

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»  
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Кваліфікаційна наукова  
праця на правах рукопису

**ЛЕОЩЕНКО СЕРГІЙ ДМИТРОВИЧ**

УДК 004.855.5: 004.93: 004.8

**ДИСЕРТАЦІЯ**  
**МЕТОДИ СИНТЕЗУ РЕКУРЕНТНИХ НЕЙРОМЕРЕЖЕВИХ**  
**МОДЕЛЕЙ ДЛЯ ДІАГНОСТУВАННЯ**

122 – «Комп'ютерні науки»

12 – «Інформаційні технології»

Подається на здобуття наукового ступеня доктора філософії  
Дисертація містить результати власних досліджень. Використання ідей,  
результатів і текстів інших авторів мають посилання на відповідне джерело

---

С.Д. Леощенко

Науковий керівник  
Субботін Сергій Олександрович, доктор технічних наук, професор  
Науковий керівник  
Олійник Андрій Олександрович, доктор технічних наук, професор

Запоріжжя – 2023

## АНОТАЦІЯ

**Леощенко С.Д. Методи синтезу рекурентних нейромережових моделей для діагностування.** – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 – «Комп’ютерні науки» (12 – Інформаційні технології). – Національний університет «Запорізька політехніка», Запоріжжя, 2023.

У дисертаційній роботі розв’язано актуальну науково-прикладну задачу у галузі комп’ютерних наук: підвищення точності та зменшення часу побудови діагностичних моделей, збільшення їхніх інтерпретовності та узагальнювальних здатностей шляхом розроблення нових та удосконалення існуючих методів синтезу рекурентних нейромережових діагностичних моделей, які поєднують принципи інтелектуальних та паралельних обчислень.

Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел та додатків.

У вступі обґрунтовано актуальність теми дисертаційного дослідження, наведено мету, об’єкт та предмет дослідження. Так, метою роботи є розроблення та дослідження методів і засобів синтезу діагностичних моделей на основі рекурентних нейромереж, які поєднують інтелектуальні та паралельні обчислення для підвищення точності нейромоделей, їх рівня узагальнення даних та інтерпретабельності. Об’єкт дослідження – процес синтезу діагностичних нейромоделей на основі історичних даних. Предметом дослідження є методи побудови діагностичних моделей на основі рекурентних нейронних мереж. Також у вступі описано зв’язок роботи з науковими програмами, планами, темами; наведено методи дослідження та відображено практичне значення одержаних результатів дослідження; презентовано списки опублікованих праць за тематикою дисертаційної

роботи та конференцій, на котрих було апробовано основні результати дисертаційної роботи.

У першому розділі «Огляд проблемної області та постановка задачі дослідження» розглянуто стан проблеми діагностування на основі історичних даних про об'єкт. Проаналізовано різні типи штучних нейронних мереж, які можуть використовуватися як основа для діагностичних моделей при вирішенні завдань діагностування. Проаналізовано процес і методи синтезу таких діагностичних нейромоделей. Досліджено суттєві недоліки існуючих нейроеволюційних методів є синтезу нейромоделей. Обґрунтовано необхідність розробки нових методів синтезу діагностичних моделей на основі рекурентних нейронних мереж, що забезпечать одночасний синтез структури та параметрів нейромержі, що буде відрізнятися високою точністю, а подальше використання паралельних обчислень дозволить прискорити процес нейроеволюційного синтезу за рахунок використання високопродуктивних методів, які відрізняються скороченням часу синтезу навіть на великих вибірках історичних даних при побудові систем діагностування.

У другому розділі «Синтез рекурентних штучних нейронних мереж на основі нейроеволюційного підходу» розв'язано актуальне завдання синтезу діагностичних моделей на основі рекурентних нейромерж. Запропоновано методи синтезу нейромоделей на основі нейроеволюційного підходу, які дозволяють виконати процес синтезу нейромоделей із високим рівнем точності, що можуть використовуватися як базис для синтезу моделей для неруйнівного діагностування за історичними даними про систему та об'єкт. Розроблено модифікований генетичний метод для синтезу рекурентних нейромерж, в якому, на відміну від існуючих методів, на етапі кодування використовується спосіб кодування інформації про нейромоделі на основі секвенування, для більш компактного зберігання даних. Використання запропонованих критеріїв регулювання розміру та напрямку розвитку (еволюції) нейромоделі у розробленому методі дозволяє на етапі мутації

адаптивно вибирати тип зміни структури нейромережі, що перешкоджає безмірному збільшенню мережі, не обмежуючи кількості прихованих нейронів, та синтезувати нейромоделі, що характеризуються високою точністю. Запропоновано паралельний генетичний метод з імплементацією механізмів селективного тиску, в якому основні етапи процесу еволюційного синтезу виконуються на паралельних обчислювальних вузлах, використовуються механізми селективного тиску та рівномірного схрещування, що дозволяє скорочувати розміри популяції, не враховуючі та не опрацьовуючі ті особини з популяції, що відрізняються малим значенням функції пристосованості. Застосування модифікованих генетичних операторів відбору та схрещування дозволяє підтримувати генетичне різноманіття популяцій, забезпечити уникнення проблем з надмірною ітеративною природою еволюційних обчислень, враховувати додаткові показники якості нейромережових моделей, більш продуктивно використовувати механізми рівномірного багатобатьківського схрещування. Збільшення точності та скорочення витрат пам'яті й обчислювальних потужностей для збереження та проведення схрещування повного обсягу популяції засвідчують високу ефективність запропонованого методу.

У третьому розділі «Структурна оптимізація синтезованих нейромережових моделей» розв'язано актуальне завдання вибору механізмів для доналаштування нейромерж на основі індикаторної системи для оцінки рівня складності задачі. Запропоновано метод структурного доналаштування синтезованих нейромоделей на основі нейроеволюційного підходу з використанням системи індикаторів та критеріїв для адаптивного визначення мутаційних змін. Запропоновано систему індикаторів для оцінювання рівня складності задачі для подальшого моделювання, діагностування або прогнозування. Так на основі врахування характеристик вхідного набору даних, рівня можливого спрощення структури, загальної кількості суттєвих та несуттєвих факторів, рівня точності вимірювань та рівня можливого контролю та управління можливо в подальшому обрати варіант

доналаштування математичної моделі. Такий підхід дозволяє попередньо проаналізувати та визначити механізми структурного доналаштування нейромереж без використання надмірних обчислювальних ресурсів для повторного синтезу нейромоделей, таким чином, запропоновані індикатори не вимагають додаткової побудови моделі, що істотно знижує використання часових та обчислювальних ресурсів під час процесу аналізу варіантів підбору та розвитку математичної моделі. Запропоновано метод структурного доналаштування для підвищення характеристик нейропластичності мереж. Так, під час роботи із великими даними застосування складних топологій нейромереж часто обмежується обчислювальними ресурсами. Саме тому структурна оптимізація дозволяє отримати полегшену та проріджену структуру попередньо синтезованої нейромоделі. Це гарантує не тільки зменшення навантаження на обчислювальну систему, а й пришвидшення роботи такої моделі. Більш того, у якості побічного ефекту структурної оптимізації іноді може спостерігатися підвищення точності роботи, адже частина вузлів та зв'язків для кодування зашумлених даних може бути видалена.

У четвертому розділі «Експериментальне дослідження методів синтезу рекурентних нейромережеских моделей діагностування складних технічних об'єктів та систем» виконано експериментальне дослідження розроблених методів аналізу даних та синтезу діагностичних нейромоделей. Узагальнюючи отримані результати проведених експериментів, можна відзначити, що застосування запропонованих у дисертаційній роботі методів синтезу діагностичних моделей на основі рекурентних нейромереж дозволило значно підвищити точність синтезу нейромоделей у порівнянні з методами, які передбачають послідовну реалізацію обчислень, збільшити рівні інтерпретовності та логічної прозорості моделей. Розроблені методи можуть застосовуватися до широкого кола задач діагностування, де є накопичені дані про об'єкт діагностування проте не є достатніми експертні знання та можуть

бути використані при синтезі діагностичних моделей на основі великих вибірок даних.

Наукові положення, висновки і рекомендації, викладені в дисертаційній роботі, використано при підготовці дисциплін «Інтелектуальні системи», «Основи обчислювального інтелекту», «Інтелектуальний аналіз даних» та НДР на кафедрі програмних засобів Національного університету «Запорізька політехніка» (акт впровадження від 20.12.2022). Результати дисертаційної роботи апробовано та впроваджено у практичну діяльність Товариства з обмеженою відповідальністю «Аптека Магнолія» (акт впровадження від 18.08.2022), у практичну діяльність Товариства з обмеженою відповідальністю «Запорізький ливарно-механічний завод» (акт впровадження від 20.09.2022).

**Ключові слова:** технічне діагностування, біомедичне діагностування, системи контролю, моделювання, інтегровані системи, машинне навчання, класифікація, штучні нейронні мережі, рекурентні нейронні мережі, нейроеволюція, генетичний алгоритм, синтез, система індикаторів, структурне доналаштування.

## ABSTRACT

**Leoshchenko S.D. Methods for synthesizing recurrent neural network models for diagnostics.** – Qualifying scientific work on the rights of the manuscript.

Thesis paper for achievement of the scientific degree Doctor of Philosophy in the specialty 122 “Computer Science” (12 “Information Technology”). – National University “Zaporizhzhia Polytechnic”, Zaporizhzhia, 2023.

In the dissertation work, an actual scientific and applied problem in the field of computer science is solved: improving the accuracy and reducing the time of constructing diagnostic models, increasing their interpretability and generalizing abilities by developing new and improving existing methods for synthesizing recurrent neural network diagnostic models that combine the principles of intelligent and parallel computing.

The dissertation consists of an introduction, four chapters, conclusions, a list of references, and appendices.

In the introduction, the relevance of the topic of dissertation research is justified, the purpose, object and subject of research are given. Thus, the aim of the work is to develop and research methods and tools for synthesizing diagnostic models based on recurrent neural networks that combine intelligent and parallel computing to improve the accuracy of neuromodels, their level of data generalization and interpretability. The object of research is the process of synthesizing diagnostic neuromodels based on historical data. The subject of the research are methods for constructing diagnostic models based on recurrent neural networks. The introduction also describes the relationship of the work with scientific programs, plans, topics; provides research methods and reflects the practical significance of the research results obtained; presents lists of published works on the subject of the dissertation work and conferences at which the main results of the dissertation work were tested.

In the first chapter, "Overview of the problem area and formulation of the research problem", the state of the diagnostic problem based on historical data about the object is considered. Various types of artificial neural networks that can be used as the basis for diagnostic models in solving diagnostic problems are analyzed. The process and methods of synthesis of such diagnostic neuromodels are analyzed. Significant disadvantages of existing neuroevolution methods for the synthesis of neuromodels are investigated. The necessity of developing new methods for synthesizing diagnostic models based on recurrent neural networks, which will provide simultaneous synthesis of the structure and parameters of neuromerge, which will be characterized by high accuracy, and further use of parallel computing will speed up the process of neuroevolution Synthesis due to the use of high-performance methods that are characterized by a reduction in synthesis time even on large samples of historical data in the construction of diagnostic systems, is justified.

In the second chapter, "Synthesis of recurrent artificial neural networks based on the neuroevolution approach", the actual problem of synthesis of diagnostic models based on recurrent neuromerges is solved. Methods for the synthesis of neuromodels based on the neuroevolution approach are proposed, which allow performing the process of synthesis of neuromodels with a high level of accuracy, which can be used as a basis for the synthesis of models for non-destructive diagnostics based on historical data about the system and object. A modified genetic method for the synthesis of recurrent neural networks has been developed, which, unlike existing methods, uses a method of encoding information about a neuromodel based on sequencing at the coding stage, for more compact data storage. The use of the proposed criteria for regulating the size and direction of development (evolution) of the neuromodel in the developed method allows adaptive selection at the mutation stage of the type of change in the structure of the neural network, which prevents the immeasurable increase in the network, without limiting the number of hidden neurons, and synthesize neuromodels characterized by high accuracy. A parallel genetic method with the implementation of selective



pressure mechanisms is proposed, in which the main stages of the process of evolutionary synthesis are performed on parallel computing nodes, selective pressure mechanisms and uniform crossing are used, which allows reducing the size of the population, without taking into account and without processing those individuals from the population that differ in a small value of the fitness function. The use of modified genetic selection and crossing operators allows maintaining the genetic diversity of populations, avoiding problems with the excessive iterative nature of evolutionary calculations, taking into account additional quality indicators of neural network models, and using mechanisms of uniform multi-parent crossing more productively. Increasing accuracy and reducing the cost of memory and computing power for preserving and crossing the full population size indicate the high efficiency of the proposed method.

In the third chapter, "Structural optimization of synthesized neural network models", the actual problem of choosing mechanisms for additional adjustment of neural network based on an indicator system for assessing the level of complexity of the problem is solved. A method of structural adjustment of synthesized neuromodels based on the neuroevolution approach using a system of indicators and criteria for adaptive determination of mutational changes is proposed. A system of indicators for assessing the level of complexity of the problem for further modeling, diagnostics or forecasting is proposed. Thus, based on taking into account the characteristics of the input data set, the level of possible simplification of the structure, the total number of significant and insignificant factors, the level of measurement accuracy and the level of possible control and management, it is possible to choose the option of additional adjustment of the mathematical model in the future. This approach allows you to pre-analyze and determine the mechanisms of structural adjustment of neural networks without using excessive computational resources for re-synthesis of neuromodels, thus, the proposed indicators do not require additional model construction, which significantly reduces the use of time and computational resources during the process of analyzing options for selecting and developing a mathematical model. A

method of structural adjustment is proposed to improve the characteristics of neuroplasticity of networks. For example, when working with big data, the use of complex neural network topologies is often limited to computing resources. This is why structural optimization allows us to obtain a simplified and thinned structure of a pre-synthesized neuromodel. This guarantees not only a reduction in the load on the computing system, but also an acceleration of the operation of such a model. Moreover, as a side effect of structural optimization, sometimes there can be an increase in the accuracy of work, because some nodes and connections for encoding noisy data can be removed.

In the fourth chapter, "Experimental study of methods for synthesizing recurrent neural network models for diagnosing complex technical objects and systems", an experimental study of the developed methods for data analysis and synthesis of diagnostic neuromodels is performed. Summarizing the results of the conducted experiments, it can be noted that the application of the methods of synthesis of diagnostic models based on recurrent neural network proposed in the dissertation work made it possible to significantly increase the accuracy of synthesis of neuromodels in comparison with methods that involve sequential implementation of calculations, to increase the levels of interpretability and logical transparency of models. The developed methods can be applied to a wide range of diagnostic tasks, where there is accumulated data about the diagnostic object, but there is insufficient expert knowledge and can be used in the synthesis of diagnostic models based on large data samples.

Scientific provisions, conclusions and recommendations set out in the dissertation work are used in the preparation of the disciplines "Intelligent Systems", "Fundamentals of computational intelligence", "Data Mining" and research at the Department of software tools of the National University "Zaporizhzhia Polytechnic" (Implementation Act of 20.12.2022). The results of the dissertation work were tested and implemented in the practical activities of the Limited Liability Company "Pharmacy Magnolia" (Implementation Act of 18.08.2022), in the practical

activities of the Limited Liability Company “Zaporizhzhia foundry and mechanical plant” (Implementation Act of 20.09.2022).

**Keywords:** technical diagnostics, biomedical diagnostics, control systems, modeling, integrated systems, machine learning, classification, artificial neural networks, recurrent neural networks, neuroevolution, synthesis, genetic algorithm, indicator system, structural adjustment.

## СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

**Список праць, в яких опубліковані основні наукові результати дисертації:**

1. Modification and Parallelization of Genetic Algorithm for Synthesis of Artificial Neural Networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Lytvyn, V. Shkarupylo] // Radio Electronics, Computer Science, Control. – 2019. – № 4. – P. 68–82. (Web of Science)

2. Usage of Swarm Intelligence Strategies During Projection of Parallel Neuroevolution Methods for Neuromodel Synthesis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko] // Technology audit and production reserves. – 2020. – Vol. 5, Issue 2/55. – P. 12–17.

3. Implementation of the Indicator System in Modeling Complex Technical Systems / [S. Leoshchenko, S. Subbotin, A. Oliinyk, O. Narivs'kiy] // Radio Electronics, Computer Science, Control. – 2021. – № 1. – P. 117–126. (Web of Science)

4. Синтез та використання нейромережових моделей з ймовірнісним кодуванням структури / [С. Д. Леощенко, А. О. Олійник, С. О.Субботін, Є. О. Гофман, М. Б. Ільяшенко] // Radio Electronics, Computer Science, Control. – 2021. – № 2. – P. 93–104. (Web of Science)

5. . Метод структурного доналаштування нейромережових моделей для забезпечення інтерпретабельності / [С. Д. Леощенко, А. О. Олійник, С. О.Субботін, Є. О. Гофман, О. В. Корнієнко] // Radio Electronics, Computer Science, Control. – 2021. – № 3. – P. 86–96. (Web of Science)

6. Synthesis of a Neural Network Model of Industrial Construction Processes Using an Indicator System / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Netrobko, Ye. Gofman] // Radio Electronics, Computer Science, Control. – 2021. – № 4. – P. 69–77. (Web of Science)

7. Neuromodeling of Operational Processes / [S.A. Subbotin, H.V. Pukhalska, S.D. Leoshchenko, A.O. Oliinyk, Ye. O. Gofman] // Radio Electronics, Computer Science, Control. – 2022. – № 1. – P. 120-129. (Web of Science)

8. Neural Network Diagnostics of Aircraft Parts Based on the Results of Operational Processes / [S. Leoshchenko, H. Pukhalska, S. Subbotin, A. Oliinyk, Ye. Gofman] // Radio Electronics, Computer Science, Control. – 2022. – № 2. – P. 69-79. (Web of Science)

**Список праць, які засвідчують апробацію матеріалів дисертації:**

9. Method of Artificial Neural Network Synthesis for Using in Integrated CAD / [S. Leoshchenko, A. Oliinyk, S. Subbotin, S. Shylo, V. Shkarupylo] // The experience of Designing and Application of CAD systems (CADSM): 15th International Conference CADSM'2019, Polyana (Svalyava), 26 February – 2 March 2019 : proceedings. – Lviv : Lviv Polytechnic National University, 2019. – P. 24-29. (Scopus)

10. Modification of the Genetic Method for Neuroevolution Synthesis of Neural Network Models For Medical Diagnosis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, N. Gorobii, V. Shkarupylo] // Proceedings of the Second International Workshop on Computer Modeling and Intelligent Systems (CMIS-2019), Zaporizhzhia, Ukraine, April 15-19, 2019. – P. 143-158. – (CEUR Workshop Proceedings, Vol. 2353). (Scopus)

11. Parallel Genetic Method for the Synthesis of Recurrent Neural Networks for Using in Medicine / [S. Leoshchenko, A. Oliinyk, S. Skrupsky, S. Subbotin, V. Lytvyn] // Proceedings of the Second International Workshop on Computer Modeling and Intelligent Systems (CMIS-2019), Zaporizhzhia, Ukraine, April 15-19, 2019. – P. 1-17. – (CEUR Workshop Proceedings, Vol. 2353). (Scopus)

12. Parallel Method of Neural Network Synthesis Based on a Modified Genetic Algorithm Application / [S. Leoshchenko, A. Oliinyk, S. Skrupsky, S. Subbotin, T. Zaiko] // Proceedings of the Workshop on Modern Machine Learning

Technologies and Data Science (MoMLLeT&DS-2019), Shatsk, Ukraine, 2-4 June 2019. – P. 11-23. – (CEUR Workshop Proceedings, Vol. 2386). (Scopus)

13. Implementation of Selective Pressure Mechanism to Optimize Memory Consumption in the Synthesis of Neuromodels for Medical Diagnostics / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko, N. Gorobii] // Proceedings of the 2nd International Workshop on Informatics and Data-Driven Medicine IDDM 2019, Lviv, Ukraine, 11–13 November 2019. – P. 109–120. – (CEUR Workshop Proceedings, Vol. 2488). (Scopus)

14. Smart Crossover Mechanism for Parallel Neuroevolution Method of Medical Diagnostic Models Synthesis / [S. Leoshchenko, S. Subbotin, A. Oliinyk, V. Lytvyn, M. Ilyashenko] // Proceedings of the Third International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020), Zaporizhzhia, Ukraine, April 27-May 1, 2020. – P. 57-69. – (CEUR Workshop Proceedings, Vol. 2608) (Scopus)

15. Using neuromodels for evaluating and determining productivity of technical processes / [S. Leoshchenko, O. Nazarenko, A. Oliinyk, S. Subbotin, T. Zaiko, V. Donenko] // Proceedings of the International Conference "Problems of Infocommunications. Science and Technology" (PIC S&T 2020), Kyiv, Ukraine, 6-9 October 2020 : proceedings of the conference. – Kharkiv : Kharkiv National University of Radio Electronics, 2020. - P.442-446 (Scopus)

16. Implementation of swarm procedures for parallelization neuroevolution methods / S. Leoshchenko, A. Oliinyk, S. Subbotin, Y. Gofman, V. Lytvyn // Proceedings of the 7th International Conference on Control and Optimization with Industrial Applications (COIA 2020), Baku, Azerbaijan, 26-28 August 2020 : proceedings of the conference. – Baku : Baku State University, 2020. – P. 221-224. (Web of Science)

17. Leoshchenko, S. Adaptive Mechanisms for Parallelization of the Genetic Method of Neural Network Synthesis / S. Leoshchenko, A. Oliinyk, S. Subbotin // Proceedings of the 10th International Conference on Advanced Computer Information Technologies (ACIT 2020), Deggendorf, Germany, 16-18

September 2020 : proceedings of the conference. – Ternopil : West Ukrainian National University, 2020. – P. 446-450. (Scopus)

18. Sequencing for Encoding in Neuroevolutionary Synthesis of Neural Network Models for Medical Diagnosis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko, S. Shylo, V. Lytvyn] // Proceedings of the 3rd International Conference on Informatics & Data-Driven Medicine (IDDM 2020), Växjö, Sweden, 19-21 October 2020. – P. 62-71. – (CEUR Workshop Proceedings, Vol. 2753) (Scopus)

19. Using the Actor-Critic method for population diversity in neuroevolutionary synthesis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Shkarupylo] // Proceedings of the 2nd International Workshop on Intelligent Information Technologies and Systems of Information Security (IntelITSIS'2021), Khmelnytskyi, Ukraine, 24-26 March 2021. – P. 99–107. – (CEUR Workshop Proceedings, Vol. 2853). (Scopus)

20. Implementation of Probabilistic Data Structures in the Processes of Neuroevolutionary Synthesis / [S. Leoshchenko, S. Subbotin, A. Oliinyk, V. Lytvyn, O. Korniienko] // Proceedings of The Fourth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2021), Zaporizhzhia, Ukraine, April 27 2021. – P. 59-72. – (CEUR Workshop Proceedings, Vol. 2864).

21. Implementation of Reinforcement Learning Strategies in the Synthesis of Neuromodels to Solve Medical Diagnostics Tasks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Lytvyn, O. Korniienko] // Proceedings of the 4th International Conference on Informatics & Data-Driven Medicine (IDDM-2021), Valencia, Spain, 19-21 November 2021. – P. 34-43. – (CEUR Workshop Proceedings, Vol. 3038) (Scopus)

22. Mechanisms of fine tuning of neuroevolutionary synthesis of artificial neural networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, M. Zaiko] // Proceedings of The Fourth IEEE International Conference on Advanced Information and Communication Technologies (AICT-2021), Lviv, Ukraine, September 21-25 2021. – P. 122-127. (Scopus)

23. Neuroevolutionary Mechanisms in the Synthesis of Spiking Neural Networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, M. Pyashenko, A. Borovikov] // Proceedings of The Fifth International Workshop on Computer Modeling and Intelligent Systems CMIS-2022), Zaporizhzhia, Ukraine, May 12 2022. – P. 88-97. – (CEUR Workshop Proceedings, Vol. 3137).

**Список праць, які додатково відображають наукові результати дисертації:**

24. Методи та засоби оброблення великих даних в системах діагностування та розпізнавання образів : монографія / [С. О. Субботін, А. О. Олійник, В. М. Льовкін, Т. О. Колпакова, М. Ю. Пришляк, С. Д. Леощенко, О. В. Корнієнко, Д. А. Каврін, Є. О. Гофман, О. Ю. Благодарьов] ; під заг. ред. С. О. Субботіна, А. О. Олійника. – Запоріжжя : ЗНТУ, 2018. – 228 с.

25. Using recurrent neural networks for data-centric business / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko] // Data-Centric Business and Applications - Evolvments in Business Information Processing and Management. – Cham, Switzerland: Springer, 2020. – P. 73-91. – (Lecture Notes on Data Engineering and Communications Technologies, Vol. 3)



## ЗМІСТ

Анотація.....	2
Abstact .....	7
Список опублікованих праць за темою дисертації.....	12
Зміст .....	17
Перелік умовних скорочень.....	19
Вступ.....	21
1 Огляд проблемної області та постановка задачі дослідження.....	30
1.1 Діагностування на основі історичних даних про об’єкт .....	30
1.2 Діагностичні моделі .....	35
1.3 Постановка завдання .....	41
1.4 Аналіз топологій штучних нейронних мереж для діагностування на основі даних.....	42
1.5 Методи синтезу штучних нейронних мереж .....	51
1.6 Висновки за розділом 1 .....	59
2 Синтез рекурентних штучних нейронних мереж на основі нейроеволюційного підходу .....	62
2.1 Генетичний метод визначення ваг прихованих і вихідних шарів PHM .....	63
2.1.1 Кодування вхідної інформації.....	66
2.1.2 Адаптивний механізм вибору типу мутації .....	71
2.1.3 Використання генетичних операторів .....	75
2.2 Паралелізація нейроеволюційного синтезу рекурентних штучних нейронних мереж .....	81
2.2.1 Проблеми паралелізації нейроеволюційних методів.....	81
2.2.2 Використання стратегій ройового інтелекту при паралелізації процесу синтезу штучних нейронних мереж .....	84
2.2.3 Використання механізму селективного тиску .....	87
2.3 Висновки за розділом 2 .....	93

	18
3 Структурна оптимізація синтезованих нейромережових моделей.....	95
3.1 Застосування індикаторної системи при визначенні механізмів доналаштування штучних нейронних мереж .....	96
3.2 Імплементация властивості нейропластичності у вигляді механізмів прорідження штучних нейронних мереж .....	102
3.3 Метод структурної оптимізації штучних нейронних мереж на основі генетичних операторів .....	108
3.4 Висновки за розділом 3 .....	112
4 Експериментальне дослідження методів синтезу рекурентних нейромережових моделей діагностування складних технічних об'єктів та систем .....	114
4.1 Умови та особливості проведення експериментальних досліджень методів синтезу рекурентних нейромережових моделей .....	114
4.2 Експериментальне дослідження послідовного методу синтезу штучних рекурентних нейронних мереж.....	115
4.3 Експериментальне дослідження паралельного методу синтезу штучних рекурентних нейронних мереж.....	126
4.4 Експериментальне дослідження індикаторної системи при визначенні механізмів доналаштування штучних нейронних мереж .....	133
4.5 Експериментальне дослідження методу структурної оптимізації попередньо синтезованих штучних нейронних мереж .....	142
4.6 Висновки за розділом 4 .....	146
Висновки.....	149
Список використаних джерел.....	153
Додаток А .....	177
Додаток Б .....	184
Додаток В.....	191

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

A3C – asynchronous advantage actor critic method (асинхронний критик-актор переваги метод);

CMA-ES – covariance matrix adaptation evolution strategy (коваріаційна матриця стратегії еволюції адаптації);

CNN – convolutional neural network (згорткова нейронна мережа);

CoSyNE – computational and systems neuroscience (обчислювальна та системна нейронаука);

CPU – central processing unit (центральний процесор);

ESP – enforced subpopulations (посилені субпопуляції);

GPU – graphics processing unit (графічний процесор);

LSTM – long short-term memory (мережі довготривалої короткочасної пам'яті);

NEAT – neuroevolution of augmenting topologies (нейроеволюційний метод наростаючої топології);

OBS – optimal brain surgeon (оптимальне проріджування нейронних мереж);

PSO – particle swarm optimization (ройові процедури);

RBF – radial basis function (нейронна мережа з радіальною базисною функцією);

SANE – symbiotic adaptive neuro-evolution (симбіотично адаптивна нейроеволюція);

WANN – weight agnostic neural network (нечутливі до ваг нейронні мережі);

БП – багатошаровий персептрон;

БС – безладна складність;

ГА – генетичний алгоритм;

ДНК – дезоксирибонуклеїнова кислота;

ГНМ – глибока нейронна мережа;

ЗПЧ – зворотне поширення в часі;

КВК – кількість власних коливань;

МГА – модифікований генетичний метод для синтезу рекурентних нейронних мереж;

МЗПП – метод зворотного поширення помилки;

МЗППвЧ – метод зворотного поширення помилки в часі;

ОП – організована простота;

ОС – організована складність;

ПГА – паралельні генетичні алгоритми;

ПГМзСТ – паралельний генетичний метод з імплементацією механізмів селективного тиску;

РНК – рибонуклеїнова кислота;

РНМ – рекурентна нейронна мережа;

СПЛФ – складність пов'язана з людським фактором;

ШНМ – штучна нейронна мережа.

## ВСТУП

**Обґрунтування вибору теми дослідження.** Для забезпечення довговічності і надійності функціонування складних об'єктів і систем використовують методи діагностування [1]-[4]. Як правило, в більшості практичних випадків об'єкти діагностування не є повністю дослідженими, оскільки теоретичні або експериментальні знання про такі системи є неповними або ж навпаки кількість експериментальних знань надлишкова, що значно ускладнює їх подальше опрацювання [5], [6]. У таких випадках доцільно використовувати методи діагностування, керовані даними, які здатні будувати моделі за історичними даними про систему (об'єкт) [7], [8]. Проте класичні методи діагностування [6]-[8] погано працюють для моделювання багатовимірних, нелінійних залежностей: через структурні та параметричні особливості такі моделі не можуть в повній мірі інтерпретувати складні, нелінійні зв'язки між усіма параметрами та складовими системи, а відповідно точність таких моделей може бути недостатньою [7], [8].

Методи обчислювального інтелекту [9]-[16] (зокрема, методи машинного навчання [9]-[11], штучні нейронні мережі [12], [13] та нейроеволюція [14]-[16]) є перспективним базисом для створення моделей при розробці систем діагностування, оскільки дозволяють будувати нелінійні моделі складних багатовимірних об'єктів за спостереженнями [17]. Рекурентні нейронні мережі [18], [19] є різновидом повнозв'язаних мереж [12], [13] та завдяки структурним особливостям (наявність зворотних зв'язків та внутрішньої пам'яті), дозволяють діагностувати та прогнозувати динамічну поведінку або стан діагностованої системи в часі [17]-[19].

Значний внесок у розвиток обчислювального інтелекту, зокрема штучних нейронних і рекурентних мереж та еволюційних обчислень, внесли А. Бергель [20], І. Боєрс [21], Г. Куйпер [21], Г. Блокдик [22], Ф. Груау [23], Т. Сугено [24], О.Г. Івахненко [25], Є.В. Бодянський [26], [27], В.Д. Дмитриєнко [28] та ін.

Проте процес побудови діагностичних моделей на основі рекурентних нейронних мереж оптимальної структури є високоітеративним. Відомі методи синтезу рекурентних нейронних мереж [48]-[78] характеризуються значними витратами часу, а синтезовані моделі, як правило, мають надлишкову структуру та недостатню точність, що спричиняє низькі рівні узагальнення даних та інтерпретабельності [67]-[78]. Це суттєво обмежує практичне застосування рекурентних нейронних мереж.

Тому актуальною науково-прикладною проблемою є підвищення точності та зменшення часу побудови діагностичних моделей, збільшення їхніх інтерпретовності та узагальнювальних здатностей шляхом розроблення нових та удосконалення існуючих методів синтезу рекурентних нейромережових діагностичних моделей, які поєднують принципи інтелектуальних та паралельних обчислень.

#### **Зв'язок роботи з науковими програмами, планами, темами.**

Дисертаційна робота відповідає «Концепції розвитку штучного інтелекту в Україні» (схвалена Розпорядженням Кабінету Міністрів України від 2.12.2020 № 1556-р), Розпорядженню Кабінету Міністрів України «Про затвердження плану заходів з реалізації Концепції розвитку штучного інтелекту в Україні на 2021-2024 роки» від 12.05.2021 р. № 438-р, Постанові Верховної Ради України «Про затвердження завдань Національної програми інформатизації на 2022-2024 роки» від 8.07.2022 № 2360-ІХ, Постанові Кабінету Міністрів України «Про затвердження Національної економічної стратегії на період до 2030 року» від 3.03.2021 р. № 179 та виконана відповідно до планів науково-дослідних робіт (НДР) Національного університету «Запорізька політехніка» у межах держбюджетних тем: «Розроблення та дослідження методів синтезу діагностичних моделей на основі методів обчислювального інтелекту» (номер державної реєстрації – 0119U100360), «Розроблення методів та засобів для аналізу та прогнозування динамічної поведінки нелінійних об'єктів» (номер державної реєстрації – 0121U107499), «Інтелектуальні методи та засоби діагностування та

прогнозування стану складних об'єктів» (номер державної реєстрації – 0122U000972). У зазначених НДР здобувач брав участь як виконавець, удосконалив та розробив методи нейроеволюційного синтезу рекурентних нейронних мереж.

**Мета та завдання дослідження.** Метою дисертаційної роботи є розроблення та дослідження методів і засобів синтезу діагностичних моделей на основі рекурентних нейромереж, які поєднують інтелектуальні та паралельні обчислення для підвищення точності нейромоделей, їх рівня узагальнення даних та інтерпретабельності.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

- проаналізувати існуючі методи синтезу моделей на основі рекурентних нейронних мереж;
- розробити метод синтезу діагностичних моделей на основі рекурентних нейронних мереж та нейроеволюційного підходу;
- розробити паралельний метод синтезу діагностичних моделей на основі рекурентних нейронних мереж;
- розробити індикаторну систему для визначення механізмів доналаштування параметрів діагностичних моделей на основі штучних нейронних мереж;
- розробити метод структурної оптимізації діагностичних нейромоделей;
- провести експериментальне дослідження розроблених методів синтезу та доналаштування діагностичних моделей.

**Об'єкт дослідження** – процес синтезу діагностичних нейромоделей на основі історичних даних.

**Предмет дослідження** – методи побудови діагностичних моделей на основі рекурентних нейронних мереж.

**Методи дослідження.** Для вирішення завдань, поставлених у дисертаційній роботі, використано методи: теорії рекурентних нейронних мереж – як базис для синтезу діагностичних моделей; теорії

нейроеволюційного підходу до синтезу – для створення нових методів синтезу рекурентних нейронних мереж; теорії паралельних обчислень – для розроблення паралельних методів синтезу рекурентних діагностичних нейромоделей; стохастичного пошуку та лінійної алгебри для створення нових методів синтезу рекурентних нейронних мереж; математичної статистики – для аналізу ефективності розроблених методів синтезу діагностичних нейромоделей та перевірки достовірності отриманих результатів.

**Наукова новизна отриманих результатів.** Основні результати, які визначають наукову новизну роботи, полягають у такому [75], [120], [139], [151], [173], [175], [190].

Вперше запропоновано генетичний метод синтезу діагностичних моделей на основі рекурентних нейронних мереж, який використовує механізм секвенування на етапі кодування інформації про популяції нейронних мереж з метою компактного збереження та передачі інформації, а також критерії регулювання розміру та напрямку розвитку нейромереж, що дозволяє підвищити адаптивність вибору мутації структури нейромоделі для підтримки генетичного різноманіття та уникнення завчасної збіжності з врахуванням перешкоджання безмірного збільшення мережі, і таким чином дозволяє підвищити точність синтезованих моделей та оптимізувати витрати ресурсів пам'яті обчислювальної системи під час синтезу для подальшого підвищення якості діагностування [120].

Вперше запропоновано паралельний генетичний метод синтезу діагностичних моделей на основі рекурентних нейронних мереж з імплементацією механізмів селективного тиску, в якому основні етапи процесу нейроеволюційного синтезу рекурентних нейромоделей виконуються на паралельних обчислювальних вузлах, а використання механізмів селективного тиску та рівномірного схрещування дозволяє підтримувати розміри популяції, скорочуючи обчислювальні витрати на опрацювання малопристосованих особин, більш того такі генетичні



оператори дозволяють скоротити високоітеративну природу еволюційних та паралельних обчислень, враховувати додаткові структурні та параметричні показники якості нейронних мереж в популяції, більш контрольовано використовувати механізм багатобатьківського схрещування та знизити вимоги до ресурсів пам'яті та обчислювальних потужностей електронно-обчислювальних машин під час синтезу моделей [120], [139].

Вперше запропоновано метод структурної оптимізації діагностичних нейромоделей для підвищення характеристик інтерпретабельності моделей, що направлено на оптимізацію роботи складних топологій нейронних мереж із великими даними, за рахунок спрощеної та прорідженої структури попередньо синтезованої нейромоделі, що полягає у використанні генетичних операторів та критеріального відбору для визначення структурного спрощення нейронної мережі, що дозволяє прорідити та оптимізувати структуру нейронної мережі, прискорюючи роботу нейромоделі та видаляючи міжнейронні зв'язки, що кодують зашумлені дані [173].

Удосконалено систему індикаторів для оцінювання рівня складності задачі з метою визначення механізмів доналаштування параметрів нейронних мереж, яка відрізняється врахуванням характеристик вхідного набору даних та характеристик задачі, серед яких: рівень можливого спрощення структури, загальна кількість суттєвих та несуттєвих факторів, рівень точності вимірювань та рівень можливого контролю та управління, що дають можливість точно обрати варіант доналаштування діагностичної моделі, без використання надмірних обчислювальних ресурсів для повторного синтезу нейромоделей або додаткової побудови моделі, що значно знижує ресурсоемність [151].

**Практичне значення отриманих результатів** полягає у тому, що запропоновані методи синтезу діагностичних моделей доведені до рівня практичної реалізації та дозволяють виконувати побудову та структурну оптимізацію діагностичних моделей, що базуються на рекурентних

нейронних мережах та відрізняються високими рівнями точності, інтерпретабельності і зручні для подальшого аналізу та використання на практиці.

Результати експериментального дослідження запропонованих у дисертаційній роботі методів синтезу рекурентних діагностичних нейромоделей дозволило значно підвищити швидкість синтезу діагностичних моделей (у 5-10 разів) у порівнянні з методами, які передбачають послідовну реалізацію обчислень [182], [183], [194]–[197], збільшити рівні інтерпретовності та узагальнення даних синтезованих моделей (на 22%) [173]. Розроблені методи можуть застосовуватися до широкого кола задач діагностування [178], [179], [182], [193], де є недостатніми експертні знання про системи та об'єкти діагностування (або природа яких є недостатньо вивченою), є адаптивними відносно історичних даних отриманих на базі експериментальних спостережень та можуть бути використані при синтезі діагностичних моделей на основі великих вибірок даних [194], [196], [197]. Розроблені методи синтезу діагностичних моделей на основі рекурентних нейронних мереж рекомендуються до застосування на практиці при створенні інтелектуальних систем діагностування у різних галузях, зокрема у авіадвигунобудуванні, військовій промисловості, приладобудуванні, медицині тощо.

Наукові положення, висновки і рекомендації, викладені в дисертаційній роботі, використано при підготовці дисциплін «Інтелектуальні системи», «Основи обчислювального інтелекту», «Інтелектуальний аналіз даних» та НДР на кафедрі програмних засобів Національного університету «Запорізька політехніка» (акт впровадження від 20.12.2022).

Результати дисертаційної роботи апробовано та впроваджено у практичну діяльність Товариства з обмеженою відповідальністю «Аптека Магнолія» (акт впровадження від 18.08.2022), у практичну діяльність Товариства з обмеженою відповідальністю «Запорізький ливарно-механічний завод» (акт впровадження від 20.09.2022).

**Особистий внесок здобувача.** Всі основні наукові результати дисертаційної роботи отримано автором особисто. Роботу з експериментального дослідження програмних модулів, що реалізують запропоновані у роботі методи синтезу діагностичних моделей, проведено разом із співавторами, прізвища яких наведено у бібліографічному списку. У роботах, написаних зі співавторами, здобувачеві належать такі результати: [120] – модифікований генетичний метод для синтезу рекурентних нейронних мереж; [120], [139] – паралельний генетичний метод з імплементацією механізмів селективного тиску; [151] – система індикаторів для оцінювання рівня складності задачі для подальшого моделювання, діагностування або прогнозування; [173] – метод структурного доналаштування для підвищення характеристик інтерпретовності штучних нейронних мереж; [75] – дослідження ефективності використання методів на основі зворотного поширення помилки для навчання рекурентних нейронних мереж; [120] – аналіз використання нейроеволюційного підходу для рекурентних нейронних мереж; [139] – дослідження методів ройового інтелекту для паралелізації нейроеволюційного синтезу рекурентних нейронних мереж; [173] – аналіз методів структурної оптимізації штучних нейронних мереж; [120], [175]–[179] – експериментальне дослідження модифікованого генетичного методу для синтезу рекурентних нейронних мереж; [120], [182], [183] – експериментальне дослідження паралельного генетичного методу з імплементацією механізмів селективного тиску; [175] – експериментальне дослідження системи індикаторів для оцінювання рівня складності задачі для подальшого моделювання, діагностування або прогнозування; [190] – експериментальне дослідження методу структурного доналаштування для підвищення характеристик нейропластичності штучних нейронних мереж.

**Апробація матеріалів дисертації.** Основні результати дисертаційної роботи доповідалися й обговорювалися на міжнародних конференціях і форумах: 15th International Conference «The experience of Designing and

Application of CAD systems (CADSM)», Polyana (Svalyava), 26 February – 2 March 2019; 2nd International Workshop on «Computer Modeling and Intelligent System»s (CMIS-2019), Zaporizhzhia, Ukraine, April 15-19, 2019; Workshop on Modern Machine Learning Technologies and Data Science (MoML&T&DS-2019), Shatsk, Ukraine, 2-4 June 2019; 2nd International Workshop “Informatics and Data-Driven Medicine (IDDM 2019)”, 11–13 November 2019 (Lviv); 3rd International Workshop on «Computer Modeling and Intelligent Systems» (CMIS-2020), Zaporizhzhia, Ukraine, 27 April – May 1, 2020; International Conference «Problems of Infocommunications. Science and Technology» (PIC S&T 2020); 7th International Conference on «Control and Optimization with Industrial Applications» (COIA 2020), Baku, Azerbaijan, 26-28 August 2020; 10th International Conference on «Advanced Computer Information Technologies» (ACIT 2020), Deggendorf, Germany, 16-18 September 2020; 3rd International Conference on «Informatics & Data-Driven Medicine» (IDDM 2020), Växjö, Sweden, 19-21 October 2020; 2nd International Workshop on «Intelligent Information Technologies and Systems of Information Security» (IntelITSIS'2021), Khmelnytskyi, 24-26 March 2021; 4th International Workshop on «Computer Modeling and Intelligent Systems» (CMIS-2021), Zaporizhzhia, Ukraine, April 27 2021; 4th «International Conference on Informatics & Data-Driven Medicine» (IDDM-2021), Valencia, Spain, 19-21 November 2021; 4th IEEE International Conference on «Advanced Information and Communication Technologies» (AICT-2021), Lviv, Ukraine, 21-25 September 2021; 5th International Workshop on Computer Modeling and Intelligent Systems CMIS-2022), Zaporizhzhia, Ukraine, 12 May 2022.

Матеріали дисертації повною мірою викладені у 30 публікаціях, з яких 25 наукових публікацій розкривають основний зміст дисертації, зокрема опубліковано 8 статей у наукових виданнях, включених до Переліку наукових фахових видань України з технічних наук (7 статей включено у міжнародні наукометричні бази Scopus та/або Web of Science), 1 розділ монографій, що опубліковані у закордонних виданнях мовами країн ЄС.

Опубліковано 21 теза доповідей у матеріалах міжнародних наукових конференцій (15 англійською мовою, включені в міжнародну наукометричну базу Scopus), які засвідчують апробацію матеріалів дисертації, 2 наукові праці, які додатково відображають наукові результати дисертації, зокрема 1 патент України на винахід.

**Структура та обсяг дисертації.** Дисертація складається зі вступу, 4 розділів, висновків, списку використаних джерел, додатків. Загальний обсяг роботи складає 195 сторінок тексту, що містять 2 анотації на 10 сторінках, 26 рисунків, 13 таблиць, список використаних джерел з 202 найменувань на 24 сторінках, 3 додатки на 19 сторінках.

# **1 ОГЛЯД ПРОБЛЕМНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ**

Для підвищення надійності та забезпечення високого рівня безпечності експлуатації складних технічних систем та об'єктів, підвищення якості промислової продукції, вчасного виявлення дефектів і помилок та їх усунення під час роботи технічних та біомедичних систем широкого використання набули інтелектуальні системи діагностування, розроблення яких пов'язано з необхідністю обробки великих масивів даних та синтезу діагностичних моделей [29]–[34].

З метою визначення й обґрунтування базису діагностичних моделей та методів синтезу моделей у розділі наводиться аналіз сучасного стану проблеми діагностування на основі історичних даних. Проведені дослідження різних типів моделей, що можуть використовуватися у якості основи для синтезу діагностичних моделей, та різних підходів до побудови таких моделей дозволили обрати у якості напрямку дослідження розробку нових методів синтезу діагностичних моделей для підвищення якості діагностування на основі даних.

## **1.1 Діагностування на основі історичних даних про об'єкт**

Процес діагностування можна розглядати як контроль технічного стану певної системи [29]. Загалом же, це визначення стану об'єкту з означеною (заданою) точністю. Об'єктом діагностування (контролю стану) є система (або об'єкт) та (або) її складові частини, що підлягають діагностуванню (контролю) [30]. Засобом діагностування виступають апаратура та методи, за допомогою яких здійснюється діагностування. Основні завдання діагностування можна розділити умовно на дві групи [31], [32]:

- 1) визначення стану системи в умовах обмеженої інформації;
- 2) визначення контролепридатності системи.

Слід зазначити, що діагностування складних технічних або біомедичних систем може відбуватися за рахунок діагностування окремих її компонентів (окремих об'єктів, що є складовими системи) або ж діагностування системи в цілому. Неруйнівне діагностування проводиться без розбирання чи руйнування системи, і, найчастіше у реальному часі та під час експлуатації [29], [32]. Через такий підхід отримання інформації вкрай складне. Як правило, за отриманою інформацією зробити однозначний висновок – не можливо і доводиться застосовувати статистичні методи [30]–[35].

Як правило, методи розпізнавання в межах діагностування частково ґрунтуються на діагностичних моделях. Діагностичні моделі – це формалізований опис об'єкту, що проходить діагностування [34]. Моделі встановлюють зв'язок між станами технічної об'єкту та їх відображенням у просторі діагностичних сигналів [35]. Основним ризиком вирішення діагностичної задачі (віднесення виробу до одного із класів, наприклад «справного» чи «несправного») є ризик помилкової тривоги. Тому для винесення обґрунтованого рішення залучають методи теорії статистичних рішень [29]–[35].

Вирішення першої задачі діагностування є малоінформативним без складання прогнозу надійності на найближчий період експлуатації (до наступного огляду) [33]–[37]. В цьому випадку, рішення ґрунтуються на моделях відмов, що вивчаються в теорії надійності [32]–[38].

Вирішення другої задачі діагностування базується на особливій властивості більшості системи: контролепридатності [28]. Контролепридатність відображає можливість виробу забезпечувати достовірну оцінку стану та змогу здійснювати раннє виявлення несправностей і відмов [29]. Вона ґрунтується на конструкції системи та прийнятій системі діагностування. Вирішення задачі контролепридатності

можна звести до вивчення засобів і методів отримання діагностичної інформації [38]–[40].

Обробка діагностичної інформації та формування керуючих сигналів у складних технічних системах виконується через автоматизований контроль стану виробів [38]. Тому методи проектування таких автоматизованих систем контролю складають один з напрямків теорії контролепридатності [39]. До додаткових завдань при вирішенні задачі контролепридатності відносяться [40]:

- розробка алгоритмів пошуку несправностей;
- розробка діагностичних тестів;
- мінімізація тривалості процесу встановлення діагнозу.

Загальна класифікація видів діагностування наведена на рисунку 1.1.



Рисунок 1.1 – Класифікація видів діагностування

Діагностування можна поділити на тестове, функціональне та експрес-діагностування.

Тестове діагностування – це діагностування, при якому об'єкт піддається тестовому впливу (наприклад, визначення ступеня зносу ізоляції



електричних машин за зміною тангенса кута діелектричних втрат при подачі напруги на обмотку двигуна від мосту змінного струму) [34]–[38].

Функціональне діагностування – це діагностування, при якому вимірюються й аналізуються параметри об'єкту при його функціонуванні за прямим призначенням [37], [40] або в спеціальному режимі, наприклад визначення технічного стану підшипників кочення за зміною вібрації під час роботи електричних машин [41]–[43].

Експрес-діагностування – це діагностування за обмеженою кількістю параметрів та протягом заздалегідь визначеного часу [41].

За об'ємом та періодичністю діагностування також поділяють на комплексне, планове та періодичне [43]. Періодичне і планове діагностування дозволяє:

- виконувати початковий контроль агрегатів і запасних вузлів при їх купівлі [29], [30];

- звести до мінімуму раптові позапланові зупинки технічного обладнання [35], [36];

- керувати зношенням обладнання.

Комплексне діагностування технічного стану обладнання дає можливість вирішувати такі завдання [37]:

- проводити ремонт (для технічних систем) або терапію (для біомедичних систем) за фактичним станом [31];

- збільшити середній час між ремонтами або терапією;

- зменшити витрату деталей в процесі експлуатації різного обладнання [32];

- зменшити обсяг запасних частин [29];

- скоротити тривалість ремонту або терапії [30];

- підвищити якість ремонту або терапії [31];

- підвищити безпеку експлуатації [29];

- зменшити ресурсоємності системи [33], [35].

Спираючись на визначення та характеристики видів діагностування визначимо основні задачі та завдання.

Основних задач, зазвичай, виділяють три [29]–[43]:

- контроль поточного стану системи;
- виявлення причин несправності системи;
- складання прогнозу подальшого функціонування.

Для вирішення основних задач діагностування слід розпочати також виділити завдання, кожної із задач. Так для вирішення задачі контролю стану об'єкта діагностування слід [40]:

- 1) визначити (розпізнати, оцінити) стан, у якому перебуває об'єкт на момент діагностування [41];
- 2) надати оцінку відповідності чи невідповідності стану об'єкта діагностування та у разі невідповідності – визначення причини невідповідності: несправність, дефект, відхилення від регулювань тощо [43].

Серед завдань виявлення причин несправності системи виділяється: виявлення ушкоджень чи дефектів на початковій стадії їх розвитку, виявлення конкретних дефектних складових чи деталей, визначення і усунення причин, що викликали дефект [32], [43]–[45].

Для успішного складання прогнозу подальшого функціонування системи слід [40]–[49]:

- 1) оцінити допустимість та доцільність подальшої експлуатації обладнання з урахуванням прогнозування його стану при виявленні дефекту, оптимізувати режим експлуатації так, щоб безпечно експлуатувати агрегат з виявленими дефектами до моменту його виводу у плановий ремонт [41];
- 2) організувати обслуговування та ремонт обладнання за технічним станом (замість регламентного обслуговування і ремонту), забезпечити підготовку та виконання якісних ремонтів [42]–[45].

Для проведення діагностування об'єкту необхідно:

- визначити показники та характеристики діагностування [43];
- забезпечити застосовність об'єкту до діагностування [44];

– реалізувати та ввести діагностичне забезпечення виробу [45]–[49].

Забезпечення пристосованості виробу до технічного діагностування повинно проводитися відповідно до вимог діючих стандартів та рекомендацій [43]. Показники та характеристики діагностування, вимоги щодо придатності об'єкту до діагностування та діагностичне забезпечення повинні включатися до технічного завдання, стандартів конкретних видів продукції та документацію, що розробляється в процесі проведення дослідницько-конструкторських робіт [45]–[50].

Технічні вимоги та рекомендації з проведення технічного діагностування, в першу чергу, залежать від характеристик обладнання [31], [32]–[36], [38]–[42]. При цьому можна виділити умовно загальний порядок контролю поточного стану пристрою:

- 1) аналіз технічної документації;
- 2) функціональна діагностування;
- 3) розробка індивідуальної програми діагностування;
- 4) проведення огляду (зовнішній і внутрішній контроль);
- 5) дослідження обладнання;
- 6) аналіз і обробка результатів діагностування.

Незалежно від типу об'єкта або системи перед діагностуванням потрібно провести підготовчі процедури, які включають в себе відключення, очищення та інші процеси, необхідні для досягнення максимально точного результату [44]–[48].

Основні розбіжності в програмі технічного діагностування починаються в ході випробувань обладнання. Для кожного типу пристроїв визначаються свої методики дослідження і вимірювання [29], [30], [45]–[48].

## 1.2 Діагностичні моделі

Для розв'язання задач діагностування необхідна наявність формального опису (моделі) об'єкту діагностування [49]–[54]. При цьому

множина постановок та вирішення найважливіших завдань діагностування припускають визначення множини допустимих несправностей та їх моделі [55].

Зазвичай під несправністю розуміють деяку модель фізичних дефектів в об'єкті діагностування [56]. Зазвичай, кожній несправності можна задати у відповідність деяку зміну зв'язків або параметрів елементів об'єкту діагностування [57]. Наприклад, часто в якості допустимих розглядаються несправності типу: обриви та короткі замикання, вихід коефіцієнта підсилення підсилювача за межі заданого допуску, наявність постійно рівня логічної одиниці або нуля на виведення цифрової мікросхеми, тощо [58]–[60].

Математична модель об'єкту діагностування може бути задана в явному або неявному вигляді [57]–[61].

Явна модель являє собою сукупність формальних описів справного об'єкта і всіх (точніше, кожної з розглянутих) [58] його несправних модифікацій [61]. Для зручності обробки всі зазначені описи бажано мати в однаковій формі.

Неявна модель містить будь-який один формальний опис об'єкта, математичні моделі його фізичних несправностей та правила отримання за цими даними всіх інших описів, що цікаві для діагностування [59]. Найчастіше заданою є математична модель справного об'єкта, по якій можна побудувати моделі його несправних модифікацій [60].

Загальна вимога до моделей справного об'єкта, а також до моделей несправностей полягає в тому [50], [52], що вони повинні з необхідною точністю описувати представлені ними об'єкти та їх несправності [53], [55]. У неявних моделях моделі об'єктів діагностування несправностей [54], крім того, повинні задовольняти вимогу зручності їх сполучення з наявним описом об'єкту [57] і тим самим забезпечити досить прості правила отримання інших описів об'єкту [58].

Справний або несправний об'єкт може бути представлений як динамічна система, стан якої в кожен момент  $t$  визначається значеннями вхідних, внутрішніх і вихідних координат [50]. Окремим є випадок, коли стан об'єкта не залежить від часу [57], [58].

Узагальнена схема діагностичної моделі наведена на рисунку 1.2.

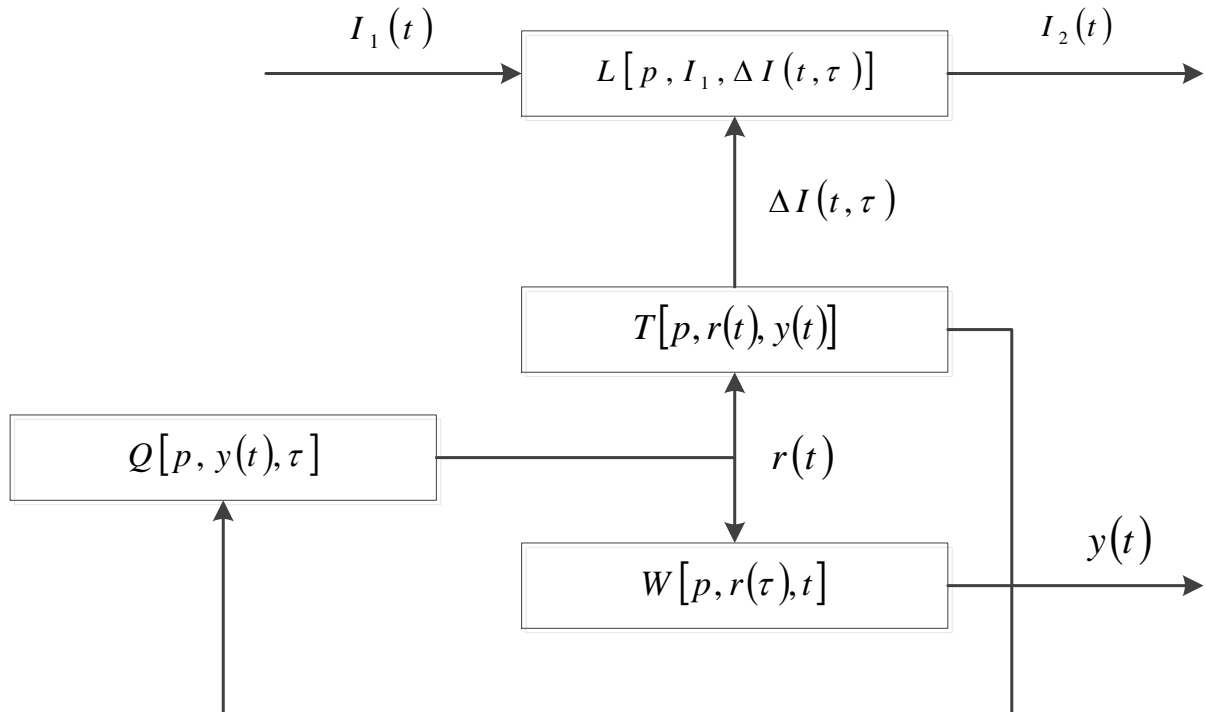


Рисунок 1.2 – Узагальнена схема діагностичної моделі

Основні властивості об'єкту характеризуються оператором  $L$ , який пов'язує вхідні і вихідні впливу  $I_1(t)$  та  $I_2(t)$ , де  $t$  – час, а також враховує залежність  $I_2(t)$  від збуджуючого фактора  $\Delta I(t, \tau)$ , (де  $\tau$  – час), породженого власними внутрішніми процесами [28], [29], [49]–[63]. Якість функціонування об'єкта залежить не тільки від конструктивних параметрів  $p$ , але і від збуджень  $\Delta I(t, \tau)$ , які змінюються в часі і можуть викликати параметричний відмова системи. Зміна технічного стану можна контролювати по зміні власних коливань  $y(t)$ , породжувани внутрішніми процесами [50]. Основним параметром, що зв'язує  $\Delta I(t, \tau)$  та  $y(t)$  в розглянутій моделі є вектор  $r(\tau)$ . Вектор  $r(\tau)$  визначається дефектами механічного вузла. Зв'язок  $\Delta I(t, \tau)$ ,  $r(\tau)$  та  $y(t)$  встановлюється оператором

$T$ , а зв'язок  $r(\tau)$  та  $y(t)$  – оператором  $W$ . Параметр  $r(\tau)$  в умовах тривалого функціонування системи змінюється не тільки в результаті процесів старіння, але і під дією вібрації. Ці зміни за часом значно повільніше в порівнянні з вібрацією і флуктуацією основних експлуатаційних показників [28], [53], [54]. Зв'язок між змінами  $r(\tau)$  і вібрацією  $y(t)$  встановлюється за допомогою оператора  $Q$  [29]. У розглянутій моделі існує два види характерних процесів: швидкі (час  $t$ ) [41] – вібрація і флуктуація експлуатаційних показників і повільні (час  $\tau$ ) – зміна параметрів  $r(\tau)$  (дефектів та інших характеристик). Швидкі процеси визначають якість функціонування в даний момент часу, а повільні-параметричну надійність системи [28], [29], [49]–[63].

Варіантів існуючих моделей для діагностування існує багато [50]–[57]. Аналізуючи переваги і недоліки моделей діагностування слід брати до уваги і відповідні їм методи. Регресійні моделі і методи відрізняються відносно високими рівнями простоти, а також схожості їх аналізу і проектування [60]. При використанні лінійних регресійних моделей результат прогнозування може бути отриманий швидше, ніж при використанні інших моделей [61]. Проте значними недоліками є складність визначення виду функціональної залежності, а також трудомісткість визначення параметрів моделі [62]. Більш того, головні проблеми пов'язані з низьким рівнем адаптивності і відсутністю здатності моделювання нелінійних, багатовимірних, складних систем, об'єктів і процесів [63].

Недоліками авторегресійних моделей і методів є: велика кількість параметрів моделі, ідентифікація яких неоднозначна і ресурсоємна [61]. Більш того, низька адаптивність моделей, а також лінійність і, як наслідок, відсутність здатності моделювання нелінійних процесів – значно обмежують простір практичного використання [60]–[63].

Ідея використання штучних нейронних мереж (ШНМ) для задач діагностики та прогнозування помітно спрощує процес оцінки технічного стану об'єктів [64]. Під ШНМ мають на увазі обчислювальні структури, що складаються з великої кількості однотипних елементів, кожен з яких виконує

відносно прості функції [64]–[67]. Процеси в штучних ШНМ іноді асоціюють з процесами, що відбуваються в нервовій системі живих організмів. Навчити нейронну мережу-значить, повідомити їй, Чого ми від неї добиваємося [65]. Причини, що послужили застосуванню ШНМ в задачах діагностики та прогнозування [66]:

а) для реалізації нейромережових методів необхідна мінімальна інформація про об'єкт [67];

б) при реалізації ШНМ можлива паралельна обробка інформації, що, по-перше, значно збільшує швидкість роботи системи і, по-друге, підвищує надійність системи [68], [69].

Виділяють такі основні можливі напрямки застосування ШНМ.

1. Застосування ШНМ для параметричного діагностування елементів технічних систем [70]–[74]. Він заснований на порівнянні математичної моделі конкретного елемента з моделлю бездефектного елемента [71], тобто в перевірці приналежності параметрів станів допустимим діапазоном їх розсіювання [73]. Вихід параметра за межі цих діапазонів буде свідчити про наявність несправності [74].

2. Застосування ШНМ для прогнозування параметрів елементів технічних систем. ШНМ на основі розробленої певної методики дозволяє побудувати залежність одного параметра від іншого у вигляді полінома [72]. Тобто вона може дозволити знайти приховані залежності, однієї величини від іншої, які неможливо визначити методами прямих вимірювань [74].

3. Застосування нейромережових технологій для прогнозування технічного стану елементів технічних систем [64]–[74]. Здібності ШНМ до прогнозування безпосередньо впливають з її здатності до узагальнення і виділення прихованих залежностей між вхідними і вихідними даними [69]–[73]. Діаграма функціонування діагностичної моделі на основі історичних даних про об'єкт діагностування наведена на рис. 1.3.

Приклад впровадження системи технічного діагностування, де діагностична модель будується на основі ШНМ, наведено на рис. 1.4.

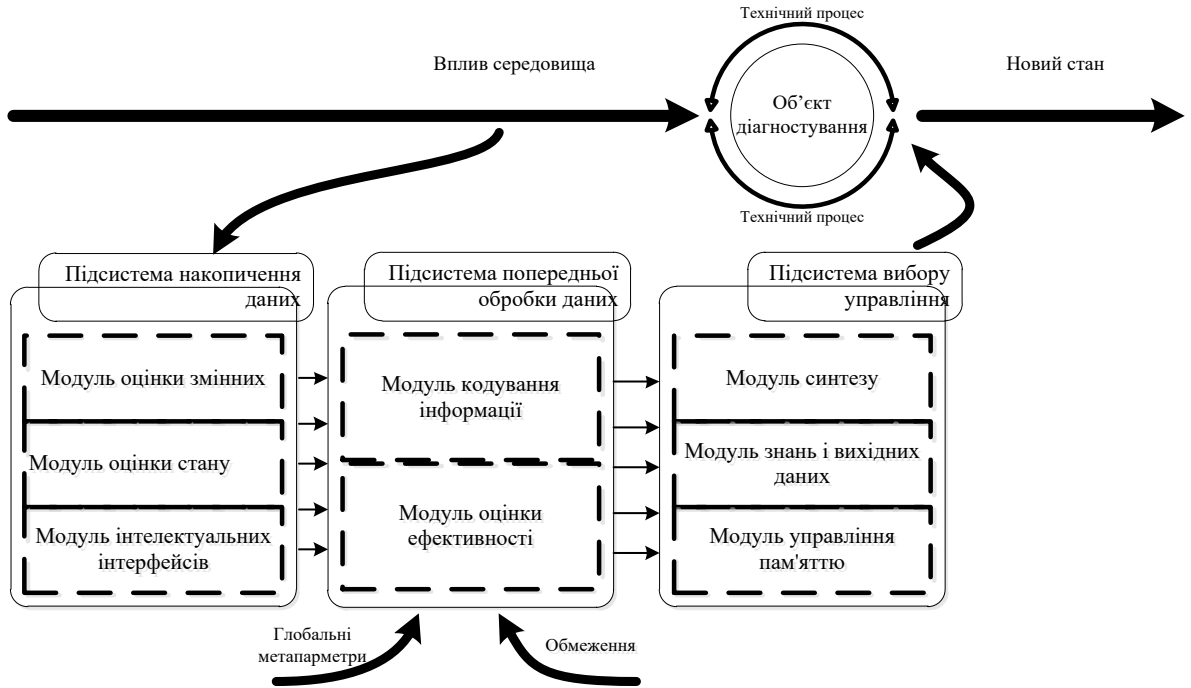


Рисунок 1.3 – Узагальнена діаграма функціонування діагностичної моделі на основі історичних даних

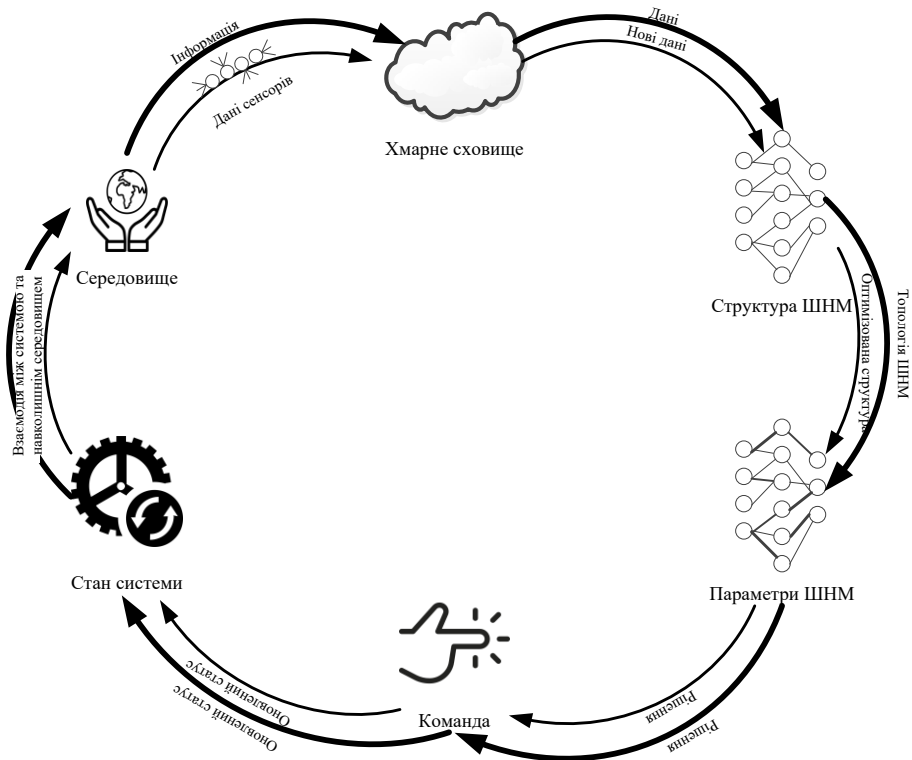


Рисунок 1.4 – Діаграма процесу впровадження системи технічного діагностування з моделлю на базі ШНМ



### 1.3 Постановка завдання

Нехай задано вибірку вхідних даних

$$Sample = \langle X, Y \rangle,$$

де  $X = \{x_1, x_2, \dots, x_n\}$  – множина незалежних змінних – ознак;

$Y = \{y_1, y_2, \dots, y_m\}$  – множина значень залежних змінних;

$n$  – кількість вхідних (незалежних) ознак, що характеризують екземпляри вибірки;

$m$  – кількість вихідних (залежних) ознак.

Спираючись на це, подамо проблему таким чином: для синтезу діагностичної моделі на основі ШНМ –  $NN$  необхідно визначити множину нейронів  $N = \{N_i, N_o, N_h\}$ , що складається із підмножин вхідних  $N_i = \{N_{i_1}, N_{i_2}, \dots, N_{i_l}\}$ ,  $l = 1, 2, \dots, |N_i|$  вихідних  $N_o = \{N_{o_1}, N_{o_2}, \dots, N_{o_p}\}$ ,  $p = 1, 2, \dots, |N_o|$  та прихованих нейронів  $N_h = \{N_{h_1}, N_{h_2}, \dots, N_{h_r}\}$ ,  $r = 1, 2, \dots, |N_h|$  та множину ваг  $w = \{w_k\}$  зв'язків  $c = \{c_1, c_2, \dots, c_k\}$ ,  $k = 1, 2, \dots, |c|$  між нейронами таким чином, щоб:

$$E = \frac{error_{diag}}{Number_{sampl}} \cdot 100\% \rightarrow 0,$$

де  $E$  – помилка моделі;

$error_{diag}$  – помилково продіагностовані екземпляри за моделлю;

$Number_{sampl}$  – загальний розмір вибірки  $Sample$  (кількість її екземплярів).

Визначивши значення елементів множин, можна вважати синтез ШНМ – завершеним.

## 1.4 Аналіз топологій штучних нейронних мереж для діагностування на основі даних

ШНМ є різновидом технології машинного навчання, що забезпечує основу методів навчання [64]. Назва і форма засновані на людському мозку, і реалізація внутрішніх обчислень максимально наближена до взаємодії нейронів, коли вони взаємодіють один з одним [74].

ШНМ складаються з вузлових шарів, кожен з яких має вхідний шар, один або кілька прихованих шарів і вихідний шар [66]. Кожен вузол, або штучний нейрон, пов'язаний з іншим і має свої власні ваги і поріг [65]. Якщо вихідні дані будь-якого конкретного вузла перевищують задане порогове значення, цей вузол активується і починає передачу даних на наступний рівень мережі [68]. В іншому випадку дані не передаються на наступний мережевий рівень [70].

Загалом, ШНМ вчать на прикладах (що наближує процес навчання до процесу пізнання світу). ШНМ налаштовується для конкретного застосування, на кшталт розпізнавання образів або класифікації даних, в процесі навчання. Навчання в значній мірі включає в себе коригування синаптичних зв'язків, що існують між нейронами [64]–[74].

Модель штучної нейронної мережі може бути задана п'ятьма сутностями (рис. 1.5) [74]:

- характером навчання (методами параметричного синтезу);
- характером налаштування внутрішніх взаємозв'язків;
- повнотою між нейронних зв'язків;
- типом вхідної інформації;
- характером топологічної моделі.

Внутрішні взаємозв'язки можуть бути визначені як спосіб, яким обчислювальні елементи (нейрон) в ШНМ з'єднані один з одним [66]. Отже, розташування цих елементів обробки і геометрія з'єднань дуже важливі в мережі [65].

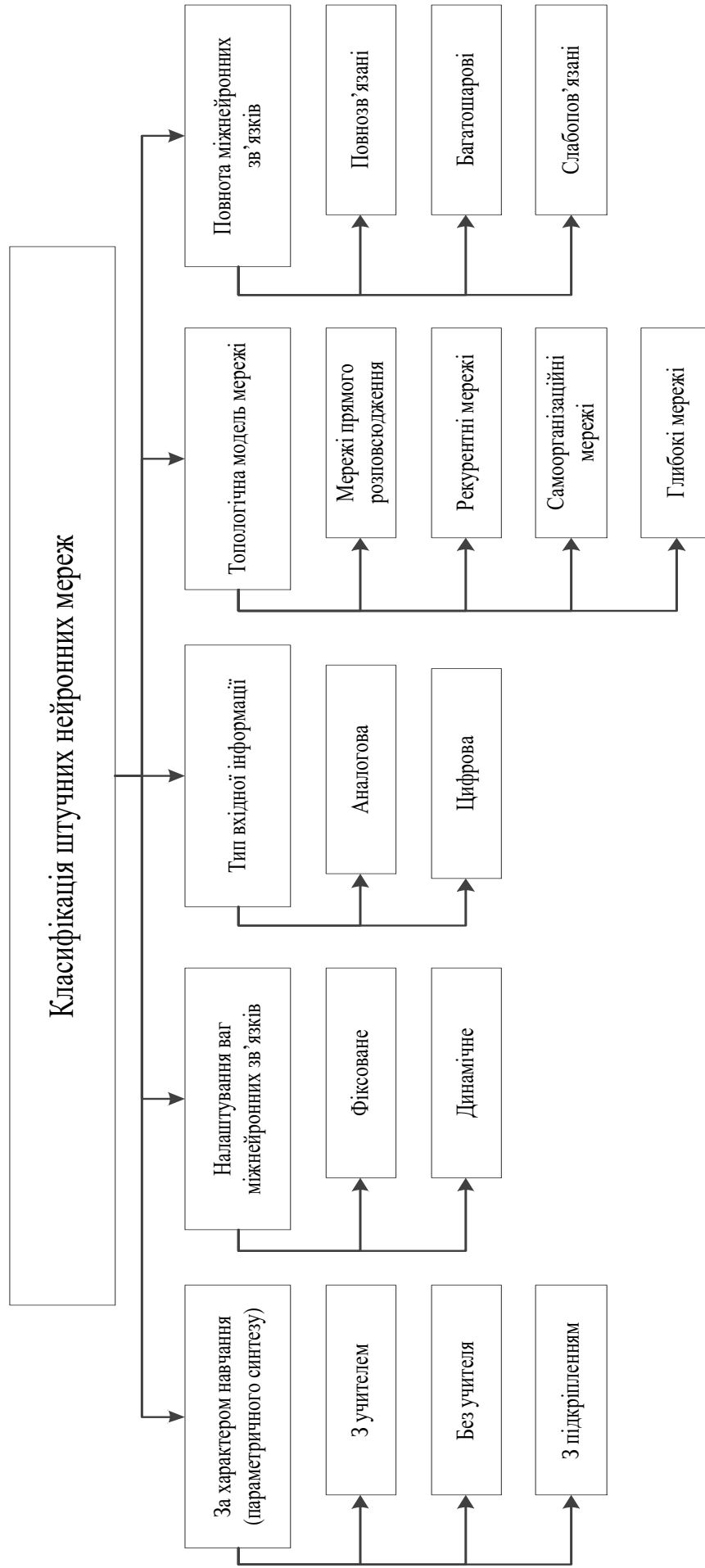


Рисунок 1.5 – Класифікація ШНМ

Ці схеми завжди мають два рівні, які є загальними для всіх мережевих архітектур, вхідний рівень і вихідний рівень, де вхідний рівень буферизує вхідний сигнал, а вихідний рівень генерує вихідні дані мережі [66]. Третій шар – це прихований шар, в якому нейрони не зберігаються ні у вхідному шарі, ні у вихідному шарі [67]. Ці нейрони приховані від дослідників, які взаємодіють з системою, і відпрацьовують для них як чорний ящик. При збільшенні прихованих шарів з нейронами обчислювальна й аналізуюча потужності системи можуть бути збільшені, але в той же час явища навчання системи стають більш складними [69].

Топологічно можна розділити усі існуючі структури ШНМ на дві великі групи: неглибокі ШНМ (ШНМ в яких є один прихований шар нейронів) та глибокі ШНМ (ГНМ) (в яких кількість прихованих шарів теоретично не обмежена) [73]. А вже серед них можна виділити найбільш відомі топології ШНМ, як наведено на рис. 1.6. Проте варто зазначити, що і в рекурентних ШНМ, і в звичайних перцептронах теоретично кількість прихованих шарів не обмежена [72].

ШНМ прямого поширення – є найпростішою формою ШНМ, в якій вхідні дані переміщуються тільки в одному напрямку, проходячи через штучні нейронні вузли і виходячи через вихідні вузли [70], [71]. В такій мережі приховані шари можуть бути присутніми, а можуть і не бути присутнім – присутні вхідні і вихідні шари. Виходячи з цього, вони можуть бути додатково класифіковані як одношарові або багатшарові ШНМ з прямим зв'язком [73].

Кількість шарів залежить від складності функції. Така топологія має односпрямоване пряме поширення, але не має зворотного поширення. Ваги тут статичні [64]–[70]. Функція активації подається вхідними даними, які помножаються на вагові коефіцієнти. Для цього використовується функція активації класифікації або функція покрокової активації [66]–[71]. Тобто:

– нейрон активується, якщо він перевищує порогове значення (зазвичай 0), і нейрон видає 1 в якості вихідного сигналу [65];

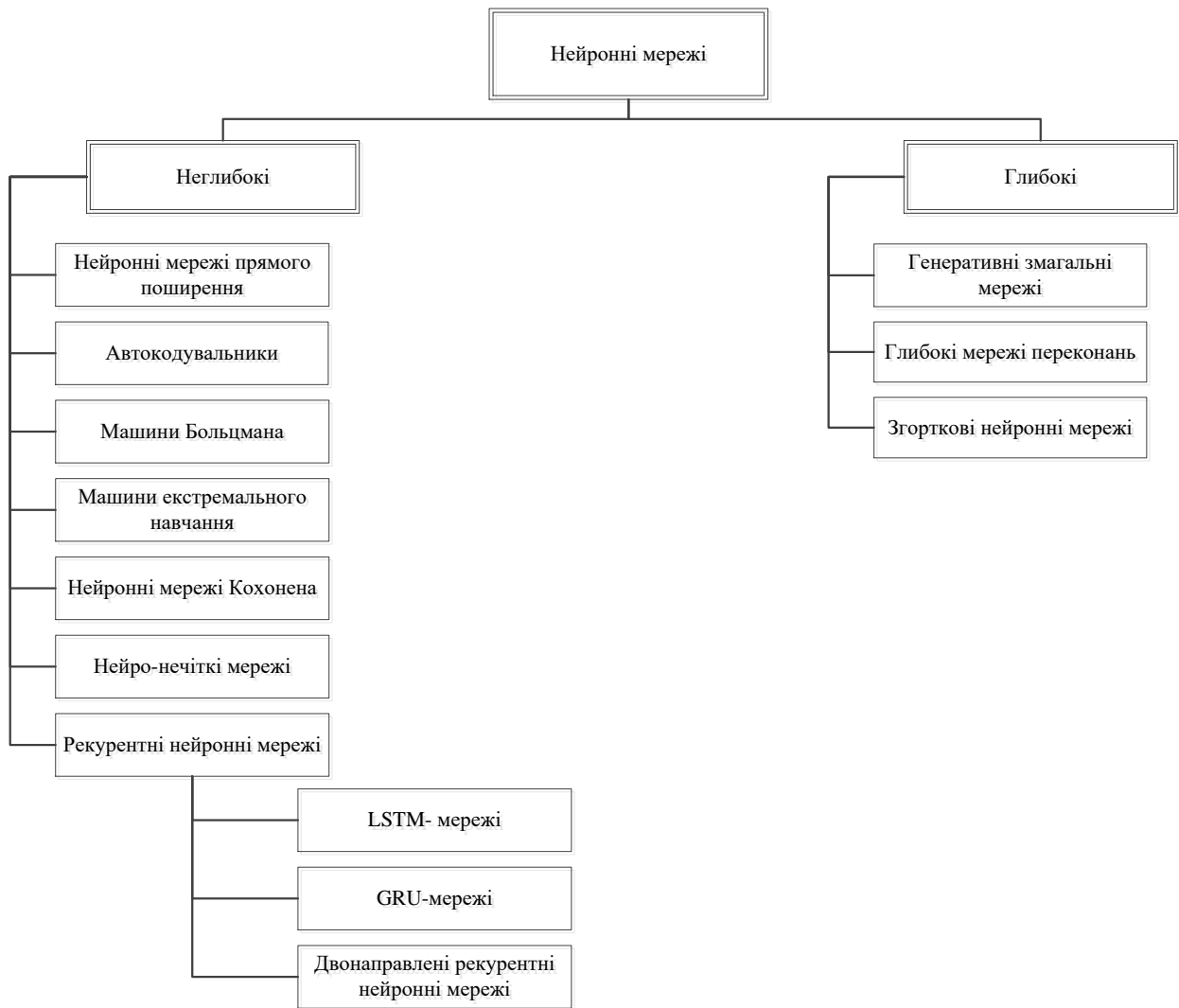


Рисунок 1.6 – Класифікація топологій ШНМ

– нейрон не активується, якщо він знаходиться нижче порогового значення (зазвичай 0), яке вважається рівним  $-1$  [67].

Такі мережі досить прості в підтримці (навчанні та подальшому впровадженні), але можуть мати певні складнощі при роботі з даними, що містять багато шуму [68].

Відносна простота у проектуванні та обслуговуванні є значною перевагою таких нейромоделей. Більш того, такі мережі відрізняються високою швидкістю (виключно одностороннє розповсюдження) [69].

Проте їх чутливість до зашумлених даних є одним з головних недоліків, адже значно знижує точність роботи з такими даними та вимагає додаткових обчислювальних операцій над даними [70].

Персептрон є окремих випадком нейронних мереж прямого поширення. Модель персептрона є однією з найпростіших і найстаріших моделей ШНМ [74]. Така ШНМ виконує певні обчислення для виявлення функцій або бізнес-аналітики у вхідних даних. Модель приймає зважені вхідні дані і застосовує функцію активації для отримання вихідних даних в якості кінцевого результату. Іноді таку топологію позначають, як пороговий логічний пристрій [74].

Персептрон підтримує методи навчання з учителем, які класифікують дані на дві категорії, таким чином, це двійковий класифікатор. Персептрон розділяє вхідний простір на дві категорії гіперплощиною [65].

Персептрони можуть реалізовувати логічні елементи типу AND, OR або NAND, що розширює коло їх практичних задач та є важливою перевагою в порівнянні з більш простими моделями.

Серед недоліків персептрона зазвичай можна виділити навчання виключно для задач, що лінійно розділяються, такі як булеві AND завдання. Для нелінійних завдань, таких як булева задача XOR, це не працює [64]–[70].

Багатошарові ШНМ або ж ГНМ. Початковий етап серед складних ШНМ, де вхідні дані проходять через різні шари штучних нейронів. Кожен окремий вузол з'єднаний з усіма нейронами наступного рівня, що робить її повністю зв'язаною нейронною мережею [65]–[70]. Присутні вхідні і вихідні шари, що мають кілька прихованих шарів, тобто в цілому не менше трьох або більше шарів.

Вхідні дані множаться на вагові коефіцієнти і подаються в функцію активації, а при зворотному поширенні вони модифікуються для зменшення втрат [67]. Вони самостійно налаштовуються залежно від різниці між прогнозованими результатами та вхідними даними навчання. Використовуються нелінійні функції активації, за якими слідує функція активації вихідного шару [72].

Перевагою ГНМ є висока точність, адже використання великої кількості прихованих шарів обчислювальних вузлів здатне забезпечити високу точність при складних вхідних даних.

З іншого боку, саме велика кількість прихованих шарів значно ускладнює проектування та подальше використання таких ШНМ, що є значним недоліком. Також одним з недоліків є відносна повільність, що звісно залежить від кількості прихованих шарів, проте зазвичай значно відрізняється від одношарових ШНМ.

Згорткова нейронна мережа (Convolutional neural network – CNN) – є окремим прикладом ГНМ. CNN містить тривимірне розташування нейронів замість стандартного двовимірного масиву [73]. Перший шар називається згортковим шаром. Кожен нейрон в згортковому шарі обробляє інформацію тільки з невеликої частини поля зору. Вхідні об'єкти відбираються пакетно, як фільтр. Мережа отримує дані по частинах і може обчислювати ці операції кілька разів, щоб завершити повну обробку масиву даних [70].

Поширення є односпрямованим, де згорткова нейронна мережа містить один або кілька згорткових шарів. Фільтри використовуються для вилучення певних частин даних [71]. У звичайних ГНМ вхідні дані множаться на ваги і передаються в функцію активації [72]. Згортка ж використовує ReLU [73]. CNN показують дуже ефективні результати в розпізнаванні зображень і відео, тож можуть використовуватися для обробки даних, наприклад, фотосенсорів, які встановлюються для моніторингу стану складної системи.

Серед основних переваг CNN виділяють можливість глибокого навчання з невеликою кількістю параметрів, через меншу кількість параметрів для навчання в порівнянні з повністю зв'язним шаром.

До недоліків же відносять більшу складність в проектуванні і обслуговуванні в порівнянні зі звичайними ГНМ. Також CNN демонструють порівняну повільність, що залежить від кількості прихованих шарів.

Нейронні мережі з радіальною базисною функцією (Radial basis function – RBF) [65]. Мережа RBF складається з вхідного вектора, за яким – шар нейронів RBF, і вихідний шар з одним вузлом на категорію [67]. Класифікація виконується шляхом вимірювання подібності вхідних даних з точками даних з навчального набору, де кожен нейрон зберігає прототип [69].

Коли необхідно класифікувати новий вхідний вектор ( $n$ -мірний вектор, який намагаються класифікувати), кожен нейрон обчислює евклідову відстань між входом і його прототипом [71]. Наприклад, якщо є два класи, тобто клас  $A$  і клас  $B$ , то нові вхідні дані, що підлягають класифікації, ближчі до прототипів класу  $A$ , ніж прототипи класу  $B$ . Отже, він може бути позначений або класифікований як клас  $A$ .

Кожен нейрон RBF порівнює вхідний вектор зі своїм прототипом і видає значення в діапазоні, який є мірою подібності від 0 до 1 [68]. Оскільки вхідний сигнал дорівнює прототипу, вихідний сигнал цього нейрона RBF буде дорівнює 1, а зі збільшенням відстані між входом і прототипом відгук експоненціально падає до 0 [69]. Крива, що генерується в результаті реакції нейрона, має тенденцію до типової кривої нормального розподілу. Вихідний шар складається з набору нейронів (по одному на категорію) [72].

Рекурентні нейронні мережі (РНМ) [75]. Розроблена для збереження вихідних даних шару, РНМ повертається назад на вхідні дані, щоб допомогти в прогнозуванні результату шару [73]. Перший рівень зазвичай являє собою нейронну мережу з прямим зв'язком, за якою йде шар РНМ, де деяка інформація, отримана на попередньому тимчасовому кроці, запам'ятовується функцією пам'яті. В цьому випадку реалізується пряме поширення. У ньому зберігається інформація, необхідна для його подальшого використання. Якщо прогноз неправильний, швидкість навчання використовується для внесення невеликих змін [75]. Отже, змушуючи його поступово збільшуватися, щоб зробити правильний прогноз під час зворотного поширення [73], [75].



Головною перевагою РНМ є моделювання залежності послідовності вхідних даних про об'єкт від попередніх станів об'єкту. Певним розвитком ідеї РНМ є можливість використання гібридних топологій з шарами згортки для підвищення точності.

Основні ж недоліки використання РНМ базуються на проблемах їх синтезу та навчання, як то проблеми зі зникненням і вибухом градієнта. Також сам процес навчання РНМ може виявитися занадто складним та ресурсоємним.

Розвиток ідеї РНМ – мережі довготривалої короткочасної пам'яті (Long short-term memory – LSTM) [75]. Мережі LSTM – це тип РНМ, який використовує спеціальні блоки на додаток до стандартних блоків. Блоки LSTM включають в себе комірку пам'яті, яка може зберігати інформацію в пам'яті протягом тривалих періодів часу [68]. Набір елементів використовується для управління процесами: коли інформація надходить в пам'ять, коли вона виводиться і коли вона стирається – забувається. Існує три типи елементів-вентилів, а саме: вхідний вентиль, вихідний вентиль і вентиль забуття [75]. Вхідний елемент визначає, скільки інформації з останньої вибірки буде збережено в пам'яті; вихідний елемент регулює обсяг даних, переданих на наступний рівень, а елементи забування керують швидкістю розриву збереженої пам'яті [70]. Ця архітектура дозволяє їм вивчати довгострокові залежності [75].

Модель Sequence to sequence [72] складається з двох РНМ. Тут існує кодер, який обробляє вхідні дані, і декодер, який обробляє вихідні дані. Кодер і декодер працюють одночасно: або використовуючи один і той же параметр, або різні. Ця модель, на відміну від фактичної РНМ, особливо застосовна в тих випадках, коли довжина вхідних даних дорівнює довжині вихідних даних [71]. Хоча вони мають ті ж переваги та обмеження, що і РНМ, ці моделі зазвичай застосовуються в основному в прикладних системах для бізнесу та комерції.

Модулярна нейронна мережа складається з декількох різних мереж, які функціонують незалежно і виконують під задачі [73], [75]. Різні мережі насправді не взаємодіють і не сигналізують один одному під час процесу обчислень. Вони працюють незалежно один від одного для досягнення результату [68].

В результаті великий і складний обчислювальний процес виконується значно швидше, якщо розбити його на незалежні компоненти. Швидкість обчислень збільшується, тому що мережі не взаємодіють і навіть не пов'язані один з одним [67].

Перевагами модулярних нейронних мереж є адаптованість до самостійного навчання та надійність на випадок втрати нейронів.

Недоліки ж модулярних нейронних мереж є похідними від власне механізмів самостійного навчання, а саме – проблеми зі зміною мети під час навчання.

У таблиці 1.1 наведено порівняння можна зробити висновок, що більш сучасніші топології ШНМ, наприклад такі, як РНМ (включно із ускладненими топологіями LSTM), а також згорткові нейронні мережі – відрізняються більшою структурною складністю, більш того, враховуючи це, процес їх побудови є більш ресурсоємним.

З іншого боку, завдяки структурним особливостям такі нейромережі мають значно більші кола практичного використання для різних предметних областей. Також саме через структурні особливості результати їх використання свідчать про більшу точність.

Саме тому, виходячи із того, що технічне діагностування потребує високого рівня точності, часто пов'язано із забезпеченням людської безпеки, а також відрізняється маловивченими та іноді надмірними вхідними параметрами про об'єкт діагностування, використання більш складних, але точних топологій ШНМ є необхідним.

Таблиця 1.1 – Порівняння найбільш розповсюджених топологій ШНМ

Критерій	Топології нейронних мереж			
	ШНМ прямого поширення	ГНМ	CNN	РНМ
Коло задач, де може бути імплементована нейромодель	Вузьке	Найширше	Ширше	Найширше
Робота із маловивченими предметними областями	Низький рівень	Високий рівень	Середній рівень	Високий рівень
Рівень узагальнення даних	Низький	Середній	Високий	Високий
Обчислювальна складність процесу побудови нейромоделі	Низька	Висока	Висока	Висока
Структурна складність побудованої моделі	Низька	Середня	Найвища	Висока

При порівнянні ж конкретно РНМ та загорткових нейронних мереж перевагу було надано само РНМ через їх простішу структурну будову та більш високий рівень контрольованості розростання мережі під час побудови.

### 1.5 Методи синтезу штучних нейронних мереж

Синтез ШНМ (структурний синтез – визначення топології, оцінка кількості прихованих нейронів, необхідних для вирішення завдань та параметричний синтез – визначення вагових коефіцієнтів міжнейронних зв'язків) – є однією з ключових проблем в полі використання ШНМ [76]–[82].

Як попередньо зазначалося, області застосування ШНМ є дуже широкими, починаючи з загального аналізу даних і закінчуючи конкретними прикладними завданнями машинобудування, біології, медицини, фінансах,

інженерії і т.п. [64]. При цьому важливо зазначити, що при вирішенні прикладних завдань можна використовувати різні види архітектур ШНМ з деякими відповідним чином підібраними функціями активації [77]. Якщо побудована модель ШНМ не здатна вирішити конкретну проблему або забезпечити прийнятну точність, можна збільшити кількість прихованих нейронів в шарах і/або збільшувати кількість самих прихованих шарів в мережевій моделі [78]. Однак при параметричному синтезі необхідно чітко визначити та вказати, скільки нейронів і скільки шарів слід використовувати в архітектурі нейронної мережі. Звичайно, чим більша кількість нейронів і прихованих шарів, тим більша ймовірність того, що мережа дасть більш точні результати [80]. На жаль, практичність зменшується зі збільшенням кількості нейронів і кількості прихованих шарів в моделі ШНМ [83].

Дослідники ШНМ пропонували і експериментували з різними типами моделей нейронних мереж, що використовують багаторівневі архітектури, наприклад, багат шарові перцептрони, згорткові нейронні мережі, звичайні глибокі нейронні мережі, рекурентні нейронні мережі та ін. [84].

Однак, з теоретичної точки зору, проблеми порогових меж нейронів в прихованих шарах і меж прихованих шарів для архітектур класифікації і розпізнавання образів ще не повністю вивчені [85]. Більш того, часто певні експериментальні висновки обмежені конкретними ситуаціями і не дають фундаментальної теорії для вивчення обчислювальних можливостей нейронної мережі в цілому [81]. Крім того, неможливо будь-яким чином поширити ці існуючі результати на усі можливі топології нейронних мереж [84].

Параметричний синтез ШНМ пов'язаний саме з визначенням вагових коефіцієнтів ШНМ [77]–[83]. Зазвичай для виконання цього етапу синтезу використовуються методи машинного навчання. Під час використання нейромоделей для технічного діагностування на основі реальних експериментальних даних про технічну систему найчастіше використовуються методи машинного навчання з учителем [78]. Навчання з

учителем – це тип машинного навчання, при якому система навчається з використанням добре позначених навчальних даних, і на основі цих даних система вчиться прогнозувати результат [79]. Позначені дані – означає, що деякі вхідні дані вже позначені правильними вихідними даними [80]. Під час навчання тренувальні дані, надані системам, працюють як вчитель, який навчає систему правильно прогнозувати вихідні дані [82]. Загалом, це процес надання вхідних даних, а також правильних вихідних даних для моделі машинного навчання [83]. Мета процесу контрольованого навчання полягає в тому, щоб знайти функцію зіставлення для зіставлення вхідної змінної з вихідною змінною [76]–[85].

Під час навчання з учителем моделі навчаються за допомогою позначеного набору даних, де модель вивчає кожен тип даних. Як тільки процес навчання завершений, модель тестується на основі тестових даних (підмножина навчального набору), а потім прогнозує вихідні дані [77]–[83].

Зворотне поширення в часі (ЗПЧ) – це один з найпоширеніших методів навчання з учителем для навчання (параметричного синтезу) саме РНМ [86]–[94]. Метод є продовженням класичного методу зворотного поширення помилки на основі градієнту для навчання більш простих топологій ШНМ.

Навчальні дані (вхідні дані) обов'язково мають бути подані у вигляді впорядкованої послідовності із пар вхідне значення – вихідне значення, як то  $\langle a_0, y_0 \rangle, \langle a_1, y_1 \rangle, \langle a_2, y_2 \rangle, \dots, \langle a_{k-1}, y_{k-1} \rangle$ , а для  $x_0$  мусить бути вказано початкове значення. Для цього зазвичай застосовують тотожний вектор, де усі значення – нулі [86].

Робота методу починається з розгортання РНМ в часі, за допомогою представлення її через звичайні нейромережі прямого поширення [87]. Тобто РНМ містить певну кількість звичайних ШНМ прямого поширення,  $f$  та  $g$ , таким чином після розгортання рекурентна мережа буде містити  $k$  примірників  $f$ , і один примірник  $g$  [88]–[90]. Після чого саме навчання відбувається подібним чином, як і при навчанні звичайної ШНМ з

використанням зворотного поширення помилки. Головною відмінністю в цьому процесі буде те, що навчальні зразки представляються протягом процесу послідовно: так кожен навчальний зразок з вибірки складається з  $\langle x_t, a_t, a_{t+1}, a_{t+2}, \dots, a_{t+k-1}, y_{t+k} \rangle$  [89]. Після подання зразка для навчання уточнюються вагові коефіцієнти у кожному з примірників  $f(f_1, f_2, \dots, f_k)$  підсумовуються, і потім застосовуються до всіх примірників  $f$  [90].

У якості протиставлення навчанню за попередньо спроектованою структурою ШНМ використовується нейроеволюційний підхід [91]–[94] – порівняння цих підходів наведено в таблиці 1.2. Нейроеволюція – форма машинного навчання, яка використовує еволюційні методи для навчання ШНМ [95]. Цей підхід використовується в багатьох галузях: медицина, розробка інтелектуальних систем цивільного та військового призначення, ігри, навчання та управління автоматизованими агентами та роботами [96]. У цих випадках досить просто виміряти продуктивність ШНМ, в той час як реалізувати навчання з учителем дуже важко або практично неможливо [94]. Виконаємо порівняння підходів до синтезу ШНМ.

Таблиця 1.2 – Порівняння різних підходів до синтезу ШНМ

Критерій	Підхід до синтезу ШНМ	
	Навчання за структурою	Синтез структури та параметрів
Маштабованість	Низький рівень	Високий рівень
Адаптивність	Низький рівень	Високий рівень
Залежність від якості вхідних даних	Висока залежність	Середній рівень
Залежність від пошукового простору	Висока залежність	Середній рівень
Ітеративність	Висока	Висока (є можливість зменшення)

Нейроеволюційний синтез нейромоделі із оптимальними структурою та точністю може бути наведено у вигляді пошуку такої архітектури, яка забезпечує найкраще (щодо обраного критерію) рішення конкретного завдання та налаштування параметрів мережі, які забезпечать найкращу точність [96]. Такий підхід передбачає перебір простору архітектури, складеного з усіх можливих варіантів, і вибір точки цього простору, найкращої щодо заданого критерію оптимальності. Після цього налаштовуються параметри (ваги міжнейронних зв'язків) для забезпечення прийнятної точності роботи мережі [97]–[100].

З урахуванням переваг еволюційного проектування архітектури в останні роки було виконано велику кількість досліджень, в яких основна увага приділялася оптимізації структури нейронної мережі: вибору оптимального числа нейронів в «прихованих» шарах мережі і топології зв'язків між ними [97]–[107]. В якості критерію оптимальності розглядаються два критерії: мінімізація помилки навчання на контрольній групі і мінімізація складності архітектури нейронної мережі [100].

Перший етап еволюційного проектування архітектури полягає у прийнятті рішення щодо відповідної форми її опису [98]. Повна інформація про архітектуру може безпосередньо кодуватися у вигляді двійкових послідовностей, тобто кожен зв'язок і кожен вузол (нейрон) прямо специфікується певною кількістю бітів [101]. Схема безпосереднього кодування архітектури ШНМ називається прямим кодуванням. Інший підхід полягає в кодуванні певних правил або законів побудови топології ШНМ. Схема кодування правил побудови архітектури називається непрямим кодуванням [103], [105].

Другий етап, відповідно до типового циклу еволюційного проектування архітектури нейронної мережі, складається з наступних кроків:

а) декодування кожної особини поточної популяції для опису архітектури ШНМ [95];

- б) навчання кожної нейронної мережі з архітектурою, отриманою на першому кроці [96];
- в) оцінювання пристосованості кожної особини (закодованої архітектури) за досягнутими результатами навчання [98];
- г) репродукція особин з ймовірністю, що відповідає їх пристосованості або рангу в залежності від використовуваного методу селекції [97];
- д) формування нового покоління в результаті застосування таких генетичних операторів, як схрещування, мутація і інверсія [100].

Вибір топології та налаштування ваг зв'язків ШНМ є найважливішими етапами при використанні нейромережових технологій для вирішення практичних завдань [64]–[75]. Від цих етапів безпосередньо залежить якість і адекватність отриманої нейромережової моделі. Використання еволюційних методів спільно з ШНМ дозволяє вирішувати завдання налаштування і навчання ШНМ як окремо, так і одночасно (рис. 1.7) [95]–[106]. Такий синтезований підхід надає уніфікований підхід до вирішення різноманітних завдань класифікації, апроксимації, управління та моделювання. Використання якісної оцінки функціонування ШНМ дозволяє застосовувати нейроеволюційні методи для вирішення завдань дослідження адаптивної поведінки інтелектуальних агентів [96], пошуку ігрових стратегій, обробки сигналів і зображень [100]. Незважаючи на те, що кількість проблем і питань, що стосуються розробки і застосування методів велике, для успішного вирішення завдання з використанням нейроеволюційного підходу досить адекватного розуміння проблеми і не підходу, свідченням чого є велика кількість цікавих і успішних робіт в даному напрямку [107].

Нейроеволюційні методи, як попередньо зазначалося, ґрунтуються на застосуванні еволюційних алгоритмів. В якості гіперпараметрів при цьому виступають різні характеристики мережі, такі, як ваги зв'язків, топологія, методи оптимізації, кількість для оптимізації гіперпараметрів, таких як ваги, широко застосовуються еволюційні методи – генетичні алгоритми (ГА)



[108]–[115], Covariance matrix adaptation evolution strategy (CMA-ES) [107]–[112], що дозволяє вирішувати завдання глобальної оптимізації, оскільки може враховувати взаємодії між параметрами.

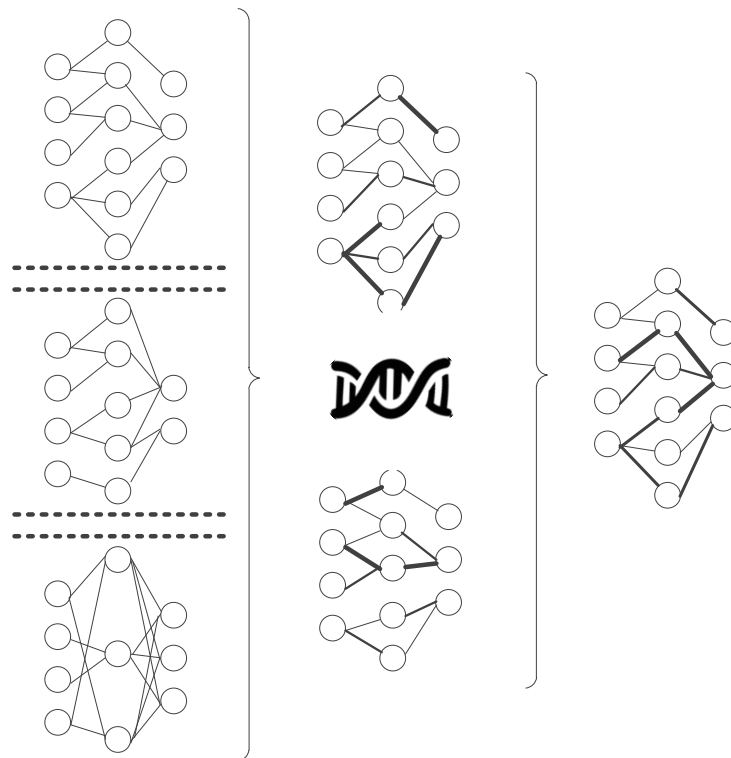


Рисунок 1.7 – Одночасний та поступовий процес визначення структури та параметрів ШНМ при нейроеволюційному синтезі

Такі підходи, як Computational and Systems Neuroscience (CoSyNE) [102]–[110], The Enforced Subpopulations (ESP) [102]–[114], Symbiotic Adaptive Neuro-Evolution (SANE) [112], дозволяють застосовувати еволюційні методи для еволюції блоків і шарів мереж з метою подальшої їх комбінації в повну ШНМ. Інші методи, такі як NeuroEvolution of Augmenting Topologies (NEAT) [114]–[119] і клітинне кодування, застосовуються для еволюції топологій ШНМ, з використанням якої можна значно підвищити ефективність мережі.

Всі перераховані вище технології [95]–[119] застосовуються для вирішення самих різних завдань, але через велику розмірності простору гіперпараметрів, їх можливості обмежуються тільки мережами прямого поширення з невеликою кількістю шарів [100]–[107]. Для підвищення

продуктивності еволюційні методи можуть бути сумісні з градієнтними методами для налаштування гіперпараметрів глибоких нейромереж. У таких методах градієнтні методи зазвичай використовуються для оптимізації збіжності пошуку оптимальних гіперпараметрів. Багато методів працюють за рахунок того, що навчання відбувається тільки у відібраних об'єктах популяції [108].

Застосування нейроеволюції для глибоких мереж трохи відходить від описаних методів [111]–[113]. У мережах, що застосовуються на практиці, градієнти для всіх шарів доступні і обчислювані за адекватний час і застосовуються для оптимізації топології. Глибока нейроеволюція є багаторівневим еволюційним процесом [113]. На різних рівнях оптимізуються гіперпараметри різних типів. Наприклад, при дворівневій нейроеволюції на верхньому рівні оптимізується топологія і зв'язаність блоків або шарів мережі. На нижньому рівні оптимізуються самі блоки. Необхідність застосування такого підходу впливає природним чином з постановок завдань, для вирішення яких застосовуються глибокі мережі [113].

Порівняння нейроеволюційних методів синтезу діагностичних моделей подано у таблиці 1.3.

На підставі висвітлених вище властивостей методів синтезу ШНМ більшість існуючих на сьогоднішній день методів є певним розвитком та модифікацією традиційних нейроеволюційних методів, що дозволяє зробити висновки про їх недоліки та обмеження (табл. 1.3):

а) більшість методів накладає обмеження на структуру (початкову топологію та структуру на основі обраної топології) ШНМ;

б) переважна більшість методів не модифікує вид функції активації і її параметри;

в) у багатьох методах еволюція спрямована виключно по шляху ускладнення (в деяких випадках – послідовного спрощення) структури;

г) деякі методи використовують підхід навчання з учителем, що вимагає наявності репрезентативної вибірки прецедентів і додаткових обмежень на структуру ШНМ.

Таблиця 1.3 – Порівняння нейроеволюційних методів синтезу ШНМ

Критерій	Нейроеволюційні методи синтезу діагностичних моделей			
	SANE	ESP	NEAT	EANT
Схема кодування інформації про нейромодель	Пряма	Пряма	Пряма	Гібридна схема
Еволюція топології	Так	Ні	Так	Так
Обмеження за топологією	Мережі прямого поширення (тільки 1 шар)	Повнозв'язні мережі	Повнозв'язні мережі, фіксована кількість нейронів	Немає
Еволюція параметрів	Так	Так	Так	Так
Параметри, що еволюціонують	Ваги	Ваги	Ваги	Ваги
Обмеження за параметрами	Функція активації	Функція активації	Немає	Немає

## 1.6 Висновки за розділом 1

У цьому розділі розглянуто стан проблеми діагностування на основі історичних даних про об'єкт. Надано постановку завдання розроблення методів синтезу рекурентних нейромережевих моделей для діагностування.

Проаналізовано різні типи ШНМ, які можуть використовуватися як основа для діагностичних моделей при вирішенні завдань діагностування

[75]. В якості основи для синтезу діагностичних моделей запропоновано використовувати ШНМ, які характеризуються порівняно високими апроксимуючими властивостями, що забезпечує більш високий рівень моделювання складних нелінійних систем для їх подальшого діагностування, що значно розширяє коло задач діагностування на основі таких моделей [75]. Проаналізовано різні види нейронних мереж. Встановлено, що в якості діагностичних моделей на основі нейронних мереж ефективно можуть застосовуватися РНМ, що являють собою ШНМ із зворотними зв'язками для фіксації попередніх станів об'єктів, що значно підвищує точність та узагальнюючі властивості і дозволяє їх використовувати для дослідження та представлення складних нелінійних залежностей різних вибірок даних [75].

Проаналізовано процес і методи синтезу діагностичних моделей на основі ШНМ. Визначено, що для синтезу РНМ можуть ефективно застосовуватися методи нейроеволюційного підходу, які одночасно забезпечують виконання і структурного синтезу, і параметричного синтезу РНМ. Досліджено, що суттєвими недоліками існуючих нейроеволюційних методів є: обмеженість початкових топологій нейромереж, неможливість еволюційної модифікації виду функції активації, виключно вертикальна еволюція по шляху ускладнення структури мережі та в деяких випадках жорсткі вимоги до рівня репрезентативності вибірки даних.

Обґрунтовано необхідність розробки нових методів синтезу діагностичних моделей на основі рекурентних нейронних мереж, що забезпечать одночасний синтез структури та параметрів нейромережі, що буде відрізнятися високою точністю, а подальше використання паралельних обчислень дозволить прискорити процес нейроеволюційного синтезу за рахунок використання високопродуктивних методів, які відрізняються скороченням часу синтезу навіть на великих вибірках історичних даних при побудові систем діагностування.

Метою дисертаційної роботи є розроблення та дослідження методів синтезу рекурентних нейромережевих моделей для діагностування, які

поєднують принципи нейроеволюції, інтелектуальних та паралельних обчислень для підвищення точності нейромоделей та збільшення швидкості процесу синтезу.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

- проаналізувати існуючі методи синтезу рекурентних нейронних мереж;
- розробити метод синтезу рекурентних нейронних мереж на основі нейроеволюційного підходу;
- розробити паралельний метод синтезу нейромоделей на основі рекурентних нейронних мереж;
- розробити індикаторну систему для визначення механізмів доналаштування штучних нейронних мереж;
- розробити метод структурної оптимізації нейромоделей;
- провести експериментальне дослідження розроблених методів синтезу та доналаштування діагностичних моделей.

Таким чином, у роботі розв'язується науково-прикладна проблема розроблення та дослідження методів синтезу рекурентних нейромережевих моделей для діагностування, які поєднують принципи нейроеволюційних, інтелектуальних та паралельних обчислень, що дозволяє підвищити швидкість процесу побудови діагностичних моделей, збільшити їх точність та узагальнюючі можливості.

Результати, наведені у першому розділі, було опубліковано в роботі [75].

## 2 СИНТЕЗ РЕКУРЕНТНИХ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ НА ОСНОВІ НЕЙРОЕВОЛЮЦІЙНОГО ПІДХОДУ

В першому розділі було проаналізовано сучасний стан проблеми діагностування на основі даних. У якості основи для побудови діагностичних моделей доцільно використовувати ШНМ, через їх можливість більш точно моделювати складні нелінійні системи й об'єкти та взаємозв'язки між інформаційними параметрами історичних даних про них [49]–[75]. Більш того, використання РНМ, як особливого класу ШНМ надає перевагу, оскільки наявність зворотних зв'язків у нейромоделях дозволяє більш точно дослідити зміни об'єкту діагностування у часі [64]–[69], [75].

Аналіз відомих методів [76]–[85] показав, що більшість існуючих методів мають специфічні вимоги до вхідних даних, тобто первинні дані мають бути представлені в особливій формі та подані на вхід мережі. Також варто відзначити проблеми оброблення великих наборів вхідних даних [76]–[85], що в подальшому може значно погіршити якість синтезованої діагностичної моделі. З іншого боку, при зменшенні об'єму вхідних даних часто виникає проблема із перенавчанням моделі. Розповсюджена також проблема із визначенням додаткових метапараметрів, оскільки часто дослідник має покладатися на власний досвід, що не є досить об'єктивним рішенням [76]–[85]. При цьому більшість методів мають серйозні проблеми з роботою в області локальних екстремумів [86]–[94]. Вирішення такої проблеми вимагає від дослідника використання додаткових методів фільтрації або штучного введення критеріїв зупинки, що значно ускладнює роботу методів. Найбільшою ж проблемою для відомих методів є їх ресурсоемність: більшість методів є багатоітеративними та не припускають використання механізму паралелізації, що значно ускладнює їх використання у реальному житті.

Тому актуальною є розробка нового методу синтезу РНМ, вільного від зазначених недоліків. Новий метод не повинен вимагати специфічного

представлення вхідних даних та має однаково добре працювати при наборах вхідних даних різних об'ємів. Тому пропонується механізм для уникнення проблеми зациклення у локальних екстремумах. Можливість паралелізації методу дозволить значно скоротити час роботи та зменшити ресурсоємність.

Другий розділ присвячено розробці та дослідженню модифікованого генетичного методу та паралельного модифікованого генетичного методу для синтезу РНМ.

## **2.1 Генетичний метод визначення ваг прихованих і вихідних шарів РНМ**

На підставі висвітлених вище властивостей методів синтезу ШНМ більшість існуючих на сьогоднішній день методів є певним розвитком та модифікацією традиційних нейроеволюційних методів, що дозволяє зробити висновки про їх недоліки та обмеження [107]–[119]:

- 1) більшість методів накладає обмеження на структуру (початкову топологію та структуру на основі обраної топології) ШНМ [111];
- 2) переважна більшість методів не модифікує вид функції активації і її параметри [113];
- 3) у багатьох методах еволюція спрямована виключно по шляху ускладнення (в деяких випадках – послідовного спрощення) структури [119];
- 4) деякі методи використовують підхід навчання з учителем, що вимагає наявності репрезентативної вибірки прецедентів і додаткових обмежень на структуру ШНМ [115].

Таким чином, жоден з існуючих методів не поєднує в собі такі властивості як:

- відсутність обмежень на оптимізовану ШНМ;
- динамічність еволюції та її адаптовність;
- модифікація множини всіх допустимих параметрів ШНМ.

Тому для розв'язання завдання параметричного синтезу рекурентних ШНМ доцільно розробити генетичний метод, що буде позбавлений попередніх недоліків та дозволить пришвидшити процес синтезу, уникнути необхідності налаштування великої кількості вільних параметрів, використати нові підходи до кодування інформації про ШНМ та забезпечить синтез нейромоделі із високою точністю роботи.

У генетичному методі визначення ваг прихованих і вихідних шарів РНМ, що пропонується [120]–[124], для пошуку рішення використовується популяція ( $P$ ) нейронних мереж –  $P = \{NN_1, NN_2, \dots, NN_n\}$ , тобто кожна особина представляє окрему ШНМ –  $Ind_i \rightarrow NN_i$  [120], [121]. Під час ініціалізації популяція поділяється на дві частини [123].

Генетична інформація ( $g_{Ind_i} = \{g_1, g_2, \dots, g_n\}$ ) першої половини особин задається випадковим чином [93–96], як зазначено в (2.1):

$$g_{Ind_i} = \{g_1 = Rand, g_2 = Rand, \dots, g_n = Rand\}, \quad (2.1)$$

де  $g_{Ind_i}$  – сукупність генетичної інформації про особину  $Ind_i$ ;

$$Rand = [-0,5; 0,5].$$

Гени другої половини популяції визначаються як інверсія генів першої половини особин [120], тоді (2.1) набуває вигляду (2.2):

$$g_{Ind_i} = \overline{\{g_1 = Rand, g_2 = Rand, \dots, g_n = Rand\}}. \quad (2.2)$$

Це дозволяє рівномірно розподілити одиничні та нульові біти в популяції, що, у свою чергу, забезпечить мінімізацію ймовірності завчасної збіжності методу ( $P_{збіж} \rightarrow \min$ ) [120], [123].

Після початкової ініціалізації генів, як в (2.1) та (2.2), всі особини кодуються мережі без прихованих нейронів ( $N_h$ ), причому всі вхідні



нейрони ( $N_i$ ) з'єднані з кожним вихідним нейроном ( $N_o$ ). Тобто спочатку всі наведені ШНМ відрізняються тільки вагами міжнейронних зв'язків  $w_i$ . В процесі оцінювання, на основі генетичної інформації розглянутої особини спочатку будується нейронна мережа, а потім перевіряється її працездатність, яка визначає пристосованість даної особини. Для перевірки працездатності мережі використовується значення цільової функції (функції пристосованості – fitness function)  $f_{fitness}$  [121]. Після оцінювання всі особини сортуються в порядку зменшення пристосованості, і до схрещування допускається більш успішна половина відсортованої популяції, причому краща особина відразу переходить в наступне покоління [122]. У процесі репродукції кожна особина схрещується з випадково обраною особиною з числа відібраних для схрещування. Отримані два нащадки додаються в нове покоління  $G = P = \{Ind_1, Ind_2, \dots, Ind_n\}$ . Після того, як нове покоління сформовано, починає працювати оператор мутації. Проте важливо зазначити, що через відбір усіканням значно послаблюється різноманітність всередині популяції, що веде до ранньої збіжності алгоритму, тому ймовірність мутації у розробленому методі запропоновано вибирати досить великою (близько  $p_{mut} = 15-25\%$ ) [123].

У випадку, якщо краща особина в популяції не змінюється протягом деякої кількості поколінь (за замовчуванням пропонується таку кількість встановити на рівні семи), відбувається перезапуск методу. Під час перезапуску вся популяція ініціалізується заново, і відбувається повторний процес пошуку рішення. Це дозволяє реалізувати вихід з області локальних екстремумів, обумовлених рельєфом цільової функції, а також великим ступенем подібності особин в одному поколінні [120].

Слід зазначити, що кількість прихованих нейронів теоретично є необмеженою. Для регуляції розмірів одержуваних мереж використовуються три критерії – критерії регулювання розміру та напрямку розвитку мережі,

що дозволяють на етапі мутації адаптивно вибирати, який тип перетворення структури більше підходить для даної мережі [120]–[124].

### 2.1.1 Кодування вхідної інформації

Більшість нейроеволюційних методів використовують поняття генотипу і фенотипу, аналогічно до генетики [100]–[107]. Генотип – це генетичне подання істоти, а фенотип – актуалізоване фізичне представлення істоти [95]–[106], [123]. Еволюційні алгоритми завжди сильно відображають біологію, нейроеволюція нічим не відрізняється в цьому плані.

Питання кодування походить від питання про те, як дослідник хоче подавати ШНМ генетично під час роботи. Спосіб, яким кодується ШНМ, визначає шлях, за яким метод буде обробляти ключові еволюційні процеси: відбір, мутацію і кросовер (або рекомбінацію). Будь-яке кодування потрапляє в одну з двох категорій, пряме або непряме [121]–[124].

Пряме кодування оперує хромосомами, що представляють деяке лінійне представлення ШНМ, в якому в явному вигляді вказані всі нейрони, ваги і зв'язку ШНМ. Таким чином, завжди можна побудувати взаємно-однозначну відповідність між структурними елементами ШНМ (нейронами, зв'язками, вагами та ін.), тобто фенотипом, і відповідними ділянками хромосоми, тобто генотипом [20]–[23].

Цей спосіб подання нейронної мережі є найбільш простим і інтуїтивним, а також дозволяє застосовувати до отриманих хромосомам вже наявний апарат генетичного пошуку (наприклад, оператори кросинговеру і мутації). З найбільш очевидних мінусів такої схеми кодування можна відзначити «розпухання» генотипу при збільшенні кількості нейронів і зв'язків ШНМ і, як наслідок, низьку ефективність за рахунок значного збільшення простору пошуку [107]. Приклад прямого кодування наведено на рисунку 2.1.

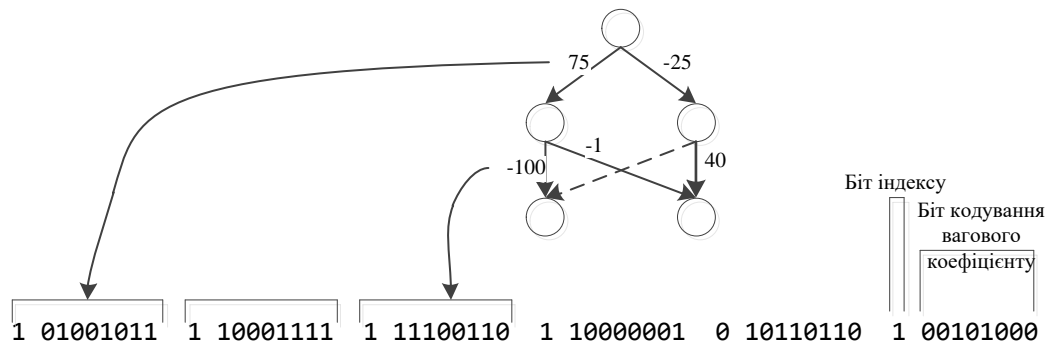


Рисунок 2.1 – Пряме кодування ШНМ

В прикладі на рис. 2.1 біт індексу використовується для вказівки того, чи існує з'єднання чи ні. Біт кодування вагового коефіцієнту кодують ваги в двійковій формі. Ряд дослідників запропонували стратегію кодування, яка кодує ваги в дійсних числах, а також визначені деякі мутаційні оператори, які підходять для кодування [20], [102], [106].

Непряме кодування використовує більш «біологічний» принцип: в генотипі кодується не сам фенотип, але правила його побудови (умовно кажучи, якась програма). При декодуванні генотипу ці правила застосовуються в певній послідовності (найчастіше, рекурсивно і, найчастіше, застосовність правил залежить від поточного контексту), в результаті чого і будується нейронна мережа [99].

При використанні непрямих методів кодування генетичне представлення (а, відповідно, і простір пошуку для генетичних алгоритмів) виходить більш компактним, а сам генотип дозволяє кодувати модульні структури, що дає в певних умовах переваги в адаптивності отриманих результатів [97]–[112]. Натомість же ми отримуємо практичну неможливість простежити, які зміни в генотипі призвели до заданих змін у фенотипі, а також безліч труднощів з підбором генетичних операторів, збіжністю і продуктивністю [113].

Історично, пряме кодування було досліджено раніше і глибше, проте ряд недоліків цього підходу змушують дослідників все більш пильно придивлятися до непрямих методів кодування [107]. Однак за своєю суттю

непрямі методи досить складні для аналізу. Наприклад, одна і та ж мутація правила, розташованого на початку «програми», надає колосальний ефект, а застосована до «кінцевих» правил – ефекту не має зовсім, а в результаті – генетичний пошук має сильну тенденцію до передчасної збіжності. Підбір операторів кросинговеру також є нетривіальним завданням, тому що використання стандартних бінарних операторів, як правило, призводить до частої появи нежиттєздатних рішень [105].

Існує також низка інших методик нейроеволюції. Розглянемо найбільш популярні з наведенням коротких описів кожної з них:

- Боерс і Куйпер – використання контексто-залежних L-систем [21];
- Ділаерт і Бір – підхід, аналогічний Кангелоссі і Елману, але з використанням випадкових булевих нейромереж [20], [100];
- Харп, Самад і Гуха – позонне пряме кодування структури [118];
- Груау – використання граматичного дерева для завдання інструкцій при діленні клітин (чимось схоже на Кангелоссі, Парісі і Нолфі) [23];
- Вааріо – ріст клітин задається L-системами [107].

В результаті непряме кодування зазвичай більш компактне. З іншого боку, встановлення правил для непрямого кодування може призвести до сильного зміщення в просторі пошуку, тому набагато важче створити непряме кодування без суттєвих знань про те, як буде використовуватися кодування.

З теорії генетики відомо, що секвенування біополімерів (білків і нуклеїнових кислот – ДНК і РНК) – це визначення їх амінокислотної або нуклеотидної послідовності [125]. В результаті секвенування отримують формальний опис первинної структури лінійної макромолекули у вигляді послідовності мономерів в текстовому вигляді [124], [125]. Приклад секвенування наведено на рисунку 2.2. Розміри секвенируємих ділянок ДНК зазвичай не перевищують 100 пар нуклеотидів і 1000 пар нуклеотидів при секвенуванні по Сенгеру [119]. В результаті секвенування ділянок ДНК, що

перекриваються, отримують послідовності ділянок генів, цілих генів, тотальної мРНК або повних геномів організмів [124].

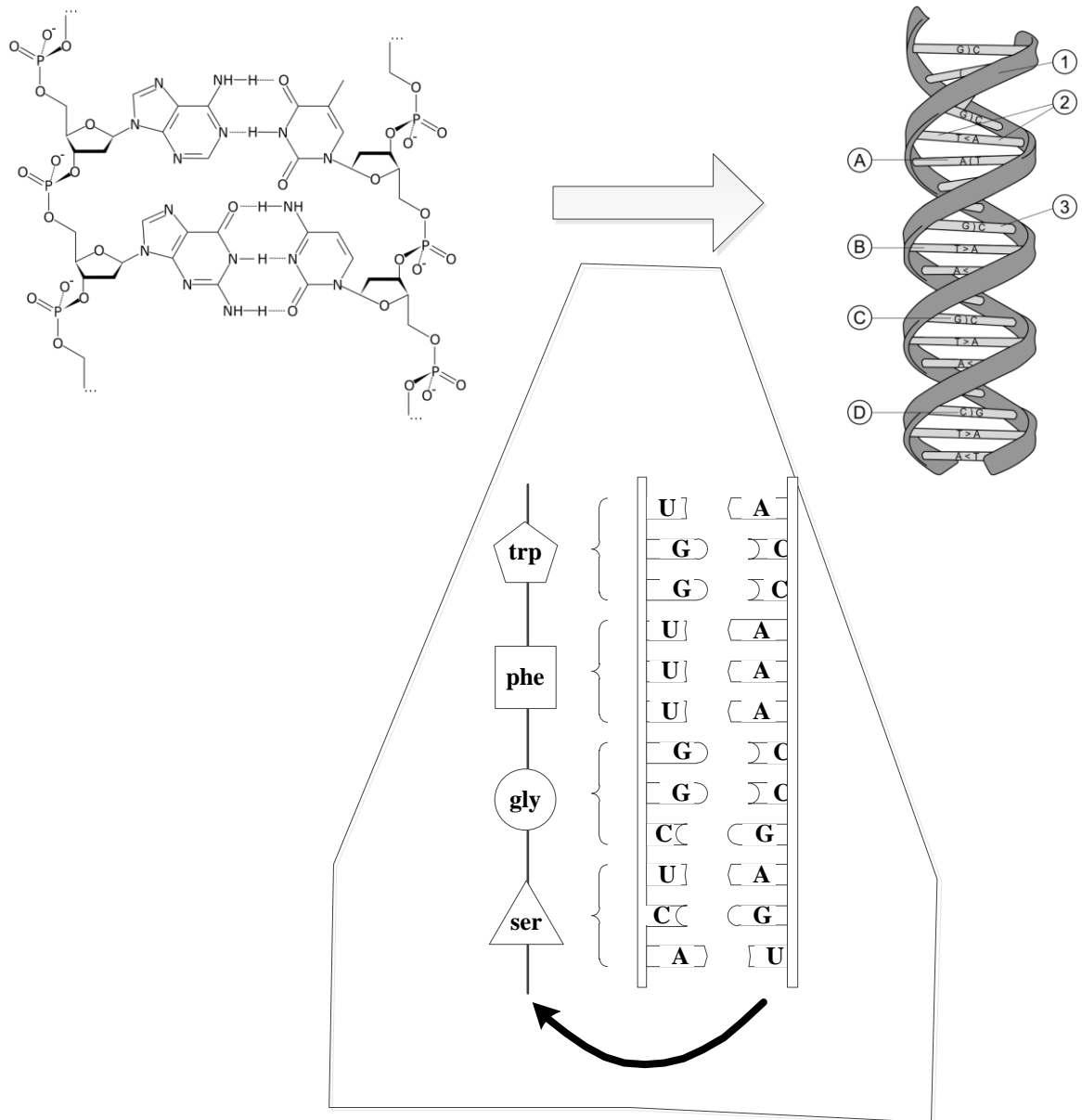


Рисунок 2.2 – Процес трансляції ДНК

Новий метод кодування інформації про ШНМ пропонується організувати базуючись на схожому принципі. Запустимо кодування зв'язків: у генотипі особини представляється інформація про ваги міжнейронних зв'язків нейромоделі [125], [120], але при цьому кожен ген буде містити інформацію про індекси початкового та кінцевого нейрону зв'язку, а також його вагу [121]. У випадку ж коли метод відпрацьовує з РНМ додається

додаткова комірка із вагою зворотнього зв'язку його ж індекс визначається індексом вихідного нейрону [120].

Введемо правила індексування нейронів:

– оскільки кількість входів і виходів мережі є фіксованою величиною, то індекси відповідних нейронів є постійними і приймають значення в інтервалі  $[0; N_i - 1]$  для вхідних нейронів, та  $[N_i; N_i + N_o - 1]$  – для вихідних, де  $N_i$  і  $N_o$  – кількість входів і виходів мережі, відповідно. Видалення вхідних і вихідних нейронів є неможливим [121];

– нові нейрони, що з'являються в результаті мутацій отримують мінімальний можливий індекс. Наприклад, якщо особина являє собою мережу з трьома входами, трьома виходами і не містить прихованих нейронів, то новому нейрону в цій мережі буде присвоєно індекс «5», наступному з'явився – «6» і т. д. [122];

– індекси нейронів в мережі, що подана особиною, не можуть містити пропущені значення, тобто не може бути ШНМ з нейронами, що мають, наприклад, індекси  $N_0, N_1, N_2, N_5, N_6$ . Якщо такий випадок виникає, наприклад, після видалення з мережі нейрона з індексом 4, індекси нейронів, що залишилися коригуються таким чином:  $N_5 \rightarrow N_4, N_6 \rightarrow N_5$ , при цьому змінюються дані у зв'язках, що належать до цих нейронів [123].

Тож отримаємо дврядковий список, де кожні чотири комірки (по дві з рядка) утворюють зберігають інформацію про нейрон [120]. Рисунок 2.3 демонструє приклад кодування мережі таким способом.

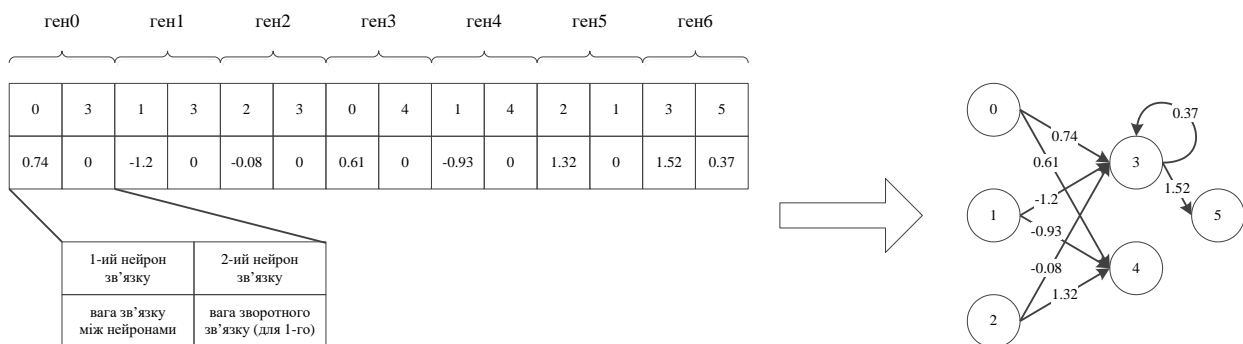


Рисунок 2.3 – Приклад кодування мережі

Додатково слід відзначити, що друге правило забезпечує певне впорядкування шарів – секвенування послідовностей в списках (рис. 2.4).

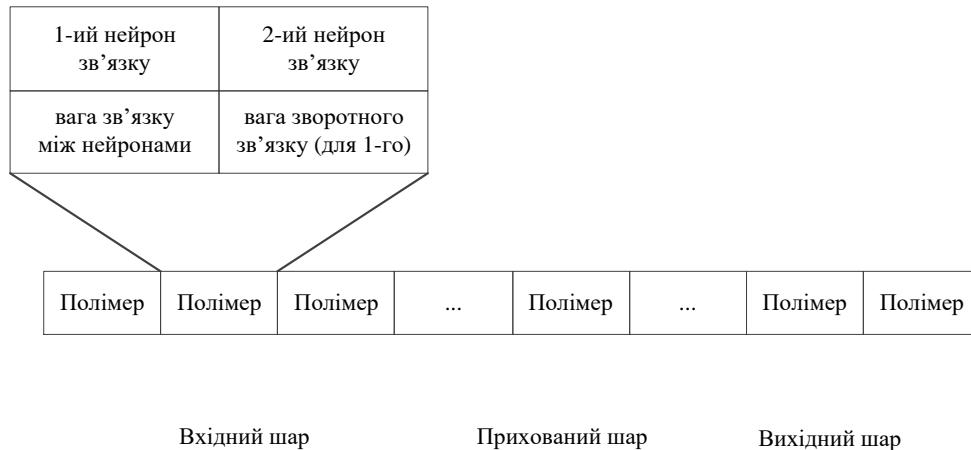


Рисунок 2.4 – Секвенування послідовностей за шарами ШНМ

Новий підхід дозволяє без істотних модифікацій використовувати його при синтезі сучасних топологій ШНМ: РНМ та ГНМ [120].

Переваги використання такого підходу можна пояснити більш компактною схемою кодування інформації про ШНМ, яка дозволяє в подальшому простіше застосовувати генетичні оператори [125].

### 2.1.2 Адаптивний механізм вибору типу мутації

При використанні запропонованого вище методу можуть використовуватися такі види оператору мутації [126]–[130]:

– додавання прихованого нейрона з присвоєнням індексу  $[N_h - 1]$ .

Новий нейрон додається разом з вхідним і вихідним зв'язками. При цьому вихідний зв'язок нейрона не може його зв'язувати із вхідним нейроном [124], [126];

– видалення випадково обраного прихованого нейрона разом з усіма вхідними і вихідними зв'язками. При цьому якщо утворюється пробіл в індексах нейронів, що залишилися, то проводиться корекція індексів

відповідно до наведеного вище алгоритму. Вхідні та вихідні нейрони мережі не можуть бути видалені [127];

– додавання зв'язку. Випадковим чином визначаються індекси початкового і кінцевого нейронів в ШНМ, представленої мутуючої особиною. При цьому зв'язок не може закінчуватися вхідним нейроном. Вага зв'язку визначається також випадково:  $w_i = Rand[-0,5;0,5]$ . Якщо в ШНМ вже існує зв'язок з аналогічними вхідними і вихідними нейронами, то його вага замінюється на випадкову [128];

– видалення випадково обраного зв'язку. При цьому може виникнути ситуація, коли видаляється останній зв'язок у прихованого нейрона. У цьому випадку нейрон також видаляється, і, якщо необхідно, проводиться корекція індексів нейронів мережі [129];

– зміна ваги випадково обраної зв'язку на випадкову величину з діапазону  $[-0,5; 0,5]$  [93], [130].

Таким чином, за допомогою мутації можна точково змінювати параметри структури ШНМ.

Хаотичне додавання (видалення) нейронів і зв'язків може призвести до ситуацій, коли, наприклад, в мережі багато нейронів і мало зв'язків. Більш логічним буде застосовувати різні види мутацій в залежності від особливостей архітектури мережі, представленої мутуючої особиною. Для цього були введені три критерії, що регулюють розмір та напрямок розвитку мережі [120].

Перший з них характеризує ступінь зв'язаності нейронів мережі і обчислюється за формулою (2.3) [120], [122]:

$$f_{\text{зв'яз}} = \frac{N_c}{2^{FB-1} [N_s(N_s - 1) - N_i(N_i - 1) - (1 - FB)N_o(N_o - 1)]}, \quad (2.3)$$

де  $N_c$  – кількість зв'язків в мережі;



$N_i$ ,  $N_o$ ,  $N_s$  – відповідно, кількість вхідних, вихідних нейронів і загальне число нейронів в мережі;

$FB$  – змінна, що позначає, чи дозволена поява зворотних зв'язків ( $FB=1$ ) чи ні ( $FB=0$ ).

Варто відзначити, що зв'язки від прихованих нейронів до вихідних можуть з'являтися в будь-якому випадку. Таким чином, чим менше  $f_{зв'яз}$  за виразом (2.3) тим більше з більшою ймовірністю в результаті мутації буде додано новий зв'язок.

Використання другого коефіцієнта засновано на припущенні, що чим більше елементів в сумі у вхідному та вихідному векторі навчальної вибірки (чим більшою є сумарна кількість вхідних і вихідних нейронів), то, ймовірно, більш складна мережа необхідна для вирішення поставленого завдання. Обчислюється другий коефіцієнт за такою формулою (2.4) [120], [123]:

$$f_{\text{умов.склад}} = \frac{N_i + N_o}{N_s}. \quad (2.4)$$

Тобто, чим більше буде нейронів у мережі, тим меншим буде значення критерію  $f_{\text{склад}}$  за виразом (2.4) і з тим меншою ймовірністю буде обрана мутація, що додає новий прихований нейрон [124].

Третій критерій також заснований на припущенні, що для вирішення більш складних завдань слід використовувати більш складну мережу. Однак цей критерій характеризує не умовну складність мережі, як другий (2.3), а топологічну. Цей критерій заснований на концепції цикломатичної складності [130]. Цикломатична складність – це структурна (або топологічна) міра складності, для вимірювання якості програмного забезпечення. Вимірювання цикломатичної складності, зазвичай, дозволяє виявити процедури з підвищеною складністю. Цикломатична складність була першою

з топологічних заходів складності, застосовувалася на практиці. Третій критерій будемо вираховувати за наступною формулою (2.5) [126]:

$$f_{\text{топ.склад}} = w_s - N_s + 2P, \quad (2.5)$$

де  $w_s$  – загальна кількість зв'язків у мережі;

$N_s$  – загальна кількість нейронів в мережі;

$P$  – кількість компонент зв'язаності.

Видалення зв'язків НМ дає побічний ефект: можуть з'являтися висячі нейрони, у яких немає вхідних зв'язків, а також тупикові нейрони, тобто нейрони без вихідних зв'язків. У випадках, коли функція активації нейронів така, що при нульовій зваженій сумі входів її значення не дорівнює нулю, наявність висячих нейронів дає можливість налаштовувати нейронні зміщення. Варто зазначити, що, окрім забезпечення розмаїтості популяції, видалення зв'язків може сприяти видаленню частини неінформативних і малоінформативних вхідних ознак.

У розробленому методі запропоновано використовувати адаптивний механізм мутації [129], який передбачає вибір виду мутації в залежності від значень критеріїв  $f_{\text{зв'яз}}$ ,  $f_{\text{умов.склад}}$  та  $f_{\text{топ.склад}}$ . Такий підхід з одного боку не обмежує кількість прихованих нейронів, з іншого – перешкоджає безмірному збільшенню мережі, оскільки додавання кожного нового нейрона в мережу буде відбуватися з меншою ймовірністю. Мутація ваги випадково існуючого зв'язку відбувається для всіх мутуючих особин з ймовірністю 0,5.

На рисунку 2.5 наведено у схематичному вигляді процес вибору типу мутації. На рис. 2.5 використовуються такі позначення: ВВ – випадкова величина. Умовно процес вибору типу мутації можна розділити на дві гілки по першому умовному переходу. Гілка збільшення  $f_{\text{зв'яз}}$  виконується по виконанню умови переходу та гілка зменшення  $f_{\text{зв'яз}}$  – виконується при невиконанні умови переходу.

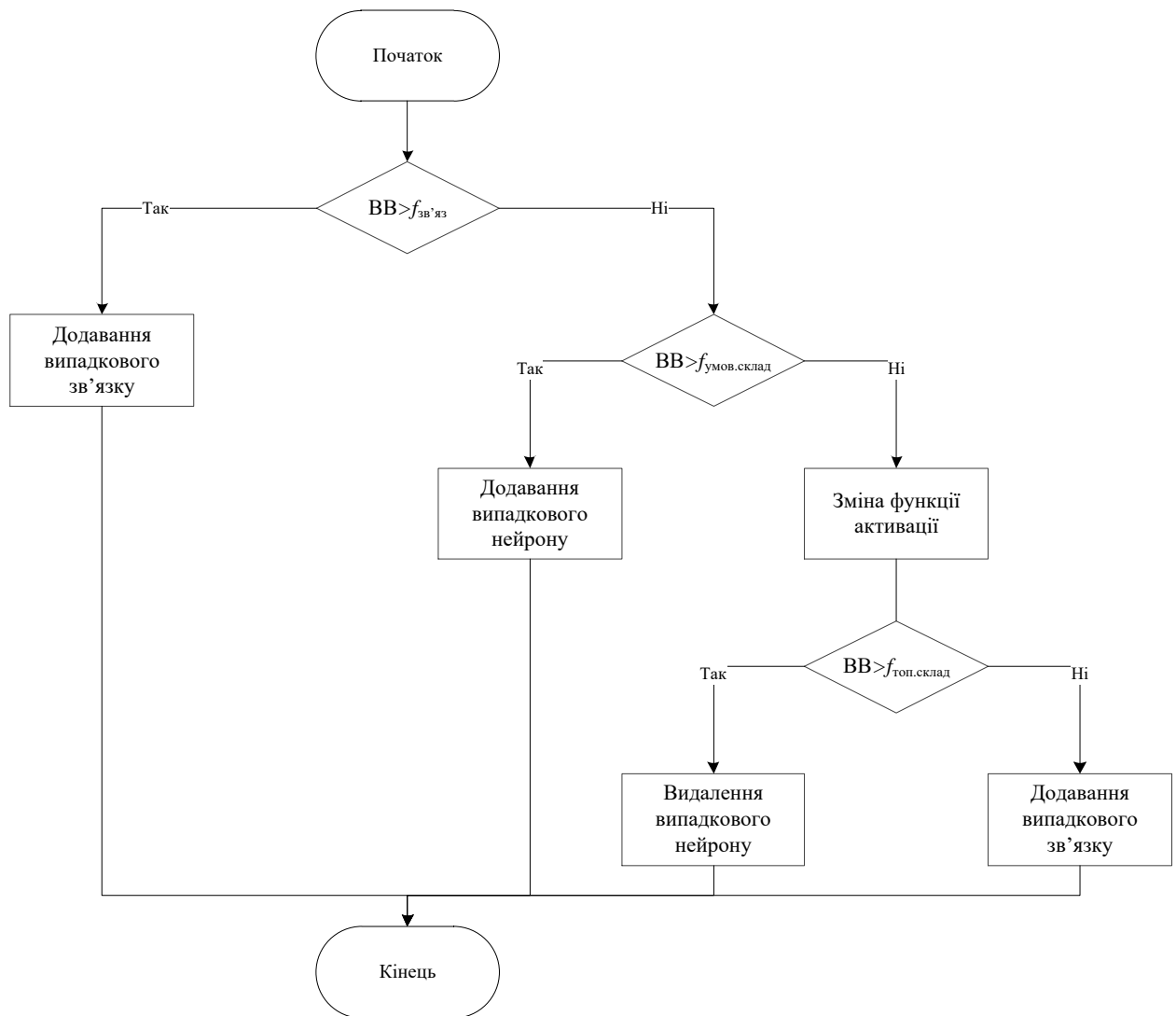


Рисунок 2.5 – Схематичне зображення процесу вибору типу мутації

### 2.1.3 Використання генетичних операторів

Враховуючи заплановане послідовне виконання методу, проаналізуємо найбільш доцільні варіанти генетичних операторів [126]–[130].

Селекція особин полягає у виборі (за розрахованими на попередньому етапі значенням функції пристосованості) тих особин, які будуть брати участь у створенні нащадків для наступної популяції, тобто для чергового покоління. Такий вибір проводиться згідно з принципом природного відбору, за яким найбільші шанси на участь у створенні нових особин мають

хромосоми з найбільшими значеннями функції пристосованості. Існують різні методи селекції [128].

Найбільш популярним вважається так званий метод, який отримав свою назву за аналогією з відомою азартною грою [129]. Кожній хромосомі може бути зіставлений сектор колеса рулетки, величина якого встановлюється пропорційно значенням функції пристосованості даної хромосоми. Тому чим більше значення функції пристосованості, тим більше сектор на колесі рулетки. Все колесо рулетки відповідає сумі значень функції пристосованості всіх особин розглянутої популяції [127]. Кожній особині,  $Ind_i$  для  $i=1,2,\dots,n$  (де  $n$  – чисельність популяції) відповідає сектор колеса  $v(Ind_i)$ , що виражається у процентах, згідно із формулою (2.6)

$$v(Ind_i) = p_s(Ind_i) \cdot 100\% , \quad (2.6)$$

де  $F(Ind_i)$  – значення функції пристосованості особини  $Ind_i$ ;

$p_s(Ind_i)$  – імовірність селекції особини  $Ind_i$ , що обраховується наступним чином:

$$p_s(Ind_i) = \frac{F(Ind_i)}{\sum_{i=1}^n F(Ind_i)} . \quad (2.7)$$

Селекція особини може бути представлена як результат повороту колеса рулетки, оскільки обрана особина (тобто та, що виграла за формулою (2.6)) відноситься сектору цього колеса, що випав [128]. Очевидно, що чим більше сектор, тим більша ймовірність обрання відповідної особини. Тому ймовірність вибору даної хромосоми виявляється пропорційною значенню її функції пристосованості [126]. Якщо всю окружність колеса рулетки представити у вигляді цифрового інтервалу  $[0; 100]$ , то вибір хромосоми

можна ототожнити з вибором числа з інтервалу  $[A, B]$ , де  $A$  та  $B$  позначають відповідно початок і кінець фрагменту кола, відповідного цьому сектору колеса; очевидно, що  $0 \leq A \leq B \leq 100$ . У цьому випадку вибір за допомогою колеса рулетки зводиться до вибору числа з інтервалу  $[0; 100]$ , яке відповідає конкретній точці на колі колеса [129].

Заснований на принципі колеса рулетки метод селекції вважається для генетичних алгоритмів основним методом відбору особин для батьківської популяції з метою подальшого їх перетворення генетичними операторами, такими як схрещування та мутація [126]. Незважаючи на випадковий характер процедури селекції, батьківські особини обираються пропорційно значенням їх функцій пристосованості, тобто згідно ймовірності селекції, що визначається за формулою:

$$p_s(Ind_i) = \frac{F(Ind_i)}{\sum_{i=1}^n F(Ind_i)}.$$

Кожна особина отримує в батьківському пулі таку кількість своїх копій, яке встановлюється виразом (2.8):

$$c(Ind_i) = p_s(Ind_i) \cdot n, \quad (2.8)$$

де  $n$  – кількість особин  $Ind_i$  для  $i = 1, 2, \dots, n$  у популяції,

$p_s(Ind_i)$  – імовірність селекції особини  $Ind_i$ , що обчислюється за відповідною формулою [126]–[130]:

$$p_s(Ind_i) = \frac{F(Ind_i)}{\sum_{i=1}^n F(Ind_i)}.$$

Строго кажучи, кількість копій даної особини в батьківському пулі дорівнює цілій частині від  $c(Ind_i)$ . При використанні формул:

$$p_s(Ind_i) = \frac{F(Ind_i)}{\sum_{i=1}^n F(Ind_i)}$$

та  $c(Ind_i) = p_s(Ind_i) \cdot n$  необхідно звертати увагу на те, що (2.8) в цьому випадку приймає вигляд:

$$c(Ind_i) = \frac{F(Ind_i)}{\bar{F}},$$

де  $\bar{F}$  – середнє значення функції пристосованості у популяції.

Можна зробити висновок, що метод рулетки можна застосовувати тоді, коли значення функції пристосованості позитивні. Цей метод може використовуватися тільки в завданнях максимізації функції (але не мінімізації) [126].

Загалом для цього методу селекції виділяють одразу декілька недоліків:

– можливість застосування методу рулетки лише для одного класу задач, тобто тільки для максимізації (або тільки для мінімізації) [112];

– особини з дуже малим значенням функції пристосованості занадто швидко виключаються з популяції, що може призвести до передчасної збіжності генетичного алгоритму. Для запобігання такого ефекту застосовується масштабування функції пристосованості [126].

З урахуванням зазначених недоліків методу рулетки створені і використовуються альтернативні алгоритми селекції. Один з них називається турнірним методом [122].

При турнірній селекції всі особини популяції розбиваються на підгрупи з подальшим вибором в кожній з них особини з найкращою пристосованістю [123]. Розрізняються два способи такого вибору: детермінований вибір і випадковий вибір. Детермінований вибір здійснюється з імовірністю, рівною 1, а випадковий вибір – з імовірністю: меншою за 1. Підгрупи можуть мати довільний розмір, але найчастіше популяція поділяється на підгрупи по 2-3 особи в кожній [124].

Турнірний метод придатний для вирішення завдань як максимізації, так і мінімізації функції. Крім того, він може бути легко поширений на завдання, пов'язані з багатокритеріальною оптимізацією, тобто на випадок одночасної оптимізації декількох функцій. У турнірному методі допускається зміна розміру підгруп, на які поділяється популяція [124]. Дослідження підтверджують, що турнірний метод діє ефективніше, ніж метод рулетки.

При рангової селекції особини популяції ранжуються за значеннями їх функції пристосованості [126]. Це можна уявити собі як відсортований список особин, упорядкованих у напрямку від найбільш пристосованих до найменш пристосованим (або навпаки), в якому кожної особини приписується число, що визначає її місце в списку і зване рангом. Кількість копій кожної особини, введених у батьківську популяцію, розраховується за апріорно заданою функцією залежно від рангу особини [127].

Перевага рангового методу полягає в можливості його застосування як для максимізації, так і для мінімізації функції [128]. Він також не вимагає масштабування через проблеми передчасної збіжності, актуальною для методу рулетки [128].

В результаті процесу селекції створюється батьківська популяція, також звана батьківським пулом з чисельністю, рівної чисельності поточної популяції [126].

Застосування генетичних операторів до особин, відібраних за допомогою селекції, призводить до формування нової популяції нащадків від створеної на попередньому етапі батьківської популяції [125].

Операція схрещування полягає в обміні фрагментами ланцюжків між двома батьківськими особинами. Пари батьків для схрещування вибираються з батьківського пулу випадковим чином так, щоб ймовірність вибору конкретної особини для схрещування дорівнювала ймовірності  $p_c$ . Наприклад, якщо в якості батьків випадковим чином вибираються дві особини з батьківської популяції чисельністю  $n$ , то  $p_c = 2/n$  [124]. Аналогічно, якщо з батьківської популяції чисельністю  $n$  вибирається  $2z$  особин ( $z \leq n/2$ ), які утворюють  $z$  пар батьків, то  $p_c = 2z/n$ . Звернемо увагу, що якщо всі особини поточної популяції об'єднані в пари до схрещування, то  $p_c = 1$ . Після операції схрещування батьки в батьківській популяції заміщаються їх нащадками [122].

Двоточкове схрещування відрізняється від точкового схрещування тим, що нащадки успадковують фрагменти батьківських особин, що визначаються двома випадково вибраними точками схрещування [124].

Таким чином, розроблено метод для синтезу РНМ на основі генетичного методу. З метою усунення проблеми завчасної збіжності методу на етапі ініціалізації генів популяція ділиться на дві половини, гени другої половини задаються як інверсія генів першої половини. У розробленому методі запропоновано встановлювати досить великі значення ймовірності мутації (близько 15-25%) [120]. Запропоновано використовувати кодування зв'язків, при якому у генотипі особини представляється інформація про ваги міжнейронних зв'язків нейромоделі. Використання критеріїв регулювання розміру та напрямку розвитку мережі дозволяє на етапі мутації адаптивно вибирати тип перетворення структури мережі. Такий підхід, з одного боку, не обмежує кількість прихованих нейронів, з іншого, – перешкоджає безмірному збільшенню мережі, оскільки додавання кожного нового нейрона в мережу буде відбуватися з меншою ймовірністю. Використання модифікованого генетичного методу дозволяє уникнути примусового вибору розміру кроку. РНМ, що синтезуються за допомогою модифікованого



генетичного методу мають високу точність, при цьому розмір вхідного масиву даних не впливає на роботу методу [120].

## **2.2 Паралелізація нейроеволюційного синтезу рекурентних штучних нейронних мереж**

### **2.2.1 Проблеми паралелізації нейроеволюційних методів**

Існує велика кількість нейроеволюційних алгоритмів, які діляться на дві групи. До першої відносяться алгоритми, які здійснюють еволюцію ваг при заданій топології мережі, до іншої – методи, які крім еволюції ваг також виробляють еволюцію топології мережі [125], [131], [132]. Хоча і не існує загальноприйнятих умов для проведення відмінностей, прийнято, що додавання або видалення зв'язків в мережі в ході еволюції називається ускладнення або спрощення відповідно [122].

Генетичний метод як один з найбільш поширених і популярних еволюційних методів набув широкого поширення для синтезу ШНМ [113]. Його застосування дозволяє синтезувати одночасно популяцію нейронних мереж і надалі виробляти точкові зміни (оператор мутації). Після цього популяція оцінюється (відбувається селекція кращих особин популяції) і запускається схрещування [114]. Як правило, використовуються більш прості види схрещування, що гарантують порівняно непогані результати при економному споживанні обчислювальних ресурсів: одноточковий або двоточковий кроссовер. Однак для більш тонкого налаштування нового покоління або ж реалізації багатобатьківського схрещування використовується оператор рівномірного схрещування [120]. Нове покоління проходить всі попередні кроки – метод виконується поки не буде отримана найбільш прийнятна ШНМ.

Використання подібних методів в перспективі надає цілий ряд переваг:

– велика різноманітність отриманих топологій дозволяє підібрати найбільш прийнятне рішення. Так, наприклад, для простих завдань, можна встановити жорсткі обмеження для отримання простих ШНМ, а більш складні структури (рекурентні, глибокі, згорткові ШНМ та т.п.) будуть отримані для більш складних завдань [113];

– адаптивність отриманих ШНМ. Тут мається на увазі, що під час синтезу метод може допомогти синтезувати більш точну та структурно прийнятну ШНМ під конкретну задачу і набір вхідних даних [115];

– універсальність. Подібні методи можуть працювати з різноманітними наборами вхідних даних або різноманітними топологіями ШНМ, не вимагаючи точних даних про структурні особливості [117];

– можливість використання паралельного підходу. Що особливо актуально в умовах стрімкого розвитку сучасних паралельних обчислювальних систем з використанням безлічі ядер процесорів (CPU) або легковажних потоків графічних процесорів (GPU) [131].

Однак використання паралельного підходу для генетичних методів пов'язане з низкою проблемних питань. Розглянемо найбільш типові проблеми [132]–[134]:

- вибір або розробка стратегії взаємодії складових частин методу;
- вибір частоти міграцій між підпопуляціями;
- визначення мігруючих особин та їх кількості;
- визначення структури еволюції окремих підпопуляцій.

Розглянемо зазначені проблеми більш детально. Структура паралельної системи є важливим фактором у продуктивності паралельного алгоритму, оскільки вона визначає, наскільки швидко (або наскільки повільно) краще рішення поширюється між популяціями [133]. Якщо система є сильно зв'язаною, то кращі рішення будуть швидко поширюватися в усі потоки і можуть швидко наситити популяцію. З іншого боку, якщо мережа слабо зв'язана, рішення будуть поширюватися повільніше і потоки будуть більш ізольованими один від одного [134]. Може відбуватися подальший

паралельний розвиток і схрещування різних рішень для отримання потенційно кращих рішень.

Загальна тенденція в паралельних генетичних алгоритмах (ПГА) – це використання статичних структур системи, які визначаються до запуску методу і залишаються незмінними [132].

Частота міграцій також має великий вплив на кінцеве рішення. Як відомо [132], занадто часті міграції призводять до виродження популяцій, а рідкісні, навпаки, до зниження збіжності. Для регулювання частоти міграції застосовуються різні методи, які можна розбити на два типи: адаптивні та подієві. У першому випадку методи адаптації використовуються для налаштування частоти міграції в процесі роботи алгоритму. У другому випадку застосовуються методи, що визначають необхідність міграції, тобто міграція здійснюється тільки при настанні якоїсь події [133].

Для відбору особин для міграції застосовуються механізми селекції. Відомо, що окремі хромосоми можуть містити важливі фрагменти генетичного коду, але дані частини можуть перебувати в хромосомах, які відрізняються поганою пристосованістю [134]. Але в той же час, виключення подібних рішень може привести до того, що відбудеться передчасна збіжність, або пропуск глобального оптимуму [131].

Використання різних стратегій накладає головне обмеження – необхідність формування однотипної структури хромосоми. Але ефект, який можливий при вдалому формуванні, може бути набагато більшим, ніж при використанні однієї структури ПГА у всіх підпопуляціях [134].

Також варто відзначити, що велика кількість міграцій і навіть динамічного обміну проміжною інформацією між потоками вимагає додаткових накладних витрат, що іноді в значній мірі нівелює досягнення паралельного виконання обчислень.

## 2.2.2 Використання стратегій ройового інтелекту при паралелізації процесу синтезу штучних нейронних мереж

З попереднього аналізу видно, що ряд проблем пов'язаний з вертикальними зв'язками всередині паралельної системи і операціями пересилання проміжних даних між потоками. Найбільш популярними еволюційними біоінспірованими алгоритмами на сьогоднішній день є, так звані, ройові процедури (Particle Swarm Optimization – PSO) [135], [136]. Методи ройової поведінки запозичені з природи, де групи тварин (зграї, рої) демонструють несподівано хороші результати при тому, що кожна з особин має досить примітивним розумом [135]. В останні кілька десятиліть в різних областях науки робляться небезуспішні спроби відтворити ройові механізми поведінки в штучних моделях. Багато моделей при цьому прагнуть відтворити властивість, притаманне природним самоорганізуються системам. Система в процесі взаємодії її елементів як ціле набуває ознак, які не були притаманні елементам окремо.

Основну ідею ройових методів можна пояснити наступним чином. Штучні частинки, що переміщуються в  $n$ -мірному пошуковому просторі, можуть вести себе подібно зграї або рою [136]. Представники рою шукають щось, на кшталт корму – екстремум заданої цільової функції. За аналогією з генетичними обчисленнями можна стверджувати, що рій подібний до популяції, а частинки (індивідууми) відповідають хромосомам. Частинки переміщуються у пошуковому просторі, змінюючи напрямок відповідно до власного досвіду та досвіду сусідів. Кожна частинка характеризується вектором швидкості і вектором позиції [137]. При оптимізації  $N$ -мірної функції такі вектори мають розмірність  $N$ . Як і в біологічному житті, передбачається, що кожна частка регулює свої вектори позиції і швидкості відповідно до власного досвіду (когнітивна складова), а також за інформацією, отриманою від інших членів рою [135]. Тут когнітивний досвід частинки розуміється як її знання про кращу позицію, в якій вона сама

перебувала, а соціальне знання частинки – як знання про кращу позицію, через яку пройшла одна з частинок в одній групі з нею. У задачі оптимізації найкраща позиція частинки розуміється як позиція, в якій мінімізується значення функції [138].

Порівнюючи методи ГА та PSO можна відзначити ряд подібностей:

- обидві групи методів є стохастичними;
- рішення шукається на базі популяції індивідуумів;
- початкова популяція, найчастіше, генерується випадковим чином;
- для роботи методу потрібен обрахунок фітнес-функції кожного індивідуума;
- області можливого використання методів збігаються.

Але існують і відмінності:

- кількість вільних параметрів, що на початку вказуються в методах PSO, є меншою;
- класичні еволюційні оператори відсутні, а самі частинки не «вмирають» – не видаляються після роботи.

Розглянемо ж, які підходи методів PSO можуть допомогти вдосконалити ПГА. У класичній теорії штучного інтелекту для певної задачі створюється одна інтелектуальна система, що має всі необхідні ресурси для її вирішення [139]. У теорії багатоагентних систем використовується протилежний принцип. Вважається, що один агент має неповне уявлення про глобальну проблему, тому створюють деяку безліч агентів і забезпечують ефективну взаємодію між ними [139]. В рамках «колективного» інтелекту глобальна поведінка всієї системи розглядається, як результат взаємодій ряду простих агентів.

Таким чином, з ройових методів можна почерпнути такі принципи:

- багатоагентна система – це популяція простих і залежних один від одного агентів;
- кожен агент самостійно визначає свої реакції на події в локальному середовищі і взаємодії з іншими агентами;

- зв'язки між агентами є горизонтальними, тобто не існує агента-супервізора, керуючого взаємодією інших агентів;
- немає точних правил, щоб визначити глобальну поведінку агентів;
- поведінка, властивості і структура на колективному рівні породжуються тільки локальними взаємодіями агентів [135]–[138].

Таким чином, можна прийти до проектування та конструювання динамічної паралельної системи. У цьому випадку потік не обмежується зв'язками з певною фіксованою кількістю потоків; замість цього мігранти направляються до потоків, які задовольняють певний критерій [139]. В якості подібного критерію береться ступінь різноманітності популяції або міра генотиповості, тобто відстані між двома популяціями (або відстані від характерної особини популяції, наприклад, улюбленої). При такій структурі необхідні механізми відстеження подій в сусідніх популяціях, причому якщо в одній з сусідніх популяцій деяка подія настала, то слід очікувати події і в другій популяції [139].

На рисунку 2.6 наведено загальну схему розробленого паралельного нейроеволюційного методу.

Зі схеми видно, що як такого головного потоку в системі не передбачається. На початку визначаються лише основні та додаткові параметри майбутніх підпопуляцій [135]–[139]. Серед таких параметрів можна відзначити: розмір кожної популяції, топологічні особливості, тощо. Ця інформація розподіляється на потоки паралельної системи. Вже на потоках самостійно формуються та кодуються підпопуляції для роботи. Основні генетичні оператори залишаються, проте поки процес виконується на окремих потоках слід використовувати найпростіші оператори, наприклад доточковий кроссовер або рангову селекцію. Коли окремо на потоках було отримано найкращі особини вони синхронізуються. Виконується селекція та перевірка найкращої особини з отриманих [139]. Якщо така особина поки не є прийнятною запускається механізм багатобатьківського кроссоверу, з визначенням долі генетичної інформації, що буде переходити від кожного з

індивідуумів за ранговим принципом. Введення на цьому етапі рангової селекції, з додаванням додаткових критеріїв для оцінки не лише точності, а й топологічних особливостей, та рівномірного схрещування допустимо, адже використовується один потік – навіть якщо система буде використовувати для паралелізації GPU цей етап буде виконуватися на CPU, тому не буде гальмувань через складність операцій [139]. У випадку, якщо із нового покоління краща особина не задовольнить критерії якості, нове покоління не буде видалено, а буде використано для оновлення даних для формування нових підпопуляцій.

Таким чином спроектована паралельна структура методу забезпечує:

- прості підпопуляції ШНМ, що розподілені між потоками та не залежать один від одного;
- кожний потік агент самостійно визначає свої реакції (типи мутацій, схрещування, тощо) на події в локальному середовищі;
- головний потік використовується лише щоб спочатку запуснути роботу, а потім синхронізувати результати, а не для управління взаємодією інших потоків;
- визначення кращих рішень на загальному рівні залежить лише від локальних сценаріїв потоків.

### **2.2.3 Використання механізму селективного тиску**

З біології відомо [125], [126], що селективний тиск – це сила впливу факторів навколишнього середовища (якість їжі, екстремальна температура, домішки у живильне середовище підчас експериментів та ін.), в результаті дії яких виживають тільки стійкі до них організми. Використовується для виявлення генотипів, що контролюють цю стійкість. Тож селективний тиск це своєрідний показник інтенсивності дії природного відбору з точки зору зміни генетичного складу популяцій в ряді поколінь. Чим інтенсивніше відбір, тим вище швидкість еволюційних змін в популяції [127].

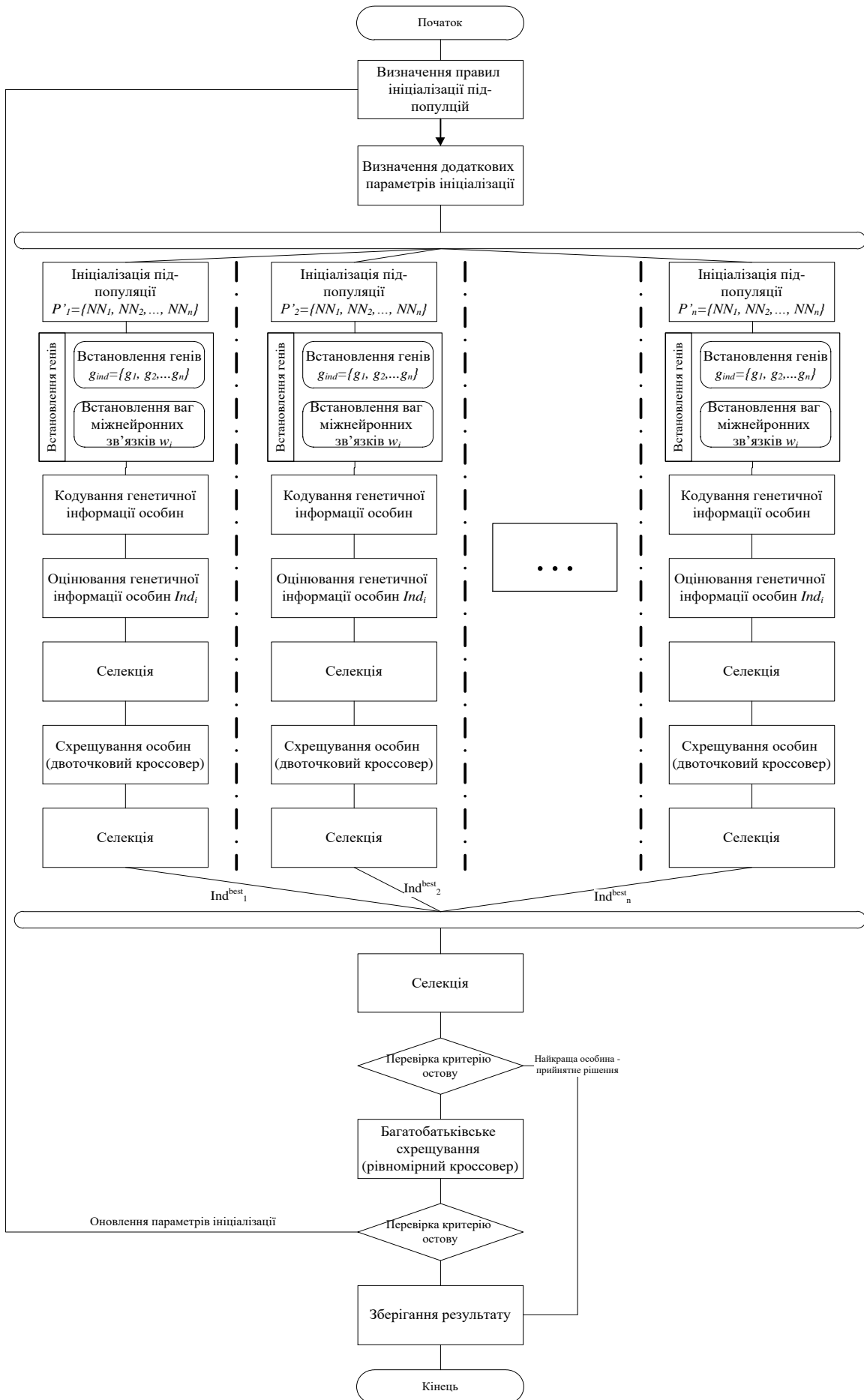


Рисунок 2.6 – Схема паралельної реалізації розробленого методу



Тому ідея використання подібного механізму спрямована на можливість відбору для подальшого схрещування та формування нової популяції одразу декількох найкращих особин, проте налаштувати ймовірність передачі генів від батьківських особин слід спираючись на особливі показники [126].

Введемо селективний тиск на етапі схрещування, розширивши операцію селекції. Встановимо залежність між ймовірністю передачі гена нащадку та значенням придатності батьків [120]. Для цього розширимо ранговою селекцію введенням додаткових критеріїв оцінки особин.

Перший критерій буде використовуватися для оцінки надмірності використаної пам'яті. Як було зазначено вище, нейромережі володіють пам'яттю, що реалізується вагами зв'язків. Чим менше пам'ять мережі, тим менше образів вона може запам'ятати. Проте, в ситуації коли дві мережі з різним об'ємом пам'яті забезпечують необхідну точність розпізнавання (оцінювання), то мережа з меншою пам'яттю, звісно має кращі узагальнюючі властивості. Надмірність пам'яті мережі будемо характеризувати за допомогою коефіцієнта надмірності для зберігання навчальної вибірки за формулою (2.9) [126]:

$$crit_m = \frac{W_{FFc} + W_{FBc}}{samp_{Inst} \cdot samp_{Feat}}, \quad (2.9)$$

де  $W_{FFc}$  – це кількість прямих зв'язків нейромережі ( $W_{FFc} = \{w_1, w_2, \dots, w_i\}$ );

$W_{FBc}$  – це кількість зворотних зв'язків для рекурентних нейромереж ( $W_{FBc} = \{w_1, w_2, \dots, w_j\}$ );

$samp_{Inst}$  – це кількість екземплярів у навчальній вибірці;

$samp_{Feat}$  – це кількість ознак у навчальній вибірці.

Якщо значення з формули (2.9)  $crit_m > 1$ , то пам'ять мережі надлишкова (розмірність пам'яті мережі більше розмірності навчальної

вибірки). Якщо  $crit_m = 1$ , то мережа може запам'ятати всю навчальну вибірку (розмірність пам'яті мережі дорівнює розмірності навчальної вибірки). Якщо  $crit_m < 1$ , то мережа не зможе в точності запам'ятати всю навчальну вибірку (розмірність пам'яті мережі менше розмірності навчальної вибірки), однак при цьому мережа буде проявляти узагальнюючі та апроксимуючі здібності [127].

Використання другого критерію пов'язано із апроксимаційними властивостями нейромережі. Однією з найважливіших характеристик нейромережеских моделей є якість апроксимації [128]. У випадку, коли рівень помилки моделей однаковий, якість апроксимації вище у тій моделі, яка використовує менше зв'язків. Коефіцієнт якості апроксимації нейромережескої моделі визначимо як середню частку помилки, що припадає на ненульові ваги мережі за формулою (2.10) [120], [121]:

$$crit_a = \frac{Error}{W_{FFc} + W_{FBc} - W_{w=0}}, \quad (2.10)$$

де  $Error$  – сукупна помилка, що допускається мережею (наприклад, середньоквадратична помилка), така, що  $Error < \xi$ , де  $\xi$  – максимально допустима помилка (мета навчання). В якості помилки  $Error$  можна використовувати помилку по навчальній вибірці ( $E_{нав.}$ ) або за тестовою вибіркою ( $E_{тест.}$ );

$W_{FFc}$  – це кількість прямих зв'язків нейромережі ( $W_{FFc} = \{w_1, w_2, \dots, w_i\}$ );

$W_{FBc}$  – це кількість зворотних зв'язків для рекурентних нейромереж ( $W_{FBc} = \{w_1, w_2, \dots, w_j\}$ );

$W_{w=0}$  – нульові ваги (зв'язки нейромережі, вага яких дорівнює 0).

Таким чином розглянемо модифікацію рангової селекції з використанням критеріїв (2.9) та (2.10) оцінки нейромережеских молей [120].

Селекція розпочинається з сортування (ранжування) особин на основі їх придатності таким чином, щоб  $F(Ind_i) \geq F(Ind_j)$  для  $i > j$ . Потім кожному індивіду призначається ймовірність  $p_s$  бути відібраним, взята із заданого розподілу з обмеженням  $\sum_i p_s = 1$ . Ймовірність селекції обчислюється за формулою (2.11):

$$p_{s_i} = \frac{1}{n} \left( a - (a - b) \frac{rank + crit_m + crit_a - 1}{n - 1} \right), \quad (2.11)$$

де  $a \in [1;2]$ ,  $b = 2 - a$ ,  $rank$  – ранг особин у відсортованому списку особин.

Використання критеріїв під час визначення ймовірності селекції за (2.11) вирішує одразу декілька проблем, а саме:

- 1) завчасну збіжність методу;
- 2) зменшення різноманітність;
- 3) вибір кращих особин, при однаковому рангу (при однаковому значенні функції пристосованості).

Відомо, що налаштування ймовірності передачі гена батьків нащадку в рівномірному схрещуванні може істотно підвищити його ефективність, а також дозволяє емулювати інші оператори схрещування (одноточковий, двоточковий) [125], [127]. Відомо також, що використання оператора рівномірного схрещування дозволяє застосовувати так звану багатобатьківську рекомбінацію, коли для породження одного нащадка використовується більше двох батьків. Рівномірне схрещування дає більшу гнучкість при комбінуванні рядків, що є важливою перевагою при роботі із ГА [126].

Оператор рівномірного схрещування є одним з найбільш ефективних операторів рекомбінації в стандартному ГА [128].

Рівномірне схрещування виконується відповідно до випадково обраним еталоном, який вказує, які гени повинні успадковуватися від

першого батька (інші гени беруться від другого з батьків). Тобто загальне правило рівномірного схрещування можна представити наступним чином у вигляді (2.12) [120], [128]:

$$Crossover(Ind_1, Ind_2, DataofCros) = Ind_3$$

$$g_{Ind_3} = \left\{ \begin{array}{l} g_1 = Rand(g_{Ind_1}, g_{Ind_2}), g_2 = Rand(g_{Ind_1}, g_{Ind_2}), \dots, \\ g_i = Rand(g_{Ind_1}, g_{Ind_2}) \end{array} \right\} \quad (2.12)$$

Приклад рівномірного схрещування (2.12) наведено на рисунку 2.7.



Рисунок 2.7 – Приклад рівномірного схрещування

Тому у якості оператора схрещування буде використовуватися рівномірне схрещування із заданим розміром батьківського пулу. Пул буде заповнюватися особинами відібраними за допомогою модифікованої рангової селекції [120]. Такий підхід додає методу гнучкості та дозволяє сподіватися на зміну поведінки алгоритму [139].

Таким чином, розроблено паралельний метод синтезу РНМ, у якому використовуються технології паралельних обчислень та ройового інтелекту, що дозволяє пришвидшити виконання процесу синтезу РНМ та підтримувати генетичне різноманіття, що, в свою чергу, мінімізує завчасну збіжність [120], [139]. Використання механізмів селективного тиску у запропонованому методі дозволяє на етапі селекції відібрати кращі особини, враховуючи структурні особливості, що дозволяє сформувати розширений батьківський пул. Оператор рівномірного схрещування забезпечує виконання

багатобатьківського наслідування параметричних та структурних особливостей РНМ. Розподілення цих механізмів між головним потоком та паралельними забезпечує більш раціональне використання обчислювальних потужностей [136]–[139].

### 2.3 Висновки за розділом 2

У цьому розділі розв'язано актуальне завдання синтезу діагностичних моделей на основі РНМ. Запропоновано методи синтезу РНМ на основі нейроеволюційного підходу, які дозволяють виконати процес синтезу нейромоделей із високим рівнем точності, що можуть використовуватися як базис для синтезу моделей для неруйнівного діагностування за історичними даними про систему та об'єкт.

Розроблено модифікований генетичний метод для синтезу РНМ (МГА для РНМ), в якому, на відміну від існуючих методів синтезу нейромоделей, на етапі кодування використовується спосіб кодування інформації про ШНМ на основі секвенування, що дозволяє мінімізувати ймовірність завчасної збіжності. Використання запропонованих критеріїв регулювання розміру та напрямку розвитку ШНМ у розробленому методі дозволяє на етапі мутації адаптивно вибирати тип перетворення структури нейромережі, що перешкоджає безмірному збільшенню мережі, не обмежуючи кількості прихованих нейронів, та синтезувати РНМ, що характеризуються високою точністю.

Вперше запропоновано паралельний генетичний метод з імплементацією механізмів селективного тиску (ПГМзСТ), в якому основні етапи процесу еволюційного синтезу РНМ виконуються на паралельних обчислювальних вузлах, використовуються механізми селективного тиску та рівномірного схрещування, що дозволяє скорочувати розміри популяції, не враховуючі та не опрацьовуючи ті особини з популяції, що відрізняються малим значенням функції пристосованості. Застосування модифікованих

генетичних операторів відбору та схрещування дозволяє підтримувати генетичне різноманіття популяцій, забезпечити уникання проблем з надмірною ітеративною природою еволюційних обчислень, враховувати додаткові показники якості нейромережових моделей, більш продуктивно використовувати механізми рівномірного багатобатьківського схрещування. Збільшення точності та скорочення витрат пам'яті й обчислювальних потужностей для збереження та проведення схрещування повного обсягу популяції засвідчують високу ефективність запропонованого методу.

Результати, наведені у другому розділі, було опубліковано в роботах [120] та [139].

### 3 СТРУКТУРНА ОПТИМІЗАЦІЯ СИНТЕЗОВАНИХ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ

Існуючі методи моделювання нелінійних систем [140]–[144] стикаються з проблемами високої розмірності задач, вимогами високої точності і узагальнюючої здатності отриманих моделей. Ці проблеми можуть бути вирішені за допомогою методів ітераційних або перебору [40], [41], [140], які засновані на принципах селекції, еволюції та адаптації, які є методами евристичної самоорганізації. При цьому в реальному житті досить важко створити адекватну модель складного об'єкта при використанні тільки одного методу індуктивного моделювання. Зазвичай потрібно об'єднувати сучасні методи і технології евристичної самоорганізації [141], застосовувати багаторівневе моделювання, розробляти гібридні алгоритми [144].

Брак або ж надлишок даних – часті проблеми при вирішенні завдань: немає достатньої кількості експериментів (при моделюванні фізичних об'єктів), не вистачає або складно виділити інформативні дані про пацієнтів (для побудови прогнозу стану здоров'я) [141], [143]. При цьому потрібно визначити параметри нового елемента, спрогнозувати результат хвороби, рекомендувати лікування. Часто для вирішення таких завдань використовують ШНМ.

Як зазначено в ряді робіт [64]–[75] ШНМ успішно застосовуються в різних областях – там, де потрібне вирішення завдань прогнозування, класифікації та управління. Проте вибір структури мережі, тобто числа шарів, числа нейронів і числа зв'язків для кожного нейрона – є однією з найбільш складних проблем. Більш того, сучасні архітектури ШНМ [68]–[75] можуть гарантувати високу точність, проте через складну структуру (наявність зворотних зв'язків, декілька прихованих шарів) швидкість їх роботи є недостатньою, що значно обмежує їх реальне застосування для медичного діагностування у реальному часі.

Оскільки останні дослідження [145]–[147] довели, що у процесі сну у людському мозку відбуваються процеси впорядкування набутої інформації за рахунок скорочення, видалення або додавання нових зв'язків між нейронами – існує припущення, що схожі методи оптимізації структури штучних ШНМ зможуть вирішити ряд існуючих проблем.

### 3.1 Застосування індикаторної системи при визначенні механізмів доналаштування штучних нейронних мереж

Маємо задачу, що характеризується умовним рівнем складності ( $Lev_{task}$ ) [148]–[151]. Рівень складності це певна інтегральна оцінка, що складається із декількох характеристик:

– нехай  $X = \{x_1, x_2, \dots, x_n\}$  – множина незалежних змінних – ознак, а  $Y = \{y_1, y_2, \dots, y_m\}$  – множина значень залежних змінних,  $n$  та  $m$  – кількість вхідних ознак, що характеризують екземпляри вибірки, тоді позначимо вибірку вхідних даних –  $Sample = \langle X, Y \rangle$ ;

–  $Lev_{smp\l fctn}$  – рівень можливого спрощення структури (наприклад, кількості вхідних даних);

–  $Lev_{fctr}$  – загальна кількість суттєвих та менш суттєвих та/або несуттєвих факторів;

–  $Lev_{accmeas}$  – рівень точності вимірювань;

–  $Lev_{manag}$  – рівень можливого контролю та управління.

Спираючись на це представимо проблему наступним чином: для синтезу ШНМ ( $NN$ ) необхідно визначити множину нейронів  $N = \{N_i, N_o, N_h\}$ , що складається із підмножин вхідних  $N_i = \{N_{i_1}, N_{i_2}, \dots, N_{i_l}\}$ ,  $l = 1, 2, \dots, |N_i|$  вихідних  $N_o = \{N_{o_1}, N_{o_2}, \dots, N_{o_p}\}$   $p = 1, 2, \dots, |N_o|$  та прихованих нейронів  $N_h = \{N_{h_1}, N_{h_2}, \dots, N_{h_r}\}$   $r = 1, 2, \dots, |N_h|$  та множину ваг  $w = \{w_k\}$



зв'язків  $c = \{c_1, c_2, \dots, c_k\}$ ,  $k = 1, 2, \dots, |c|$  між нейронами. Визначивши значення елементів множин, можна вважати синтез ШНМ – завершеним [149], [151].

Для цього будемо використовувати оцінку складності задачі, отримавши яку вирішимо як надалі будемо проектувати топологію  $NN_{struct}$  ШНМ. Якщо задачу за рівнем складності можна віднести до класу «організована простота», то використовуючи відбір ознак отримаємо найбільш інформативні та використовуючи дослідження вхідних даних зазначимо необхідну кількість нейронів в прихованому шарі мережі  $N_h = \{N_{h_1}, N_{h_2}, \dots, N_{h_r}\}$ ,  $r = 1, 2, \dots, |N_h|$ . Після цього можна буде виконати навчання нейромоделі [148], [149].

Складність системи визначається кількістю складових її елементів і можливих зв'язків між ними. Ступінь складності вимірюється різноманітністю системи. Різноманітність характеризує число можливих станів системи [150].

За роботами Пітера Сенга [148], системна складність існує в двох основних формах.

Складність деталізації (структурна складність) виникає в результаті великої кількості систем, системних елементів і встановлених зв'язків в будь-який з двох основних топологій (ієрархія або мережа). Ця складність пов'язана з системами, як вони є; а саме, з їх статичним існуванням.

Динамічна складність (поведінкова складність) пов'язана з взаємозв'язками, які виникають між готовими, функціонуючими системами в процесі їх роботи, тобто між очікуваним і навіть несподіваним поведінкою, яке фактично виникає.

У. Уівер [149] сформулював вихідну точку зору, виділивши наступні категорії складності: організована простота, організована складність, безладна складність. Ці категорії і пізніші роздуми, зокрема Р.Л. Флада і Е.Р. Карсона [150] і автора книги, дають підстави для використання наведеної нижче класифікації складності.

Організована простота (ОП) має місце, коли є невелика кількість істотних факторів і велика кількість менш істотних і/або несуттєвих факторів. Спочатку ситуація може здатися складною, але після її вивчення менш істотні і несуттєві фактори можуть бути виключені з картини і може бути виявлена прихована простота [149].

Організована складність (ОС) переважає в таких фізичних і абстрактних системах, структура яких організована так, щоб бути зрозумілою, і тому податливою вченим при описі складної поведінки і структуруванні процесу створення складних систем і управління їх життєвими циклами. Це багатство, яке не повинно бути надмірно спрощено [148], [149].

Безладна складність (БС) виникає, коли є багато змінних, які демонструють у високому ступені випадкове, безладне поведінку. Вона також може бути результатом відсутності необхідного контролю над структурою складних неоднорідних систем через неадекватне управління архітектурою протягом життєвого циклу системи (повзучої складності) [150].

Складність пов'язана з людським фактором (СПЛФ), виникає там, де сприйняття будь-якої системи викликає почуття складності. У цьому контексті люди стають системами спостережень. Ми могли б також пов'язати цю категорію з системами, в яких люди є елементами і можуть ґрунтовно посприяти організованій простоті, організованій складності і безладній складності. Розумне або нерозумне поведінка окремих осіб в конкретних ситуаціях, природно, є істотним фактором по відношенню до складності [151].

Як вже повідомлялося для майбутньої роботи на початку будемо використовувати оцінку складності задачі. Так за оцінкою визначимо до якого з 4 типів за Уівером відноситься задача. Основний розподіл наведено на рисунку 3.1. Так на початку вивчаються вхідні дані задачі: чим більше об'єм вибірки вхідних даних – тим вищою буде оцінка, тому що такі задачі вимагають або більш складних операцій з попередньої обробки вхідних

даних (редукція даних, відбір інформаційних ознак, тощо), або ж більш складних топологій ШНМ, аби закодувати усі ознаки та відслідковувати їх взаємозв'язок.

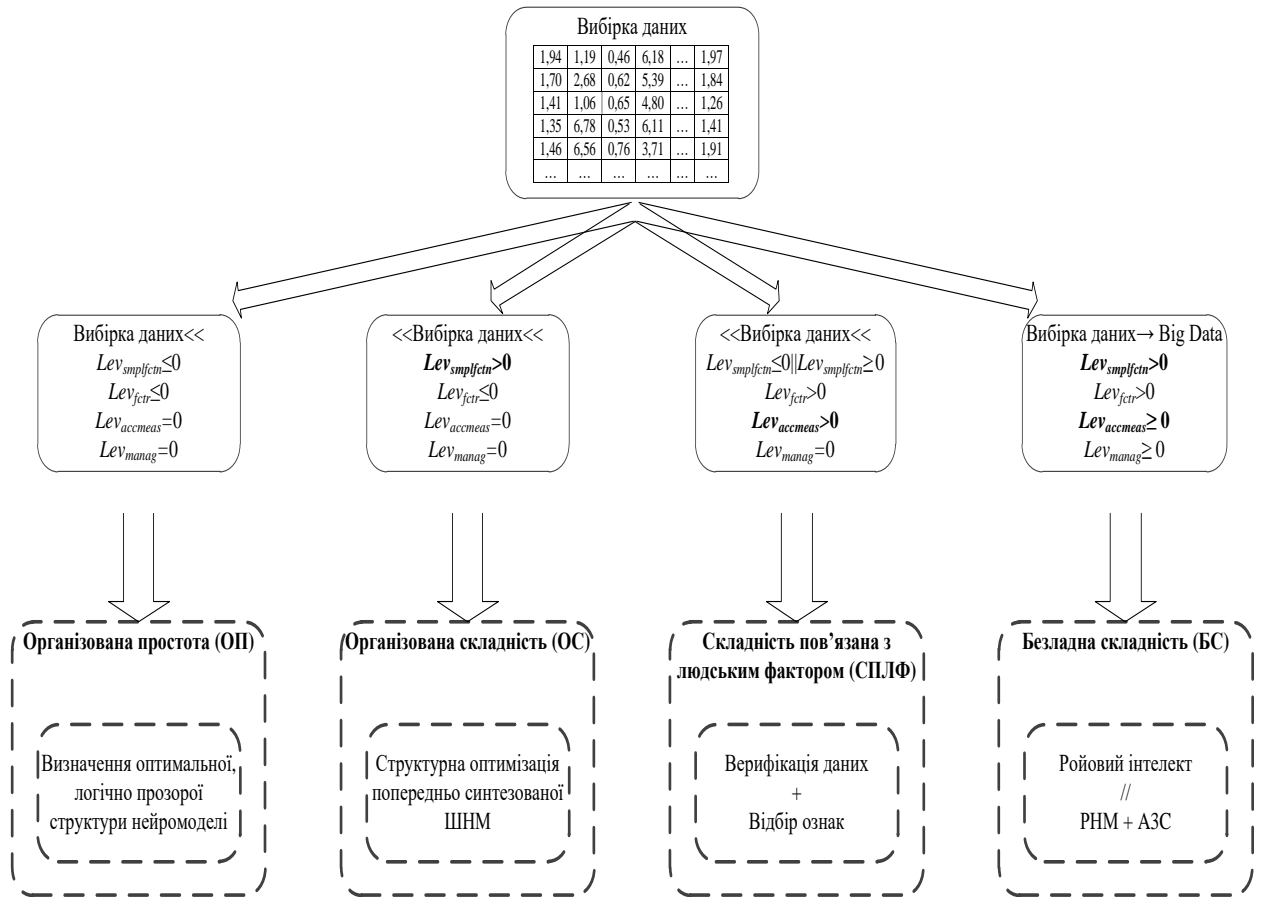


Рисунок 3.1 – Загальна схема визначення рівню складності

Далі визначимо рівень можливого спрощення структури ( $Lev_{smpifctn}$ ). Зазвичай будемо говорити, про загальну можливість видалення мало інформаційних вхідних даних. Так  $Lev_{smpifctn} = -1$ , якщо кількість вхідних ознак не велика (за замовченням, пропонується до 200, але можливо збільшення//зменшення з огляду на технічні ресурси, що використовуються) та спрощення не можливе,  $Lev_{smpifctn} = 0$ , якщо кількість велика (від 200 до 1000), але можливе їх скорочення та  $Lev_{smpifctn} = 1$ , якщо кількість велика та скорочення не можливе (при роботі в онлайн режимі) або занадто ресурсоємне.

Дослідження загальної кількості суттєвих та менш суттєвих та/або несуттєвих факторів вимагає попереднього проведення факторного аналізу. З огляду на спрощення цього етапу пропонується виконувати звичайний факторний аналіз методом головних компонент [148]–[151]. Так  $Lev_{fctr} = -1$ , якщо проведення факторного аналізу засвідчило невелику кількість суттєвих факторів;  $Lev_{fctr} = 0$ , якщо кількість суттєвих факторів невелика, але є певна кількість менш суттєвих факторів або ж під час проведення факторного аналізу зустрічалися кореляційні матриці, що були погано обумовлені (їх детермінант дорівнював 0) та  $Lev_{fctr} = 1$ , якщо факторний аналіз засвідчив велику кількість суттєвих факторів та/або велику кількість менш суттєвих [151].

При визначенні рівню точності вимірювань будемо керуватися двома основними випадками [149], [151]:

- $Lev_{accmeas} = 1$ , якщо вимірювання пов'язані із людським фактором, тобто люди (або людські сенсорні системи) стають системами спостережень;
- $Lev_{accmeas} = 0$ , якщо вимірювання відбуваються з використанням спеціальних сенсорів (систем сенсорів).

Рівень можливого контролю та управління також може демонструвати виключно два стани, так:

- $Lev_{manag} = 1$  – якщо система демонструє у високому ступені випадкову, безладну поведінку або така поведінка є результатом відсутності необхідного контролю над структурою складних неоднорідних систем;
- $Lev_{manag} = 0$  – якщо система знаходилась під контролем (людським чи автоматизованим) та навіть у випадку ризької зміни певних ознак (наприклад деформації) була під наглядом.

Слід зазначити, що якщо система, наприклад, після оцінювання задачі її складність визначається, як ОС, то на початку рекомендується синтезувати нейромоделі (наприклад, з використанням нейроеволюційних методів), а вже

потім проводити структурну оптимізацію, якщо в цьому є необхідність. Це пояснюється тим, що в такому випадку провести відбір інформаційних ознак не можливо за умовами задачі, а от спростити та виключити зв'язки в моделі – можливо. Якщо ж оцінка визначається, як СПЛФ, то перед відбором інформаційних ознак рекомендується перевірити істинність вхідних даних. У видку БС для синтезу нейромоделі рекомендується використовувати особливий клас нейроеволюційних методів, а саме методи PSO або ж використовувати топологію РНМ з методом Actor-Critic method (А3С) [152]–[154] для її навчання. Такий підхід дозволить значно краще опрацювати дані, адже методи ройового інтелекту позбавлені недоліків більшості нейроеволюційних методів та емулюють поведінку незалежних агентів, а РНМ дозволяє закодувати у своїй структурі більшість прихованих взаємозв'язків [75].

Таким чином, запропоновано систему індикаторів, яка, використовуючи дані про задачу та характеристики вхідних даних, дозволяє визначити рівень складності задачі для подальшого вибору доопрацювання задля найбільш оптимальної структури нейронної моделі для подальшого моделювання системи [151]. Такий підхід дозволяє скоротити час та підвищити рівень логічної прозорості нейромоделі, що значно розширює практичне використання таких моделей, без використання нейроеволюційних методів, що можуть бути не виправдано ресурсоемними при ряді задач. У якості індикаторів запропоновано використовувати [149]:

– рівень можливого спрощення структури, що дозволяє визначити на основі об'єму вхідних даних чи можливо подальше їх скорочення (відбір інформаційних ознак) [148];

– загальна кількість суттєвих та менш суттєвих та несуттєвих факторів, використання якого засвідчує наявність об'єднання ознак за факторами впливу на систему (або об'єкт) та в подальшому відсіювання не важливих [149];

– рівень точності вимірювань, що засвідчує довіру до накопичених історичних даних, що в подальшому також може свідчити про наявність зачумлених даних які впливають на параметри нейромоделі [151];

– рівень можливого контролю та управління, що також свідчить про ймовірність втручання в набір початкових даних і зниження їх репрезентативності [150].

### **3.2 Імплементация властивості нейропластичності у вигляді механізмів прорідження штучних нейронних мереж**

На сьогоднішній день, практичні фахівці можуть використовувати можливості ШНМ для діагностування, контролю, прогнозування та попереднього аналізу вхідних даних, зокрема для очищення біологічних сигналів від шумів, виділення корисних даних з різноманіття наявних за певними критеріями [58]–[69].

Однак, коли у якості вхідних даних для навчання, тестування роботи ШНМ або навіть вже при роботі використовуються великі дані це значно ускладнює обробку даних у реальному часі [155]–[159]. Достатньо часто це пояснюється надмірною структурною складністю ШНМ. Адже при нелінійності систем та розподіленості інформаційних мереж в діагностуванні використовують більш складні архітектури ШНМ: рекурентні ШНМ (LSTM, Gated Recurrent Units), ГНМ (CNN, Residual Network) [159]. За рахунок наявності зворотних зв'язків або декількох шарів прихованих нейронів такі мережі значно підвищують тоничність діагностики, проте значно втрачають у швидкості роботи.

Кілька досліджень підтвердили ефективність різних підходів до стиснення ваг [129]–[137], потрібних для навчання моделі. Ці підходи відкривають важливі можливості: використання ШНМ для обробки мультимедійної інформації та великих даних у реальному часі. Існують різні стратегії пошуку оптимальної структури мережі: поступове нарощування,

побудова свідомо занадто складної мережі з подальшим спрощенням, почергове нарощування і спрощення [160], [161]. Проблема вибору структури тісно пов'язана з проблемами недонавчання і перенавчання. Занадто прості мережі не здатні адекватно моделювати цільові залежності в реальних завданнях. Занадто складні мережі мають надлишкове число вільних параметрів, які в процесі навчання налаштовуються не тільки на відновлення цільової залежності, але і на відтворення шуму [162], [163].

Останні дослідження [146]–[148] довели, що впродовж сну (особливо, глибокого сну) у людському мозку відбуваються процеси очищення, або «brainwashed». Загалом можна відмітити, що дослідження довели, що під час сну мозок проводить процеси впорядкування набутої інформації за рахунок скорочення, видалення або додавання нових зв'язків між нейронами. Роботи [146]–[148] засвідчили, що під час сну нейрони зменшуються в розмірах і між ними виникають простору, які заповнюються мозковою рідиною. Результати [146]–[148] засновані на відкритті так званої глімфатичної системи, яка діє в мозку спеціально для видалення шкідливих речовин. Клітини мозку-можливо, гліальні клітини, які оточують і підтримують нейрони, зіщулюються під час сну. Це призводить до збільшення міжклітинного простору в речовині мозку, що в свою чергу підсилює приплив рідини, яка виносить з мозку токсини. Тож, припущення про те, що схожі методи оптимізації структури штучних ШНМ зможуть вирішити ряд існуючих проблем не є безґрунтовним.

Оптимальне проріджування нейронних мереж (Optimal brain surgeon – OBS) – метод спрощення структури регресійної моделі, наприклад, нейронної мережі [164], [165]. Основна ідея OBS полягає в тому, що ті елементи моделі або ті ШНМ, які мають малий вплив на помилку апроксимації, можна виключити з моделі без значного погіршення якості апроксимації.

Метод другого порядку (що використовує аналіз чутливості, заснований на обчисленні других похідних) був запропонований Л. Кюном і

називався «optimal brain damage» [164]. Потім він був розвинений Б. Хассібі і отримав назву «optimal brain surgery» [164]. Крім завдання видалення елементів вирішувалися також інші проблеми спрощення: зменшення розрядності ваг і сигналів (огрубіння), спрощення функцій активації нейронів, отримання інтерпретованого знання і т.д. Вся сукупність підходів отримала також назву «контрастування нейронних мереж». Для підготовки нейронної мережі до спрощення виявляється корисним ввести в оцінку її роботи, що мінімізується при навчанні, штрафні доданки, штрафують за складність.

Детально представити метод OBS можна наступним чином [165]. Нехай задана вибірка  $Sample$ , модель  $f(w, \varepsilon)$ , де  $w$  – сукупність параметрів регресійної моделі,  $\varepsilon$  – незалежна змінна та функція помилки  $E_{Sample}$ . Для спрощення структури регресійної моделі виконуються такі кроки [164]:

- 1) налаштовується модель, отримуються параметри  $w^* = \arg \min (E_{Sample}(w | f, Sample))$ ;
- 2) для прирощення  $w^* + \Delta w$  вирішується оптимізаційна задача, знаходяться для кожного індексу  $i$  мінімальне значення Лагранжиана  $L_i$ ;
- 3) обирається серед  $L_i$  мінімальне, відсікаємо елемент моделі, відповідний  $i$ -му параметру;
- 4) додається до вектора параметрів  $w^*$ , вектор прирощень  $\Delta w$ , відповідний відсіченому параметру;
- 5) отримується спрощена модель. Модель переналаштовувати не потрібно;
- б) процедуру можна повторювати до тих пір, поки значення помилки не перевершить заздалегідь задане.

Регуляризація в машинному навчанні – метод додавання деяких додаткових обмежень до умови з метою вирішити некоректно поставлене завдання або запобігти перенавчанню [166], [167]. Ця інформація часто має вигляд штрафу за складність моделі. Наприклад, це можуть бути обмеження



гладкості результуючої функції або обмеження за нормою векторного простору. З байєсівської точки зору багато методів регуляризації відповідають додаванню деяких апіорних розподілів на параметри моделі. Найчастіше виділяють:

1)  $L_1$ -регуляризацію, або регуляризація лассо-регресією [166]:

$$L_1 = \sum_{i=1}^N (y_n - \hat{y}_n)^2 + \lambda \|w\|_1;$$

2)  $L_2$ -регуляризацію, або регуляризація регресією Ріджа [167]:

$$L_2 = \sum_{i=1}^N (y_n - \hat{y}_n)^2 + \lambda \|w\|_2^2.$$

В основі  $L_1$ -регуляризації лежить досить проста ідея [166]: як і у випадку  $L_2$ -регуляризації, ми просто додаємо штраф до початкової функції витрат. Подібно до того, як в  $L_2$ -регуляризації ми використовуємо  $L_2$ -нормування для поправки вагових коефіцієнтів, в  $L_1$ -регуляризації ми використовуємо спеціальне  $L_1$ -нормування.  $L_2$ -регуляризація допомагає вирішити проблему перенавчання моделі [167]. Суть в тому, що надто важкі вагові коефіцієнти відштовхують лінію найкращої відповідності, побудовану на основі мінімізації квадрата помилок, від основної тенденції.  $L_1$ -регуляризація сприяє розрідженості функції, коли лише деякі фактори не дорівнюють нулю.  $L_2$ -регуляризація сприяє появі малих вагових коефіцієнтів моделі, але не сприяє їх точній рівності нулю. Це пояснюється тим, що:

–  $L_1$ -регуляризація реалізує це шляхом відбору найбільш важливих факторів, які найсильніше впливають на результат [167]. Для простоти можна

вважати, що фактори з малою величиною впливу на кінцевий результат фактично допомагає передбачати лише шум в наборі навчальних даних;

–  $L_2$ -регуляризація запобігає перенавчанню моделі шляхом заборони на непропорційно великі вагові коефіцієнти.

Нечутливі до ваг нейронні мережі (WANN) – це архітектура ШНМ, представлена розробниками Google, здатна імітувати вроджені інстинкти і рефлекси живих істот, з подальшим донавчанням протягом життя [168]–[171]. А також може значно зменшує кількість зв'язків всередині мережі, підвищуючи тим самим їх швидкодію [170].

Ідея WANN [169] полягає в тому, щоб замінити всі ваги ШНМ однією – загальною вагою. І в процесі навчання підбирати не ваги між нейронами, як в звичайних нейромережах, а підбирати структуру самої мережі (кількість і розташування нейронів), яка з однаковими вагами показує найкращі результати. А після ще й оптимізувати її, щоб мережа добре працювала з усіма можливими значеннями цієї загальної ваги (загальним для всіх з'єднань між нейронами [169]). У підсумку це дає структуру нейронної мережі, яка не залежить від конкретних значень ваг, а працює добре з усіма. Тому що працює за рахунок загальної структури мережі. Це схоже на ще не ініціалізований конкретними вагами мозок тварини при народженні, але вже містить вшиті інстинкти за рахунок своєї загальної структури [170]. А подальше тонке налаштування ваг під час навчання протягом життя, робить цю ШНМ ще краще.

Загальна схема роботи методу дуже схожа на нейроеволюційні методи синтезу ШНМ [170], [171]:

- 1) створюється популяція простих ШНМ;
- 2) кожна мережа ініціалізує всі свої ваги одним числом, причому для широкого діапазону чисел:  $w = [-2; 2]$ ;
- 3) отримані мережі сортуються за якістю рішення задачі і за кількістю нейронів (у бік зменшення);

4) у частині кращих представників додається один нейрон, один зв'язок або змінюється функція активації в одному нейроні;

5) ці модифіковані мережі використовуються в якості початкових для нової ініціалізації – так метод повертається до першого кроку.

У таблиці 3.1 наведено порівняння основних методів оптимізації структури ШНМ.

Таблиця 3.1 – Порівняння сучасних методів структурної оптимізації ШНМ

Критерії	OBS	Регуляризація ( $L_1, L_2$ )	WANN
Рівень теоретичного обґрунтування методу	Високий	Високий	Середній
Вимоги до додаткових обрахунків мета-параметрів	Високі	Високі	Середні
Збільшення апроксимаційних властивостей	В певній мірі забезпечується	В певній мірі забезпечується	В певній мірі забезпечується
Робота при специфічних задачах	Ускладнена	Неможлива	Можлива
Робота із складними топологіями ШНМ	Неможлива	Неможлива	Ускладнена

Як видно із таблиці 3.1 порівняння усіх сучасних методів структурної оптимізації ШНМ мають низку недоліків. Так OBS та методи регуляризації мають хороше теоретичне обґрунтування та використовуються вже достатньо давно, проте слід зазначити, що ці методи вимагають додаткових обрахунків мета-параметрів про ШНМ [171], [172].

Більш того, варто зазначити, що ці обрахунки значно підвищують ресурсоємність: значно підвищуються вимоги до пам'яті та обчислювальних потужностей – у майбутньому це може призвести до неможливості або слабкої продуктивності паралельних модифікацій цих методів [172]. Також

важливо пам'ятати про складність застосування регуляризації для специфічних задач, адже для цих методів дійсно важлива – похідна функції.

Так, знаходження похідної – є постійною задачею, оскільки методи базуються на градієнтному спуску, який в основному рухається в напрямку похідної [168]. При квадратичному члені чим ближче величина знаходиться до нуля, тим меншою стає похідна, поки також не наблизиться до нуля. Тому в такій ситуації зміни при регуляризації будуть неістотними. Найважливішою ж рисою є те, що OBS та методи регуляризації майже неможливо застосовувати для структурної оптимізації ШНМ із складними структурами: рекурентні ШНМ зі зворотними зв'язками або глибокі ШНМ із декілька ми прихованими шарами нейронів [165]. З іншого боку, WANN виграють через те, що можуть поступово змінювати топологію ШНМ та налаштовувати її відповідно до задачі. Також слід відзначити значну перевагу WANN за рахунок того, що немає необхідності зберігати та обраховувати мета-параметри мережі. Проте через недостатнє теоретичне обґрунтування ця технологія поки що нерозповсюджена.

### **3.3 Метод структурної оптимізації штучних нейронних мереж на основі генетичних операторів**

Нейроеволюційні методи передбачають використання специфічних генетичних операторів. В ряді робіт пропонувалося використовувати адаптивні механізми мутації [64], [111], [112], [120]. Загалом визначимо основні види мутацій, що можуть використовуватися при синтезі ШНМ [103]–[107]:

- 1) додавання прихованого нейрона. Новий нейрон додається разом з вхідним і вихідним зв'язками. При цьому вихідний зв'язок нейрона не може його зв'язувати із вхідним нейроном [105];

- 2) видалення випадково обраного прихованого нейрона разом з усіма вхідними і вихідними зв'язками. При цьому якщо утворюється пробіл в

індексах нейронів, що залишилися, то проводиться корекція індексів. Вхідні та вихідні нейрони мережі не можуть бути видалені [106];

3) додавання зв'язку. Випадковим чином визначаються індекси початкового і кінцевого нейронів в ШНМ, представленої мутуючої особиною. При цьому зв'язок не може закінчуватися вхідним нейроном. Вага зв'язку визначається також випадково з діапазону  $[-0,5; 0,5]$ :  $w_i = \text{Rand}[-0,5; 0,5]$ . Якщо в мережі вже існує зв'язок з аналогічними вхідними і вихідними нейронами, то його вага замінюється на випадкову [107];

4) видалення випадково обраного зв'язку. При цьому може виникнути ситуація, коли видаляється останній зв'язок у прихованого нейрона. У цьому випадку нейрон також видаляється, і, якщо необхідно, проводиться корекція індексів нейронів мережі [111];

5) зміна функції активації випадкового прихованого нейрону [113].

Таким чином, за допомогою мутації можна точково змінювати параметри структури ШНМ.

Зазначимо, що таким чином схожі механізми можуть використовуватися для оптимізації структури попередньо синтезованих ШНМ. Саме тому пропонується використовувати адаптивний механізм мутації проте з певними додатковими установками [173]. Хаотичне додавання (видалення) нейронів і зв'язків може призвести до ситуацій, коли, наприклад, в мережі багато нейронів і мало зв'язків. Більш логічним буде застосовувати різні види мутацій в залежності від особливостей архітектури мережі, поданої мутуючої особиною. Для цього буде введено критерії та індикатори, для оцінювання та регуляції розміру та напрямку доналаштування мережі [168], [169], [173].

Використання першого критерію засновано на припущенні, що чим більше елементів в сумі у вхідному та вихідному векторі навчальної вибірки (чим більшою є сумарна кількість вхідних і вихідних нейронів), то, ймовірно, більш складна мережа необхідна для вирішення поставленого завдання –

тобто це повторює зміст критерію (2.3), але в цьому випадку введемо для нього нове позначення [120]:

$$f_{\text{умов.склад}} = \frac{N_i + N_o}{N_s}.$$

Перший же індикатор буде використовуватися для оцінки надмірності використаної пам'яті і повторювати зміст критерію з (2.9), але і в цьому випадку введемо для нього нове позначення:

$$ind_{\text{memor}} = crit_m = \frac{W_{FFc} + W_{FBc}}{samp_{Inst} \cdot samp_{Feat}}. \quad (3.1)$$

Логічна прозорість мережі в значній мірі залежить від загальної кількості зв'язків в мережі і кількості зв'язків, що з'єднують конкретні нейрони (кількості входів нейронів прихованих шарів). Чим менше зв'язків, тим простіше мережа і тим зручніше вона для аналізу та інтерпретації людиною [173]. Тож індикатор розрідженості зв'язків мережі прямого поширення визначимо за формулою (3.2):

$$ind_{\text{distrat}} = \frac{W_{w=0}}{\sum_{\mu=1}^L NumN_{(\mu-1)} NumN_{\mu}}, \quad (3.2)$$

$$\text{де } 0 \leq W_{w=0} \leq \sum_{\mu=1}^L NumN_{(\mu-1)} NumN_{\mu};$$

$W_{w=0}$  – ваги зв'язків, що дорівнюють нулю;

$L$  – кількість шарів у ШНМ;

$NumN_{\mu}$  – кількість нейронів у  $\mu$ -му шарі ШНМ.

Індикатор середньої зв'язаності нейронної мережі показує середню кількість входів нейронів всіх шарів, крім першого за формулою (3.3) [168]:

$$ind_{degrconn} = \frac{1}{\sum_{\mu=2}^L NumN_{\mu}} \sum_{\mu=2}^L \sum_{i=1}^{NumN_{\mu}} \left( NumW^{(\mu,i)} - NumW_{w=0}^{(\mu,i)} \right), \quad (3.3)$$

де  $NumW^{(\mu,i)}$  – кількість ваг  $i$ -го нейрона у  $\mu$ -му шарі ШНМ;

$NumW_{w=0}^{(\mu,i)}$  – кількість ваг  $i$ -го нейрона у  $\mu$ -му шарі ШНМ, що дорівнюють нулю.

Індикатор логічної прозорості  $i$ -го нейрона  $\mu$ -го шару мережі  $ind_{activ}^{(\mu,i)}$  буде визначатися типом використовуваної функції активації. Для лінійної та порогової функції активації приймемо:  $ind_{activ}^{(\mu,i)} = 1$ , для всіх інших функцій активації  $ind_{activ}^{(\mu,i)} = 0$ .

Виходячи з цього, коефіцієнт логічної прозорості ШНМ  $ind_{logtransp}$  будемо розраховуватися за (3.4) [173]:

$$ind_{logtransp} = \frac{\sum_{\mu=1}^L \sum_{i=1}^{NumN_{\mu}} ind_{activ}^{(\mu,i)}}{ind_{degrconn} \sum_{\mu=1}^L NumN_{\mu}}, \quad ind_{degrconn} \neq 0. \quad (3.4)$$

На рис. 3.2 наведено у схематичному вигляді процес вибору типу мутації. На рис. 3.2 використовуються такі позначення:  $Val_1$  та  $Val_2$  – певні граничні значення, що визначаються для завдання. Умовно процес вибору типу мутації можна розділити на дві гілки по першому умовному переходу.

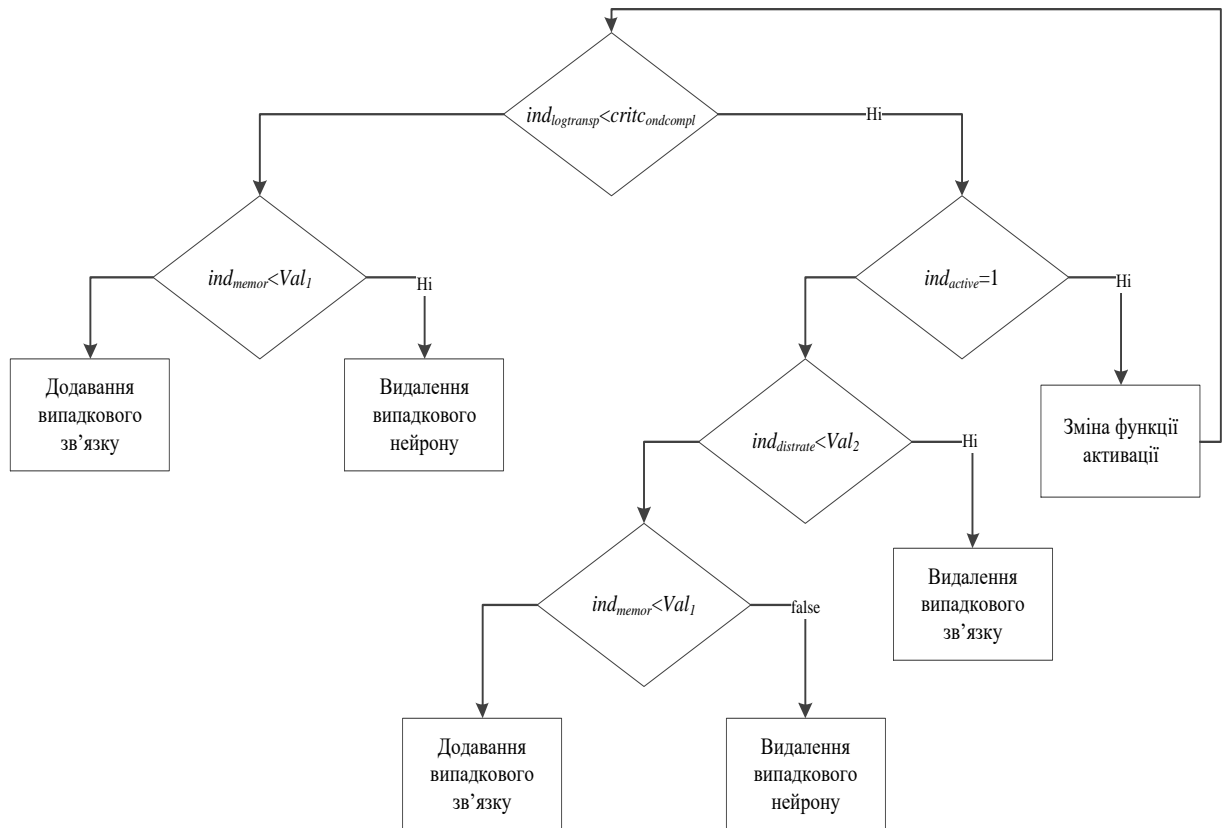


Рисунок 3.2 – Схематичне зображення вибору типу мутації

Гілка зменшення  $ind_{logtransp}$  з (3.4), виконується по виконанню умови переходу та гілка збільшення  $ind_{logtransp}$ , виконується при невиконанні умови переходу.

### 3.4 Висновки за розділом 3

У цьому розділі розв'язано актуальне завдання вибору механізмів для доналаштування ШНМ на основі індикаторної системи для оцінки рівня складності задачі. Запропоновано метод структурного доналаштування синтезованих ШНМ на основі нейроеволюційного підходу з використанням системи індикаторів та критеріїв для адаптивного визначення мутаційних змін.

Вперше запропоновано систему індикаторів для оцінювання рівня складності задачі для подальшого моделювання, діагностування або



прогнозування. Так враховуючи характеристики вхідного набору даних, рівень можливого спрощення структури, загальну кількість суттєвих та несуттєвих факторів, рівень точності вимірювань та рівень можливого контролю та управління можливо в подальшому обрати варіант доналаштування математичної моделі. Такий підхід дозволяє попередньо проаналізувати та визначити механізми доналаштування ШНМ без використання надмірних обчислювальних ресурсів для повторного синтезу нейромоделей, таким чином, запропоновані індикатори не вимагають додаткової побудови моделі, що істотно знижує використання часових та обчислювальних ресурсів під час процесу аналізу варіантів підбору та розвитку математичної моделі.

Вперше запропоновано метод структурного доналаштування для підвищення характеристик нейропластичності ШНМ. Так, під час роботи із великими даними застосування складних топологій ШНМ часто обмежується обчислювальними ресурсами. Саме тому структурна оптимізація дозволяє отримати полегшену та проріджену структуру попередньо синтезованої нейромоделі. Це гарантує не тільки зменшення навантаження на обчислювальну систему, а й пришвидшення роботи такої ШНМ. Більш того, у якості побічного ефекту структурної оптимізації іноді може спостерігатися підвищення точності роботи, адже частина вузлів та зв'язків для кодування зашумлених даних може бути видалена.

Результати, наведені у третьому розділі, було опубліковано в роботах [151] та [173].

## **4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ МЕТОДІВ СИНТЕЗУ РЕКУРЕНТНИХ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ ДІАГНОСТУВАННЯ СКЛАДНИХ ТЕХНІЧНИХ ОБ'ЄКТІВ ТА СИСТЕМ**

В попередніх розділах було з'ясовано, що розв'язання завдань технічного діагностування нелінійних технічних об'єктів та систем пов'язано з необхідністю синтезу діагностичних моделей, що дозволяють класифікувати екземпляри з високою точністю та відрізняються високим рівнем логічної прозорості для експертів прикладних областей.

У попередніх розділах запропоновано методи, які дозволяють розв'язувати задачі синтезу і структурної оптимізації діагностичних моделей на основі рекурентних нейронних мереж. Метою цього розділу є експериментальне дослідження розроблених методів аналізу даних та синтезу діагностичних моделей на основі рекурентних мереж.

### **4.1 Умови та особливості проведення експериментальних досліджень методів синтезу рекурентних нейромережових моделей**

Для перевірки та визначення ефективності розроблених методів синтезу діагностичних моделей на основі рекурентних мереж розв'язувалися різні практичні завдання діагностування та розпізнавання, що характеризувалися різними властивостями навчальних вибірок даних  $Sample = \langle X, Y \rangle$ .

Для проведення експериментального дослідження використовувалося устаткування кафедри програмних засобів Національного університету «Запорізька політехніка». При дослідженні паралельних методів використовувалися засоби програмної бібліотеки Message Passing Interface (MPI) [174].

Запропоновані методи досліджувалися у порівнянні з існуючими методами, що дозволяють розв'язувати аналогічні завдання синтезу та навчання ШНМ та РНМ. Як критерії порівняння було виішено використовувати критерії, що дозволяють оцінити ефективність методів для розв'язання відповідних задач, зокрема використовувалися критерії оцінювання часу роботи методів, кількості ітерацій, точності синтезованих моделі та ін.

#### **4.2 Експериментальне дослідження послідовного методу синтезу штучних рекурентних нейронних мереж**

Для експериментального дослідження ефективності використання запропонованого методу МГА для РНМ було проведено експерименти, які ґрунтувалися на розв'язанні задач технічного та медичного діагностування [175], [176].

При вирішенні задачі технічного діагностування лопаток компресора газотурбінного двигуна, в якості об'єкта досліджень обрані лопатки I ступеня компресора двигуна ТВ3-117, які мають експлуатаційне пошкодження пера лопаток двигунів [176]. У дослідженнях спостерігалися двигуни були в експлуатації в різних країнах, відповідно фізичні характеристики експлуатаційних процесів відрізнялися. З цього можна зробити висновок, що авіаційні деталі мають різне напруження і, відповідно, різний ступінь пошкодження лопаток. Дані авіадвигуни експлуатувались в різних кліматичних зонах таких країн, як Ємен, Індія, ОАЕ, Перу, Кіпр, Алжир, Іспанія [176]. Таким чином, загалом задачу технічного діагностування в такому разі можна навести у вигляді схеми на рис. 4.1 та рис. 4.2. У таблиці 4.1 наведено приклад фрагменту вибірки вхідних даних.

З таблиці 4.1 видно, що головними інформаційними ознаками у вибірці є:

– країна (підприємство) експлуатації;

- середня температура місцевості;
- геометричні та фізичні показники: величини хорди в різних перетинах та товщина вхідний кромки;
- твердість вихідної лопатки;
- межа плинності вихідного матеріалу;
- межа міцності;
- величини напрацювань.

Залежною ж величиною виступає – кількість власних коливань (КВК)  
– на основі якої можна встановити чи справний, чи дефектний виріб.

Таблиця 4.1 – Фрагмент вибірки вхідних даних про лопатки I ступеня компресора двигуна ТВЗ-117, що мають експлуатаційне пошкодження пера лопаток двигунів

Країна	Середня температура	Перетин А2-А2			Перетин А3-А3			Перетин А4-А4		...	КВК
		$V_{2-2}$	$C_{1_{2-2}}$	$C_{2_{2-2}}$	$C_{1_{3-3}}$	$C_{2_{3-3}}$	$V_{3-3}$	$C_{1_{4-4}}$	$C_{2_{4-4}}$		
$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	...	$y_1$
Індія-1	24,6	26,5	1,33	0,64	27,0	0,58	28,0	0,97	0,58	...	590,0
Індія-2	24,6	27,1	1,35	0,66	27,6	0,60	28,6	0,99	0,60	...	620,0
Індія-3	24,6	25,9	1,31	0,62	26,4	0,56	27,4	0,95	0,56	...	650,0
Індія-4	24,6	26,5	1,39	0,54	26,9	0,46	27,6	0,89	0,46	...	647,3
Індія-5	24,6	26,6	1,37	0,68	26,8	0,62	27,6	0,90	0,62	...	646,7
Індія-6	24,6	26,4	1,53	0,60	26,8	0,54	27,5	0,98	0,54	...	642,4
Індія-7	24,6	26,5	1,25	0,62	26,9	0,51	27,8	0,81	0,51	...	643,1
Індія-8	24,6	26,7	1,35	0,57	27,1	0,45	27,8	0,91	0,45	...	648,5
Індія-9	24,6	26,7	1,57	0,70	27,1	0,62	27,8	0,99	0,62	...	668,6
Індія-10	24,6	26,2	1,46	0,62	26,7	0,57	27,2	0,96	0,57	...	645,2
...	...	...	...	...	...	...	...	...	...	...	...
Йемен-20	20,5	26,6	1,20	0,59	27,0	1,15	0,44	27,9	0,95	...	627,4

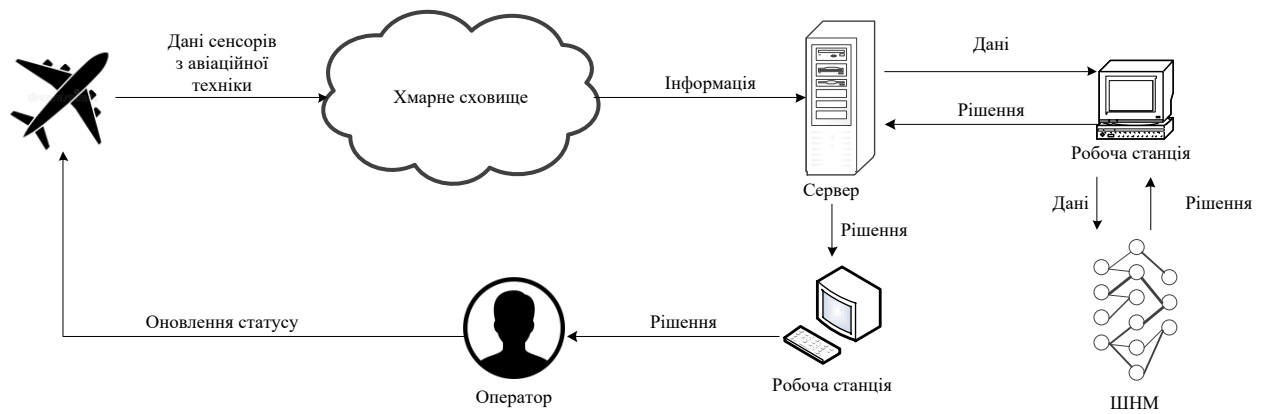


Рисунок 4.1 – Діаграма впровадження нейромоделі в систему діагностування авіаційних деталей

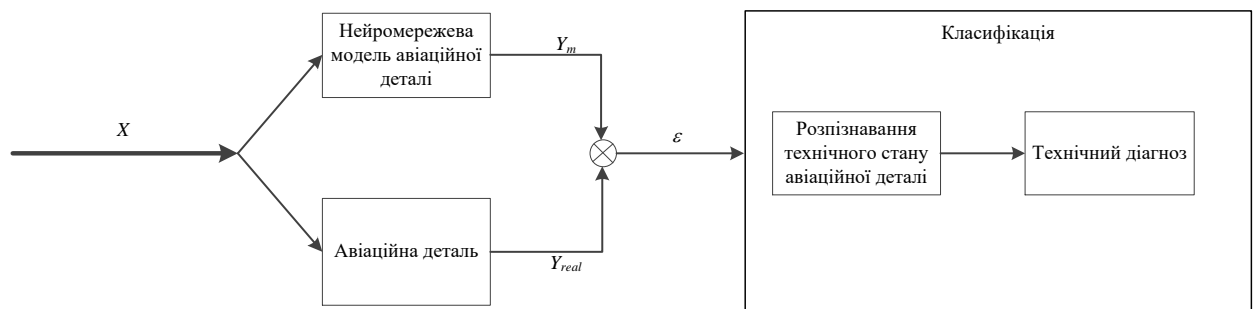


Рисунок 4.2 – Узагальнена діаграма діагностування авіаційної деталі

Дослідження запропонованого методу МГА для РНМ відбувалося у порівнянні з відомими моделями та методами їх синтезу: регресійними моделями [175], [176], багатошаровий перцептрон та метод зворотного поширення помилки (БП та МЗПП) [176], РНМ та метод зворотного поширення помилки в часі (РНМ та МЗППвЧ), РНМ та метод синтезу NEAT (РНМ та NEAT).

У якості показників для визначення ефективності роботи методів синтезу нейромоделей, що були розроблені використовувалися:

- час роботи – час витрачений на синтез РНМ;
- точність роботи на навчальній вибірці – точність роботи моделі під час навчання;

– точність роботи на тестовій вибірці – точність роботи моделі під час тестування.

Результати експериментів для порівняння методів синтезу нейромоделей для порівняння наведено в табл. 4.2.

Таблиця 4.2 – Порівняння експериментально отриманих результатів роботи методів синтезу (навчання) нейромоделі для технічного діагностування на базі РНМ

Метод синтезу (навчання) моделі	Значення критеріїв порівняння		
	Час роботи, с	Точність на початковій вибірці	Точність на тестовій вибірці
Регресійні моделі	156	0,035	0,058
БП та МЗПП	637	0,03	0,035
РНМ та МЗППвЧ	2234	0,027	0,03
РНМ та NEAT	3782,5	0,017	0,02
МГА для РНМ	3502,7	0,016	0,018

Порівнюючи час роботи можна відзначити, що регресійні моделі були синтезовані найбільш швидко. Це пояснюється простотою самої методології. В той же час, інші підходи, що у якості основи моделі використовували нейроммержі різної складності, виявилися значно повільнішими. Більш того, варто зазначити, що нейроеволюційні методи синтезу (група з РНМ та NEAT й МГА для РНМ) виявилися повільнішими від методів заснованих на зворотному поширенні помилки (група з БП та МЗПП й РНМ та МЗППвЧ), що також може бути пояснено ітераційністю підходів. Методи нейроеволюційного підходу відпрацювали повільніше, що пояснюється тим, що вони працювали в однопотоковому режимі і повністю синтезували нову архітектуру мережі, оперуючи популяцією з нейроммержевих моделей. Загалом же можна зробити підсумок, що залежно від складності топології моделі час її синтезу буде збільшуватися, саме тому більш доцільно

придивитися під час порівняння до комбінації: РНМ та МЗППвЧ, РНМ та NEAT й МГА для РНМ. В такій підбірці МГА для РНМ виявився другим за швидкістю при синтезі однаково складних топологій (на 7,4% швидше від РНМ та NEAT).

А ось порівнюючи точність роботи синтезованих рішень на навчальних та тестових вибірках можна прийти до висновків, що саме використання більш складних топологій для діагностичних моделей забезпечує більш точніші результати. Так методи, що оперують нейромережевими моделями (БП та МЗПП, РНМ та МЗППвЧ, РНМ та NEAT, МГА для РНМ) виявилися значно точнішими на обох наборах даних ніж регресійні моделі. Більш того, такі методи не відрізняються такими ризькими відмінностями на навчальних та тестових вибірках. Проте БП та МЗПП продемонстрували так само порівняно не високий рівень точності, що можна пояснити відсутністю зворотних зв'язків для збереження попереднього технічного стану об'єкта. РНМ, що була синтезована за допомогою МГА для РНМ, виявилися найточнішою при порівнянні результатів.

З цього можна зробити висновок, що в простих завданнях нейроеволюційним методам можуть знадобитися підвищені часові ресурси. При цьому більш висока точність синтезованого рішення (яка підтвердилася експериментально) може не виправдати повною мірою такі витрати часу.

Однак, в складних завданнях, коли процес препроцесингу вхідних даних не можливий або значною мірою утруднений і вкрай важлива точність моделі, нейроеволюційні методи можуть показати велику ефективність. Це пояснюється меншою залежністю роботи таких методів від зашумленості вхідних даних, а так само пропорційно збільшуються тимчасовими витратами при навчанні складних топологій за допомогою ітераційних методів.

Наступною задачею для дослідження ефективності МГА для РНМ була задача діагностування рівня родючості ґрунту для визначення схем зрошення [177]. Щоб розробити ефективні технології для схем зрошення,

слід спочатку провести моделювання врожайності сільськогосподарських культур за рахунок крапельного зрошення. Для збору даних було проаналізовано 136 ділянок сільськогосподарського виробництва в Запорізькій та Херсонській областях за 2010-2020 роки, щоб виявити нестачу вологи для розвитку сільськогосподарських рослин [177]. Так, у тих випадках, коли спостерігається дисбаланс, необхідно проводити крапельне зрошення з найближчого басейну [177].

Використовуючи сенсори та спеціалізоване обладнання, вимірюється глибина проникнення вологи в ґрунт. Згідно з нормами на 1 рослину соняшнику потрібно  $0,13 \text{ м}^3$  вологи для 1 циклу (при температурі) [177]. Залежно від типу ґрунту, типу рослин, клімату, навичок фермерів, кількості шарів ґрунту, коефіцієнтів поглинання ґрунту змінюється просочення вологою на 1 м глибини ґрунту. Загальне представлення задачі наведено на рис. 4.3. У додатку А.1 міститься приклад фрагменту вибірки.

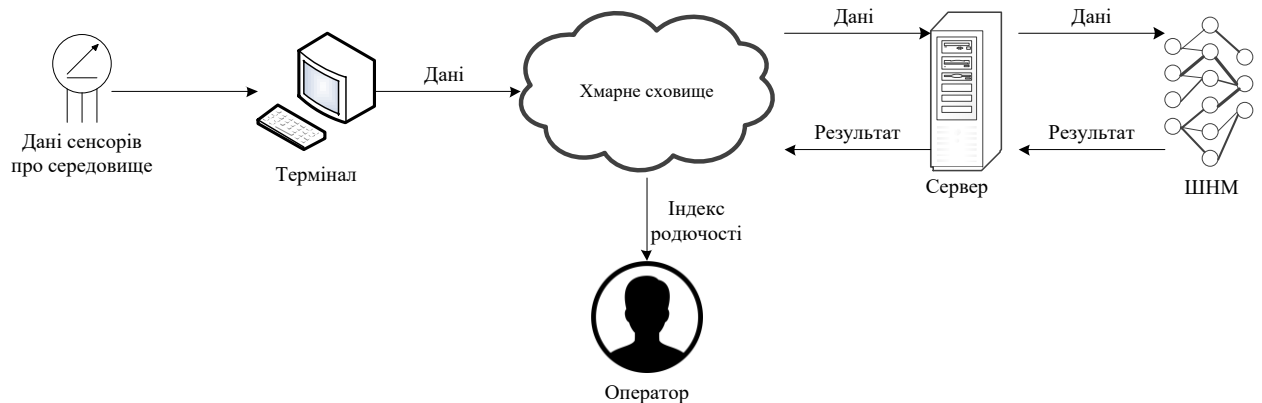


Рисунок 4.3 – Діаграма впровадження нейромоделі в систему діагностування рівня родючості ґрунту для визначення схем зрошення

Для дослідження ефективності МГА для РНМ буде проведено порівняння результатів із іншими методами, що використовуються для синтезу та навчання саме РНМ, які стануть основою для діагностичної моделі: для порівняння також використаємо РНМ та МЗППВЧ й РНМ та NEAT. Порівняння загальних результатів роботи методів наведено в табл. 4.3.



Таблиця 4.3 – Порівняння експериментально отриманих результатів роботи методів синтезу нейромоделей

Метод синтезу (навчання) моделі	Значення критеріїв порівняння		
	Час роботи, с	Точність на початковій вибірці	Точність на тестовій вибірці
РНМ та МЗППВЧ	572,4	0,0136	0,0214
РНМ та NEAT	1406,9	0,0127	0,02
МГА для РНМ	1223,8	0,0008	0,019

Як видно з таблиці 4.3, кращий результат досягається при використанні методу МГА для РНМ. Невелику різницю можна пояснити послідовним підходом до реалізації методу МГА для РНМ (зазвичай методи нейроеволюції програють у швидкості).

Однак методи нейроеволюції також можуть виконуватися в паралельних системах з використанням окремих вузлів (або ядер) таких систем, а використання механізмів інтелектуального кросовера і селективного тиску значно прискорить роботу. Однак, можливо, в даному випадку точність роботи ще важливіше. Таким чином, модель, синтезована з використанням методів РНМ та МЗППВЧ й РНМ та NEAT, не показали істотних відхилень у навчальній вибірці, тому велика помилка в тестовій вибірці цілком логічна. Модель, синтезована за допомогою методу МГА для РНМ, продемонструвала більшу точність.

Однак слід зазначити, що помилка також може бути виправлена, якщо кількість вимірювань збільшиться. Такі показники прийнятні для даної задачі.

Побудована діаграма (рис. 4.4) наочно демонструє відсутність очевидних помилок, оскільки значення, отримані за допомогою нейромоделі, не накладаються на реальні значення. У поточному варіанті відсутність явних збігів даних є прийнятним для подальшої роботи.

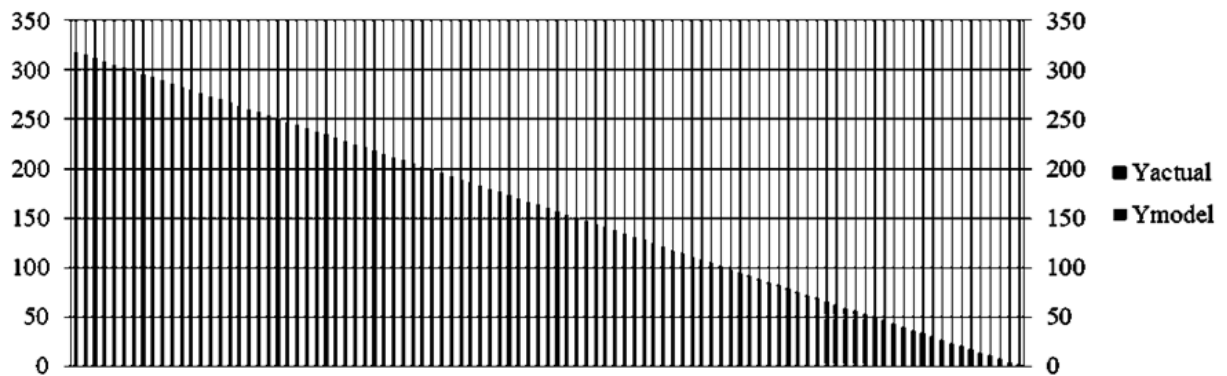


Рисунок 4.4 – Порівняння реальних і отриманих на базі нейромоделі значень вихідного атрибута

Наступною задачею для дослідження ефективності МГА для РНМ була задача діагностування індексу тяжкості пневмонії [178], [179]. Індекс тяжкості пневмонії використовується для оцінки прогнозу пацієнтів з позалікарняною пневмонією, незалежно від виявлення респіраторного вірусу. Вибірка є у вільному доступі [180]. Загальний розмір вибірки: 77490 значень. У табл. 4.4 наведено характеристику вибірки даних, а у таблиці А.2 міститься більш детальний фрагмент вибірки.

Таблиця 4.4 – Загальна характеристика набору даних [180]

Загальна кількість значень	77490	Кількість ознак	54
Характер даних	Числові (після опрацювання)	Кількість екземплярів	1435

В цьому випадку пропонується більш детально порівняти саме нейроеволюційні підходи до синтезу РНМ, а саме РНМ та NEAT й МГА для РНМ.

У таблиці 4.5 наведено порівняння результатів синтезу моделей з використанням двох методів.

Таблиця 4.5 – Результати експериментального тестування методів синтезу нейромоделі

Метод синтезу (навчання) моделі	Значення критеріїв порівняння			
	Розмір популяції	Час роботи, с	Точність на початковій вибірці	Точність на тестовій вибірці
PHM та NEAT	300	14265	0,0262	0,0374
МГА для PHM	300	13949	0,0182	0,0216

З результатів видно, що при однакових розмірах популяції кількість ітерацій та обчислень значно знизилася. Це може свідчити, що використання запропонованого методу кодування на основі секвенування в МГА для PHM вимагає меншої кількості обрахунків при застосуванні однакових генетичних операторів на етапі мутації та кроссоверу.

В PHM та NEAT, зв'язки та ваги закодовані відразу, і схрещування індивідумів здійснюється за унікальною міткою з'єднання. Мережева структура популяції може ефективно розвиватися в ітераційному процесі. Хоча PHM та NEAT – це високоефективний нейроеволюційний метод, він повинен кодувати велику кількість специфічної інформації, щоб забезпечити передачу інформації між поколіннями. Запропонований метод кодування в МГА для PHM, заснований на секвенуванні полімерів і використовує більш компактну стратегію кодування і декодування. Він може ефективно розвивати індивідуальну структуру і вагові коефіцієнти популяції, просто кодуючи невелику кількість інформації.

В запропонованому методі різні особи можуть розвиватися в різні структури, і кожна структура являє собою простір різних вимірювань. Крім того, простий генетичний оператор змушує метод намагатися вирішити проблему багатьма різними способами. Простір модифікацій (мутацій) дуже великий, і це вигідно для пошуку оптимального рішення, що в подальшому відображає його переваги.

Додатково для визначення ефективності методу МГА для РНМ в порівнянні саме з іншими нейроеволюційними методами була розглянута задача з відкритих джерел: вибірка даних Breast Cancer Coimbra Data Set з джерела UC Irvine Machine Learning Repository [181], [182]. Клінічні особливості спостерігалися або вимірювалися у 64 пацієнтів з раком молочної залози та 52 здорових контрольних груп. Існує 10 предикторів, всі кількісні, і бінарна залежна змінна, яка вказує на наявність або відсутність раку молочної залози. Предикторами є антропометричні дані і параметри, які можуть бути зібрані при звичайному аналізі крові. Моделі діагностування, засновані на цих предикторах, якщо вони точні, потенційно можуть бути використані в якості біомаркера раку молочної залози. У таблиці 4.6 наведені основні характеристики вибірки даних. 75% вибірки було використано для навчання, 25% вибірки було використано для тестування.

Таблиця 4.6 – Відомості про вибірку даних [181]

Критерій	Характеристика
Характеристики вибірки даних:	Багатоваріантна
Атрибутивні характеристики:	Integer
Кількість примірників:	116
Кількість атрибутів	10

У таблиці 4.7 наведено загальні результати роботи запропонованого МГА у порівнянні із результатами існуючих аналогів. Особливу увагу приділено визначенню часу, що знадобився для синтезу ШНМ, значенню середньої помилки при навчанні та значенню середньої помилки при роботі у тестовому режимі.

Як видно з результатів у таблиці 4.7, послідовна реалізація МГА для РНМ за часом синтезу РНМ поступається двом існуючим аналогам, а саме РНМ та ESP й РНМ та SANE, проте значно випереджає РНМ та NEAT. Якщо ж порівнювати значення середньої помилки при синтезі РНМ, то при

використанні МГА для РНМ її вдалося мінімізувати до 1,01%, що значно випереджає результати аналогів.

Таблиця 4.7 – Загальні результати експериментально отриманих результатів роботи методів синтезу нейромоделей

Метод	Значення критеріїв порівняння		
	Час синтезу РНМ	Середня помилка на навчальній вибірці	Помилка на тестовій вибірці
РНМ та ESP	653,34	0,0272	0,0288
РНМ та SANE	672,23	0,0302	0,0339
РНМ та NEAT	1197,60	0,0171	0,0209
МГА для РНМ	835,64	0,0101	0,0147

Також слід відзначити, що при тестуванні вже синтезованих РНМ у випадку МГА для РНМ результати значно кращі ніж у аналогів, так середнє значення помилки вихідної РНМ в 2,3 рази менше ніж, наприклад, у РНМ синтезованої методом РНМ та SANE. Тож можна зробити висновок, що запропонований МГА для РНМ значно перевершує існуючі методи за точністю синтезованої нейронної мережі.

Таким чином, результати експериментів показали, що розроблений модифікований метод МГА для синтезу РНМ продемонстрував свою ефективність при використанні на різних вибірках даних для різних задач діагностування. При великих наборах даних МГА для РНМ іноді відпрацьовував недостатньо швидко в порівнянні із методами заснованими на зворотному поширенні помилки проте отримані моделі відрізнялися вищою точністю як на навчальних наборах даних, так і на тестових. В порівнянні ж з іншими нейроеволюційними методами при використанні однакових генетичних операторів розроблений метод відрізнявся вищою швидкістю синтезу фінального рішення, що досягалося за рахунок використання більш компактного способу кодування інформації про особини.

### 4.3 Експериментальне дослідження паралельного методу синтезу штучних рекурентних нейронних мереж

Для визначення ефективності методу ПГМзСТ обиралися задачі із великими наборами вхідних даних. Більш того, важливим було дослідити вплив використання різної кількості обчислювальних вузлів та різних обчислювальних систем для розпаралелювання обчислень [183]. Так була розглянута навчальна вибірка Physical Unclonable Functions Data Set з відкритого репозиторію UCI Machine Learning Repository [184]. Загальні відомості про вибірку наведено у таблиці 4.8.

Таблиця 4.8 – Відомості про вибірку даних [184]

Критерій	Характеристика
Характеристики вибірки даних:	Багатоваріантна
Атрибутивні характеристики:	Integer
Кількість примірників:	6000000
Кількість атрибутів	129

Така задача дозволить більш детально вивчити вплив та зв'язок від використання різної кількості обчислювальних вузлів та зміни розміру батьківського пулу. Детально результати наведено в таблицях А.3 та А.4.

Ефективність використання паралельного підходу визначається співвідношенням часу розв'язання задачі в послідовній системі до часу виконання паралельного методу, помноженого на кількість потоків – у вигляді (4.1).

$$E_p(n) = \frac{T_1(n)}{T_p(n) \cdot p}, \quad (4.1)$$

де  $E_p(n)$  – це ефективність;

$T_1(n)$  – час розв'язання задачі в послідовній системі;

$T_p(n)$  – час виконання паралельного методу;

$p$  – кількість потоків у паралельній системі;

$n$  – величина використовується для параметризації обчислювальної складності розв'язуваної задачі і може розумітися, наприклад, як кількість вхідних даних задачі.

За результати тестування можна зробити висновок, що використання селективного тиску та рівномірного схрещування скорочують розміри популяції, не враховуючі та не опрацьовуючи ті особини з популяції, що відрізняються малим значенням функції пристосованості. Також селективний тиск дозволяє враховувати додаткові показники якості нейромережових моделей. Це дозволяє уникнути проблеми однакових рангів для моделей з одноковою оцінкою функції пристосованості.

З результатів експерименту видно, що найбільш прийнятні показники роботи методу спостерігаються при розмірі батьківського пулу 2 та 7. При інших кількостях батьків співвідношення використаних ресурсів та часу роботи до точності вихідної нейромоделі – не задовільні. Також можна відмітити, що кількість батьківських особин для схрещування більше 10 не має сенсу, адже при великих значеннях часу виконання та накладних витрат на пересилання даних точність значно погіршується.

Збільшення точності та скорочення витрат пам'яті й обчислювальних потужностей для збереження та проведення схрещування повного обсягу популяції засвідчують високу ефективність запропонованого модифікування. Проте слід зазначити зростання вхідних параметрів. Тож наступним етапом може стати автоматизація підбору вхідних параметрів, залежно від задачі та її меж.

Наступною була розглянута знову задача з відкритих джерел: діагностування індексу тяжкості пневмонії [180]. Загальні відомості вже були

наведені у таблиці 4.4. А результати наведено в табл. 4.9-4.11 та у вигляді діаграм на рис. 4.5-4.7.

Таблиця 4.9 – Залежність витраченого часу (с) на синтез діагностичних моделей від кількості використаних ядер обчислювальної системи

Використання CPU					
Кількість ядер	1	2	4	8	16
Витрачений час	13949,00	7556,54	3981,98	2239,84	1304,32
Використання GPU					
Кількість ядер	60	120	240	480	960
Витрачений час	14343,80	8678,00	5597,31	3974,09	3338,24

Таблиця 4.10 – Залежність прискорення (разів) виконання методу від кількості ядер обчислювальної системи

Використання CPU					
Кількість ядер	1	2	4	8	16
Прискорення	1,00	1,85	3,50	6,23	10,69
Використання GPU					
Кількість ядер	60	120	240	480	960
Прискорення	0,97	1,61	2,49	3,51	4,18

Таблиця 4.11 – Залежність долі накладних витрат обчислювальної системи від кількості задіяних ядер системи

Використання CPU					
Кількість ядер	1	2	4	8	16
Накладні витрати обчислювальної системи	0,00	0,08	0,14	0,24	0,41
Використання GPU					
Кількість ядер	60	120	240	480	960
Накладні витрати обчислювальної системи	0,16	0,21	0,29	0,42	0,68



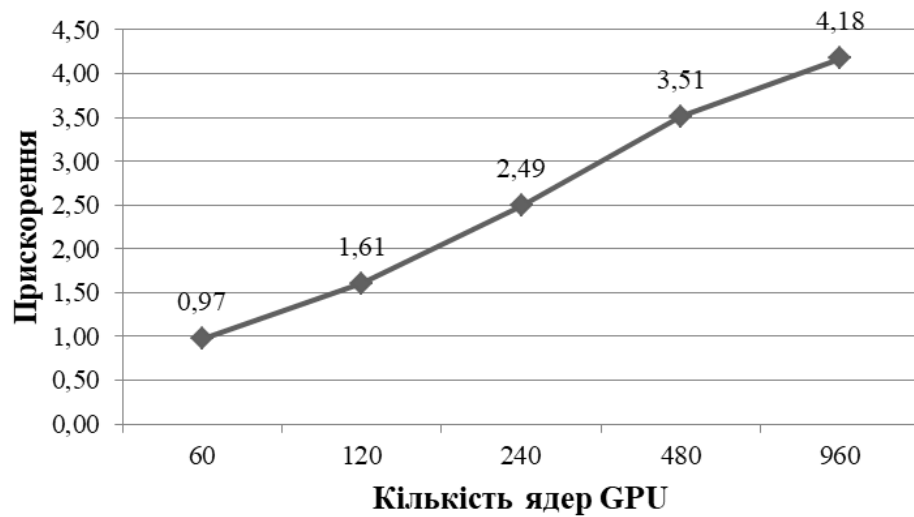


Рисунок 4.6 – Зміна прискорення при тестуванні на GPU

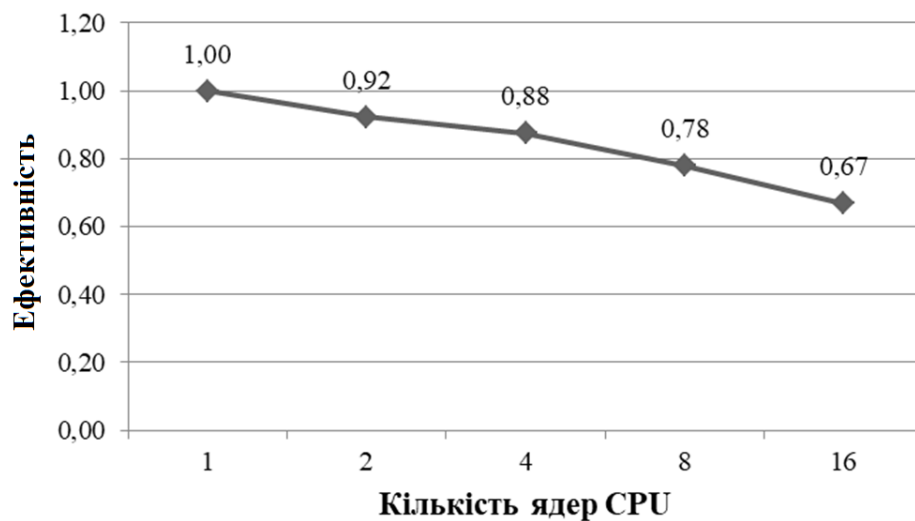


Рисунок 4.7 – Зміна ефективності при тестуванні на CPU

Під час дослідження, особливу увагу було приділено аналізу накладних витрат обчислювальної системи на переселку даних між потоками паралельної обчислювальної системи та синхронізацію обчислень цих потоків [120], [139].

Як видно з табл. 4.9, запропонований метод має прийнятний ступінь паралелізму та ефективно виконується на MIMD- та SIMD-системах [185], [186]. Так при використанні ядер CPU вдалося скоротити час виконання методу з 13949 секунд, що приблизно дорівнює 3,87 годинам (на одному ядрі), до 1304,32 секунд на 16-ти ядрах, що приблизно дорівнює 0,36

годинам. Проте слід зазначити, що при використанні дещо іншої MIMD-системи, такої як кластер, були б наявні значні розбіжності в продуктивності обумовлені архітектурними особливостями. В кластері ядра зв'язані за допомогою комунікатора InfiniBand [187], [188], а в багатоядерному комп'ютері вони розташовані на одному кристалі, що пояснює менший вплив накладних витрат (пересилок та синхронізацій). Крім того, модель процесора в багатоядерному комп'ютері підтримує технологію Turbo Boost, завдяки чому час виконання методу на одному такому ядрі набагато менший, за час виконання на ядрі кластера, що не підтримує таку технологію.

На графічному процесорі при задіяних 960 ядрах час виконання склав 3338,24 секунд, що можна приблизно порівняти з чотирма ядрами комп'ютера.

З таблиці 4.10 та графіків на рисунках 4.4 та 4.5 помітно, що прискорення хоч і не лінійне, але наближається до лінійного. Це пояснюється тим, що доля накладних витрат (таблиця 4.11) виконання запропонованого методу в комп'ютерних системах відносно не значна, а кількість паралельних операцій істотно перевищує кількість послідовних операцій та синхронізацій. Під накладними витратами розуміємо відношення часу, витраченого системою на пересилки та синхронізації між ядрами, до часу цільових обчислень на заданому числі ядер.

Графік ефективності наведено на рис. 4.6. З нього видно, що використання навіть 16-ти ядер комп'ютерних систем при виконанні запропонованого методу зберігає ефективність на відносно прийнятному рівні та свідчить про потенціал при необхідності та можливості задіяти ще більше ядер.

Таким чином, запропонований метод добре розпаралелюється на сучасних комп'ютерних архітектурах, що дозволяє істотно скоротити час виконання задачі синтезу РНМ. Паралельний підхід значно підвищує ефективність послідовного МГА для РНМ та робить його ще більш

прийнятним для синтезу ШНМ, через значне зниження часових витрат та збереження високої точності отриманих нейронних мереж.

Слід також дослідити комбінування використання кількості обчислювальних вузлів саме CPU з відповідною кількістю місць у батьківському пулі для рівномірного схрещування. Для цього розглянемо задачу на основі набору даних HIGGS Data Set [189] з відкритого джерела UCI Machine Learning Repository – інформація про вибірку даних наведена у таблиці 4.12. Результати тестування наведені у таблицях 4.13.

Таблиця 4.12 – Загальна характеристика вибірки даних HIGGS Data Set [189]

Критерій	Характеристика
Характеристики вибірки даних:	Багатоваріантна
Атрибутивні характеристики:	Real
Кількість примірників:	11000000
Кількість атрибутів	28

Таблиця 4.13 – Результати виконання методу на системі під управлінням CPU

Розмір батьківського пулу	2	4	8	14
Кількість ядер	2	4	8	14
Помилка	0,036	0,064	0,079	0,220
Витрачений час, с	46660,08	28009,37	18240,33	48392,25
Прискорення	1,25	2,02	5,07	1,07
Накладні витрати	0,08	0,16	0,30	0,65
Ефективність	0,63	0,51	0,63	0,08

Як видно з наведеної таблиці 4.13 під час експериментів найбільш оптимальних показників часу вдалося набути під час використання 8 ядер та відповідному розмірі батьківського пулу у системі під управлінням CPU. Проте слід зазначити що розмір помилки дещо збільшився. При використанні

більш ніж 8 ядер та батьківських особин для рівномірного схрещування час виконання значно зріз, а отримані результати погіршилися.

Як видно з наведеної таблиці 4.13 під час експериментів час виконання значно зріз у порівнянні із системою під управлінням CPU. Така поведінка може бути пояснена достатньо складними операціями з якими ядра GPU пораються значно повільніше від ядер CPU. Найбільш оптимальних показників, коли витрачений час корелювався із розміром помилки вдалося досягнути у випадку використання 8 батьківських особин для рівномірного схрещування та задіяні 240 потоків GPU. При використанні 960 потоків час виконання зростає, через складнощі в обчисленнях, а отримані результати погіршуються.

За результатами тестування можна зробити висновок, що використання селективного тиску та рівномірного схрещування скорочують розміри популяції, не враховуючі та не опрацьовуючі ті особини з популяції, що відрізняються малим значенням функції пристосованості. Також селективний тиск дозволяє враховувати додаткові показники якості нейромережових моделей. Це дозволяє уникнути проблеми однакових рангів для моделей з одноковою оцінкою функції пристосованості.

З результатів експерименту видно, що найбільш прийнятні показники роботи методу спостерігаються при розмірі батьківського пулу від 2 до 8. При інших кількостях батьків співвідношення використаних ресурсів та часу роботи до точності вихідної нейромоделі – не задовільні. Також можна відмітити, що кількість батьківських особин для схрещування менша за 10 не має сенсу, адже при великих значеннях часу виконання та накладних витрат на пересилання даних точність значно погіршується.

Збільшення точності та скорочення витрат пам'яті й обчислювальних потужностей для збереження та проведення схрещування повного обсягу популяції засвідчують високу ефективність запропонованого модифікування.

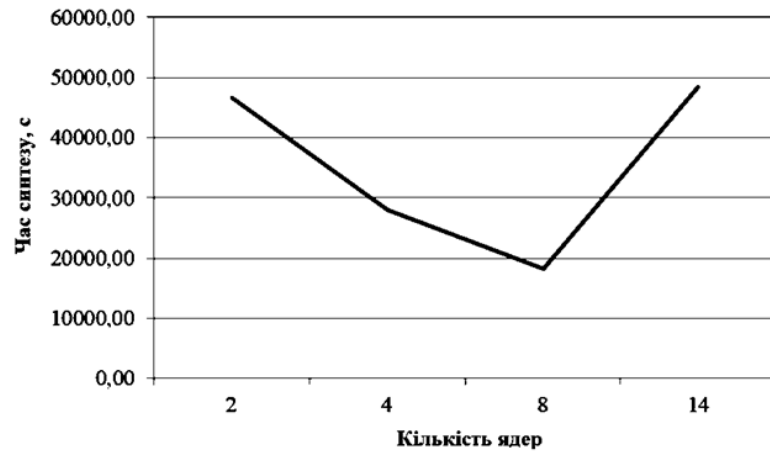


Рисунок 4.8 – Розподіл витраченого часу в залежності від кількості використаних ядер системи на CPU

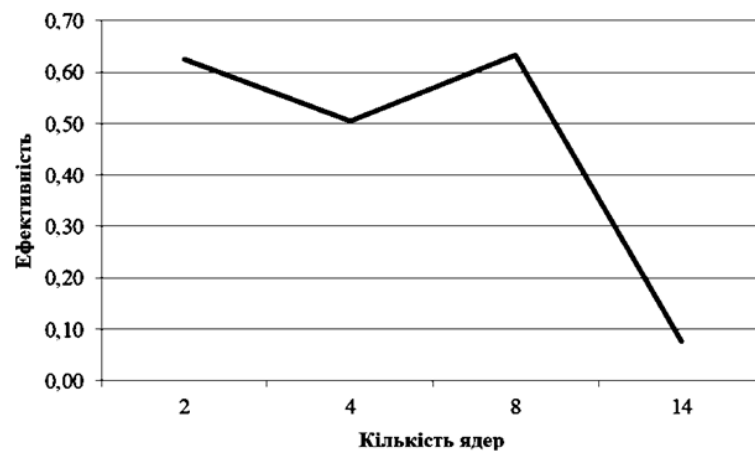


Рисунок 4.9 – Зміна ефективності в залежності від кількості використаних ядер системи на CPU

Проте слід зазначити зростання вхідних параметрів. Тож наступним етапом може стати автоматизація підбору вхідних параметрів, залежно від задачі та її меж.

#### 4.4 Експериментальне дослідження індикаторної системи при визначенні механізмів доналаштування штучних нейронних мереж

Для дослідження розробленої індикаторної системи використаємо задачу моделювання динаміки піттінгових процесів сталених сплавів [151].

Піттингова (точкова) корозія – це вид корозійного руйнування, якому піддаються виключно пасивні метали і сплави. Піттингова корозія спостерігається у нікелевих, цирконієвих, хромонікелевих, хромистих, алюмінієвих сплавах та ін. При піттингової корозії руйнуванню піддаються тільки окремі ділянки поверхні, на яких утворюються глибокі ураження – піттинги (точкові виразки). Сталь AISI304 застосовується у багатьох областях, наприклад, у виготовленні збірних і зварних металоконструкцій, складових частин трубопровідної арматури, а також побутового обладнання: огорожі балконів і сходів, кухонна апаратура, автомобільні вихлопні системи. Такі конструкційні матеріали мають високу корозійну стійкість у багатьох агресивних середовищах, але можуть піддаватися піттинговій корозії в розчинах, що містять хлорид ( $C_{Cl}$ ). Саме тому є актуальним завданням побудова моделей залежності піттингоутворення від характеристик використовуваної сталі і оборотних розчинів [151]. Перспективним базисом для таких моделей є ШНМ, завдяки їхнім здатностям до навчання за експериментальними даними, узагальнення даних і витягу знань з даних. ШНМ можуть використовуватися для побудови моделей залежностей за спостереженнями при умовах відсутності або невідомості залежностей у даних. У табл. 4.14 наведено фрагмент вхідних даних. [151]

Таблиця 4.14 – Фрагмент вибірки вхідних даних [151]

$x_1$	$x_2$	$x_3$	...	$x_{18}$	$y_1$	$y_2$	$y_3$	...	$y_7$
4	300	0,0199	...	2,27	0,01421	0,18776	0,01620	...	0
4	300	0,0216	...	2,27	0,01109	0,63335	0,01337	...	0
5	300	0,0199	...	2,27	0,00275	0,00296	0,01479	...	1
5	300	0,0216	...	2,27	0,00379	0,00275	0,01400	...	1
...	...	...	...	...	...	...	...	...	...
8	600	0,0216	...	2,27	0,00412	0,02114	0,03708	...	1

Вибірка даних складалася із 50 екземплярів — точкових спостережень, що характеризувалися значеннями вхідних ознак:  $x_1$  – рН (характеристика середовища, безрозмірна);  $x_2$  –  $C_{Cl}$ , мг/л (концентрація

хлоридів в середовищі);  $x_3 - V_{ок}$ , об. % (об'єм оксидів в сталі);  $x_4$  – кількість оксидів розміром до 1,98 мкм на 100 перешкод зору оптичного мікроскопа ( $\times 320$ ) в сталі, шт.;  $x_5$  – кількість оксидів розміром від 1,98 мкм до 3,95 на 100 перешкод зору оптичного мікроскопа ( $\times 320$ ) в сталі, шт.;  $x_6 - L_{ок}$ , мкм (середня відстань між оксидами в сталі);  $x_7 - d_3$ , мкм (середній діаметр зерна аустеніту в сталі);  $x_8 - P_{\alpha}$ , об. % (об'єм  $\delta$ -фериту в сталі);  $x_9 - C$ , мас. % – масові відсотки (вміст вуглеців у сталі);  $x_{10} - Mn$ , мас. % (вміст марганцю в сталі);  $x_{11} - Si$ , мас. % (вміст кремнію в сталі);  $x_{12} - Cr$ , мас. % (вміст хрому в сталі);  $x_{13} - Ni$ , мас. % (вміст нікелю в сталі);  $x_{14} - N$ , мас. % (вміст азоту в сталі);  $x_{15} - Ti$ , мас. % (вміст титану в сталі);  $x_{16} - S$ , мас. % (вміст сірки в сталі);  $x_{17} - P$ , мас. % (вміст фосфору в сталі);  $x_{18} - X_o$  м<sup>3</sup>/кг (питома магнітна сприйнятливність аустеніту) [128].

Залежні змінні:  $y_1 - \Delta Cr$ , мг ( $10^{-5}$ ) – втрати хрому зі сталі після її витримки в розчині;  $y_2 - \Delta Fe$ , мг ( $10^{-5}$ ) – втрати заліза зі сталі після її витримки в розчині;  $y_3 - \Delta Ni$ , мг ( $10^{-5}$ ) – втрати нікелю зі сталі після її витримки в розчині;  $y_4 = ZCr$ ;  $y_5$  – якщо  $ZCr < 1$ , то сталь в розчині з такими параметрами (ознаками  $x_1, x_2$ ) піддається піттинговій корозії з утворенням стабільних піттингів, в іншому випадку – сталь в розчині з такими параметрами (ознаками  $x_1, x_2$ ) піддається піттинговій корозії з утворенням метастабільних піттингів;  $y_6 = ZNi$ ;  $y_7$  – якщо  $ZNi < 1$ , то піттинги інтенсивно ростуть, в іншому випадку – піттинги збільшуються в розмірі не інтенсивно [151].

Проведемо аналіз вхідних даних та вибірки. Так  $Lev_{smp\lfactn} = 0$ , адже кількість вхідних ознак не велика та можливо провести їх скорочення.

В нашому випадку  $Lev_{fctr} = 0$ , через те, що під час проведення факторного аналізу зустрічалися кореляційні матриці, що були погано

обумовлені (їх детермінант дорівнював 0). Факторні навантаження наведено у табл. 4.15.

Таблиця 4.15 – Факторні навантаження

$x$	Фактор №1	Фактор №2	Фактор №3
$x_1$	-0,000000	-0,000000	-0,894181
$x_2$	0,000000	0,000000	-0,447706
$x_3$	-0,989883	0,031646	-0,000000
$x_4$	0,677879	-0,644594	0,000000
$x_5$	-0,980149	0,158833	0,000000
$x_6$	0,817799	0,493587	-0,000000
$x_7$	0,837964	-0,210144	-0,000000
$x_9$	0,533363	0,743210	0,000000

За умовами задачі вимірювання відбувалися із використанням спеціальних сенсорів, тому  $Lev_{accmeas} = 0$ . А  $Lev_{manag} = 0$  тому що експериментальна система весь час знаходилась під наглядом та контролем. Відповідно задача відноситься до складності ОП, що свідчить про подальшу побудову ШНМ із визначеною кількістю нейронів в прихованому шарі.

У табл. 4.16 наведено коефіцієнти парної кореляції вхідних ознак між собою. Слід зазначити, що  $x_{15}$  – взагалі не корелюється.

Із результатів таблиці 3 можна зробити висновок, що ознаки  $x_1$ ,  $x_2$  та  $x_{15}$  не залежать від інших ознак, а усі інші ознаки сильно зв'язані.

У табл. 4.17 наведено коефіцієнти парної кореляції між вхідними та вихідними ознаками. Слід зазначити, що  $x_{15}$  – взагалі не корелюється.

Із результатів таблиці 4.17 можна зробити висновок, що найбільш індивідуально значущим для вихідних ознак є ознаки:  $x_1$ ,  $x_2$  та  $x_9$ .

Тож отримуємо коефіцієнти  $K_{corrY} = 3$ ,  $K_{ntcorrX} = 3$ ,  $K_{imp} = 8$ . У результаті найбільш оптимальна кількість прихованих нейронів  $|N_h| = 4$ .



Проведемо тестування та порівняємо значення середньої помилки на повних даних та на даних після відбору ознак та з оптимальною кількістю прихованих нейронів у прихованому шарі. Результати порівняння у табл. 4.18. Для відбору ознак використовувався метод поступового видалення та додавання ознак.

Таблиця 4.16 – Коефіцієнти парної кореляції вхідних ознак

$\Gamma$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{13}$	$x_{14}$	$x_{15}$	$x_{16}$	$x_{17}$	$x_{18}$
$x_1$	1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
$x_2$	0,00	1,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
$x_3$	0,00	0,00	1,00	-0,68	0,99	-0,83	-0,80	0,94	-0,46	0,15	-0,13	-0,49	0,00	-0,16	0,25	0,89
$x_4$	0,00	0,00	-0,68	1,00	-0,78	0,18	0,59	-0,79	0,00	-0,39	0,58	-0,09	0,00	-0,40	0,31	-0,62
$x_5$	0,00	0,00	0,99	-0,78	1,00	-0,75	-0,80	0,96	-0,39	0,20	-0,23	-0,40	0,00	-0,05	0,15	0,89
$x_6$	0,00	0,00	-0,83	0,18	-0,75	1,00	0,54	-0,71	0,69	-0,02	-0,16	0,82	0,00	0,61	-0,65	-0,73
$x_7$	0,00	0,00	-0,80	0,59	-0,80	0,54	1,00	-0,61	0,28	-0,08	0,13	0,13	0,00	-0,34	-0,03	-0,60
$x_8$	0,00	0,00	0,94	-0,79	0,96	-0,71	-0,61	1,00	-0,30	0,16	-0,20	-0,40	0,00	-0,19	0,10	0,93
$x_9$	0,00	0,00	-0,46	0,00	-0,39	0,69	0,28	-0,30	1,00	-0,61	0,35	0,92	0,00	0,48	-0,94	-0,11
$x_{10}$	0,00	0,00	0,15	-0,39	0,20	-0,02	-0,08	0,16	-0,61	1,00	-0,95	-0,38	0,00	0,12	0,42	-0,20
$x_{11}$	0,00	0,00	-0,06	-0,49	0,04	0,32	0,18	0,11	-0,18	0,87	-0,93	-0,01	0,00	0,26	0,01	-0,23
$x_{13}$	0,00	0,00	-0,13	0,58	-0,23	-0,16	0,13	-0,20	0,35	-0,95	1,00	0,09	0,00	-0,38	-0,11	0,16
$x_{14}$	0,00	0,00	-0,49	-0,09	-0,40	0,82	0,13	-0,40	0,92	-0,38	0,09	1,00	0,00	0,77	-0,94	-0,28
$x_{15}$	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1,00	0,00	0,00	0,00
$x_{16}$	0,00	0,00	-0,16	-0,40	-0,05	0,61	-0,34	-0,19	0,48	0,12	-0,38	0,77	0,00	1,00	-0,68	-0,24
$x_{17}$	0,00	0,00	0,25	0,31	0,15	-0,65	-0,03	0,10	-0,94	0,42	-0,11	-0,94	0,00	-0,68	1,00	-0,02
$x_{18}$	0,00	0,00	0,89	-0,62	0,89	-0,73	-0,60	0,93	-0,11	-0,20	0,16	-0,28	0,00	-0,24	-0,02	1,00

За результатами початкового аналізу задачу було розподілено до групи ОП. Адже вхідна вибірка була не надмірною, ризику людського впливу виключалися. Єдиний суттєвий ускладнюючий фактор – це кореляційні матриці, що були погано обумовлені (їх детермінант дорівнював 0), що було підтверджено факторним аналізом.

Тож, на початку було протестовано навчання нейромоделі з використанням усіх вхідних даних, а оскільки за результатами факторного аналізу було виявлено 8 найзначущих змінних: для Фактору 1 –  $x_3 - x_7$ ; для Фактору 2 –  $x_4, x_9$ ; для Фактору 3 –  $x_1$ , то на початку  $|N_h| = 8$ .

Таблиця 4.17 – Коефіцієнти парної кореляції вхідних та вихідних

ознак

$r$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$y_6$	$y_7$
$x_1$	-0,0240	-0,3045	0,0292	0,1242	0,1260	0,1317	0,1132
$x_2$	-0,3102	-0,2726	0,1651	-0,2519	-0,3563	-0,2464	-0,1601
$x_3$	0,1198	0,0172	0,2271	0,2153	0,0417	-0,0259	0,1020
$x_4$	0,0757	0,0193	-0,0486	-0,1540	0,0086	0,1167	-0,0194
$x_5$	0,0862	0,0104	0,2032	0,2145	0,0337	-0,0461	0,0909
$x_6$	-0,2262	-0,0467	-0,2617	-0,1657	-0,0708	-0,0532	-0,1225
$x_7$	-0,1016	-0,0253	-0,1831	-0,2068	0,0060	-0,0075	-0,0460
$x_8$	0,0404	-0,0113	0,1788	0,1881	0,0445	-0,0828	0,1013
$x_9$	-0,3200	-0,1464	-0,1721	-0,0998	-0,0484	-0,1651	-0,0282
$x_{10}$	0,1397	0,1280	-0,0134	0,0441	-0,0052	0,0852	-0,0676
$x_{11}$	-0,0254	0,0618	-0,1145	-0,0126	-0,0192	-0,0108	-0,0820
$x_{12}$	-0,0717	0,0230	-0,0507	0,0833	-0,0306	-0,0713	-0,0511
$x_{13}$	-0,0442	-0,0918	0,0562	-0,0462	0,0249	-0,0267	0,0806
$x_{14}$	-0,2920	-0,1089	-0,1929	-0,0861	-0,0731	-0,1287	-0,0718
$x_{15}$	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
$x_{16}$	-0,1545	-0,0229	-0,1222	0,0133	-0,0840	-0,0535	-0,0961
$x_{17}$	0,3191	0,1352	0,1549	0,0456	0,0576	0,1793	0,0345
$x_{18}$	0,0025	-0,0547	0,1910	0,1750	0,0472	-0,1046	0,1274

Таблиця 4.18 – Порівняння результатів навчання нейромоделей

Цільова змінна	Кількість прихованих нейронів	Час навчання, с	Середня помилка	Використані незалежні змінні	Кількість прихованих нейронів	Час навчання, с	Середня помилка
$y_1$	8	4,2400	0,00336	$x_2, x_9$	4	3,2190	0,00165
$y_2$	8	5,6440	0,00686	$x_1, x_2, x_9, x_{18}$	4	4,3480	0,00639
$y_3$	8	4,3280	0,00641	$x_1, x_2, x_8, x_9, x_{18}$	4	3,3090	0,00181
$y_4$	8	5,5470	0,00344	$x_1, x_2, x_9, x_{18}$	4	4,9320	0,16357
$y_5$	8	4,4030	0,00316	$x_1, x_2, x_9, x_{18}$	4	3,3320	0,00191
$y_6$	8	7,8150	0,72003	$x_1, x_2, x_9, x_{18}$	4	4,9700	0,84592
$y_7$	8	4,4480	0,00501	$x_1, x_2, x_9, x_{18}$	4	4,3400	0,00357

Аналізуючи початкові результати, слід зазначити достатньо великий розбіг серед часу навчання моделі: від 4,24 до 7,82. При цьому розмір

середньої помилки був досить прийнятним та коливався від 0,00316 до 0,72003.

Результати, отримані на даних після редукції, засвідчили, що точність у деяких випадках (для  $y_1$ ,  $y_2$ ,  $y_3$ ,  $y_5$  та  $y_7$ ) покращилася, а час значно знизився, більш того розподіл часу між ітераціями навчання розподілився більш рівномірно. Проте для низки випадків ( $y_4$ ,  $y_6$ ) точність моделі знизилася. Це може пояснюватися незначним впливом усунених ознак, що може мати групову природу. Тож можна зробити висновок, що для відбору інформативних ознак в даному випадку слід використовувати методи, що здатні відібрати не тільки окремі ознаки, а й групи ознак.

Іншою задачею є діагностування оброблюваності різанням високохромистих чавунів [190]. Високохромисті чавуни із вмістом хрому 10-35 % використовують для виготовлення деталей механізмів, що працюють в умовах інтенсивного абразивного зношування. Ці деталі повинні мати значну твердість. Удосконалення конструкції деталей різних механізмів (грунтових, шламових та піскових насосів, молоткових дробарок, кульових млинів, флотомашин) потребує їх механічної обробки різанням. Це призводить [190] до виникнення суперечливих вимог до матеріалів, а саме: для забезпечення зносостійкості потрібна максимальна твердість, а для обробки різанням мінімальна.

Вирішення цієї проблеми полягає у тому, що матеріал повинен мати задовільну оброблюваність різанням у стані литва (низьку твердість), а після механічної обробки різанням потрібно виконати термічну обробку, що забезпечить достатній рівень твердості і відповідно зносостійкості.

Були виготовленні зразки високохромистих чавунів, що містили: 1,09...3,54% С, 11,26...29,68% Cr, 0,77...5,71% Mn, 0,13...2,86% Ni, 0,99...1,46% Si. Для цих зразків визначали знос різцю ( $I_{\text{різ}}$ ) при механічній обробці різанням. Мінімальний знос різцю 0,45 мкм/м спостерігали при різанні зразку 1 (див. табл.) із мінімальним вмістом вуглецю 1,09% і кількості карбідів (К) – 9,2%. Твердість чавунів (HRC) та мікротвердість складових

структури основи ( $HV_{50}^{осн}$ ) і карбідів ( $HV_{50}^{кар}$ ) визначали у стані литва та після термічної обробки відпалу при  $690^{\circ}\text{C}$  та  $730^{\circ}\text{C}$  протягом 9 годин та нормалізації від  $1050^{\circ}\text{C}$  із витримкою 4,5 години [150].

Необхідно встановити функції: Знос різцю ( $I_{різ}$ ), Кількість карбідів ( $K$ ), Твердість ( $HRC$ ), Мікротвердість основи ( $HV_{50}^{осн}$ ) і карбідів ( $HV_{50}^{кар}$ ) в стані литва та після термічної обробки (відпалу, нормалізації) від аргументів: вмісту в чавуні  $C, Cr, Mn, Ni, Si$ . Фрагмент вибірки даних наведено у таблиці 4.19 [190].

Позначемо:

- $x_1$  – вміст в сплаві  $C$ , %;
- $x_2$  – вміст в сплаві  $Cr$ , %;
- $x_3$  – вміст в сплаві  $Mn$ , %;
- $x_4$  – вміст в сплаві  $Ni$ , %;
- $x_5$  – вміст в сплаві  $Si$ , %;
- $y_1$  – знос різцю ( $I_{різ}$ );
- $y_2$  – кількість карбідів ( $K$ );
- $y_3$  – твердість ( $HRC$ );
- $y_4$  – мікротвердість основи ( $HV_{50}^{осн}$ );
- $y_5$  – мікротвердість карбідів ( $HV_{50}^{кар}$ ).

Таблиця 4.19 – Фрагмент вибірки даних [190]

$x_1$	$x_2$	$x_3$	...	$x_{18}$	$y_1$	$y_2$	$y_3$	...	$y_7$
1,09	17,33	1,99	...	1,37	0,45	9,2	34,1	...	433
1,52	18,73	1,49	...	1,33	0,91	13,4	35,7	...	415
2,55	11,91	3,8	...	1,26	9,12	18,7	38,9	...	330
2,24	22,82	1,21	...	1,29	2,96	19,4	49,1	...	529
...	...	...	...	...	...	...	...	...	...
3,01	29,68	0,72	...	1,09	39,63	32,1	57,7	...	1402

Будемо використовувати оцінку складності задачі, отримавши яку вирішимо як надалі будемо проектувати топологію  $NN_{struct}$  ШНМ. Якщо задачу за рівнем складності можна віднести до класу «організована простота», то використовуючи відбір ознак отримаємо найбільш

інформативні та використовуючи дослідження вхідних даних зазначимо необхідну кількість нейронів в прихованому шарі мережі  $N_h = \{N_{h_1}, N_{h_2}, \dots, N_{h_r}\}, r = 1, 2, \dots, |N_h|$ . Після цього можна буде виконати навчання нейромоделі.

Отже можемо зробити висновок, що має рівень складності категорії ОП, що робить можливим подальше використання індикаторів ля визначення структурних особливостей моделі.

У нашому випадку  $Num_{element} = 3$  за замовченням, адже припускається, що використовуються виключно вхідні, приховані та вихідні нейрони. Далі необхідно визначити кількість незалежних змінних, що сильно корелюються із вихідними ознаками  $K_{corrY}$ . Додатково визначається кількість незалежних змінних, що слабо залежать від інших або не корелюються між собою  $K_{ntcorrX}$ . Додатково з попереднього етапу підтягується інформація про кількість найбільш значущих незалежних змінних серед факторів  $K_{imp}$ . Після цього визначення кількості нейронів прихованого шару можна визначити за формулою:

$$|N_h| = K_{input} - K_{corrY} - K_{imp} - K_{ntcorrX} = 3.$$

Отже в подальшому ШНМ будемо синтезувати із 3-ма нейронами в прихованому шарі.

У таблиці 4.20 наведено результати для моделей на основі ШНМ.

Таблиця 4.20 – Результати для моделей на основі ШНМ

Цільова змінна	Кількість прихованих нейронів	Час навчання, с	Середня помилка
$y_1$	3	3	0,061046
$y_2$	3	2	0,093592
$y_3$	3	4	0,040249
$y_4$	3	2	0,011954
$y_5$	3	2	0,096513

У таблиці 4.21 наведено результати моделей на тестових даних.

Таблиці 4.21 – Результати моделей на тестових даних

Цільова змінна	Лінійна модель першого порядку	Лінійна модель другого порядку	Лінійна модель другого порядку з часткою першого порядку	Лінійна регресійна поліноміальна модель зі скороченим числом ознак	ШНМ
$y_1$	5,1	1,8	0,2	0,3	0,1
$y_2$	1,7	1,5	0,4	0,3	0,12
$y_3$	3,0	2,6	0,6	0,5	0,08
$y_4$	4,2	3,8	0,8	0,6	0,05
$y_5$	1,42	3,58	0,3	0,4	0,13

За результатами початкового аналізу завдання було покладено на групу ОП. Зрештою, вхідна вибірка не була надмірною, і ризики людського впливу були виключені. Єдиним істотним ускладнюючим фактором є погано обумовлені кореляційні матриці.

Подальші розрахунки показали, що використання 3 нейронів у прихованому шарі достатньо для побудови нейронної моделі з прийнятною точністю роботи.

Аналізуючи початкові результати, слід відзначити досить великий розбіг за часом навчання моделі: з 4 секунд (найбільший показник серед моделей, заснованих на ШНМ) до 34,37 для моделей лінійної регресії.

Результати, отримані на даних після редукції, показали, що точність зросла при побудові нейронної моделі з певною структурою на основі системи індикаторів, а час значно скоротився.

Крім того, слід відзначити високий рівень логічної прозорості отриманих моделей на основі ШНМ.

#### **4.5 Експериментальне дослідження методу структурної оптимізації попередньо синтезованих штучних нейронних мереж**

Для експериментального дослідження результатів використання запропонованого методу оптимізації структури НМ проведемо ряд тестувань

та на початку визначимо чи варто взагалі виконувати структурну оптимізацію нейромоделі.

Наступною задачею було обрано задачу на основі набору даних Crop mapping using fused optical-radar data set з відкритого джерела UCI Machine Learning Repository [191]. Цей великий набір даних являє собою злиті двухчасові оптико-радіолокаційні дані для класифікації орних земель. Зображення були зібрані супутниками RapidEye (оптичними) і системою безпілотного літального апарату Synthetic Aperture Radar (UAVSAR) над сільськогосподарським районом поблизу Вінніпега, Манітоба, Канада, в 2012 році. Загалом вибірка містить 98 радарних ознак і 76 оптичних ознак, що фіксувалися протягом двох днів 5 і 14 липня 2012 року. Для цього набору даних існує сім класів типів культур: кукурудза; горох; ріпак; соя; овес; пшениця і широколисті. Такі дані максимально наближені до реальних даних, що можуть бути отримані наприклад з використанням дронів. Характеристики набору даних у таблиці 4.22.

Таблиця 4.22 – Характеристика набору даних Crop mapping using fused optical-radar data set

Загальна кількість значень	57020950	Кількість ознак	175
Характер даних	Числові	Кількість екземплярів	325834

Визначаючи рівень складності задачі моделювання  $Lev_{task}$  були отримані наступні значення:  $Lev_{smplfctn} = 1$ ;  $Lev_{fctr} = -1$ ;  $Lev_{accmeas} = 0$ ;  $Lev_{manag} = 0$ . Отже можемо зробити висновок, що має рівень складності категорії ОС, що робить можливим подальше використання структурної оптимізації попередньо синтезованої ШНМ.

На рис. 4.10 наведено приклад РНМ. Як видно з рисунку структура РНМ є надмірною, зазвичай в таких мережах є нульові ваги у деяких синопсисів.

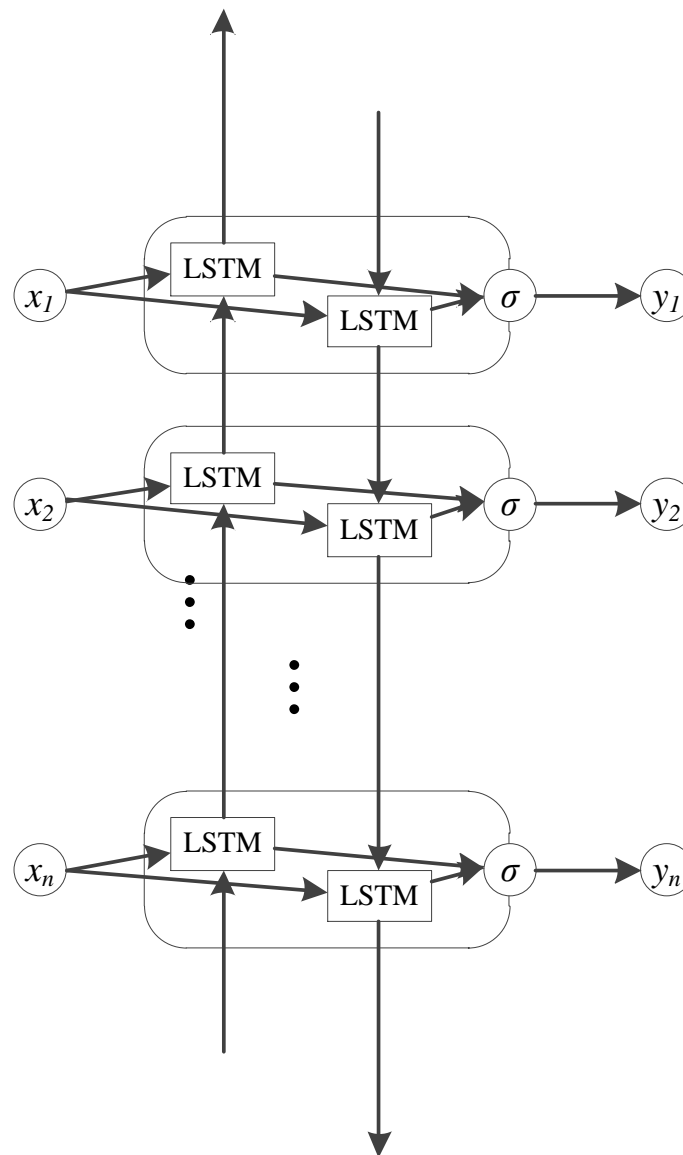


Рисунок 4.10 – НМ до оптимізації

Під час тестування будемо звертати увагу на час роботи мережі та розподіл часу за ітераціями. Додатково будемо оцінювати точність роботи. Для цього введемо відносну помилку, яка в цьому випадку буде обчислюватися, як відношення кількості невірно діагностованих об'єктів до загального розміру вибірки (кількості її екземплярів).

Точність роботи не є цільовим параметром, проте важливо, щоб оптимізація не погіршувала цього значення.

У табл. 4.23 наведено порівняння результатів роботи для звичайної мережі та мережі після оптимізації.



Таблиця 4.23 – Порівняння результатів роботи

Критерії	Час роботи, с	Помилка, %
Звичайна мережа	7545	0,052
Оптимізована мережа	4860	0,0359

З таблиці 4.23 видно, що час роботи значно скоротився. Це пояснюється тим, що оптимізована мережа не задіює зайві синопси та нейрони, що значно пришвидшує роботу із нею. Також важливим є те, що точність роботи оптимізованої мережі зросла. Такий ефект можна пояснити тим, що надмірна структура мережі має надлишкове число вільних параметрів, які в процесі навчання налаштовуються не тільки на відновлення цільової залежності, але і на відтворення шуму, що може впливати на точність роботи.

На рис. 4.11 наведено графік, що демонструє різницю у розподілі часу між ітераціями під час роботи РНМ.

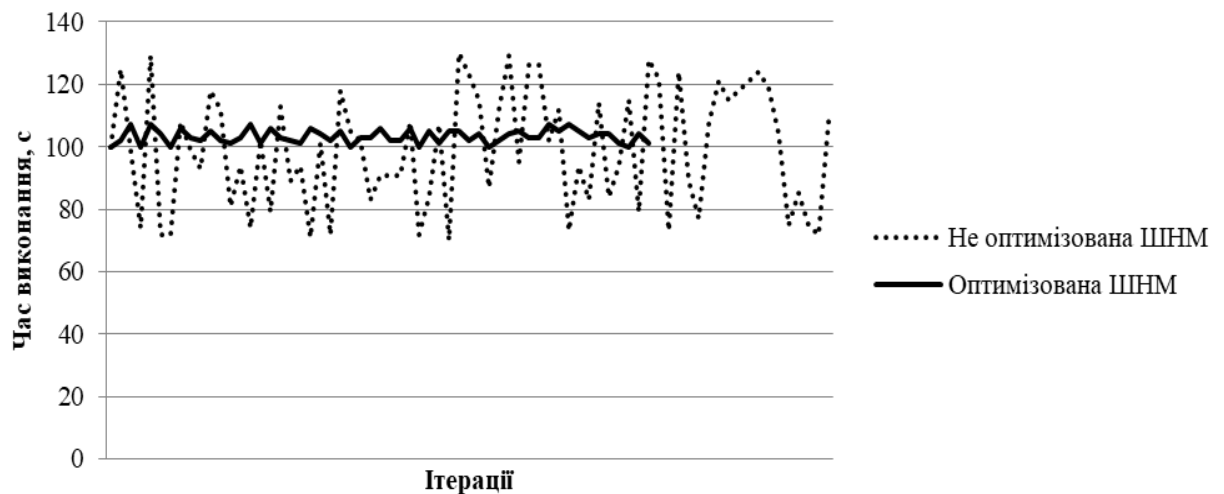


Рисунок 4.11 – Розподіл часу між ітераціями під час роботи РНМ

Під час аналізу графіка можна зробити висновок, що як раз надмірна топологічна складність мережі до оптимізації призводить до різких стрибків у часі між ітераціями. У той же час графік розподілу для оптимізованої мережі більш плавний. І хоча певні стрибки все ж є, їх вплив значно менше

та частково його можна списати на особливості складної рекурентної архітектури мережі.

Також варто зазначити, що спрощення структури РНМ впливає і на ресурсоемність, а саме на використання пам'яті. Зазвичай, найбільше пам'яті вимагає процес серіалізації ваг синопсисів. Оскільки оптимізована структура має меншу кількість нейронів та зв'язків між ними – це скорочує витрати пам'яті на збереження мережі.

#### **4.6 Висновки за розділом 4**

У розділі виконано експериментальне дослідження розроблених методів аналізу даних та синтезу діагностичних моделей на основі РНМ.

Результати проведених експериментів показали, що розроблений метод МГА для РНМ відпрацьовує в 1,08-1,5 разів повільніше від методів на основі зворотного поширення, проте точність отриманих моделей відрізнялася на 10-40%. Це засвідчує, що в складних задачах, де точність важливіше, запропонований метод відпрацьовує найкраще. Синтезовані нейромоделі можуть бути використані для діагностування, оцінки та управління складними технічними системами, що значно підвищує продуктивність виробництв, дозволяючи уникнути додаткових витрат. При порівнянні із іншими методами нейроеволюційного підходу можна відмітити підвищену ефективність методу – метод працює краще завдяки більш компактній схемі кодування інформації про ШНМ, що спрощує використання генетичних операторів у майбутньому та пришвидшує виконання синтезу.

Визначено, що метод ПГМзСТ для синтезу РНМ, в якості основи для діагностичних моделей, відрізняється підвищенням точності і зниженням витрат пам'яті та обчислювальної потужності на зберігання і перетин загального обсягу сукупності, що підтверджує високу ефективність. Використання механізмів селективного тиску в комбінації із рівномірним схрещуванням дійсно підвищують адаптивні характеристики методу. Це

пов'язано з тим, що в процесі синтезу система самостійно вибирає і зменшує кількість осіб, що беруть участь в схрещуванні, що дозволяє акумулювати додаткові характеристики РНМ і значно знизити споживання пам'яті, а також знизити накладні витрати на передачу даних. З іншого боку, важливо відзначити, що цей метод складно виконати на ядрах GPU, оскільки легкі потоки призначені для однотипних і простих дій, і такі операції можуть виконуватися на них набагато повільніше.

Розроблена індикаторна система, використовуючи оцінку складності задачі і систему показників, дозволяє визначити найбільш оптимальний і логічний спосіб доопрацювання ШНМ [151]. Було засвідчено, що система дозволяє використовувати чіткі механізми для підвищення рівня логічної прозорості моделі на основі комплексної оцінки проблеми. При тестуванні методу час навчання нейромоделей було прискорено в середньому на 22%. Тому результати проведених експериментів свідчать про реальне спрощення структури і більш раціональне використання обчислювальних ресурсів. А отримані на основі оцінки складності задачі і системи показників нейромоделі відрізняються високим рівнем логічної прозорості з високим рівнем точності роботи. Такі нейромережеві моделі можуть бути використані при реалізації реальних технічних процесів на виробничих об'єктах. Їх використання дозволить значно знизити виробничі витрати і автоматизувати процес моделювання.

Запропонований метод структурної оптимізації попередньо синтезованих ШНМ на основі нейроеволюційного підходу з використанням системи індикаторів та критеріїв виконує адаптивне визначення мутаційних змін [173]. Система індикаторів використовується для оцінки існуючої структури ШНМ. Враховуючи отриману оцінку за структурними особливостями та спираючись на критеріальні значення, обирається тип структурного доналаштування. У якості варіантів структурного доналаштування використовується варіанти мутаційних змін із групи методів нейроеволюційної модифікації топології і ваг нейромережі. Запропонований

метод дозволяє знизити ресурсоємність під час роботи нейромоделі, за рахунок пришвидшення обробки великих даних. Практичне дослідження структурного доналаштування та спрощення попередньо синтезованих ШНМ засвідчило, що, враховуючи полегшену структуру, такі нейромоделі можна використовувати при роботі з великими об'ємами вхідних даних. Результати експериментів показали, що запропонований метод дозволяє більш прорідити ШНМ для її подальшого використання у якості моделі для діагностування, прогнозування, оцінювання та моделювання.

Таким чином, узагальнюючи отримані результати проведених експериментів, можна відзначити, що застосування запропонованих у дисертаційній роботі методів синтезу діагностичних моделей на основі РНМ дозволило значно підвищити точність синтезу нейромоделей у порівнянні з методами, які передбачають послідовну реалізацію обчислень, збільшити рівні інтерпретовності та логічної прозорості моделей. Розроблені методи можуть застосовуватися до широкого кола задач діагностування, де є накопичені дані про об'єкт діагностування проте не є достатніми експертні знання та можуть бути використані при синтезі діагностичних моделей на основі великих вибірок даних. Розроблені методи синтезу діагностичних моделей на основі РНМ можуть знайти широке застосування та практичне використання при створенні інтелектуальних систем діагностування у різних галузях промисловості та економіки, зокрема у авіадвигунобудуванні, військовій промисловості, приладобудуванні, медицині тощо – приклади практичного впровадження, що підтверджено відповідними актами наведено в додатку Б.

Результати, наведені у четвертому розділі, було опубліковано в роботах [75], [120], [139], [151], [173], [175]–[179], [182], [183], [190], [192]–[202].

## ВИСНОВКИ

У дисертаційній роботі вирішено актуальну науково-прикладну проблему розроблення та дослідження методів синтезу рекурентних нейромережових діагностичних моделей, які поєднують принципи інтелектуальних та паралельних обчислень, що дозволяє підвищити точність діагностичних моделей, збільшити їх інтерпретовність та узагальнювальні можливості.

Проведені дослідження дозволили зробити такі висновки.

1. Проведено аналіз процесу та методів синтезу діагностичних моделей на базі рекурентних нейронних мереж [75], [120]. Визначено, що для синтезу рекурентних нейронних мереж можуть ефективно застосовуватися методи нейроеволюційного підходу, які одночасно забезпечують виконання і структурного синтезу, і параметричного синтезу нейромоделей. Встановлено, що суттєвим недоліками існуючих нейроеволюційних методів є обмеженість початкових топологій нейромереж, неможливість еволюційної модифікації виду функції активації, виключно вертикальна еволюція по шляху ускладнення структури мережі та в деяких випадках жорсткі вимоги до рівня репрезентативності вибірки даних. Обґрунтовано необхідність розробки нових методів синтезу діагностичних моделей на основі рекурентних нейронних мереж, що забезпечать одночасний синтез структури та параметрів нейромережі, що буде відрізнятися високою точністю, а подальше використання паралельних обчислень дозволить прискорити процес нейроеволюційного синтезу за рахунок використання високопродуктивних методів, які відрізняються скороченням часу синтезу навіть на великих вибірках історичних даних при побудові систем діагностування.

2. Розроблено генетичний метод для синтезу рекурентних нейронних мереж [120], [182], [183], [195], в якому, на відміну від існуючих, на етапі кодування використовується спосіб кодування інформації про штучні нейронні мережі на основі секвенування, що дозволяє мінімізувати

ймовірність завчасної збіжності. Використання запропонованих критеріїв регулювання розміру та напрямку розвитку штучних нейронних мереж у розробленому методі дозволяє на етапі мутації адаптивно вибирати тип перетворення структури нейромережі, що перешкоджає безмірному збільшенню мережі, не обмежуючи кількість прихованих нейронів, та синтезувати рекурентні нейронні мережі, що характеризуються високою точністю (до 98%). Розроблений метод дозволяє, оптимізуючи використання ресурсів під час синтезу, отримати діагностичні моделі з високою точністю роботи.

3. Розроблено паралельний генетичний метод з імплементацією механізмів селективного тиску [120], [139], [194], [196], [197], в якому основні етапи процесу еволюційного синтезу рекурентних нейронних мереж виконуються на паралельних обчислювальних вузлах, використовуються механізми селективного тиску та рівномірного схрещування, що дозволяє скорочувати розміри популяції, не враховуючі та не опрацьовуючі ті особини з популяції, що відрізняються малим значенням функції пристосованості. Застосування модифікованих генетичних операторів відбору та схрещування дозволяє підтримувати генетичне різноманіття популяцій, забезпечити уникання проблем з надмірною ітеративною природою еволюційних обчислень, враховувати додаткові показники якості нейромережових моделей, більш продуктивно використовувати механізми рівномірного багатобатьківського схрещування. Збільшення точності (до 96-98%) та скорочення витрат пам'яті й обчислювальних потужностей для збереження та проведення схрещування повного обсягу популяції засвідчують помірний рівень ресурсоемності запропонованого методу.

4. Запропоновано систему індикаторів для оцінювання рівня складності задачі для подальшого моделювання, діагностування або прогнозування [151]. Так, враховуючи характеристики вхідного набору даних, рівень можливого спрощення структури, загальну кількість суттєвих та несуттєвих факторів, рівень точності вимірювань та рівень можливого

контролю та управління, можливо в подальшому обрати варіант доналаштування математичної моделі. Такий підхід дозволяє попередньо проаналізувати та визначити механізми доналаштування штучних нейронних мереж без використання надмірних обчислювальних ресурсів для повторного синтезу нейромоделей. Таким чином, запропоновані індикатори не вимагають додаткової побудови моделі, що істотно знижує використання часових та обчислювальних ресурсів під час процесу аналізу варіантів підбору та розвитку математичної моделі.

5. Запропоновано метод структурного доналаштування для підвищення характеристик інтерпретовності штучних нейронних мереж [173]. На відміну від застосування складних топологій штучних нейронних мереж під час роботи із великими даними, що часто обмежується обчислювальними ресурсами, запропонований метод структурної оптимізації дозволяє отримати полегшену та проріджену структуру попередньо синтезованої нейромоделі. Це гарантує не тільки зменшення навантаження на обчислювальну систему, а й пришвидшення роботи (в 1,5 рази) такої штучної нейронної мережі. Більш того, у якості побічного ефекту структурної оптимізації іноді може спостерігатися підвищення точності роботи (до 96-97%), адже частина вузлів та зв'язків для кодування зашумлених даних може бути видалена.

6. Проведено експериментальне дослідження розроблених методів синтезу та доналаштування діагностичних моделей. Результати експериментів показали, що застосування запропонованих у дисертаційній роботі методів синтезу діагностичних моделей дозволило значно (на 10-40%) підвищити точність діагностування синтезованих моделей у порівнянні з іншими нейроеволюційними методами, а також збільшити рівні інтерпретовності та узагальнення даних за допомогою синтезованих моделей.

7. Практичні результати підтверджені актами впровадження та доводять коректність теоретичних положень дисертаційної роботи. Результати дисертаційної роботи апробовано та впроваджено у практичну

діяльність Товариства з обмеженою відповідальністю «Аптека Магнолія» (акт впровадження від 18.08.2022), у практичну діяльність Товариства з обмеженою відповідальністю «Запорізький ливарно-механічний завод» (акт впровадження від 20.09.2022).

Наукові положення, висновки і рекомендації, викладені в дисертаційній роботі, використано при підготовці дисциплін «Інтелектуальні системи», «Основи обчислювального інтелекту», «Інтелектуальний аналіз даних» на кафедрі програмних засобів Національного університету «Запорізька політехніка» (акт впровадження від 20.12.2022).

Можливі напрями подальших досліджень за тематикою дисертації можуть бути пов'язані з розробленням методів синтезу діагностичних та розпізнавальних моделей на основі глибоких нейронних мереж для застосування та практичного використання у різних галузях науки і техніки, пов'язаних зі створенням інтелектуальних систем діагностування, зокрема у авіадвигунобудуванні, військовій промисловості, приладобудуванні, медицині тощо.



**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Hatch, S.V. Computerized Engine Controls / S.V. Hatch. – Boston: Cengage Learning, 2016. – 688 p.
2. Czichos, H. Measurement, Testing and Sensor Technology. Fundamentals and Application to Materials and Technical Systems / H. Czichos. – Berlin: Springer, 2018. – 213 p.
3. Kaźmierczak, J. Data Processing and Reasoning in Technical Diagnostics / J. Kaźmierczak, W. Cholewa. – Warszawa: Wydawnictwa Naukowo-Techniczne, 1995. – 186 p.
4. Diagnostics as a Reasoning Process: From Logic Structure to Software Design / [M. Cristani, F. Olivieri, C. Tomazzoli, L. Vigano, M. Zorzi] // Journal of Computing and Information Technology. – 2018. – Vol. 27 (1). – P. 43-57.
5. Machine Learning Techniques in Additive Manufacturing: a State of the Art Review on Design, Processes and Production Control / [S. Saxena, T. & Gopi, N. Harikeerthana, M. Gupta, V. Gaur, G. Krolczyk, C.S. Wu] // Journal of Intelligent Manufacturing. – 2022. – Vol. 1. – P. 1-35.
6. Wieczorek, A.N. Analysis of the Possibility of Integrating a Mining Right-Angle Planetary Gearbox with Technical Diagnostics Systems / A.N. Wieczorek // Scientific Journal of Silesian University of Technology. Series Transport. – 2016. – Vol. 93. – P. 149-163.
7. Tso, B. Classification Methods for Remotely Sensed Data / B. Tso, P.M. Mather. – Boca Raton : CRC Press, 2016. – 352 p.
8. Rokach, L. Ensemble Learning: Pattern Classification Using Ensemble Methods / L. Rokach. – Singapore : World Scientific Publishing Co Pte Ltd., 2019. – 302 p.
9. Deisenroth, M.P. Mathematics for Machine Learning / M.P. Deisenroth. – Cambridge : Cambridge University Press, 2020. – 398 p.

10. Golden, R. *Statistical Machine Learning: a Unified Framework* / R. Golden. – London : Chapman and Hall, 2020. – 506 p.
11. Hardt, M. *Patterns, Predictions, and Actions: Foundations of Machine Learning* / M. Hardt, B. Recht. – Princeton : Princeton University Press, 2022. – 320 p.
12. Graupe, D. *Principles of Artificial Neural Networks: Basic Designs to Deep Learning* / D. Graupe. – Singapore: World Scientific Publishing Co Pte Ltd., 2019. – 440 p.
13. Russell, S. *Human Compatible: Artificial Intelligence and the Problem of Control* / S. Russell. – London : Penguin Books, 2020. – 352 p.
14. Sarti, S. *Recombination and Novelty in Neuroevolution: a Visual Analysis* / S., J. Adair, G. Ochoa // *SN Computer Science*. – 2022. – Vol. 3(3). – P. 185-200.
15. *Towards the Neuroevolution of Low-level artificial general intelligence* / [S. Pontes-Filho, K. Olsen, A. Yazidi, M. A. Riegler] // *Frontiers in Robotics and AI*. – 2022. – Vol. 9. – P. 1-15.
16. Rodrigues, N. M. *A Study of Fitness Landscapes for Neuroevolution* / Nuno M. Rodrigues, Sara Silva, Leonardo Vanneschi // *2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, United Kingdom, 19–24 July 2020 : proceedings*. – Glasgow : IEEE, 2020. – P. 1-8.
17. *Algorithm of Technical Diagnostics of the Complicated Damage to the Continued Resource of the Circulation Pipeline of the Nuclear Power Plant* / [H. Hrinchenko, R. Trisch, V. Burdeina, S. Chelysheva] // *Problems of Atomic Science and Technology*. – 2019. – Vol. 120(2). – P. 104-110.
18. Salem, F.M. *Recurrent Neural Networks: from Simple to Gated Architectures* / F.M. Salem. – Berlin : Springer. – 2022. – 141 p.
19. Sahu, N. *Recurrent Neural Network and Application* / N. Sahu. – Seattle : Kindle Direct Publishing, 2022. – 84 p.

20. Bergel, A. Agile Artificial Intelligence in Pharo: Implementing Neural Networks, Genetic Algorithms, and Neuroevolution / A. Bergel. – New York : Apress, 2020. – 407 p.
21. Boers, E.J.W. Evolving Neural Networks Using the «Baldwin Effect» / E.J.W. Boers, M.V. Borst, I.G. Sprinkhuizen-Kuyper // Artificial Neural Nets and Genetic Algorithms. – 1995. – Vol. 1. – 333-336 p.
22. Blokdyk, G. Neuroevolution of Augmenting Topologies: Second Edition / G. Blokdyk. – Ohio : 5STARCOoks, 2018. – 128 p.
23. Gruau, F. Automatic Definition of Modular Neural Networks / F. Gruau // Adaptive Behaviour. – 1995. – Vol. 3(2). – P. 151–183.
24. Takagi, T. Fuzzy Identification of Systems and Its Application to Modeling and Control / T. Takagi, M. Sugeno // IEEE Transactions on Systems, Man, and Cybernetics. – Vol. SMC-15, Issue 1. – 1985. P. 116-132.
25. Ivakhnenko, G.A. Self-Organization of Neuronet with Active Neurons for Effects of Nuclear Test Explosions Forecastings / G.A. Ivakhnenko // System Analysis Modeling Simulation. – 1995. – Vol. 20. – P.107-116.
26. Бодянский, Е. В. Нейро-фаззи сети Петри в задачах моделирования сложных систем / Е. В. Бодянский, Е. И. Кучеренко, А. И. Михалев. – Днепропетровск : Системные технологии, 2005. – 311 с.
27. Бодяньський, Є. В. Нейро-фаззі моделі в системах штучного інтелекту : навч. посібник / Є. В. Бодяньський, Є. І. Кучеренко. – Харків : ХНУРЕ, 2006. – 196 с.
28. Дмитриенко, В.Д. Моделирование и оптимизация процессов управления движением дизель-поездов / В.Д. Дмитриенко, А.Ю. Заковоротный. – Харьков : Изд. Центр "НТМТ", 2013. – 248 с.
29. Beihelt, F. Zuverlässigkeit und Instandhaltung: Mathematische Methoden / F. Beihelt and P. Franken. – Berlin : Verlag Technik, 1983. – 315 p.
30. Карпаш, Я. М. Про деякі проблемні питання методичного забезпечення у сфері технічної діагностики / О. М. Карпаш, Я. М. Зінчак, М.

О. Карпаш // Технічна діагностика та неруйнівний контроль. – №3. – 2010. – С. 55-58.

31. ДСТУ 2389-94. Технічне діагностування та контроль технічного стану. Терміни та визначення. – Чинний від 01.07.95 Київ : Держстандарт України, 1995. – 75с.

32. Яременко, В. Методологія технічного діагностування сільськогосподарської техніки за граничним станом: монографія / В. Яременко, О. Черниш. – Київ : Центр навчальної літератури, 2018. – 602 с.

33. Войтюк, В.Д. Теоретичне обґрунтування об'єктів і засобів діагностування при їх сертифікації / В.Д. Войтюк. – Київ : Центр навчальної літератури, 2022. – 264 с.

34. Пріб К. А. Діагностика в системі управління / К.А. Пріб, Н.І. Патица. – Київ : Центр навчальної літератури, 2019. – 432 с.

35. Барау Н. І. Вібраційна діагностика протизсувних захисних споруд / Н. І. Барау. – Київ : Центр навчальної літератури, 2022. – 142 с.

36. Технічна діагностика матеріалів і конструкцій : довідниковий посібн. у 8–ми т. / за заг. ред. З. Т. Назарчука. – Львів. : Простір–М, 2018 – . Т. 4 : Електрофізичні методи неруйнівного контролю дефектності елементів конструкцій / Р. М. Джала, В. Р. Джала, І. Б. Івасів [та ін.] ; НАН України, Фізикомеханічний інститут імені Г. В. Карпенка. – Л. : Простір–М, 2018. – 356 с.

37. Liu, P. Neural Network Evolution Using Expedited Genetic Algorithm for Medical Image Denoising / [P. Liu, Y. Li, M.D. El Basha, R. Fang] // Medical Image Computing and Computer Assisted Intervention (MICCAI 2018) : 21st International Conference, Berlin, Germany, 16-18 September 2018 : proceedings. – Berlin : Nature Springer, 2018. – P.12–20.

38. Capellman, J. Hands-On Machine Learning with ML.NET: Getting Started with Microsoft ML.NET to Implement Popular Machine Learning Algorithms in C# / J. Capellman. – Birmingham : Packt Publishing, 2020. – 296 p.

39. Prospects and Limitations of Non-Invasive Blood Glucose Monitoring Using Near-Infrared Spectroscopy / [J. Yadav, A. Rani, V. Singh, B. Murari] // Biomedical Signal Processing and Control. – 2015. – Vol. 18. – P. 214–227.
40. Babichev, S. Methods, Models and Information Technology of Complex Data Processing in the Fields of Technical Diagnostics and Bioinformatics / S. Babichev, B. Durnyak. – Lviv : Ukrainian Academy of Printing, 2020. – 180 p.
41. Kozlovskiy, O. A Measuring Transducer in the System of Technical Diagnosis of Overhead Lines Icing in Electrical Distribution Networks / O. Kozlovskiy, D. Trushakov, S. Rendzinyak // Computational Problems of Electrical Engineering. – 2015. – Vol. 5, № 1. – P. 17-22.
42. Жиров, Г.Б. Алгоритми моделювання процесів технічного обслуговування за станом складних радіоелектронних об'єктів / Г. Б. Жиров, Є. С. Ленков // Радіоелектроніка, інформатика, управління. – 2018. – № 2. – С. 14-21.
43. Кучмистенко, О. В. Підвищення ефективності автоматизованої системи керування процесом транспортування нафти на засадах інтелектуальних технологій / О. В. Кучмистенко // Розвідка та розробка нафтових і газових родовищ. – 2018. – № 2. – С. 7-13.
44. Удосконалення метрологічної експертизи військової техніки зв'язку / [М. Ю. Яковлев, Л. М. Сакович, Є. В. Рижов, П. Л. Аркушенко] // Наука і техніка Повітряних сил Збройних сил України. – 2017. – № 3. – С. 92-99.
45. Матусевич, О. О. Метод визначення технічного ресурсу силового трансформатора тягових підстанцій в умовах експлуатації / О. О. Матусевич, М. В. Хворост, В. В. Малишева // Східно-Європейський журнал передових технологій. – 2017. – № 3/8. – С. 4-9.
46. Лосев, М.Ю. Методика синтезу пристроїв контролю і тестової діагностики розподілених систем / М. Ю. Лосев // Збірник наукових праць Харківського університету Повітряних сил. – 2018. – № 2. – С. 89-92.

47. Технічна діагностика матеріалів і конструкцій [Текст] : довідниковий посібн. у 8–ми т. / за заг. ред. З. Т. Назарчука. – Львів : Простір–М, 2017 – . Т. 2 : Математичне моделювання взаємодії фізичних полів із дефектами матеріалу / З. Т. Назарчук, Д. Б. Куриляк, В. В. Михаськів [та ін.] ; НАН України, Фізикомеханічний інститут імені Г. В. Карпенка. – Л. : Простір–М, 2018. – 512 с.

48. Аналіз результатів технічного діагностування кожухотрубних теплообмінників для ранжування експлуатаційних дефектів та оцінки їх впливу на залишковий ресурс / [М. Д. Рабкіна, А. О. Перепічай, І. І. Перепічай, В. В. Мутас] // Технічна діагностика та неруйнівний контроль. – 2017. – № 2. – С. 50-54.

49. Smith, D.J. Reliability, Maintainability and Risk: Practical Methods for Engineers / D.J. Smith. – Oxford: Butter-worth-Heinemann, 2021. – 516 p.

50. System Assurances: Modeling and Management / [P. Johri, A. Anand, J. Vain, J. Singh, M.T. Quasim]. – Cambridge: Academic Press, 2022. – 614 p.

51. Batyuk, L. Monitoring System of Biophysical Properties of the Red Blood Cells of Patients for Medical Diagnosis Purposes / L. Batyuk, N. Kizilova // Системи обробки інформації. – 2020. – Вип. 3. – С. 13-20.

52. Kravchenko, O. Development Of Medical Diagnostic Decision Support Systems and Their Economic Efficiency / O. Kravchenko // Technology Audit and Production Reserves. – 2018. – № 2(2). – С. 4-10.

53. Discovering Workflow Nets Using Integer Linear Programming / [S.J. van Zelst, B.F. van Dongen, W.M.P. van der Aalst, H.M.W. Verbeek] // Computing. – 2018. – Vol. 100. – P. 529–556.

54. Rathore, S.S. A Decision Tree Logic Based Recommendation System to Select Software Fault Prediction Techniques / S. S. Rathore, S. Kumar // Computing. – 2017. – Vol. 99. – P. 255–285.

55. Kim, T. Incorporating Receiver Operating Characteristics into Naive Bayes for Unbalanced Data Classification / [T. Kim, B.D. Chung, J.S. Lee] // Computing. – 2017. – Vol. 99. – P. 203-218.

56. Automated Nuclei Detection in Serous Effusion Cytology Based on Machine Learning / [E. Baykal, H. Doğan, M. Ekinçi, M. E. Ercin, Ş. Ersöz] // 2017 25th Signal Processing and Communications Applications Conference (SIU), Antalya, Turkey, 15-18 May 2017 : proceedings. – Antalya, IEEE, 2017. – P. 1-4.
57. Perova, I. Fast Medical Diagnostics Using Autoassociative Neuro-fuzzy Memory / I. Perova, Ye. Bodyanskiy // International Journal of Computing. – 2017. – Vol. 16, Issue 1. – P. 34-40.
58. Artificial Intelligence-Based Smart Power Systems / [S. Padmanaban, S. Palanisamy, S. Chenniappan, J.B. Holm-Nielsen]. – New York : Wiley-IEEE Press, 2022. – 400 p.
59. Firsov, S. N. Intelligent Support of Multilevel Functional Stability of Control and Navigation Systems / S. N. Firsov, O. A. Pishchukhina // Radio Electronics, Computer Science, Control. – 2018. – Vol. 2. – P. 177-183.
60. Башинська, О. Метод діагностики компонентів системи керування безпілотного авіаційного комплексу / О. Башинська, Ю. Камак, С. Нестеренко // Технічні науки та технології. – 2018. – № 2. – С. 142-150.
61. Нечітка система діагностування патологічних станів молочної залози на основі гістологічних зображень / [Л. О. Дубчак, С. О. Вербовий, Н. А. Максимів, Т. В. Дацко] // Інформатика та матичні методи в моделюванні. – 2017. – 7, № 1/2. – С. 47-53.
62. Berezsky, O. M. Evaluation Methods of Image Segmentation Quality / O. M. Berezsky, O. Y. Pitsun // Radio Electronics, Computer Science, Control. – 2018. – Vol. 1. – P. 119-128.
63. Притискач, І. В. Особливості використання уточнених теплових моделей силового трансформатора в системах діагностування / І. В. Притискач // Енергетика: економіка, технології, екологія. – 2017. – № 2. – С. 13-20.
64. Scalable Training of Artificial Neural Networks with Adaptive Sparse Connectivity Inspired by Network Science / [D. Mocanu, E. Mocanu, P. Stone,

P. Nguyen, M. Gibescu, A. Liotta] // Nature Communications. – 2018. – № 9. – P. 1-17.

65. Goodfellow, I. Deep Learning / I. Goodfellow, Y. Bengio, A. Courville // Deep Learning. – Massachusetts: MIT Press, 2016. – 800 p. – (Adaptive Computation and Machine Learning series).

66. A Review on Neural Networks with Random Weights / [W. Cao, X. Wang, Z. Ming, J. Gao] // Neurocomputing. – 2018. – Vol. 275. – P. 278-287.

67. Convolutional Neural Networks for Histopathology Image Classification: Training vs. Using Pre-Trained Networks / [B. Kieffer, M. Babaie, S. Kalra et al.] // Conference on Image Processing Theory, Tools and Applications (IPTA 2017) : Seventh International conference, Montreal, Canada, 28 November – 1 December 2017 : proceedings. – Montreal: IEEE, 2017. – P. 1-6.

68. Landslide Spatial Modeling: Introducing New Ensembles of ANN, MaxEnt, and SVM Machine Learning Techniques / [W. Chen, H.R. Pourghasemi, A. Kornejady, N. Zhang] // Geoderma. – 2017. – Vol. 305 – P. 314-327.

69. Combining Desorption Electrospray Ionization Mass Spectrometry Imaging and Machine Learning for Molecular Recognition of Myocardial Infarction / [K. Margulis, Z. Zhou, Q. Fang, R.E. Sievers, R.J. Lee, R.N. Zare] // Analytical Chemistry. – 2018. – Vol. 90, Issue 20. – P. 12198-12206.

70. Fast and Accurate Estimation of Quality of Results in High-Level Synthesis with Machine Learning / [S. Dai, Y. Zhou, H. Zhang, E. Ustun, E.F.Y. Young, Z. Zhang] // 2018 IEEE 26th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), Boulder, CO, USA, 29 April-1 May 2018 : proceedings. – Boulder : IEEE, 2018. – P. 129-132.

71. Exploration of the Nanomedicine-Design Space with High-Throughput Screening and Machine Learning / [G. Yamankurt, E.J. Berns, A. Xue, A. Lee, N. Bagheri, M. Mrksich, C.A. Mirkin] // Nature Biomedical Engineering. – 2019. – Vol. 3, Issue 4. – P. 318-327.

72. Жильцов, А. В. Математичне моделювання електротехнічних систем та їх елементів [Текст] : навч. посіб. для студентів ВНЗ : [в 2 т.] /



Жильцов А. В., Мірських Г. О., Реутська Ю. Ю. - Київ : Компринт, 2017 – . Т. 1 : Багатополюсники. - 2017. - 483 с.

73. Кононова, К.Ю. Машинне навчання. Методи та моделі : підручник / К. Ю. Кононова ; Харків. нац. ун-т ім. В. Н. Каразіна. – Харків : ХНУ ім. В. Н. Каразіна, 2020. – 279 с.

74. Тимощук, П.В. Принципи штучних нейронних мереж та їх застосування : навч. посіб. / П.Тимощук, М. Лобур ; Нац. ун-т "Львів. політехніка". – Львів : Вид-во Львівівської політехніки, 2020. – 291 с.

75. Using Recurrent Neural Networks for Data-Centric Business / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko] // Data-Centric Business and Applications - Evolvments in Business Information Processing and Management. – Cham, Switzerland: Springer, 2020. – P. 73-91. – (Lecture Notes on Data Engineering and Communications Technologies, Vol. 3).

76. Kanaan, M. T-Minus AI: Humanity's Countdown to Artificial Intelligence and the New Pursuit of Global Power / M. Kanaan. – Dallas : BenBella Books, 2019. – 270 p.

77. Taulli, T. Artificial Intelligence Basics: a Non-Technical Introduction / T. Taulli. – New York : Apress, 2019. – 199 p.

78. Rothman, D. Artificial Intelligence By Example: Acquire advanced AI, machine learning, and deep learning design skills / D. Rothman. – Birmingham : Packt Publishing, 2020. – 578 p.

79. Chang, A.C. Intelligence-Based Medicine: Artificial Intelligence and Human Cognition in Clinical Medicine and Healthcare / A.C. Chang. – Cambridge : Academic Press, 2020. – 534 p.

80. Ponteves, de P. AI Crash Course: A Fun and Hands-On Introduction to Machine Learning, Reinforcement Learning, Deep Learning, and Artificial Intelligence With Python / P. de Ponteves. – Birmingham : Packt Publishing, 2019. – 360 p.

81. Artasanchez, A. Artificial Intelligence with Python: Your Complete Guide to Building Intelligent Apps Using Python 3.x / A. Artasanchez, P. Joshi. – Birmingham : Packt Publishing, 2020. – 618 p.
82. Hiesinger, P.R. The Self-Assembling Brain: How Neural Networks Grow Smarter / P.R. Hiesinger. – Princeton: Princeton University Press, 2021. – 384 p.
83. Ekman, M. Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow / M. Ekman. – Boston : Addison-Wesley Professional, 2021. – 752 p.
84. Russell, R. Neural Networks: Easy Guide To Artificial Neural Networks / R. Russell. – Scotts Valley : CreateSpace Independent Publishing Platform, 2018. – 80 p.
85. Glassner, A. Deep Learning: A Visual Approach / A. Glassner. – San Francisco : No Starch Press, 2021. – 776 p.
86. Smith-Miles, K. Neural Networks in Business: Techniques and Applications for the Operations Researcher / K. Smith-Miles, J. Gupta // Computers and Operations Research. – 2000. – Vol. 27, Issue 11. – P. 1023–1044.
87. Van Tuc, N. Approximation Contexts in Addressing Graph Data Structures / N. Van Tuc. – Wollongong: University of Wollongong, 2015. – 230 p.
88. Handa, A. Backpropagation in Cnvolutional LSTMS. Tutorial / A. Handa, V. Patraucean. – Cambridge : University of Cambridge, 2015. – P. 1–5.
89. Boden, M. A Guide to Recurrent Neural Networks and Back-Propagation [Electronic resource]. – Access mode: [https://axon.cs.byu.edu/~martinez/classes/678/Papers/RNN\\_Intro.pdf](https://axon.cs.byu.edu/~martinez/classes/678/Papers/RNN_Intro.pdf).
90. Guo, J. BackPropagation Through Time [Electronic resource]. – Access mode: <http://ir.hit.edu.cn/~jguo/docs/notes/bppt.pdf>.
91. Yue, B. Residual Recurrent Neural Networks for Learning Sequential Representations / B. Yue, J. Fu, J. Liang // Information. – 2018. – Vol. 9, Issue 56. – P. 1–14.

92. Eluyodel, O.S. Comparative Study of Biological and Artificial Neural Networks / O. S. Eluyodel, D. T. Akomolafe // European Journal of Applied Engineering and Scientific Research. – 2013. – Vol. 2, Issue 1. – P. 36-46.
93. Blokdyk, G. Backpropagation Complete Self-Assessment Guide / G. Blokdyk. – Ohio : 5STARCOoks, 2022. – 306 p.
94. Sisodia, D.S. A Practical Approach for Sign Language's Static Gesture Recognition: Using Backpropagation Training Algorithm and Skin Color Modeling Method / D.S. Sisodia. – Chisinau : Scholars' Press, 2017. – 100 p.
95. Iba, H. Evolutionary Approach to Machine Learning and Deep Neural Networks: Neuro-Evolution and Gene Regulatory Networks / H. Iba. – New York : Springer, 2018. – 258 p.
96. Lockett, A.J. General-Purpose Optimization Through Information Maximization (Natural Computing Series) / A.J. Lockett. – New York : Springer, 2020. – 579 p.
97. Rouhiainen, L. Artificial Intelligence: 101 Things You Must Know Today About Our Future / L. Rouhiainen. – Scotts Valley : CreateSpace Independent Publishing Platform, 2018. – 300 p.
98. Koul, A. Practical Deep Learning for Cloud, Mobile, and Edge: Real-World AI & Computer-Vision Projects Using Python, Keras & TensorFlow / A. Koul, S. Ganju, M. Kasam. – Newton : O'Reilly Media, 2019. – 620 p.
99. Singh, A. Mobile Deep Learning with TensorFlow Lite, ML Kit and Flutter: Build Scalable Real-World Projects to Implement End-To-End Neural Networks on Android and iOS / A. Singh, R. Bhadani. – Birmingham : Packt Publishing, 2020. – 380 p.
100. Davies, J. The Internet of Things: From Data to Insight / J. Davies, C. Fortuna. – Hoboken : Wiley, 2020. – 240 p.
101. Zheng, N. Learning in Energy-Efficient Neuromorphic Computing: Algorithm and Architecture Co-Design / N. Zheng, P. Mazumder. – New York : Wiley-IEEE Press, 2019. – 296 p.

102. Recurrent Neural Networks for Short-Term Load Forecasting: An Overview and Comparative Analysis (SpringerBriefs in Computer Science) / [F.M. Bianchi, E. Maiorino, M.C. Kampffmeyer, A. Rizzi, R. Jenssen]. – New York : Springer, 2017. – 81 p.

103. Zavertanyy, V. V. Genotype Dynamic for Agent Neuroevolution in Artificial Life Model / V. V. Zavertanyy, A. S. Makarenko // Системні дослідження та інформаційні технології. – 2017. – № 1. – С. 75-87.

104. New Artificial Life Boid Model Based on Neural Networks / [B. V. Sachkovskiy, O. M. Agafonov, D. B. Gryaznov, Yu. V. Boyko] // Bulletin of Taras Shevchenko National University of Kyiv. Series: Physics and Mathematics. – 2017. – Vol. 3. – P. 281-284.

105. Nissen, V. A Brief Introduction to Evolutionary Algorithms from the Perspective of Management Science / V. Nissen // Innovative Research Methodologies in Management. – 2018. – P. 165-210.

106. Soltoggio, A. Born to Learn: The inspiration, Progress, and Future of Evolved Plastic Artificial Neural Networks / A. Soltoggio, K. O. Stanley, S. Risi // Neural Networks. – 2018. – Vol. 108. – P. 48-67.

107. Kearney, W. T. Using Genetic Algorithms to Evolve Artificial Neural Networks [Electronic resource]. – Access mode: <https://digitalcommons.colby.edu/cgi/viewcontent.cgi?article=1836&context=honorsthesis>.

108. Shabash, B. EvoNN: a Customizable Evolutionary Neural Network With Heterogenous Activation Functions / B. Shabash, K. Wiese // The Genetic and Evolutionary Computation Conference Companion : A Recombination of the 27th International Conference on Genetic Algorithms (ICGA) and the 23rd Annual Genetic Programming Conference (GP), Kyoto, Japan, 15-19 July 2018 : proceedings. – New York: ACM, 2018. – P. 1449-1456.

109. Baldominos, A. On the Automated, Evolutionary Design of Neural Networks-Past, Present, and Future / A. Baldominos, Y. Sáez, P. Isasi // Neural Computing and Applications. – 2019. – Vol. 32(2) – P. 1-45.

110. Baldominos, A. Evolutionary Design of Convolutional Neural Networks for Human Activity Recognition in Sensor-Rich Environments / A. Baldominos, Y. Sáez, P. Isasi // *Sensors*. – 2018. – Vol. 18, Issue 4. – P. 1-24.

111. Behjat, A. Adaptive Genomic Evolution of Neural Network Topologies (AGENT) for State-to-Action Mapping in Autonomous Agents / A. Behjat, S. Chidambaran, S. Chow-dhury // *Conference on Robotics and Automation (ICRA' 2019) : IEEE International Conference, Montreal, Canada, 20-24 May 2019 : proceedings*. – Montreal: IEEE, 2019. – P. 1-7.

112. Pagliuca, P. Maximizing Adaptive Power in Neuroevolution / P. Pagliuca, N. Milano, S. Nolfi // *PLoS ONE*. – 2018. – Vol. 13, Issue 7. – P. 1-27.

113. Deep Neuroevolution: Genetic Algorithms Are a Competitive Alternative for Training Deep Neural Networks for Reinforcement Learning [Electronic resource]. – Access mode: <https://arxiv.org/abs/1712.06567>.

114. A Tensor-Based Mutation Operator for Neuroevolution of Augmenting Topologies (NEAT) / [A. Marzullo, C. Stamile, G. Terracina, F. Calimeri, S. Huffel] // *IEEE Congress on Evolutionary Computation (CEC), Donostia, San Sebastián, Spain, 5-8 June 2017 : proceedings*. – NY: IEEE, 2017. – P. 681-687.

115. Omelianenko, I. Performance Eevaluation of NEAT Algorithm's Implementation in GO Programming Language [Electronic resource]. – Access mode: [https://www.researchgate.net/publication/325854836\\_Performance\\_evaluation\\_of\\_NEAT\\_algorithm's\\_implementation\\_in\\_GO\\_programming\\_language](https://www.researchgate.net/publication/325854836_Performance_evaluation_of_NEAT_algorithm's_implementation_in_GO_programming_language).

116. Kadish, D. Clustering Sensory Inputs Using NeuroEvolution of Augmenting Topologies / D. Kadish // *The Genetic and Evolutionary Computation Conference Companion : A Re-combination of the 27th International Conference on Genetic Algorithms (ICGA) and the 23rd Annual Genetic Programming Conference (GP), Kyoto, Japan, 15-19 July 2018 : proceedings*. – New York: ACM, 2018. – P. 1-2.

117. Jansen, S. Machine Learning for Algorithmic Trading: Predictive Models to Extract Signals from Market and Alternative Data for Systematic

Trading Strategies with Python, 2nd Edition / S. Jansen. – Birmingham: Packt Publishing, 2020. – 820 p.

118. Buontempo, F. Genetic Algorithms and Machine Learning for Programmers: Create AI Models and Evolve Solutions / F. Buontempo. – USA, Sebastopol : Pragmatic Bookshelf, 2019. – 236 p.

119. Ozkan, L. RNA Sequencing: Principles and Data Analysis / L. Ozkan. – Traverse City : Independently published, 2020. – 118 p.

120. Modification and Parallelization of Genetic Algorithm for Synthesis of Artificial Neural Networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Lytvyn, V. Shkarupylo] // Radio Electronics, Computer Science, Control. – 2019. – № 4. – P. 68–82.

121. Robinson, T.R. Genetics for Dummies / T.R. Robinson, L. Spock. – New York : For Dummies, 2020. – 400 p.

122. Tan, T.W. Beginners Guide to Bioinformatics for High Throughput Sequencing / T.W. Tan, E. Lee. – Singapore : World Scientific Publishing Co Pte Ltd, 2018. – 300 p.

123. Weightless: Lossy Weight Encoding For Deep Neural Network Compression / [B. Reagen, U. Gupta, R. Adolf, M. Mitzenmacher, A. Rush et al.] // International Conference on Machine Learning (ICML 2018), Stockholmsmässan, Sweden, 10-15 July : proceedings. – Stockholmsmässan, PMLR, 2018. – P. 1-10.

124. Introduction to Algorithms / [T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein]. – Cambridge : The MIT Press, 2009. – 1292 p.

125. Neural Network Evolving Algorithm Based on the Triplet Codon Encoding Method / [X. Yang, S. Deng, M. Ji, J. Zhao, W. Zheng] // Genes (Basel). – 2018. – Vol. 9(12). – P. 1-30.

126. Nam, C.S. Neuroergonomics: Principles and Practice / C.S. Nam. – Berlin : Springer, 2020. – 492 p.

127. Intelligent Systems and Learning Data Analytics in Online Education / [S. Caballé, S.N. Demetriadis, E.G.-Sánchez, P.M. Papadopoulos, A. Weinberger]. – Cambridge : Academic Press, 2021. – 424 p.

128. Aggarwal, C.C. Neural Networks and Deep Learning: A Textbook / C.C. Aggarwal. – Berlin : Springer, 2018. – 520 p.

129. D'souza, R.N. Structural Analysis and Optimization of Convolutional Neural Networks with a Small Sample Size / R.N. D'souza, P.Y. Huang, F.C. Yeh // Scientific Reports. – 2020. – Vol. 10. – P. 1–13.

130. Nowakowski, G. Neural Network Structure Optimization Algorithm / G. Nowakowski, Y. Dorogyy, O. Doroga-Ivaniuk // Journal of Automation, Mobile Robotics and Intelligent Systems. – 2018 – Vol. 12. – P. 5-13.

131. Committee of SGTm Neural-Like Structures with RBF Kernel for Insurance Cost Prediction Task / [I. Izonin, R. Tkachenko, N. Kryvinska, M. Gregus, P. Tkachenko, P. Vitynskyi] // 2nd Ukraine Conference on Electrical and Computer Engineering (UKRCON), Lviv, Ukraine, 2-6 July 2019 : proceedings. – Lviv : Lviv Polytechnic National University, 2019. – P. 1037-1040.

132. A Cartesian Genetic Programming Based Parallel Neuroevolutionary Model for Cloud Server's CPU Usage Prediction / [Q. Zia Ullah, G. M. Khan, S. Hassan, A. Iqbal] // Electronics. – 2021. – Vol. 10(1):67. – P. 18–36.

133. Asseman, A. Accelerating Deep Neuroevolution on Distributed FPGAs for Reinforcement Learning Problems / A. Asseman, N. Antoine, A. S. Ozcan // ACM Journal on Emerging Technologies in Computing Systems. – 2021. – Vol. 17 (2). – P. 1-17.

134. Achour, I.Z. Distributed Implementation of Neuroevolution of Augmenting Topologies Method / I.Z. Achour, A.Yu. Doroshenko // Problems in Programming. – Vol. 3. – 2021. – P. 3-15.

135. Hassanien, A.E. Swarm Intelligence / A. E. Hassanien, E. Emary. – Boca Raton: CRC Press, 2020. – 210 p.

136. Parallel Computing and Swarm Intelligence Based Artificial Intelligence Model for Multi-Step-Ahead Hydrological Time Series Prediction /

[W. Niua, Z. Fengb, B. Fenga, Y. Xua, Y. Mina] // Sustainable Cities and Society. – 2021. – Vol. 66. – P. 22–34.

137. Parallel Swarm Intelligence Strategies for Large-Scale Clustering Based on Mapreduce With Application to Epigenetics of Aging / [Z. Benmounah, S. Meshoul, M. Batouche, P. Lio] // Applied Soft Computing. – 2018. – Vol. 69. – P. 771-783.

138. Morris, G. SWARM: Artificial Intelligence Decodes End Time Prophecies / G. Morris. – Cambridge : Bowker, 2020. – 444 p.

139. Usage of Swarm Intelligence Strategies During Projection of Parallel Neuroevolution Methods for Neuromodel Synthesis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko] // Technology Audit and Production Reserves. – 2020. – Vol. 5, Issue 2/55. – P. 12–17.

140. Kirk, D. Programming Massively Parallel Processors 3rd Edition. A Hands-on Approach / D. Kirk, Hwu Wen-mei. – Burlington: Morgan Kaufmann, 2016. – 576 p.

141. How to count Big Data: Probabilistic Data Structures and Algorithms [Electronic resource]. – Access mode: <https://www.kdnuggets.com/2019/08/count-big-data-probabilistic-data-structures-algorithms.html>.

142. Shikhman, V. Mathematical Foundations of Big Data Analytics / V. Shikhman, D. Müller. – Berlin : Springer, 2021. – 288 p.

143. Deitel, P. Intro to Python for Computer Science and Data Science: Learning to Program with AI, Big Data and The Cloud / P. Deitel, H. Deitel. – London : Pearson, 2019. – 880 p.

144. Мальком, Ф. Що робити, коли машини почнуть робити все. Як роботи та штучний інтелект змінять життя та роботу / М. Фрэнк, П. Рериг, Б. Принг ; [пер. з англ. И. Савиной]. – Київ : Бомбора-Форс, 2019. – 315 с.

145. Coupled Electrophysiological, Hemodynamic, and Cerebrospinal Fluid Oscillations in Human Sleep / [N.E. Fultz, G. Bonmassar, K. Setsompop, R.A. Stickgold, B.R. Rosen, J.R. Polimeni, L.D. Lewis] // Science. – 2019. – Vol. 366, Issue 6465. – P.628-631.



146. Ayushi, S. Artificial Neural Networks: Applications in Management / S. Ayushi // IOSR Journal of Business and Management. – 2013. – Vol. 12. – P. 32-40.
147. Are We “Brain Washed” during Sleep? [Electronic resource]. – Access mode: <https://www.bu.edu/articles/2019/cerebrospinal-fluid-washing-in-brain-during-sleep>.
148. Senge, P. M. The Fifth Discipline: The Art & Practice of the Learning Organization / P. M. Senge. – New York: Doubleday, 2006. – 445 p.
149. Weaver, W. Science and Complexity / W. Weaver // American Scientist. – 1948. – Vol. 36, Issue 4. – P. 536-544.
150. Flood, R.L. Dealing with Complexity / R. L. Flood, E. R. Carson. – Berlin: Springer, 1993. – 296 p.
151. Implementation of the Indicator System in Modeling Complex Technical Systems / [S. Leoshchenko, S. Subbotin, A. Oliinyk, O. Narivs'kiy] // Radio Electronics, Computer Science, Control. – 2021. – № 1. – P. 117–126.
152. Sutton, R.S. Reinforcement Learning, Second Edition: An Introduction / R.S. Sutton, A.G. Barto. – Cambridge : Bradford Books, 2018. – 552 p.
153. Meyn, S. Control Systems and Reinforcement Learning / S. Meyn. – Cambridge : Cambridge University Press, 2022. – 450 p.
154. Yoon, C. Understanding Actor Critic Methods and A2C [Electronic resource]. – Access mode: <https://towardsdatascience.com/understanding-actor-critic-methods-931b97b6df3f>.
155. Zhang, H. Parallel Implementation Of Multilayered Neural Networks Based on Map-Reduce on Cloud Computing Clusters / H. Zhang, N. Xiao // Soft Computing. – 2016. – Vol. 20, Issue 4. – P. 1471-1483.
156. Томас, Е. Основи Big Data. Концепції, алгоритми та технології / Е. Томас, Х. Ваджид, Хаттак, Б. Пол ; [пер. з англ. А. Гладуна]. – Дніпро : Баланс Бізнес Букс, 2018. – 308 с.

157. Big Data Analysis with Python: Combine Spark and Python to unlock the powers of parallel computing and machine learning / [I. Marin, A. Shukla, S. VK]. – Birmingham : Packt Publishing, 2019. – 276 p.

158. Kale, V. Parallel Computing Architectures and APIs: IoT Big Data Stream Processing / V. Kale. – Boca Raton: Chapman and Hall/CRC, 2019. – 409 p.

159. Robey, R. Parallel and High Performance Computing / R. Robey, Y. Zamora. – New York: Manning, 2021. – 704 p.

160. Литвинов, О. Паралельні та роздіблені обчислення в комп'ютерних системах : навч. посіб. / О. Литвинов, В. Хандецький ; Дніпров. нац. ун-т ім. Олеся Гончара. – Дніпро : Ліра, 2021. – 172 с.

161. Герасимов, В.В. Розробка програмного забезпечення на платформі Java. Багатопоточне програмування і паралельні обчислення : навч. посіб. / В. В. Герасимов, Н. О. Матвеева ; Дніпров. нац. ун-т ім. Олеся Гончара. – Дніпро : Ліра, 2020. – 173 с.

162. Назарова, І.А. Паралельні обчислення : навч. посіб. / І. А. Назарова, О. А. Дмитрієва ; Держ. ВНЗ "Донецький національний технічний університет", Кафедра прикладної математики та інформатики. – Покровськ : ДВНЗ "ДонНТУ", 2020. – 245 с.

163. Методи математичного моделювання та ідентифікації складних процесів і систем на основі високопродуктивних обчислень (нейро- та нанопористі кіберфізичні системи із зворотніми зв'язками моделі з даними розрідженої структури, паралельні обчислення) : [монографія] / Хіміч О. М. [та ін.] ; НАН України, Інститут кібернетики ім. В. М. Глушкова. – Київ : Видавництво НАН України, 2019. – 175 с.

164. Optimal Brain Surgeon on Artificial Neural Networks in Nonlinear Structural Dynamics / [N. H. Christiansen, J. Hultmann Job, K. Klyver, J. Nøgsbrg] // 25th Nordic Seminar on Computational Mechanics (NSCM), Lund, Sweden, 25-26 October 2012 : proceedings. – Lund : Lund University, 2012. – P.319-324.

165. Endisch, C. Optimal Brain Surgeon for General Dynamic Neural Networks / C. Endisch, C. Hackl, D. Schröder // Progress in Artificial Intelligence. – 2007. – Vol. 4874. – P. 15-28.

166. Oppermann, A. Regularization in Deep Learning – L1, L2, and Dropout [Electronic resource]. – Access mode: <https://www.deeplearning-academy.com/p/ai-wiki-regularization>.

167. Classic Regularization Techniques in Neural Networks [Electronic resource]. – Access mode: <https://medium.com/@ODSC/classic-regularization-techniques-in-neural-networks-68bccee03764>.

168. Осауленко, В. М. Двовимірна модель навчання в спайкових нейронних мережах з гомеостазом та навчанням з підкріпленням / В. М. Осауленко // Системні дослідження та інформаційні технології. – 2017. – № 2. – С. 130-140.

169. Бочкаръов, О. Ю. Структурна адаптація процесів збирання даних в автономних розподілених системах із використанням методів навчання з підкріпленням / О. Ю. Бочкаръов // Computer systems and networks. – 2020. – Vol. 2, № 1. – С. 13-26.

170. Is Deep Reinforcement Learning Ready for Practical Applications in Healthcare? A Sensitivity Analysis of Duel-DDQN for Hemodynamic Management in Sepsis Patients / [M.Y. Lu, Z. Shahn, D. Sow, F. Doshi-Velez, Liwei H. Lehman] // AMIA Annual Symposium, Rockvill, USA, 14-18 November 2020 : proceedings. – Rockvill: AMIA 2020 – P. 773-782.

171. Continuous State-Space Models for Optimal Sepsis Treatment – a Deep Reinforcement Learning Approach / [A. Raghu, M. Komorowski, I. Ahmed, L. Celi, P. Szolovits, M. Ghassemi] // Journal of Machine Learning Research. – 2017. – Vol. 68. – P. 147-163.

172. Improving Sepsis Treatment Strategies by Combining Deep and Kernel-Based Reinforcement Learning / [X. Peng, Y. Ding, D. Wihl, O. Gottesman, M. Komorowski, L.-W. H. Lehman, A. Ross, A. Faisal, F. Doshi-

Velez] // Proceedings of the AMIA Annual Symposium, San Francisco, USA, 3-7 November 2018. – San Francisco: AMIA 2018 – P. 887-896.

173. Метод структурного доналаштування нейромережових моделей для забезпечення інтерпретабельності / [С. Д. Леоценко, А. О. Олійник, С. О. Субботін, Є. О. Гофман, О. В. Корнієнко] // Radio Electronics, Computer Science, Control. – 2021. – № 3. – P. 86–96.

174. The Message Passing Interface (MPI) standard [Electronic resource]. – Access mode: <https://www.mcs.anl.gov/research/projects/mpi/>

175. Neuromodeling of operational processes / [S.A. Subbotin, H.V. Pukhalska, S.D. Leoshchenko, A.O. Oliinyk, Ye.O. Gofman] // Radio Electronics, Computer Science, Control. – 2022. – № 1. – P. 120-129.

176. Neural Network Diagnostics of Aircraft Parts Based on the Results of Operational Processes / [S. Leoshchenko, H. Pukhalska, S. Subbotin, A. Oliinyk, Ye. Gofman] // Radio Electronics, Computer Science, Control. – 2022. – № 2. – P. 69-79.

177. Using Neuromodels for Evaluating and Determining Productivity of Technical Processes / [S. Leoshchenko, O. Nazarenko, A. Oliinyk, S. Subbotin, T. Zaiko, V. Donenko] // International Conference "Problems of Infocommunications. Science and Technology" (PIC S&T 2020), Kyiv, Ukraine, 6-9 October 2020 : proceedings. – Kharkiv : Kharkiv National University of Radio Electronics, 2020. – P.442-446.

178. Sequencing for Encoding in Neuroevolutionary Synthesis of Neural Network Models for Medical Diagnosis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko, S. Shylo, V. Lytvyn] // Proceedings of the 3rd International Conference on Informatics & Data-Driven Medicine (IDDM 2020), Växjö, Sweden, 19-21 October 2020. – P. 62-71. – (CEUR Workshop Proceedings, Vol. 2753).

179. Implementation of Reinforcement Learning Strategies in the Synthesis of Neuromodels to Solve Medical Diagnostics Tasks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Lytvyn, O. Korniienko] // Proceedings of the 4th

International Conference on Informatics & Data-Driven Medicine (IDDM-2021), Valencia, Spain, 19-21 November 2021. – P. 34-43. – (CEUR Workshop Proceedings, Vol. 3038).

180. M.-A. Kim, J. Seok Park, C. W. Lee, W.-I. Choi, Pneumonia Severity Index in Viral Community Acquired Pneumonia in Adults [Electronic resource]. – Access mode: <http://proceedings.mlr.press/v80/reagan18a/reagan18a.pdf> .

181. Breast Cancer Coimbra Data Set [Electronic resource]. – Access mode: <https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Coimbra>.

182. Smart Crossover Mechanism for Parallel Neuroevolution Method of Medical Diagnostic Models Synthesis / [S. Leoshchenko, S. Subbotin, A. Oliinyk, V. Lytvyn, M. Ilyashenko] // Proceedings of the Third International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020), Zaporizhzhia, Ukraine, April 27-May 1, 2020. – P. 57-69. – (CEUR Workshop Proceedings, Vol. 2608).

183. Implementation of Selective Pressure Mechanism to Optimize Memory Consumption in the Synthesis of Neuromodels for Medical Diagnostics / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko, N. Gorobii] // Proceedings of the 2nd International Workshop on Informatics and Data-Driven Medicine IDDM 2019, Lviv, Ukraine, 11–13 November 2019. – P. 109–120. – (CEUR Workshop Proceedings, Vol. 2488).

184. Physical Unclonable Functions Data Set [Electronic resource]. – Access mode: <https://archive.ics.uci.edu/ml/datasets/Physical+Unclonable+Functions>.

185. Adeleye, B. Analysis of Parallel Architectures: SIMD, tightly-coupled MIMD, and loosely-coupled MIMD / B. Adeleye, S. Jiddah // International Journal of Computer Trends and Technology. – 2017. – Vol. 53(1). – P. 6-8.

186. Differences Between SIMD and MIMD [Electronic resource]. – Access mode: <https://www.techwalla.com/articles/differences-between-simd-and-mimd>.

187. InfiniBand [Electronic resource]. – Access mode: <https://www.techtargget.com/searchstorage/definition/InfiniBand>.

188. The NVIDIA Quantum InfiniBand Platform [Electronic resource]. – Access mode: <https://www.nvidia.com/en-us/networking/products/infiniband>.
189. HIGGS Data Set [Electronic resource]. – Access mode: <https://archive.ics.uci.edu/ml/datasets/HIGGS>.
190. Synthesis of a Neural Network Model of Industrial Construction Processes Using an Indicator System / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Netrobko, Ye. Gofman] // Radio Electronics, Computer Science, Control. – 2021. – Vol. 4. – P. 69–77.
191. Crop Mapping Using Fused Optical-Radar Data Set [Electronic resource]. – Access mode: <https://archive.ics.uci.edu/ml/datasets/Crop+mapping+using+fused+optical-radar+data+set>.
192. Method of Artificial Neural Network Synthesis for Using in Integrated CAD / [S. Leoshchenko, A. Oliinyk, S. Subbotin, S. Shylo, V. Shkarupylo] // The experience of Designing and Application of CAD systems (CADSM): 15th International Conference CADSM'2019, Polyana (Svalyava), Ukraine, 26 February – 2 March 2019 : proceedings of the conference. – Lviv : Lviv Polytechnic National University, 2019. – P. 24-29.
193. Modification of the Genetic Method for Neuroevolution Synthesis of Neural Network Models For Medical Diagnosis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, N. Gorobii, V. Shkarupylo] // Proceedings of the Second International Workshop on Computer Modeling and Intelligent Systems (CMIS-2019), Zaporizhzhia, Ukraine, April 15-19, 2019. – P. 143-158. – (CEUR Workshop Proceedings, Vol. 2353).
194. Parallel Genetic Method for the Synthesis of Recurrent Neural Networks for Using in Medicine / [S. Leoshchenko, A. Oliinyk, S. Skrupsky, S. Subbotin, V. Lytvyn] // Proceedings of the Second International Workshop on Computer Modeling and Intelligent Systems (CMIS-2019), Zaporizhzhia, Ukraine, April 15-19, 2019. – P. 1-17. – (CEUR Workshop Proceedings, Vol. 2353).
195. Parallel Method of Neural Network Synthesis Based on a Modified Genetic Algorithm Application / [S. Leoshchenko, A. Oliinyk, S. Skrupsky, S.

Subbotin, T. Zaiko] // Proceedings of the Workshop on Modern Machine Learning Technologies and Data Science (MoMLLeT&DS-2019), Shatsk, Ukraine, 2-4 June 2019. – P. 11-23. – (CEUR Workshop Proceedings, Vol. 2386).

196. Implementation of swarm procedures for parallelization neuroevolution methods / [S. Leoshchenko, A. Oliinyk, S. Subbotin, Y. Gofman, V. Lytvyn] // Proceedings of the 7th International Conference on Control and Optimization with Industrial Applications (COIA 2020), Baku, Azerbaijan, 26-28 August 2020 : proceedings of the conference. – Baku : Baku State University, 2020. – P. 221-224.

197. Leoshchenko, S. Adaptive Mechanisms for Parallelization of the Genetic Method of Neural Network Synthesis / S. Leoshchenko, A. Oliinyk, S. Subbotin // Proceedings of the 10th International Conference on Advanced Computer Information Technologies (ACIT 2020), Deggendorf, Germany, 16-18 September 2020 : proceedings of the conference. – Ternopil : West Ukrainian National University, 2020. – P. 446-450.

198. Using the Actor-Critic Method for Population Diversity in Neuroevolutionary Synthesis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Shkarupylo] // Proceedings of the 2nd International Workshop on Intelligent Information Technologies and Systems of Information Security (IntelITSIS'2021), Khmelnytskyi, Ukraine, 24-26 March 2021. – P. 99–107. – (CEUR Workshop Proceedings, Vol. 2853).

199. Implementation of Probabilistic Data Structures in the Processes of Neuroevolutionary Synthesis / [S. Leoshchenko, S. Subbotin, A. Oliinyk, V. Lytvyn, O. Korniienko] // Proceedings of The Fourth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2021), Zaporizhzhia, Ukraine, April 27 2021. – P. 59-72. – (CEUR Workshop Proceedings, Vol. 2864).

200. Синтез та використання нейромережевих моделей з ймовірнісним кодуванням структури / [С. Д. Леощенко, А. О. Олійник, С. О. Субботін, Є. О. Гофман, М. Б. Ільяшенко] // Радіоелектроніка, інформатика, управління. – 2021. – № 2. – P. 93–104.

201. Mechanisms of Fine Tuning of Neuroevolutionary Synthesis of Artificial Neural Networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, M. Zaiko] // The Fourth IEEE International Conference on Advanced Information and Communication Technologies (AICT-2021), Lviv, Ukraine, 21-25 September 2021 : proceedings. – Lviv : Lviv Polytechnic National University, 2021. – P. 122-127.

202. Neuroevolutionary Mechanisms in the Synthesis of Spiking Neural Networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, M. Ilyashenko, A. Borovikov] // Proceedings of The Fifth International Workshop on Computer Modeling and Intelligent Systems CMIS-2022), Zaporizhzhia, Ukraine, 12 May 2022. – P. 88-97. – (CEUR Workshop Proceedings, Vol. 3137).



## ДОДАТОК А

Фрагменти вхідних даних та результати експериментальних досліджень

Таблиця А.1 – Фрагмент вхідних даних для задачі діагностування рівня родючості ґрунту для визначення схем зрошення [177]

Індекс	$x_1$	$x_2$	$x_3$	...	$y$
1	33,22	694,97	8760	...	318,24
2	32,92	688	8650	...	315
3	32,62	681,03	8580	...	311,76
4	32,32	674,06	8483,333	...	308,52
5	32,02	667,09	8393,333	...	305,28
6	31,72	660,12	8303,333	...	302,04
7	31,42	653,15	8213,333	...	298,8
8	31,12	646,18	8123,333	...	295,56
9	30,82	639,21	8033,333	...	292,32
10	30,52	632,24	7943,333	...	289,08
11	30,22	625,27	7853,333	...	285,84
12	29,92	618,3	7763,333	...	282,6
13	29,62	611,33	7673,333	...	279,36
14	29,32	604,36	7583,333	...	276,12
15	29,02	597,39	7493,333	...	272,88
16	28,72	590,42	7403,333	...	269,64
17	28,42	583,45	7313,333	...	266,4
18	28,12	576,48	7223,333	...	263,16
19	27,82	569,51	7133,333	...	259,92
20	27,52	562,54	7043,333	...	256,68
21	27,22	555,57	6953,333	...	253,44
22	26,92	548,6	6863,333	...	250,2
23	26,62	541,63	6773,333	...	246,96
24	26,32	534,66	6683,333	...	243,72
25	26,02	527,69	6593,333	...	240,48
26	25,72	520,72	6503,333	...	237,24
27	25,42	513,75	6413,333	...	234
28	25,12	506,78	6323,333	...	230,76
29	24,82	499,81	6233,333	...	227,52
30	24,52	492,84	6143,333	...	224,28
31	24,22	485,87	6053,333	...	221,04
...	...	...	...	...	...
1360	3,82	11,91	21	...	0,72

Таблиця А.2 – Фрагмент вхідних даних для задачі визначення індексу тяжкості для діагностування пневмонії [180]

Індекс	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	...	y
1	0	0	0	0	0	1	0	1	6,66	1	...	negative
2	0	0	0	0	0	1	0	1	6,85	1	...	influenza a
3	0	0	0	0	0	1	0	1	6,89	1	...	negative
4	118	0	21	0	100	0	35,5	0	6,89	1	...	negative
5	110	0	28	0	200	0	37,1	0	6,93	1	...	negative
6	0	0	0	0	0	1	0	1	6,99	1	...	negative
7	102	0	20	0	100	0	36	0	7,014	1	...	negative
8	120	0	22	0	140	0	36,2	0	7,031	1	...	negative
9	107	0	32	1	130	0	36,3	0	7,04	1	...	influenza b
10	0	0	0	0	0	1	35,7	0	7,04	1	...	negative
11	86	0	26	0	130	0	36,6	0	7,041	1	...	influenza a
12	91	0	30	1	180	0	35,7	0	7,06	1	...	negative
13	132	1	28	0	210	0	36	0	7,08	1	...	negative
14	51	0	36	1	100	0	37,6	0	7,08	1	...	rhinovirus
15	104	0	32	1	95	0	35,7	0	7,09	1	...	negative
16	75	0	20	0	135	0	36,8	0	7,09	1	...	rhinovirus
17	110	0	22	0	120	0	37,2	0	7,1	1	...	negative
18	145	1	35	1	180	0	35,2	0	7,1	1	...	negative
19	104	0	20	0	80	1	35	0	7,107	1	...	negative
20	82	0	18	0	120	0	37,2	0	7,109	1	...	negative
21	62	0	20	0	150	0	36	0	7,11	1	...	negative
22	133	1	33	1	110	0	36,8	0	7,11	1	...	negative
23	51	0	28	0	154	0	36,6	0	7,112	1	...	negative
24	145	1	18	0	118	0	38,9	0	7,116	1	...	negative
25	73	0	33	1	170	0	36,5	0	7,12	1	...	negative
26	120	0	28	0	200	0	36	0	7,12	1	...	negative
27	136	1	21	0	100	0	36,2	0	7,12	1	...	negative
28	72	0	20	0	110	0	36,6	0	7,13	1	...	negative
29	72	0	20	0	110	0	36,1	0	7,14	1	...	negative
30	78	0	20	0	120	0	36,4	0	7,15	1	...	rsv
31	96	0	24	0	100	0	36,6	0	7,15	1	...	negative
32	120	0	30	1	160	0	36,5	0	7,152	1	...	negative
33	100	0	22	0	130	0	36,9	0	7,164	1	...	negative
...	...	...	...	...	...	...	...	...	...	...	...	...
1435	96	0	0	0	110	0	36,6	0	7,68	0	...	negative

Таблиця А.3 – Результати виконання на CPU [184]

Використання CPU										
Розмір батьківського пулу	2					3				
	1	2	4	8	14	1	2	4	8	14
Кількість ядер	1,234	1,255	1,263	1,266	1,275	1,508	1,520	1,538	1,541	1,590
Помилка, %	1,234	1,255	1,263	1,266	1,275	1,508	1,520	1,538	1,541	1,590
Витрачений час, с	86132,3	46660,1	24587,9	13830,6	8053,9	155038,1	49535,3	27031,8	16281,8	10190,4
Прискорення	1,00	1,85	3,50	6,23	10,69	1,00	1,73	3,17	5,25	15,04
Накладні витрати	0,00	0,08	0,14	0,24	0,41	0,00	0,09	0,15	0,25	0,42

Продовження таблиці А.3

Використання CPU										
Розмір батьківського пулу	4					5				
	1	2	4	8	14	1	2	4	8	14
Кількість ядер	1,723	1,738	1,739	1,765	1,781	1,834	1,850	1,853	1,866	1,881
Помилка, %	1,723	1,738	1,739	1,765	1,781	1,834	1,850	1,853	1,866	1,881
Витрачений час, с	182600,4	50685,3	28009,4	17262,3	11045,0	242892,9	53201,1	30147,8	19407,1	12914,5
Прискорення	1,00	1,66	2,97	4,73	15,38	1,00	1,55	2,69	4,04	16,67
Накладні витрати	0,00	0,10	0,16	0,28	0,47	0,00	0,10	0,17	0,29	0,50



Продовження таблиці А.3

Використання CPU					
Розмір батьківського пулу	10				
Кількість ядер	1	2	4	8	14
Помилка	2,138	2,144	2,156	2,159	2,199
Витрачений час, с	484063,27	63264,17	38701,40	27986,37	20392,25
Прискорення	1,00	1,23	1,93	2,54	18,97
Накладні витрати	0,00	0,11	0,19	0,32	0,55

Таблиця А.4 – Результати виконання на GPU [184]

Використання GPU										
Розмір батьківського пулу	2					3				
	60	120	240	480	960	60	120	240	480	960
Кількість ядер	60	120	240	480	960	60	120	240	480	960
Помилка, %	1,245	1,254	1,268	1,292	1,296	1,527	1,537	1,544	1,580	1,593
Витрачений час, с	156176,586	88692,15169	54282,20303	37175,0845	29925,94302	191750,14	108559,54	66488,02	45954,96	37912,84
Прискорення	0,55	0,97	1,59	2,32	2,88	0,45	0,79	1,30	1,87	2,27
Накладні витрати	0,12	0,14	0,22	0,37	0,61	0,11	0,13	0,22	0,38	0,65

Продовження таблиці А.4

Використання GPU										
Розмір батьківського пулу	4					5				
Кількість ядер	60	120	240	480	960	60	120	240	480	960
Помилка, %	1,713	1,748	1,754	1,788	1,800	1,821	1,828	1,851	1,884	1,887
Витрачений час, с	219514,87	129513,77	80441,95	56782,55	48265,17	225588,40	142390,35	97348,50	73706,72	63387,78
Прискорення	0,39	0,67	1,07	1,52	1,78	0,38	0,60	0,88	1,17	1,36
Накладні витрати	0,11	0,18	0,24	0,41	0,70	0,19	0,26	0,37	0,51	0,72

Продовження таблиці А.4

Використання GPU										
Розмір батьківського пулу	6					7				
Кількість ядер	60	120	240	480	960	60	120	240	480	960
Помилка, %	1,594	1,636	1,637	1,641	1,673	1,238	1,259	1,270	1,285	1,299
Витрачений час, с	295433,64	55393,40	32011,25	21276,15	14543,56	276484,54	54602,73	31339,18	20602,06	12914,47
Прискорення	1,00	1,49	2,53	3,69	18,09	1,00	1,51	2,59	3,82	19,22
Накладні витрати	0,00	0,10	0,17	0,29	0,49	0,00	0,10	0,17	0,29	0,49

Продовження таблиці А.4

Використання GPU										
Розмір батьківського пулу	8					9				
	Кількість ядер	60	120	240	480	960	60	120	240	480
Помилка, %	1,707	1,716	1,747	1,764	1,792	1,817	1,870	1,882	1,897	1,898
Витрачений час, с	268970,79	177635,91	127461,01	101329,62	91196,66	282853,15	187529,47	135334,30	108466,46	98704,48
Прискорення	0,32	0,48	0,68	0,85	0,94	0,30	0,46	0,64	0,79	0,87
Накладні витрати	0,24	0,32	0,44	0,59	0,80	0,24	0,33	0,44	0,60	0,82

Продовження таблиці А.4

Використання GPU					
Розмір батьківського пулу	10				
	Кількість ядер	60	120	240	480
Помилка, %	2,000	2,055	2,065	2,166	2,181
Витрачений час, с	295000,22	203895,23	152619,81	125617,84	115568,41
Прискорення	0,29	0,42	0,56	0,69	0,75
Накладні витрати	0,29	0,38	0,50	0,65	0,84

**ДОДАТОК Б**

Акти впровадження дисертаційної роботи



**ЗАТВЕРДЖУЮ**

Керівник

ТОВ «АПТЕКА МАГНОЛІЯ»

посада керівника, назва організації (підприємства)



Сергій ШАТІЛОВ  
ПІБ

Сергій 2022р.

**АКТ**

впровадження результатів дисертації

Леощенка Сергія Дмитровича

Ми, що нижче підписалися, комісія у складі: керівника ТОВ «АПТЕКА МАГНОЛІЯ» Шатілова С.В., головного бухгалтера Помазан В.А., начальника юридичного відділу Дарінської Ю.М., начальника відділу автоматизованої системи керування виробництвом Гончаренко А.В., склали цей акт про те, що практичні результати, отримані аспірантом кафедри програмних засобів Національного університету «Запорізька політехніка» Леощенком Сергієм Дмитровичем під час виконання ним дисертаційної роботи, впроваджено у виробництво Товариства з обмеженою відповідальністю «АПТЕКА МАГНОЛІЯ», де використовуються для автоматизації побудови діагностичних моделей в інтелектуальних системах діагностування.

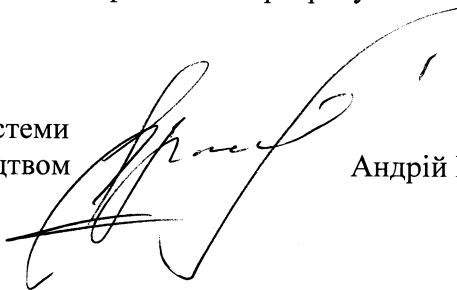
Леощенко С.Д. розробив методи синтезу діагностичних рекурентних нейромоделей на основі паралельного комп'ютингу та обчислювального інтелекту, які забезпечують можливість синтезу безнадлишкових та зручних для сприйняття й аналізу діагностичних моделей на основі вибірок даних великого об'єму. Запропоновані Леощенко С.Д. методи синтезу нейромоделей дозволяють збільшити швидкість процесу побудови діагностичних моделей, забезпечуючи високі рівні точності, інтепретовності та узагальнення моделей. Розроблений метод синтезу рекурентних

діагностичних моделей на основі паралельних обчислень на відміну від послідовних методів, які є застосовуваними лише для вузького класу об'єктів діагностування з лінійною природою, дозволяє будувати діагностичні моделі слабоформалізованих об'єктів та процесів, передбачає паралельну реалізацію, що дозволяє в рази скоротити час синтезу моделей об'єктів діагностування.

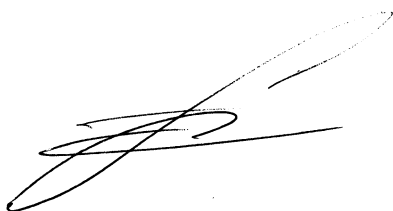
Використання розробленого Леошенко С.Д. математичного забезпечення при розв'язанні завдань діагностування та управління якістю продукції дозволяє зменшити складність синтезованих діагностичних моделей та підвищити їх рівні узагальнення й інтепретовності у порівнянні з аналітичними моделями та нейромоделями, побудованими шляхом безпосереднього відображення множини екземплярів у правила.

Цей акт не є підставою для фінансових розрахунків.

Начальник відділу  
автоматизованої системи  
керування виробництвом



Андрій ГОНЧАРЕНКО



**ЗАТВЕРДЖУЮ**

**В. о. генерального директора ТОВ**  
**«Запорізький ливарно-механічний завод»**



**І.В. Долгозвяго**  
під

бересня 2022 р.

**АКТ**

**впровадження результатів дисертації**

**Сергія Дмитровича Леоценка**

Цим підтверджуємо, що результати дисертаційної роботи аспіранта кафедри програмних засобів С. Д. Леоценка, виконаної у Національному університеті «Запорізька політехніка», впроваджено у виробництво Товариства з обмеженою відповідальністю «Запорізький ливарно-механічний завод», де використовуються для діагностування та контролю якості продукції.

Розроблено генетичний метод для синтезу рекурентних нейронних мереж на основі обчислювального інтелекту, який забезпечує можливість синтезу діагностичних моделей на базі рекурентних нейронних мереж з високим рівнем точності та інтерпретабельності для діагностування промислових виробів на основі історичних даних про їх виробництво і тестування. Запропоновано генетичний метод синтезу рекурентних нейромоделей дозволяє підвищити точність роботи діагностичних моделей при аналізі великих вибірок даних про промислові вироби. Розроблений метод синтезу рекурентних діагностичних нейромоделей на відміну від існуючих методів, які є застосовуваними лише для вузького класу об'єктів з лінійною природою та не промислового характеру, дозволяє будувати діагностичні моделі слабоформалізованих об'єктів та систем, що дозволяє розширити коло практичного використання таких діагностичних моделей.

Використання розробленого математичного забезпечення при розв'язанні завдань діагностування та управління якістю виробів дозволяє забезпечити високий рівень точності оцінювання та діагностування промислових виробів та забезпечує високі рівні узагальнення й інтепретовності аналітичних моделей в порівнянні з моделями, побудованими шляхом безпосереднього відображення множини екземплярів у правила.

Цей акт не є підставою для фінансових зобов'язань.

керівник проекту

Підписано електронно  
підписом: Анатолій  
Титаренко  
Дата: 2022.09.20  
16:18:05 +0200

О. М. Титаренко

Узгоджено:

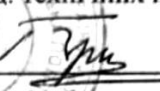
В.о. начальника механічного цеху

Діордієв  
Юрій  
Валерьевич


Підписано електронно  
підписом: Діордієв  
Юрій Валерьевич  
Дата: 2022.09.20  
16:21:57 +0200

Ю.В. Діордієв

ЗАТВЕРДЖУЮ  
 Ректор Національного університету  
 «Запорізька політехніка»,  
 канд. технічних наук, професор

 Віктор ГРЕШТА

20 " грудня 2022 р.



**АКТ**  
**впровадження результатів дисертації**

Ми, що нижче підписалися, комісія у складі проректора з наукової роботи Національного університету «Запорізька політехніка» (НУ «Запорізька політехніка»), доктора технічних наук, професора Вадима ШАЛОМЄЄВА, декана факультету комп'ютерних наук і технологій НУ «Запорізька політехніка», кандидата технічних наук, доцента Миколи КАСЬЯНА, доцента кафедри програмних засобів, кандидата технічних наук Тетяни КАПЛІЄНКО, склали цей акт про те, що результати, отримані аспірантом, асистентом кафедри програмних засобів НУ «Запорізька політехніка» Сергієм ЛЕОЩЕНКО під час виконання ним дисертації, впроваджено на кафедрі програмних засобів факультету Комп'ютерних наук і технологій НУ «Запорізька політехніка», де вони використовуються у навчальному процесі та науково-дослідній роботі.

Дисертаційна робота Сергія ЛЕОЩЕНКА виконувалася в межах трьох держбюджетних науково-дослідних робіт НУ «Запорізька політехніка»: «Розроблення та дослідження методів синтезу діагностичних моделей на основі методів обчислювального інтелекту» (номер державної реєстрації – 0119U100360), «Розроблення методів та засобів для аналізу та прогнозування динамічної поведінки нелінійних об'єктів» (номер державної реєстрації – 0121U107499), «Інтелектуальні методи та засоби діагностування та прогнозування стану складних об'єктів» (номер державної реєстрації – 0122U000972).

Сергій ЛЕОЩЕНКО розробив та дослідив нові методи синтезу рекурентних нейромережових моделей для діагностування. Ним розроблено метод синтезу рекурентних нейронних мереж на основі нейроеволюційного підходу, який використовує підхід більш компактного кодування інформації про нейромодель та критеріальний механізм вибору розвитку мережі, що забезпечує синтез точніших рішень. Розроблено паралельний метод синтезу нейромоделей на основі рекурентних нейронних мереж, що використовує метод селективного тиску для зменшення ресурсоемності та пришвидшення процесу синтезу нейромоделей. Розроблено метод структурної оптимізації нейромоделей, що використовуючи нейроеволюційні механізми, забезпечує структурне доналаштування попередньо синтезованих нейронних мереж для підвищення їх інтерпретабельності. Удосконалено індикаторну систему для визначення рівня складності задачі, що вирішується, з метою механізмів точного вибору механізму доналаштування штучних нейронних мереж.

Запропоновані Сергієм ЛЕОЩЕНКОМ методи синтезу рекурентних нейромережових моделей для діагностування дозволяють підвищити точність синтезованих моделей та забезпечують високий рівень їх інтерпретовності, що розширяє коло задач діагностування на основі таких моделей.

Результати наукових досліджень, викладені у дисертаційній роботі, використані при підготовці дисциплін «Інтелектуальні системи», «Основи обчислювального інтелекту», «Інтелектуальний аналіз даних» на кафедрі програмних засобів Національного університету «Запорізька політехніка».

Проректор з наукової роботи  
НУ «Запорізька політехніка»,  
доктор технічних наук, професор



Вадим ШАЛОМСЄВ

Декан факультету комп'ютерних наук і технологій  
НУ «Запорізька політехніка»,  
кандидат технічних наук, доцент



Микола КАСЬЯН

Доцент кафедри програмних засобів,  
кандидат технічних наук, доцент



Тетяна КАПЛІЄНКО

## ДОДАТОК В

### Список публікацій здобувача за темою дисертації

#### **Список праць, в яких опубліковані основні наукові результати дисертації:**

26. Modification and Parallelization of Genetic Algorithm for Synthesis of Artificial Neural Networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Lytvyn, V. Shkarupylo] // Radio Electronics, Computer Science, Control. – 2019. – № 4. – P. 68–82. (Web of Science)

27. Usage of Swarm Intelligence Strategies During Projection of Parallel Neuroevolution Methods for Neuromodel Synthesis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko] // Technology audit and production reserves. – 2020. – Vol. 5, Issue 2/55. – P. 12–17.

28. Implementation of the Indicator System in Modeling Complex Technical Systems / [S. Leoshchenko, S. Subbotin, A. Oliinyk, O. Narivs'kiy] // Radio Electronics, Computer Science, Control. – 2021. – № 1. – P. 117–126. (Web of Science)

29. Синтез та використання нейромережових моделей з ймовірнісним кодуванням структури / [С. Д. Леощенко, А. О. Олійник, С. О.Субботін, Є. О. Гофман, М. Б. Ільяшенко] // Radio Electronics, Computer Science, Control. – 2021. – № 2. – P. 93–104. (Web of Science)

30. . Метод структурного доналаштування нейромережових моделей для забезпечення інтерпретабельності / [С. Д. Леощенко, А. О. Олійник, С. О.Субботін, Є. О. Гофман, О. В. Корнієнко] // Radio Electronics, Computer Science, Control. – 2021. – № 3. – P. 86–96. (Web of Science)

31. Synthesis of a Neural Network Model of Industrial Construction Processes Using an Indicator System / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Netrebko, Ye. Gofman] // Radio Electronics, Computer Science, Control. – 2021. – № 4. – P. 69–77. (Web of Science)

32. Neuromodeling of Operational Processes / [S.A. Subbotin, H.V. Pukhalska, S.D. Leoshchenko, A.O. Oliinyk, Ye. O. Gofman] // Radio Electronics, Computer Science, Control. – 2022. – № 1. – P. 120-129. (Web of Science)

33. Neural Network Diagnostics of Aircraft Parts Based on the Results of Operational Processes / [S. Leoshchenko, H. Pukhalska, S. Subbotin, A. Oliinyk, Ye. Gofman] // Radio Electronics, Computer Science, Control. – 2022. – № 2. – P. 69-79. (Web of Science)

**Список праць, які засвідчують апробацію матеріалів дисертації:**

34. Method of Artificial Neural Network Synthesis for Using in Integrated CAD / [S. Leoshchenko, A. Oliinyk, S. Subbotin, S. Shylo, V. Shkarupylo] // The experience of Designing and Application of CAD systems (CADSM): 15th International Conference CADSM'2019, Polyana (Svalyava), 26 February – 2 March 2019 : proceedings. – Lviv : Lviv Polytechnic National University, 2019. – P. 24-29. (Scopus)

35. Modification of the Genetic Method for Neuroevolution Synthesis of Neural Network Models For Medical Diagnosis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, N. Gorobii, V. Shkarupylo] // Proceedings of the Second International Workshop on Computer Modeling and Intelligent Systems (CMIS-2019), Zaporizhzhia, Ukraine, April 15-19, 2019. – P. 143-158. – (CEUR Workshop Proceedings, Vol. 2353). (Scopus)

36. Parallel Genetic Method for the Synthesis of Recurrent Neural Networks for Using in Medicine / [S. Leoshchenko, A. Oliinyk, S. Skrupsky, S. Subbotin, V. Lytvyn] // Proceedings of the Second International Workshop on Computer Modeling and Intelligent Systems (CMIS-2019), Zaporizhzhia, Ukraine, April 15-19, 2019. – P. 1-17. – (CEUR Workshop Proceedings, Vol. 2353). (Scopus)

37. Parallel Method of Neural Network Synthesis Based on a Modified Genetic Algorithm Application / [S. Leoshchenko, A. Oliinyk, S. Skrupsky, S. Subbotin, T. Zaiko] // Proceedings of the Workshop on Modern Machine Learning Technologies and Data Science (MoMLeT&DS-2019), Shatsk, Ukraine, 2-4 June 2019. – P. 11-23. – (CEUR Workshop Proceedings, Vol. 2386). (Scopus)



38. Implementation of Selective Pressure Mechanism to Optimize Memory Consumption in the Synthesis of Neuromodels for Medical Diagnostics / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko, N. Gorobii] // Proceedings of the 2nd International Workshop on Informatics and Data-Driven Medicine IDDM 2019, Lviv, Ukraine, 11–13 November 2019. – P. 109–120. – (CEUR Workshop Proceedings, Vol. 2488). (Scopus)

39. Smart Crossover Mechanism for Parallel Neuroevolution Method of Medical Diagnostic Models Synthesis / [S. Leoshchenko, S. Subbotin, A. Oliinyk, V. Lytvyn, M. Ilyashenko] // Proceedings of the Third International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020), Zaporizhzhia, Ukraine, April 27-May 1, 2020. – P. 57-69. – (CEUR Workshop Proceedings, Vol. 2608) (Scopus)

40. Using neuromodels for evaluating and determining productivity of technical processes / [S. Leoshchenko, O. Nazarenko, A. Oliinyk, S. Subbotin, T. Zaiko, V. Donenko] // Proceedings of the International Conference "Problems of Infocommunications. Science and Technology" (PIC S&T 2020), Kyiv, Ukraine, 6-9 October 2020 : proceedings of the conference. – Kharkiv : Kharkiv National University of Radio Electronics, 2020. - P.442-446 (Scopus)

41. Implementation of swarm procedures for parallelization neuroevolution methods / S. Leoshchenko, A. Oliinyk, S. Subbotin, Y. Gofman, V. Lytvyn // Proceedings of the 7th International Conference on Control and Optimization with Industrial Applications (COIA 2020), Baku, Azerbaijan, 26-28 August 2020 : proceedings of the conference. – Baku : Baku State University, 2020. – P. 221-224. (Web of Science)

42. Leoshchenko, S. Adaptive Mechanisms for Parallelization of the Genetic Method of Neural Network Synthesis / S. Leoshchenko, A. Oliinyk, S. Subbotin // Proceedings of the 10th International Conference on Advanced Computer Information Technologies (ACIT 2020), Deggendorf, Germany, 16-18 September 2020 : proceedings of the conference. – Ternopil : West Ukrainian National University, 2020. – P. 446-450. (Scopus)

43. Sequencing for Encoding in Neuroevolutionary Synthesis of Neural Network Models for Medical Diagnosis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko, S. Shylo, V. Lytvyn] // Proceedings of the 3rd International Conference on Informatics & Data-Driven Medicine (IDDM 2020), Växjö, Sweden, 19-21 October 2020. – P. 62-71. – (CEUR Workshop Proceedings, Vol. 2753) (Scopus)

44. Using the Actor-Critic method for population diversity in neuroevolutionary synthesis / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Shkarupylo] // Proceedings of the 2nd International Workshop on Intelligent Information Technologies and Systems of Information Security (IntelITSIS'2021), Khmelnytskyi, Ukraine, 24-26 March 2021. – P. 99–107. – (CEUR Workshop Proceedings, Vol. 2853). (Scopus)

45. Implementation of Probabilistic Data Structures in the Processes of Neuroevolutionary Synthesis / [S. Leoshchenko, S. Subbotin, A. Oliinyk, V. Lytvyn, O. Korniienko] // Proceedings of The Fourth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2021), Zaporizhzhia, Ukraine, April 27 2021. – P. 59-72. – (CEUR Workshop Proceedings, Vol. 2864).

46. Implementation of Reinforcement Learning Strategies in the Synthesis of Neuromodels to Solve Medical Diagnostics Tasks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, V. Lytvyn, O. Korniienko] // Proceedings of the 4th International Conference on Informatics & Data-Driven Medicine (IDDM-2021), Valencia, Spain, 19-21 November 2021. – P. 34-43. – (CEUR Workshop Proceedings, Vol. 3038) (Scopus)

47. Mechanisms of fine tuning of neuroevolutionary synthesis of artificial neural networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, M. Zaiko] // Proceedings of The Fourth IEEE International Conference on Advanced Information and Communication Technologies (AICT-2021), Lviv, Ukraine, September 21-25 2021. – P. 122-127. (Scopus)

48. Neuroevolutionary Mechanisms in the Synthesis of Spiking Neural Networks / [S. Leoshchenko, A. Oliinyk, S. Subbotin, M. Ilyashenko, A. Borovikov] // Proceedings of The Fifth International Workshop on Computer Modeling and Intelligent Systems (CMIS-2022), Zaporizhzhia, Ukraine, May 12 2022. – P. 88-97. – (CEUR Workshop Proceedings, Vol. 3137).

**Список праць, які додатково відображають наукові результати дисертації:**

49. Методи та засоби оброблення великих даних в системах діагностування та розпізнавання образів : монографія / [С. О. Субботін, А. О. Олійник, В. М. Льовкін, Т. О. Колпакова, М. Ю. Пришляк, С. Д. Леоценко, О. В. Корнієнко, Д. А. Каврін, Є. О. Гофман, О. Ю. Благодарьов] ; під заг. ред. С. О. Субботіна, А. О. Олійника. – Запоріжжя : ЗНТУ, 2018. – 228 с.

50. Using recurrent neural networks for data-centric business / [S. Leoshchenko, A. Oliinyk, S. Subbotin, T. Zaiko] // Data-Centric Business and Applications - Evolvments in Business Information Processing and Management. – Cham, Switzerland: Springer, 2020. – P. 73-91. – (Lecture Notes on Data Engineering and Communications Technologies, Vol. 3)