

Міністерство освіти та науки України

Харківський національний автомобільно-дорожній університет

Коваль О. А., Коваль А. О.

**НЕЙРОННІ МЕРЕЖІ В ІНТЕЛЕКТУАЛЬНИХ
ВИМІРЮВАЛЬНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ**

Методичні вказівки до виконання лабораторних
та курсових робіт

Затверджено методичною радою
факультету, протокол № 1 від 7
«вересня» 2018 р.

Харків
2018

Укладачі: к. т. н. доцент Коваль О. А.
к.т.н. Коваль А. О.

Кафедра метрології та безпеки життєдіяльності

Лабораторний практикум та курсові роботи є важливим елементом навчального процесу, служать для закріплення лекційного матеріалу та придбання практичних навичок роботи з вимірною технікою та аналізу вимірних даних. В процесі лабораторної роботи студент перевіряє і закріплює свої теоретичні знання на практиці.

Виконання лабораторного завдання містить в собі такі етапи: позааудиторну самостійну підготовку, здачу допуску до роботи, експеримент, оформлення звіту та захист роботи.

Самостійна робота студента передбачає ознайомлення з вказівками до лабораторної роботи, визначення її мети і задач, вивчення відповідних розділів рекомендованої літератури, ознайомлення з правилами експлуатації використовуваної вимірної техніки.

Допуск до занять проводиться за контрольними питаннями, наведеними в кожній роботі, після чого студент приступає до проведення експерименту. По закінченні роботи студент подає результати викладачу для перевірки та відмітки про виконання, наводить порядок на робочому місці та приступає до оформлення звіту, виконуючи необхідні розрахунки, графічні роботи тощо.

Звіт оформлюється індивідуально з дотриманням вимог нормативної документації. Оформлення звіту починається з титульної сторінки, на якій вказуються кафедра, назва роботи, прізвище та ініціали студента, котрий виконав звіт, група, в якій студент навчається, прізвище та ініціали викладача, який прийняв звіт. До звіту включають: мету роботи; лабораторне завдання; результати експериментальних досліджень; висновки з оцінкою проведеного експерименту та поясненням отриманих результатів.

Захист роботи проводиться в навчальний період після завершення експерименту та підготовки звіту. До наступної роботи не допускаються студенти, які не захистили попередню роботу.

Курсові роботи виконуються та оформляються у відповідності до Стандарту ХНАДУ.

ЗАГАЛЬНІ ВІДОМОСТІ ПРО НЕЙРОННІ МЕРЕЖІ В ІНТЕЛЕКТУАЛЬНИХ ВИМІРЮВАЛЬНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ

Штучна нейронна мережа (ШНМ) – це математична модель, що представляє собою систему з'єднаних і взаємодіючих між собою нейронів. ШНМ успішно використовується в наступних областях:

- розпізнавання символів тексту та інших об'єктів;
- розпізнавання мови;
- управління рухом транспортного засобу і т. д.;
- класифікація ситуацій;
- кластеризація (категоризація) – «класифікація без вчителя»;
- прогнозування;
- апроксимація;
- прийняття рішень.

Нейронні мережі не програмуються у звичному розумінні цього слова, вони навчаються. Можливість навчання – одна з головних переваг нейронних мереж перед традиційними алгоритмами. Технічно навчання полягає в знаходженні коефіцієнтів зв'язків між нейронами.

Розробникам рішення на основі нейронної мережі потрібно:

1. Вибрати відповідну модель мережі, визначити топологію мережі (число елементів і їх зв'язку).
2. Вказати параметри навчання.

На першому етапі слід вибрати наступне:

- які нейрони ми хочемо використовувати (число входів, передатні функції);
- яким чином слід з'єднати їх між собою;
- що взяти в якості входів і виходів нейронної мережі.

Найбільш популярні і вивчені архітектури – це багатошаровий перцептрон, нейронна мережа із загальною регресією, нейронні мережі Кохонена, мережа Ворда, мережа Хопфільда та інші.

На другому етапі нам слід "навчити" обрану нейронну мережу, тобто підібрати такі значення її вагових коефіцієнтів, щоб вона працювала потрібним чином. Ненавчена нейронна мережа подібна дитині – її можна навчити чому завгодно. У використовуваних на практиці нейронних мережах кількість ваг може становити кілька десятків тисяч, тому навчання – дійсно складний процес. Для

багатьох архітектур розроблені спеціальні алгоритми навчання, які дозволяють настроїти ваги нейронної мережі певним чином. Найбільш популярний з цих алгоритмів – метод зворотного поширення помилки (Error Back Propagation), використовуваний, наприклад, для навчання перцептрона.

НЕЙРОМЕРЕЖЕВА СИСТЕМА ІДЕНТИФІКАЦІЇ СТАЦІОНАРНИХ ВИМІРЮВАНИХ СИГНАЛІВ

Основний підхід для вирішення важко формалізованих завдань базується на застосуванні множини нейромережових експертів – сукупності нейронних мереж різної архітектури з механізмом об'єднання рішень (асоціативна машина).

Виходячи з цього розроблено загальна структурна схема нейромережової системи ідентифікації безпечного стану нерухомих об'єктів (рис. 1.).

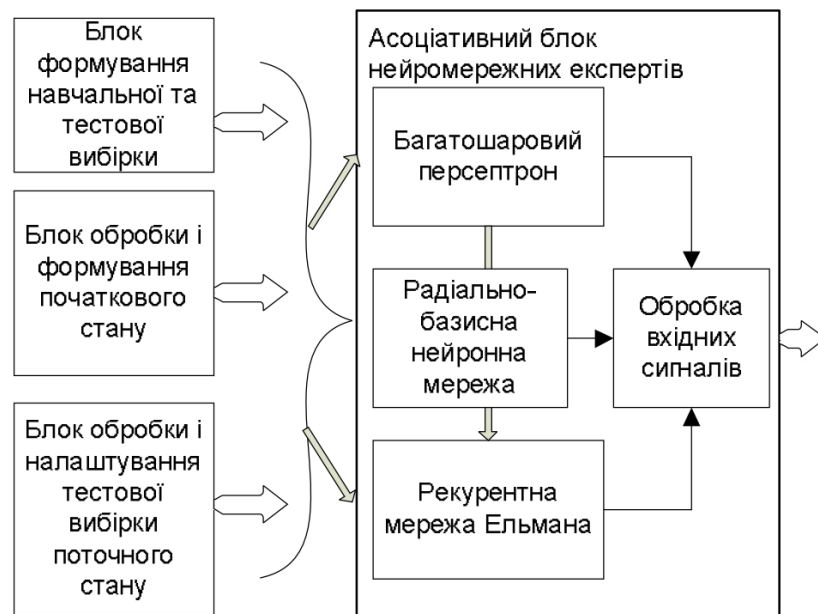


Рисунок 1 - Структурна схема нейромережової системи ідентифікації стаціонарних вимірюваних сигналів

Удосконалення алгоритмів навчання експертів. При вирішенні складних завдань може виникнути ситуація, коли спроби отримати прийнятне рішення навіть при використанні різних алгоритмів, що паралельно вирішують одну задачу, не дають результатів. У цьому випадку об'єднання декількох алгоритмів в композицію дозволяє

знайти рішення поставленого завдання. При вирішенні завдань за допомогою нейромережових методів, побудованих на застосуванні множини нейронних мереж ансамблів, вхідні дані обробляються за допомогою декількох нейронних мереж.

У статті багатошаровий перцептрон об'єднується в ансамбль нейронних мереж спільно з радіальнобазисною нейронною мережею і рекуррентною мережею Ельмана. Навчання кожної нейронної мережі проводилося з застосуванням градієнтних методів із застосуванням комбінованих евристик. Удосконалені алгоритми навчання багатошарового перцептрона, радіально базисної мережі і рекуррентної мережі Ельмана представлені на рис. 2 – 4 відповідно.

Перед виконанням процедури навчання всіх нейромережових експертів необхідно спроектувати їх архітектуру, тобто здійснити правильний вибір кількості шарів і елементів в кожному шарі. Кількість нейронів у вхідному і вихідному шарах всіх нейромережових експертів визначається умовами розв'язуваної задачі. Число прихованих шарів необхідно вибрати в залежності від того, наскільки складну залежність мережа повинна відтворити. Для експерта, який представлений багатошаровим перцептронем, ґрунтуючись на складності розв'язуваної задачі, було вибрано три приховані прошарки.

У радіально-базисної нейронної мережі, як і в стандартній моделі, був застосований лише один прихований шар, що складається з радіально-базисних нейронів. У рекуррентної мережі Ельмана був використаний тільки один шар прихованих нейронів. Після визначення кількості шарів необхідно правильно обрати кількість нейронів в прихованих шарах, число яких безпосередньо не визначається вихідними даними розв'язуваної задачі. Метод спрощення структури мережі в даній роботі застосовувати недоцільно, тому що мережі функціонують в складі комітету.

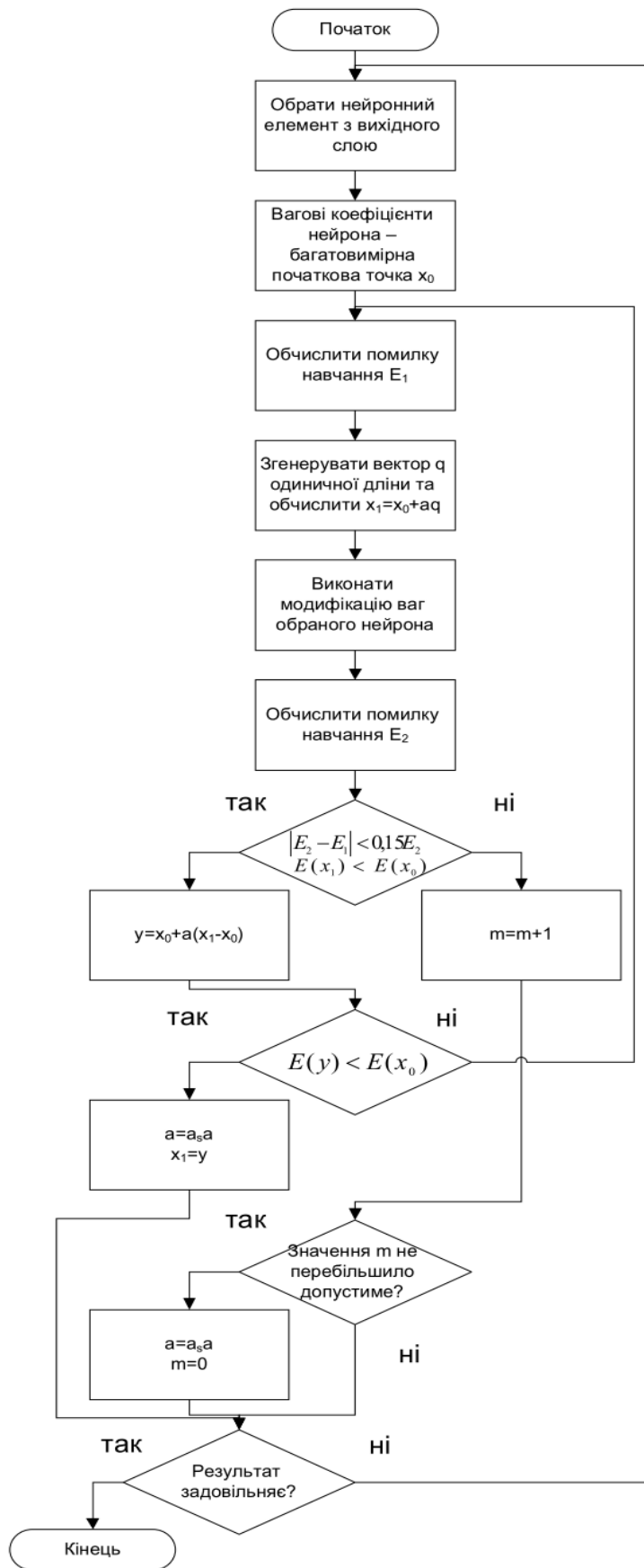


Рисунок 2 - Алгоритм навчання із застосуванням евристичної процедури для багатошарового перцептрона

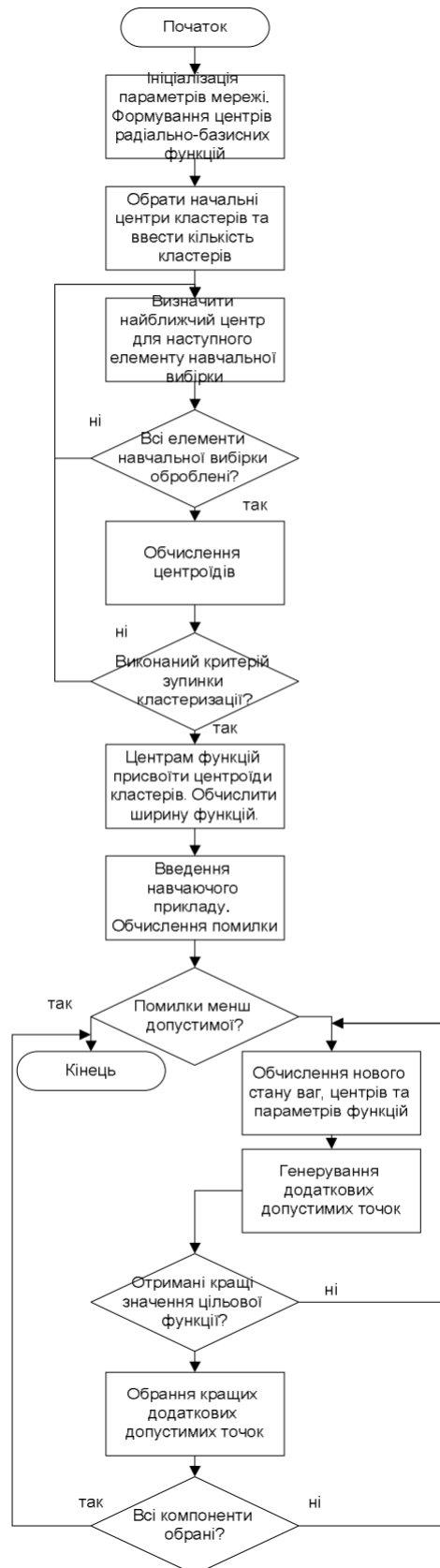


Рисунок 3 - Алгоритм навчання із застосуванням евристичної процедури для радіально-базисної функції

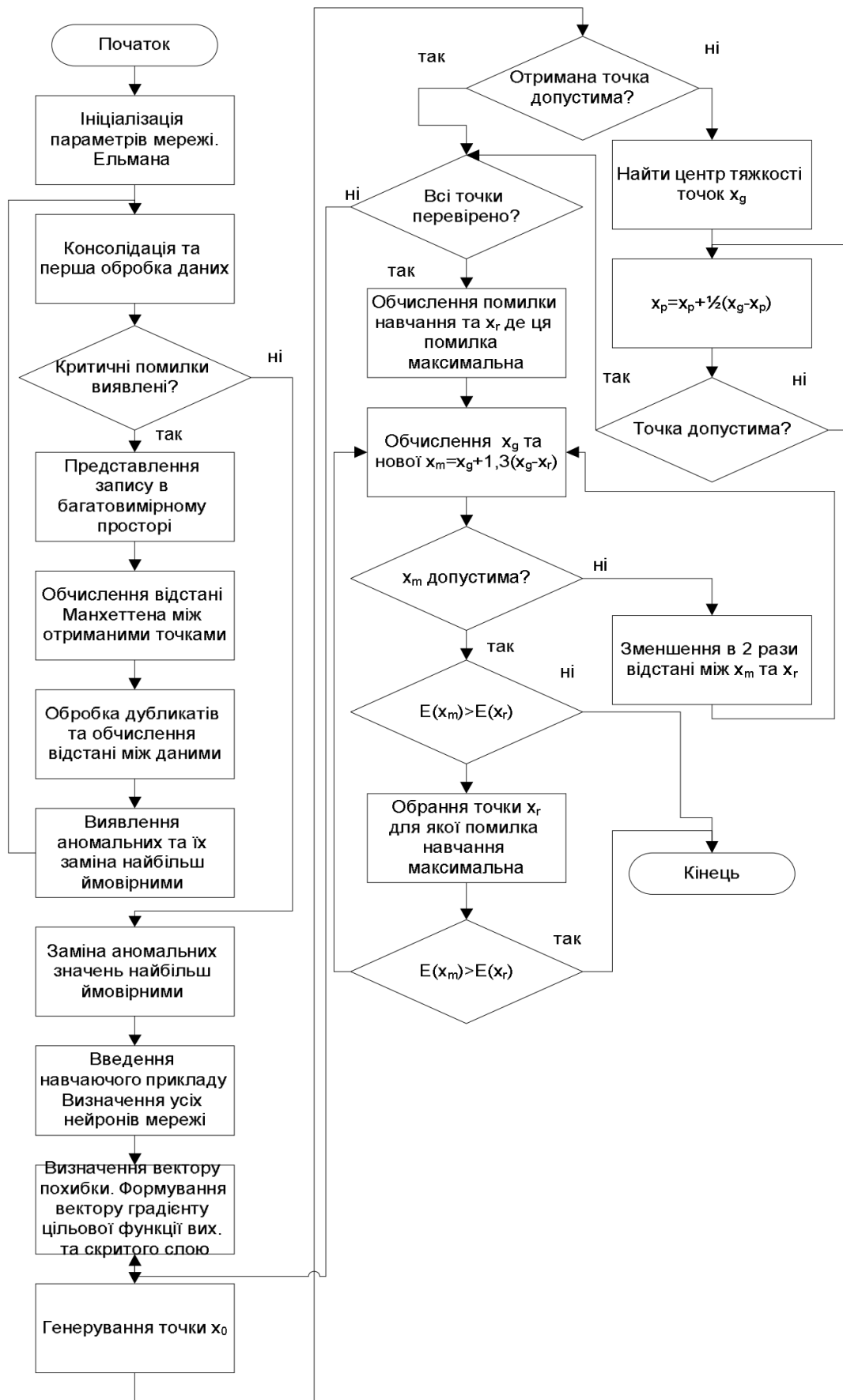


Рисунок 4 - Алгоритм навчання для нейронної мережі Ельмана

Ініціалізація обчислювальних нейронних структур, що володіють ресурсами, котрі значно перевищують потреби обчислювальної задачі, призводить до ускладнення експериментів.

Для вирішення даної проблеми, кожен нейромережевий експерт на початковому етапі роботи всієї асоціативної машини має мінімальну кількість нейронів в прихованих шарах (для багат шарового персептрона початкове число нейронів в прихованих шарах - 5, для радіально-базисної мережі визначалося процедурою кластеризації методом k-середніх, для нейронної мережі Ельмана – 4 нейрона).

Після попереднього визначення архітектури всіх мереж, виконувалось поступове додавання нейронів в приховані шари мережі до досягнення необхідної мінімально середньоквадратичної помилки навчання.

Визначивши топологію нейронних мереж, застосовують методи навчання. Алгоритми навчання мереж, засновані на градієнтному підході, застосовуються до тих пір, поки різниця помилок навчання для двох сусідніх ітерацій була більше деякого порога, якщо величина зміни помилки опускалася нижче заданого порогу, то застосовувалася евристика. Можливість використання евристики зменшувалася у міру розвитку процесу навчання мережі, щоб на початковому етапі навчання перешкоджатиме потраплянню в локальний мінімум і дати можливість більш детального налаштування параметрів мережі на завершальних етапах.

ДОСЛІДЖЕННЯ АЛГОРИТМІВ НАВЧАННЯ ВИМІРЮВАЛЬНИХ НЕЙРОННИХ МЕРЕЖ

Розглядаються три групи алгоритмів навчання нейронних мереж: алгоритми генетичної селекції, алгоритми градієнтного спуску, алгоритм зворотного розповсюдження помилки.

Алгоритм генетичної селекції

Генетичні алгоритми навчання штучних нейронних мереж базуються на теоретичних досягненнях синтетичної теорії еволюції, яка враховує мікробіологічні механізми спадкування ознак у природних і штучних популяціях організмів, а також на

накопиченому людському досвіді в селекції тварин і рослин. При цьому механізмами зміни визначених ознак виступають схрещування (гібридизація) та мутації.

Методологічна основа генетичних алгоритмів ґрунтується на гіпотезі селекції, яка в самому загальному вигляді може бути сформульована таким чином: чим вища пристосованість особини, тим вища імовірність того, що в потомстві, отриманому з її участю, ознаки, які визначають пристосованість, будуть виражені ще більше.

Оскільки генетичні алгоритми мають справу з популяціями постійної чисельності, тут особливої актуальності нарівні з селекцією у батьків набуває відбір на знищення. Найчастіше особини, які володіють низькою пристосованістю, не тільки не беруть участі у генерації нового покоління, а й вилючаються з популяції на черговому дискретному кроці еволюції.

Вхідні дані подаються у вигляді вектора. Координати векторів несуть змістовну «генетичну» інформацію. Подібно до того, як у природі схрещування організмів здійснюється на генетичному рівні, у процедурі оптимізації координати нових спробних точок отримуються як результат маніпулювання координатами старих. Відмінною рисою генетичного алгоритму від інших методів навчання штучних нейронних мереж є те, що він дозволяє відшукати глобальний мінімум похибки при налаштуванні ваг, хоча з малою точністю.

У загальному випадку генетичний алгоритм відрізняється від інших чисельних методів оптимізації тим, що запозичає з біології:

- понятійний апарат;
- ідею колективного пошуку екстремуму за допомогою популяції особин;
- способи подання генетичної інформації;
- способи передачі генетичної інформації в черзі поколінь (генетичні оператори);
- ідею про переважність розмноження найбільш пристосованих особин.

Вектор вхідних і вихідних ваг нейронної мережі $W = (W_{inp}, W_{mid}^j, W_{out})$, де W_{inp} - вектор вхідних ваг нейронної мережі; W_{mid}^j - вектор ваг між j -м і $(j+1)$ -м прихованими шарами нейронної мережі (для багатошарової нейронної мережі);

$j = \overline{1..k-1}$, де k – число прихованих шарів; W_{out} – вектор вихідних ваг нейронної мережі. Розмірність W дорівнює

$$(N+1) \cdot J_1 + J_1 \cdot J_2 + \dots + J_{k-1} \cdot J_k + J_k \cdot J \cdot M,$$

де N – число входів НМ;

J_j – число нейронів j -го прихованого шару;

M – число виходів НМ.

Навчання нейронної мережі за допомогою генетичного алгоритму – ітеративний процес відшукування такого набору ваг нейронної мережі за допомогою випадкової зміни компонентів, яка б задовольняла заданий критерій навчання.

За критерій навчання візьмемо суму квадратів відхилень, отриманих за моделлю виходів і бажаних виходах.

$$e(W) = \frac{1}{2} \cdot \|d - y(W)\|^2 = \frac{1}{2} \cdot \sum_{k=1}^M (d_k - y_k(W))^2.$$

Процес навчання припиняється при досягненні критерієм заданого порогу для будь-якого з шуканого набору ваг. На практиці точність налаштування ваг за допомогою генетичного алгоритму невисока й задана точність (порядку 0,001 чи вище) не досягається. Тому процес навчання припиняється після закінчення заданої кількості кроків еволюції. Кращий набір ваг на цей момент є шуканим.

У роботі величина навчальної послідовності визначається при побудові моделі вхідних даних для кожної прогнозованої точки. Генетичний алгоритм призначений для навчання нейронної мережі шляхом налаштування її вхідних і вихідних ваг. У результати навчання виходи нейронної мережі повинні збігатися з бажаними виходами (реальними даними) із заданим ступенем вірогідності (критерій навчання). Процедуру навчання ШНМ за допомогою ГА можна зобразити на блок-схемі наступним чином:

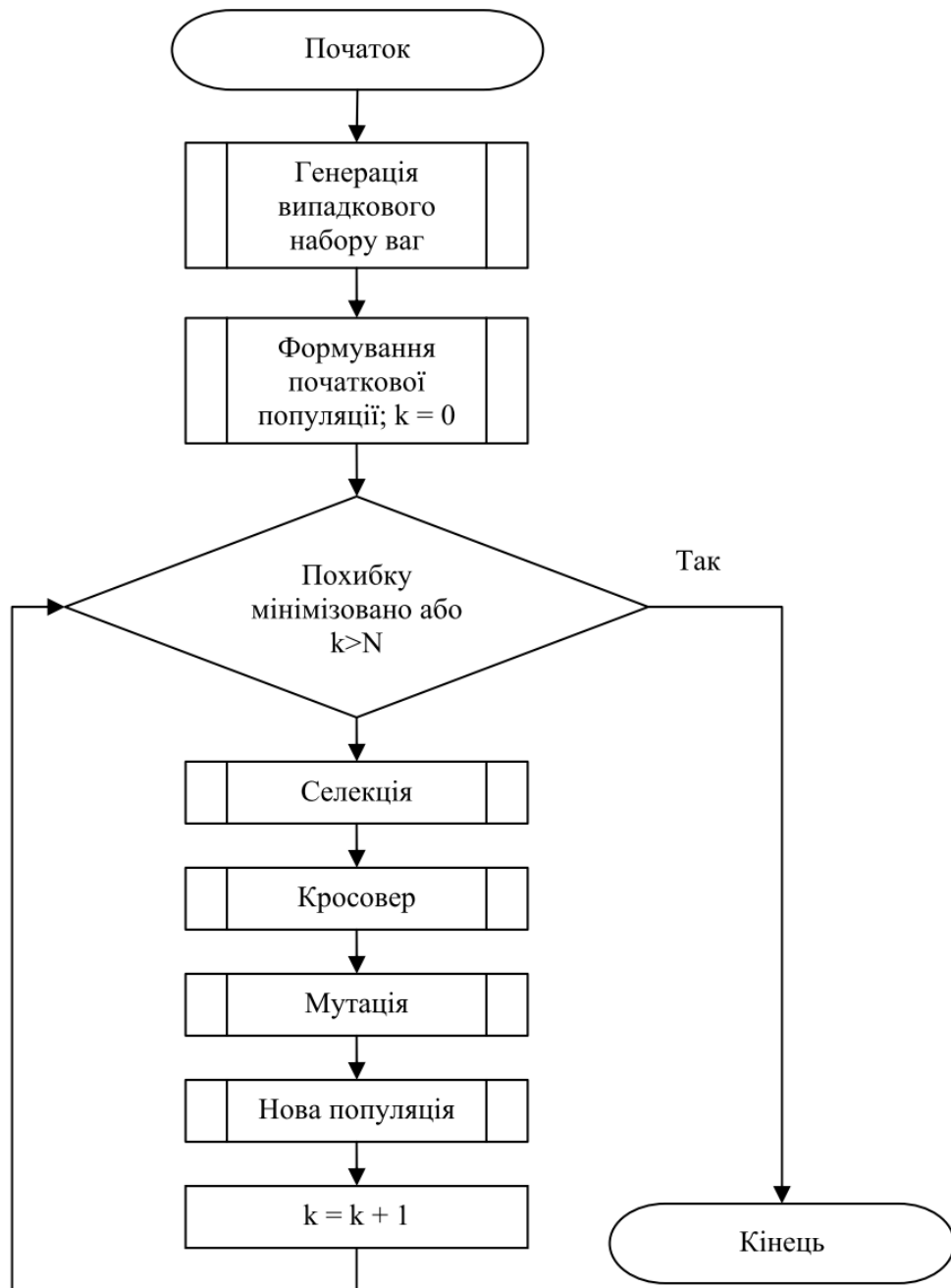


Рисунок 5- Блок-схема навчального алгоритму генетичної селекції

Алгоритми з використанням методів градієнтного спуску

Задачу навчання мережі у випадку застосування градієнтних алгоритмів розглядають як задачу мінімізації апріорно визначеної цільової функції $\xi(w)$. Градієнтні методи зв'язані з розкладанням цільової функції $\xi(w)$ в ряд Тейлора в найближчому okolí точки наявного рішення w . У випадку цільової функції від багатьох змінних $\left(w = [w_1, w_2, \dots, w_n]^T\right)$ таке представлення пов'язується з

околом раніше визначеної точки (при старті алгоритму – це вихідна точка w_0) в напрямку x . Подібне розкладання описується універсальною формулою виду:

$$\xi(w+x) = \xi(w) + [p(w)]^T \cdot x + \frac{1}{2} \cdot x^T \cdot H(w) \cdot x + \dots,$$

де $p(w) = \nabla \cdot \xi = \left[\frac{\delta \xi}{\delta w_1}, \frac{\delta \xi}{\delta w_2}, \dots, \frac{\delta \xi}{\delta w_n} \right]^T$ – це вектор градієнта, а

симетрична квадратна матриця

$$H(w) = \begin{bmatrix} \frac{\delta^2 \xi}{\delta w_1 \delta w_1} & \dots & \frac{\delta^2 \xi}{\delta w_1 \delta w_n} \\ \dots & \dots & \dots \\ \frac{\delta^2 \xi}{\delta w_{1n} \delta w_1} & \dots & \frac{\delta^2 \xi}{\delta w_n \delta w_{1n}} \end{bmatrix}$$

є матрицею похідних другого порядку і називається гессіаном.

X відіграє роль направляючого вектора, що залежить від фактичних значень вектора w . На практиці частіше розраховуються три перших члени ряду, а наступні просто ігноруються.

Для спрощення опису значення змінних, одержані в k -му циклі, будемо записувати з нижнім індексом k . Точкою рішення $w = w_k$ будемо вважати точку, в якій досягається мінімум цільової функції $\xi(w)$ та $p(w_k) = 0$, а гессіан $H(w_k)$ є невід'ємно визначеним. При виконанні цих умов функція в будь-якій точці, що лежить в околі w_k , буде мати більше значення, ніж у точці w_k , тому точка w_k є рішенням, що відповідає критерію мінімізації цільової функції.

У процесі пошуку мінімального значення цільової функції напрямки пошуку p та крок h підбираються таким чином, щоб для кожної наступної точки $w_{k+1} = w_k + \eta_k \cdot x_k$ виконувалася умова $\xi(w_{k+1}) < \xi(w_k)$. Пошук мінімуму виконується до тих пір, поки норма градієнта не опуститься нижче апріорі заданого значення допустимої похибки, або поки не буде перевищено максимальної кількості ітерацій. Універсальний оптимізаційний алгоритм навчання нейронної мережі можна представити в наступному

вигляді (будемо вважати, що початкове значення вектора, що оптимізується, відоме і складає $w_k = w_0$):

Крок 1: Перевірка зходжованості та оптимальності поточного значення w_k . Якщо точка w_k відповідає градієнтним умовам зупинки процесу – завершення обчислень. У протилежному випадку перейти до кроку 2.

Крок 2: Визначення вектора напрямку оптимізації x_k для точки w_k .

Крок 3: Вибір величини кроку η_k у напрямку x_k , при якому виконується умова $\xi(w_k + \eta_k \cdot x_k) < \xi(w_k)$.

Крок 4: Визначення нового рішення $w_{k+1} = w_k + \eta_k \cdot x_k$, а також відповідних йому значень $\xi(w)$ та $p(w_k)$, а якщо треба, то і $H(w_k)$, та повернення до кроку 1.

Алгоритм спряжених градієнтів

Цей алгоритм відрізняється від загального алгоритму градієнтного спуску тим, що при виборі напрямку мінімізації не використовується інформація про гессіан. Напрямок пошуку x_k вибирається таким чином, щоб він був ортогональним та спряженим до всіх попередніх напрямків x_0, x_1, \dots, x_{k-1} . Множина векторів $x_i, i = 0, 1, \dots, k$ буде взаємно спряженою стосовно матриці A коефіцієнтів зв'язку між i -м вузлом $(L-1)$ -го шару чи з k -м вузлом i -го шару, якщо

$$x_i^T \cdot A \cdot p_j = 0, \quad i \neq j. \quad (1)$$

Вектор, що задовольняє задані умови, має вигляд:

$$x_{k+1} = x_k + \alpha_k \cdot p_k, \quad (2)$$

де $p_k = p(w_k)$ є фактичним значенням вектору градієнта.

Із формули (2) впливає, що новий напрямок мінімізації залежить тільки від значення градієнта в точці рішення w_k та від попереднього напрямку пошуку p_k , помноженого на коефіцієнт спряження α_k .

Алгоритм зворотного поширення помилки.

В загальному вигляді алгоритм зворотного поширення помилки являє собою наступну послідовність кроків:

Крок 1: Ініціювати ваги малими випадковими величинами.

Крок 2: Якщо умова зупинки не виконується, виконати кроки 3-10.

Крок 3: Для кожної навчальної пари виконати кроки 4-9.

Прямий прохід:

Крок 4: Кожен вхідний нейрон $x_i = 1..n$ приймає вхідний сигнал і поширює його до всіх нейронів прихованого шару.

Крок 5: Кожен нейрон прихованого шару $v_j = 1..q$ підсумовує свої зважені вхідні сигнали:

$$h_j = \sum_{i=1}^n w_{ij} \cdot x_i, \quad (3)$$

застосовує до одержаної суми функцію активації, формуючи вихідний сигнал: $v_j = f(h_j)$, котрий надсилається до всіх нейронів вихідного шару.

Крок 6: Кожен вихідний нейрон $y_k, k = 1..m$ підсумовує зважені сигнали:

$$h_k = \sum_{j=1}^q w_{jk} \cdot x_j, \quad (4)$$

формуючи після застосування функції активації вихідний сигнал мережі: $y_k = f'(h_k)$.

Зворотне поширення помилки:

Крок 7: Кожен вихідний нейрон співставляє своє значення виходу з потрібною цільовою функцією і вираховує $\delta_k = (t_k - y_k) \cdot f'(h_k)$, після чого визначається корегуючий член ваг: $\Delta w_{jk} = \eta \cdot \delta_k \cdot v_j$, а параметри δ_k надсилаються в нейрони прихованого шару.

Крок 8: Кожен нейрон прихованого шару v_j підсумовує свої δ -входи від нейронів вихідного шару:

$$h_k = \sum_{k=1}^m \delta_k \cdot w_{jk}, \quad (5)$$

результат помножують на похідну від функції активації для визначення δ_j :

$$\delta_j = f'(h_j) \cdot \sum_{k=1}^m \delta_k \cdot w_{jk} \quad (6)$$

та вираховується корегуючий член:

$$\Delta w_{ij} = \eta \cdot \delta_k \cdot w_{jk} . \quad (7)$$

Корегування ваг:

Крок 9: Ваги між прихованим та вихідним шарами модифікуються так:

$$w_{jk} (new) = w_{jk} (old) + \Delta w_{jk} . \quad (8)$$

Аналогічно корегуються ваги між вхідним та прихованим шарами:

$$w_{ij} (new) = w_{ij} (old) + \Delta w_{ij} . \quad (9)$$

Крок 10: Перевіряється умова зупинки: мінімізація похибки між потрібним та реальним виходом мережі.

За допомогою блок-схеми поданий алгоритм можна зобразити наступним чином (рис. 3).

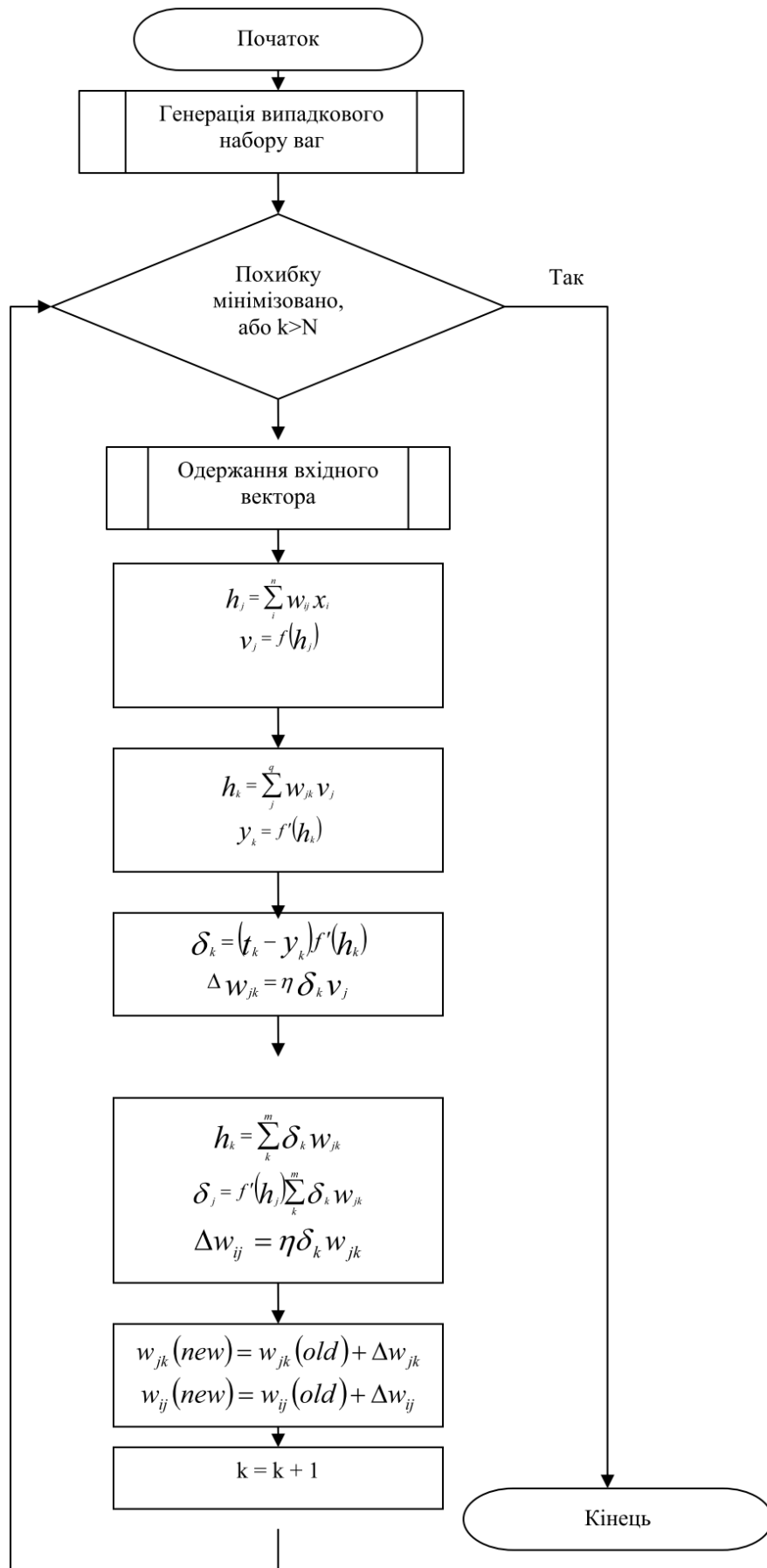


Рисунок 3 - Блок-схема алгоритму зворотного розповсюдження помилки

КРИТЕРІЙ ЗУПИНКИ НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ ПРИ ВИЗНАЧЕННІ ДИНАМІЧНИХ ХАРАКТЕРИСТИК ЗВТ

В процесі визначення ДХ ВКТ з використанням нейронної мережі виникає задача автоматичного визначення ступеню подібності вимірних сигналів, тобто визначення метрики (відстані між сигналами).

Найбільш простий клас метрик порівняння сигналів на виході ВКТ - це порівняння вимірних сигналів за їх формою для кожного моменту часу. Наприклад, можна порівнювати максимальне відхилення амплітуд сигналів, але така метрика чутлива до одиничних відмінностей в амплітудах сигналів

$$U = \max_i |a_i - b_i|, \quad (10)$$

де вектори a_i і b_i - значення амплітуд порівнюваних вхідних дій (сигналів) ВКТ.

Іншим критерієм оцінки може служити середньоквадратичне відхилення амплітуд сигналів [146]:

$$\sigma_u = \sqrt{\frac{\sum_{i=1}^n (a_i - b_i)^2}{n}}, \quad (11)$$

де n – кількість вимірів в часовій вибірці вхідного сигналу. Даному методу властиві недоліки:

— висока чутливість до середньої відмінності сигналів по амплітуді, що може привести до помилкового результату у випадку, коли сигнали незначно відрізняються в середньому по амплітуді;

— вимірювальний канал має різну чутливість до спотворень вхідного сигналу у різних частинах частотного діапазону, що пов'язано з амплітудно-частотною характеристикою вимірювального каналу тиску. З цього випливає, що спотворення порівнюваних сигналів на низьких (0.01-5 Гц) і середніх частотах (5-20 Гц) будуть більшими, ніж

на високих (20-50 Гц).

Таким чином, дана метрика не може враховувати різну чутливість ВКТ в різних частотних смугах, а при порівнянні двох різних сигналів з білим шумом швидше за все дасть висновок про те, що вони зовсім різні.

Іншим підходом є частотно-часова метрика вхідних сигналів ВКТ. Для одержання даної метрики вхідні сигнали спочатку послідовно покриваються невеликими інтервалами з деяким кроком Δt в часі і Δf по частоті. У кожному з цих інтервалів сигнал розкладається в ряд Фур'є, після чого будується спектр (без врахування фаз частотних складових). Отримані спектри записуються у двовимірний масив (час, частота) - спектрограму.

Перевагою даної метрики над описаною вище є те, що можна порівнювати значення амплітуд сигналів згідно з даними про сприйняття ВКТ тієї або іншої частотної складової, тобто робити порівняння за формулою

$$\sigma_u = \frac{\sqrt{\sum_{j=1}^{N_f} \frac{1}{\alpha_j} \cdot \sum_{i=1}^{N_t} (Sa_{ji} - Sb_{ji})^2}}{N_f \cdot N_t}, \quad (12)$$

де Sa_{ji} , Sb_{ji} – двовимірні масиви амплітуд спектрограм двох вхідних сигналів a і b ;

N_f , N_t – число елементів аналізу відповідно по частоті та часу.

Коефіцієнт α_j залежить від чутливості ВКТ в даній j -тій частотній ділянці амплітудно-частотної характеристики каналу тиску, значення для якої отримуються експериментально.

Для даного методу, як і для амплітудно-часової метрики, так само необхідно, щоб сигнали містили однакову енергію, тобто середньоквадратичне відхилення в спектрах для всього сигналу повинне бути мінімальним. У порівнянні зі звичайною метрикою в даній мірі практично вирішується проблема порівняння сигналів з різними амплітудами й ураховується нерівномірна чутливість ВКТ до різних частотних складових.

Для тестування цього методу визначення метрики вхідних сигналів ВКТ більш правильно застосовувати модифікацію даного

методу: з спектрограми послідовно виділяється кілька k частотних смуг і в них визначається середньоквадратичне відхилення. Частотні смуги вибираються неоднакової ширини, тому що змістовної інформації для технологічного процесу ВКТ в межах 0...5 Гц більше, ніж у межах 20...40 Гц, отже й смуги в низькочастотній області вибираються вужчими, ніж у високочастотній. Так ширина частотних смуг Δf у межах 0.01...40 Гц змінюється від $\Delta f_{\min} = 0.05$ Гц до $\Delta f_{\max} = 5$ Гц. Відповідно буде змінюватися і число елементів аналізу N_f для кожної k -ї частотної смуги

$N_f = \frac{f_{\max_k} - f_{\min_k}}{\Delta t_k}$. З урахуванням цього вираз (12) прийме вид:

$$\sigma_k = \frac{\sqrt{\sum_{j=1}^{N_f} \cdot \sum_{i=1}^{N_t} (S a_{ji} - S b_{ji})^2}}{N_f \cdot N_t}. \quad (13)$$

Перевагою даного методу є те, що стає можливим порівняння вхідних сигналів ВКТ, оброблених фільтром низьких частот для зменшення кількості вимірювальної інформації. Такі сигнали можна порівнювати тільки в тих частотних смугах, які характерні для даного сталого режиму роботи ТСО. Також у цьому випадку не обов'язково зберігати енергію сигналів у всьому частотному діапазоні при переході від одного сталого режиму ТСО до іншого. При коректному збереженні енергії сигналу в частотній смузі можна досить точно визначити середню відмінність сигналів в цій смузі. Одним з недоліків методу є мала розрізнявальна здатність як по частоті, так і за часом.

Для зменшення впливу цього недоліку на якість роботи нейронної мережі весь частотний діапазон вхідного сигналу ВКТ розбивається на декілька n смуг. В залежності від їх значимості для характеристики технологічного процесу та точності вимірювання тиску в каналі ТСО ширина кожної смуги dF_n різна. Аналогічним чином весь часовий інтервал аналізу подібності вхідних сигналів розбивається на m підінтервалів часу з різною шириною dF_m , в межах яких буде проводитись порівняння сигналів. Таким чином, можна отримати середнє відхилення між вхідними сигналами ВКТ

в області розміром $n \times m$:

$$\sigma_{nm} = \frac{\sum_{j=1}^n \cdot \sum_{i=1}^m |Sa_{ji} - Sb_{ji}|}{dF_n \cdot dT_m}. \quad (14)$$

З точки зору реалізації пошуку екстремуму зручніше оцінювати не мінімуми σ_{nm} , а максимуми $1/\sigma_{nm}$. Аналіз метрики відновленого сигналу нейромережевим алгоритмом та опорного вхідного сигналу за критерієм максимуму $1/\sigma_{nm}$ дає можливість визначити ДХ ВКТ у полі аналізу.

Вихідний сигнал ВКТ характеризується декількома частотними складовими. Для кожного сталого режиму роботи ТСО кількість цих складових різна. Характерним є наявність амплітудних і частотних флуктуацій як в межах одного, так і при переході до іншого сталого режиму ТСО. Ці флуктуації, на перший погляд, і не значні, але це приводить до того що метрика σ_{nm} порівнюваних вхідних дій (фактично їх енергія) буде розподілена в декількох елементах аналізу, що значно ускладнює пошук екстремуму $1/\sigma_{nm}$. Як наслідок, зростає час та похибка навчання нейронної мережі, в окремих випадках нейронна мережа взагалі не може навчитися. На рис. 4 зображено ізолінії відображення одної з реалізацій відносної метрики сигналів в часово-частотному полі аналізу, які формуються при навчанні нейронної мережі. Такий вид метрики аналізуємих вхідних дій є характерним при малому відношенню сигнал/шум ($q < 7$ дБ) або при тривалості квазістаціонарної часової вибірки менше 1 хвилини. Для усунення цього недоліку визначення метрики аналізуємих вхідних дій необхідно проводити в три етапи (рис. 4).

На першому етапі ковзаючим зрізом в площині "час-частота" проводиться грубий пошук екстремуму. При його знаходженні проводиться точне визначення елементів аналізу шляхом одночасного часово-частотного аналізу в часовому та частотному стробах — другий етап. Спочатку одночасно оцінюються всі максимуми, які попали в строб часу, і якщо знайдений за результатами грубої оцінки є дійсно найбільшим максимумом, то далі проводиться його уточнення частотним стробом. Так як вихідний сигнал ВКТ має флуктуації як по амплітуді так і по частоті,

то необхідно проводити уточнення місцезнаходження максимуму відносної метрики в межах двох сусідніх елементів поля "час-частота" — третій етап. Таким підходом усувається неоднозначність оцінки критерію зупинки навчання нейронної мережі.

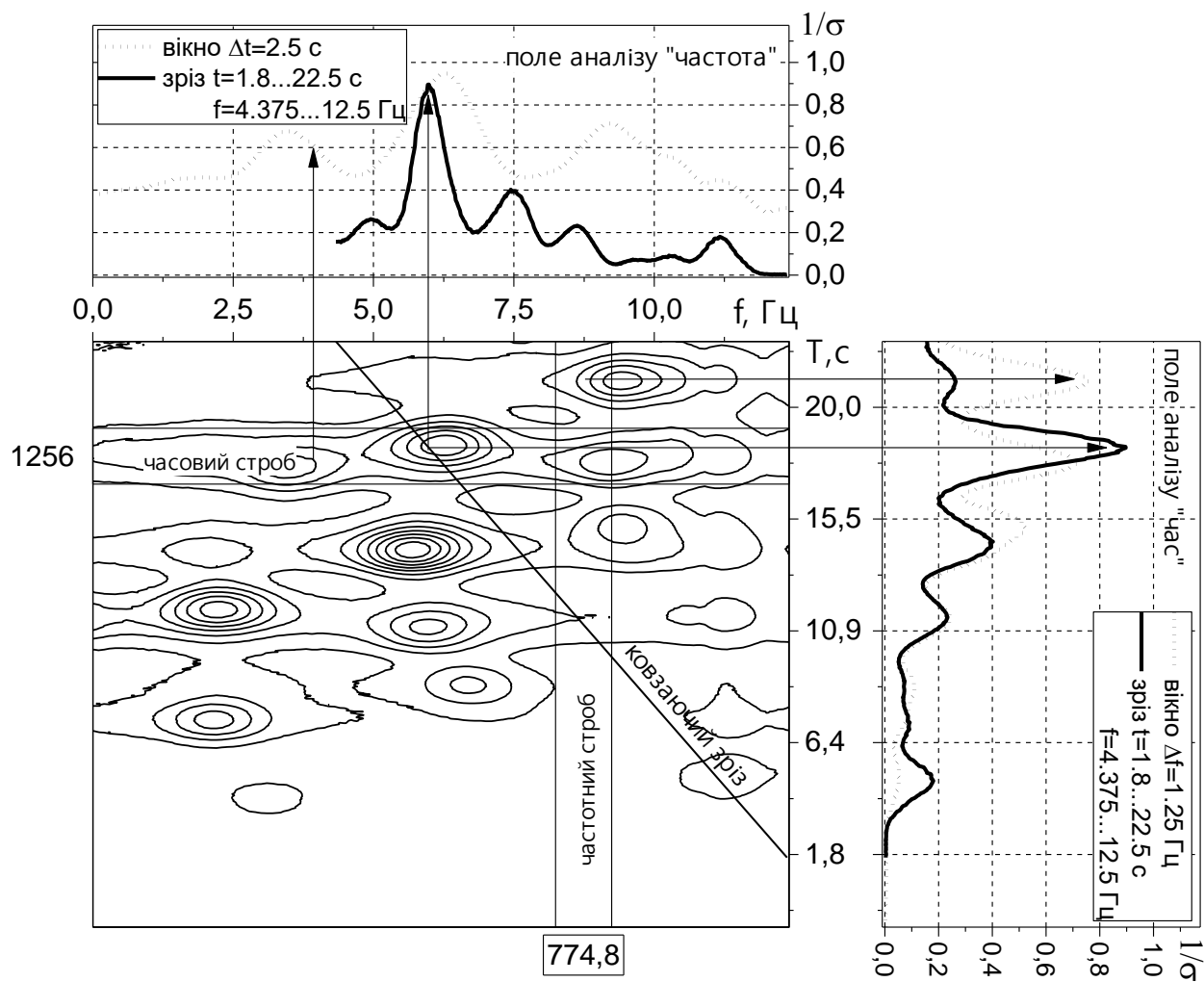


Рисунок 4 – Відображення відносної метрики сигналів в часово-частотному полі аналізу

ЗМІСТ ЛАБОРАТОНОГО ПРАКТИКУМУ

Лабораторна робота №1. Дослідження алгоритму навчання вимірювальної нейронної мережі з використанням багат шарового перцептрона

Завдання на дослідження:

1. Розробити алгоритм навчання нейронної мережі для ідентифікації датчиків тиску інтелектуальної вимірювальної інформаційної системи у середовищі LabView.

2. Дослідити вплив кількості перцептронів в прихованих шарах нейронної мережі на час та похибку навчання нейромережевого алгоритму.

Результати дослідження оформити у формі звіту.

Лабораторна робота №2. Дослідження алгоритму навчання вимірювальної нейронної мережі з використанням радіально-базисної функції

Завдання на дослідження:

1. Розробити алгоритм навчання нейронної мережі для ідентифікації датчиків тиску інтелектуальної вимірювальної інформаційної системи у середовищі LabView.

2. Дослідити вплив ваг та центрів радіально-базисної функції і на час та похибку навчання нейромережевого алгоритму.

Результати дослідження оформити у формі звіту.

Лабораторна робота №3. Дослідження алгоритму навчання вимірювальної нейронної мережі з використанням алгоритму Ельмана.

Завдання на дослідження:

1. Розробити алгоритм навчання нейронної мережі для ідентифікації датчиків тиску інтелектуальної вимірювальної інформаційної системи у середовищі LabView.

2. Дослідити вплив кількості нейронів в прихованих шарах нейронної мережі на час та похибку навчання нейромережевого алгоритму.

Результати дослідження оформити у формі звіту.

Лабораторна робота №4. Дослідження алгоритму навчання вимірювальної нейронної мережі з використанням генетичного

алгоритму.

Завдання на дослідження:

1. Розробити алгоритм навчання нейронної мережі для ідентифікації датчиків тиску інтелектуальної вимірювальної інформаційної системи у середовищі LabView.

2. Дослідити вплив вектора вхідних і вихідних ваг нейронної мережі на час та похибку навчання нейромережевого алгоритму.

Результати дослідження оформити у формі звіту.

Лабораторна робота №5. Дослідження алгоритму навчання вимірювальної нейронної мережі з використанням методу градієнтного спуску.

Завдання на дослідження:

1. Розробити алгоритм навчання нейронної мережі для ідентифікації датчиків тиску інтелектуальної вимірювальної інформаційної системи у середовищі LabView.

2. Дослідити вплив вектора напрямку x_k та величини кроку η_k оптимізації на час та похибку навчання нейромережевого алгоритму.

Результати дослідження оформити у формі звіту.

Лабораторна робота №6. Дослідження алгоритму навчання вимірювальної нейронної мережі з використанням алгоритму зворотного поширення помилки.

Завдання на дослідження:

1. Розробити алгоритм навчання нейронної мережі для ідентифікації датчиків тиску інтелектуальної вимірювальної інформаційної системи у середовищі LabView.

2. Дослідити вплив ваг між шарами на час та похибку навчання нейромережевого алгоритму.

Результати дослідження оформити у формі звіту.

Лабораторна робота №7. Дослідження алгоритму прийняття рішення на основі критерію просторово-часової метрики виміряних сигналів каналів тиску.

Завдання на дослідження:

1. Розробити алгоритм навчання нейронної мережі для ідентифікації датчиків тиску інтелектуальної вимірювальної інформаційної системи у середовищі LabView.

2. Дослідити вплив параметрів ковзаючого часово-частотного

вікна на час та похибку навчання.

3. Результати дослідження оформити у формі звіту.

ПЕРЕЛІК ТЕМ КУРСОВИХ РОБІТ

Завдання 1

Розробити в середовищі LabView нейромережевий алгоритм навчання ІВІС класифікації динамічних навантажень на робочі органи автогрейдера ДЗК-250.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 2

Розробити в середовищі LabView нейромережевий алгоритм вимірювання тиску в гідравлічній системі автогрейдера ДЗК-250.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 5

Розробити в середовищі LabView нейромережевий алгоритм розпізнавання просторового положення контактної рейки відносно колії метрополітену.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 6

Розробити в середовищі LabView нейромережевий алгоритм ідентифікації режимів роботи двигуна автогрейдера ДЗК-250.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 7

Розробити в середовищі LabView нейромережевий алгоритм ідентифікації вібрацій в дорожній машині.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 8

Розробити в середовищі LabView нейромережевий алгоритм ідентифікації вібрацій автомобіля.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 9

Розробити в середовищі LabView нейромережевий алгоритм розпізнавання дорожньої розмітки.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 10

Розробити в середовищі LabView нейромережевий алгоритм розпізнавання дорожніх знаків.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 11

Розробити в середовищі LabView нейромережевий алгоритм ідентифікації рівня забруднення повітря в технологічних приміщеннях.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 12

Розробити в середовищі LabView нейромережевий алгоритм прогнозування динамічних похибок вимірювання тиску на технічно складних об'єктах.

Описати процес розробки алгоритму з приведенням блок-

схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 13

Розробити в середовищі LabView нейромережевий алгоритм розпізнавання дефектів об'єм підшипників.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 14

Розробити в середовищі LabView нейромережевий алгоритм розпізнавання дефектів блоку циліндрів двигуна внутрішнього згорання.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

Завдання 15

Розробити в середовищі LabView нейромережевий алгоритм ідентифікації режимів навантаження на генератор дорожньої машини.

Описати процес розробки алгоритму з приведенням блок-схеми. Привести результати роботи нейромережевого алгоритму та скріншоти екранів.

