

буде незначна кількість. У випадку, коли на аналізованому відрізку  $L$  зустрінеться каскад зсувів та помітна кількість збурених результатів, описані алгоритми можуть виявитись неефективними. Хоча в практиці геомагнітних спостережень випадки подібного поєднання таких дефектів зустрічаються досить рідко, однак задача розробки сумісного алгоритму ідентифікації та сортування обох типів дефектів має значний практичний і теоретичний інтерес і буде нами розглянута в майбутньому.

1. Ю.М.Городиський, Є.С.Струк, Т.А.Климкович Метод сортування часових рядів з помітною кількістю некондиційних даних. // Вісник ДУ "Львівська політехніка" №392. С.141-145.
2. Дж.Бокс, Г.Дженкінс Аналіз временных рядов. Прогноз и управление. Вып. I., М., Мир, 1974.
3. Е.И.Пустельник Статистические методы анализа и обработки наблюдений. М., Наука, 1968.
4. Р.Отнес, Л.Эноксон Прикладной анализ временных рядов. Основные методы. М., Мир, 1982.

УДК 681.142.37

## НЕЙРОМЕРЕЖА З ФУНКЦІОНАЛЬНИМ РОЗШИРЕННЯМ ВХОДІВ

© Ірина Юрчак

НУ "Львівська політехніка", м. Львів, вул.С. Бандери, 12

*На основі парадигми "Функціонал на множині табличних функцій" реалізована віртуальна нейромережа. Створено програмний продукт штучної нейронної мережі, який можна використовувати при прогнозуванні процесів з різних галузей економіки, енергетики, математики.*

*There is realised virtual neural network that is based on the paradigm "Functional on the table function set". The software of artificial neural network that can be used for the prognose of processes from the differ branches of economic, energetic, mathematic and etc. are builded.*

За останні роки в більшості промислово розвинутих країнах були розгорнуті широкомасштабні національні та міжнародні програми досліджень та розробок, які спрямлені на створення нейрокомп'ютерів – ЕОМ на основі штучних нейронних мереж. Нейрокомп'ютери дозволяють з високою ефективністю вирішувати цілий ряд "інтелектуальних" задач, таких як розпізнавання образів, обробка сигналів, виконання прогнозів, діагностики, оптимізації, асоціативної пам'яті та управління.

На даний час найбільш масовим напрямком є моделювання нейронних мереж на звичайних комп'ютерах, перш за все персональних. Моделювання мереж виконується для наукових досліджень, для рішення практичних задач, а також для оцінки значень параметрів електронних та оптоелектронних нейрокомп'ютерів. Неослабна наукова активність в цій області дає надії, що нові інтелектуальні системи дозволять замінити традиційні ЕОМ в певних областях.

Представляємо програмну реалізацію штучної нейронної мережі на основі моделі "Функціонал на множині табличних функцій" з функціональним розширенням входів [1]. Програма дає змогу досліджувати вплив зміни кількості додаткових входних потоків на результат процесу прогнозування. Програмна реалізація здійснена з використанням середовища розробки *Borland Delphi 3*. Для роботи з програмою потрібно ПК з операційною системою сумісною з *Win32*, не менше 3 мегабайт дискового простору та не менше 16 мегабайт оперативної пам'яті.

Функціональне розширення входів розширяє стандартну архітектуру нейромереж, щоб включити вузли на входному рівні, які надають мережі більш повне розуміння входу. Головним чином, входи перетворюються добре зрозумілим шляхом таким чином, що мережа не повинна навчатись деяким базовим математичним функціям. Ці математичні функції розширюють розуміння мережею заданої проблеми і перетворюють входи за допомогою функцій вищого порядку, таких як квадрати, куби або синуси. Із власне назви цих функцій, вищого порядку або функціонально пов'язаних відображень, ці дві назви для того ж самого поняття є похідними.

Ця методика, як було показано, вражаючи покращує оцінки навчання деяких застосувань. Додатковою перевагою цього розширення є те, що ці функції вищого порядку можуть застосовуватись до будь-яких інших мереж.

Існує два основних способи додавання входних вузлів. Спочатку, у модель можуть додаватись перехресні добутки входних елементів. Це називається вихідним добутком або тензорною моделлю, де кожна компонента входного образу множиться з усім вектором входного образу. Прийнятним шляхом, щоб зробити це, додають всі елементи взаємодії між входними значеннями. Наприклад, для мережі з трьома входами ( $A$ ,  $B$  і  $C$ ), перехресні добутки включали б  $AA$ ,  $BB$ ,  $CC$ ,  $AB$ ,  $AC$  та  $BC$ . Цей приклад додає елементи другого порядку до входної структури мережі. Також можуть додаватись елементи третього порядку, такі як  $ABC$ .

Другим методом для додавання входних вузлів є функціональне розширення базових входів. Таким чином, модель зі входами  $A$ ,  $B$  і  $C$  могла б перетворитись у модель нейронної мережі вищого порядку з входами:  $A$ ,  $B$ ,  $C$ ,

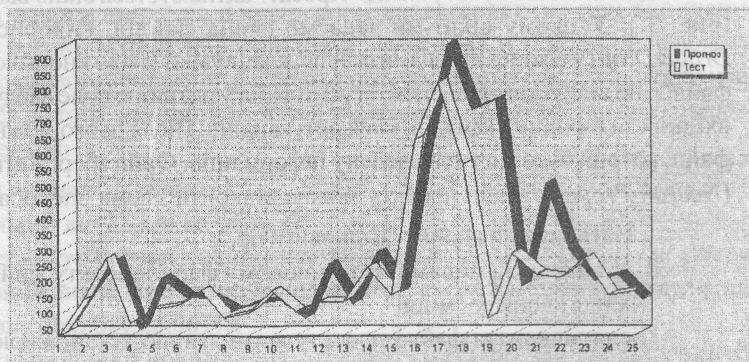


Рис. 1. Область виводу графічного відображення результатів

$SIN(A)$ ,  $COS(B)$ ,  $LOG(C)$ ,  $MAX(A, B, C)$  та ін. У цій моделі на вхідні змінні окремо діють відповідними функціями. Можуть використовуватись багато різних функцій. Повний ефект повинна забезпечити мережа з розширеним представленням вводу. Є навіть можливим об'єднати моделі тензорного та функціонального розширення разом.

Ніякої нової інформації не додається, але представлення входів розширюється. Високорівневе представлення вхідних даних може зробити мережу простішою для навчання. Комбіновані або функціональні активації стануть безпосередньо доступними для моделі. У деяких випадках схований рівень більше не потрібний. Проте існують обмеження для моделі мережі. Щоб використати перетворення початкових входів треба обробити набагато більше вхідних вузлів. З системами вищого порядку проблема загострюється. Проте через скінченний процесорний час комп'ютерів є важливим, що входи не розширюються більше, ніж це необхідно, щоб отримати точне рішення.

Функціонально-пов'язані мережі були розроблені Йох-Хан Пао та документовані у статті "Адаптивне розпізнавання образів та нейронні мережі" [2]. Пао проводить різницю між насправді додаванням елементів вищого порядку у тому сенсі, що деякі з

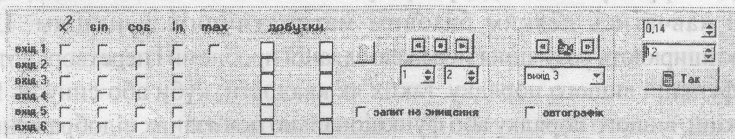


Рис. 2. Область вибору параметрів розрахунку

цих елементів є комбінованими активаціями та функціональними розширеннями, яке збільшує розмірність простору представлення без додавання комбінованих активацій. В той час, як більшість розробників розпізнають різницю, дослідники звичайно трактують ці два аспекти у той самий спосіб.

По суті, дана розробка є новим продовженням віртуального нейрокомп'ютера [3], у розробці якого брав безпосередню участь Ядловський В.С.

Програма виконана з типовим віконним інтерфейсом системи Win32. Графічне відображення системи розділено на три складові:

1. Область виводу графічного відображення результатів.
2. Область вибору параметрів розрахунку.
3. Область виводу додаткових повідомлень.

З допомогою пункту меню "Проект" активізується вікно вводу вхідних параметрів (рис. 3.). У цьому вікні ми задасмо імена файлів, використовуючи технологію Drag&Drop (у віконці з іменами файлів, клацнувши лівою кнопкою "миші" на потрібному файлі і, не відпускаючи її, перетягуємо файл у потрібне поле вводу), задаємо розмірності вхідних та вихідних каналів, кількості каналів. Після цього можна утворити проектний файл, котрий буде містити введену інформацію. Файл збережеться з розширенням *npr* (*Neuron Project*). Файл проекту можна відкрити двома способами:

1. Натиснувши кнопку "Відновити пр."
2. У віконці з файлами на потрібному файлі проекту здійснити подвійне натиснення кнопкою "миші".

Після відкриття файла проекту автоматично встановлюються початкові вхідні дані.




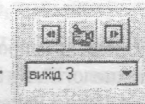
В основному користувач працює з областю вибору параметрів. В ній він задає додаткові функціональні розширення для нейромережі. Тут він може вибирати канал, який є активним для графічного відображення. Може здійснювати зміну режиму відображення (автоматичний/командний), задавати якісні параметри системи.

Складові області вибору параметрів:

1. Область вибору додаткових функціонально розширених входів. Тут користувач простим вибором вказує чи потрібно доповняти вхідний потік вказаними функціональними розширеннями. Для задання добуток використовується технологія *Drag&Drop*.

	<input checked="" type="checkbox"/>	sin	cos	ln	max	добутки	
вхід 1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
вхід 2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
вхід 3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
вхід 4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
вхід 5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
вхід 6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. Допоміжна область вибору додаткових попарних добуток. За допомогою курсорів здійснюється вибір потрібного добутку. Кнопка  здійснює включення/виключення даного добутку у вхідний потік.



3. Область вибору вихідного каналу для відображення.




4. Область вибору якісних коефіцієнтів.

5. Додаткові управляючі елементи:

**запит на знищення** увімкнення/вимкнення попередження,

**автографік** автоматичне поновлення результатів при заміні вхідних параметрів,

 **знищення останнього заданого попарного добутку.**

При використанні даного програмного продукту спостерігалось видиме покращання результатів вихідних потоків. Поєднання двох методів функціонального розширення входів, а також використання додаткових показало доцільність застосування даних методів.

Програма була використана для порівняльної характеристики при вирішенні популярної задачі по бетону. Вхідні дані були взяті з книги [4], в якій автор наводить приклад обчислення кускового метод опису кривих повзучості бетону І.Е.Прокоповича. Суть методу зводиться до того, що бетони з точки зору їх віку при завантаженні підрозділяються на три категорії:

1. старі бетони - понад 360 діб;
2. старіючі бетони - від 28 діб до 360 діб;
3. інтенсивно старіючі бетони - менше ніж 28 діб.

Для апроксимації кривих повзучості були використані математичні формули, що дали відносну граничну похибку 28,067%.

Далі була здійснена спроба покращання результату методом найшвидшого спус-



ку; при цьому було досягнуто покращання результату, що становило 25,391%. Знову знайдені значення параметрів були взяті за початкові приближення при повторному використанні методу Ньютона-Рафасона і отримано швидкосхідне рішення яке і було визнане кінцевим та дало результат 23,887%.

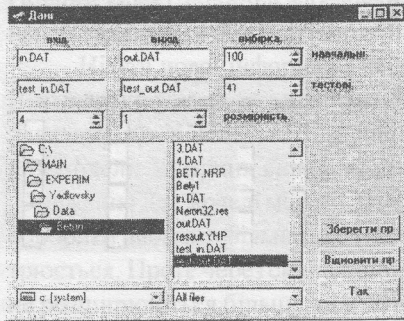


Рис. 3. Вікно вводу вхідних даних

При використанні даної програми без розширення входів було досягнуто результат 26,704%, а при розширенні вхідних потоків спостерігалось покращання результату - 11,014% та 10,095%.

Треба зауважити, що надлишкове розширення вхідних потоків негативно впливало на результат. При експериментах спостерігалась схожість результату різних варіантів розширення, котрі давали майже рівноцінні покращання, але об'єднування варіантів, як правило приводило до погіршення результату – що і підтверджує вищезгадану думку.

1. Ткаченко Р.О. Модель нейронних мереж// Вісник ДУ "Львівська політехніка" "Комп'ютерна інженерія та інформаційні технології". № 349. 1998. С. 386.
2. Yoh-Han Pao Adaptive Pattern Recognition and Neural Networks // Proceedings of the IEEE, October, 1990. № 10. V. 78. Pp. 1550-1560.
3. Ткаченко Р.О., Юрчак І.Ю., Ядловський В.С. Віртуальна нейромережа// Технічні вісті, 1998. № 1(6), 2(7). С. 21-23.

УДК 683.1

## ФАКТОР ЧАСУ В РЕЛЯЦІЙНИХ БАЗАХ ДАНИХ

© Павло Жежнич

НУ "Львівська політехніка", м. Львів, вул.С. Бандери, 12

*Розглядаються питання збереження залежної від часу інформації в реляційних базах даних. Виділено два основні семантично незалежні часові параметри: дійсний час та час транзакції. Також розглядаються інші часові параметри.*

*This paper considers time-dependent information storage problems in relational*