensor networks is ensuring a prolonged operational period of the nodes without the need to replace power

sources is a complex and economically inefficient task. This is why energy efficiency becomes a crucial factor that directly impacts the viability and practical applicability of sensor networks.

The issue of node energy consumption is further complicated by the fact that a typical sensor device expends energy not only for data transmission but also for data reception, processing, and maintaining functionality in standby mode. Given the limited power supply resources, the need for a comprehensive approach to energy efficiency is evident. Therefore, the relevance of researching energy-saving methods lies in finding a balance between performance, reliability, and the duration of autonomous node operation.

This work presents a detailed analysis of modern methods for reducing energy consumption in wireless sensor networks. Particular attention is given to energy-efficient routing protocols, algorithms for managing

networks under real-world conditions.

The purpose of this work is the development and investigation of an energy-saving method in wireless sensor networks based on the use of machine learning algorithms, particularly artificial neural networks of the Kohonen map type and their modifications, to ensure the most efficient utilization of node energy resources, optimize data transmission and processing processes, and increase the duration of autonomous operation and the reliability of sensor network functioning under variable operating conditions.

#### Main part

The operation of sensor nodes in wireless sensor networks is determined by a combination of hardware, software, and environmental factors, each of which affects performance, stability, energy consumption, and the duration of autonomous operation. Key factors include: the type and sensitivity of the sensor (the choice of sensor depends on the type of physical quantity being measured); the higher the resolution and sensitivity, the more energy the sensor may consume; the power and architecture of the microcontroller (the processor's computing power determines the ability to locally process data before transmission); more powerful microcontrollers reduce the amount of data transmitted but may consume more energy themselves; the type and capacity of the power source (the battery or energy storage device limits the node's operation time); the type of power source and the possibility of energy harvesting are also important; energy consumption modes (the availability of sleep, wake-up, standby, and active communication modes are critical for optimizing energy usage); switching between modes must be controlled and optimized; the communication protocol (the selected protocol affects transmission range, speed, and energy cost); network topology and routing algorithms (a node may be an endpoint or a relay. Acting as a relay for others increases its energy consumption); uneven load distribution among nodes leads to early depletion of certain elements in the network; data collection and transmission frequency (frequent measurements and transmissions increase energy usage); optimizing this frequency is critical in designing energy-efficient algorithms; environmental conditions (temperature, humidity, interference levels, terrain, and other physical factors can affect sensor reliability and communication quality, causing additional energy expenditures); node software (optimized software, task schedulers, and adaptive algorithms can significantly reduce energy usage); local decision-making capability (nodes capable of analyzing collected data independently – e.g., classifying events or filtering noise – can reduce the number of transmitted

environmental conditions, battery degradation, or unexpected overloads of individual nodes. Therefore, the application of machine learning algorithms, which allow adaptive optimization of network operation considering variable parameters and accumulated experience, is gaining increasing relevance.

One of the key tasks in the context of energy saving is node clustering, i.e., grouping nodes into clusters based on such features as battery level, geographic location, traffic density, and load. Machine learning enables this clustering to be performed dynamically rather than statically, with the structure of the clusters being updated as the network state changes. For instance, k-means methods, hierarchical clustering, or algorithms based on self-organizing maps can automatically determine the optimal number and configuration of clusters, helping reduce energy losses from routing and prevent overload of individual nodes.

Another important area is activity prediction of nodes and intelligent scheduling of sleep/wake cycles. Using historical activity data, typical sensing patterns, or external parameters (e.g., daily temperature variations, movement, or noise), learning algorithms can predict when a node is likely to be heavily loaded and adjust its operating mode accordingly. This approach maintains a balance between energy consumption and data transmission quality.

In addition, the use of classification and regression methods enables real-time assessment of the network state and decision-making regarding route changes or switching nodes to energy-saving modes.

Reinforcement learning-based approaches also deserve attention. In such models, nodes or node groups learn to make energy-efficient decisions through interaction with the environment – by trial and error. For example, an agent can learn to deactivate secondary nodes or adjust transmission power in response to changing network conditions, receiving "rewards" for saving energy without compromising service quality.

Finally, recent research focuses on integrating deep learning to detect complex dependencies in the large volumes of data generated by the network. For example, convolutional neural networks enable the identification of recurring traffic patterns, while recurrent neural networks allow for forecasting future node loads. This creates the foundation for proactive energy and transmission management.

In article [1], a modified Kohonen map method is proposed to improve the performance of artificial neural networks through the clustering of sensor nodes – parameters such as hop count, energy levels, sensitivity, and delay are calculated. The author suggests using artificial neural network-based clustering of sensor data, which leads to enhanced energy savings in wireless sen-

Harmash V., Diachenko V., Mikhal O., Znaidiuk V. Energysaving method in wireless sensor networks. Системи управління, навігації та зв'язку, вип.2. Полтава, 2025. С. 54-58

15

## Висновки

У результаті проведеного дослідження було здійснено комплексне вивчення проблеми енергозбереження в безпроводних сенсорних мережах та обґрунтовано доцільність впровадження методів машинного навчання для оптимізації споживання енергетичних ресурсів вузлів. Особливу увагу зосереджено на використанні модифікованих самоорганізаційних карт Кохонена, які забезпечують динамічну кластеризацію сенсорних вузлів з урахуванням поточного енергетичного стану, структури трафіку та географічного розміщення елементів мережі. Проведений теоретичний аналіз показав, що енергоспоживання в WSN залежить не лише від характеристик апаратного забезпечення, але й від обраної топології, частоти комунікації, механізмів маршрутизації та адаптивності до змін середовища.

Запропоновано та реалізовано метод енергозбереження, який поєднує алгоритм модифікованої карти Кохонена з динамічним енергозваженим навчанням, що дозволяє автоматично обирати кластерні голови, будувати маршрути передачі даних з мінімальними витратами енергії та керувати режимами активності вузлів. Реалізація методу у середовищі Google Colab підтвердила його ефективність: побудовано повноцінну модель кластеризації на основі навчального датасету, здійснено адаптивне групування вузлів, реалізовано перенавчання моделі в режимі оновлення, а також візуалізовано отримані результати. Застосування інструментів Python, зокрема бібліотек MiniSom, scikit-learn, pandas і matplotlib, дало змогу створити гнучке, модульне та масштабоване середовище для аналізу WSN.

Встановлено, що розроблений підхід дозволяє зменшити середнє енергоспоживання вузлів до 30–40% порівняно з традиційними методами кластеризації, рівномірно розподіляє навантаження між вузлами, знижує частоту переважання та подовжує час автономної роботи мережі. Крім того, модель демонструє стійкість до змін конфігурації та забезпечує можливість адаптації в реальному часі без потреби повного перенавчання, що є критично важливим для функціонування WSN у реальних динамічних умовах.

16

## ДОДАТОК Б

### Програмний код

#### Б.1 Встановлення бібліотек та завантаження даних

```
import numpy as np
import pandas as pd
!pip install minisom --quiet

np.random.seed(42)
n_nodes = 100

data = pd.DataFrame({
    'x': np.random.uniform(0, 100, n_nodes),
    'y': np.random.uniform(0, 100, n_nodes),
    'battery': np.random.uniform(0.1, 1.0, n_nodes),
    'traffic': np.random.uniform(0.0, 1.0, n_nodes),
    'rssi': np.random.uniform(-90, -30, n_nodes),
    'dist_to_bs': np.sqrt((100 - np.random.uniform(0, 100,
n_nodes))**2 +
                                (100 - np.random.uniform(0, 100,
n_nodes))**2)
})
data.head()
```

#### Б.2 Передобробка та нормалізація. Навчання модифікованої карти

##### Кохонена

```
from sklearn.preprocessing import MinMaxScaler
features = ['battery', 'traffic', 'rssi', 'dist_to_bs']
scaler = MinMaxScaler()
data_scaled = scaler.fit_transform(data[features])
from minisom import MiniSom
som_size = 10
som = MiniSom(som_size, som_size, len(features), sigma=1.0,
learning_rate=0.5)
som.random_weights_init(data_scaled)
som.train_random(data_scaled, 1000)

winner_coordinates = np.array([som.winner(x) for x in
data_scaled])
cluster_index =
np.ravel_multi_index(np.array(winner_coordinates).T, (som_size,
som_size))
```

### Б.3 Кластеризація та вибір кластерних голів. Побудова маршрутів та імітація трафіку

```

data['cluster'] = cluster_index
cluster_heads = data.groupby('cluster').apply(lambda df:
df.loc[df['battery'].idxmax()])
cluster_heads = cluster_heads.reset_index(drop=True)
cluster_heads

# Імітуємо просту маршрутизацію – від вузла до найближчого
кластерного голови
from scipy.spatial.distance import cdist

distances = cdist(data[['x', 'y']], cluster_heads[['x', 'y']])
nearest_heads = np.argmin(distances, axis=1)
data['assigned_head'] = nearest_heads
data.head()

```

### Б.4 Візуалізація результатів

```

import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(10, 8))
sns.scatterplot(x=data['x'], y=data['y'], hue=data['cluster'],
palette='tab10', s=40, legend=None)
plt.scatter(cluster_heads['x'], cluster_heads['y'], s=200,
c='red', edgecolors='black', label='Cluster Heads')
for i, head in cluster_heads.iterrows():
    plt.annotate(f'H{i}', (head['x'] + 1, head['y'] + 1),
color='black')
plt.title("WSN Clustering and Cluster Heads using Modified SOM")
plt.xlabel("X Coordinate")
plt.ylabel("Y Coordinate")
plt.grid(True)
plt.legend()
plt.show()

```

## ДОДАТОК В

### Додаткові дослідження

#### Б.3 Енергоефективне управління в безпроводних сенсорних мережах з використанням карт Кохонена. Код

```

!pip install minisom --quiet
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from minisom import MiniSom
from sklearn.preprocessing import MinMaxScaler
import networkx as nx
# Синтетичні дані
np.random.seed(42)
n_nodes = 100
coords = np.random.rand(n_nodes, 2) * 100
battery = np.random.uniform(0.2, 1.0, n_nodes)
traffic = np.random.uniform(0, 1, n_nodes)
signal_quality = np.random.uniform(0, 1, n_nodes)
dist_to_base = np.linalg.norm(coords - np.array([50, 50]),
axis=1)

data = np.vstack([coords.T, battery, traffic, signal_quality,
dist_to_base]).T
scaler = MinMaxScaler()
data_scaled = scaler.fit_transform(data)
som_size = 10
som = MiniSom(som_size, som_size, data_scaled.shape[1],
sigma=1.0, learning_rate=0.5)
som.random_weights_init(data_scaled)
som.train_random(data_scaled, 1000)

from pylab import bone, pcolor, colorbar, plot, show

bone()
pcolor(som.distance_map().T)
colorbar()
plt.title("U-Matrix SOM")
show()
winners = np.array([som.winner(x) for x in data_scaled])
cluster_map = {}
for i, win in enumerate(winners):
    cluster_map.setdefault(tuple(win), []).append(i)

cluster_heads = {}
for key, members in cluster_map.items():

```

```

    best = max(members, key=lambda i: battery[i] / (1 +
traffic[i]))
    cluster_heads[key] = best
G = nx.Graph()
for i, coord in enumerate(coords):
    G.add_node(i, pos=tuple(coord))

for members in cluster_map.values():
    for i in members:
        winner_tuple = tuple(winners[i])
if i != cluster_heads[winner_tuple]:
    G.add_edge(i, cluster_heads[winner_tuple])

base_station = np.array([50, 50])
for head in cluster_heads.values():
    G.add_edge(head, n_nodes) # нода для базової станції
G.add_node(n_nodes, pos=(50, 50))

pos = nx.get_node_attributes(G, 'pos')
nx.draw(G, pos, with_labels=False, node_size=100,
node_color='lightblue')
nx.draw_networkx_nodes(G, pos, nodelist=[n_nodes],
node_color='red', node_size=300, label='Base')
plt.title("Маршрути в WSN через кластерні голови")
plt.show()
energy_map = np.zeros((som_size, som_size))
for i, x in enumerate(data_scaled):
    w = som.winner(x)
    energy_map[w[0], w[1]] += battery[i]

plt.imshow(energy_map.T, cmap='viridis')
plt.title("Карта розподілу заряду батарей")
plt.colorbar(label="Battery sum")
plt.show()

```

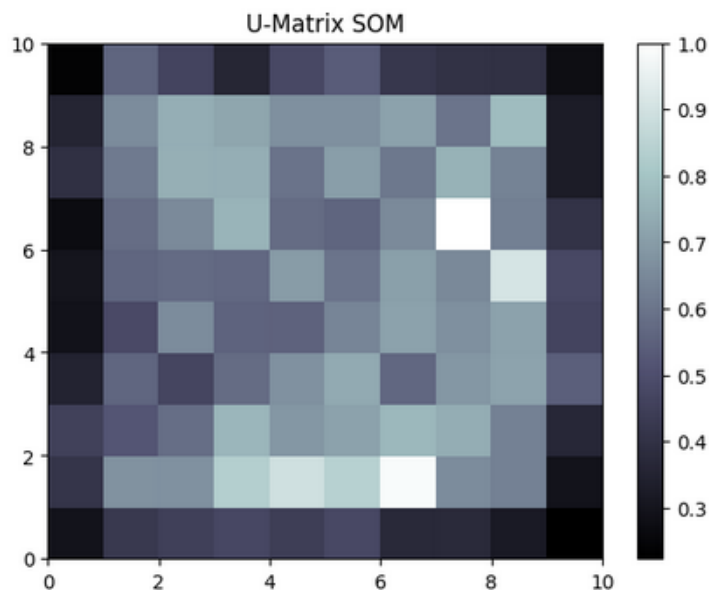


Рисунок В.1 – U-matrix SOM

Маршрути в WSN через кластерні голови

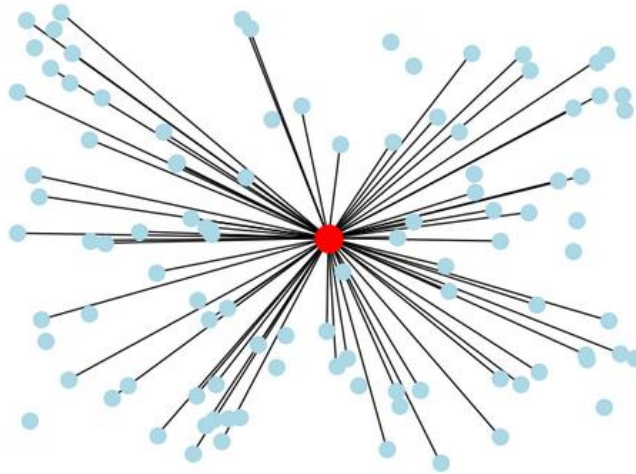


Рисунок Б.2 – Маршрути в WSN через кластерні голови

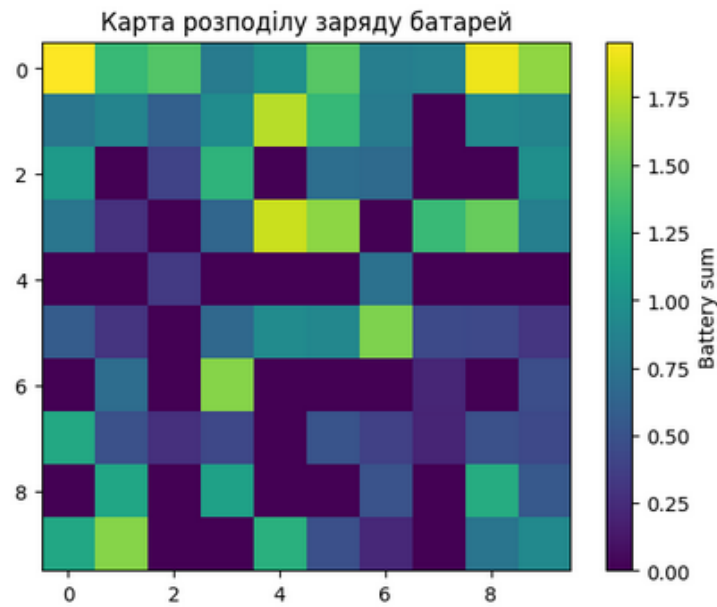


Рисунок В.3 – Карта розподілу заряду батарей