

УДК 519.7:004.8  
КП  
№ДР 0110U000458  
Інв. №

Міністерство освіти і науки, молоді та спорту України  
Харківський національний університет радіоелектроніки  
(ХНУРЕ)

61166, м. Харків, пр. Леніна, 14  
тел. (057) 702-13-97; телефакс (057) 702-14-13

ЗАТВЕРДЖУЮ  
Проректор з наукової роботи ХНУРЕ  
д. ф.-м. н., проф.

М. І. Сліпченко

ЗВІТ  
ПРО НАУКОВО-ДОСЛІДНУ РОБОТУ  
**ЕВОЛЮЦІЙНІ ГІБРИДНІ СИСТЕМИ  
ОБЧИСЛЮВАЛЬНОГО ІНТЕЛЕКТУ ЗІ ЗМІННОЮ  
СТРУКТУРОЮ ДЛЯ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ**  
№ 245  
(проміжний)

Керівник НДР

д-р техн. наук, професор

Є. В. Бодянський

2011

Рукопис завершено 17 грудня 2011 р.

Результати цієї роботи розглянуто науково – методичною радою ХНУРЕ  
протокол від 28.12.2011 № 15

## СПИСОК АВТОРІВ

Керівник НДР, проф., г.н.с., д.т.н., проф.	Бодяньський Є.В. (реферат, вступ, висновки, розд. 1)
Відповідальний виконавець, п.н.с., к.т.н., с.н.с. п.н.с., к.т.н., с.н.с.	Плісс І.П. (підрозд. 1.3) Винокурова О.А. (підрозд. 1.1, 1.3, 1.5)
с.н.с.	Близнюк В.Г. (підрозд. 1.5)
с.н.с., к.т.н.	Тесленко Н.О. (підрозд. 1.2)
с.н.с., к.т.н., с.н.с.	Сліпченко О.В. (підрозд. 1.7)
г.н.с., д.т.н., с.н.с.	Попов С.В. (підрозд. 1.7)
асп.	Шкуро К.А. (підрозд. 1.7)
м.н.с., к.т.н.	Долотов А.І. (підрозд. 1.5)
м.н.с., к.т.н.	Вікторов Є.О. (підрозд. 1.4)
асп.	Тищенко О.К. (підрозд. 1.8)
ст. викл.	Губін В.О. (підрозд. 1.8)
н.с.	Чепенко Т.Є. (підрозд. 1.7)

с.н.с., к.т.н., доц.

Машталір С.В.

(підрозд. 1.6.1)

доц., к.т.н., доц.

Кириченко Л.О.

(підрозд. 1.6.2)

Керівник розділу НДР,

г.н.с., д.т.н., проф.

Тімофєєв В.О.

(розд. 2)

асп.

Чуб О.В.

(розд. 2)

асп.

Мельникова М.О.

(розд. 2)

асп.

Волкотруб С. В.

(розд. 2)

асп.

Балакірева О.Г.

Керівник розділу НДР,

с.н.с., к.т.н., доц.

Корабльов М.М.

(розд. 3)

к.т.н., доц.

Аксак Н.Г.

(розд. 3)

асп.

Кушнар'ов М.В.

(розд. 3)

інж.

Новосельцев І.В.

(розд. 3)

асп.

Шкловець А.В.

(розд. 3)

асп.

Мохаммад Амін Салех

(розд. 3)

стар. викл.

Сорокіна І.В.

М.Н.С.	(розд 3) Фомічев А.А.
інж.	(розд. 3) Макогон А.Е.
асистент	(розд. 3) Лебьодкіна А.Ю.
Керівник розділу НДР, Г.Н.С., Д.Т.Н., проф.	(розд. 3) Кучеренко Є.І.
К.Т.Н., СТ. ВИКЛ.	(розд. 4) Творошенко І.С.
СТ. ВИКЛ.	(розд. 4) Глушенкова І.С.
маг.	(розд. 4) Корніловський А.В.
маг.	(розд. 4) Дрюк О.Д.
маг.	(розд. 4) Корнєєв Е.О.
студ.	(розд. 4) Кобозєв О.О.
студ.	(розд. 4) Андрієнко К.О.
Керівник розділу НДР, Г.Н.С., Д.Т.Н., проф.	(розд. 4) Філатов В.О.
доц., К.Т.Н., доц.	(розд. 5) Танянский С.С.

к.т.н.	(розд. 5) Касаткіна Н.В.
асп.	(розд. 5) Чапланова О.Б.
інж.	(розд. 5) Костіна З.Л.
	(розд. 5)

## РЕФЕРАТ

Звіт про НДР: 196 с., 54 рис., 11 табл., 159 посилань.

Об'єкт дослідження – інтелектуальний аналіз даних за умов повної або часткової апріорної та поточної невизначеності на основі гібридних систем обчислювального інтелекту зі змінною структурою, що еволюціонують.

Предмет дослідження – еволюційні гібридні системи обчислювального інтелекту зі змінною структурою, що в процесі свого навчання в режимі послідовної обробки інформації настроюють не тільки свої синаптичні ваги, але й архітектуру.

Методи дослідження – еволюційні конекціоністські системи, теорія оптимізації, методи обчислювального інтелекту.

Мета дослідження – розробка МГУА-нейромереж з вузлами типу ортонейронів, Q-нейронів та адаптивних вейвлонів, що мають підвищені апроксимуючі властивості. Розробка та дослідження методів прийняття рішень про стан об'єктів за умов невизначеності, що ґрунтуються на еволюційних інтервальних фаззі моделях. Розробка та дослідження об'єктно-орієнтованих моделей та структур даних для систем обчислювального інтелекту.

Розроблено оптимальні методи навчання МГУА нейромереж із змінною архітектурою, методи прийняття рішень про стан об'єктів за умов невизначеності, що ґрунтуються на еволюційних інтервальних фаззі моделях. Розроблено методи ітераційного аналізу індукованих відеоданими багатовимірних часових рядів. Розроблено еволюційні системи зі змінною структурою для нечіткої ідентифікації нелінійних динамічних об'єктів на основі використання штучних імунних систем.

**ЕВОЛЮЦІЙНІ ГІБРИДНІ СИСТЕМИ, ОБЧИСЛЮВАЛЬНИЙ ІНТЕЛЕКТ, ЗМІННА СТРУКТУРА, МЕТОД ГРУПОВОГО УРАХУВАННЯ АРГУМЕНТІВ, ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ДАНИХ.**

## ЗМІСТ

ВСТУП.....	10
1 РОЗРОБКА МГУА-НЕЙРОМЕРЕЖ ІЗ ЗМІННОЮ АРХІТЕКТУРОЮ З ВУЗЛАМИ ТИПУ ОРТОНЕЙРОНІВ, Q-НЕЙРОНІВ ТА АДАПТИВНИХ ВЕЙВЛОНІВ, ЩО МАЮТЬ ПІДВИЩЕНІ АПРОКСИМУЮЧІ ВЛАСТИВОСТІ. ОПТИМАЛЬНІ МЕТОДИ НАВЧАННЯ МГУА-НЕЙРОМЕРЕЖ ІЗ ЗМІННОЮ АРХІТЕКТУРОЮ.....	
1.1 МГУА-нейронні мережі на основі поліноміальної апроксимації та оптимальні алгоритми їх навчання .....	14
1.2 МГУА-нейронні мережі на основі R-Нейронів та оптимальні алгоритми їх навчання.....	24
1.3 МГУА-нейронна мережа на основі адаптивних вейвлетів ( W-нейронів) та оптимальні алгоритми її навчання.....	33
1.4 МГУА-нейронна мережа на основі нейро-фаззи нейронів та оптимальні алгоритми її навчання.....	43
1.5 Результати імітаційного експерименту з МГУА-нейронними мережами .	49
1.6 Виявлення змін, які відбуваються в мультимедійних даних за допомогою аналізу часових рядів.....	58
1.6.1 Виявлення змін у потоці відеоданих на основі аналізу багатовимірних часових рядів.....	58
1.6.2 Застосування вейвлет-ентропії для дослідження зміни структури стохастичних сигналів.....	63
1.7 Еволюційні мережі на базі гібридних нейроподібних елементів .....	72
1.8 Прогнозувальна модель стислих даних на основі багат шарового персептрону та нео-нейро-фаззи мережі зустрічного поширення.....	79
2 РОЗРОБКА МОДЕЛЕЙ ОБ'ЄКТІВ КОНТРОЛЮ, УПРАВЛІННЯ ТА ІДЕНТИФІКАЦІЇ В АДАПТИВНИХ ЕВОЛЮЦІЙНИХ СИСТЕМАХ .....	
2.1 Математичні моделі замкнутих адаптивних еволюційних систем.....	86

2.2 Рекурентні алгоритми ідентифікації, побудовані на системах нерівностей .....	93
2.3 Розробка моделей контролю, ідентифікації та управління в умовах інтервальної невизначеності .....	103
2.3.1 РМНК і умови його збіжності. ....	105
2.3.2 Модифікований РМНК с критичними властивостями. ....	108
3 ЕВОЛЮЦІЙНІ СИСТЕМИ ЗІ ЗМІННОЮ СТРУКТУРОЮ НА ОСНОВІ НЕЧІТКОЇ ЛОГІКИ І ШС ДЛЯ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ ДИНАМІЧНИХ ОБ'ЄКТІВ ЗА УМОВ НЕВИЗНАЧЕНОСТІ .....	115
3.1 Адаптація нечіткої моделі Такагі-Сугено з використанням штучних імунних систем.....	118
3.1.1 Адаптивна нечітка модель Такагі-Сугено. ....	118
3.1.2 Взаємодія етапів структурної і параметричної адаптації нечіткої моделі Такагі-Сугено. ....	120
3.1.3 Структурна адаптація нечіткої моделі Такагі-Сугено.....	121
3.1.4 Імунний алгоритм структурної адаптації нечіткої моделі Такагі-Сугено.....	123
3.1.5 Імунний алгоритм параметричної адаптації нечіткої моделі Такагі-Сугено.....	126
3.2 Адаптація нечіткої нейронної мережі ANFIS з використанням штучних імунних систем.....	130
3.2.1. Адаптивні нечіткі нейронні мережі.....	130
3.2.2 Модель кодування параметрів АННМ.....	135
3.2.3 Імунний алгоритм адаптації нечіткої нейронної мережі.....	136
3.3 Експериментальні дослідження .....	140
4 МЕТОДИ ПРИЙНЯТТЯ РІШЕНЬ ПРО СТАН ОБ'ЄКТІВ ЗА УМОВ НЕВИЗНАЧЕНОСТІ, ЩО ҐРУНТУЮТЬСЯ НА ЕВОЛЮЦІЙНИХ ІНТЕРВАЛЬНИХ ФАЗЗИ МОДЕЛЯХ .....	149



4.1 Дослідження існуючих підходів і методів оцінювання станів просторово розподілених об'єктів на основі еволюційних алгоритмів .....	149
4.2 Розробка фаззи моделей процесів оцінювання станів просторово розподілених об'єктів.....	152
4.3 Розвиток модифікованих методів налаштування та навчання функцій належності на основі багатозначної інтервальної логіки в знання орієнтованих технологіях.....	156
4.4 Розвиток інформаційних технологій на основі розроблених методів і моделей .....	159
5 ДОСЛІДЖЕННЯ ТА РОЗРОБКА МЕТОДІВ ПОДАННЯ НЕЧІТКИХ МОДЕЛЕЙ ЗАСОБАМИ РЕЛЯЦІЙНИХ СИСТЕМ .....	165
5.1 Особливості нечітких моделей і нечіткого моделювання.....	166
5.2 Нечіткі граматики .....	168
5.3 Бази даних в системах нечіткого моделювання .....	171
ВИСНОВКИ.....	178
ПЕРЕЛІК ДЖЕРЕЛ.....	180

## ВСТУП

У цей час системи обчислювального інтелекту й, насамперед, штучні нейронні мережі одержали широке поширення для розв'язку великого класу задач ідентифікації, емуляції, інтелектуального керування, прогнозування часових рядів і т.п., завдяки своїм універсальним прогнозуючим можливостям і здатністю до навчання. Оскільки при розв'язанні низки практичних задач, коли обсяг навчальної вибірки обмежений, на перший план виходить фактор швидкості навчання, при цьому далеко не всі нейронні мережі (і, насамперед, найбільш популярні багат шарові архітектури, ті, які навчаються, за допомогою зворотного поширення похибок) задовольняють умовам реальної задачі через низьку швидкість процесу настроювання й можливого ефекту перенавчання. У якості одних з найбільш швидкодіючих мереж, які навчаються, на основі оптимізаційних процедур другого порядку, можна відзначити радіально-базисні нейронні мережі (RBFN), вихідний сигнал яких лінійно залежить від синаптичних ваг, що настроюються. Замість із тим RBFN страждають від, так званого, «прокльону розмірності», коли кількість нейронів прихованого шару (R-нейронів) експоненційно росте з ростом розмірності вхідного векторного сигналу.

Більш просунуті системи такі, як нейро-фаззі-мережі типу ANFIS, Такагі-Сугено-Канга, Ванга-Менделя й т.п., володіючи покращеними апроксимуючими властивостями і можливостями лінгвістичної інтерпретації результатів, піддані тим же недолікам, включаючи одночасно як перенавчання, так і прокляти розмірності.

Те ж саме можна сказати і про вейвлет-нейро-фаззі системи (WNFS), що побудовані на основі нейро-фаззі-систем і запропоновані для обробки нестационарних сигналів.

Подолати проблеми можна, розбиваючи тим або іншим способом вихідну задачу на множину підзадач з нижчою розмірністю та поєднуючи далі деяким чином одержувані розв'язки для досягнення необхідного результату. З

обчислювальної точки зору найбільш зручним у цій ситуації представляється Метод Групового Урахування Аргументів (МГУА), що продемонстрував свою ефективність при розв'язку множини різних практичних задач.

Є доцільним об'єднання ідей МГУА та систем обчислювального інтелекту для одержання якісно нових результатів в області Data Mining, інтелектуального керування та інших наукових напрямках.

У цей час достатнє широко відома МГУА-нейронна мережа, вузлами якої є двухвходові N-Адаліни [1], кожна з яких містить набір синаптичних ваг, що настроюються та обчислених за допомогою стандартного методу найменших квадратів, і забезпечує квадратичну апроксимацію відновлюваного нелінійного відображення.

Для забезпечення необхідної якості апроксимації такій нейронній мережі може знадобитися значна кількість прихованих шарів.

Більш висока якість із використанням квадратичної апроксимації може бути забезпечена за допомогою введених Q-нейронів, основними перевагами яких є висока швидкість навчання.

Універсальні апроксимуючі властивості МГУА-нейронних мереж можуть бути досягнуті на основі використання складених R-нейронів, що представляють собою по суті двухвходові RBFN з решіточним розбиттям площини вхідних змінних. Передбачене настроювання не тільки синаптичних ваг, але й центрів і рецепторних полів, при цьому замість традиційних Гауссіанів використовується ядра Єпанечнікова, похідні яких лінійні по параметрам, що настроюються, що дозволяє покращити швидкість процесу навчання.

В [2-4] були запропоновані нейронні мережі, що самоорганізують свою архітектуру з використанням МГУА-алгоритмів. Прикладом такої мережі є двічі багат шарова нейронна мережа, що має звичайну багат шарову архітектуру прямого поширення, у якій кожний, так званий, нейрон синтезує усередині себе багат шарову МГУА-структуру. Дана мережа продемонструвала

свою ефективність при розв'язку ряду практичних задач, однак її використання обмежується чисельною громіздкістю.

В [5-11] були введені архітектури мереж як прямого поширення, так і рекурентних із зворотним зв'язком (GMDH-FL), нейрони яких можуть мати поліноміальні, сигмоїдні або радіально-базисні активаційні функції. Синаптичні ваги нейронів цієї мережі визначаються в пакетному режимі за допомогою покрокового регресійного аналізу, при цьому покладається, що нейрони можуть мати як два, так і більш входів.

В [12-15] розглянута адаптивна нейронна мережа з радіально-базисними нейронами з використанням часткових описів МГУА. Ця мережа є багатошаровою системою з нечітким виведенням, у зв'язку із чим автори назвали її нейро-фаззі-МГУА (NF-GMDH). Для синтезу оптимальної архітектури мережі використовується деструктивний підхід, що видаляє неінформативні нейрони й зв'язки між ними.

В [16] показано, що NF-GMDH реалізує нечітке виведення по Мамдані [17] і запропонована процедура навчання цієї мережі.

В [18] на основі NF-GMDH і NF-GMDH запропонована тришарова нейро-фаззі система з покращеним зворотним зв'язком (NF-GMDH-IFL), що є об'єднанням описаних вище мереж.

Нейро-фаззі нейронна мережа, уведена в [19], оптимізує свою архітектуру за допомогою традиційного МГУА, а також реалізує нечітке виведення Сугено-Канга нульового порядку й характеризується чисельною простотою своєї реалізації.

Цей звіт присвячений синтезу чисельно простих і швидкодіючих алгоритмів навчання МГУА-нейронних мереж, що використовують різні функції активації: поліноміальні, радіально-базисні й інші, для розв'язку різних задач Data Mining на основі адаптивного підходу.

Більш складні задачі, пов'язані з обробкою стохастичних послідовностей, можуть бути вирішені на основі введених адаптивних W-нейронів (вейвлонів), відмінною рисою яких є те, що крім власне синаптичних ваг і параметрів

розтягання і зсуву, настроюється й форма активаційної функції, яка може утворювати різний вид від Гауссіана до вейвлета «Мексиканський капелюх». Уведені алгоритми навчання із властивостями, що згладжують та стежать і, на основі квадратичного критерію навчання, а так само робастні алгоритми, що дозволяють виключити вплив аномальних викидів на процес настроювання.

Теоретичні результати ілюструються імітаційним експериментом, що підтверджуються ефективність підходу, що розвивається.

# 1 РОЗБОРКА МГУА-НЕЙРОМЕРЕЖ ІЗ ЗМІННОЮ АРХІТЕКТУРОЮ З ВУЗЛАМИ ТИПУ ОРТОНЕЙРОНІВ, Q-НЕЙРОНІВ ТА АДАПТИВНИХ ВЕЙВЛОНІВ, ЩО МАЮТЬ ПІДВИЩЕНІ АПРОКСИМУЮЧІ ВЛАСТИВОСТІ. ОПТИМАЛЬНІ МЕТОДИ НАВЧАННЯ МГУА-НЕЙРОМЕРЕЖ ІЗ ЗМІННОЮ АРХІТЕКТУРОЮ

## 1.1 МГУА-нейронні мережі на основі поліноміальної апроксимації та оптимальні алгоритми їх навчання

Множина апроксимуючих штучних нейронних мереж природно не вичерпується багатошаровими перцептронами і радіально-базисними нейронними мережами, оскільки, маючи безсумні переваги, вони не позбавлені й істотних недоліків. Своєрідним гібридом багатошарових перцептронів і радіально-базисних мереж є функціонально зв'язані нейронні мережі (Functionally Linked Neural Networks) [20, 21], архітектура яких наведена на рис. 1.1.

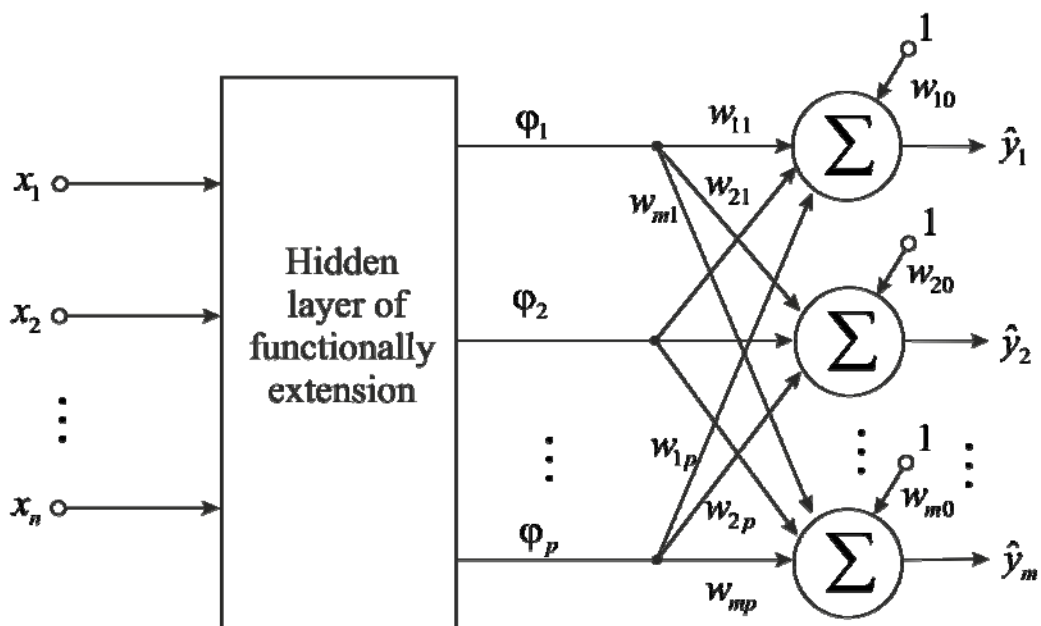


Рисунок 1.1 - Функціонально зв'язана нейронна мережа

Вхідний шар утворений  $n$ -вимірним вектором  $x = (x_1, x_2, \dots, x_n)^T$ . Прихований шар розширює вхідний простір за допомогою деякого нелінійного перетворення вектора  $x = (x_1, x_2, \dots, x_n)^T$  так, що відбувається відображення входів у новий простір підвищеної розмірності

$$x \in R^n \rightarrow (\varphi_1(x), \varphi_2(x), \dots, \varphi_p(x))^T \in R^p, \quad n < p. \quad (1.1)$$

Вихідний шар утворений  $m$  адаптивними лінійними асоціаторами й реалізує відображення розширеного простору у вихідний

$$\varphi(x) \in R^p \rightarrow \hat{y} = (\hat{y}_1, \hat{y}_2, \dots, \hat{y}_m)^T \in R^m, \quad m < p \quad (1.2)$$

так, що

$$\hat{y}_l = w_{l0} + \sum_{h=1}^p w_{lh} \varphi_h(x), \quad l = 1, 2, \dots, m. \quad (1.3)$$

Важливо, що кожна з функцій  $\varphi_h(x)$  залежить тільки від вхідного сигналу й використовуваного функціонального розширення й не містить вільних параметрів, що вигідно відрізняє функціонально зв'язані мережі від радіально-базисних.

На практиці використовується досить велика кількість різних функцій  $\varphi_h(x)$ , наприклад,  $|x_i - x_j|$ ,  $\sin \pi x_i$ ,  $\cos 2\pi x_i$ ,  $\exp(-x_i^2)$ ,  $\tanh x_i$ ,  $\exp(-x_i^2)x_j$  і т.п., хоча найбільше поширення одержало поліноміальне розширення, що дало назву підкласу поліноміальних нейронних мереж. Дані мережі реалізують перетворення

$$\begin{aligned}
y_l = & w_{l0} + \sum_{i=1}^n w_{li} x_i + \sum_{i_1=1}^n \sum_{i_2=i_1}^n w_{li_1 i_2} x_{i_1} x_{i_2} + \dots \\
& + \sum_{i_1=1}^n \sum_{i_2=i_1}^n \dots \sum_{i_h=i_{h-1}}^n w_{li_1 i_2 \dots i_h} x_{i_1} x_{i_2} \dots x_{i_h} = w_l^T \varphi(x),
\end{aligned} \tag{1.4}$$

(тут  $w_l$  - вектор синаптичних ваг  $l$ -го лінійного асоціатора,  $\varphi(x)$  - вихідний сигнал прихованого шару функціонального розширення, доповнений одиницею, що надходить на пороговий блок вихідного шару), в основі якого лежить поліноміальне розкладання Колмогорова-Габора [22]. Зпираючись на теорему Стоуна-Вейєрштрасса [23], можна показати, що при досить високих ступенях сигналів  $x_i$ , може бути досягнута як завгодно висока точність апроксимації, при цьому для настроювання синаптичних ваг поліноміальної мережі може бути використаний будь-який з лінійних алгоритмів навчання нейронних мереж.

Основним недоліком даної мережі є велика кількість налаштовуваних параметрів, обумовлена значенням

$$H = \frac{(n+h-1)!}{n!(h-1)!}, \tag{1.5}$$

яке при  $h > 3$  приводить до так званого «прокльону розмірності» [24].

Нескладно бачити, що при використанні кубічного розширення для двадцяти входів, число синаптичних ваг досягає значення 1270 [25], що дійсно викликає серйозні проблеми зі швидкістю навчання навіть в оптимальних по швидкодії алгоритмів.

Саме тут на допомогу приходить метод групового урахування аргументів, що лежить в основі МГУА-мережі [1], архітектура якої для випадку  $n = 4$ ,  $m = 1$  наведена на рис. 1.2.



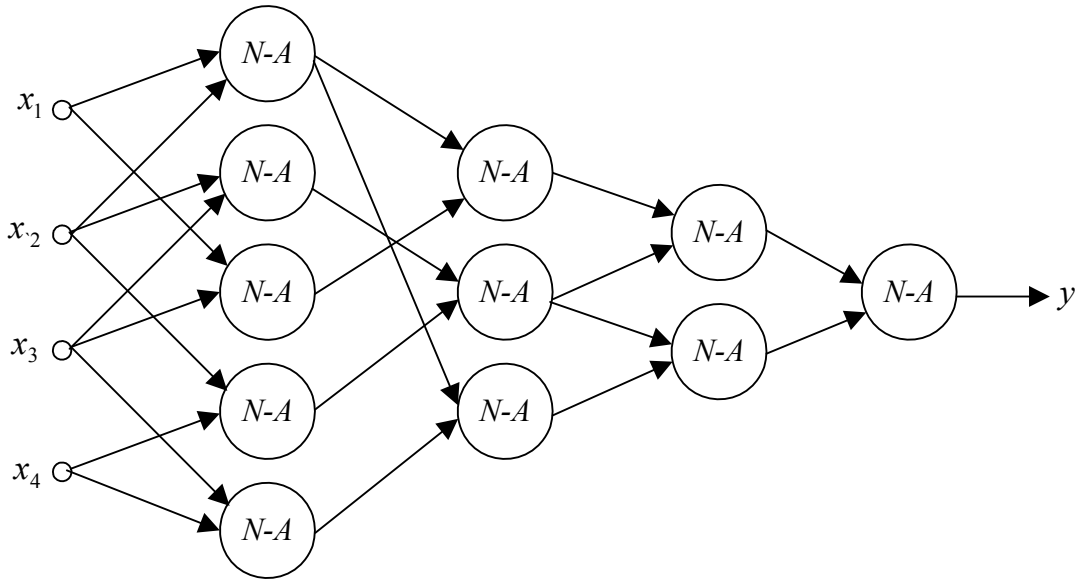


Рисунок 1.2 - МГУА – нейронна мережа

У якості вузлів GMDH-NN використовується Нелінійна Адаліна (N-A), схема якої наведена на рис. 1.3.

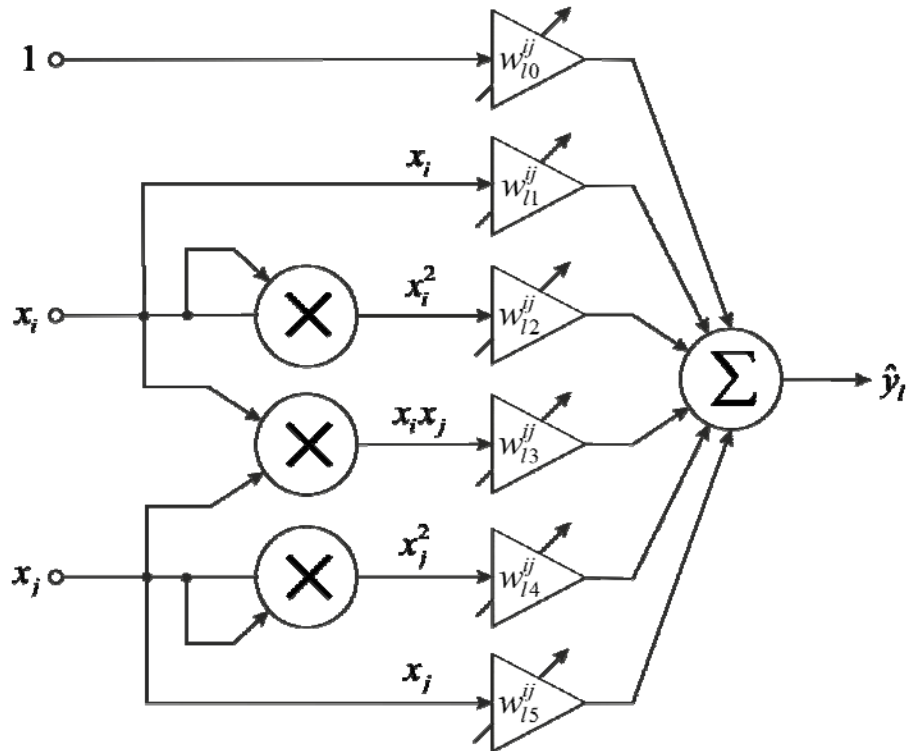


Рисунок 1.3 – N-адаліна

На відміну від традиційних нейромереж з фіксованою архітектурою МГУА-мережа має змінну структуру, яка може еволюціонувати в процесі навчання.

Кожний нейрон мережі -  $N$ -адаліна являє собою адаптивний лінійний асоціатор із двома входами й нелінійним препроцесором, утвореним трьома стандартними блоками множення, який обчислює квадратичну комбінацію виду

$$\hat{y}_i = w_{i0}^{ij} + w_{i1}^{ij}x_i + w_{i2}^{ij}x_i^2 + w_{i3}^{ij}x_ix_j + w_{i4}^{ij}x_j^2 + w_{i5}^{ij}x_j. \quad (1.6)$$

Процес навчання МГУА-мережі полягає в її конфігуруванні, починаючи із вхідного шару, настроюванні синаптичних ваг кожного нейрона й збільшенні кількості шарів для досягнення необхідної точності апроксимації.

Число нейронів першого шару мережі визначається розмірністю вхідного вектора  $n$  й не перевищує значення  $0.5n(n-1)$  - кількості комбінацій  $C_n^2$ . Кожний нейрон настроюється за допомогою кожного з алгоритмів навчання із вчителем, а в найпростішому випадку можна використовувати стандартний метод найменших квадратів у вигляді

$$w_i^{ij}(N) = \left( \sum_{k=1}^N \varphi^{ij}(x(k)) (\varphi^{ij}(x(k)))^T \right)^{-1} \sum_{k=1}^N \varphi^{ij}(x(k)) y_i(k), \quad (1.7)$$

де  $w_i^{ij}(N) = (w_{i0}^{ij}(N), w_{i1}^{ij}(N), w_{i2}^{ij}(N), w_{i3}^{ij}(N), w_{i4}^{ij}(N), w_{i5}^{ij}(N))^T$ ,

$\varphi^{ij}(x(k)) = (1, x_i(k), x_i^2(k), x_i(k)x_j(k), x_j^2(k), x_j(k))^T$ ,  $k = 1, 2, \dots, N$ ,  $k$  - номер

спостереження в навчальній вибірці або індекс поточного дискретного часу,  $N$  - обсяг навчальної вибірки,  $y_i(k)$  - значення зовнішнього навчального сигналу в  $k$ -й момент часу.

У режимі послідовної обробки може бути використаний кожний з лінійних алгоритмів навчання, прийнятих у теорії навчання нейронних мереж [26] або адаптивної ідентифікації [27], наприклад популярний оптимальний по швидкодії алгоритм Качмажа-Уїдроу-Гоффа [28, 29]

$$w_i^{jj}(k) = w_i^{jj}(k-1) + \frac{y_i(k) - (w_i^{jj}(k-1))^T \varphi^{jj}(x(k))}{\|\varphi^{jj}(x(k))\|^2} \varphi^{jj}(x(k)), \quad (1.8)$$

де  $w_i^{jj}(N) = (w_{i0}^{jj}(N), w_{i1}^{jj}(N), w_{i2}^{jj}(N), w_{i3}^{jj}(N), w_{i4}^{jj}(N), w_{i5}^{jj}(N))^T$ ,  
 $\varphi^{jj}(x(k)) = (1, x_i(k), x_i^2(k), x_i(k)x_j(k), x_j^2(k), x_j(k))^T$ .

Після пред'явлення мережі всієї навчальної вибірки оцінюється точність (залишкова дисперсія) кожного нейрона й формується група з нейронів, що дають помилку нижче деякого апріорі заданого порога. Саме виходи цієї групи є входами другого прихованого шару. Далі за допомогою тієї ж навчальної вибірки настроюються нейрони другого шару при «заморожених» вагах першого й знову формується група, що забезпечує найкращу точність. Процес нарощування шарів і настроювання синаптичних ваг триває доти, поки точність не перестане збільшуватися. Кращий нейрон останнього шару покладається вихідним нейроном мережі в цілому.

МГУА-мережа забезпечує високу точність апроксимації при порівняно невеликій кількості параметрів, що настроюються, однак навчатися може тільки в пакетному режимі (у випадку апріорно заданої навчальної вибірки).

У ряді випадків замість N-Адаліни із двома входами та шістьома синаптичними вагами, що настроюються доцільно використовувати нейрон, що забезпечує квадратичну апроксимацію (Q-Нейрон) із  $2 < q < n$  входами [30, 31]. На рис. 1.4 наведена структура Q-нейрона, що здійснює перетворення

$$\hat{y}_l(k) = w_{l_0}(k-1) + \sum_{i=1}^q w_{li}(k-1)x_i(k) + \sum_{i_1=1}^n \sum_{i_2=1}^n w_{li_1i_2}(k-1)x_{i_1}(k)x_{i_2}(k), \quad (1.9)$$

або

$$\hat{y}_l(k) = w_{l_0}(k-1) + b_l^T(k-1)x(k) + x^T(k)C_l(k-1)x(k), \quad (1.10)$$

або в більш компактній матричній формі [32]

$$\hat{y}_l(k) = \vec{x}^T(k)W_l(k-1)\vec{x}(k), \quad (1.11)$$

де  $b_l(k-1) = (w_{l_1}(k-1), w_{l_2}(k-1), \dots, w_{l_q}(k-1))^T - (q \times 1)$  – вектор,  
 $C_l(k-1) = \{w_{li_1i_2}(k-1)\} - (q \times q)$  - матриця,  $x(k) = (x_1(k), x_2(k), \dots, x_q(k))^T - (q \times 1)$   
 - вектор,  $\vec{x}^T(k) = (1, x^T(k))$ ,

$$W_l(k-1) = \left\{ \begin{array}{c|c} w_{l_0}(k-1) & 0.5b_l^T(k-1) \\ \hline 0.5b_l(k-1) & C_l(k-1) \end{array} \right\} \quad (1.12)$$

- блокова  $(q+1) \times (q+1)$  - матриця.

Природно, що кількість синаптичних ваг у цьому випадку суттєво збільшується, однак простота реалізації найчастіше забезпечує перевага саме таким моделям.

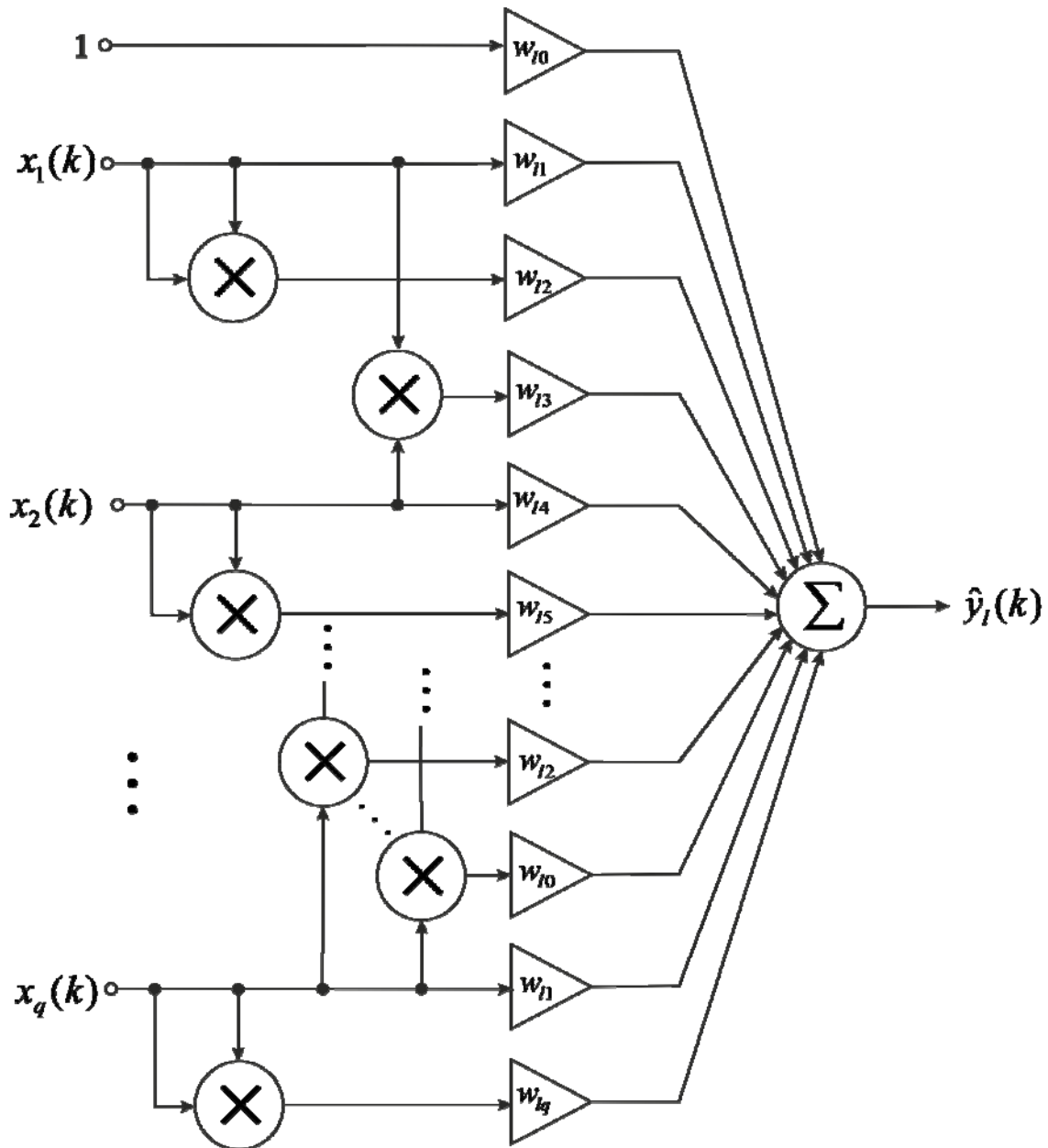


Рисунок 1.4 - Структура Q-Нейрона

Настроювання матриці синаптичних ваг  $W_l(k)$  будемо здійснювати шляхом мінімізації критерію навчання

$$E_l(k) = \frac{1}{2} e_l^2(k) = \frac{1}{2} (y_l(k) - \vec{x}^T(k) W_l(k-1) \vec{x}(k))^2 \quad (1.13)$$

за допомогою градієнтної процедури

$$W_l(k) = W_l(k-1) + \eta(k) e_l(k) \vec{x}(k) \vec{x}^T(k), \quad (1.14)$$

де

$$e_i(k) = y(k) - \vec{x}^T(k)W_i(k-1)\vec{x}(k). \quad (1.15)$$

$\eta(k)$  - крок навчання.

Для знаходження параметра  $\eta(k)$ , що забезпечує алгоритму (1.14) оптимальні властивості, введемо функцію Ляпунова виду [33]

$$V(k) = Tr(\tilde{W}_i(k-1)\tilde{W}_i^T(k-1)) - Tr(\tilde{W}_i(k)\tilde{W}_i^T(k)), \quad (1.16)$$

максимізація якої на кожному кроці відповідає вимозі найбільшої швидкості убування середньої дисперсії коефіцієнтів матриці  $\tilde{W}_i(k)$ , де  $\tilde{W}_i(k) = W_i - W_i(k)$  - матриця відхилень поточних значень  $W_i(k)$  від невідомих оптимальних  $W_i$ , а  $Tr(\bullet)$  - символ сліду матриці.

Записуючи (1.14) у формі

$$\tilde{W}_i(k) = \tilde{W}_i(k-1) + \eta(k)e_i(k)\vec{x}(k)\vec{x}^T(k) \quad (1.17)$$

і підставляючи (1.17) в (1.16), можна записати

$$\begin{aligned} V(k) &= \eta(k)e_i(k)Tr(\tilde{W}_i(k-1)\vec{x}(k)\vec{x}^T(k)) + \eta(k)e_i(k)Tr(\vec{x}(k)\vec{x}^T(k)\tilde{W}_i(k-1)) - \\ &- \eta^2(k)e_i^2(k)Tr(\vec{x}(k)\vec{x}^T(k)\vec{x}(k)\vec{x}^T(k)) = 2\eta(k)e_i(k)\vec{x}^T(k)\tilde{W}_i(k-1)\vec{x}(k) - \\ &- \eta^2(k)e_i^2(k)\|\vec{x}(k)\|^4, \end{aligned} \quad (1.18)$$

після чого, вирішуючи диференціальне рівняння

$$\frac{\partial V(k)}{\partial \eta(k)} = 0, \quad (1.19)$$

с урахуванням того, що

$$e_i(k) = \vec{x}^T(k) \tilde{W}_i(k-1) \vec{x}(k),$$

можна одержати оптимальне значення параметра кроку у вигляді [32]

$$\eta(k) = \|\vec{x}(k)\|^{-4}. \quad (1.20)$$

Підстановка (1.20) в (1.14) приводить до алгоритму навчання

$$W_i(k) = W_i(k-1) + \frac{y_i(k) - \vec{x}^T(k) W_i(k-1) \vec{x}(k)}{\|\vec{x}(k)\|^4} \vec{x}(k) \vec{x}^T(k), \quad (1.21)$$

що є розширенням алгоритму Качмажа-Уїдрой-Гоффа на Q-нейрон. Помітимо, що використовуючи операцію псевдообернення матриць, можна переписати (1.21) у вигляді

$$\begin{aligned} W_i(k) &= W_i(k-1) + (y_i(k) - \vec{x}^T(k) W_i(k-1) \vec{x}(k)) (\vec{x}(k) \vec{x}^T(k))^+ = \\ &= W_i(k-1) + e_i(k) (\vec{x}(k) \vec{x}^T(k))^+. \end{aligned} \quad (1.22)$$

де  $(\bullet)^+$  - символ псевдообернення матриці по Муру-Пенроузу.

## 1.2 МГУА-нейронні мережі на основі R-Нейронів та оптимальні алгоритми їх навчання

Покращити апроксимуючі властивості МГУА-нейронних мереж можна, поєднуючи підходи, засновані на МГУА й радіально-базисних нейронних мережах, що мають єдиний прихований шар, утворений, так званими, R-нейронами, при цьому навчання таких мереж зводиться, як правило, до налаштування синаптичних ваг вихідного шару, утвореного адаптивними лінійними асоціаторами. На відміну від поліноміальних або сигмоїдальних активаційних функцій багат шарових архітектур R-нейрони мають дзвонувату функцію активації  $\varphi(x, c, \sigma)$ , аргументами якої є відстань між вхідним вектором-образом  $x$  і власним центром  $c$  і параметр ширини  $\sigma$ .

Основною перевагою РБНС є висока швидкість навчання, обумовлена тим, що вихідний сигнал мережі  $\hat{y}$  лінійно залежить від синаптичних ваг, що дозволяє для їхнього налаштування використовувати процедури оптимізації другого порядку, включаючи звичайний метод найменших квадратів як у пакетній, так і в рекурентній формах.

Разом з тим, як ми вже відзначали, звичайні радіально-базисні мережі мають істотний недолік, пов'язаний з «прокльоном розмірності», коли число нейронів у прихованому шарі експоненційно росте зі збільшенням розмірності вхідного простору  $n$  й визначається величиною порядку  $\exp n$  [34]. Зрозуміло, що при досить невеликій розмірності вхідного вектора  $n=10$ , кількість параметрів, що налаштовуються, має порядок  $5 \cdot 10^4$ , що суттєво ускладнює процес навчання. Кількість R-нейронів у принципі можна скоротити, вибираючи або налаштовуючи тим або іншим способом їх центри й ширину, однак при цьому губиться головна перевага RBFN – швидкість навчання.

Введемо до розгляду структуру складеного R-нейрона (CR-N), що наведена на рис.1.5 і співпадаючу по суті зі спрощеною архітектурою звичайної радіально-базисної мережі із двома входами  $x_i$  й  $x_j$ . CR-N містить дев'ять (exp



$2 \approx 9$ ) дзвонуватих активаційних функцій  $\varphi_h^{ij}$ ,  $h=1,2,\dots,9$ ;  $\varphi_0^{ij} \equiv 1$  і десять синаптичних ваг, що настраюються, об'єднаних у вектор  $w_i^{ij} = (w_{i0}^{ij}, w_{i1}^{ij}, \dots, w_{i9}^{ij})^T$ .

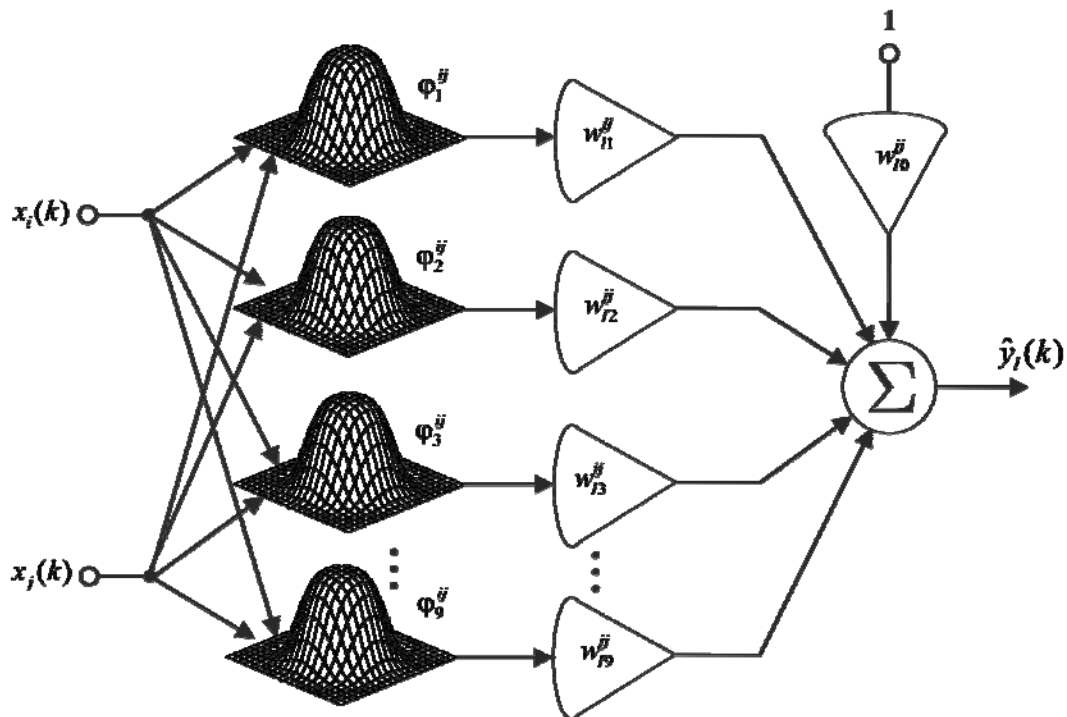


Рисунок 1.5 - Складений R-Нейрон

У якості активаційних функцій  $\varphi_h^{ij}$  можуть бути використані гаусіани

$$\varphi_h^{ij}(x, c, \sigma) = \exp\left(-\frac{(x_i - c_h^i)^2 + (x_j - c_h^j)^2}{2\sigma^2}\right) = \exp\left(-\frac{\|x^{ij} - c_h^{ij}\|^2}{2\sigma^2}\right),$$

де  $c_h^{ij} = (c_h^i, c_h^j)^T$  - вектор центрів, при цьому, якщо вхідні дані попередньо перетворені так, що  $-1 \leq x_i \leq 1, -1 \leq x_j \leq 1$ , ці центри утворюють решітку, показану на рис.1.6.

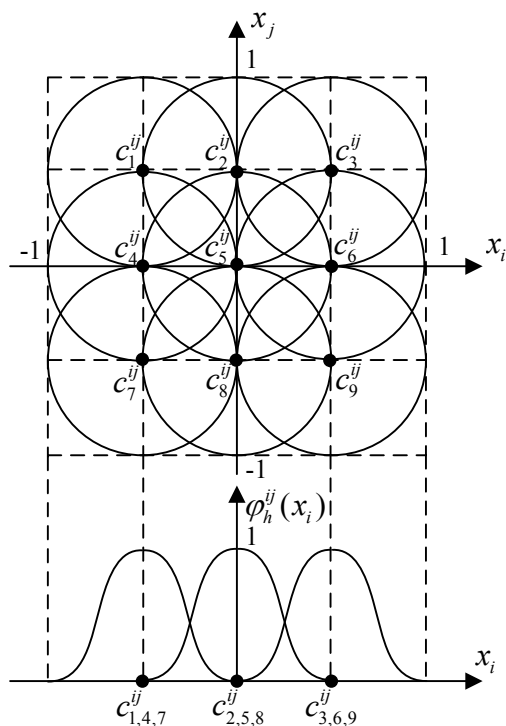


Рисунок 1.6 - Активаційна функція складеного R-нейрона

Параметр ширини  $\sigma$  для гаусіана в цьому випадку може бути обраний на рівні  $\sigma \leq 0,33$ , а для запобігання можливості виникнення «дір» (holes) у просторі активаційних функцій вони можуть бути ренормалізовані [35] безпосередньо в процесі навчання нейрона.

Процес навчання полягає у визначенні вектора синаптичних ваг  $w_l^{ij}$  по навчальній вибірці, що має  $N$  спостережень  $x^{ij}(k) = (x_i(k), x_j(k))^T$ ,  $y(k)$ ,  $k = 1, \dots, N$ , де  $y(k)$  - навчальний зовнішній сигнал.

Вводячи до розгляду  $(10 \times 1)$ -вектор активаційних функцій  $\varphi^{ij}(x(k)) = (1, \varphi_1^{ij}(x^{ij}(k)), \dots, \varphi_9^{ij}(x^{ij}(k)))^T$  і критерій навчання

$$E_l^N = \sum_{k=1}^N (y(k) - \hat{y}_l(k))^2 = \sum_{k=1}^N e_l^2(k) = \sum_{k=1}^N (y(k) - (w_l^{ij})^T \varphi^{ij}(x(k)))^2 \quad (1.23)$$

за допомогою стандартного методу найменших квадратів нескладно одержати шукані оцінки у вигляді

$$w_l^{ij} = \left( \sum_{k=1}^N \varphi^{ij}(x(k)) (\varphi^{ij}(x(k)))^T \right)^+ \sum_{k=1}^N \varphi^{ij}(x(k)) y(k). \quad (1.24)$$

Якість навчання може бути оцінена за допомогою перевіркової вибірки  $x^{ij}(k), y(k), k = N + 1, N + 2, \dots, N + M$  на підставі вибіркової дисперсії

$$\sigma_{\hat{y}_l}^2 = \frac{1}{M} \sum_{k=N+1}^{N+M} (y(k) - (w_l^{ij})^T \varphi^{ij}(x(k)))^2.$$

Складений R-нейрон є основним вузлом в архітектурі еволюційної багаторядної МГУА-нейронної мережі, наведеної на рис.1.7.

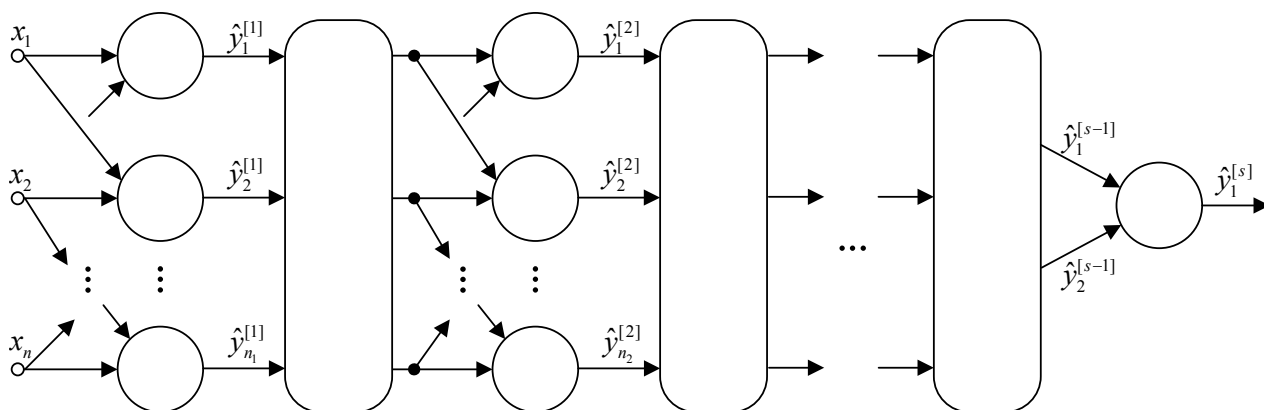


Рисунок 1.7 - Еволюційна багаторядна МГУА-нейронна мережа

На нульовий (рецепторний) шар мережі надходить  $(n \times 1)$ -вимірний вектор вхідних сигналів  $x = (x_1, x_2, \dots, x_n)^T$ , який далі подається на перший прихований шар, що містить  $n_1 = c_n^2$  складених нейронів, на виході яких з'являються сигнали  $\hat{y}_l^{[1]}, l = 1, 2, \dots, 0.5n(n-1)$ . Ці сигнали надходять на блок селекції першого прихованого шару  $SB^{[1]}$ , який відбирає із множини  $\hat{y}_l^{[1]} n_1^* < n_1$  найкращих у сенсі дисперсії  $\sigma_{\hat{y}_l^{[1]}}^2$  з яких формується  $n_2 = c_{n_1}^2$  (звичайно  $n \leq n_2 \leq 2n$ ) комбінацій  $\hat{y}_i^{[1]}, \hat{y}_j^{[1]}$ , що подаються на другий прихований шар,  $CR-N^{[2]}$

утворений нейронами  $CR - N^{[2]}$ . З вихідних сигналів цього шару  $\hat{y}_i^{[2]}$  блок селекції  $SB^{[2]}$  відбирає тільки ті, які перевершують по точності найкращий сигнал першого прихованого шару  $\hat{y}_i^{[1]*}$ . Третій прихований шар формує сигнали, що перевершують найкращий сигнал  $\hat{y}_i^{[2]*}$  і т.д. Процес еволюції мережі відбувається доти, поки блок селекції  $SB^{[s-1]}$  не сформує на своєму виході всього два сигнали  $\hat{y}_1^{[s-1]}$  й  $\hat{y}_2^{[s-1]}$ . Саме ці два сигнали надходять на вихідний нейрон  $CR - N^{[s]}$ , який обчислює вихідний сигнал мережі в цілому  $\hat{y}^{[s]}$ , при цьому автоматично виконується умова  $\sigma_{\hat{y}_1^{[s]}}^2 < \sigma_{\hat{y}_2^{[s-1]*}}^2 < \dots < \sigma_{\hat{y}_1^{[1]*}}^2$ .

Чисельна громіздкість багаторядної МГУА-нейронної мережі змушує шукати альтернативний варіант, у якості якого може бути використана еволюційна каскадна МГУА-нейронна мережа, архітектура якої наведена на рис.1.8.

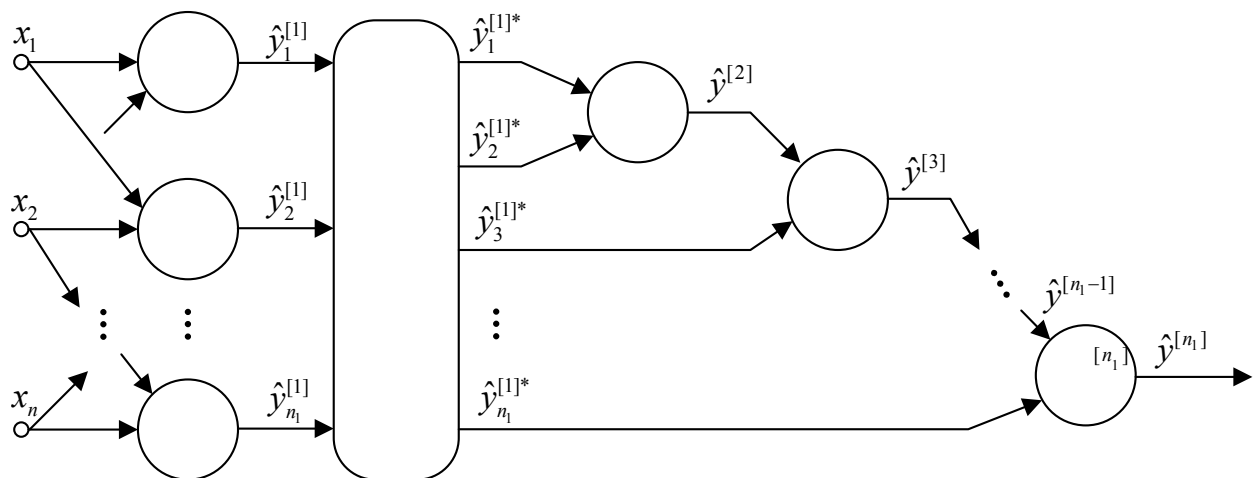


Рисунок 1.8 - Еволюційна каскадна МГУА-нейронна мережа

Перший прихований шар цієї мережі аналогічний першому шару мережі, наведеної на рис.1.7, з тою лише різницею, що блок селекції  $SB$  робить упорядкування всіх сигналів  $\hat{y}_i^{[1]}$  так, що  $\sigma_{\hat{y}_1^{[1]*}}^2 < \sigma_{\hat{y}_2^{[1]*}}^2 < \dots < \sigma_{\hat{y}_{n_1}^{[1]*}}^2$ .

Виходи першого шару (каскаду)  $\hat{y}_1^{[1]*}$  і  $\hat{y}_2^{[1]*}$  далі надходять на єдиний нейрон другого каскаду  $CR - N^{[2]}$ , що формує сигнал  $\hat{y}^{[2]}$ , який далі в третьому

селекції  $\hat{y}_3^{[1]*}$ . Процес нарощування каскадів триває до досягнення необхідної точності, при цьому загальне число нейронів такої мережі визначається значенням  $2n_1 - 1$ . У випадку, якщо точність, що досягається, незадовільна, можна підвищити її, налаштовуючи не тільки синаптичні ваги, але й центри й параметри ширини складених нейронів. Для цього може бути використана або звичайна градієнтна процедура мінімізації критерію навчання (1.23) по вектору центрів  $c_h^{ij}$  і параметру ширини  $\sigma^2$  [36] або модифікація адаптивного алгоритму навчання, що має фільтруючими властивостями [37] виду

$$\left\{ \begin{aligned} c_h^{ij}(k) &= c_h^{ij}(k-1) - \eta_c(k) \nabla_{c_h^{ij}} E_l(k) = \\ &= c_h^{ij}(k-1) + \eta_c(k) e_l(k) w_{lh}^{ij} \varphi_h^{ij}(x(k)) \sigma^{-2}(k-1) (x^{ij}(k) - c_h^{ij}(k-1)) = \\ &= c_h^{ij}(k-1) + \eta_c(k) e_l(k) J_{c_h^{ij}}(k), \\ \eta_c(k) &= \left( \alpha_c \eta_c^{-1}(k-1) + \|J_{c_h^{ij}}(k)\|^2 \right)^{-1}, \quad 0 \leq \alpha_c \leq 1, \end{aligned} \right. \quad (1.25)$$

$$\left\{ \begin{aligned} \sigma^{-2}(k) &= \sigma^{-2}(k-1) - \eta_\sigma(k) \frac{\partial E_l(k)}{\partial \sigma^{-2}} = \\ &= \sigma^{-2}(k-1) - \eta_\sigma(k) e_l(k) \sum_{h=1}^9 w_{lh}^{ij} \varphi_h^{ij} \|x(k) - c_h^{ij}(k-1)\|^2 = \\ &= \sigma^{-2}(k-1) - \eta_\sigma(k) e_l(k) J_{\sigma^{-2}}(k), \\ \eta_\sigma(k) &= \left( \alpha_\sigma \eta_\sigma^{-1}(k-1) + J_{\sigma^{-2}}^2(k) \right)^{-1}, \quad 0 \leq \alpha_\sigma \leq 1, \end{aligned} \right. \quad (1.26)$$

де

$$E_l(k) = e_l^2(k) = (y(k) - \hat{y}_l(k))^2 = \left( y(k) - w_{l0}^{ij} - \sum_{h=1}^9 w_{lh}^{ij} \varphi_h^{ij}(x^{ij}(k)) \right)^2 \quad (1.27)$$

- локальний критерій навчання.

Природно, що після настроювання центрів і параметрів ширини, необхідно перерахувати синаптичні ваги, згідно з вираженням (1.24). Якщо ж з якихось причин таке настроювання небажане, можна спробувати поліпшити апроксимуючі властивості складових R-Нейронів, розміщаючи їх центри не у вузлах рівномірних решіток (рис.1.6), а більш ефективними способами [38].

У ситуаціях, коли швидкість навчання, що забезпечується процедурами (1.25) і (1.26) недостатня, можна спробувати в якості вузлів мережі використовувати гібриди Q-нейронів і R-нейронів, що забезпечують необхідні апроксимуючі властивості.

Введемо до розгляду структуру складеного R-нейрона (ECR-N), що наведено на рис.1.9 і співпадаючу по суті з архітектурою CR-N, описаної вище.

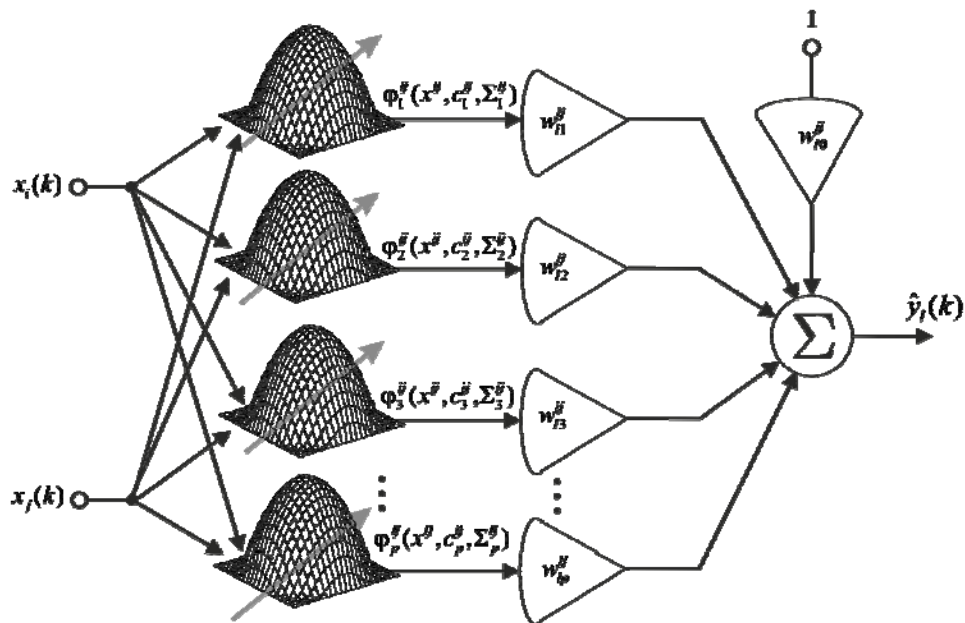


Рисунок 1.9 - Складеної R-нейрон з ядрами Єпанечнікова (ECR-N)

Основна відмінність ECR-N від CR-N полягає в тому, що в якості активаційних функцій  $\varphi_h^{ij}(x^{ij}, c_h^{ij}, \Sigma_h^{ij})$  тут використовуються багатовимірні ядра Єпанечнікова [39, 40] виду

$$\varphi_h^{ij}(x^{ij}, c_h^{ij}, \Sigma_h^{ij}) = 1 - \left\| x^{ij} - c_h^{ij} \right\|_{(\Sigma_h^{ij})}^2, \quad (1.28)$$

що мають дзвонувату форму при додатньо визначеній  $(2 \times 2)$ -матриці рецепторного поля  $\Sigma_h^{ij}$ . Перевага активаційної функції (1.28) у порівнянні із традиційними гаусіанами полягає в лінійності її похідних по всіх параметрах, що дозволяє досить просто налаштувати не тільки синаптичні ваги нейрона, але й центри з рецепторними полями.

При цьому перетворення, реалізоване ECR-N, має вигляд

$$\hat{y}_l = w_{l0}^{ij} + \sum_{h=1}^p w_{lh}^{ij} \varphi_h^{ij}(x^{ij}, c_h^{ij}, \Sigma_h^{ij}) = w_{l0}^{ij} + \sum_{h=1}^p w_{lh}^{ij} \left( 1 - \frac{\|x^{ij} - c_h^{ij}\|^2}{(\Sigma_h^{ij})^{-1}} \right).$$

Вводячи до розгляду  $(p+1) \times 1$  – вектор активаційних функцій  $\varphi^{ij}(x(k)) = (1, \varphi_1^{ij}(x^{ij}(k), c_1^{ij}, \Sigma_1^{ij}), \dots, \varphi_p^{ij}(x^{ij}(k), c_p^{ij}, \Sigma_p^{ij}))^T$  і критерій навчання (1.27), нескладно одержати оцінки синаптичних ваг у формі (1.24).

Якщо ж дані на обробку надходять послідовно в on-line режимі, замість (1.24) може бути використаний рекурентний варіант у формі

$$\begin{cases} w_l^{ij}(k) = w_l^{ij}(k-1) + \frac{P_{ij}(k-1)(y(k) - (w_l^{ij}(k-1))^T \varphi^{ij}(x(k)))}{1 + (\varphi^{ij}(k))^T P_{ij}(k-1) \varphi^{ij}(k)} \varphi^{ij}(k), \\ P_{ij}(k) = P_{ij}(k-1) - \frac{P_{ij}(k-1) \varphi^{ij}(x(k)) (\varphi^{ij}(x(k)))^T P_{ij}(k-1)}{1 + (\varphi^{ij}(x(k)))^T P_{ij}(k-1) \varphi^{ij}(x(k))}, \\ P_{ij}(0) = \gamma I, \quad \gamma \gg 0 \end{cases} \quad (1.29)$$

Алгоритми (1.25), (1.29) ефективні тільки в ситуаціях, коли шуканий розв'язок є стаціонарним, тобто оптимальні значення синаптичних ваг не міняються в часі. Оскільки в багатьох практичних задачах це не так, наприклад, адаптивний ідентифікація нестационарних об'єктів або прогнозування нестационарних часових рядів, може бути використаний швидкодіючий

адаптивних алгоритм навчання, що має як фільтруючі, так і стежачі властивості [41]:

$$\begin{cases} w_i^{ij}(k) = w_i^{ij}(k-1) + \eta_w(k)(y(k) - (w_i^{ij}(k-1))^T \varphi^{ij}(x(k)))\varphi^{ij}(x(k)) = \\ = w_i^{ij}(k-1) + \eta_w(k)e_l(k)\varphi^{ij}(x(k)), \\ \eta_w^{-1}(k) = r_w(k) = \alpha r_w(k-1) + \|\varphi^{ij}(x(k))\|^2, \quad 0 \leq \alpha \leq 1, \end{cases}$$

де  $\alpha$  – параметр згладжування, що задає компроміс між фільтруючими та слідкуючими властивостями.

З метою поліпшення апроксимуючих властивостей ECR-N нейрона можна налаштовувати не тільки синаптичні ваги, але й центри з рецепторними полями. При цьому, завдяки використанню ядер Єпанечнікова, алгоритми навчання мають досить простий вид.

Використовуючи градієнтну процедуру мінімізації критерію (1.27) і техніку її оптимізації по швидкодії [42], приходимо до алгоритму навчання ECR-нейрона виду

$$\begin{cases} w_i^{ij}(k) = w_i^{ij}(k-1) + \eta_w(k)e_l(k)\varphi^{ij}(x(k)), \\ \eta_w^{-1}(k) = r_w(k) = \alpha r_w(k-1) + \|\varphi^{ij}(x(k))\|^2, \\ c_h^{ij}(k) = c_h^{ij}(k-1) + \eta_c(k)e_l(k)w_i^{ij}(k)(\Sigma_h^{ij}(k-1))^{-1}(x^{ij}(k) - c_h^{ij}(k-1)) = \\ = c_h^{ij}(k-1) + \eta_c(k)e_l(k)g_h(k), \\ \eta_c^{-1}(k) = r_c(k) = \alpha r_c(k-1) + \|g_h(k)\|^2, \\ (\Sigma_h^{ij}(k))^{-1} = (\Sigma_h^{ij}(k-1))^{-1} - \eta_\Sigma(k)e_l(k)w_i^{ij}(k)(x^{ij}(k) - c_h^{ij}(k))(x^{ij}(k) - c_h^{ij}(k))^T = \\ = (\Sigma_h^{ij}(k-1))^{-1} - \eta_\Sigma(k)e_l(k)G_h(k), \\ \eta_\Sigma^{-1}(k) = \Gamma_\Sigma(k) = \alpha \Gamma_\Sigma(k-1) + TrG_h(k)G_h^T(k). \end{cases} \quad (1.30)$$



### 1.3 МГУА-нейронна мережа на основі адаптивних вейвлетів ( W-нейронів) та оптимальний алгоритми її навчання

Поряд з нейронними мережами для обробки сигналів різної природи останнім часом достатнє часто використовується теорія вейвлет-перетворення [43-45], що забезпечує компактний локальний опис сигналів як у частотній, так і в часовій області. На стику теорій ШНМ і вейвлетів виникли вейвлет-нейронні мережі [25, 46-49], що підтвердили свою ефективність у задачах обробки нестационарних нелінійних сигналів і процесів. Елементарними вузлами вейвлет-нейронних мереж є, так звані, радіальні вейвлони [50], активаційними функціями яких є парні вейвлети з аргументом у вигляді евклідової відстані між  $\vec{x}(k)$  і центром вейвлета  $c_{lh}^{ij}$ , при цьому кожна компонента відстані  $\|\vec{x}(k) - c_{lh}^{ij}\|^2$  нормується на параметр ширини  $(\sigma_{lh}^{ij})^2$ . Рецепторними полями таких вейвлонов є гіпереліпсоїди з осями колінеарними координатним осям простору  $X$ .

Цей підрозділ присвячено питанням синтезу складеного адаптивного вейвлону і його алгоритму навчання, що має підвищену швидкість збіжності, що й забезпечує покращені апроксимуючі властивості, для використання як вузла нейронної мережі, зокрема, як нейрона МГУА-нейронної мережі [19, 30, 31, 51, 52].

Введемо до розгляду структуру адаптивного вейвлону [37], , що наведено на рис. 1.10 і співпадаючу по суті зі спрощеною архітектурою традиційної радіально-базисної нейронної мережі із двома входами.

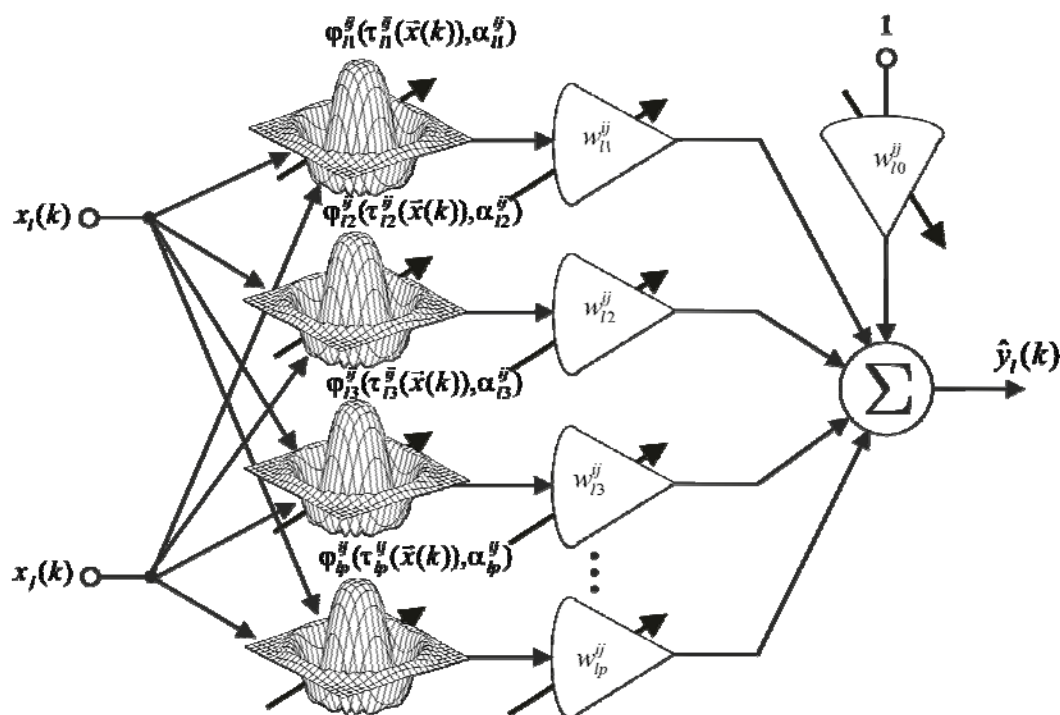


Рисунок 1.10 - Структура адаптивного вейвлону

Нульовий шар структури є рецепторним і в поточний момент часу  $k$  на нього подається вхідний сигнал у формі двовимірного вектора  $x^{ij}(k) = (x_i(k), x_j(k))^T$ . Прихований шар на відміну від радіально-базисної мережі утворений не  $R$ -нейронами, а вейвлонами з активаційними функціями-вейвлетами виду

$$\varphi_{ih}^j(\vec{x}(k)) = \varphi_{ih}^j \left( (\vec{x}(k) - c_{ih}^j(k))^T (Q_{ih}^j(k))^{-1} (\vec{x}(k) - c_{ih}^j(k)) \right), \quad h = 1, 2, \dots, p, \quad (1.31)$$

у яких замість скалярних параметрів розтягань використовується матриця  $(Q_{ih}^j)^{-1}$ , тобто використовується не евклідова метрика, а відстань Ітакури-Саїто [53]. Це приводить до того, що рецепторні поля вейвлонів (1.2) можуть приймати довільну орієнтацію щодо координатних осей простору  $X$ , що розширює функціональні можливості адаптивного вейвлону.

Ґрунтуючись на вейвлет-функції «Mexican Hat», введемо нову активаційну функцію [54] у структуру адаптивного вейвлону, яка буде налаштовуватися в процесі навчання мережі й має вигляд

$$\varphi_{lh}^{ij}(\vec{x}(k)) = (1 - \alpha_{lh}^{ij}(k) (\tau_{lh}^{ij}(x^{ij}(k)))^2) \exp\left(-(\tau_{lh}^{ij}(x^{ij}(k)))^2 / 2\right), \quad (1.32)$$

де  $\tau_{lh}^{ij}(x^{ij}(k)) = \left( (x^{ij}(k) - c_{lh}^{ij}(k))^T (Q_{lh}^{ij}(k))^{-1} (x^{ij}(k) - c_{lh}^{ij}(k)) \right)$ ,  $\alpha_{lh}^{ij}$  - параметр, що наструюється  $0 \leq \alpha_{lh}^{ij} \leq 1$  ( ).

Параметр  $\alpha_{lh}^{ij}$  дозволяє підлаштувати форму активаційної функції в процесі навчання адаптивного вейвлону так, що при  $\alpha_{lh}^{ij} = 0$  одержуємо Гаусову функцію активації, при  $\alpha_{lh}^{ij} = 1$  одержуємо вейвлет-функцію «Mexican Hat» і при  $0 < \alpha_{lh}^{ij} < 1$  - гібридну функцію активації.

На рис. 1.11 наведені форми активаційних функцій вейвлонів (1.32) при різних  $Q_{lh}^{ij}$  і параметрі  $\alpha_{lh}^{ij}$ .

І, нарешті, вихід адаптивного вейвлону утворено звичайним адаптивним лінійним асоціатором із синаптичними вагами  $w_{lh}^{ij}$ , що наструюються

$$\begin{aligned} \hat{y}_l(k) &= w_{l0}^{ij} + \sum_{h=1}^p w_{lh}^{ij} \varphi_l^{ij} \left( (x^{ij}(k) - c_{lh}^{ij}(k)) (Q_{lh}^{ij}(k))^{-1} (x^{ij}(k) - c_{lh}^{ij}(k)) \right) = \\ &= (w_l^{ij})^T \varphi_l^{ij}(x^{ij}(k)), \end{aligned} \quad (1.33)$$

де

$$\varphi_{l0}^{ij}(x^{ij}(k)) \equiv 1, w_l^{ij} = (w_{l0}^{ij}, w_{l1}^{ij}, w_{l2}^{ij}, \dots, w_{lp}^{ij})^T,$$

$$\varphi_l^{ij}(x^{ij}(k)) = (1, \varphi_{l1}^{ij}(x^{ij}(k)), \dots, \varphi_{lp}^{ij}(x^{ij}(k)))^T.$$

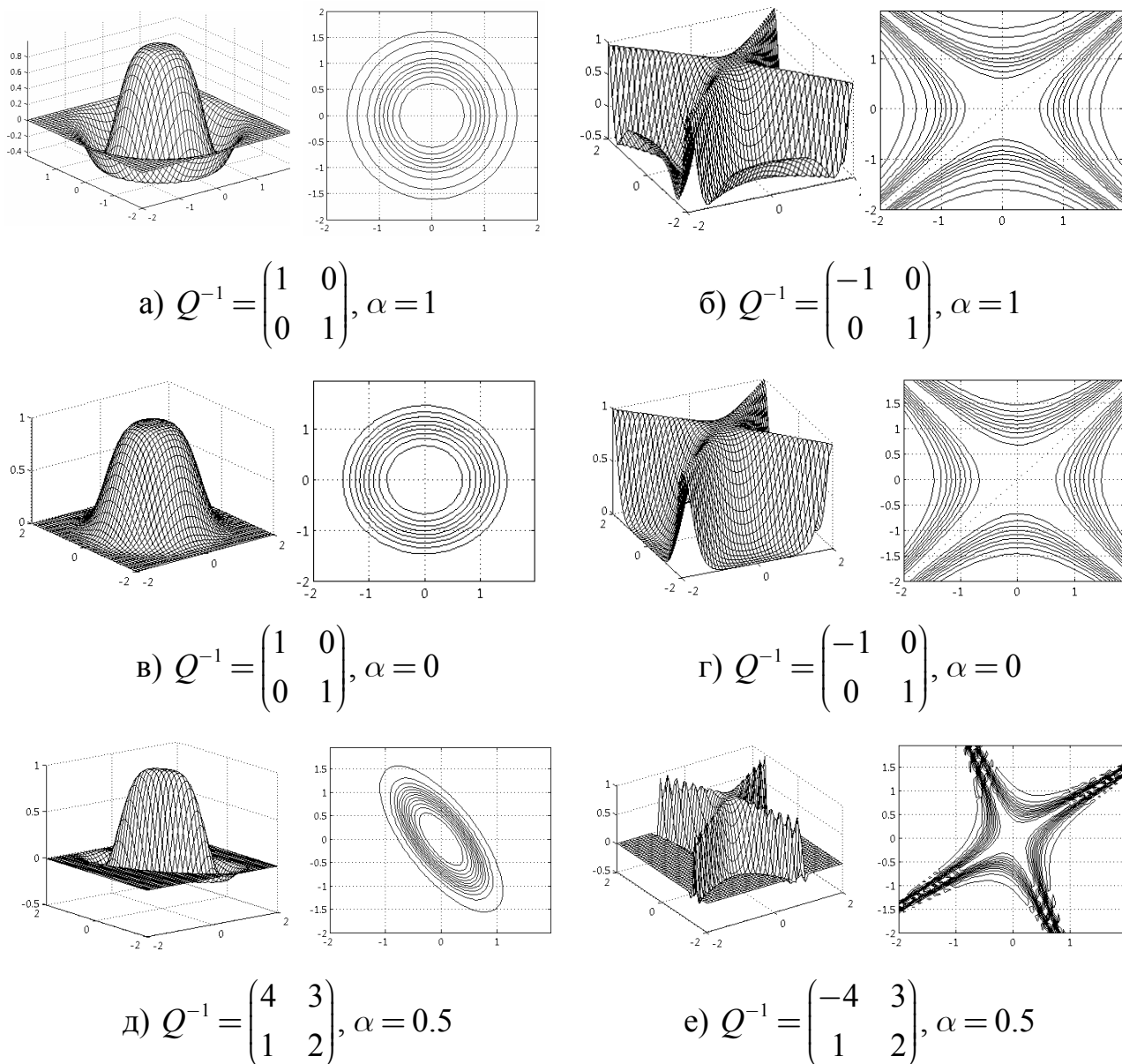


Рисунок 1.11 - Активаційні функції вейвлонів при

різних матрицях  $Q_{lh}^{ij}$  і параметрі  $\alpha_{lh}^{ij}$ .

Параметрами адаптивного вейвлону, що підлягають налаштуванню в процесі навчання, є  $p + 1$  синаптичних ваг  $w_{lh}^{ij}$ ,  $p$  ( $n \times 1$ )-векторів  $c_{lh}^{ij} = [c_{lh}^i, c_{lh}^j]$ ,  $p$  ( $n \times n$ )-матриць  $(Q_{lh}^{ij})^{-1}$  та  $p$  параметрів  $\alpha_{lh}^{ij}$ . Усього ж така структура вейвлону містить  $p(2 + n + n^2) + 1$  параметрів, що уточнюються (якщо  $n = 2$ , то  $8p + 1$  параметрів, що уточнюються).

Для настроювання параметрів вейвлонів (синаптичних ваг  $w_{lh}^{ij}$ , векторів  $c_{lh}^{ij}$ , матриць  $(Q_{lh}^{ij})^{-1}$ , параметрів  $\alpha_i^{ij}$ ) будемо використовувати градієнтну мінімізацію локального критерію навчання.

У загальному випадку алгоритм навчання може бути записаний у вигляді

$$\begin{cases} w_i^{ij}(k) = w_i^{ij}(k-1) - \eta_{w_i^{ij}} \nabla_{w_i^{ij}} E(k), \\ c_{lh}^{ij}(k) = c_{lh}^{ij}(k-1) - \eta_{c_{lh}^{ij}} \nabla_{c_{lh}^{ij}} E(k), \quad h = 1, 2, \dots, p, \\ (Q_{lh}^{ij}(k))^{-1} = (Q_{lh}^{ij}(k-1))^{-1} - \eta_{Q_{lh}^{ij}} \left\{ \partial E(k) / \partial (Q_{lh}^{ij})^{-1} \right\}, \quad h = 1, 2, \dots, p, \\ \alpha_i^{ij}(k) = \alpha_i^{ij}(k-1) - \eta_{\alpha_i^{ij}} \nabla_{\alpha_i^{ij}} E(k), \end{cases} \quad (1.34)$$

де  $\nabla_{w_i^{ij}} E$ ,  $\nabla_{c_{lh}^{ij}} E$  і  $\nabla_{\alpha_i^{ij}} E$  -  $(n \times 1)$ - вектори-градієнти критерію (1.5) по  $w_i^{ij}$ ,  $c_{lh}^{ij}$  і  $\alpha_i^{ij}$  відповідно;  $\left\{ \partial E(k) / \partial (Q_{lh}^{ij})^{-1} \right\}$  -  $(n \times n)$ -матриця, що утворена частинними похідними  $E$  по компонентах  $(Q_{lh}^{ij})^{-1}$ ;  $\eta_{w_i^{ij}}$ ,  $\eta_{c_{lh}^{ij}}$ ,  $\eta_{(Q_{lh}^{ij})^{-1}}$  і  $\eta_{\alpha_i^{ij}}$  - параметри кроку алгоритму навчання.

Для введеної функції-активації (1.32) можна записати

$$\left\{ \begin{aligned}
& \nabla_{w_l^{ij}} E(k) = -e_l(k) \left( 1 - \alpha_l^{ij}(k) \odot (\tau_l^{ij}(\vec{x}(k)))^2 \right) \odot \exp\left(-(\tau_l^{ij}(\vec{x}(k)))^2 / 2\right) = \\
& = -e_l(k) J_{w_l^{ij}}^l(k), \\
& \nabla_{c_{lh}^{ij}} E(k) = 2e_l(k) w_{lh}^{ij}(k) (Q_{lh}^{ij}(k))^{-1} (x^{ij}(k) - c_{lh}^{ij}(k)) \cdot \\
& \cdot \left( \alpha_{lh}^{ij}(k) (\tau_{lh}^{ij}(x^{ij}(k)))^3 - (2\alpha_{lh}^{ij}(k) + 1) \tau_{lh}^{ij}(x^{ij}(k)) \right) \exp\left(-(\tau_{lh}^{ij}(x^{ij}(k)))^2 / 2\right) = \\
& = e_l(k) J_{c_{lh}^{ij}}^l(k), \\
& \left\{ \partial E(k) / \partial (Q_{lh}^{ij}(k))^{-1} \right\} = e_l(k) w_{lh}^{ij}(k) (x^{ij}(k) - c_{lh}^{ij}(k)) (x^{ij}(k) - c_{lh}^{ij}(k))^T \cdot \\
& \cdot \left( \alpha_{lh}^{ij}(k) (\tau_{lh}^{ij}(x^{ij}(k)))^3 - (2\alpha_{lh}^{ij}(k) + 1) \tau_{lh}^{ij}(x^{ij}(k)) \right) \exp\left(-(\tau_{lh}^{ij}(x^{ij}(k)))^2 / 2\right) = \\
& = -e_l(k) J_{(Q_{lh}^{ij})^{-1}}^l(k), \\
& \nabla_{\alpha_l^{ij}} E(k) = -e_l(k) w_l^{ij}(k) \odot (\tau_l^{ij}(x^{ij}(k)))^2 \odot \exp\left(-(\tau_{lh}^{ij}(x^{ij}(k)))^2 / 2\right) = \\
& = -e_l(k) J_{\alpha_l^{ij}}^l(k).
\end{aligned} \right. \quad (1.35)$$

Підвищення швидкості збіжності може бути досягнуте шляхом використання більш складних, ніж градієнтні, процедур типу Хартлі або Марквардта. Скориставшись лемою обернення матриць та після досить простих перетворень, можна одержати простий і ефективний алгоритм навчання параметрів вейвлонів у вигляді

$$\left\{ \begin{aligned}
& w_l^{ij}(k) = w_l^{ij}(k-1) + \lambda_{w_l^{ij}} \frac{e_l(k) J_{w_l^{ij}}^l(k)}{\eta_{w_l^{ij}} + \|J_{w_l^{ij}}^l(k)\|^2}, \\
& c_{lh}^{ij}(k) = c_{lh}^{ij}(k-1) - \lambda_{c_{lh}^{ij}} \frac{e_l(k) J_{c_{lh}^{ij}}^l(k)}{\eta_{c_{lh}^{ij}} + \|J_{c_{lh}^{ij}}^l(k)\|^2}, \\
& \alpha_l^{ij}(k) = \alpha_l^{ij}(k-1) + \lambda_{\alpha_l^{ij}} \frac{e_l(k) J_{\alpha_l^{ij}}^l(k)}{\eta_{\alpha_l^{ij}} + \|J_{\alpha_l^{ij}}^l(k)\|^2},
\end{aligned} \right. \quad (1.36)$$

що з точністю до позначень збігається з оптимальним (при  $\lambda_{w_i^{jj}} = \lambda_{c_{ih}^{jj}} = \lambda_{\alpha_i^{jj}} = 1, \eta_{w_i^{jj}} = \eta_{c_{ih}^{jj}} = \eta_{\alpha_i^{jj}} = 0$ ) по швидкодії алгоритмом Качмажа-Уїдрю-Гоффа.

Для настроювання матриць  $(Q_{lh}^{jj})^{-1}$  можна скористатися матричною модифікацією алгоритму (1.36) у вигляді [37]

$$(Q_{lh}^{jj}(k))^{-1} = (Q_{lh}^{jj}(k-1))^{-1} + \lambda_{(Q_{lh}^{jj})^{-1}} \frac{e_l(k) J_{(Q_{lh}^{jj})^{-1}}^l(k)}{\eta_{(Q_{lh}^{jj})^{-1}} + \text{Tr}(J_{(Q_{lh}^{jj})^{-1}}^{lT}(k) J_{(Q_{lh}^{jj})^{-1}}^l(k))}, \quad (1.37)$$

де  $\lambda_{(Q_{lh}^{jj})^{-1}}, \eta_{(Q_{lh}^{jj})^{-1}}$  мають такі самі значення, що й відповідні параметри в (1.36).

Таким чином, остаточно алгоритм навчання параметрів вейвлонів прихованого шару в оптимальному за швидкодією варіанті може бути записаний у вигляді

$$\begin{cases} w_i^{jj}(k) = w_i^{jj}(k-1) + \left\| J_{w_i^{jj}}^l(k) \right\|^{-2} e_l(k) J_{w_i^{jj}}^l(k), \\ c_{ih}^{jj}(k) = c_{ih}^{jj}(k-1) - \left\| J_{c_{ih}^{jj}}^l(k) \right\|^{-2} e_l(k) J_{c_{ih}^{jj}}^l(k), h=1,2,\dots,p, \\ (Q_{lh}^{jj}(k))^{-1} = (Q_{lh}^{jj}(k-1))^{-1} + \left( \text{Tr}(J_{(Q_{lh}^{jj})^{-1}}^{lT}(k) J_{(Q_{lh}^{jj})^{-1}}^l(k)) \right)^{-1} e_l(k) J_{(Q_{lh}^{jj})^{-1}}^l(k), h=1,2,\dots,p, \\ \alpha_i^{jj}(k) = \alpha_i^{jj}(k-1) + \left\| J_{\alpha_i^{jj}}^l(k) \right\|^{-2} e_l(k) J_{\alpha_i^{jj}}^l(k). \end{cases} \quad (1.38)$$

Відомо, що однокрокові алгоритми типу Качмажа-Уїдрю-Гоффа, забезпечують високу швидкодію, але не мають фільтруючих властивостей, тобто погано працюють за умов інтенсивних збурень і завад. Для додання процесу навчання згладжуючих властивостей можна ввести процедуру навчання вигляду

$$\left\{ \begin{array}{l}
w_l^{ij}(k) = w_l^{ij}(k-1) + \lambda_{w_l^{ij}} \frac{e_l(k) J_{w_l^{ij}}^l(k)}{\eta_{w_l^{ij}}^l(k)}, \quad \eta_{w_l^{ij}}^l(k) = \gamma_{w_l^{ij}} \eta_{w_l^{ij}}^l(k-1) + \left\| J_{w_l^{ij}}^l(k) \right\|^2, \\
c_{lh}^{ij}(k) = c_{lh}^{ij}(k-1) - \lambda_{c_{lh}^{ij}} \frac{e_l(k) J_{c_{lh}^{ij}}^l(k)}{\eta_{c_{lh}^{ij}}^l(k)}, \quad \eta_{c_{lh}^{ij}}^l(k) = \gamma_{c_{lh}^{ij}} \eta_{c_{lh}^{ij}}^l(k-1) + \left\| J_{c_{lh}^{ij}}^l(k) \right\|^2, \\
(Q_{lh}^{ij}(k))^{-1} = (Q_{lh}^{ij}(k-1))^{-1} + \lambda_{(Q_{lh}^{ij})^{-1}} \frac{e_l(k) J_{(Q_{lh}^{ij})^{-1}}^l(k)}{\eta_{(Q_{lh}^{ij})^{-1}}^l(k)}, \quad \eta_{(Q_{lh}^{ij})^{-1}}^l(k) = \gamma_{(Q_{lh}^{ij})^{-1}} \eta_{(Q_{lh}^{ij})^{-1}}^l(k-1) + \\
+ Tr \left[ J_{(Q_{lh}^{ij})^{-1}}^{lT}(k) J_{(Q_{lh}^{ij})^{-1}}^l(k) \right], \\
\alpha_l^{ij}(k) = \alpha_l^{ij}(k-1) + \lambda_{\alpha_l^{ij}} \frac{e_l(k) J_{\alpha_l^{ij}}^l(k)}{\eta_{\alpha_l^{ij}}^l(k)}, \quad \eta_{\alpha_l^{ij}}^l(k) = \gamma_{\alpha_l^{ij}} \eta_{\alpha_l^{ij}}^l(k-1) + \left\| J_{\alpha_l^{ij}}^l(k) \right\|^2.
\end{array} \right. \quad (1.39)$$

(тут  $0 \leq \gamma_{i,j} \leq 1$  - параметри зважування застарілої інформації), що є нелінійним гібридом алгоритму Качмажа-Уїдроу-Гоффа та Гудвіна-Ремеджа-Кейнеса, що має як слідкуючи, так і фільтруючи властивості.

Досвід показує, що методи навчання та ідентифікації, засновані на критерії найменших квадратів, виявляються надзвичайно чутливими до відхилень фактичного закону розподілу даних від нормального. В умовах різного роду викидів, грубих помилок, негаусівських збурювань із «важкими хвостами» алгоритми, пов'язані із критерієм найменших квадратів, втрачають свою ефективність.

У цій ситуації на перший план виходять методи робастного оцінювання [55], які до теперішнього часу одержали широке поширення й для навчання штучних нейронних мереж [56, 57].

Уводячи в розгляд помилку навчання

$$e_l(k) = y_l(k) - \hat{y}_l(k) = y_l(k) - w_l^{iT}(k) \varphi_l^i(k) \quad (1.40)$$

і робастний критерій ідентифікації [56, 57]



$$E(k) = f(k) = \beta^2 \ln(\cosh(e_i(k)/\beta)), \quad (1.41)$$

де  $\beta$  - додатний параметр, обраний з емпіричних міркувань і визначальний розмір зони нечутливості до викидів, як показано на рис. 1.12.

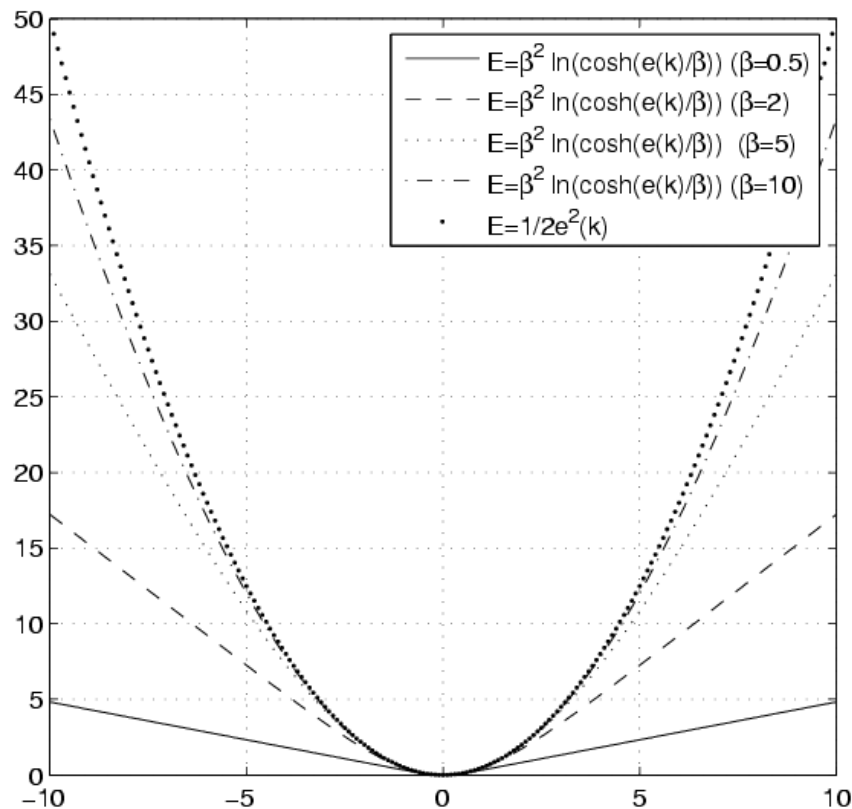


Рисунок 1.12 – Робастний критерій якості с різними параметрами  $\beta$

Необхідно відзначити, що робастний критерій (1.41) задовольняє всім аксіомам метричного простору.

Далі розглянемо процес синтезу алгоритму навчання. Для настроювання вектора синаптичних ваг  $w_i^{jj}$  і параметрів вейвлонів (векторів  $c_{lh}^{ij} = [c_{lh}^i, c_{lh}^j]$  і матриць  $(Q_{lh}^{ij})^{-1}$ ) будемо використовувати градієнтну процедуру мінімізації критерію (1.41), при цьому на відміну від покомпонентного навчання, будемо робити уточнення у векторно-матричній формі, що, по-перше, простіше з обчислювальної точки зору, а, по-друге, дозволяє оптимізувати процес

навчання по швидкодії.

У загальному випадку алгоритм навчання може бути записаний у вигляді (1.41).

Очевидно, що для адаптивного багатовимірного вейвлета (1.32) можна записати

$$\left\{ \begin{aligned}
 \nabla_{w_l^{ij}} E(k) &= -\beta \tanh(e_l(k)/\beta) \left(1 - \alpha_l^{ij}(k) \odot (\tau_l^{ij}(\vec{x}(k)))^2\right) \odot \exp\left(-(\tau_l^{ij}(\vec{x}(k)))^2/2\right) = \\
 &= -e_l(k) J_{w_l^{ij}}^l(k), \\
 \nabla_{c_{lh}^{ij}} E(k) &= 2\beta \tanh(e_l(k)/\beta) w_{lh}^{ij}(k) (Q_{lh}^{ij}(k))^{-1} \cdot (\vec{x}(k) - c_{lh}^{ij}(k)) \cdot \\
 &\cdot \left(\alpha_{lh}^{ij}(k) (\tau_{lh}^{ij}(\vec{x}(k)))^3 - (2\alpha_{lh}^{ij}(k) + 1) \tau_{lh}^{ij}(\vec{x}(k))\right) \exp\left(-(\tau_{lh}^{ij}(x(k)))^2/2\right) = e_l(k) J_{c_{lh}^{ij}}^l(k), \\
 \left\{ \partial E(k) / \partial (Q_{lh}^{ij}(k))^{-1} \right\} &= \beta \tanh(e_l(k)/\beta) w_{lh}^{ij}(k) (\vec{x}(k) - c_{lh}^{ij}(k)) (\vec{x}(k) - c_{lh}^{ij}(k))^T \cdot \\
 &\cdot \left(\alpha_{lh}^{ij}(k) (\tau_{lh}^{ij}(\vec{x}(k)))^3 - (2\alpha_{lh}^{ij}(k) + 1) \tau_{lh}^{ij}(\vec{x}(k))\right) \exp\left(-(\tau_{lh}^{ij}(x(k)))^2/2\right) = \\
 &= -e_l(k) J_{(Q_{lh}^{ij})^{-1}}^l(k), \\
 \nabla_{\alpha_l^{ij}} E(k) &= -\beta \tanh(e_l(k)/\beta) w_l^{ij}(k) \odot (\tau_l^{ij}(x(k)))^2 \odot \exp\left(-(\tau_{lh}^{ij}(x(k)))^2/2\right) = \\
 &= -e_l(k) J_{\alpha_l^{ij}}^l(k).
 \end{aligned} \right. \quad (1.42)$$

Помітимо також, що швидкість збіжності до оптимальних значень  $w_l^{ij}$ ,  $c_{lh}^{ij}$ ,  $(Q_{lh}^{ij})^{-1}$  і  $\alpha_l^{ij}$  повністю визначається параметрами кроку  $\eta_{w_l^{ij}}$ ,  $\eta_{c_{lh}^{ij}}$ ,  $\eta_{(Q_{lh}^{ij})^{-1}}$  і  $\eta_{\alpha_l^{ij}}$ . Підвищення швидкості збіжності може бути досягнуте шляхом використання модифікованої процедури навчання [58]:

$$\left\{ \begin{array}{l}
w_l^{ij}(k) = w_l^{ij}(k-1) + \lambda_{w_l^{ij}} \frac{\beta \tanh(e_l(k) / \beta) J_{w_l^{ij}}^l(k)}{\eta_{w_l^{ij}}^l(k)}, \\
\eta_{w_l^{ij}}^l(k) = \gamma_{w_l^{ij}} \eta_{w_l^{ij}}^l(k-1) + \left\| J_{w_l^{ij}}^l(k) \right\|^2, \\
c_{lh}^{ij}(k) = c_{lh}^{ij}(k-1) - \lambda_{c_{lh}^{ij}} \frac{\beta \tanh(e_l(k) / \beta) J_{c_{lh}^{ij}}^l(k)}{\eta_{c_{lh}^{ij}}^l(k)}, \\
\eta_{c_{lh}^{ij}}^l(k) = \gamma_{c_{lh}^{ij}} \eta_{c_{lh}^{ij}}^l(k-1) + \left\| J_{c_{lh}^{ij}}^l(k) \right\|^2, \\
(Q_{lh}^{ij})^{-1}(k) = (Q_{lh}^{ij})^{-1}(k-1) + \lambda_{(Q_{lh}^{ij})^{-1}} \frac{\beta \tanh(e_l(k) / \beta) J_{(Q_{lh}^{ij})^{-1}}^l(k)}{\eta_{(Q_{lh}^{ij})^{-1}}^l(k)}, \\
\eta_{(Q_{lh}^{ij})^{-1}}^l(k) = \gamma_{(Q_{lh}^{ij})^{-1}} \eta_{(Q_{lh}^{ij})^{-1}}^l(k-1) + Tr \left( J_{(Q_{lh}^{ij})^{-1}}^{lT}(k) J_{(Q_{lh}^{ij})^{-1}}^l(k) \right), \\
\alpha_l^{ij}(k) = \alpha_l^{ij}(k-1) + \lambda_{\alpha_l^{ij}} \frac{\beta \tanh(e_l(k) / \beta) J_{\alpha_l^{ij}}^l(k)}{\eta_{\alpha_l^{ij}}^l(k)}, \\
\eta_{\alpha_l^{ij}}^l(k) = \gamma_{\alpha_l^{ij}} \eta_{\alpha_l^{ij}}^l(k-1) + \left\| J_{\alpha_l^{ij}}^l(k) \right\|^2.
\end{array} \right. \quad (1.43)$$

#### 1.4 МГУА-нейронна мережа на основі нейро-фаззі нейронів та оптимальні алгоритми її навчання

На поточний момент алгоритми методу групового урахування аргументів (МГУА) досить рідко використовуються для отримання оптимальних архітектур нейронних мереж. Однак такий підхід дозволяє підвищити точність нейромоделей при вирішенні різних завдань і автоматизувати процес генерації архітектури мережі. Як відомо, штучні нейронні мережі мають серйозний недолік - відсутність лінгвістичної інтерпретації отриманих результатів, внаслідок чого їх застосування в ряді випадків неможливо. Тому на сьогоднішній день найбільший інтерес із практичної точки зору являють гібридні нейро-фаззі системи, які поєднують у собі переваги двох підходів: штучних нейронних мереж і нечіткої логіки. При

цьому необхідно зазначити, що питання структурної адаптації цих систем на етапі їхнього навчання є досить актуальним.

Розглянемо спочатку нео-фаззі нейронну мережу, що має багат шарову архітектуру, з прямим розповсюдженням сигналу, і яка складається з нео-фаззі нейронів. Тришарова нео-фаззі нейронна мережа з входами і виходами представлена на рис. 1.13.

Пропонована архітектура структурно повністю збігається з архітектурою тришарового персептрона за винятком того, що в ній в якості вузлів використовуються нео-фаззі нейрони замість елементарних персептронів Розенблатта. Використання нео-фаззі нейронів робить можливим лінгвістичну інтерпретацію результатів, одержуваних за допомогою цієї мережі, що дозволяє віднести таку мережу до класу нейро-фаззі систем. Однак, вирішуючи конкретну прикладну задачу за допомогою цієї мережі, дослідник в першу чергу стикається з питанням, скільки необхідно прихованих шарів і скільки нео-фаззі нейронів необхідно помістити в кожен з них, щоб отримати прийнятний (або навіть оптимальний) результат. Для вирішення цього питання можна використовувати один з алгоритмів МГУА.

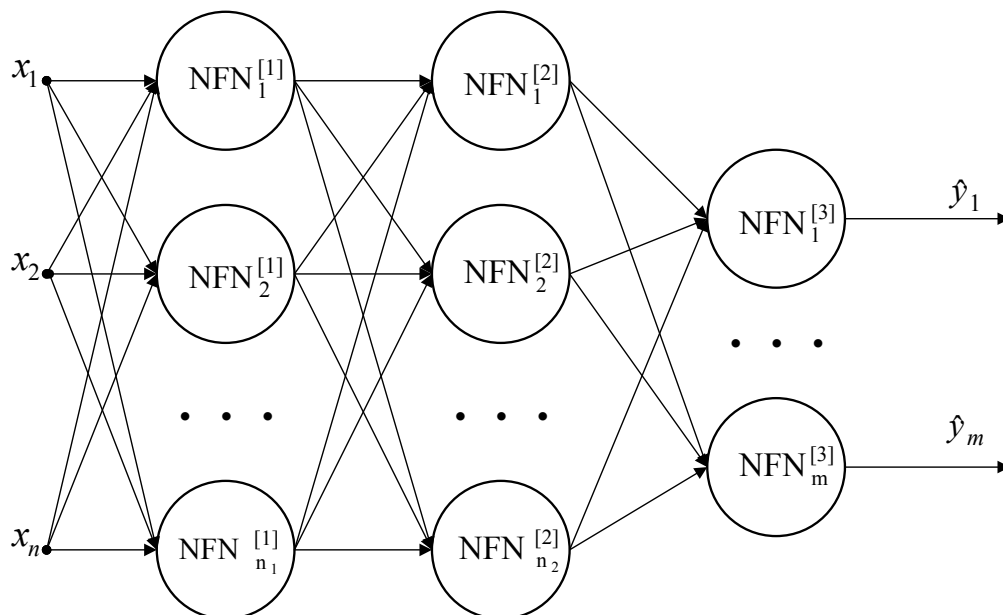


Рисунок 1.13 - Нео-фаззі нейронна мережа

Найбільшого поширення у галузі штучних нейронних мереж отримали узагальнені алгоритми МГУА: алгоритм з повним перебором всіх комбінацій трендів (комбінаторний) і алгоритм з багаторядною селекцією (багатошаровий ітераційний алгоритм). Перший з них непридатний до нейро-фаззі систем з тієї причини, що його використання при вирішенні завдань з великою розмірністю вектора вхідних аргументів може призвести до значних часових витрат на побудову нейро-нечіткої моделі. Тому для самоорганізації архітектури нео-фаззі нейронної мережі представляється доцільним використовувати багатошаровий ітераційний алгоритм.

Основна ідея багатошарового ітераційного алгоритму полягає в послідовному нарощуванні шарів у мережі до того моменту, поки зовнішній критерій якості не починає зростати. У кожному шарі мережі використовуються виключно двухвходові нео-фаззі нейрони. Процедура синтеза архітектури нео-фаззі нейронної мережі за допомогою багатошарового ітераційного алгоритму зводиться до наступного:

- формування пар виходів нео-фаззі нейронів поточного шару (на першому етапі використовується безліч вхідних сигналів). Кожна отримана пара сигналів подається на вхід відповідного нео-фаззі нейрона;
- налаштування вагових коефіцієнтів кожного нео-фаззі нейрона, використовуючи навчальну вибірку;
- обчислення зовнішнього критерію для кожного нео-фаззі нейрона, використовуючи тестову вибірку:

$$\varepsilon_p^{[s]} = \frac{1}{N_{nep}} \sum_{k=1}^{N_{nep}} \left( y(k) - \hat{y}_p^{[s]}(k) \right)^2 \quad (1.44)$$

де  $N_{nep}$  - розмір тестової вибірки;

$s$  - номер шару;

$p$  - номер нейрона в поточному шарі  $p = \overline{1, n_s}$ ;

$\hat{y}_p^{[s]}(k)$  - вихідний сигнал  $p$ -го нейрона  $s$ -го шару, отриманий в результаті подачі на вхід мережі  $i$ -го вхідного вектора;

- знаходження мінімального значення зовнішнього критерію, беручи до уваги всі нео-фаззі нейрони поточного шару:

$$\varepsilon^{[s]} = \min_p \varepsilon_p^{[s]}; \quad (1.45)$$

- перевірка умови

$$\varepsilon^{[s]} > \varepsilon^{[s-1]}, \quad (1.46)$$

де  $\varepsilon^{[s]}, \varepsilon^{[s-1]}$  - мінімальні значення зовнішнього критерію для  $s$ -го і  $(s - 1)$ -го шарів відповідно.

У разі якщо умова (1.46) виконується, то необхідно повернутися до попереднього шару і залишити один єдиний нео-фаззі нейрон з мінімальним зовнішнім критерієм (1.44), видаливши при цьому всі інші. Вихід цього нейрона буде вважатися виходом мережі. У протилежному випадку, необхідно вибрати «кращі» нейрони з мінімальними значеннями критерію (1.44) і повернутися до першого кроку процедури для додавання наступного шару нейронів;

- визначення остаточної структури нейронної мережі, послідовно просуваючись від вихідного нео-фаззі нейрона до вхідного шару, зберігаючи нейрони, виходи яких використовуються в наступних шарах, і видаляючи інші.

Після закінчення роботи алгоритму МГУА ми отримуємо оптимальну архітектуру нео-фаззі нейронної мережі для вирішення поставленого завдання. Очевидно, що результатом є не тільки оптимальна архітектура, але і нейронна мережа з налаштованими синаптичними ваговими коефіцієнтами, здатна обробляти нові вхідні дані. Крім того, необхідно відзначити, що в загальному випадку для навчання нео-фаззі нейронної мережі необхідно використовувати алгоритми засновані на зворотному поширенні помилки, оскільки ця нейро-фаззі конструкція має багат шарову архітектуру. Застосування МГУА на етапі навчання мережі дозволяє проводити налаштування синаптичних вагових

коефіцієнтів нео-фаззі нейронів, використовуючи оптимальні за швидкістю процедури, як наприклад метод найменших квадратів.

Для апробації запропонованого підходу до самоорганізації архітектури нео-фаззі нейронної мережі було вирішено задачу класифікації типу раку молочної залози.

Для цього була використана вибірка, що складається з 699 зразків. 16 образів містили параметри з пропущеними значеннями і були видалені з вибірки, а 683 образи, що залишилися, були розділені на навчальну вибірку - 478 образів (70%) і тестову вибірку - 205 образів (30%).

Кожен приклад містив 9-вимірний вектор параметрів і 1 параметр належності до класу (1 або 0), який необхідно було визначити, встановлюючи тим самим, злоякісна чи доброякісна пухлина у обстежуваній.

Для розв'язуваної задачі було введено відразу два критерії якості: 1) кількість коректно класифікованих прикладів, 2) так званий «інтервал довіри». Очевидно, що якщо на виході нейронної мережі ми маємо значення близьке до значення одного з класів, то ймовірність коректної класифікації більше, ніж у зворотному випадку. Оскільки в розглянутій задачі ми маємо тільки два класи (доброякісна пухлина - 0, злоякісна пухлина - 1), то якщо вихідний сигнал лежав у межах  $[0.3; 0.7]$ , вважалось, що такий приклад не потрапляє в довірчий інтервал.

Отримані результати з урахуванням обох критеріїв якості наведені в таблиці 1.1. Також там наведено результат отриманий за допомогою повної нео-фаззі нейронної мережі, тобто архітектури, в котрій кожний нео-фаззі нейрон попереднього каскаду зв'язаний з кожним нео-фаззі нейроном подальшого.

Архітектура синтезована багат шаровим ітераційним алгоритмом наведена на рис. 1.14.

Таблиця 1.1 - Результати класифікації

Архітектура НМ	Якість на навч. вибірці / кількість образів, що не попали в «інтервал довіри»	Якість на тест. вибірці / кількість образів, що не попали в «інтервал довіри»
Мережа синтезована за допомогою МГУА	98,4% / 20	97,2% / 12
Повна нео-фаззі нейронна мережа	98,7% / 5	91,7% / 35

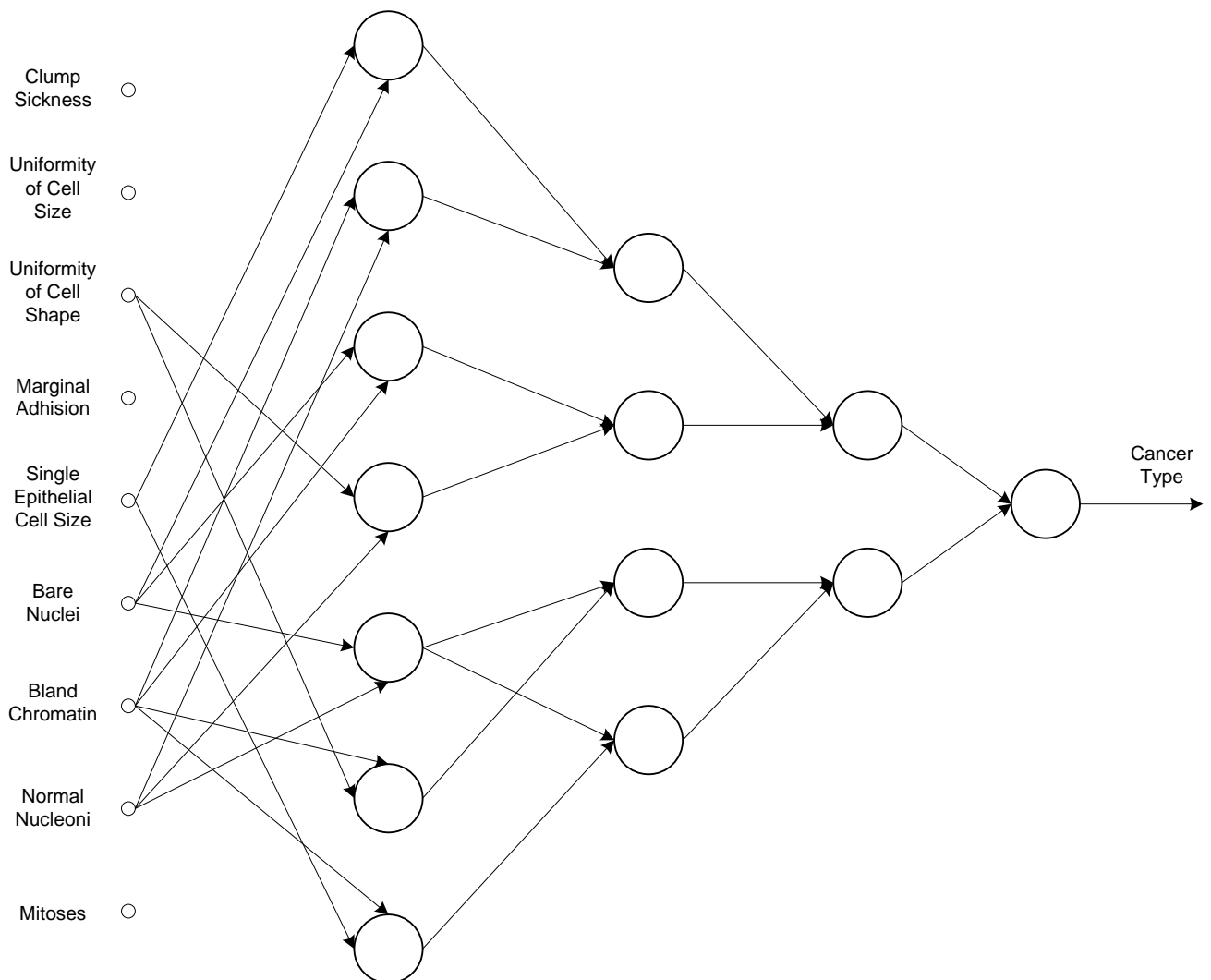


Рисунок 1.14 - Архітектура нео-фаззі нейронної мережі, що була синтезована за допомогою МГУА

З результатів очевидно, що нео-фаззі нейронна мережа з архітектурою, що була синтезована за допомогою МГУА показує доволі точні результати



класифікації і випереджає за цим показником повну нео-фаззі нейронну мережу, особливо на тестовій вибірці. Це можна пояснити тим, що повна мережа представляє собою більш складну модель а, як відомо, складність моделі призводить до втрати її узагальнюючих властивостей, внаслідок чого зменшується точність класифікації (саме на тестовій вибірці). МГУА дозволяє синтезувати оптимальну за складністю нейро-фаззі архітектуру, в якій не враховуються незначущі чи малозначущі входи, як показано на рис. 1.14. При цьому остаточна нейро-фаззі модель буде значно простішою, ніж повна мережа, але незважаючи на це, вона дозволить досягти більш високої якості класифікації.

### **1.5 Результати імітаційного експерименту з МГУА-нейронними мережами**

Ефективність запропонованого підходу була продемонстрована при розв'язку задачі емуляції нелінійного динамічного об'єкта [23], описуваного рівнянням

$$y(k+1) = 0.3y(k) + 0.6y(k-1) + f(u(k)). \quad (1.47)$$

Навчання нейронної мережі проводиться на основі вибірки, отриманої за допомогою рівняння (1.47) з керуючими сигналами  $f(u(k)) = 0.6\sin u(k) + 0.3\sin 3u(k) + 0.1\sin 5u(k)$  й  $u(k) = \sin(2\pi k / 250)$  для  $k = 1 \dots 1500$ . Після 1500 кроків навчання було зупинено. У якості перевіірочних даних емулятора використовувався динамічний об'єкт (1.47) з тими ж керуючими сигналами для  $k = 1501 \dots 2500$ , а далі в момент часу  $k = 2501$  вид керуючих сигналів змінювався на  $f(u(k)) = u^3(k) + 0.3u^2(k) - 0.4u(k)$  й  $u(k) = \sin(2\pi k / 250) + \sin(2\pi k / 250)$  для  $k = 2501 \dots 4000$ .

Для емуляції об'єкта використовувалася МГУА-нейронна мережа на Q-нейронах з початковою кількістю входів  $n_{GMDH} = 6$ . Кожний Q-нейрон навчався в пакетному режимі 10 епох. Початкові значення параметрів Q-нейрона були прийняті нульовими.

У якості критерію якості емуляції була використана середньоквадратична похибка (MSE). Результати емуляції динамічного об'єкта (1.47) наведені на рис. 1.15. Як видно, дві криві, що представляють фактичні значення (пунктирна лінія) і значення емуляції (суцільна лінія), практично ідентичні.

У таблиці 1.2 наведено порівняльний аналіз процесу емуляції на основі розширеної МГУА-нейронної мережі на Q-нейронах з кількістю входів у вузлі  $n = 3$  з такою ж архітектурою з кількістю входів у вузлі  $n = 2$  й результатами стандартної МГУА-нейронної мережі.

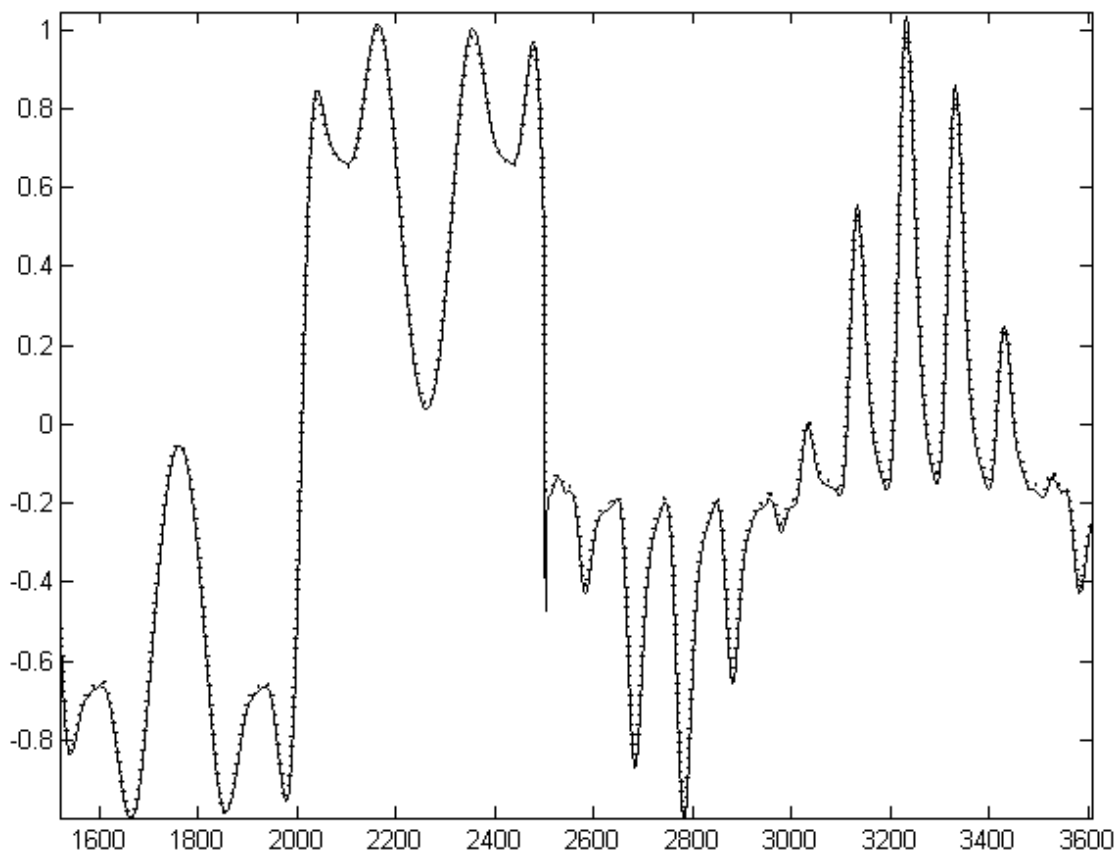


Рисунок 1.15 - Фрагмент результатів емуляції динамічного об'єкта (1.47)

Таблиця 1.2 – Результати емуляції динамічного об'єкта

Нейронна мережа / Алгоритм навчання	Кіл. шарів у МГУА-мережі	Кіл. входів	MSE
Модифікована розширена МГУА-нейронна мережа заснована на Q-Нейронах / Запропонований алгоритм навчання (22)	2	3	0.000349
Модифікована розширена МГУА-нейронна мережа заснована на Q-Нейронах / Запропонований алгоритм навчання (22)	2	2	0.000676
МГУА-нейронна мережа [59]	2	2	0.0032

Як випливає з наведених експериментальних результатів, запропонований підхід забезпечує більш високу точність емуляції в порівнянні з подібною архітектурою зі стандартним числом входів у вузлі, а також зі стандартної МГУА-нейронною мережею.

Наступний експеримент проведений на сигналі електроенцефалограми (ЕЕГ) при глибокій штучній гіпотермії на основі МГУА-нейронної мережі на адаптивному складеному вейвлоні.

МГУА-нейронна мережа мала кількість входів  $n_{GMDH} = 5$  і значення  $x(t-4), x(t-3), x(t-2), x(t-1), x(t)$  були використані для одержання прогнозу  $x(t+1)$ . Адаптивний складовий вейвлон навчався в пакетному режимі за допомогою алгоритму (1.7), (1.15) 1000 ітерація (1000 навчальних точок для  $t = 1, \dots, 1000$ ). Параметри алгоритму навчання були взяті як  $\gamma_{c_j} = \gamma_{Q_j^{-1}} = \gamma_{\alpha_j} = 0.99$  і  $\lambda_{c_j} = \lambda_{Q_j^{-1}} = \lambda_{\alpha_j} = 0.99$ . Початкові значення були  $\eta_{c_j} = \eta_{Q_j^{-1}} = \eta_{\alpha_j} = 10000$  й  $\alpha = (\alpha_1, \dots, \alpha_h) = 0.1$ . Після 1000 ітерацій навчання було

зупинен, а наступні 500 точок для  $t = 20001 \dots 20500$  використовувалися як тестові для обчислення прогнозу. Початкові значення синоптичних вестів були взяті нульовими. У якості критерію прогнозу використовувалася середньоквадратична похибка.

На рис. 1.16 показані результати прогнозування ЕЕГ сигналу. Дві кривих представляють актуальні (крапкова лінія) і прогнозні значення (суцільна лінія), як можна побачити вони збігаються.

У такий спосіб як можна побачити з результатів експериментів запропонована МГУА-нейронна мережа на адаптивних складових вейвлоних з (1.7), (1.15) маючи теж число, що настраюються параметрів показує більш краща якість прогнозування й високу швидкість навчання в порівнянні із традиційними структурами. У таблиці 1.3 показані результати прогнозування сигналу ЕЕГ.

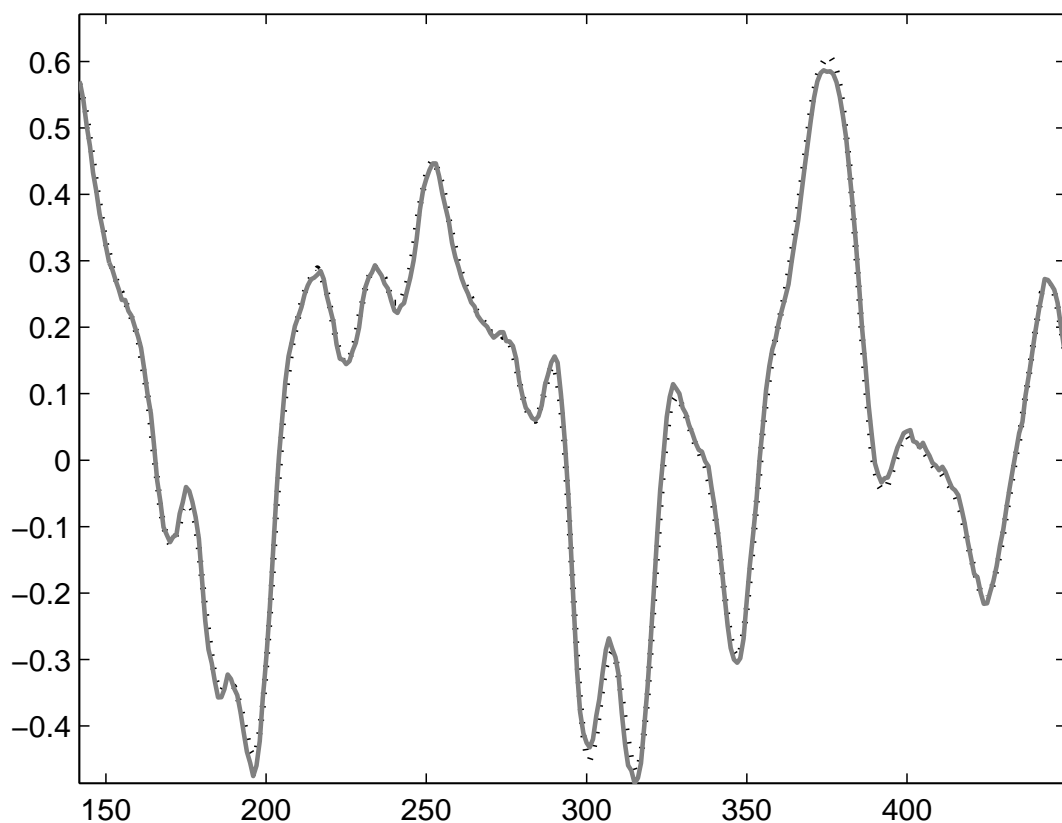


Рисунок 1.16 - Фрагмент прогнозу сигналу ЕЕГ

Таблиця 1.3 - Результати прогнозування сигналу ЕЕГ

Нейронна мережа / Алгоритм навчання	Кіл. шарів у Мгуа-Мережі	Кіл. входів	RMSE
МГУА-нейронна мережа з адаптивним складовим вейvloном / Запропонований алгоритм навчання (39) (Навчання всіх параметрів)	2	2	0.0113
МГУА-нейронна мережа з адаптивним складовим вейvloном / Градієнтний алгоритм (Навчання ваг і рецепторних полів)	2	2	0.0394
МГУА-нейронна мережа [59]	2		0.1997

Наступний експеримент був проведений для вирішення задачі прогнозування часового ряду погодинного енергоспоживання в одній з федеральних земель Німечинни [60].

Для прогнозування часового ряду було взяте МГУА-нейронну мережу з адаптивними вейvloнами з початковою кількістю входів  $n = 6$ . Кількість вейвлет-активаційних функцій в адаптивному вейvloні та їхні початкові параметри були отримані за допомогою процедури субтрактивної кластеризації [61]. Застосування цього методу дозволяє одержати не тільки матрицю координат центрів кластерів, алі також вектор, компоненти якого, визначають діапазон впливу центру кластера.

Для навчання мережі використовувалося 5000 точок ряду, а наступні 200 були використані для тестування мережі. Кожний адаптивний вейвлон навчався в пакетному режимі 5 епох.

Як критерій якості прогнозу було використано середньоквадратичну похибку (MSE) та середня абсолютна похибка у відсотках (MAPE). Результати

прогнозування вартового ряду енергоспоживання наведено на рис 1.17 (реальні (пунктирна лінія) і значення прогнозу (суцільна лінія)).

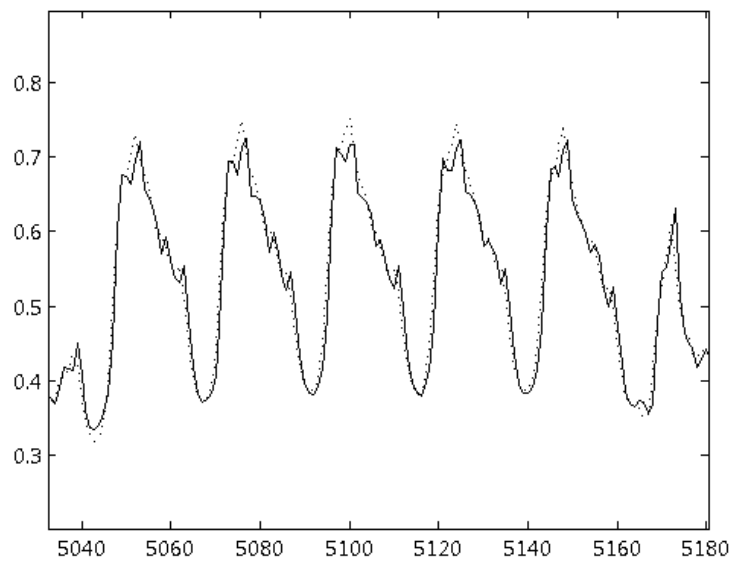


Рисунок 1.17 - Результати прогнозування ряду енергоспоживання

У таблиці 1.4 наведений порівняльний аналіз процесу прогнозування на основі запропонованої МГУА-нейронної мережі з настроюванням всіх параметрів адаптивного вейвлону з результатами традиційної МГУА-мережі [59].

Таблиця 1.4 - Порівняльний аналіз прогнозування вартового ряду енергоспоживання

Нейронна мережа / Алгоритм навчання	Кількість шарів	MSE	MAPE
МГУА-нейронна мережа на адаптивних вейвлоних / Запропонований метод навчання (39)	2	0.0019	4,01%
МГУА-нейронна мережа [59]	2	0.0042	9,06%

Як видно з наведених результатів, запропонований підхід забезпечує більш високу точність прогнозування в порівнянні зі звичайною МГУА-нейронною мережею.

Експериментальне дослідження запропонованої функціонально зв'язаної вейвлет-нейро-фаззі системи й робастного алгоритму навчання проводилося на основі сигналу, забрудненого інтенсивними викидами. Сигнал був отриманий на основі нелінійного динамічного об'єкта Нарендри-Партхасаратхі [23] (стандартний тест, широко розповсюджений для порівняння нейро- і нейро-фаззі систем при моделюванні й прогнозуванні часових рядів), чий вихідний сигнал був штучно забруднений випадковим шумом з розподілом Коші (розподіл з важкими хвостами) виду

$$F_X^{-1}(x) = x_0 + \gamma_{ch} \operatorname{tg}[\pi(x - 0.5)], \quad (1.48)$$

де  $x_0$  параметр локалізації,  $\gamma_{ch}$  параметр масштабу ( $\gamma_{ch} > 0$ ),  $x$  носій ( $x \in (-\infty, +\infty)$ ).

Нелінійний динамічний об'єкт був сгенерований згідно з рівнянням (1.47), де  $f(u(k)) = 0.6 \sin(u(k)) + 0.3 \sin(3u(k)) + 0.1 \sin(5u(k))$  й  $u(k) = \sin(2k/250)$ ,  $k$  дискретний час. Значення  $x(k-3), x(k-2), x(k-1), x(k)$  були використані для емуляції  $x(k+1)$ . У такий спосіб початкове значення числа входів було рівне  $n = 4$ , а кількість вузлів першого шару функціонально зв'язаної вейвлет-нейро-фаззі системи було рівне  $C_4^2 = 6$ . Параметри алгоритму навчання були прийняті  $\beta_w = 0.5, \beta_c = 0.5, \beta_Q = 0.5, \beta_\alpha = 0.5,$   $\gamma_w = \gamma_c = \gamma_Q = \gamma_\alpha = 0.99,$   
 $\lambda_w = \lambda_{c_j} = \lambda_{Q_j^{-1}} = \lambda_\alpha = 0.99.$  Початкові значення рівні  
 $\eta_w(0) = \eta_{c_j}(0) = \eta_{Q_j^{-1}}(0) = \eta_\alpha(0) = 10000.$

Навчальна вибірка містить 2000 точок, при цьому кожний вузол мережі був навчений за п'ять епох по 2000 ітерацій у кожній. Далі наступні 1000 точок були використані як тестова вибірка. Тестова вибірка містить у собі  $f(u(k)) = 0.6 \sin(\pi u(k)) + 0.3 \sin(3\pi u(k)) + 0.1 \sin(5\pi u(k))$  й  $u(k) = \sin(2\pi k/250)$  для  $2001 < k < 2500$  і  $f(u(k)) = u^3(k) + 0.3u^2(k) - 0.4u(k)$  й

$u(k) = \sin(2\pi k/250) + \sin(2\pi k/25)$  для  $2501 < k < 3000$ . Початкові значення синаптичних ваг були випадково сгенеровані в інтервалі від  $-0.1$  до  $+0.1$ .

На рис. 1.18 а представлені результати прогнозування зашумленого нестационарного динамічного сигналу, а на рис. 1.18 б показаний сегмент процесу навчання; можна бачити, що викид з великою амплітудою, що перебуває на початку вибірки, не вплинув на процес навчання.

Порівняння результатів прогнозування на основі робастного алгоритму навчання проводилося з результатами прогнозування на основі такої ж мережі з робастним алгоритмом навчання тільки синаптичних ваг, із градієнтним алгоритмом, з алгоритмом на основі рекурентного методу найменших квадратів, а також зі стандартної МГУА - нейронною мережею із градієнтним алгоритмом навчання.

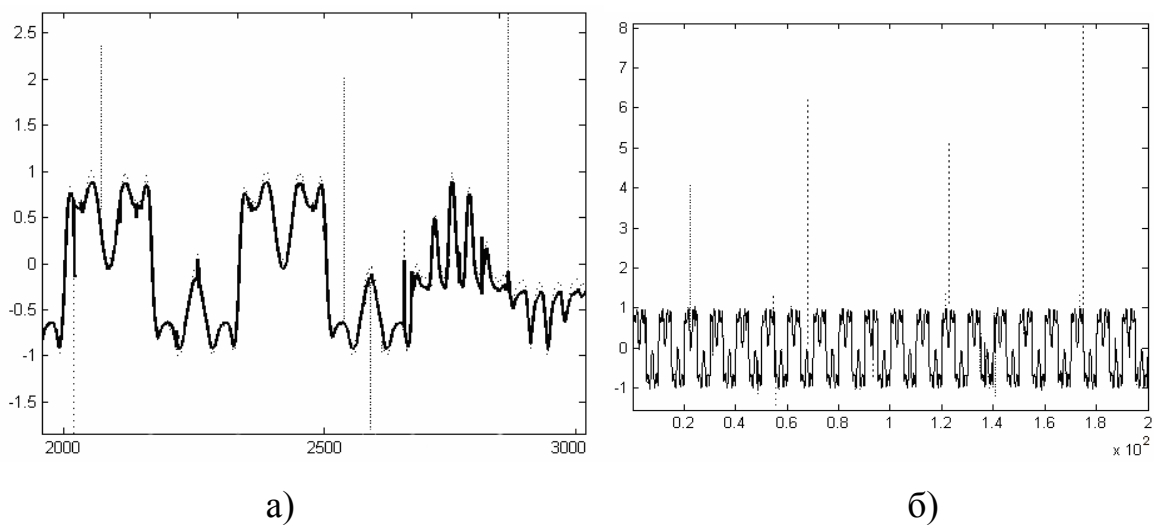


Рисунок 1.18 - Результати прогнозування

Найкращий результат прогнозування забезпечує запропонована система з робастним алгоритмом навчання всіх параметрів вузла. При навчанні тільки синаптичних ваг за допомогою робастного алгоритму помилка прогнозування збільшується. При навчанні градієнтним алгоритмом перший же викид на початку вибірки сильно впливає на процес навчання і як результат - більша помилка прогнозування. При навчанні системи рекурентним методом



найменших квадратів при першому ж викиді відбувається, так званий, «вибух параметрів» коваріаційної матриці і як результат - неможливість прогнозування сигналів, зашумлених аномальними викидами. Результати порівняння наведені в табл. 1.5.

Таблиця 1.5 - Результати прогнозування

Нейронна мережа / Алгоритм навчання	Кіл. шарів мережі / Кіл. входів у вузлі	RMSE
МГУА-нейронна мережа на W-Нейронах / Запропонований робастний алгоритм (настроюються всі параметри)	3/3	0.0493
МГУА-нейронна мережа на W-Нейронах / Запропонований робастний алгоритм (настроюються всі параметри)	3/2	0.0557
МГУА-нейронна мережа на W-Нейронах / Запропонований робастний алгоритм (настроюються тільки синаптичні ваги)	4/2	0.0899
МГУА-нейронна мережа на W-Нейронах / Гرادієнтний алгоритм	4/2	0.1398
МГУА-нейронна мережа на W-нейронах / Метод найменших квадратів	4/2	$\infty$
МГУА-нейронна мережа / Градієнтний алгоритм	4/2	1.4436

У такий спосіб видно, що запропонований робастний алгоритм навчання дозволяє обробляти сигнали в умовах істотного забруднення аномальними викидами.

## **1.6 Виявлення змін, які відбуваються в мультимедійних даних за допомогою аналізу часових рядів**

Одним з напрямів розвитку еволюційних систем є обробка мультимедійної інформації, зокрема задача виявлення змін мультимедійних даних набуває сьогодні все більшої уваги. Це пов'язано насамперед зі значним збільшенням інформаційних потоків і з тим, що виникає потреба виявлення змін в масивах даних, що динамічно змінюються, таких, як звук або відеодані. При цьому слід зазначити, що одним із шляхів обробки мультимедійних даних є представлення у вигляді одно- чи багатовимірних часових рядів з подальшим виявленням змін його властивостей. Задача дослідження змін властивостей часових рядів і їх сегментації досить давно привертає увагу дослідників, а для її вирішення до теперішнього часу склалася низка відносно незалежних підходів, серед яких, в першу чергу, необхідно відмітити статистичний, адаптивний та заснований на методах класифікації. І якщо для одновимірних рядів розроблені досить ефективні алгоритми аналізу, результати [62, 63], що відносяться до багатовимірних послідовностей, є значно скромнішим. В той же час, зрозуміло що у важливих застосуваннях, пов'язаних насамперед з обробкою відеосигналів, приходиться мати справу з багатовимірними сигналами і необхідністю їх аналізу в реальному часі [64, 65].

**1.6.1 Виявлення змін у потоці відеоданих на основі аналізу багатовимірних часових рядів.** У зв'язку з цим можна говорити про таку важливу задачу у рамках інтелектуального аналізу даних та розробки еволюційних систем, як представлення відеоданих в вигляді багатомірних часових рядів з подальшим розбиттям їх на відносно однорідні сегменти і виявлення моменту переходу від одного сегменту до іншого. Іншими словами можна казати про необхідність розв'язання задачі покадрового аналізу результатів сегментації відеоданих. Найбільш компактною формою опису багатовимірних часових рядів, яким можливо представити довільні відеодані, особливо у разі, коли число спостережень не фіксоване і росте з часом, є його

математична модель, що в тій або іншій формі описує характеристики, які цікавлять нас.

Для вирішення задачі виявлення зміни властивостей багатовимірних рядів зручним, компактним і ефективним їх описом є, так звані, векторні авторегресійні моделі (VAR-моделі).

В загальному випадку VAR-модель зв'язує минулі і поточні спостереження векторного сигналу  $x(k)$  у формі

$$x(k) = B_0 + \sum_{l=1}^p B_l x(k-l) + \xi(k), \quad (1.49)$$

де  $B_0 = \{b_{0i}\} - (n \times 1)$  вектор середніх значень,  $B_l = \{b_{lij}\} - (n \times n)$  матриці параметрів,  $p$  – порядок моделі.

Початковою інформацією для вирішення задачі ідентифікації і виявлення змін є тільки сам  $n$ -вимірний часовий ряд  $x(k)$ , значення якого в реальному часі надходять на адаптивний ідентифікатор.

Для спрощення подальших викладок введемо в розгляд складену матрицю  $B = (B_0 : B_1 : \dots : B_p)$  і вектор передісторії  $X(k) = (1, x^T(k-1), \dots, x^T(k-p))^T$  розмірності  $((pn+1) \times 1)$ , після чого перепишемо рівняння (1.49) у вигляді

$$x(k) = BX(k) + \xi(k), \quad (1.50)$$

де матриця апіорі невідомих параметрів  $B$  містить практично усю необхідну інформацію про властивості контрольованого сигналу.

Задача ідентифікації полягає в тому, що у відповідність сигналу (1.50) ставиться модель, що настроюється

$$\hat{x}(k) = B(k-1)X(k), \quad (1.51)$$

матриця параметрів  $B(k)$  якої, уточнюється на кожному такті часу  $k$  шляхом мінімізації прийнятого критерію ідентифікації, що є деякою функцією різниці розрахованих  $\hat{x}(k)$  і експериментальних даних  $x(k)$ . При цьому синтезована модель (1.51) зобов'язана бути працездатною і в режимі прогнозування, а порушення прогнозуючих властивостей може бути ознакою виникнення тих або інших змін у сигналі. Процес настроювання моделі (1.51) можливо реалізовувати за допомогою рекурентних процедур, які зручно представити в узагальненому вигляді

$$\begin{cases} B(k) = B(k-1) + \gamma(k)e(k)X^T(k), \\ e(k) = x(k) - \hat{x}(k) = x(k) - B(k-1)X(k), \end{cases}$$

де  $\gamma(k)$  – скалярний або матричний коефіцієнт кроку пошуку алгоритму, що визначає його властивості і залежить від прийнятого критерію ідентифікації,  $e(k)$  – векторна похибка ідентифікації.

Пропонується ввести в розгляд багатовимірну модифікацію алгоритму експоненційно зваженої стохастичної апроксимації у вигляді

$$\begin{cases} B(k) = B(k-1) + \frac{e(k)X^T(k)}{\beta r(k-1) + \|X(k)\|^2}, \\ r(k) = \beta r(k-1) + \|X(k)\|^2, \end{cases} \quad (1.52)$$

(тут  $0 \leq \beta \leq 1$  – параметр пам'яті алгоритму), що є своєрідним компромісом між однокроковою процедурою що є узагальненням оптимального за швидкодією алгоритму Качмажа на векторно-матричну модель, і рекурентним експоненційно зваженим методом найменших квадратів. Ця модифікація позбавлена недоліків своїх прототипів і має необхідні згладжуючі і слідкуючі властивості.

Механізм адаптації алгоритму (1.52) заснований на "пригніченні" застарілої інформації, при цьому динамічні властивості алгоритму повністю визначаються пам'яттю, на основі якої відбувається уточнення матриці поточних оцінок  $B(k)$ .

Слід зазначити, що найкращі фільтруючі властивості, мають алгоритми з необмеженою пам'яттю такі, як рекурентний метод найменших квадратів, в той же час ці процедури мають погані слідкуючі властивості у разі, якщо характеристики контрольованого сигналу змінюються в часі. В цьому випадку перевагу мають алгоритми з «короткою» пам'яттю типу алгоритму Качмажа, проте ці процедури погано працюють в умовах перешкод.

Таким чином, при оцінюванні систем, в яких можуть відбуватися зміни, об'єм пам'яті алгоритму слід вибирати на основі компромісу між його згладжуючими і слідкуючими властивостями. На жаль, в реальних ситуаціях характеристики збурень, дрейфів та стрибків апіорі невідомі і можуть змінюватися в процесі функціонування контрольованої системи. У цих умовах важко віддати перевагу одному алгоритму з фіксованою пам'яттю, а тому доцільно використовувати алгоритми зі змінною пам'яттю, величину якої можна оперативно змінювати залежно від наявності або відсутності змінень у сигналі.

Для контролю за змінами багатовимірного часового ряду доцільно використовувати модифікацію критерію Манна-Уїтні у вигляді

$$\max_i \left( \sum_{u=k-S+1}^k \text{sign}(x_i(u) - \hat{x}_i(u)) \right) \geq \gamma,$$

(тут  $\gamma$  – деякій поріг, що обирається з емпіричних міркувань), тобто одночасно контролювати усі компоненти і фіксувати факт зміни, якщо розладнання станеться в хоч би одній компоненті  $x_i(k)$ .

С метою встановлення застосовності підходу, описаного вище був проведений експеримент, над фрагментом відеоданих медичної спрямованості.

Спочатку вихідні дані було розбито на сукупність кадрів, для кожного з яких була проведена сегментація за допомогою алгоритму JSeg. Приклад одного з кадрів, а також його сегментація представлені на рис. 1.19.



Рисунок 1.19 - Приклад сегментації відеокадра

Надалі для пошуку змін в відеоданих був сформований вектор характеристик сегментації (площа сегменту, периметр і т.п.), на основі якого була сформована матриця для кожного сегменту кожного з відеокадрів. Значення кількості характеристик в даному випадку було вибране рівним 11. Таким чином, для кожного кадру ми отримали матрицю розмірності  $n \times m$ , де  $n$  – число сегментів, а  $m$  – число характеристик.

Після цього були знайдені значення параметрів після перетворень для кожного з кадрів. Таким чином, ми отримали свертку від багатовимірного часового ряду до одновимірного випадку. Результати цього процесу показані на рис. 1.20.

З приведенного графіку зміни поведінки відео даних у часі можна виділити два явні перепади значень. Перший з них в районі діапазону кадрів 40-60. Як ми бачимо з представлених прикладів відеокадрів на лівій і правій межі інтервалу маємо приблизно однакові зображення. В той же час всередині інтервалу була явна зміна сцени, яка визвано зміщенням камери та появою сто раннього об'єкту. Таким чином, можна стверджувати, що зміни, які сильно змінюють початкові дані за короткий проміжок часу досить ефективно детектуються. Це підтверджується і другим перепадом, що виник в районі 300-го кадру. З

наведених прикладів кадрів ми бачимо, як істотно змінилися дані в цей час. При цьому стався і серйозний перепад знайдених значень.

У той же час, слід зазначити деякий негативний момент, а саме, розпочинаючи приблизно з 240-250 кадру у відеоданих стала відбуватися плавна зміна сцени, викликана зрушенням камери і появою стороннього об'єкту

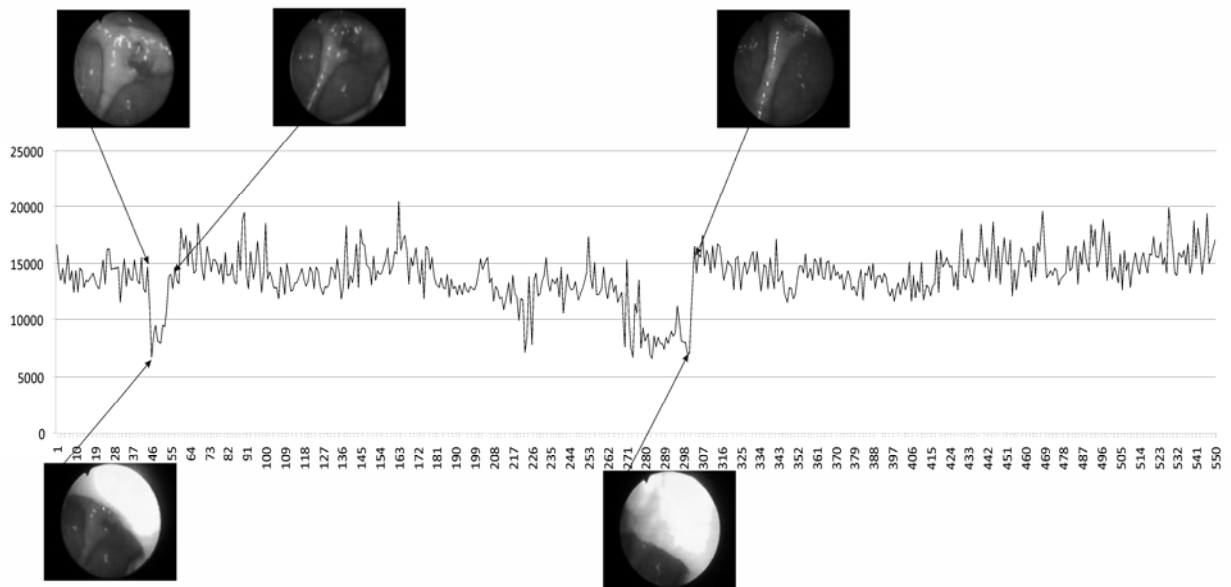


Рисунок 1.20 - Результати обробки відеопослідовності

в кадрі. Але у відмінності від першого випадку цей процес був дуже плавним і тут ми спостерігаємо неоднозначну картину. З одного боку загальний тренд характеристик, які знайдені, явно видно – значення стали зменшуватися, але цей процес відбувався настільки плавно, що якби було не різке зміщення камери в районі 300-го кадру, а таке ж плавне зміщення у зворотний бік, то такий процес зміни сцени можна було б звичайно детектувати, але вірогідність помилки детектування сильно зросла б.

**1.6.2 Застосування вейвлет-ентропії для дослідження зміни структури стохастичних сигналів.** Багато проблем аналізу та управління складними технологічними та інформаційними процесами зводяться до виявлення різких змін (розладнання) характеристик аналізованого сигналу. Розладнання являє

собою порушення закономірності в динаміці або структурі сигналу і може служити передвісником критичних явищ і аварійних ситуацій.

Як правило, реальні сигнали, які потрібно проаналізувати, є нестационарними, зашумленими та неоднорідними. Вейвлет-перетворення було розроблено в середині 80-х років як альтернатива перетворенню Фур'є для дослідження часових рядів із сильною неоднорідністю. Вейвлет-перетворення, що володіє рухомим частотно-часовим вікном, однаково добре виявляє як низькочастотні, так і високочастотні характеристики сигналу на різних часових масштабах. Вейвлет-аналіз особливо ефективний для дослідження нестационарних рядів, що містять сегменти з різним характером поведінки, для рядів, які є істотно неоднорідними і володіють локальними особливостями [66-69].

Як інструмент для виявлення розладнання стохастичного ряду запропоновано використовувати таку кількісну характеристику вейвлет-перетворення, як вейвлет-ентропія. Адаптивне оцінювання вейвлет-ентропії дозволяє відстежити появу сегментів часового ряду з різним характером поведінки. Це, в свою чергу, дозволяє визначати час реакції системи на зовнішні впливи.

У разі дискретних сигналів дискретне вейвлет-перетворення приводить до наступного результату: для початкового набору значень сигналу ми отримуємо сукупність наборів значень деталізуючих і апроксимуючих вейвлет-коефіцієнтів

$$\left\{ \left\{ apr(N, k), k = 1, 2, \dots, [T \cdot 2^{-N}] \right\}, \left\{ \left\{ det(j, k), k = 1, 2, \dots, [T \cdot 2^{-j}] \right\}, j = 1, 2, \dots, N \right\} \right\},$$

де операція  $[\cdot]$  позначає взяття цілої частини [ 67, 68].

Величина вейвлет-енергії на заданому рівні вейвлет-розкладу  $j$  з кількістю деталізуючих вейвлет-коефіцієнтів  $N_j$  визначається як [67, 69-71]



$$E_j = \frac{1}{N_j} \sum_{k=1}^{N_j} \det^2(j, k).$$

Набір величин  $E_j$  для кожного рівня розкладу становить спектр вейвлет-енергії сигналу. Повна вейвлет-енергія спектру являє собою суму енергій кожного рівня:

$$E_{tot} = \sum_{j=1}^N E_j,$$

де  $N$  – максимальний рівень розкладу. Відносна вейвлет-енергія показує розподіл енергії за рівнями розкладу:

$$p_j = \frac{E_j}{E_{tot}}.$$

Вейвлет-ентропія  $WE$  є кількісною мірою впорядкованості сигналу і визначається за формулою [70-72]:

$$WE = - \sum_{j=1}^N p_j \ln(p_j).$$

Відносна вейвлет-ентропія  $RWE$  [71, 72] є мірою подібності розподілів вейвлет-енергій  $\{p_j\}$  і  $\{q_j\}$  для двох сигналів або двох ділянок одного сигналу і обчислюється як

$$RWE(p|q) = \sum_{j=1}^N p_j \ln \left( \frac{p_j}{q_j} \right).$$

Значення відносної вейвлет-ентропії позитивно і дорівнює нулю тільки в разі збігів розподілів  $\{p_j\}$  і  $\{q_j\}$ . Чим більше значення  $RWE$ , тим більше розходження між розподілами та, відповідно, між сигналами.

Важливим питанням при виконанні дискретного вейвлет-перетворення є вибір типу вейвлет-функції та кількості рівнів розкладу. Зазвичай вейвлет-функція підбирається в залежності від часових і частотних характеристик кожного аналізованого сигналу. Максимальний рівень розкладу залежить від того, які частотні діапазони необхідно досліджувати.

Адаптивне оцінювання вейвлет-ентропії модельних сигналів. [73-75] Для проведення чисельних досліджень в якості основних модельних сигналів були розглянуті гармонічні сигнали і реалізації процесів авторегресії. Відносна вейвлет-ентропія  $RWE(p|q)$  розраховувалася двома способами. У першому способі вейвлет-ентропія показує частотну схожість двох послідовних ділянок сигналу  $X(i-1)$  і  $X(i)$ :

$$RWE(p(i-1)|p(i)) = \sum_{j=1}^N p(i-1)_j \ln \left( \frac{p(i-1)_j}{p(i)_j} \right),$$

$$RWE = \sum_j p_j \ln \left( \frac{p_j}{q_j} \right) \quad (1.53)$$

де  $\{p(i-1)_j\}$  і  $\{p(i)_j\}$  – спектри вейвлет-енергій суміжних ділянок сигналу,  $N$  – максимальний рівень розкладу. У другому випадку вейвлет-ентропія показує, наскільки відрізняється  $i$ -та ділянка сигналу  $X(i)$  від обраного базового сигналу  $X_0$ :

$$RWE(p_0|p(i)) = \sum_{j=1}^N p_0_j \ln \left( \frac{p_0_j}{p(i)_j} \right),$$

$$.RWE = \sum_j p_j \ln \left( \frac{p_j}{q_j} \right). \quad (1.54)$$

де  $\{p_{0,j}\}$  – спектр вейвлет-енергії базового сигналу,  $\{p(i)_j\}$  – спектр  $i$ -тої ділянки сигналу.

На рис. 1.21 представлені модельні гармонічні сигнали (вгорі). Сигнал А являє собою синусоїду виду  $X(t) = \sin(13\pi t)$ , сигнал В є сумою двох синусоїд  $X(t) = 2\sin(3\pi t) + \sin(25\pi t)$ . У разі С розглянуто зашумлений адитивним білим шумом гармонічний сигнал А. На рис. 1.21 (внизу) показані відповідні спектри вейвлет-енергій, які отримано розкладом сигналу на 8 рівнів. Спектри вейвлет-енергій наочно відображають частотну структуру сигналів. Відповідні значення вейвлет-ентропій рівні  $WE(A) = 0.556$ ,  $WE(B) = 0.912$ ,  $WE(C) = 1.668$ . Очевидно, що сигнал А є більш упорядкованим, ніж сигнал В, а ентропія зашумленого сигналу є найбільшою. Значення відносних вейвлет-ентропій відповідно дорівнюють:  $RWE(A|B) = 2.79$ ,  $RWE(A|C) = 0.45$ ,  $RWE(B|C) = 1.76$ , що дозволяє зробити висновок про спектральну близькість сигналів А і С.

Розглянемо зміну величини відносної вейвлет-ентропії при зміні сигналу у часі. На рис. 1.22 (вгорі) представлено модельний гармонічний сигнал із змінною амплітудою, який до середини реалізації є синусоїдою (сигнал А), а після – сумою двох синусів (сигнал В). Модельний сигнал містить 2048 відліків. Для визначення структурних відмінностей між першою і другою половинами сигналу, сигнал був розбитий на ділянки по 128 відліків. Для кожного з ділянок була розрахована відносна вейвлет-ентропія. У середній частині рис. 1.22 показано зміну величини  $RWE(p(i-1)|p(i))$  для послідовних ділянок сигналу, обчисленою відповідно до (1.53). Оскільки повільна зміна амплітуди відображається тільки в зміні апроксимуючих вейвлет-коефіцієнтів, значення вейвлет-спектрів на суміжних ділянках однакові, за винятком ділянки, де відбувається зміна частоти сигналу. У нижній частині рис. 1.22 представлено значення величини  $RWE(p_0|p(i))$ , що розрахована за формулою (1.54), де в якості базового спектру було обрано спектр вейвлет-енергії сигналу А. В цьому

випадку величина відносної вейвлет-ентропії змінюється, починаючи з ділянки появи сигналу В.

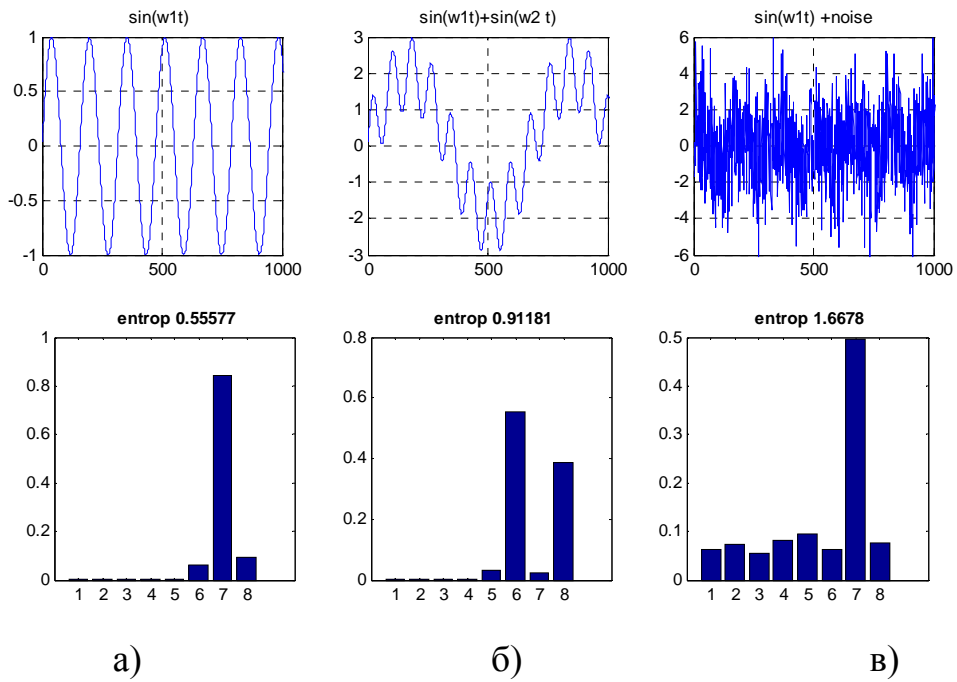


Рисунок 1.21 - Гармонічні сигнали (вгорі) та відповідні спектри вейвлет-енергій (внизу)

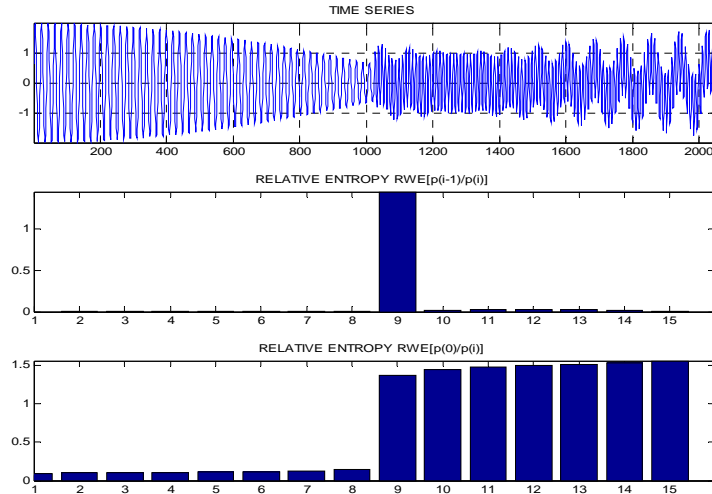


Рисунок 1.22 - Сигнал (угорі), змінення відносної вейвлет-ентропії  $RWE(p(i-1)|p(i))$  (в середині), змінення відносної вейвлет-ентропії  $RWE(p_0|p(i))$  (внизу)

На рис. 1.23 представлені реалізації процесів авторегресії 1-го і 2-го порядку (вгорі). Часові ряди А і В є авторегресіями 1-го порядку:

$X(t) = \phi X(t-1) + \varepsilon(t)$ , де  $\varepsilon(t)$ - некорельований білий шум з відповідними коефіцієнтами авторегресії  $\phi = 0.9$  и  $\phi = -0.4$ . У випадку С розглянуто процес авторегресії 2-го порядку:  $X(t) = \phi_1 X(t-1) + \phi_2 X(t-2) + \varepsilon(t)$  з коефіцієнтами  $\phi_1 = 0.2$  и  $\phi_2 = 0.3$ .

На рис. 1.23 (внизу) показані відповідні спектри вейвлет-енергій, які отримано розкладанням часових рядів на 6 рівнів. Спектри вейвлет-енергій відображають частотну структуру реалізацій: у реалізації А переважають повільні коливання, обумовлені великим коефіцієнтом авторегресії; реалізація В містить різкі скачки і швидкі коливання, які викликані негативною кореляцією; реалізація С володіє відносно рівномірним вейвлет-спектром. Відповідні значення вейвлет-ентропій дорівнюють:  $WE(A) = 1.072$ ,  $WE(B) = 1.603$ ,  $WE(C) = 1.784$ . Очевидно, що реалізація А є найбільш упорядкованою з представлених. Значення відносних вейвлет-ентропій відповідно дорівнюють:  $RWE(A|B) = 1.08$ ,  $RWE(A|3) = 0.78$ ,  $RWE(B|C) = 0.19$ .

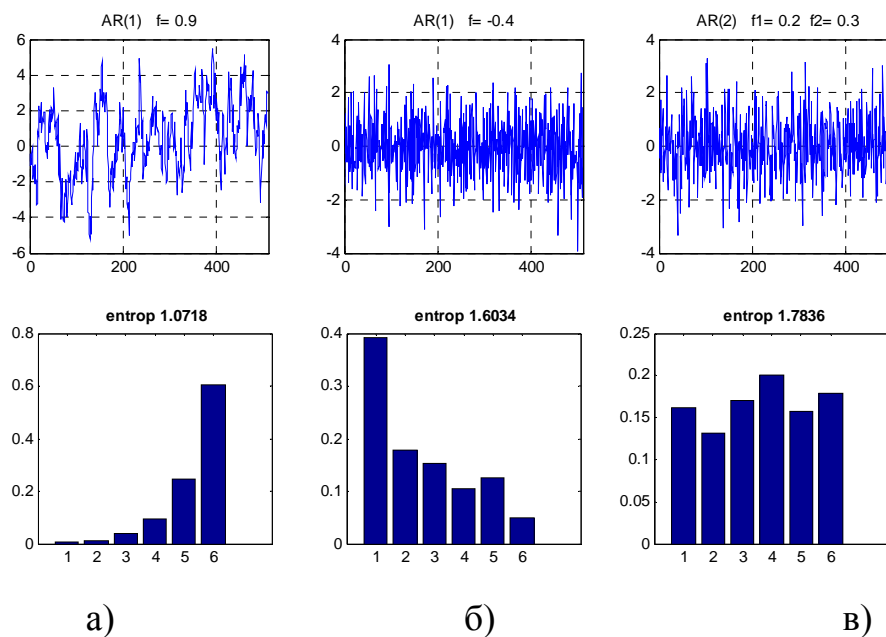


Рисунок 1.23 - Реалізації процесів авторегресії (вгорі) і відповідні спектри вейвлет-енергій (внизу)

На рис. 1.24 (угорі) представлена модельна реалізація, яка містить поліноміальний тренд, довжиною 5000 відліків. Перша частина реалізації

відповідає авторегресії 1-го порядку з параметром  $\phi = -0.4$ . Друга частина є авторегресією 2-го порядку з коефіцієнтами  $\phi_1 = 0.2$  і  $\phi_2 = 0.3$ . Для визначення структурних відмінностей між першою і другою половинами часового ряду, реалізація була розбита на інтервали, що містить 500 відліків кожен. Для кожної з ділянок була розрахована відносна вейвлет-ентропія.

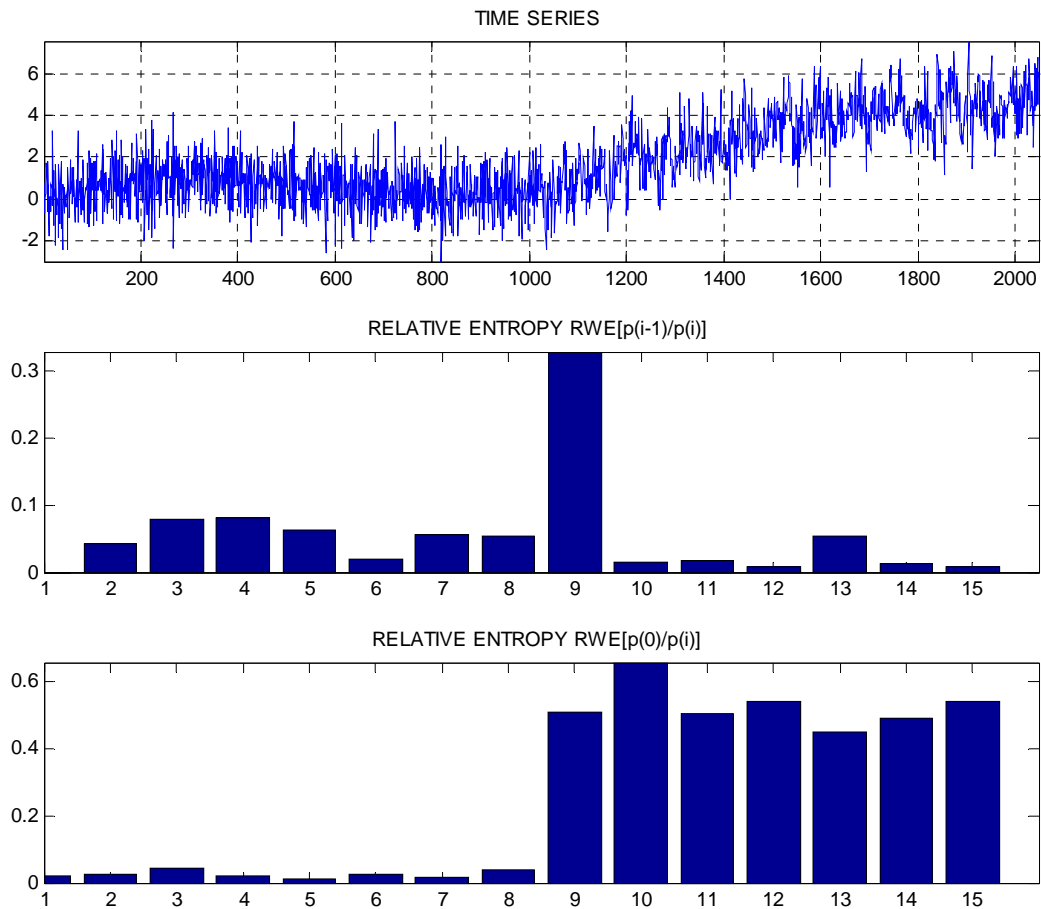


Рисунок 1.24 - Часовий ряд (угорі), змінення відносної вейвлет-ентропії  $RWE(p(i-1)|p(i))$  (в середині), змінення відносної вейвлет-ентропії  $RWE(p(0)|p(i))$  (внизу)

У середній частині рис. 1.24 показано змінення величини  $RWE(p(i-1)|p(i))$  для послідовних ділянок ряду, обчисленої за формулою (1.53). Оскільки повільна зміна трендової складової відображається тільки в

зміні апроксимуючих вейвлет-коефіцієнтів, значення вейвлет-спектрів на суміжних ділянках реалізацій авторегресії близькі один одному, а відповідна відносна вейвлет-ентропія близька до нуля, за винятком ділянки, де відбувається зміна параметрів процесу авторегресії. У нижній частині рис. 1.24 представлені значення величини  $RWE(p_0|p(i))$ , розрахованої відповідно до (1.54), де в якості базового спектру обраний спектр вейвлет-енергії авторегресії 1-го порядку. Очевидно, що значення вейвлет-ентропії для першої половини реалізації близькі до нуля, а для другої половини стають значно більше. Це говорить про наявність спектральної близькості між першою половиною досліджуваного ряду і авторегресією 1-го порядку, і відмінності від неї другої половини ряду.

Таким чином, проведені чисельні дослідження на модельних даних показують, що адаптивне оцінювання відносної вейвлет-ентропії дозволяє відстежити появу сегментів часового ряду з різними спектральними (кореляційними) властивостями.

Застосування відносної вейвлет-ентропії для виявлення різких звуків в аудіозапису. [74] Розглянемо застосування вейвлет-ентропії на прикладі звукового сигналу з метою виявлення раптових частотних сплесків, таких як клацання або тріск. На рис. 1.25 представлені графічне зображення звукового запису, довжиною 12 секунд. Звуковий сигнал містить різкий клацання в момент часу, рівний 2 с.

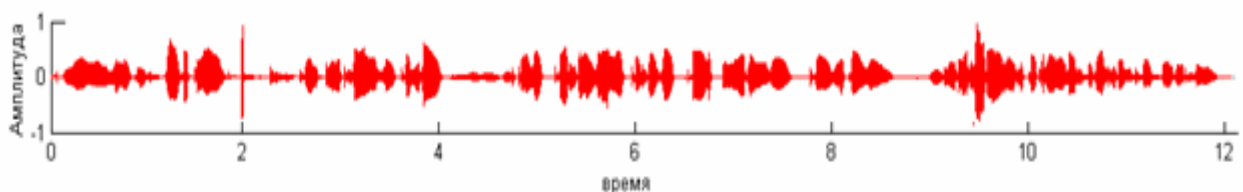


Рисунок 1.25 - Графік звукового сигналу

Для визначення клацання було використано розрахунок відносної вейвлет-ентропії  $RWE(p(i-1)|p(i))$ , обчисленої за формулою (1.53) і

визначальною частотну схожість двох прилеглих ділянок сигналу. Звуковий сигнал був розбитий на ділянки довжиною 200 мс і розрахована відносна ентропія цих ділянок. На рис. 1.26 показаний графік зміни відносної ентропії  $RWE(p(i-1)|p(i))$ . Вейвлет-розклад сигналу було виконано на 9 рівнів за допомогою вейвлет-функції сімейства вейвлетів Добеші db3.

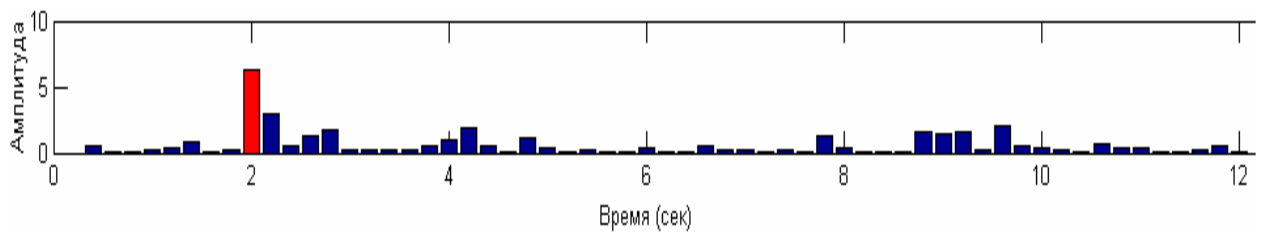


Рисунок 1.26 - Змінення відносної вейвлет-ентропії  $RWE(p(i-1)|p(i))$  звукового сигналу

Максимальне значення відносної ентропії спостерігається на часовому інтервалі 2000-2200 мс, що говорить про максимальне розходження вейвлет-спектра цієї ділянки в порівнянні з сусідніми, тобто саме на цій ділянці міститься звукове клацання.

Розглянуті методи були протестовані на модельних сигналах і застосовані до аналізу реальних сигналів. Залежно від природи сигналу і завдання дослідження, можна використовувати оцінювання різних кількісних характеристик. Адаптивне оцінювання відносної ентропії, яке проведене на основі вейвлет-спектрів суміжних ділянок аудіозаписи, дозволяє виявити моменти появи раптових коротких частотних сплесків, таких як клацання або тріск.

### 1.7 Еволюційні мережі на базі гібридних нейроподібних елементів

Запорукою успішності моделювання складних систем, крім іншого, є максимальне врахування апріорної інформації про властивості системи та її



вхідних і вихідних сигналів. Коли такої інформації немає, застосовуються універсальні моделі типу «темна скриня», параметри яких не пов'язані з фізичними властивостями досліджуваного процесу. Ці моделі здатні забезпечувати достатньо точну апроксимацію залежностей типу «вхід-вихід», проте вони не можуть покращити розуміння внутрішнього функціонування досліджуваної системи. Серед подібних моделей останнім часом широко використовуються штучні нейронні мережі (ШНМ) і нейро-фаззі системи, здатні забезпечувати як завгодно точну апроксимацію безперервних функцій багатьох змінних [26].

Для врахування апріорної інформації використовують спеціалізовані архітектури штучних нейронних мереж [76, 77], зокрема, побудовані з використанням нейро-фаззі елементів [78, 79] і динамічних нейронів зі скінченною імпульсною характеристикою [80]. Комбінуючи в прихованих шарах мережі ці типи нейронів, а також стандартні нейрони Маккалоха-Пітса і динамічні нейрони з нескінченною імпульсною характеристикою [81], використовуючи неповнозв'язні архітектури, можна створювати спеціалізовані нейронні мережі, структура яких максимально відповідає специфіці розв'язуваної задачі [82]. Таким чином, отримують моделі типу «сіра скриня», які мають часткову відповідність зі структурою досліджуваної системи. Описаний спосіб спеціалізації архітектур нейронних мереж заснований на виборі типів використовуваних нейронів і структури зв'язків між ними.

Для досягнення більш гнучкої спеціалізації і, відповідно, зменшення кількості параметрів, що настроюються, можна опуститися до рівня підбору окремих синапсів різних нейронів, для чого введемо в розгляд новий тип будівельного блоку – гібридний нейроподібний елемент (HNU), структура якого представлена на рис. 1.27 [83].

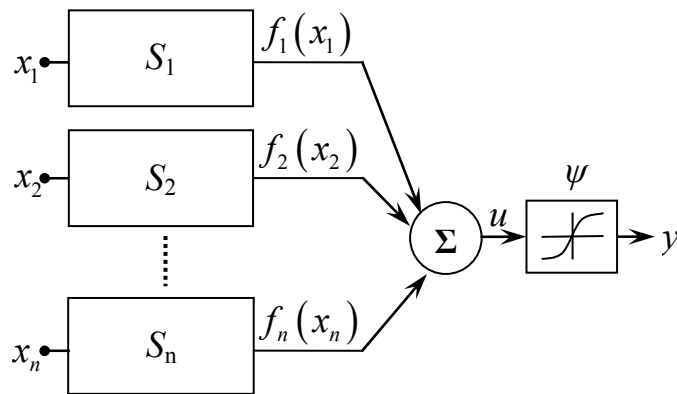


Рисунок 1.27 - Гібридний нейроподібний елемент

Вхідні сигнали  $x_i (i=1, \dots, n)$  перетворюються за допомогою синапсів різних типів  $S_i$  в сигнали  $f_i(x_i)$ , які потім об'єднуються в сигнал внутрішньої активації  $u = \sum_{i=1}^n f_i(x_i)$ . Вихідний сигнал нейрона формується за допомогою нелінійної активаційної функції

$$y = \psi(u) = \psi\left(\sum_{i=1}^n f_i(x_i)\right), \quad (1.55)$$

де в якості  $\psi(u)$ , зазвичай, використовується сигмоїдальна функція або гіперболічний тангенс.

У гібридному нейроподібному елементі використовуються синапси чотирьох типів: лінійний синапс (рис. 1.28), синапс-фільтр з нескінченною імпульсною характеристикою (рис. 1.29), синапс-фільтр із скінченною імпульсною характеристикою (рис. 1.30) і нелінійний синапс на основі нечіткої системи (рис. 1.31). У наведених схемах  $w_i, w_{ij}, v_{ij}$  – синаптичні ваги, що настроюються,  $\mu_{ij}$  – функції належності,  $z^{-1}$  – елементи чистої затримки,  $d_w, d_v$  – максимальні порядки затримок.

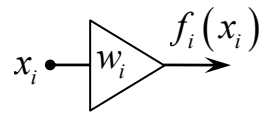


Рисунок 1.28 - Лінійний синапс

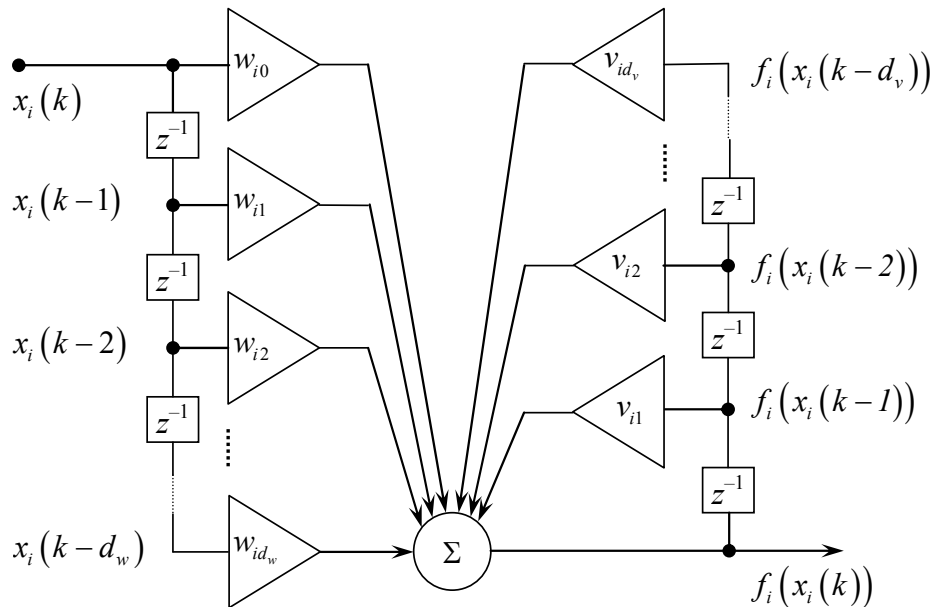


Рисунок 1.29 - Синапс-фільтр з нескінченною імпульсною характеристикою

Відповідно до даних схем вихідні сигнали синапсів різних типів формуються таким чином:

– для лінійного синапсу:  $f_i(x_i) = w_i x_i$ ;

– для синапсу-фільтра з нескінченною імпульсною характеристикою:

$$f_i(x_i(k)) = \sum_{j=0}^{d_w} w_{ij} x_i(k-j) + \sum_{j=1}^{d_v} v_{ij} f_i(x_i(k-j));$$

– для синапсу-фільтра із скінченною імпульсною характеристикою:

$$f_i(x_i(k)) = \sum_{j=0}^{d_w} w_{ij} x_i(k-j);$$

– для нелінійного синапсу на основі нечіткої системи:  $f_i(x_i) = \sum_{j=1}^{h_i} w_{ij} \mu_{ij}(x_i)$ , де  $h_i$

– кількість функцій належності для  $i$ -го входу.

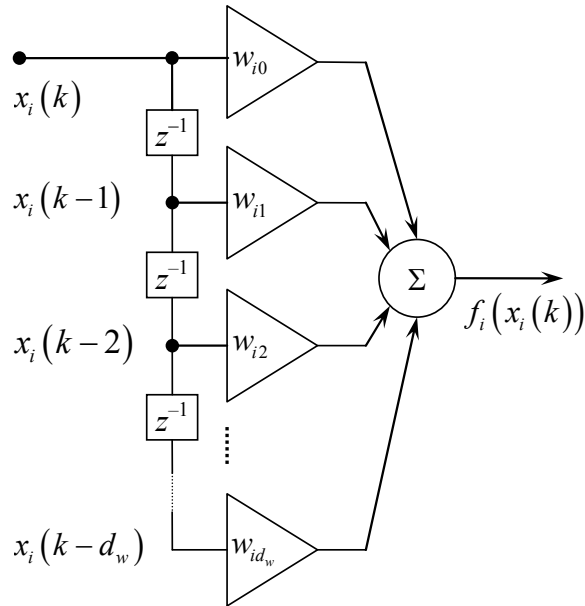


Рисунок 1.30 - Синапс-фільтр із скінченною імпульсною характеристикою

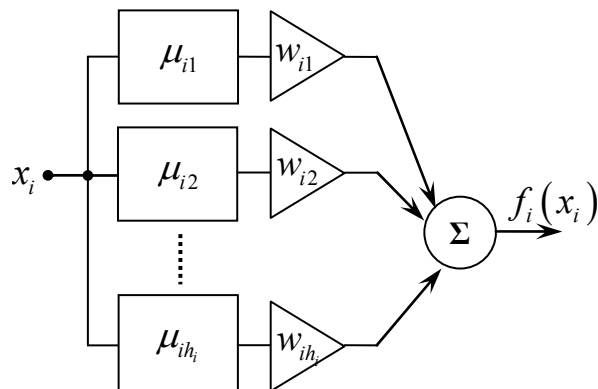


Рисунок 1.31 - Нелінійний синапс на основі нечіткої системи

У нелінійних синапсах на основі нечіткої системи зазвичай застосовуються трикутні функції належності (рис. 1.32), значення яких визначаються відстанню між входом  $x_i$  і центрами  $c_{ij}$ :

$$\mu_{ij}(x_i) = \begin{cases} \frac{x_i - c_{i,j-1}}{c_{ij} - c_{i,j-1}}, & x_i \in [c_{i,j-1}, c_{ij}]; \\ \frac{c_{i,j+1} - x_i}{c_{i,j+1} - c_{ij}}, & x_i \in [c_{ij}, c_{i,j+1}]; \\ 0 & \text{в протилежному випадку,} \end{cases} \quad (1.56)$$

при цьому автоматично забезпечується розбиття Руспіні  $\sum_{j=1}^{h_i} \mu_{ij}(x_i) = 1, \forall i$ .

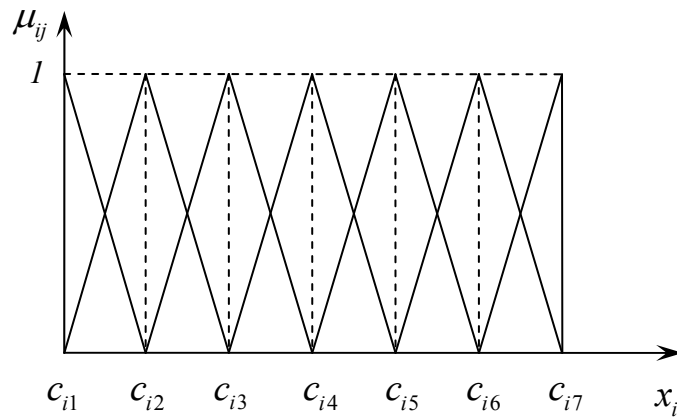


Рисунок 1.32 - Фазифікація змінної трикутними функціями належності

Якщо активний нечіткий інтервал  $p$ , вихід нелінійного синапсу можна виразити таким чином:

$$\begin{aligned} f_i(x_i) &= \sum_{j=1}^{h_i} w_{ij} \mu_{ij}(x_i) = w_{ip} \mu_{ip}(x_i) + w_{i,p+1} \mu_{i,p+1}(x_i) = \\ &= \frac{c_{i,p+1} - x_i}{c_{i,p+1} - c_{ip}} w_{ip} + \frac{x_i - c_{ip}}{c_{i,p+1} - c_{ip}} w_{i,p+1} = a_i x_i + b_i, \end{aligned} \quad (1.57)$$

$$\text{де } a_i = \frac{w_{i,p+1} - w_{ip}}{c_{i,p+1} - c_{ip}},$$

$$b_i = \frac{c_{i,p+1}W_{ip} - c_{ip}W_{i,p+1}}{c_{i,p+1} - c_{ip}}.$$

Застосування гібридних нейроподібних елементів дозволяє будувати нейронну мережу за стандартною багат шаровою архітектурою, спеціалізація якої досягається за рахунок:

- вибору типу синапсу для кожного зв'язку мережі, включаючи можливість розриву зв'язку (відсутність синапсу);
- вибору кількості та порядку елементів затримок в синапсах-фільтрах із скінченною і нескінченною імпульсними характеристиками;
- вибору кількості та параметрів функцій належності в нелінійних синапсах.

У першому прихованому шарі мережі такий вибір може здійснюватися на основі апріорної інформації про властивості оброблюваних сигналів, проте в подальших прихованих шарах і в вихідному шарі ця інформація відсутня, що веде до необхідності розв'язання задачі структурної оптимізації для синапсів цих шарів. Для вирішення цієї задачі пропонується застосувати еволюційний підхід, який також відноситься до методів обчислювального інтелекту. Це дозволить знайти оптимальні значення не тільки названих характеристик, але й кількість прихованих шарів мережі, а також кількість гібридних нейроподібних елементів в кожному з них [84]. Параметрична ідентифікація згенерованої таким чином структури може потім здійснюватися або традиційними методами на основі зворотного поширення похибки, або також з використанням еволюційного підходу.

Таким чином, побудова гібридних мереж на базі гібридних нейроподібних елементів дає можливість гнучкого вибору між універсальними (всі синапси одного типу, повнозв'язна архітектура) і спеціалізованими (синапси різних типів та/або неповнозв'язна архітектура) архітектурами. При цьому можлива реалізація нелінійних моделей авторегресії (NAR), авторегресії із зовнішніми входами (NARX), авторегресії – ковзного середнього (NARMA), авторегресії – ковзного середнього з зовнішніми входами (NARMAX).

Запропонований підхід було успішно застосовано авторами для розв'язання задачі прогнозування ожеледного навантаження на повітряних лініях електропередачі [85], що дозволило підвищити точність прогнозування, знизити кількість параметрів моделі, що настроюються, підвищити її стійкість до різного роду збурень. Тим самим було підтверджено працездатність і перспективність запропонованого класу моделей.

### **1.8 Прогнозувальна модель стислих даних на основі багат шарового перцептронну та нео-нейро-фаззі мережі зустрічного поширення**

Запропоновано нейромережу архітектуру, що складається з тришарового перцептронну «bottle-neck» та нейро-нео-фаззі мережі зустрічного поширення, а також запропоновано алгоритм навчання, що відзначається високою швидкодією. Запропонована гібридна система призначена для вирішення задачі прогнозування стислих даних за умов короткої навчальної вибірки.

Задача компресії інформації полягає у зменшенні кількості зберігаємих або передаваних даних з можливістю їх повної декомпресії. Застосування багат шарової нейронної мережі дозволяє вирішити задачу компресії з втратами (з припустимою втратою визначеної частини інформації) за умов гарної узагальнюючої здатності та відносно високого коефіцієнту компресії. При цьому, на відміну від стандартного аналізу головних компонент, багат шарова нейромережа здатна відображати та зберігати нелінійні залежності всередині даних. Запропоновану архітектуру можна використати, наприклад, у медицині при обробці кардіограми, коли інформація надходить з декількох датчиків.

Запропонована система є гібридом двох нейромереж – багат шарового перцептронну та нейро-нео-фаззі мережі зустрічного поширення.

У загальному випадку не існує теоретичного обмеження кількості шарів у перцептронні, але на практиці їх звичайно три. Усі нейрони одного шару односпрямовано пов'язані з нейронами суміжних шарів. Такі зв'язки є

синаптичними вагами. У тришаровому персептроні нейрони об'єднуються в послідовно пов'язані шари. Нейрони шару 0 не виконують жодних обчислень, а лише подають вхідні сигнали до нейронів першого шару, котрий називається першим прихованим шаром. Останній шар є вихідним, у якому обчислюється реакція мережі.

Кожний нейрон багатошарового персептрону характеризується одним виходом та багатьма входами, які, в свою чергу, є виходами нейронів попереднього шару. Кількість нейронів у всіх шарах, окрім другого, дорівнює  $n$ , а у другому –  $m$ , причому  $m < n$ .

Нехай  $u_j^{[s]}$  - внутрішній сигнал  $j$ -го нейрона  $s$ -го шару, що містить  $n_s$  нейронів, є взваженою сумою виходів попереднього шару

$$u_j^{[s]} = \sum_{i=0}^{n_{s-1}} w_{ji}^{[s]} o_i^{[s-1]}, \quad (1.58)$$

де  $w_{ji}^{[s]}$  - синаптичні ваги, що поєднують кожний нейрон  $s$ -го шару зі всіма нейронами попереднього шару так, що  $o_i^{[s-1]} = x_i^{[s]}$ ,  $o_i^{[0]} = x_i$ ,  $o_i^{[3]} = \tilde{x}_i$ .

Тоді вихід кожного нейрона є нелінійним перетворенням виду

$$o_j^{[s]} = \psi_j^{[s]}(u_j^{[s]}) = \psi_j^{[s]} \left( \sum_{i=0}^{n_{s-1}} w_{ji}^{[s]} o_i^{[s-1]} \right), \quad (1.59)$$

де для скорочення запису величина зміщення  $\theta_j^{[s]}$  позначається як синаптична вага  $w_{j0}^{[s]} = \theta_j^{[s]}$ , а  $o_0^{[s-1]} \equiv 1$ .

Предиктор становить собою нейро-нео-фаззі мережу зустрічного поширення, що в свою чергу складається з шару нечіткої кластеризації Кохонена і шару нео-фаззі-нейронів. На вхід шару-предиктору подаються вихідні сигнали з другого шару персептрона (шару-компресору), таким чином на вхід подається вектор стислих значень  $o^{[2]}(k) = (o_1^{[2]}(k), o_2^{[2]}(k), \dots, o_m^{[2]}(k))^T$ . У подальшому позначимо цей вектор вхідних значень через



$y(k) = (y_1(k), y_2(k), \dots, y_m(k))^T$ . Робота мережі полягає в обчисленні рівней приналежності поточного спостереження до кластерів, що визначаються вагами нейронів шару Кохонена, обчисленні значень вихідних сигналів нео-фаззі-нейронів та ранжуванні цих значень згідно відповідних ступенів приналежності на виходах шару Кохонена. Виходом мережі є вектор прогнозів  $\tilde{y}(k) = (\tilde{y}_1(k), \tilde{y}_2(k), \dots, \tilde{y}_{n_2}(k))^T$ .

Таким чином, між нейронами шарів Кохонена і шаром нео-фаззі-нейронів встановлюється взаємно однозначна відповідність, а саме: кожному нейрону Кохонена  $\phi_l$  відповідає нео-фаззі-нейрон  $NFN_l$ ,  $l=1, 2, \dots, p$ ,  $p$  – кількість нейронів у прихованих шарах.

Виходи нейронів шару Кохонена обчислюються у вигляді

$$\phi_l(k) = \frac{\Phi_l(k)}{\sum_{i=1}^p \Phi_i(k)}, \quad (1.60)$$

де  $\phi_l(k) = \Phi(\|x(k) - c_l\|_{R_l^{-1}}^2)$  – функції належності нейронів (функції сусідства),

$\|\cdot\|_{R_l^{-1}}$  – норма вектора по матриці  $R_l^{-1}$ .

Таким чином, виходи нейронів шару Кохонена задовольняють умові

$$\sum_{l=1}^p \phi_l(k) = 1.$$

Архітектура нео-фаззі-нейрона зображена на рис. 1.33. Тут

$$q_l(k) = \sum_{i=1}^m f_i^{[l]}(y_i(k)), \quad (1.61)$$

$$f_i^{[l]}(y_i(k)) = \sum_{j=1}^h \mu_{ij}(y_i(k)) w_{ij}^{[l]}(k), \quad (1.62)$$

де  $h$  – кількість функцій приналежності у нелінійному синапсі  $NS_i$ ,

$m$  – довжина вектора  $y$ .

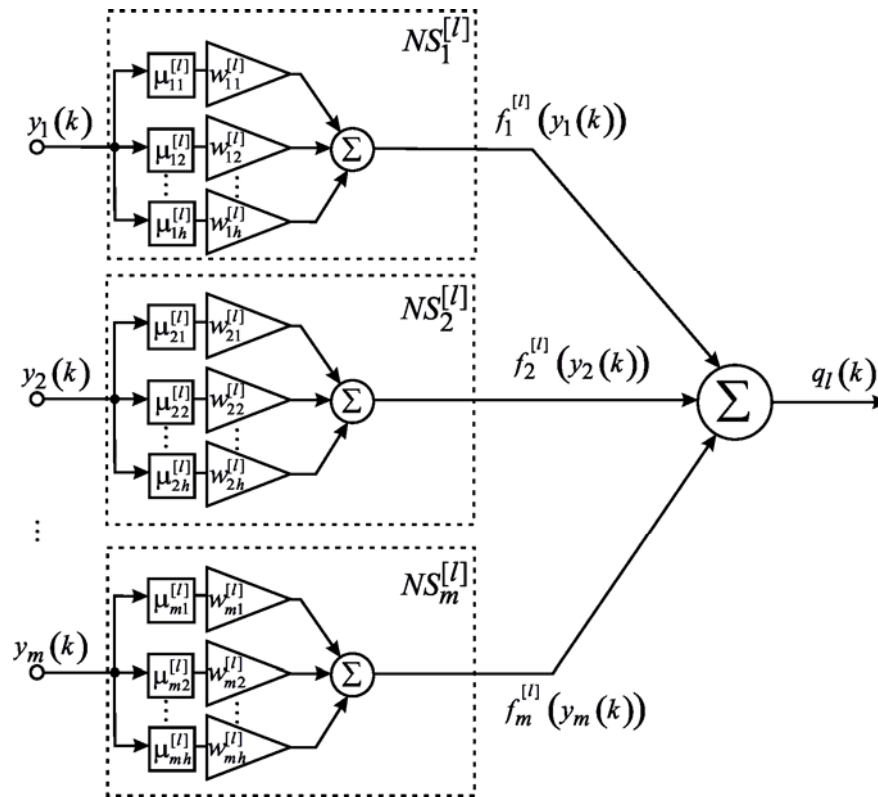


Рисунок 1.33 – Нео-фаззі-нейрон

У якості функцій приналежності в нео-фаззі-нейроні звичайно використовуються трикутні конструкції, а їх значення визначаються відстанню між величиною вхідного сигналу  $y$  та параметрами центрів  $c_{ij}$ . Функції приналежності  $\mu_{ij}$  нелінійних синапсів забезпечують розкладення одиниці на деяким інтервалі  $[a, b]$ . Для трикутних функцій  $\mu_{ij}$  розраховується наступним чином:

$$\mu_{ij}(y_i) = \begin{cases} \frac{c_{ij} - y_i}{c_{ij} - c_{i,j-1}}, & y_i \in (c_{i,j-1}; c_{ij}], \\ \frac{y_i - c_{ij}}{c_{i,j+1} - c_{ij}}, & y_i \in (c_{ij}; c_{i,j+1}], \\ 0, & \text{інакше} \end{cases} \quad (1.63)$$

для  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, h$ .

Позначимо

$$m_{\Sigma} = \sum_{i=1}^m h_i, \quad (1.64)$$

$$M(k) = (\mu_{11}, \mu_{12}, \dots, \mu_{1,h}, \mu_{21}, \mu_{22}, \dots, \mu_{2,h}, \dots, \mu_{m1}, \mu_{m2}, \dots, \mu_{m,h})^T \quad (1.65)$$

– вектор функцій приналежності нео-фаззі-нейронів розмірності  $(m_{\Sigma} \times 1)$  та

$$W_{NFN}^T = \begin{pmatrix} w_{11}^{[1]} & w_{21}^{[1]} & \dots & w_{mh}^{[1]} \\ w_{11}^{[2]} & w_{21}^{[2]} & \dots & w_{mh}^{[2]} \\ \dots & \dots & \ddots & \dots \\ w_{11}^{[p]} & w_{21}^{[p]} & \dots & w_{mh}^{[p]} \end{pmatrix} = (w_{NFN}^{[1]}, w_{NFN}^{[2]}, \dots, w_{NFN}^{[p]})^T \quad (1.66)$$

– матрицю вагів нео-фаззі-нейронів розмірністю  $(h \times m_{\Sigma})$ , при цьому виходом мережі в цілому є вектор:

$$\tilde{y}(k) = (\tilde{y}_1(k), \tilde{y}_2(k), \dots, \tilde{y}_m(k))^T, \quad (1.67)$$

де  $\tilde{y}_l(k) = q_l(k)\phi_l(k)$ ,  $l = 1..m$ .

Навчання багатoshарового персеプトру може бути реалізовано за допомогою будь-якої процедури зворотнього поширення помилки, але у якості навчального образу  $d(k)$  використовується сам вхідний сигнал  $x(k)$ , що треба стиснути.

Власне кажучи, компресія інформації відбувається у другому прихованому шарі. З виходу другого шару береться «стиснутий» сигнал  $y(k) = o^{[2]}(k)$ .

Для вирішення задачі компресії інформації пропонуємо використовувати багат шарову автоасоціативну нейромережу прямого поширення, побудовану на формальних нейронах з поліноміальною функцією активації, де параметр крутості  $\gamma_j(k)$  є постійним і дорівнює одиниці.

$$\psi(\gamma_j(k)u_j(k)) = \gamma_j(k)u_j(k) - \frac{1}{3}\gamma_j^3(k)u_j^3(k). \quad (1.68)$$

Процес навчання нейромережі зводиться до настроювання матриць синаптичних ваг  $W^{[s]}$ ,  $s=1,2,3$ , шляхом мінімізації прийнятої цільової функції, в якості котрої використовується вираз:

$$E(k) = \frac{1}{2} \sum_{j=1}^n E_j(k) = \frac{1}{2} \sum_{j=1}^n e_j^2(k),$$

де  $e_j(k) = x_j(k) - \hat{x}_j(k)$ ,  $\hat{x}(k) - (n \times 1)$  – вихідний векторний сигнал мережі;

$x(k) - (n \times 1)$  – вхідний вектор.

Таким чином, запропонований алгоритм навчання становить собою процедуру настроювання трьох матриць синаптичних ваг, починаючи з вихідного шару та має вигляд:

$$\begin{cases} \Delta w_{ji}^{[3]}(k) = \eta(k)(1 - u_j^{[3]2}(k))e(k)o_i^{[2]}(k), \\ \Delta w_{ji}^{[2]}(k) = \eta(k)(1 - u_j^{[2]2}(k))o_i^{[1]} \sum_{l=1}^n \delta_l^{[3]}(k)w_{li}^{[3]}(k), \\ \Delta w_{ji}^{[1]}(k) = \eta(k)(1 - u_j^{[1]2}(k))x_i(k) \sum_{l=1}^m \delta_l^{[2]}(k)w_{li}^{[2]}(k). \end{cases} \quad (1.69)$$

Навчання шару Кохонена відбувається у режимі самонавчання без використання навчальних сигналів. Для знаходження центрів  $c_l(k)$  нейронів цього шару слід використовувати рекурентний алгоритм кластеризації нечітких с-середніх за Бездеком у модифікації для навчання нечіткої кластеризаційної мережі Кохонена:

$$\begin{cases} \phi_l(k+1) = \frac{\left( \|y(k+1) - c_l(k)\|_{R_l^{-1}}^2 \right)^{\frac{1}{1-\beta}}}{\sum_{i=1}^p \left( \|y(k+1) - c_i(k)\|_{R_i^{-1}}^2 \right)^{\frac{1}{1-\beta}}}, \\ c_l(k+1) = c_l(k) + \tilde{\eta}(k) \phi_l^\beta(k+1) (y(k+1) - c_l(k)), \quad l = 1, 2, \dots, p, \end{cases} \quad (1.70)$$

де  $\beta$  – невід’ємний параметр, «фазификатором», що визначає нечітку межу між кластерами з центроїдами у точках  $c_l(k)$ , що відповідають нейронам шару Кохонена;  $\tilde{\eta}(k)$  – спадний коефіцієнт, визначаючий швидкість навчання.

На другому етапі для навчання параметрів нео-фаззі-нейронів використовується процедура зважених найменших квадратів. Навчальним сигналом для кожного нео-фаззі-нейрона є відповідне значення на вході мережі, а настроювання параметрів виконується з урахуванням виходу відповідного нейрону шару Кохонена в якості вагового коефіцієнту. Процедура навчання в рекурентному вигляді:

$$\begin{cases} w_{NFN}^{[l]}(k) = w_{NFN}^{[l]}(k-1) + \frac{P_l(k-1)(\tilde{y}(k) - M^T(k)w_{NFN}^{[l]}(k-1))}{1 + \phi_l(k)M^T(k)P_l(k-1)M(k)}M(k), \\ P_l(k) = \phi_l(k) \left( P_l(k-1) - \frac{\phi_l(k)P_l(k-1)M(k)M^T(k)P_l(k-1)}{1 + \phi_l(k)M^T(k)P_l(k-1)M(k)} \right), \end{cases} \quad (1.71)$$

де  $M = (M(1), M(2), \dots, M(N))^T$ ,

де  $P_l(k)$  – матриця розмірності  $(m_\Sigma \times m_\Sigma)$ .

## 2 РОЗРОБКА МОДЕЛЕЙ ОБ'ЄКТІВ КОНТРОЛЮ, УПРАВЛІННЯ ТА ІДЕНТИФІКАЦІЇ В АДАПТИВНИХ ЕВОЛЮЦІЙНИХ СИСТЕМАХ

### 2.1 Математичні моделі замкнутих адаптивних еволюційних систем

Динамічні об'єкти описуються диференціальними, інтегральними й функціональними рівняннями д деяких координат, що характеризують їхній стан. Використання засобів обчислювальної техніки для ідентифікації та керування такими об'єктами приводить до необхідності опису досліджуваних об'єктів різницевиими або диференційними рівняннями.

Лінійні динамічні стохастичні об'єкти (рис 2.1) у загальному випадку можуть бути уявлені лінійним різницевим рівнянням:

$$y(k) + \sum_{i=1}^n a_i y(k-i) = \sum_{i=0}^m b_i u(k-i) + \sum_{i=0}^l c_i w(k-i), \quad (2.1)$$

де  $y(k)$  - вихідний сигнал,  $u(k)$  - вхідний сигнал,  $w(k)$  - збурювання (завада);  $a_i (i = 1, \dots, n)$ ,  $b_i (i = 0, 1, \dots, m)$ ,  $c_i (i = 1, \dots, l)$  - параметри об'єкта, що підлягають визначенню;  $k = 0, 1, 2, \dots$  - дискретний час.

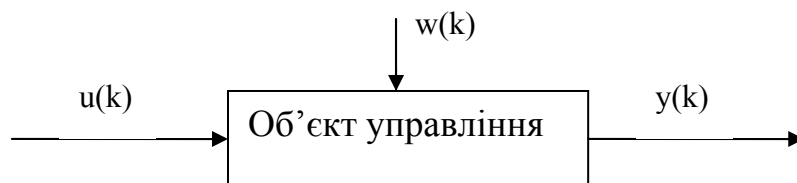


Рисунок 2.1 - Лінійний динамічний стохастичний об'єкт у загальному вигляді

Введення оператора запізнювання  $q^{-1}$ , обумовленого відповідно до умови

$$q^{-d} y(k) = y(k-d), \quad (2.2)$$

дозволяє представити рівняння (2.1) в операторній формі

$$A(q^{-1})y(k) = B(q^{-1})u(k) + C(q^{-1})w(k) \quad (2.3)$$

або

$$y(k) = K_u(q^{-1})u(k) + K_w(q^{-1})w(k), \quad (2.4)$$

де

$$K_u(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})}, \quad K_w(q^{-1}) = \frac{C(q^{-1})}{A(q^{-1})} \quad (2.5)$$

- передатні функції об'єкта по керуванню й збурюванню відповідно.

Задача структурної ідентифікації полягає у визначенні порядку використовуваних в описі поліномів. Задача параметричної ідентифікації складається у визначенні параметрів поліномів, порядок яких уже відомий або оцінений [86]. У цьому випадку всі підлягаючі визначенню параметри  $a_i (i = 1, 2, \dots, n)$ ,  $b_i (i = 1, 2, \dots, m)$ ,  $c_i (i = 1, 2, \dots, l)$  включаються в деякий вектор невідомих параметрів  $\Theta$ .

$$y(k) = K_u(q^{-1}, \Theta)u(k) + K_w(q^{-1}, \Theta)w(k). \quad (2.6)$$

Рівняння (2.1), (2.3), (2.4) є найбільш загальними й охоплюють як приватні випадки різні об'єкти й процеси [87].

Зупинимося на деяких розповсюджених способах опису динамічних об'єктів. Розглянемо авторегресійну модель для опису динамічних об'єктів. Авторегресійні моделі або ARX-Моделі були досліджені в роботах [88-100]. ARX-Моделі являють собою опис у вигляді наступного різницевого рівняння

$$y(k) + a_1y(k-1) + \dots + a_ny(k-n) = b_1u(k) + \dots + b_mu(k-m) + w(k) \quad (2.7)$$

або

$$A(q^{-1})y(k) = B(q^{-1})y(k) + w(k). \quad (2.8)$$

У рівнянні (2.5) білий шум  $w(k)$  входить як неспостережувана величина.

Якщо позначити  $K_u(q^{-1}, \Theta) = \frac{B(q^{-1})}{A(q^{-1})}$  й  $K_v(q^{-1}, \Theta) = \frac{1}{A(q^{-1})}$ , то видно, що (2.8)

збігається з (2.6).

Для подання нестационарного динамічного об'єкта ARX-Моделлю використовують наступне співвідношення:

$$\begin{aligned} y(k) + a_1(k)y(k-1) + \dots + a_n(k)y(k-n) = \\ = b_1(k)u(k-1) + \dots + b_m(k)u(k-m) + w(k) \end{aligned} \quad (2.9)$$

або

$$A(q^{-1}, \Theta(k))y(k) = B(q^{-1}, \Theta(k))u(k) + w(k). \quad (2.10)$$

Дана модель містить залежний від часу вектор параметрів виду, що набудовуються

$$\Theta(k) = (a_1(k), \dots, a_n(k), b_1(k), \dots, b_m(k))^T. \quad (2.11)$$

Позначивши

$$\varphi(k) = (-y(k-1), \dots, -y(k-n), u(k-1), \dots, u(k-m))^T, \quad (2.12)$$

можна переписати (2.10) у такий спосіб:

$$y(k) = \Theta^T(k)\varphi(k) + w(k). \quad (2.13)$$

Отримане співвідношення (2.13) описує модель типу псевдолінійної регресії, для дослідження якої можна застосовувати розвинені методи лінійного регресійного аналізу [88,91]. Слід зазначити, що опис лінійного динамічного



об'єкта у вигляді (2.13) отримав широке поширення, оскільки дозволяє уявити алгоритм ідентифікації й керування в досить простій і наочній формі.

Поряд з авторегресійними моделями широко використовуються авторегресійні моделі ковзного середнього (ARMAX-Моделі). ARMAX-Моделі були введені для вирішення завдань ідентифікації систем Остремом і Болінім і з тих пор такі моделі є одними з найбільш популярних структур.

Основна відмінність ARMAX-Моделі від ARX-моделі (1.9) полягає в доданні більшої гнучкості опису властивостей завади. У даній моделі помилка моделі представляється як ковзне середнє білого шуму, що приводить до наступного співвідношення:

$$\begin{aligned} y(k) + a_1(k)y(k-1) + \dots + a_n(k)y(k-n) = \\ = b_1(k)u(k-1) + \dots + b_m(k)u(k-m) + w(k) + \\ + c_1(k)w(k-1) + \dots + c_l(k)w(k-l). \end{aligned} \quad (2.14)$$

Якщо позначити

$$C(k, q^{-1}) = 1 + c_1(k)q^{-1} + \dots + c_l(k)q^{-l},$$

те рівняння (2.14) можна записати у вигляді

$$A(k, q^{-1})y(k) = B(k, q^{-1})u(k) + C(k, q^{-1})w(k), \quad (2.15)$$

що приводить до (2.6), якщо покласти  $K_u(k, q^{-1}, \Theta) = \frac{B(k, q^{-1}, \Theta)}{A(k, q^{-1}, \Theta)}$ ,

$$K_w(k, q^{-1}, \Theta) = \frac{C(k, q^{-1}, \Theta)}{A(k, q^{-1}, \Theta)}$$

Тоді

$$\Theta(k) = (a_1(k), \dots, a_n(k), b_1(k), \dots, b_m(k), c_1(k), \dots, c_l(k))^T.$$

У цьому випадку ініціалізація процедури оцінювання в момент  $k = 0$

вимагає знання наступних величин:

$y(0), \dots, y(-k^* + 1)$ ,  $k^* = \max(n, l)$ ,  $\hat{y}(0), \dots, \hat{y}(-l + 1)$ ,  $u(0), \dots, u(-m + 1)$ , де  $\hat{y}(k)$  є оцінкою вихідного сигналу моделі.

Якщо ці величини відсутні, то їх можна прийняти рівними нулю, у результаті чого в модель вноситься неточність. Можна також почати рекурсію в момент  $\max(k^*, u(l))$  і включити невідомі початкові умови  $y(k), k = 1, \dots, l$  у вектор  $\Theta$ .

Ввівши позначення

$$\varepsilon(k) = y(k) - \hat{y}(k) \quad (2.16)$$

і

$$y(k) = (-y(k-1), \dots, -y(k-n), u(k-1), \dots, u(k-m), \varepsilon(k-1), \dots, \varepsilon(k-l))^T, \quad (2.17)$$

вираження (2.15) можна записати так:

$$y(k) = \Theta^T(k) \varphi(k) + w(k). \quad (2.18)$$

Використовувану в співвідношенні змінну  $\varepsilon(k)$  називають помилкою прогнозування або узагальненою нев'язкою. Саме рівняння (2.18) називається псевдолінійною регресією в силу нелінійної залежності вектора  $\varphi(k)$  від  $\Theta(k)$ .

Структури, що розглянуто вище, фактично можуть приводити до великої кількості моделей, залежно від того, які із множин  $A, B, C$  включаються в модель. Для зручності можна використовувати узагальнену модельну структуру [92] опису нестационарного об'єкта:

$$A(k, q^{-1})y(k) = \frac{B(k, q^{-1})}{\tilde{B}(k, q^{-1})}u(k) + \frac{C(k, q^{-1})}{\tilde{C}(k, q^{-1})}w(k), \quad (2.19)$$

де

$$\begin{aligned}\tilde{B}(k, q^{-1}) &= 1 + \tilde{b}_1(k)q^{-1} + \dots + \tilde{b}_1(k)q^{-1}; \\ \tilde{C}(k, q^{-1}) &= 1 + \tilde{c}_1(k)q^{-1} + \dots + \tilde{c}_1(k)q^{-1}.\end{aligned}$$

Однак для більшості практичних цілей дана структура є занадто загальною. У додатках звичайно припускають один або трохи з п'яти багаточленів рівними 1. Використання розглянутого вище вектора в загальному випадку нестационарних параметрів  $\Theta(k)$  і узагальненого вектора передісторії  $\varphi(k)$  дозволяє уявити модель (2.19) у вигляді псевдолінійної регресії (2.18). Якщо ж вхідні сигнали  $u(k)$  й виходи  $y(k)$  є векторами  $(p_u \times 1)$  й  $(p_y \times 1)$  відповідно, то найпростішим узагальненням множини моделей (2.6) є

$$\begin{aligned}y(k) + A_1(k)y(k-1) + \dots + A_n(k)y(k-n) = \\ = B_1(k)u(k-1) + \dots + B_m(k)u(k-m) + w(k),\end{aligned}\quad (1.20)$$

де  $A_i(k), B_i(k)$  - матриці розмірностей  $p_y \times p_y$  і  $p_y \times p_u$  відповідно. Аналогічно вводяться матричні багаточлени від змінної  $q^{-1}$ :

$$\begin{aligned}A(k, q^{-1}) &= A_1(k)q^{-1} + \dots + A_n(k)q^{-n}; \\ B(k, q^{-1}) &= B_1(k)q^{-1} + \dots + B_m(k)q^{-m}.\end{aligned}$$

У цьому випадку виходить не вектор, а матриця шуканих параметрів  $\Theta(k)$  розмірності  $((np_y + mp_u)p_y)$ :

$$\Theta(k) = [A_1(k), A_2(k), \dots, A_n(k), B_1(k), \dots, B_m(k)]^T \quad (2.21)$$

і

$$\varphi(k) = [-y^T(k-1), \dots, -y^T(k-n), u^T(k-1), \dots, u^T(k-m)]^T \quad (2.22)$$

- вектор-стовпець розмірності  $(n \times p_y + m \times p_u) \times 1$ , що дозволяє представити (2.21) матричним рівнянням лінійної регресії (2.13).

Ці формули можна розглядати як сукупність  $p_y$  лінійних регресій, записаних одна під іншою з тим самим вектором входів. Таким чином, з вищевикладеного випливає, що опис лінійного динамічного стохастичного об'єкта рівнянням (2.13) є досить загальним. З іншого боку, подання досліджуваного об'єкта моделлю (2.13) дозволяє використовувати для ідентифікації параметрів добре розроблені методи регресійного аналізу й методи ідентифікації статичних об'єктів.

Однак застосування методів регресійного аналізу вимагає виконання певних припущень щодо статистичних властивостей корисних сигналів і завад. Зокрема, передбачається, що завада являє собою випадковий вектор  $\Psi$  з нульовим математичним очікуванням і діагональною коваріаційною матрицею:

$$M\{\Psi(k)\} = 0, \quad i = 1, 2, \dots, k, \quad \text{cov}\{\Psi(k)\} = \sigma_w^2 I. \quad (2.23)$$

де  $M\{\bullet\}$  – символ математичного очікування,

$\sigma_w^2$  – дисперсія перешкоди,

$I$  – одинична матриця розмірності  $k \times k$ .

Всі обмеження, пов'язані із класичним регресійним аналізом, визначаються традиційним критерієм квадратів помилок і відповідно методом найменших квадратів, досить жорстко прив'язаним до нормального закону розподілу змінних і збурень.

Якщо ж щодо завади відомо тільки, що вона обмежена по амплітуді

$$|w(k)| < \delta, \quad (2.24)$$

то для оцінювання параметрів застосовуються інші методи, які не використовують ніякої інформації про статистичні властивості збурень крім їхньої приналежності деякому обмеженому інтервалу.

Для оцінювання параметрів при наявності обмежених по амплітуді завад (2.24) найбільш широке поширення отримали алгоритми, в основі побудови

яких лежить метод політопів (і як окремий випадок - ортогопов), алгоритми, що містять зону нечутливості й алгоритми на основі побудови еліпсоїдів. Це напрямок у цей час бурхливо розвивається.

## 2.2 Рекурентні алгоритми ідентифікації, побудовані на системах нерівностей

Розглянемо об'єкт, що описується рівнянням псевдолінійної регресії (2.13):

$$y(k) = \Theta^T(k)\varphi(k) + w(k)$$

с завадою  $w(k)$ , що задовольняє обмеженню (2.24)

$$|w(k)| < \delta.$$

Тоді рівняння (2.13) може бути переписане у вигляді

$$\varepsilon(k) = y(k) - \Theta^T(k)\varphi(k)$$

с вектором шуканих параметрів  $\hat{\Theta}(k)$ , що задовольняють системі нерівностей

$$(y(k) - \Theta^T(k)\varphi(k))^2 \leq \delta^2 \quad \forall k = 1, 2, \dots$$

В якості оцінки  $\hat{\Theta}(k)$  можуть бути використані тільки ті, які належать множині:

$$S(k) = \left\{ \hat{\Theta} : (y(k) - \hat{\Theta}^T \varphi(k))^2 \leq \delta^2 \right\}. \quad (2.25)$$

Невизначені параметри  $\Theta$  належать області, обмеженої гіперплощинами (2.25). Область  $S(k)$  є простий зв'язковий областю. Простір  $S(k)$  є опуклим політопом, що може стати досить складним, якщо  $k$  - велико (рис 2.2).

З геометричної точки зору множина  $S(k)$  являє собою монотонно зростаючу послідовність перетинань опуклих політопів  $F(k)$ :

$$S(k) = \bigcap_{j=1}^k F_j = S(k-1) \cap F(k); \quad (2.26)$$

$$F(k) = \left\{ \hat{\Theta} : \left| y(k) - \hat{\Theta}^T \varphi(k) \right| \leq \delta \right\}. \quad (2.27)$$

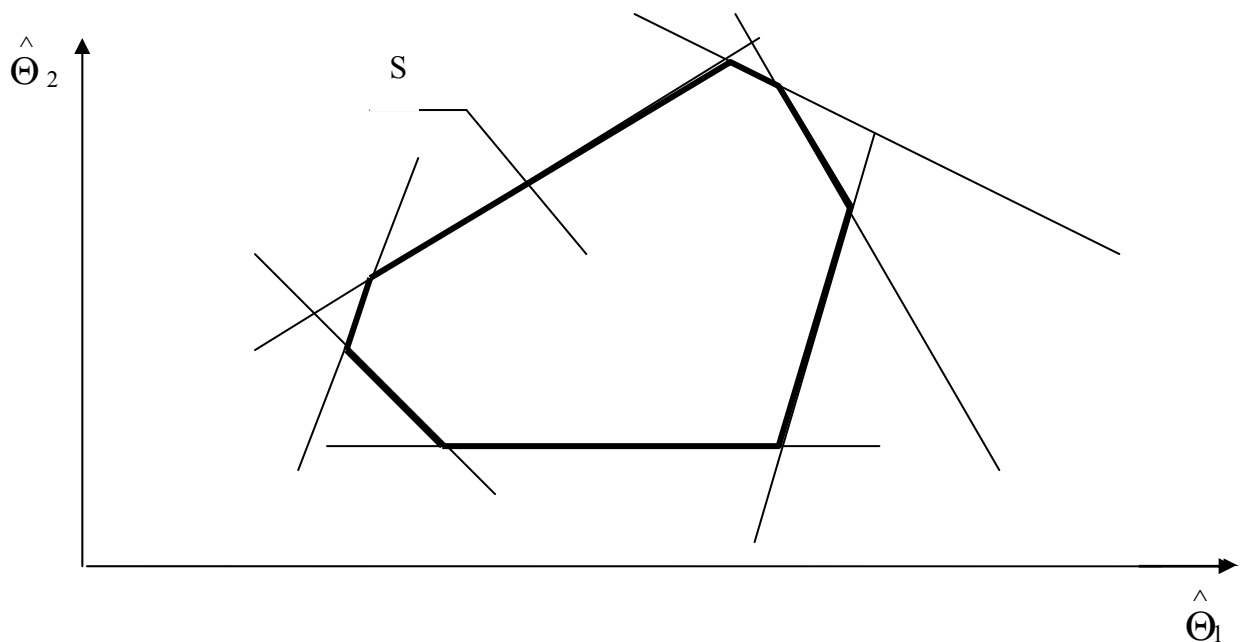


Рисунок 2.2 - Знаходження області припустимих значень  $S_k$

Обчислення оцінок  $\hat{\Theta} \in S(k)$  являє собою складну задачу, рішення якої може бути істотно спрощене шляхом побудови деякої множини, що обмежує

$S(k)$  [93]. Ці обмеження можуть задаватися у формі ортогопів. Задача побудови алгоритму ідентифікації полягає в пошуку процедури рекурентного уточнення  $\hat{\Theta}(k)$  на основі наявних значень  $\hat{\Theta}(k-1)$  при надходженні нових спостережень вхідних і вихідних сигналів. Область невизначених параметрів  $S(k)$ , обмежена гіперплощинами (2.25), може бути апроксимована ортогопом, грані якого паралельні осям. Геометрична ілюстрація такого підходу наведена на рис.2.3.

Границі  $\Theta_{\min}$  й  $\Theta_{\max}$  невизначеного інтервалу пов'язані з  $i$ -м параметром і дають уявлення про мінімальну й максимальну величини критерію :

$$J_i(\Theta) = \Theta_i, \quad (2.28)$$

де припустима область параметрів  $\Theta$  є область  $S(k)$ .

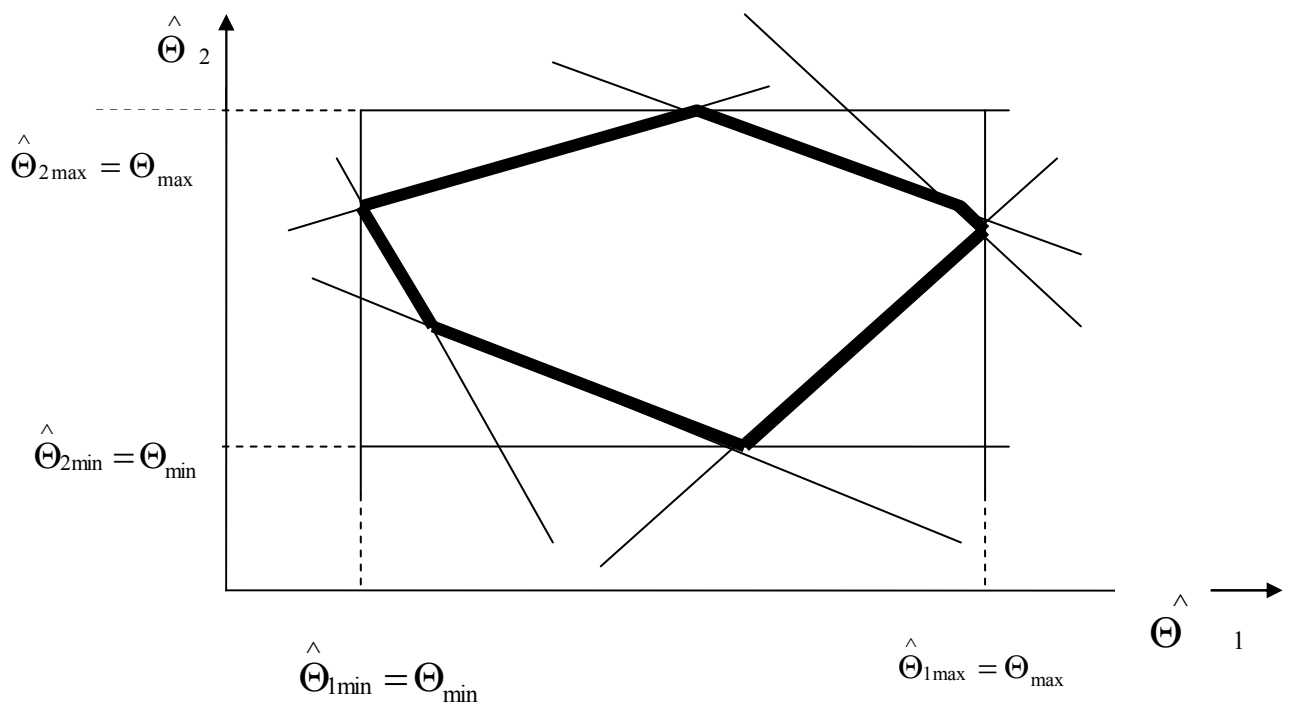


Рисунок 2.3 - Апроксимація невизначеного інтервалу ортогопом, грані якого паралельні осям

Критерій (2.28) визначається набором  $2k$  лінійних нерівностей. Кожна границя, отже, отримана рішенням задачі лінійного програмування. Обчислення границь ортогопа вимагає рішення  $2p$  задач лінійного програмування, де  $p$  - розмірність простору. Кожна із задач лінійного програмування містить  $2k$  лінійних нерівностей. Результуючий алгоритм не є рекурентним і пов'язаний з великою кількістю більш складних обчислень, ніж алгоритми методу найменших квадратів (МНК).

Якщо область  $S(k)$  є вузькою й орієнтованою під більшим кутом до осей, то обмеження осьовим ортогопом може давати дуже неточні результати. Однак можна спробувати працювати з ортогопом мінімального об'єму.

Як відзначалося раніше, обчислення оцінок  $\Theta \in S(k)$  являє собою складну задачу, рішення якої може бути істотно спрощене шляхом побудови деякої множини, що обмежує  $S(k)$  у формі еліпсоїдів [93], центр яких збігається с.  $\hat{\Theta}(k)$

У задачах ідентифікації поширення отримали алгоритми, засновані на побудові еліпсоїдів, що використовують наступні властивості одиничного еліпсоїда :

$$E = \{x \in R^p : x^T A^{-1} x \leq 1\}, A = A^T > 0$$

- осі еліпсоїда утворюють ортогональну систему власних векторів матриці  $A$ , а півосі рівні  $\lambda_i^{1/2}$ , де  $\lambda_i$  - власні значення  $A$  ( $i = 1, 2, \dots, p$ );

- об'єм еліпсоїда пропорційний добутку півосей, тобто пропорційний  $\sqrt{\det A}$ ;

- сума квадратів півосей  $\sum_{i=1}^k (\lambda_i^{1/2})^2$  дорівнює сліду матриці  $A$ .

Нехай



$$E(k-1) = \left\{ \Theta : (\Theta - \hat{\Theta}(k-1))^T P^{-1}(k-1) (\Theta - \hat{\Theta}(k-1)) \leq r^2 \right\} \quad (2.45)$$

деякий еліпсоїд, що містить перетинання  $S(k-1)$ , тобто  $S(k-1) \in E(k-1)$ . Тут  $\hat{\Theta}(k)$  є центром еліпсоїда  $E(k-1)$  й  $P(k-1)$  - матриця, що визначає півосі, (тобто орієнтацію й розміри еліпсоїда). Матриця  $P(k-1)$  є симетрично - перестановочною, причому  $P(k-1) = P^T(k-1) > 0$ . Для виріджених еліпсоїдів обмеження відповідають двом паралельним гіперплощинам, де  $\hat{\Theta}(k-1)$  визначається неєдиним образом і  $P(k-1)$  - матриця з одиничним рангом.

Центр еліпсоїда  $\hat{\Theta}(k-1)$  є оцінкою шуканого вектора параметрів  $\Theta$  у момент часу  $k-1$ . Надходження нових вимірів  $y(k)$  і  $\varphi(k)$  дає наступне значення  $F_k(\Theta)$ , а перетинання (2.26) внаслідок того, що  $S(k-1) \in E(k-1)$ , повністю входить у перетинання  $E(k-1) \cap F(k)$ . Новий еліпсоїд будується так, щоб

$$\{E(k-1) \cap F(k)\} \subset E(k). \quad (2.46)$$

Крім того, нова оцінка повинна забезпечувати умову

$$E_k \leq E_{k-1}. \quad (2.47)$$

Таким чином, задача побудови алгоритму полягає в пошуку такого рекурентного уточнення  $\hat{\Theta}(k), P^{-1}(k), r^2$  на основі наявних значень  $\hat{\Theta}(k), P^{-1}(k), r^2$  при надходженні нових спостережень вхідних і вихідних сигналів, щоб забезпечувалося виконання (2.46) і (2.47). Геометрична ілюстрація цього підходу зображена на рис. 2.6.

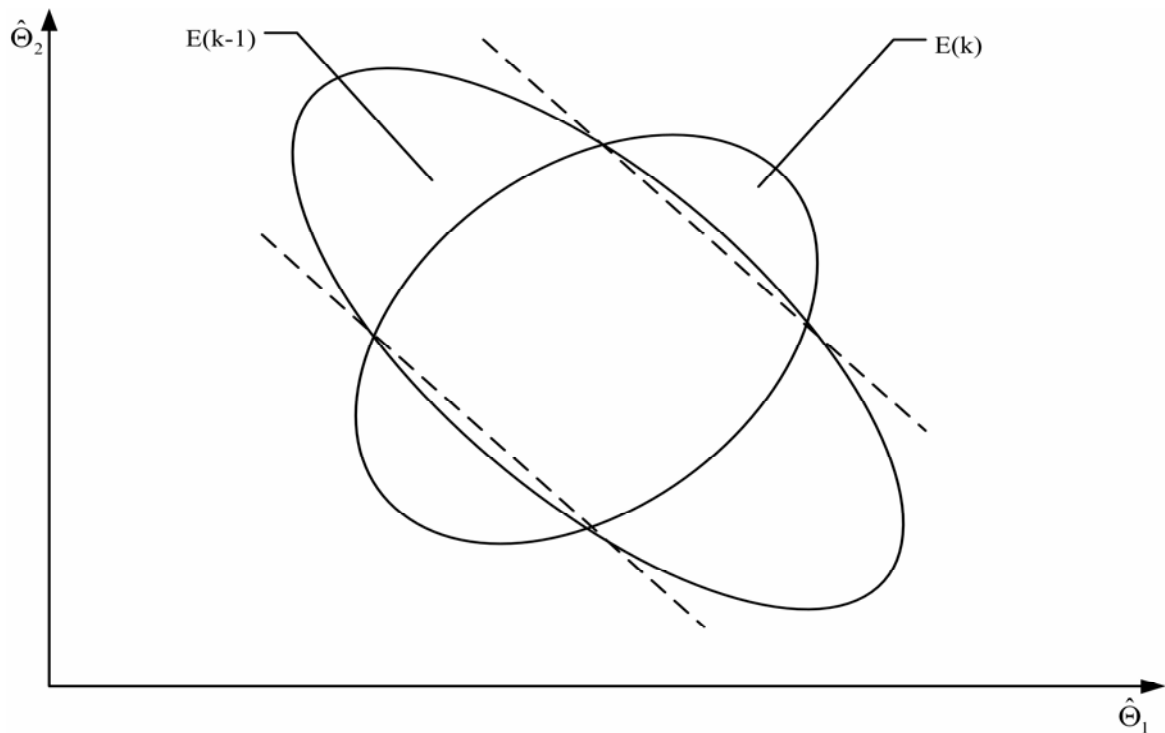


Рисунок 2.6 - Обмеження еліпсоїдом

Еліпсоїд  $E(k)$  побудований з початковими умовами  $\Theta_0 = 0, E_0 = \alpha I$ , де  $I$  - одинична матриця розміром  $p \times p$ , і  $\alpha$  - деяке досить велике число. Це значить, що початковий еліпсоїд  $E_0$  є сферою радіусом  $\alpha^{1/2}$ . Вираження для побудови  $E(k)$  виглядає в такий спосіб:

$$\Theta_0 = \Theta, \quad P_0 = \alpha I,$$

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \frac{P(k-1)(y(k) - \varphi^T(k)\hat{\Theta}(k-1))}{1 + \varphi^T(k)P(k-1)\varphi(k)}\varphi(k), \quad (2.48)$$

$$P(k) = P(k-1) - \frac{P(k-1)\varphi(k)\varphi^T(k)P(k-1)}{1 + \varphi^T(k)P(k-1)\varphi(k)}, \quad (2.49)$$

що збігається за структурою з рекурентним методом найменших квадратів (РМНК).

Для мінімізації геометричних розмірів еліпсоїда  $E(k)$  використовують

такі критерії як, наприклад, мінімальний об'єм або мінімальний слід. Отриманий при цьому алгоритми називаються відповідно «алгоритмом мінімального об'єму» (або «алгоритмом, що мінімізує об'єм») і «алгоритмом мінімального сліду» (або «алгоритмом, що мінімізує слід»).

У роботі [94] була запропонована рекурентна процедура обчислення еліпсоїда  $E(k)$ :

$$\begin{aligned} E(k) = \{ \Theta : \alpha(k)(\Theta - \hat{\Theta}(k-1))^T P^{-1}(k-1)(\Theta - \hat{\Theta}(k-1)) + \\ + \beta(k)(y(k) - \hat{\Theta}^T(k-1)\varphi(k))^2 \leq \alpha^2(k) + \beta(k)r^2 \} = \\ = \left\{ \Theta : (\Theta - \hat{\Theta}(k-1))^T P^{-1}(k-1)(\Theta - \hat{\Theta}(k-1)) \leq r^2 \right\}, \end{aligned} \quad (2.50)$$

де  $\alpha(k) \in (0,1)$  – параметр забування застарілої інформації;

$\beta(k) \in (0,1)$  – параметр зважування (селекції) знову вступник інформації.

Задача побудови алгоритму зводиться до пошуку центру  $\hat{\Theta}(k)$  еліпсоїда  $E(k)$ . Після перетворень приходимо до стандартного оптимального алгоритму, побудованому на основі обмежених еліпсоїдів:

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \beta(k)P(k)\varphi(k)e(k); \quad (2.51)$$

$$P(k) = \frac{1}{\alpha(k)} \left( P(k-1) - \frac{\beta(k)P(k-1)\varphi(k)\varphi^T(k)P(k-1)}{\alpha(k) + \beta(k)\varphi^T(k)P(k-1)\varphi(k)} \right); \quad (2.52)$$

$$e(k) = y(k) - \hat{\Theta}(k)\varphi(k). \quad (2.53)$$

Різні модифікації алгоритму (2.50)-(2.53) можуть бути отримані шляхом завдання правил настроювання вільно обраних параметрів алгоритму  $\alpha(k)$  й  $\beta(k)$ .

У роботі [95] було запропоновано вибирати параметри в такий спосіб:

$$\alpha(k) = \frac{1}{r^2}, \quad \varphi(k) = \frac{\lambda(k)}{r^2}. \quad (2.54)$$

Змінна  $\lambda(k)$ , котра визначається шляхом мінімізації геометричних розмірів еліпсоїда  $E(k)$ , залежить як від номера кроку, так і від значень вхідних змінних. Принципово можна прийняти  $\lambda(k) = 0$ , однак це означає, що виміри, що знову надійшли, не несуть ніякої нової інформації. У цьому випадку розміри еліпсоїда не змінюються й уточнення оцінок не відбудеться.

Еліпсоїдальні алгоритми отримали поширення не тільки в задачах ідентифікації, але й у задачах контролю при виявленні розладнань і контролю за змінами параметрів. Разом з тим ці алгоритми в силу їхньої чисельної громіздкості погано пристосовані для роботи в реальному часі, а отже, їхнє використання в задачах контролю й керування в реальному часі є досить проблематичним. У зв'язку із цим доцільним є синтез алгоритмів ідентифікації чисельно простих, швидкодіючих і пристосованих для роботи в реальному часі в замкнутому контурі адаптивної системи керування.

На сьогоднішній день проблема керування технічними об'єктами в умовах невизначеності є однією із центральних проблем сучасної теорії керування. Адекватним математичним апаратом для вирішення цієї проблеми є теорія адаптивних систем керування, а широке поширення мікропроцесорної техніки привело до розвитку дискретних адаптивних систем керування [96].

Застосування традиційних методів теорії автоматичного керування на практиці ускладнюється тим, що багато істотних властивостей керованих об'єктів і умови їхнього функціонування заздалегідь невідомі. У той же час знання цих властивостей і умов необхідно для синтезу закону керування. Отже, система керування в процесі роботи повинна накопичувати відсутню інформацію й на її основі постійно поліпшувати закон керування. Ця властивість є визначальним для адаптивних систем.

Разом із цим, застосування принципів адаптації в порівнянні з неадаптивним керуванням дозволяє забезпечити високу якість керування в умовах непередбаченої зміни характеристик об'єкта, що підданий впливу неконтрольованих завад, підвищити надійність системи керування, забезпечити її інваріантність до різних типів об'єктів, скоротити строки розробки й налагодження систем. Все це дозволить здійснити перехід від індивідуальних методів проектування систем керування до масових, котрий викликаний необхідністю різкого збільшення кількості розроблювальних систем і зводиться до широкого використання засобів типового проектування.

Значні функціональні можливості, що забезпечують ефективне керування багатомірними нестационарними об'єктами, що обурюються завадами невідомої природи, дозволяють вирішувати задачі моделювання й контролю складних об'єктів, адаптації синтезованих моделей до реальних даних, відшукування оптимальних законів керування й реалізації їх за допомогою більш простих регуляторів. У принципі будь-яка задача автоматичного керування перетворюється в задачу адаптивного керування, якщо ряд характеристик об'єкта й умови його функціонування невідомі, однак навіть ті задачі, які «уписуються» у рамки традиційної теорії, значно простіше можуть бути вирішені за допомогою адаптивного підходу за рахунок відсутності етапу формалізації апріорних відомостей.

Процес синтезу адаптивної системи складається з ряду етапів, при цьому в якості апріорної інформації, як правило, приймається структурний опис моделі об'єкта, заданого з точністю до параметрів. У більш складному випадку, коли структура моделі покладається невідомою, доводиться вирішувати задачу структурної ідентифікації або оперувати із множиною можливих різних структур.

На першому етапі синтезу вибирається структура керуючого пристрою (регулятора), що багато в чому визначається прийнятими цілями керування. Другий етап пов'язаний з формуванням множини параметрів, що набудовуються. Далі треба провести вибір алгоритму адаптації й, нарешті, на

четвертому етапі здійснюється обґрунтування його вибору. Оскільки адаптивна система в загальному випадку є нелінійною й нестационарною, таке обґрунтування може представляти певні труднощі. На жаль, більшість практичних робіт з адаптивних систем не мають строгої теоретичної основи. Цей напрямок ще вимагає свого подальшого розвитку, створення нових алгоритмів адаптації, нових методів проектування й дослідження. Варто враховувати, що більшість алгоритмів адаптації забезпечує збіжність до оптимальних режимів при нескінченному інтервалі керування. Тому гостро постає проблема синтезу прискорених алгоритмів, що забезпечують високу якість керування на малих інтервалах і в той же час досить простих, щоб забезпечити їхню реалізацію на мікропроцесорних засобах.

Будь-яка адаптивна система керування містить, як мінімум, два контури. Внутрішній контур - це контур регулювання, зовнішній - контур адаптації. Можливо кінцеве введення додаткових контурів, що розширює функціональні можливості системи, однак значно ускладнює її дослідження.

Ключовою властивістю адаптивних систем повинна бути здатність спостереження за дрейфом параметрів об'єкта. Слід зазначити, що задачі керування нестационарними об'єктами є винятковою прерогативою теорії адаптивних систем. У випадку повільного дрейфу проводиться дисконтування застарілої інформації за допомогою алгоритмів з кінцевою пам'яттю. Тут, однак, виникає досить складна задача встановлення компромісу між фільтруючими та спостережуваними властивостями алгоритмів, що до теперішнього часу далека від свого вирішення. У випадку швидкого дрейфу доводиться прибігати до спеціальних процедур, пов'язаних з тією або іншою формою параметризації дрейфу.

До теперішнього часу сформувався ряд щодо незалежних напрямків у теорії адаптивних систем. Тут, насамперед, варто виділити адаптивні регулятори з мінімальною дисперсією, адаптивні регулятори з узагальненою мінімальною дисперсією, адаптивні системи з необхідним розміщенням нулів і полюсів (основні роботи), системи з адаптивними випереджувачами.

У всіх цих підходах передбачається, що діючи у системі збурювання мають стохастичну природу, причому це, як правило, білий шум з нульовим математичним очікуванням і обмеженою дисперсією. У практичних ситуаціях статистичні передумови найчастіше є надуманими, у зв'язку із чим набагато більш реальними представляються допущення лише щодо обмеженості шуму або його різниць по амплітуді. У цих умовах використання методів ідентифікації, заснованих на квадратичних критеріях і, насамперед, рекурентного методу найменших квадратів є неефективним. Виникаючі утруднення частково можуть бути переборені в рамках адаптивних робастних систем керування [97], у яких, проте, однаково «заховані» певні статистичні передумови.

У зв'язку із цим представляється доцільним розробити адаптивні критичні методи контролю, ідентифікації й керування динамічними об'єктами, що функціонують в умовах істотної невизначеності щодо характеристик об'єкту й навколишнього середовища на основі об'єднання принципів теорії адаптивного й критичного керування.

### **2.3 Розробка моделей контролю, ідентифікації та управління в умовах інтервальної невизначеності**

Розглянемо динамічний об'єкт, що функціонує в замкнутій системі керування  $S_D(P,C)$ , описуваний різницеvim рівнянням

$$A(q)y(k) = q^{-d}B(q)u(k) + w(k), \quad (2.55)$$

де поліноми  $A(q) \in R[q,n]$  з  $a_0 = 1$ ,  $B(q) \in R[q,m]$ ;

$d$  – час чистого запізнювання  $d \in N^+$ ;

$y$ ,  $u$  і  $w$  – вихідний, керуючий і збуджуючий сигнали відповідно.

Щодо збурювань передбачається  $w \in D(0, \delta)$ , тобто обмеженість їхніх перших різниць.

У випадку, якщо параметри об'єкта апріорі відомі й незмінні, задача критичного керування може бути вирішена за допомогою регулятора, тобто

$$\Delta F(q)B(q)u(k) = -E(q)y(k) , \quad (2.56)$$

де поліноми  $F(q) \in R[q, d-1]$  з  $f_0 = 1$  і  $E(q) \in R[q, n]$  задаються рівнянням.

У тому випадку, якщо параметри об'єкту невідомі, можна скористатися тим або іншим методом ідентифікації, а потім застосувати закон керування (2.55), у якому щирі значення параметрів об'єкта замінені їхніми оцінками. У цьому й складається суть адаптивного підходу до проектування систем керування об'єктами, що функціонують в умовах невизначеності. Як правило, в якості процедури ідентифікації використовуються ті або інші модифікації рекурентного методу найменших квадратів або проекційні алгоритми, так чи інакше пов'язані із квадратичними критеріями. Використання критеріїв, відмінних від квадратичних, наприклад, модульних, хоча й приводить до робасних процедур, статистичний зміст задачі ідентифікації проте зберігається. Природно, що такі алгоритми ідентифікації не можуть бути використані в критичних системах керування.

У зв'язку із цим виникає необхідність синтезу адаптивних алгоритмів ідентифікації, не зв'язаних ні з якими статистичними передумовами, що мають високу швидкість збіжності, обчислювальну простоту й придатні для роботи в реальному часі в контурі критичної системи керування динамічним об'єктом.

Розглянемо поліном

$$G(q) = 1 - \Delta A(q) , \quad (2.57)$$

де  $G(q) = g_1q^{-1} + g_2q^{-2} + \dots + q^{-n-1}$ , і перепишемо рівняння об'єкта (2.55) у вигляді



$$y(k) = \Theta^T \psi(k-1) + \Delta w(k) , \quad (2.58)$$

де  $\Theta = (g_1, g_2, \dots, g_{n+1}, b_0, b_1, \dots, b_m)^T$  ;

$$\psi(k-1) = (y(k-1), y(k-2), \dots, y(k-n-1), \Delta u(k-d), \Delta u(k-d-1), \dots, \Delta u(k-d-m))^T ;$$

$$\Delta u(k) = u(k) - u(k-1) ;$$

$$\Delta w(k) = w(k) - w(k-1) .$$

Помітимо також, що символом «кришка»  $\hat{x}$  надалі ми будемо позначати оцінку невідомого параметра  $x$  .

**2.3.1 РМНК і умови його збіжності.** РМНК отримав, мабуть, найбільш широке поширення при вирішенні задач ідентифікації, прогнозування й адаптивного керування.

Існують різні підходи до тримання РМНК, серед яких найпоширенішим є побудова РМНК зі звичайного МНК на підставі блокового подання матриці спостережень і використанні леми про обіг матриць. Модифікація квадратичного функціоналу, що мінімізується, (введення вагових параметрів, механізму експонентного згладжування тощо.) приводить до відповідної модифікації РМНК.

Однак РМНК може бути отриманий безпосередньо шляхом мінімізації деякого квадратичного функціоналу. Тому що для аналізу питань збіжності рекурентних алгоритмів широко використовуються критерії (функція Ляпунова) виду  $\tilde{\Theta}^T P^{-1} \tilde{\Theta}$ , де  $\tilde{\Theta}$  - вектор помилок ідентифікації,  $P^{-1}$  - позитивно певна матриця, то доцільно скористатися таким критерієм для отримання самого алгоритму РМНК.

Такий підхід було застосовано у роботах [98-100]. Як показано в [89], мінімізація функціоналу

$$I(\Theta) = (\Theta - \hat{\Theta}(k-1))^T P(k-1) (\Theta - \hat{\Theta}(k-1) + \gamma(y(k) - \Theta^T \psi(k-1)))^2 \quad (2.59)$$

приводить до наступного алгоритму:

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \gamma(k)P(k)\psi(k-1)(y(k) - \hat{\Theta}^T(k-1)\psi(k-1)); \quad (2.60)$$

$$P^{-1}(k) = P^{-1}(k-1) + \gamma(k)\psi(k-1)\psi^T(k-1). \quad (2.61)$$

Застосування до останнього співвідношення леми про обіг матриць дає

$$\begin{aligned} P(k) &= [P^{-1}(k-1) + \gamma(k)\psi(k-1)\psi^T(k-1)]^{-1} = \\ &= P(k-1) - \gamma(k) \frac{P(k-1)\psi(k-1)\psi^T(k-1)P(k-1)}{1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)}. \end{aligned} \quad (2.62)$$

Підставляючи (2.61) в (2.59), одержуємо формулу для рекурентного обчислення оцінки

$$\begin{aligned} \hat{\Theta}(k) &= \hat{\Theta}(k-1) + \\ &+ \gamma(k) \frac{P(k-1)\psi(k-1)}{1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)} (y(k) - \hat{\Theta}^T(k-1)\psi(k-1)). \end{aligned} \quad (2.63)$$

У цьому випадку модифікований алгоритм РМНК описується співвідношеннями (2.62), (2.63), з яких при  $\gamma(k) = 1$  треба звичайний РМНК. Вибираючи різні значення параметра  $\gamma(k)$ , одержуємо різні модифікації РМНК.

Визначимо вимоги, яким повинен задовольняти параметр  $\gamma(k)$  для забезпечення збіжності алгоритму.

Розглянемо функцію Ляпунова виду

$$V(k) = \tilde{\Theta}^T(k)P^{-1}(k)\tilde{\Theta}(k). \quad (2.64)$$

Віднімаючи з обох частин (2.60)  $\Theta$  і позначаючи  $\tilde{\Theta}(i) = \hat{\Theta}(i) - \Theta$ , отримаємо рекурентне вираження для  $\tilde{\Theta}(k)$ , підстановка якого в (2.64) дає

$$\begin{aligned} V(k) &= (\tilde{\Theta}(k-1) - \gamma(k)P(k)\psi(k-1)e(k))^T P^{-1}(k)(\tilde{\Theta}(k-1) - \\ &- \gamma(k)P(k)\psi(k-1)e(k)) = \tilde{\Theta}^T(k-1)P^{-1}(k)\tilde{\Theta}(k-1) - \\ &- \gamma(k)\tilde{\Theta}^T(k-1)\psi(k-1)e(k) + \gamma^2(k)\psi^T(k-1)P(k)\psi(k-1)e^2(k) \quad . \end{aligned} \quad (2.65)$$

Прийmemo до уваги, що  $e(k) = \tilde{\Theta}^T(k-1)\psi(k-1) + w(k)$  й матриця  $P(k)$  пов'язана зі  $P(k-1)$  співвідношенням (2.62),  $P^{-1}(k)$  з  $P^{-1}(k-1)$  - співвідношенням (2.61), одержуємо

$$\begin{aligned} V(k) &= \tilde{\Theta}^T(k-1)P^{-1}(k)\tilde{\Theta}(k-1) + \\ &+ \gamma(k)(\tilde{\Theta}^T(k-1)\psi(k-1))^2 - 2\gamma(k)(\tilde{\Theta}^T(k-1)\psi(k-1))e(k) + \\ &+ \gamma^2(k)\psi^T(k-1)\left[P(k-1) - \gamma(k)\frac{P(k-1)\psi(k-1)\psi^T(k-1)P(k-1)}{1 - \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)}\right] \cdot \\ &\cdot \psi(k-1)e^2(k) = V(k-1) + \gamma(k)w^2(k) - \gamma(k)e^2(k) + \\ &+ \gamma^2(k)\frac{\psi^T(k-1)P(k-1)\psi(k-1)e^2(k)}{1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)} = \\ &= V(k-1) + \gamma(k)w^2(k) - \gamma(k)\frac{e^2(k)}{1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)}. \end{aligned}$$

Якщо  $|w(k)| \leq \delta$ , то

$$V(k) \leq V(k-1) + \gamma(k)\delta^2 - \gamma(k)\frac{e^2(k)}{1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)}$$

і для збіжності алгоритму, тобто для виконання умови

$$\Delta V(k) = V(k) - V(k-1) \leq 0 \quad (2.66)$$

повинна виконуватися нерівність

$$\gamma(k)\delta^2 - \gamma(k) \frac{e^2(k)}{1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)} \leq 0. \quad (2.67)$$

У цьому випадку функція Ляпунова  $V(k)$  є ненегативною й обмеженою.

Її обмежує  $V(0) = \tilde{\Theta}^T(0)P^{-1}(-1)\tilde{\Theta}(0)$ . А оскільки  $V(k) \geq 0$ , то послідовність  $V(k)$  сходиться.

Таким чином, з (2.66) і (2.67) виникає

$$\lim_{k \rightarrow \infty} \frac{\delta^2(1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)) - e^2(k)}{1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)} = 0; \quad (2.68)$$

$$\delta^2(1 + \gamma(k)\psi^T(k-1)P(k-1)\psi(k-1)) - e^2(k) \leq 0.$$

Таким чином, встановлення збіжності модифікацій РМНК при різному виборі параметра  $\gamma(k)$  зводиться до перевірки виконання нерівності (2.68).

**2.3.2 Модифікований РМНК с критичними властивостями.** Задача ідентифікації з позиції теорії критичних систем зводиться до знаходження оцінок невідомого вектору параметрів  $\Theta$  таких, що

$$D(\hat{\Theta}) = \{\hat{\Theta} : |y(k) - \hat{\Theta}^T \psi(k-1)| \leq \delta, \forall k \in \mathbb{N}\}. \quad (2.69)$$

Нині склався ряд підходів до задачі ідентифікації, пов'язаної з нерівністю (2.69). Це, насамперед, підхід Фогеля-Хуанга [101], в основі якого лежать деякі геометричні побудови. Відома також процедура Лозано-Ліла-Ортеги [102,103], синтезована як на геометричних передумовах, так і виходячи з умов стійкості процесу збіжності. Не можна не відзначити також алгоритм Канудас де Віта-

Каррільо [104], що є деякою модифікацією експоненціально зваженого рекурентного методу найменших квадратів. Незважаючи на ефективність цих процедур, їхнє використання в еволюційних системах натрапляє на серйозні утруднення.

По-перше, оптимальний алгоритм Фогеля-Хуанга настільки складний з обчислювальної точки зору, що не може бути й мови про його використання в режимі реального часу. Ця складність обумовлена, насамперед, необхідністю відшукування на кожній ітерації глобального мінімуму багатоекстремальної функції  $n + m + 2$  змінних, що саме по собі є досить складною проблемою.

По-друге, в алгоритмі Лозано-Лила-Ортеги, що має вид

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \frac{\alpha P(k)\psi(k-1)}{1 + \psi^T(k-1)P(k-1)\psi(k-1)} (|e(k)| - \delta_i) \text{sign}e(k), \quad (2.70)$$

$$P^{-1}(k) = \begin{cases} P^{-1}(k-1) + \frac{\alpha P(k-1)\psi(k-1)}{(1 + \psi^T(k-1)P(k-1)\psi(k-1))e(k)} \cdot (|e(k)| - \delta_i) \text{sign}e(k), & |e(k)| > \delta_i; \\ P^{-1}(k-1), & |e(k)| \leq \delta_i. \end{cases} \quad (2.71)$$

$$\delta_i = \sqrt{1 + \alpha} \delta, \quad \alpha \in (0,1) \quad (2.72)$$

$$e(k) = y(k) - \hat{\Theta}^T(k-1)\psi(k-1), \quad (2.73)$$

апріорі передбачається обмеженість значення  $\psi^T(k-1)P(k-1)\psi(k-1)$ , з якого треба умова збіжності

$$\lim_{k \rightarrow \infty} |e(k)| = \sqrt{1 + \alpha} \delta, \quad \alpha \in (0,1), \quad (2.74)$$

тобто помилка ідентифікації  $e(k)$  ніколи не може бути по модулі менше заданих обмежень  $\delta$ .

По-третє, в алгоритмі Канудас де Віта-Каррильо

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \frac{\alpha P(k-1)\psi(k-1)}{\psi^T(k-1)P(k-1)\psi(k-1)} (|e(k)| - \delta) \text{sign } e(k); \quad (2.75)$$

$$P(k) = \gamma^{-1} (P(k-1) - \frac{\alpha(k)P(k-1)\psi(k-1)\psi^T(k-1)P(k-1)}{\psi^T(k-1)P(k-1)\psi(k-1)} (1 - \frac{\delta}{|e(k)|})), \quad (2.76)$$

$$\gamma \in (0, 1];$$

$$\alpha(k) = \begin{cases} 1, & |e(k)| > \delta \quad \text{або} \quad \psi^T(k-1)P(k-1)\psi(k-1) = 0; \\ 0, & |e(k)| \leq \delta \quad , \end{cases} \quad (2.77)$$

$$e(k) = y(k) - \hat{\Theta}^T(k-1)\psi(k-1) \quad (2.78)$$

у ситуації, коли  $|e(k)| > \delta$  й значення  $\psi^T(k-1)P(k-1)\psi(k-1)$  близько до нуля, виникає режим нестійкості, оскільки компоненти вектора

$$P(k-1)\psi(k-1)(\psi^T(k-1)P(k-1)\psi(k-1))^{-1} \quad (2.79)$$

можуть необмежено зростати. Крім того, у ситуації, коли  $\alpha(k) = 0$ , неможливо гарантувати виконання умови  $|e(k)| \leq \delta$  в припущенні, що  $\psi^T(k-1)P(k-1)\psi(k-1)$  обмежено.

Поєднуючи переваги розглянутих процедур, можна ввести комбінований алгоритм, що є своєрідною комбінацією рекурентного методу найменших квадратів і процедур Лозано-Ліла-Ортеги й Канудас де Віта-Каррильо [105].

Розглянемо алгоритм

$$\hat{\Theta}(k) = \hat{\Theta}(k-1) + \frac{\alpha(k)P(k-1)\psi(k-1)}{1 + \psi^T(k-1)P(k-1)\psi(k-1)} (|e(k)| - \delta) \text{sign } e(k) \quad , \quad (2.80)$$

$$P(k) = P(k-1) - \frac{\alpha(k)P(k-1)\psi(k-1)\psi^T(k-1)P(k-1)}{|\mathbf{e}(k)| + (2|\mathbf{e}(k)| - \delta)\psi^T(k)P(k-1)\psi(k-1)} (|\mathbf{e}(k)| - \delta), \quad (2.81)$$

$$\mathbf{e}(k) = \mathbf{y}(k) - \hat{\Theta}^T(k-1)\psi(k-1), \quad (2.82)$$

$$\alpha(k) = \begin{cases} 1, & \text{якщо } |\mathbf{e}(k)| > \delta, \\ 0, & \text{якщо } |\mathbf{e}(k)| \leq \delta \end{cases} \quad (2.83)$$

і проаналізуємо його збіжність.

Розглянемо вектор відхилень оцінок від ширих значень параметрів

$$\tilde{\Theta}(k) = \Theta - \hat{\Theta}(k) \quad (2.84)$$

і функцію Ляпунова (2.64).

Поєднуючи (2.58) з (2.80)-(2.82), отримуємо наступне вираження для функції Ляпунова:

$$\begin{aligned} V(k) = V(k=1) + & \frac{\alpha(k)(|\mathbf{e}(k)| - \delta)}{(1 + \psi^T(k-1)P(k-1)\psi(k-1))|\mathbf{e}(k)|} \times \\ & \times \left( \Delta w^2(k) - \frac{(1 + \psi^T(k-1)P(k-1)\psi(k-1))|\mathbf{e}(k)|^3}{|\mathbf{e}(k) + (2|\mathbf{e}(k)| - \delta)\psi^T(k-1)P(k-1)\psi(k-1)} \right). \end{aligned} \quad (2.85)$$

З урахуванням раніше введеної умови  $w \in D(0, \delta)$  нескладно переписати (2.85) у вигляді нерівності

$$\begin{aligned}
V(k) &\leq V(k-1) + \frac{\alpha(k)(|e(k)|-\delta)}{(1+\psi^T(k-1)P(k-1)\psi(k-1))|e(k)|} \times \\
&\times \left( \delta^2 - \frac{(1+\psi^T(k-1)P(k-1)\psi(k-1))|e(k)|^3}{|e(k)|+(2|e(k)|-\delta)\psi^T(k-1)P(k-1)\psi(k-1)} \right) = \\
&= V(k-1) - \frac{\alpha(k)(|e(k)|-\delta)}{(1+\psi^T(k-1)P(k-1)\psi(k-1))|e(k)|} \times \\
&\times \frac{(|e(k)|^3 - 2|e(k)|\delta^2 + \delta^3)(1+\psi^T(k-1)P(k-1)\psi(k-1)) + (|e(k)|-\delta)\delta^2}{|e(k)|+(2|e(k)|-\delta)\psi^T(k-1)P(k-1)\psi(k-1)} \leq \\
&\leq V(k-1) - \frac{\alpha(k)(|e(k)|-\delta)(|e(k)|^3 - 2|e(k)|\delta^2 + \delta^3)}{(|e(k)|+2|e(k)|-\delta)\psi^T(k-1)P(k-1)\psi(k-1)|e(k)|}
\end{aligned} \tag{2.86}$$

справедливої у випадку  $|e(k)| \geq \delta$ . Крім того, оскільки в цьому випадку

$$e^2(k) - 2\delta^2 + \frac{\delta^3}{|e(k)|} \geq |e(k)|(|e(k)| - \delta), \tag{2.87}$$

очевидно, що

$$V(k) \leq V(k-1) - \frac{\alpha(k)(|e(k)|(|e(k)| - \delta)^2)}{2(1+\psi^T(k-1)P(k-1)\psi(k-1))}, \tag{2.88}$$

звідки

$$\lim_{k \rightarrow \infty} \frac{\alpha(k)(|e(k)| - \delta)^2}{1 + \psi^T(k-1)P(k-1)\psi(k-1)} = 0, \tag{2.89}$$

що свідчить про критеріальну збіжності алгоритму (2.80)-(2.82).

З (2.81) виникає, що

$$\begin{aligned}
\left\| \hat{\Theta}(k) - \hat{\Theta}(k-1) \right\|_{V(k)}^2 &= \frac{\alpha(k)\psi^T(k-1)P^2(k-1)\psi(k-1)}{(1+\psi^T(k-1)P(k-1)\psi(k-1))^2} (|e(k)| - \delta^2) \leq \\
&\leq \frac{\alpha(k)\gamma_{\max}(P(k-1))}{1 + \psi^T(k-1)P(k-1)\psi(k-1)} (|e(k)| - \delta^2),
\end{aligned} \tag{2.90}$$



де  $\gamma_{\max}(P(k-1))$  - максимальне власне значення матриці  $P(k-1)$ .

З (2.81) виникає умова

$$\gamma_{\max}(P(k)) \leq \gamma_{\max}(P(k-1)) \leq \dots \leq \gamma_{\max}(P(0)) , \quad (2.91)$$

що дозволяє переписати (2.90) у вигляді

$$\|\hat{\Theta}(k) - \hat{\Theta}(k-1)\|_{V(k)}^2 \leq \alpha(k) \gamma_{\max}(P(0)) \frac{(|e(k)| - \delta)^2}{1 + \psi^T(k-1)P(k-1)\psi(k-1)} , \quad (2.92)$$

який разом з вираженням (2.89) свідчить про аргументну збіжність алгоритму.

Далі, використовуючи лему про обіг матриць, запишемо

$$P^{-1}(k) = P^{-1}(k-1) + \frac{\alpha(k) y(k-1)^T (k-1)}{1 + \psi^T(k-1)P(k-1)\psi(k-1)} \left(1 - \frac{\delta}{|e(k)|}\right) , \quad (2.93)$$

звідки виникає нерівність

$$\gamma_{\min}(P^{-1}(k)) \geq \gamma_{\min}(P^{-1}(k-1)) \geq \dots \geq \gamma_{\min}(P^{-1}(0)) , \quad (2.94)$$

де  $\gamma_{\min}(P^{-1}(k))$  - мінімальне власне значення матриці  $P^{-1}(k)$ .

Це нерівність разом з (2.88) приводить до того, що

$$V(k) \leq V(0) \quad (2.95)$$

та

$$\gamma_{\min}(P^{-1}(0)) \left\| \tilde{\Theta}(k) \right\|_{V(k)}^2 \leq \gamma_{\max}(P^{-1}(k)) \left\| \tilde{\Theta}(0) \right\|_{V(k)}^2 , \quad (2.96)$$

звідки можна записати вираження

$$\left\| \tilde{\Theta}(\mathbf{k}) \right\|_{V(\mathbf{k})}^2 \leq \frac{\gamma_{\max}(\mathbf{P}^{-1}(0))}{\gamma_{\min}(\mathbf{P}^{-1}(0))} \left\| \tilde{\Theta}(\mathbf{k}) \right\|_{V(\mathbf{k})}^2, \quad (2.97)$$

визначальна швидкість збіжності введеного алгоритму.

### **3 ЕВОЛЮЦІЙНІ СИСТЕМИ ЗІ ЗМІННОЮ СТРУКТУРОЮ НА ОСНОВІ НЕЧІТКОЇ ЛОГІКИ І ШІС ДЛЯ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ ДИНАМІЧНИХ ОБ'ЄКТІВ ЗА УМОВ НЕВИЗНАЧЕНОСТІ**

В останні роки спостерігається підвищений науковий та практичний інтерес до методів інтелектуальної обробки інформації за умов невизначеності. До них відносяться: штучні нейронні мережі, нечітка логіка, еволюційні обчислення, штучні імунні системи (ШІС) тощо.

Нечіткі моделі дозволяють описувати явища та процеси природньою мовою за допомогою лінгвістичних змінних, а механізм нечіткого виведення є зрозумілим людині, що зумовило широке застосування нечітких моделей для розв'язання задач ідентифікації, розпізнавання, управління, прийняття рішень, прогнозування тощо. Нечіткі моделі є універсальними апроксиматорами широкого класу багатомірних нелінійних функцій, що дозволяє за наявності навчальної вибірки апроксимувати функції або обмірювані дані з необхідною точністю.

Різним аспектам вирішення проблеми створення нечітких моделей присвячені роботи Заде Л., Ротштейна О.П., Круглова В.В., Борисова В.В., Ярушкіної Н.Г., Зайченка Ю.П., Бідюка П.І., Рутковської Д., Коско Б., Мамдані Є.Г., Такаги Т., Сугено М., Штовби С.Д., Янга Дж.-С.Р. та інших вчених.

Нечіткі моделі мають також недоліки, обумовлені необхідністю залучення експертів до формування баз знань, що вносить елементи суб'єктивності. Тому сьогодні отримали розповсюдження адаптивні нечіткі моделі, в яких реалізовано можливість зміни структури та параметрів моделі на основі навчання за експериментальними даними з метою досягнення оптимального стану при зміні зовнішніх умов.

Задачу адаптації нечітких моделей вирішують за допомогою різних градієнтних методів оптимізації [106, 107, 108, 112, 113, 114], серед яких

найбільш поширеними в літературі для вирішення даного завдання є:

- метод Левенберга-Марквардта [115,116] – метод оптимізації, спрямований на розв’язання задач про найменші квадрати. Традиційний варіант алгоритму Левенберга-Марквардта передбачає використання матриці Якобі для побудови квадратичної моделі, здійснення одного кроку відповідно до моделі, потім побудова нової моделі. У такого підходу є ряд недоліків – загальна неефективність алгоритму, високі вимоги до машинних ресурсів, погіршення збіжності до мінімуму на певному класі задач.

- градієнтний метод Коші [107], який є неефективним через низьку швидкість збіжності та погіршення збіжності до мінімуму, тому що приріст змінних безпосередньо залежить від градієнта, який наближається до нуля в околиці точки мінімуму. Крім того, відсутній механізм прискорення руху до точки мінімуму на останніх ітераціях.

- квазіньютонівські методи [117] - методи оптимізації, засновані на накопиченні інформації про кривизну цільової функції за спостереженнями за зміною градієнта, неефективні з обчислювальної точки зору при великій розмірності задачі оптимізації;

- фільтр Калмана [118, 119] - метод, в основі якого лежить метод градієнтного спуску.

Особливістю приведених методів адаптації нечітких моделей є відсутність можливості модифікації структури бази правил.

Таким чином, при виконанні адаптації нечітких моделей з використанням градієнтних методів виникають певні труднощі, спроби подолання яких призвели до використання еволюційних алгоритмів, до яких відносяться генетичні алгоритми, імунні обчислювальні системи та ін., які мають такі основні переваги [120]:

- відсутні значні математичні вимоги до видів цільових функцій і обмежень. При цьому можуть використовуватися найрізноманітніші цільові функції і види обмежень (лінійні та нелінійні), визначені на дискретних, безперервних та змішаних універсальних множинах;

- при використанні класичних покрокових методик глобальний оптимум може бути знайдений тільки в тому випадку, коли проблема має властивість опуклості. У той же час еволюційні алгоритми дозволяють ефективно відшукувати глобальний оптимум.

Генетичні алгоритми використовуються для оптимізації параметрів бази знань шляхом схрещування вихідних варіантів з наступною селекцією по деякому критерію [121]. В даний час ГА використовують для настройки функцій належності (параметрична адаптація), або для формування бази правил (структурна адаптація). Відповідно до об'єктів оптимізації обирають одиницю кодування - хромосому. Для налаштування ФН за хромосому обирають одне правило (Мічиганський підхід), для налаштування бази правил за хромосому обирають варіант бази правил (Пітсбургський підхід, підхід ітеративного навчання правил) [108]. Відповідно до кодування уточнюються правила генерації нових хромосом. Функція оптимальності являє собою механізм нечіткого виведення. Як видно з опису механізму нечіткого виведення, всі нечіткі правила вносять вклад в остаточний результат, тобто правила співпрацюють. Але при відборі правил (хромосом) ГА зберігає тільки правила, що зробили максимальний внесок у загальний результат, тобто хромосоми конкурують. Рішення проблеми конкуренції і кооперації в генетичних нечітких системах в кожному конкретному випадку будується евристично, наприклад, при підході ітеративного навчання правил використовують два етапи оптимізації. На першому кроці правила конкурують за право увійти в базу правил, а на другому - взаємодіють при формуванні загального результату [108].

ГА також властиві недоліки, пов'язані з потраплянням в локальні екстремуми.

ШС мають низку переваг перед ГА, пов'язаних з пам'яттю ШС, динамічним захистом, мережною взаємодією, самоорганізацією тощо [109]. В зв'язку з цим, створення методів і моделей адаптації нечітких моделей на

основі ШС є актуальним як з теоретичної, так і з практичної точки зору [110, 111].

### 3.1 Адаптація нечіткої моделі Такагі-Сугено з використанням штучних імунних систем

**3.1.1 Адаптивна нечітка модель Такагі-Сугено.** Представимо адаптивну нечітку модель у вигляді кортежу [122]:

$$AFM = \langle \{X_r, y_r\}, RB, DB, I, G(RB), L(DB), F \rangle, \quad (3.1)$$

де  $\{X_r, y_r\}, r = \overline{1, M}$  навчальна вибірка;

$RB$  – база правил;

$DB$  – база даних;

$I$  – механізм нечіткого логічного виведення;

$G(RB)$  – генерація і оптимізація бази правил;

$L(DB)$  – оптимізація бази даних (параметрів функцій належності (ФН));

$F$  – функція, що оцінює ефективність нечіткої моделі.

Структура адаптивної нечіткої моделі приведена на рис.1.1.

Розглянемо нечіткі моделі на основі алгоритмів нечіткого виведення Такагі-Сугено як одні з найбільш розповсюджених нечітких моделей.

Нечітка модель Такагі-Сугено задається алгоритмом нечіткого виведення Такагі-Сугено і базою нечітких правил виду:

$$\begin{aligned} & \text{П}_i : \text{ЯКЩО } x_i \in A_{i1} \text{ ТА } \dots \text{ ТА } x_j \in A_{ij} \text{ ТА } \dots \text{ ТА } x_m \in A_{im}, \\ & \text{ТО } y = k_{i1}x_1 + \dots + k_{ij}x_j + \dots + k_{im}x_m + k_{i0}, \quad i = 1, \dots, n, \end{aligned} \quad (3.2)$$

де  $A_{ij}$  – лінгвістичні терми, що представляють собою нечіткі множини, визначені на  $X_j$  з ФН  $\mu_{A_{ij}}(x_j) \in [0, 1]$ .

$k_{ij}$  ( $j = 1, \dots, m$ ) – коефіцієнти аргументів функції;

$k_{i0}$  – зміщення.

Використовуються гаусові ФН виду

$$\mu(x) = \exp\left[-\left(\frac{x-c}{\sigma}\right)^2\right], \quad (3.3)$$

де параметр  $c$  - центр нечіткої множини;

параметр  $\sigma$  – коефіцієнт розтягування функції.

Алгоритми навчання адаптивних нечітких моделей відносно трудомісткі і складні, і, як правило, складаються з двох стадій:

1. Структурна адаптація – генерація й оптимізація лінгвістичних правил. На даному етапі здійснюється формування і груба настройка моделі об'єкта шляхом побудови бази знань за доступною експериментальною інформацією [120].

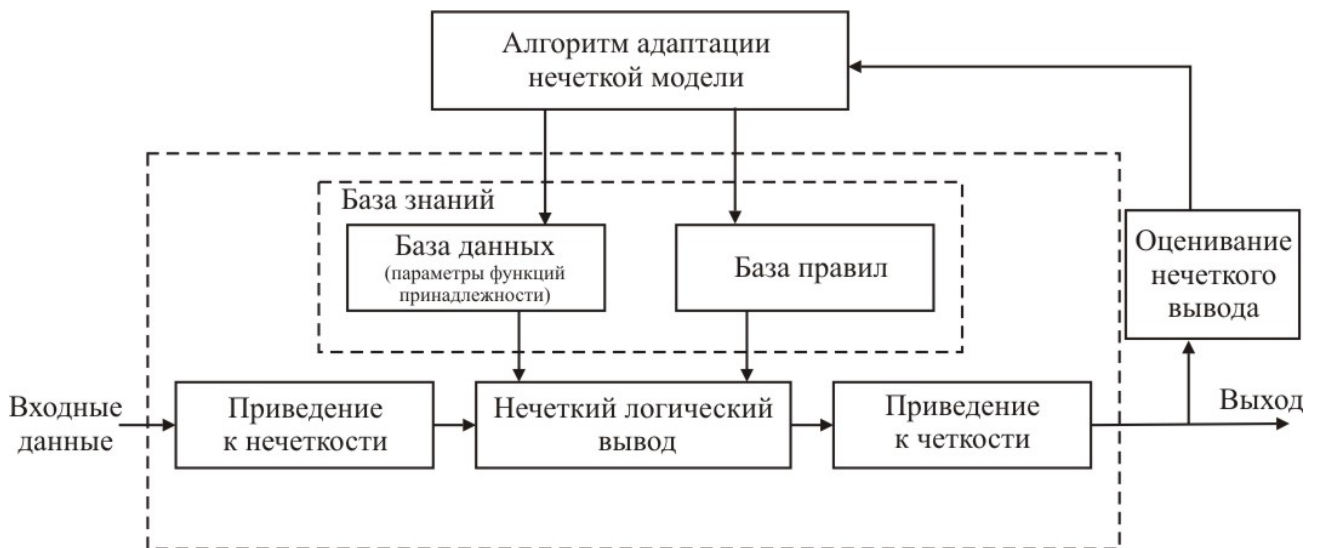


Рисунок 3.1 - Структура адаптивної нечіткої моделі

2. Параметрична адаптація - оптимізація параметрів ФН. Завдання параметричної адаптації нечітких моделей полягає в налаштуванні вектора параметрів ФН термів лінгвістичних змінних.

**3.1.2 Взаємодія етапів структурної і параметричної адаптації нечіткої моделі Такагі-Сугено.** Адаптація нечіткої моделі Такагі-Сугено зводиться до задачі оптимізації:

$$\sqrt{\frac{1}{M} \sum_{r=1}^M [F_{T-S}(X_r, P, B) - y_r]^2} \Rightarrow \min_{P, B}, \quad (3.4)$$

де  $B$  – вектор коефіцієнтів лінійних функцій в нечітких правилах бази правил;

$F_{T-S}(X_r, P, B)$  – результат нечіткого виведення по нечіткій базі знань з параметрами  $(P, B)$  при значенні входів  $X_r$ .

Вихідна лінгвістична змінна оцінюється лінійною функцією від входів (3.2), що значно ускладнює формування бази нечітких правил експертом.

Адаптація нечіткої моделі Такагі-Сугено представляє собою взаємодію етапів структурної і параметричної адаптації. На етапі структурної адаптації формується база правил нечіткого виведення. Структурна адаптація нечіткої моделі Такагі-Сугено полягає в синтезі нечітких правил виду (3.2) та настроюванні їх коефіцієнтів за наявними експериментальними даними. Структурна адаптація є ітераційною процедурою послідовної ідентифікації спостережень з навчальної вибірки за допомогою ШС.

База знань нечіткої моделі Такагі-Сугено може вважатися остаточно сформованою, якщо здійснено етап параметричної адаптації залишених у ній правил. Цей етап зводиться до знаходження, відповідно до наявних експериментальних даних та прийнятим критерієм, оптимальних значень параметрів ФН термів вхідних лінгвістичних змінних.

За результатами виконання структурної і параметричної адаптації формується база знань нечіткої моделі Такагі-Сугено.

Розробимо методи структурної та параметричної адаптації нечіткої моделі Такагі-Сугено першого порядку шляхом визначення імунних алгоритмів адаптації, а також способів кодування антитіл і антигенів.



**3.1.3 Структурна адаптація нечіткої моделі Такагі-Сугено.** Кодування антигенів та антитіл в задачі структурної адаптації нечіткої моделі Такагі-Сугено.

Представимо популяцію антигенів у вигляді множини:

$$Ag = \{Ag_1, Ag_2, \dots, Ag_M\}, \quad (3.5)$$

де  $M$  – розмір популяції антигенів, що відповідає кількості прикладів у навчальній вибірці. Кожний елемент множини  $Ag$  – приклад з навчальної вибірки, представлений у вигляді вектора фіксованої довжини:

$$Ag_i = \langle x_1, x_2, \dots, x_n, y \rangle, \quad (3.6)$$

де  $i = \overline{1, M}$ ;

$x_1, x_2, \dots, x_n$  – вхідні змінні об'єкта;

$y$  – вихідна змінна;

Представимо популяцію антитіл у вигляді множини  $Ab$ :

$$Ab = \{Ab_1, Ab_2, \dots, Ab_N\}, \quad (3.7)$$

де  $N$  – розмір популяції антитіл.

Кожний елемент множини  $Ab$  – антитіло  $Ab_i$ ,  $i = \overline{1, N}$  – сукупність параметрів, що настраюються в задачі, записана в вигляді вектору фіксованої довжини:

$$Ab_i = \langle ab_1, ab_2, \dots, ab_L \rangle, \quad (3.8)$$

де  $ab_j$  –  $j$ -ий параметр антитіла;

$L$  – довжина вектору  $Ab_i$ ;  $i = \overline{1, N}$ .

В популяції антитіл виду (3.7) кодуються правила нечіткого виведення. Пропонується наступний підхід до кодування правил нечіткого виведення в антитілі: кожне антитіло в популяції антитіл відповідає одному правилу з бази нечітких правил. У процесі роботи алгоритму антитіла з кращою афінністю формують базу правил. При цьому правила конкурують одне з одним за право увійти в базу правил. Структура бази правил при цьому не є фіксованою. Такий підхід дає можливість змінювати структуру бази правил для видалення надлишкових правил нечіткого виведення. При цьому розмір популяції антитіл виду (3.9) приймається рівним кількості прикладів у навчальній вибірці, тобто кожний приклад навчальної вибірки відповідає окремому правилу нечіткого виведення.

Правила в антитілі виду (3.10) кодуються числами, що представляють собою індекс нечітких множин вхідних змінних і коефіцієнти нечітких правил:

$$Ab = \langle IFS_1, IFS_2, \dots, IFS_n, k_0, k_1, \dots, k_n \rangle, \quad (3.9)$$

де  $IFS_1, IFS_2, \dots, IFS_n$  – індекси нечітких множин для  $n$  вхідних змінних;  
 $k_0, k_1, \dots, k_n$  – коефіцієнти нечітких правил виду (3.2).

Використання дійсного кодування дає переваги при вирішенні задач оптимізації безперервних параметрів, у той час як використання двійкового кодування параметрів антитіл веде до втрати інформації та підвищує обчислювальну складність алгоритму.

Відповідно до вибраного підходу до кодування антитіл база правил формується антитілами з кращою афінністю, для зберігання яких вводиться додаткова популяція – популяція клітин пам'яті виду:

$$MC = \{MC_1, MC_2, \dots, MC_R\}, \quad (3.10)$$

де  $MC_r$  –  $r$ -е антитіло, що містить параметри нечіткого правила,  $r = \overline{1, R}$ ;  
 $R$  – кількість правил нечіткого виведення.

**3.1.4 Імунний алгоритм структурної адаптації нечіткої моделі Такагі-Сугено.** На етапі структурної адаптації моделі нечіткого виведення будемо вирішувати задачу синтезу бази нечітких правил виду (3.2) з використанням прикладів навчальної вибірки і задачу настроювання коефіцієнтів правил нечіткого виведення. Спочатку кількість правил в базі правил приймається рівною кількості прикладів у навчальній вибірці. Оптимізація коефіцієнтів нечітких правил заснована на принципі клонального відбору. У процесі роботи алгоритму в кожному поколінні антитіла з афінністю, значення якої не перевищує заданий поріг афінності, заповнюють популяцію клітин пам'яті виду (3.20), формуючи базу нечітких правил.

Сформована таким чином база правил може бути надмірною. Оскільки структура бази правил не є фіксованою, надлишкові правила можуть бути видалені з бази шляхом виконання супресії.

В основі реалізації супресії в імунному алгоритмі структурної адаптації лежить теорія імунної мережі [123, 124]. Використання мережної взаємодії антитіл всередині популяції клітин пам'яті виду (3.20) дозволяє видаляти надлишкові правила шляхом обчислення афінності антитіл між собою і видалення антитіл, афінність яких перевищує коефіцієнт стиснення мережі антитіл.

Представимо імунний алгоритм структурної адаптації формально наступним чином:

$$\text{ImmAlg} = (P^L, L, Ab, N, Ag, M, MC, Op, n_c, N_c, d, \mathcal{G}_{Aff}, \delta_{net}, Alph, A, Aff, gen, t), \quad (3.11)$$

де  $MC$  – популяція клітин пам'яті виду (3.10);

$Op$  – множина імунних операторів виду

$Op = \{Clone, Mutate, Edit, Suppress\}$ ;

$\mathcal{G}_{Aff}$  – граничне значення афінності для заповнення популяції клітин пам'яті;

$\delta_{net}$  – коефіцієнт стиснення мережі.

Афінність антитіла і антигену  $Aff_{Ab-Ag}$  обчислюється як абсолютна величина різниці значень виходу моделі нечіткого виведення та очікуваного результату:

$$Aff_{Ab-Ag} = |F_{T-S}(X_r, P, B) - y_r|. \quad (3.12)$$

Афінність двох антитіл  $Ab_a$  і  $Ab_b$ , що використовується при виконанні супресії, обчислюється як евклідова відстань між векторами їх параметрів:

$$Aff_{Ab-Ab} = \|Ab_a - Ab_b\| = \sqrt{\sum_{j=1}^L (ab_{a,j} - ab_{b,j})^2}. \quad (3.13)$$

Виділимо основні етапи методу структурної адаптації нечіткої моделі Такагі-Сугено:

1. Генерація початкової популяції антитіл  $Ab$ .
2. Цикл для кожного антигену  $Ag_k$ ,  $k = \overline{1, M}$ 
  - 2.1 Обчислення афінності кожного антитіла  $Ab_i$ ,  $i = \overline{1, N}$  до антигену  $Ag_k$
  - 2.2. Вибір  $n_c$  антитіл з кращою афінністю.
  - 2.3. Клонування обраних в пункті 2.2 антитіл  $N_c$  разів і формування популяції клонів  $Ab_C$ .
  - 2.4. Формування популяції  $Ab_{CM}$  шляхом мутації клонів в популяції  $Ab_C$ .
  - 2.5. Обчислення афінності кожного антитіла  $Ab_j$ ,  $j = 1, \sum_{l=1}^{n_c} n_{c_l} N_{c_l}$  до

антигену  $Ag_k$  шляхом підстановки правила нечіткого виведення, закодованого в антитілі  $Ab_j$ , в базу правил нечіткої моделі Такагі-Сугено.

2.6. Редагування популяції антитіл  $Ab$ , в результаті якого виконується заміна антитіла  $Ab_i$ ,  $i = \overline{1, N}$  в популяції  $Ab$  на відповідне антитіло з популяції  $Ab_{CM}$ , якщо його афінність покращилася в результаті мутації і заміна  $d$  гірших антитіл в популяції  $Ab$  новими випадково згенерованими антитілами.

2.7 Заповнення популяції клітин пам'яті  $MC$  антитілами з популяції  $Ab$ , афінність яких не перевищує заданий поріг афінності  $\mathcal{G}_{Aff}$ .

2.8. Супресія всередині популяції  $MC$ .

3. Перевірка критерію зупинення. Якщо не досягнуто – перехід до кроку 2, в протилежному випадку – припинення алгоритму.

В загальному вигляді, зміни в популяції антитіл в поточному поколінні  $gen$  в результаті дії всіх імунних операторів можна представити наступним чином:

$$Ab^{gen+1} = Edit(Mutate(Clone(Ab^{gen})), Ab^{gen}); \quad (3.14)$$

$$Clone: Ab^{gen} \rightarrow Ab_c^{gen}; \quad (3.15)$$

$$Mutate: Ab_c^{gen} \rightarrow Ab_{CM}^{gen}; \quad (3.16)$$

$$Edit: (Ab_{CM}^{gen}, Ab^{gen}) \rightarrow Ab^{gen+1}. \quad (3.17)$$

Формування популяції клітин пам'яті можна представити наступним чином:

$$MC^{gen+1} = Suppress(MC^{gen} \cup Ab_s^{gen+1}), \quad (3.18)$$

де  $Ab_s^{gen+1} = \left\{ Ab_i^{gen+1} \mid Aff_{Ab-Ag} < \mathcal{A}_{Aff} \right\}$  – множина антитіл з популяції  $Ab$ , афінність яких не перевищує заданий поріг афінності  $\mathcal{A}_{Aff}$ ;

*Suppress* – оператор супресії, що видаляє з популяції клітин пам'яті антитіла, афінність антитіло-антитіло яких менше порогу стиснення мережі.

В результаті роботи алгоритму формується популяція клітин пам'яті  $MC$ , яка є набором правил нечіткого виведення з обчисленими коефіцієнтами виду:

$$\begin{aligned}
 R_1: & \text{ IF } x_1 \text{ is } IFS_{11} \text{ ...AND... } x_n \text{ is } IFS_{1n} \\
 & \text{ THEN } y = k_{10} + k_{11}x_1 + \dots + k_{1n}x_n; \\
 R_2: & \text{ IF } x_1 \text{ is } IFS_{21} \text{ ...AND... } x_n \text{ is } IFS_{2n} \\
 & \text{ THEN } y = k_{20} + k_{21}x_1 + \dots + k_{2n}x_n; \\
 & \dots \\
 R_q: & \text{ IF } x_1 \text{ is } IFS_{q1} \text{ ...AND... } x_n \text{ is } IFS_{qn} \\
 & \text{ THEN } y = k_{q0} + k_{q1}x_1 + \dots + k_{qn}x_n.
 \end{aligned} \tag{3.19}$$

**3.1.5 Імунний алгоритм параметричної адаптації нечіткої моделі Такагі-Сугено.** У методі параметричної адаптації нечіткої моделі Такагі-Сугено в популяції антигенів виду (3.7) кодуються приклади навчальної вибірки. Розмір популяції антигенів відповідає кількості прикладів у навчальній вибірці.

У популяції антитіл виду (3.9) в кожному антитілі виду (3.10) кодуються параметри термів ФН виду (3.2) для вхідних змінних.

У антитілі будемо кодувати центри нечітких множин і коефіцієнти розтягіння (параметри  $c_{ij}, \sigma_{ij}$  відповідно). У цьому випадку, антитіло виду (3.10) представимо наступним чином:

$$Ab = \langle c_{11}, \dots, c_{1m}, \dots, c_{n1}, \dots, c_{nm}, \sigma_{11}, \dots, \sigma_{1m}, \dots, \sigma_{n1}, \dots, \sigma_{nm} \rangle \tag{3.20}$$

де  $c_{ij}, \sigma_{ij}$ ,  $i = \overline{1, n}$ ;  $j = \overline{1, m}$  – параметри гаусових ФН для  $n$  вхідних

змінних, кожна з яких має  $m$  термів.

При такому кодуванні в одному антитілі закодовані всі параметри розв'язуваної задачі.

У разі великої кількості параметрів антитіла, використання двійкового кодування параметрів вектора є нераціональним через підвищення обчислювальних витрат на операції кодування-декодування. Пропонується використовувати дійсне кодування антитіл, при якому кожен параметр вектора антитіла описується окремим дійсним числом. Такий підхід дозволяє підвищити точність знайдених рішень і прискорити роботу алгоритму. Вихідний вектор параметрів ФП генерується випадковим чином.

Розмір популяції антитіл в імунному алгоритмі параметричної адаптації обирається невеликим, тому що використання великої кількості антитіл в популяції веде до великих обчислювальних витрат.

Імунний алгоритм параметричної адаптації заснований на принципі клонального відбору. Афінність антитіл обчислюється відповідно до (3.18).

У процесі роботи імунного алгоритму на вхід нечіткої моделі послідовно подається множина значень вхідних змінних з навчальної вибірки. Налаштування параметрів ФН в кожному антитілі виконується за результатами нечіткого виведення після пред'явлення нечіткої моделі всіх прикладів з навчальної вибірки.

Представимо формально імунний алгоритм параметричної адаптації наступним чином:

$$\text{ImmAlg} = (P^L, L, Ab, N, Ag, M, Op, n_c, N_c, d, Alph, A, Aff, gen, t), \quad (3.21)$$

де  $P^L$  – простір пошуку;

$L$  – розмірність простору пошуку;

$Ab$  – популяція антитіл виду (3.7);

$Ab_i$  –  $i$ -е антитіло популяції  $Ab$  виду (3.8);

$N$  – розмір популяції антитіл;

$Ag$  – популяція антигенів виду (3.5);

$M$  – розмір популяції антигенів;

$Op$  – множина імунних операторів виду  $Op = \{Clone, Mutate, Edit\}$ ;

$n_c$  – кількість антитіл для клонування;

$N_c$  – кількість клонів одного антитіла;

$d$  – кількість антитіл з гіршою афінністю, що замінюються при редагуванні популяції антитіл;

$Alph$  – алфавіт, з використанням якого кодуються антитіла;

$A$  – потужність алфавіту  $Alph$ ;

$Aff$  – функція афінності;

$gen$  – покоління роботи імунного алгоритму,

$t$  – критерій припинення роботи алгоритму.

В імунному алгоритмі параметричної адаптації будемо використовувати режим пакетного навчання, при якому значення параметрів нечеткої моделі, що настроюються, змінюються тільки після пред'явлення їй всіх прикладів з навчальної вибірки.

Імунний алгоритм представляє собою наступну послідовність кроків:

1. Генерація початкової популяції антитіл  $Ab$ .
2. Обчислення афінності кожного з антитіл  $Ab_i$ ,  $i = \overline{1, N}$  популяції  $Ab$  до всіх антигенів популяції антигенів  $Ag$  шляхом підстановки параметрів, закодованих в антитілі  $Ab_i$  в базу знань нечіткої моделі Такагі-Сугено та виконання нечіткого виведення.
3. Вибір  $n_c$  антитіл з кращою афінністю.
4. Клонування  $n_c$  обраних в п.3 антитіл  $N_c$  разів, в результаті якого формується популяція клонів  $Ab_C$ .
5. Формування популяції  $Ab_{CM}$  шляхом мутації клонів в популяції  $Ab_C$ ,
6. Обчислення афінності кожного з антитіл  $Ab_{CMj}$ ,  $j = \overline{1, \sum_{l=1}^{n_c} n_{c_l} N_{c_l}}$



популяції  $Ab_{CM}$  до всіх антигенів популяції антигенів  $Ag$  шляхом підстановки параметрів, закодованих в антитілі  $Ab_{CMj}$  в базу знань нечіткої моделі Такагі-Сугено і виконання нечіткого виведення.

7. Редагування популяції антитіл  $Ab$ , в результаті якого виконується заміна антитіла  $Ab_i$ ,  $i = \overline{1, N}$  в популяції антитіл  $Ab$  на відповідне антитіло з популяції  $Ab_{CM}$ , якщо його афінність покращилась в результаті мутації, і заміна  $d$  гірших антитіл в популяції антитіл  $Ab$  новими випадково сгенерованими антитілами.

8. Перевірка критерію зупинення. Якщо не досягнуто – перехід до кроку 2, в іншому випадку – припинення алгоритму.

В загальному виді, зміни в популяції антитіл в поточному поколінні  $gen$  в результаті дії всіх імунних операторів можна представити наступним чином:

$$Ab^{gen+1} = Edit(Mutate(Clone(Ab^{gen})), Ab^{gen}); \quad (3.22)$$

$$Clone: Ab^{gen} \rightarrow Ab_C^{gen}; \quad (3.23)$$

$$Mutate: Ab_C^{gen} \rightarrow Ab_{CM}^{gen}; \quad (3.24)$$

$$Edit: (Ab_{CM}^{gen}, Ab^{gen}) \rightarrow Ab^{gen+1}. \quad (3.25)$$

Результатом роботи імунного алгоритму параметричної адаптації є антитіло з кращою афінністю популяції, що містить параметри ФН термів вхідних змінних.

Таким чином розроблено методи структурної і параметричної адаптації нечіткої моделі Такагі-Сугено першого порядку на основу ШС, що дозволяють виконувати генерацію бази нечітких правил за експериментальними даними,

настроювати коефіцієнти правил нечіткого виведення, а також модифікувати структуру бази нечітких правил, що дозволяє усунути надмірність та суперечливість нечітких правил. Параметрична та структурна адаптація виконуються як два відокремлені у часі процеси.

Нечітка модель Такагі-Сугено також може бути представлена у вигляді адаптивної нечіткої нейронної мережі (АННМ), що дає додаткові можливості з настроювання параметрів бази знань нечіткої моделі.

### **3.2 Адаптація нечіткої нейронної мережі ANFIS з використанням штучних імунних систем**

**3.2.1 Адаптивні нечіткі нейронні мережі.** АННМ є багат шаровою мережею прямого поширення, в якій кожен шар виконує певну функцію (введення нечіткості, нечітке виведення, дефазифікація) [106, 125, 126].

Основними типами нечітких нейронних мереж є мережі ANFIS, Ванга-Менделя, Такагі-Сугено-Канга з параметричною оптимізацією правил на основі алгоритмів навчання [106].

Найбільше поширення в даний час отримали архітектури АННС виду ANFIS (Adaptive Network-based Fuzzy Inference System), що представляють собою окремий тип нечітких нейронних мереж, що навчаються на основі класичних градієнтних алгоритмів [126]. Нечіткі мережі типу ANFIS реалізують алгоритми нечіткого виведення Цукамото і Такагі-Сугено. Доведено, що такі мережі є універсальними апроксиматорами [126].

Розглянемо АННМ з архітектурою виду ANFIS, що реалізує алгоритм нечіткого виведення Такагі-Сугено для  $n$  вхідних змінних, що зображуються  $m$  нечіткими множинами з гаусовими ФН виду (3.3); однією вихідною змінною і  $m^n$  нейронами в прихованих шарах 2-4 [126], де  $m^n$  – максимальна кількість правил нечіткого виведення.

Узагальнена структура АННМ для алгоритму нечіткого виведення Такагі-Сугено представлена на рис. 3.2.

Перший шар виконує функцію введення нечіткості. Виходи нейронів цього шару ( $\mu_{A_{ij}}(x_i)$ ,  $i = \overline{1, n}$ ,  $j = \overline{1, m}$ , де  $A_{ij}$  – лінгвістичні терми, що представляють нечіткі множини;  $i$  – кількість вхідних змінних,  $j$  – кількість термів для лінгвістичної оцінки вхідних змінних) представляють собою значення гаусових ФН виду (3.3) для вхідних змінних  $x_i$ .

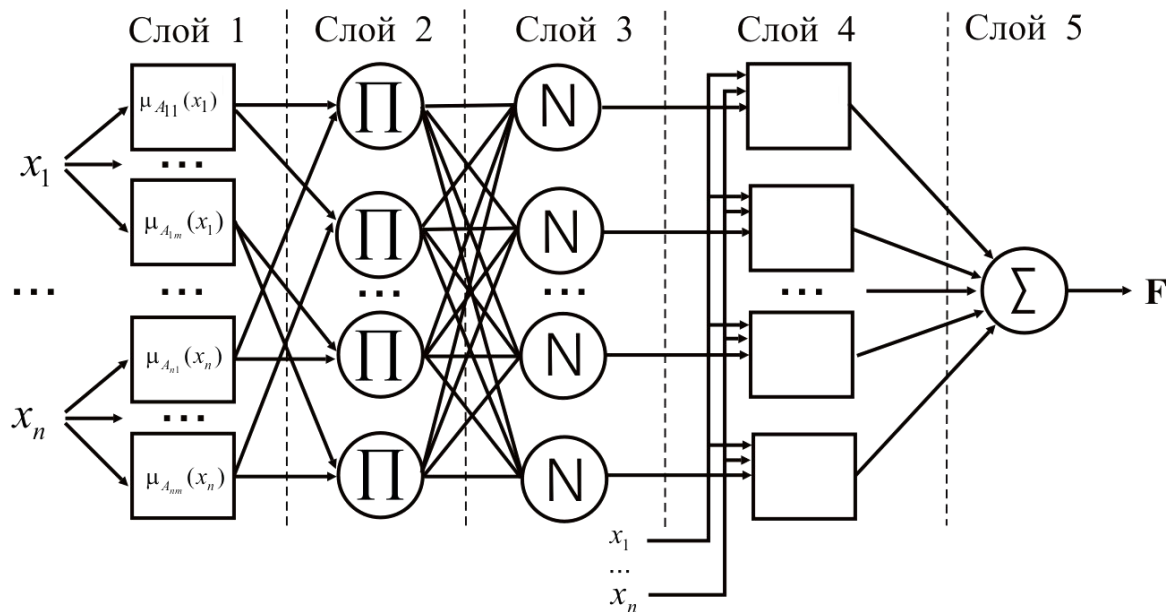


Рисунок 3.2 - Структура нечіткої нейронної мережі

Нейрони другого шару виконують множення вхідних сигналів і використовуються для формування ваги правила:

$$w_i = \prod_{j=1}^m \mu_{A_{ij}}(x_i), \quad i = \overline{1, m^n}, \quad (3.26)$$

де  $m^n$  – кількість нечітких правил.

Кожний  $i$ -й нейрон третього шару використовується для обчислення відношення ваги  $i$ -го правила до суми ваги всіх правил:

$$\bar{w}_i = \frac{w_i}{\sum_{i=1}^{m^n} w_i}, \quad i = \overline{1, m^n}. \quad (3.27)$$

Четвертий шар реалізує функцію активації нейронів, кожен нейрон в ньому описується виразом:

$$\overline{w_i f_i} = k_{i0} + k_{i1}x_1 + k_{i2}x_2 + \dots + k_{in}x_n, \quad (3.28)$$

де  $k_{ij}$ ,  $i = \overline{1, m^n}$ ,  $j = \overline{0, n}$  – параметри  $i$ -го правила Такагі-Сугено першого порядку.

П'ятий шар містить єдиний нейрон, який обчислює загальний вихід мережі як суму вхідних сигналів попереднього шару:

$$F = \sum \overline{w_i f_i}, \quad i = \overline{1, m^n}. \quad (3.29)$$

Перший і четвертий шари є адаптивними шарами, налаштування параметрів яких здійснюється в процесі навчання мережі. У першому шарі настраюються параметри ФН (центри нечітких множин  $c$  і коефіцієнти розтягування  $\sigma$  для гаусової ФН), а в другому - параметри нечітких правил  $k_{ij}$ ,  $i = \overline{1, m^n}$ ,  $j = \overline{0, n}$ .

Частина нейронів в АННМ є адаптивними, і вихід кожного з них залежить від параметрів, що відносяться до цього нейрону. Правило навчання визначає, як змінюються ці параметри для мінімізації помилки нечіткого виведення.

В існуючих системах з нечіткими нейронними мережами одним з найбільш важливих питань є розробка оптимального методу настроювання нечіткої бази правил, виходячи з навчальної вибірки, для отримання конструктивних і оптимальних моделей нечітких систем з подальшим використанням у практичних системах.

Задачу адаптації АННМ можна представити у вигляді наступної задачі оптимізації:

$$\sqrt{\frac{1}{M} \sum_{r=1}^M [F_{afns}(X_r, P) - y_r]^2} \Rightarrow \min_P, \quad (3.30)$$

де  $P$  – вектор параметрів АННМ;

$F_{afns}(X_r, P)$  – вихідне значення АННМ з параметрами  $P$  при значенні входів  $X_r$ .

Так як нечітка нейронна мережа ANFIS представляється у вигляді багатошарової структури з прямим розповсюдженням сигналу, а значення вихідної змінної можна змінювати, коригуючи параметри елементів шарів, то для навчання мережі можуть бути використані методи, використовувані для навчання класичних нейронних мереж [130, 118]:

1. Стохастичні методи навчання, які базуються на псевдовипадкових змінах ваги із збереженням тих змін, які ведуть до поліпшень. Стохастичні алгоритми вимагають дуже великого числа кроків навчання.

2. Градієнтні методи оптимізації першого і другого порядку[118]:

- градієнтні методи мінімізації;

- метод найменших квадратів (МНК);

- алгоритм Уїдроу-Хоффа;

- рекурентний алгоритм МНК;

- фільтр Калмана;

- метод суміжних градієнтів;

- метод Ньютона [117];

- алгоритм Левенберга-Марквардт [115, 116];

- метод зворотного поширення помилки [106, 125, 118, 126].

Градієнтні методи оптимізації першого порядку використовують обчислення градієнту функціоналу, при цьому функції, що використовуються, повинні бути диференційованими. Деякі з даних методів є неефективними

через низьку швидкість збіжності та погіршення збіжності до мінімуму, а також є чутливими до точності обчислень, особливо при вирішенні задач оптимізації великої розмірності.

Гradientні методи оптимізації другого порядку мають більшу швидкість збіжності, ніж методи першого порядку, проте припускають обчислення других похідних, що може бути пов'язано з істотними труднощами. Крім цього, метод Ньютона є чутливим до вибору стартової точки, тобто може розходитися, якщо цільова функція не є опуклою, а початкове наближення знаходиться далеко від точки мінімуму. Застосування алгоритму Левенберга-Марквардта, пов'язано з високими вимогами до машинних ресурсів, погіршенням збіжності до мінімуму на певному класі задач. Недоліком методу зворотного поширення помилки є недостатні точність і швидкість збіжності.

Одним з найбільш ефективних методів навчання нечітких нейронних мереж є гібридний gradientний метод [126], що є комбінацією методу зворотного поширення помилки і МНК. Його головна риса полягає в поділі процесу навчання на два відокремлених у часі етапи. На кожному етапі уточнюється тільки частина параметрів мережі. Параметри, що підлягають адаптації, поділяються на дві групи: перша складається з коефіцієнтів нечітких правил, друга складається з параметрів ФН. Уточнення параметрів проводиться в два етапи. На першому етапі при фіксації певних значень параметрів ФН шляхом вирішення системи лінійних рівнянь розраховуються коефіцієнти правил нечіткого виведення. На другому етапі (при фіксованих коефіцієнтах нечітких правил) розраховуються параметри ФН. Цей цикл повторюється до стабілізації всіх параметрів процесу.

Таким чином, існуючі методи навчання АННМ є достатньо трудомісткими, а також висувають значні математичні вимоги до видів цільових функцій і обмежень. Основною їх рисою є відсутність можливості модифікації бази правил. Для усунення недоліків існуючих методів навчання нечітких моделей пропонується використання ШС.

**3.2.2 Модель кодування параметрів АННМ.** Будемо розробляти метод адаптації АННМ [127, 128, 129] для налаштування параметрів ФН вхідних змінних і коефіцієнтів нечітких правил, а також для синтезу АННМ -видалення надлишкових правил і відповідної зміни структури мережі на основі ШС.

Сформуємо популяцію антигенів виду (3.7) із прикладів навчальної вибірки. Кожний антиген представимо вектором фіксованої довжини виду (3.8).

Будемо використовувати модель представлення всіх параметрів, що настроюються, у вигляді структурованого адаптивного мультиантитіла  $mAb_i$ . Популяцію мультиантитіл представимо наступним чином:

$$mAb = \{mAb_1, mAb_2, \dots, mAb_N\}, \quad (3.31)$$

де  $mAb_i = \{ab_0, ab_1, ab_2, \dots, ab_{L-1}\}$ ,  $i = \overline{1, N}$  –  $i$ -е адаптивне мультиантитіло, що представляє собою структурований вектор, довжина якого вимірюється в процесі роботи імунного алгоритму;

$L$  – довжина вектора  $mAb_i$ ;

$N$  – розмір популяції мультиантитіл.

Кожне мультиантитіло  $mAb_i$ ,  $i = \overline{1, N}$  популяції характеризується повною множиною параметрів нечіткої нейронної мережі, що настроюються.

У структурі мультиантитіла, наведеній на рис. 3.3, використовується поділ параметрів, що настроюються, на дві незалежні частини: частина з параметрами ФН виду (3.3) і частина з параметрами нечітких правил виду (3.2).

$c_{11}, \dots, c_{1m}, \dots, c_{n1}, \dots, c_{nm}; \sigma_{11}, \dots, \sigma_{1m}, \dots, \sigma_{n1}, \dots, \sigma_{nm}$	$k_{10}, \dots, k_{1n}$	...	$k_{q0}, \dots, k_{qn}$
$Ab_0$	$Ab_1$	...	$Ab_q$
Частина 1	Частина 2		
Мультиантитіло			

Рисунок 3.3 - Структура мультиантитіла

На рис.1.3  $c_{ij}, \sigma_{ij}$ ,  $i = \overline{1, n}$ ;  $j = \overline{1, m}$  – параметри гаусових ФН для  $n$  вхідних змінних, кожна з яких має  $m$  термів;  $k_{i0}, \dots, k_{in}$ ,  $i = \overline{1, q}$ ; – коефіцієнти  $q$  нечітких правил виду (3.2).

Частина 1 мультиантитіла представляє собою єдине антитіло  $Ab_0$ , довжина якого є фіксованою, так як число ФН для вхідних змінних є постійним.

Частина 2 мультиантитіла складається з  $q$  незалежних антитіл  $Ab_1, Ab_2, \dots, Ab_q$ , кожне з яких містить коефіцієнти  $k_{i0}, \dots, k_{in}$ ,  $i = \overline{1, q}$ ; для одного правила нечіткого виведення. Друга частина мультиантитіла є адаптивною, так як в процесі навчання крім оптимізації коефіцієнтів нечітких правил, що містяться в цій частині, змінюється їх загальна кількість, тобто виконується структурний синтез. Структурний синтез, в свою чергу, приводить до зміни кількості нейронів в прихованих шарах 2-4 нейронної мережі (рис. 3.2).

Структурований спосіб формування мультиантитіла дозволяє підвищити ефективність імунного алгоритму. За рахунок роздільного застосування імунних операторів до кожної з частин антитіла - імунні оператори застосовуються до параметрів ФН і коефіцієнтам нечітких правил окремо. Обчислення афінності виконується для мультиантитіла в цілому, оперуючи обома складовими.

Початкова популяція мультиантитіл генерується випадковим чином. Використовується дійсне кодування параметрів антитіла та антигену, що дозволяє підвищити точність рішень та зменшити обчислювальну складність. Розмір популяції мультиантитіл обирається невеликим, тому що використання великої кількості мультиантитіл в популяції веде до великих обчислювальних витрат.

**3.2.3 Імунний алгоритм адаптації нечіткої нейронної мережі.** Імунний алгоритм, який реалізує метод адаптації АННМ [127, 128, 129], заснований на принципі клонального відбору та теорії імунної мережі.



За принципом клонального відбору реалізується настроювання параметрів АННМ. Алгоритм клонального відбору забезпечує підтримку розмаїття мультиантитіл в популяції шляхом глобального перегляду простору параметрів, що настроюються, уникаючи локальних мінімумів.

Застосування положень теорії імунної мережі дозволяє оцінити взаємодію антитіл між собою і провести супресію, усуваючи таким чином надмірність правил нечіткого виведення.

Представимо імунний алгоритм адаптації АННМ формально наступним чином:

$$\text{ImmAlg} = \left( P^L, L, mAb, N, Ag, M, Op, n_c, N_c, d, \delta_{net}, Alph, A, Aff, gen, t \right), \quad (3.32)$$

де  $mAb$  – популяція мультиантитіл виду (3.7),  $mAb_i$ , –  $i$ -е мультиантитіло популяції  $mAb$ ;

$N$  – розмір популяції мультиантитіл;  $Op$  – множина імунних операторів виду  $Op = \{Clone, Mutate, Edit, Suppress\}$ ;

$n_c$  – кількість мультиантитіл для клонування;

$N_c$  – кількість клонів одного мультиантитіла;

$d$  – кількість мультиантитіл з гіршою афінністю, що замінюються при редагуванні популяції антитіл.

Афінність мультиантитіла  $mAb_i$  визначається як середньоквадратичне відхилення реального виходу АННМ від очікуваного для всіх антигенів популяції  $Ag$ :

$$Aff_{mAb-Ag} = \sqrt{\frac{1}{M} \sum_{r=1}^M [F_{afms}(X_r, P) - y_r]^2}. \quad (3.33)$$

Адаптація АННМ з використанням імунного алгоритму виконується в режимі пакетного навчання, при якому значення параметрів мережі змінюються

тільки після пред'явлення мережі всіх навчальних образів. Таким чином, афінність кожного мультиантитіла  $mAb_i$  обчислюється відносно всієї популяції антигенів  $Ag$ .

Основними етапами методу адаптації АННМ є наступні кроки:

1. Генерація початкової популяції мультиантитілів  $mAb$ .
2. Створення  $N$  АННМ  $AFNNet_i$  з параметрами, декодованими з відповідних мультиантитілів  $mAb_i$ ,  $i = \overline{1, N}$ . Обчислення афінності мультиантитілів  $mAb_i$  до всіх антигенів популяції антигенів  $Ag$ .
3. Вибір  $n_c$  мультиантитілів з кращою афінністю.
4. Клонування  $n_c$  обраних мультиантитілів  $N_c$  разів, формування популяції клонів  $mAb_C$ .
5. Формування популяції  $mAb_{CM}$  шляхом мутації клонів в популяції  $mAb_C$ .
6. Створення  $\sum_{j=1}^{n_c} n_{c_j} N_{c_j}$  АННМ  $AFNNet_j$  з параметрами, декодованими з відповідних мультиантитілів  $mAb_{CM_j}$ . Обчислення афінності мультиантитілів  $mAb_{CM_j}$  до всіх антигенів популяції антигенів  $Ag$ .
7. Редагування популяції мультиантитілів  $mAb$ : заміна мультиантитіла  $mAb_i$  в популяції  $mAb$  на відповідне мультиантитіло з популяції  $mAb_{CM}$ , якщо його афінність покращилась в результаті мутації. Заміна  $d$  гірших мультиантитілів в популяції  $mAb$  новими випадково сгенерованими мультиантитілами.
8. Супресія в популяції антитілів  $mAb$ .
9. Перевірка критерію зупинення. Якщо не досягнуто – перехід до шагу 2, в протилежному випадку – припинення алгоритму.

В загальному виді зміни в популяції антитілів в поточному поколінні  $gen$  в результаті дії всіх імунних операторів можна представити наступним чином:

$$mAb^{gen+1} = Suppress(Edit(Mutate(Clone(mAb^{gen})), mAb^{gen})). \quad (3.34)$$

Оператори клонування, мутації і редагування популяції застосовуються до обох частин мультиантитіла для всіх мультиантитіл популяції:

$$Clone: mAb^{gen} \rightarrow mAb_C^{gen}; \quad (3.35)$$

$$Mutate: mAb_C^{gen} \rightarrow mAb_{MC}^{gen}; \quad (3.36)$$

$$Edit: (mAb_{MC}^{gen}, mAb^{gen}) \rightarrow mAb^{gen+1}. \quad (3.37)$$

На етапі супресії виконується видалення надлишкових правил шляхом видалення антитіл зі схожою афінністю, що містяться в другій частині мультиантитіла для кожного мультиантитіла  $mAb_i^{gen+1}$ ,  $i = \overline{1, N}$ . Афінність двох антитіл  $Ab_1$  і  $Ab_2$  обчислюється за допомогою виразу:

$$Aff_{Ab-Ab} = \|Ab_1 - Ab_2\| = \sqrt{\sum_{j=0}^n (k_{1j} - k_{2j})^2}. \quad (3.38)$$

Антитіла, афінність між якими не перевищує поріг стиснення мережі, видаляються з мультиантитіла:

$$Suppress: mAb^{gen+1} \rightarrow mAb_{sup}^{gen+1}, \quad (3.39)$$

де  $mAb_{sup}^{gen+1}$  – популяція мультиантитіл після виконання супресії.

При зменшенні кількості нечітких правил (антитіл у частині 2 мультиантитіла) змінюється структура АННМ: зменшується кількість нейронів у прихованих шарах 2-4. Структура АННМ (кількість нейронів прихованих

шарів) встановлюється відповідно до кількості нечітких правил, закодованих в мультиантитілі.

Результатом роботи алгоритму є мультиантитіло з кращою по популяції афінністю, що містить параметри правил нечіткого виведення і ФН вхідних змінних.

Таким чином запропоновано метод адаптації АННМ, яка реалізує нечітке виведення за алгоритмом Такагі-Сугено, на основі ШС, що дозволяє виконувати одночасне настроювання всіх параметрів АННМ. Розроблено модель представлення параметрів АННМ, що настроюються, за допомогою адаптивного мультиантитіла, яке дозволяє застосовувати імунні оператори до кожної з частин мультиантитіла окремо, що забезпечує можливість виконувати модифікацію структури АННМ для видалення надмірності правил нечіткого виведення.

### **3.3 Експериментальні дослідження**

Програмну реалізацію алгоритмів адаптації нечітких моделей розроблено в програмному середовищі Microsoft Visual Studio 2008 на мові C++.

Для оцінки ефективності запропонованих імунних алгоритмів адаптації нечітких моделей проведемо експерименти на модельних прикладах.

Методика експерименту полягає у виконанні наступних кроків:

- 1) генерація навчальної вибірки для кожної тестової функції;
- 2) конструювання адаптивної нечіткої моделі (Такагі-Сугено, ННС);
- 3) адаптація нечіткої моделі з застосуванням відповідних імунних алгоритмів;
- 4) відображення динаміки адаптації нечіткої моделі за допомогою графіків афінності.

Експерименти будемо проводити з використанням тестових функцій (ТФ) двох змінних, приведених в табл. 3.1.

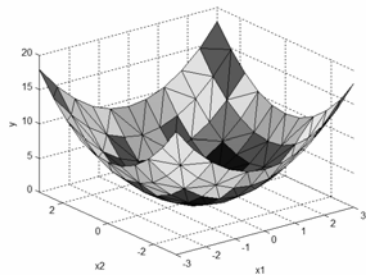
Таблиця 3.1 - Тестові функції

	Тестові функції	Область визначення функції
1	$F_1(x_1, x_2) = x_1^2 + x_2^2$	$x_1, x_2 \in [-3; 3]$
2	$F_2(x_1, x_2) = 3(1 - x_1)^2 \exp(-x_1^2 - (x_2 + 1)^2) - 10\left(\frac{x_1}{5} - x_1^3 - x_2^5\right) \exp(-x_1^2 - x_2^2) - \frac{1}{3} \exp(-(x_1 + 1)^2 - x_2^2)$	$x_1, x_2 \in [-3; 3]$

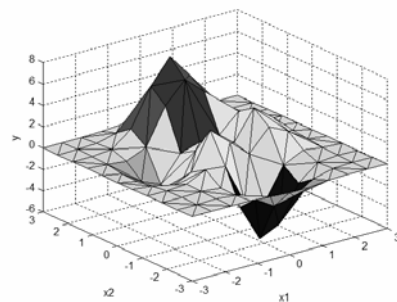
Поверхні ТФ  $F_1 - F_3$  приведені на рис. 3.4.

Для кожної тестової функції сформуємо навчальну вибірку виду  $(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}, y^{(k)})$ ,  $k = \overline{1, K}$ , де  $x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}, y^{(k)}$  – значення входних змінних  $x_1, x_2, \dots, x_n$  і вихідної змінної  $y$  в  $k$ -ом прикладі;  $K$  – загальна кількість прикладів в навчальній вибірці.

В [131, 107] приведені графіки кривих, отриманих при адаптації моделі Такагі-Сугено з різними за величиною навчальними вибірками. Відзначено, що якість результату підвищується, при збільшенні навчальної вибірки таким чином, що її обсяг в 2-3 рази більше, ніж кількість параметрів, що настроюються. Враховуючи це, сформуємо навчальну вибірку для тестових функцій  $F_1, F_2$  відповідно до діапазону входних змінних і кількістю параметрів, що настроюються.



а



б

Рисунок 3.4 - Поверхня тестових функцій: а – ТФ  $F_1$ ; б – ТФ  $F_2$ .

За даними з навчальної вибірки створюються нечіткі моделі ТФ за алгоритмом нечіткого виведення Такагі-Сугено, а також з використанням АННМ.

Входи нечіткої моделі Такагі-Сугено і АННМ розглядаються як лінгвістичні змінні, значення яких визначаються з терм-множини {«низький», «нижче середнього», «середній», «вище середнього», «високий»}.

Кожний терм представлений нечіткою множиною з гаусовими ФН виду (3.3).

Початкові значення параметрів ФН встановлюються таким чином, щоб ФН були рівномірно розподілені по області визначення функцій.

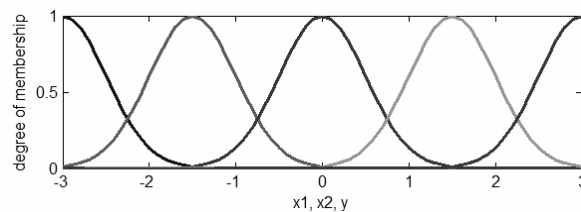


Рисунок 3.5 - Вихідні ФН вхідних змінних для тестових функцій  $F_1, F_2$

Параметри використовуваних імунних алгоритмів виду (2.9), (2.18), (2.26) наведено в табл. 3.2.

Таблиця 3.2 - Параметри імунних алгоритмів

Параметр	Структурна адаптація нечіткої моделі Такагі-Сугено	Параметрична адаптація нечіткої моделі, Такагі-Сугено, адаптація ННМ
Розмір популяції	відповідає розміру навчальної вибірки	15
Кількість антитіл для клонування	30	7
Мінімальна кількість клонів	10	10

Продовження таблиці 3.3

Максимальна кількість клонів	20	20
Максимальна ймовірність мутації	0.7	0.7
Кількість антитіл для редагування	1	1
Коефіцієнт супресії	0,001	-
Максимальна кількість поколінь	15000	15000
Поріг афінності	0.001	0.001

Критерієм зупинення в роботі імунних алгоритмів адаптації нечітких моделей є досягнення заданого порогу афінності, або, якщо він не досягнутий, заданої кількості поколінь роботи алгоритму.

Кількість параметрів, що настраюються в нечітких моделях – 95. Розмір навчальної вибірки – 289.

У табл. 1.3 наведено дані по середньоквадратичному відхиленню за результатами 10 запусків імунних алгоритмів адаптації нечітких моделей. Динаміка навчання, настроєні ФН, відновлені поверхні функцій наведено на рис. 3.6-3.9.

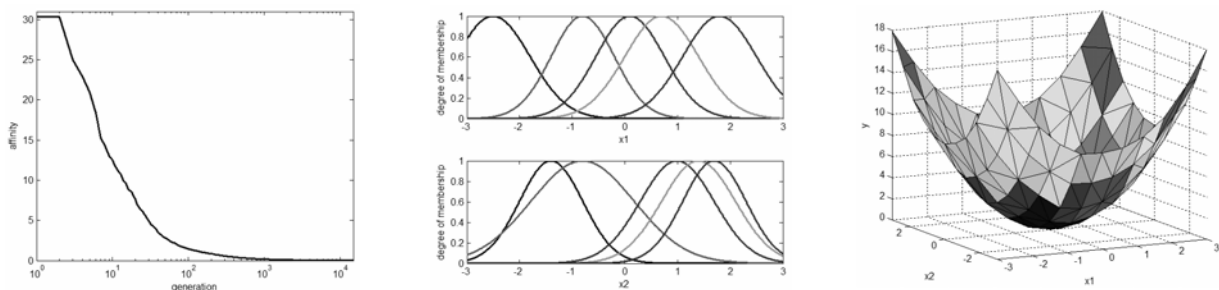
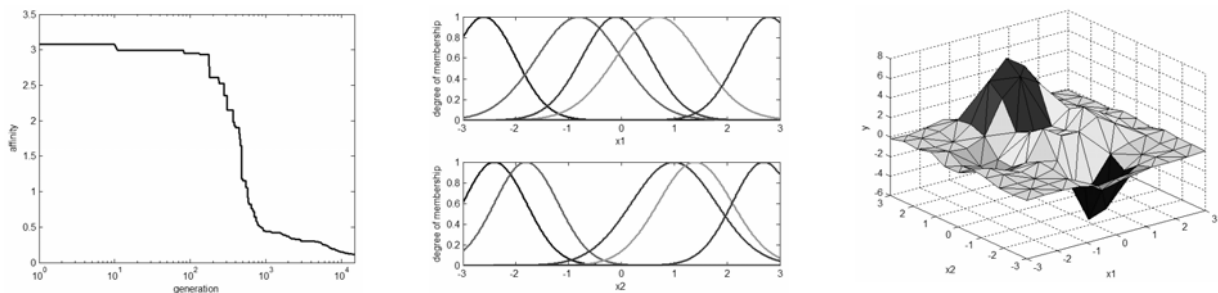


Рисунок 3.6 - Тестова функція  $F_1$ . Результати навчання нечіткої моделі Такагі-Сугено– динаміка навчання, настроєні ФН, відновлена поверхня функції

Таблиця 3.4 - Результати експериментів: нечітка модель Такагі-Сугено

Кількість лінгвістичних термів	Кількість параметрів, що настраюються (параметри ФН\ коефіцієнти нечітких правил)	Розмір навчальної вибірки	Середньоквадратичне відхилення			
			$F_1$	$F_2$	$F_3$	$F_4$
2	8\12	121	-	-	-	0.018
3	12\27	121	0.080	0.197	-	-
4	16\48	216	-	-	0.002	-
5	20\75	289	0.010	0.016	-	-
7	28\147	400	0.009	0.012	-	-

Рисунок 3.7 - Тестова функція  $F_2$ . Результати навчання нечіткої моделі Такагі-Сугено – динаміка навчання, настроєні ФН, відновлена поверхня функції

Таблиця 3.5 - Результати експериментів: АННМ

Кількість лінгвістичних термів	Кількість параметрів, що настраюються	Розмір навчальної вибірки	Середньоквадратичне відхилення			
			$F_1$	$F_2$	$F_3$	$F_4$
2	20	121	-	-	-	0.010
3	39	121	0.065	0.157	-	-
4	64	216	-	-	0.008	-
5	95	289	0.008	0.012	-	-
7	175	400	0.006	0.010	-	-



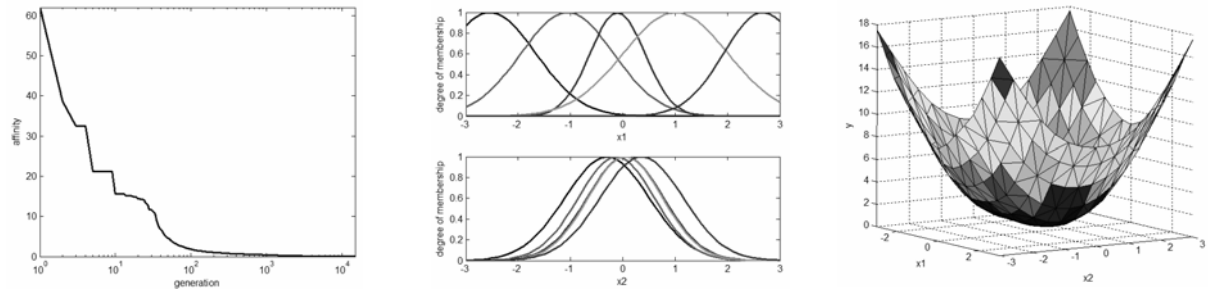


Рисунок 3.8 - Тестова функція  $F_1$ . Результати адаптації нечіткої нейронної мережі – динаміка навчання, настроєні ФН, відновлена поверхня функції.

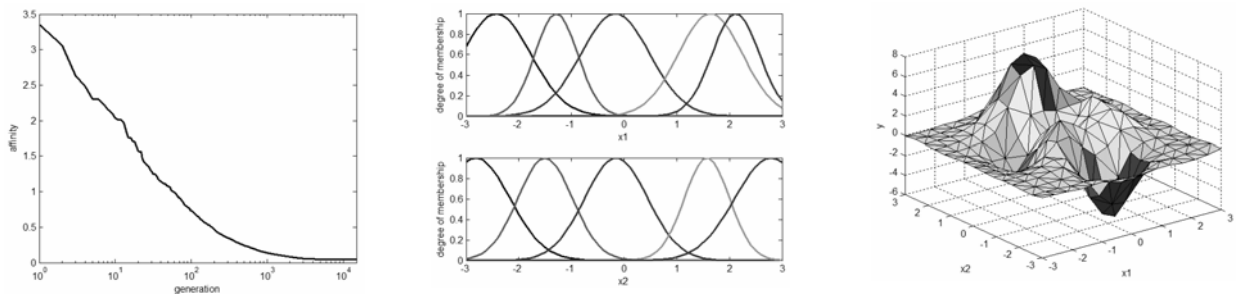


Рисунок 3.9 - Тестова функція  $F_2$ . Результати адаптації нечіткої нейронної мережі – динаміка навчання, настроєні ФН, відновлена поверхня функції

Перевагою запропонованого імунного підходу до навчання нечітких моделей ідентифікації нелінійних залежностей в порівнянні з відомими методами [126, 115, 116, 132, 133, 119] є можливість ефективного пошуку глобального оптимуму і відсутність значних математичних вимог до видів цільових функцій і обмежень. Немає необхідності спрощувати модель об'єкта, втрачаючи її адекватність і штучно добиваючись можливості застосування доступних математичних методів.

Порівняльний аналіз методів адаптації нечітких моделей Такагі-Сугено і АННМ (табл. 3.5) показує ефективність імунних алгоритмів адаптації нечітких моделей в порівнянні з іншими розглянутими методами.

Таблиця 3.6 - Адаптація нечіткої моделі Такагі-Сугено, АННМ

	Середньоквадратичне відхилення	
	$F_1$	$F_2$
Імунний алгоритм адаптації нечіткої моделі Такагі-Сугено	0.010	0.016
Імунний алгоритм адаптації АННМ	0.008	0.012
Метод зворотного розповсюдження помилки	0.012	0.088
Гібридний градієнтний метод	0.001	0.019
Метод Левенберга-Марквардта	0.017	0.120
Генетичний алгоритм	0.021	0.098

При моделюванні унімодальної нелінійної функції гібридний метод адаптації АННМ показує найбільшу точність, але при моделюванні мультимодальних функцій з кількома екстремумами є менш ефективним, ніж імунні алгоритми адаптації нечітких моделей. При цьому імунні алгоритми адаптації нечітких моделей Такагі-Сугено дозволяють скорочувати кількість нечітких правил для усунення надмірності та скорочення часу обчислень (табл. 3.6 - 3.7).

Питання збіжності запропонованих методів розглянуто в [ 134, 135].

Таким чином, проведено експериментальні дослідження на тестових функціях двох змінних, а також порівняльний аналіз запропонованого підходу з існуючими, які показують ефективність запропонованих імунних алгоритмів адаптації нечітких моделей [136, 137, 138].

Таблиця 3.7 - Кількість правил після супресії – нечітка модель Такагі-Сугено

Кількість лінгвістичних термів	Макс. кількість правил	$F_1$	$F_2$	$F_3$	$F_4$
2	8	-	-	-	7
3	9	8	9	-	-
4	16	-	-	14	-
5	25	19	23	-	-
7	49	40	45	-	-

Таблиця 3.8 - Кількість правил після супресії – АННМ

Кількість лінгвістичних термів	Макс. кількість правил	$F_1$	$F_2$	$F_3$	$F_4$
2	8	-	-	-	7
3	9	8	8	-	-
4	16	-	-	15	-
5	25	22	24	-	-
7	49	42	45	-	-

До теперішнього часу ще недостатньо повно досліджені важливі аспекти проблеми адаптації нечітких моделей, зокрема, питання генерації бази нечітких правил за експериментальними даними і модифікації бази знань в процесі адаптації для усунення надмірності, питання модифікації структури АННМ для видалення надмірності, застосування ШС в задачах адаптації нечітких моделей.

Тому в даній роботі вирішено актуальну задачу розробки методів адаптації нечітких моделей за експериментальними даними з використанням ШС з метою підвищення ефективності інтелектуальної обробки інформації в умовах невизначеності та зниження впливу суб'єктивності експертних оцінок.

Запропоновано методи структурної та параметричної адаптації нечітких моделей на основі ШС, що дозволяють генерувати набір нечітких правил за експериментальними даними, настроювати їх параметри, а також модифікувати структуру бази нечітких правил, що дає можливість усунути їх надмірність і суперечливість.

Запропоновано метод адаптації АННМ на основі ШС, в якому виконується одночасна настройка всіх параметрів мережі, а також модифікація її структури шляхом зменшення кількості нейронів у прихованих шарах мережі. Це дозволяє спростити АННМ і підвищити точність моделювання.

Проведені експериментальні дослідження розроблених методів адаптації нечітких моделей показали їх високу ефективність.

#### **4 МЕТОДИ ПРИЙНЯТТЯ РІШЕНЬ ПРО СТАН ОБ'ЄКТІВ ЗА УМОВ НЕВИЗНАЧЕНОСТІ, ЩО ҐРУНТУЮТЬСЯ НА ЕВОЛЮЦІЙНИХ ІНТЕРВАЛЬНИХ ФАЗЗИ МОДЕЛЯХ**

Важливим аспектом прийняття попередніх оперативних оцінок про стан просторово розподілених об'єктів є відсутність вихідних даних про стан об'єкта, на який впливає множина факторів, оцінювання та аналіз яких є складною проблемою внаслідок відсутності апріорних даних [139]. До них, в першу чергу, слід віднести багатоаспектність, різноплановість та багатокрітеріальність множини існуючих факторів, що функціонують в умовах невизначеності нечіткого простору станів.

Значний внесок в створення та розвиток м'яких обчислень і нечіткої логіки внесли L.A. Zadeh [140], R.R. Yager, L.H. Tsoukalas [141].

Нагальними є розробка методів, моделей та інформаційних технологій, які дозволяють на основі знання орієнтованих технологій оперативно реалізовувати процеси оцінювання станів складних просторово розподілених об'єктів [142]. Аналіз останніх публікацій в [139] показав, що розглянуті питання досліджено недостатньо, а наукові результати не у всіх випадках доведено до практичної реалізації та потребують додаткового вивчення, що підтверджує актуальність досліджень.

##### **4.1 Дослідження існуючих підходів і методів оцінювання станів просторово розподілених об'єктів на основі еволюційних алгоритмів**

У цьому підрозділі проаналізовано методи оцінювання просторово розподілених об'єктів. Виконано аналітичний огляд існуючих підходів і проблем моделювання процесів у складних просторово розподілених об'єктах.

На основі змістовного аналізу методів оцінювання складних просторово розподілених об'єктів, які характеризуються просторовим та функціональним розподілом, асинхронною взаємодією процесів, представлених на множині

відносин «умова-дія», визначено, що існує множина методик оцінювання станів таких об'єктів. До них у першу чергу слід віднести детерміновані, стохастичні, експертні, ієрархій, а також, основані на методах багатокритеріальної оптимізації, нечітких множин, нечітких мереж Петрі, нечіткої логіки.

Встановлено, що існуючі методи об'єктно-орієнтовані та їх ефективність у значній мірі залежить від предметної області. оцінювання станів таких об'єктів є важливою та актуальною проблемою [143, 144]. Проте, вимоги до підвищення достовірності в умовах жорстких обмежень на обчислювальні ресурси та специфіка предметної області визвали необхідність створення та дослідження таких моделей.

Розглянемо формальні аспекти. Нехай існує множина взаємодіючих динамічних просторово розподілених об'єктів

$$\{O_i\} \supseteq \{O_i^A\}, i \in I, \quad (4.1)$$

де  $O_i^A$  – кортеж актуальних об'єктів,

$I$  – множина індексів об'єктів.

При взаємодії об'єктів (4.1) виникає складна просторово розподілена асинхронна взаємодія динамічних процесів

$$\{Pr_{ij}\}, i \in I, j \in J, \quad (4.2)$$

де  $J$  – множина індексів процесів, які можуть носити детермінований характер  $D$ , стохастичний характер  $P$ , нечіткий характер  $\tilde{F}$ , при цьому часто є невизначеними та суперечними.

Процеси (4.2) відображають дані  $Dt$ , знання  $Kn$  про предметну область, їх складну асинхронну взаємодію представлено на множині відносин «умова (C)» – «дія (A)»:

$$R(CDt, ADt), \quad (4.3)$$

$$R(CKn, AKn). \quad (4.4)$$

Виходячи з особливостей предметної області дослідження, будемо знання орієнтовану модель  $\tilde{S}_\Sigma$  для множини (4.1) с урахуванням характеру процесів (4.2) взаємодії об'єктів множини (4.1).

Зважаючи на важливість для предметної області просторових характеристик, що змінюються у часі, рішення повинні бути орієнтовані на сучасні геоінформаційні технології [145], які дозволяють виконувати просторову та часову прив'язку об'єктів, вирішувати розподілені задачі.

З усієї множини процесів, що відбуваються у системі, засобами геоінформаційних технологій можна представити

$$\{Pr^{(GIS)}\} \subseteq \{Pr_j\}. \quad (4.5)$$

Згідно (4.2), процеси в об'єктах (4.1) можуть бути представлені детермінованими  $D$ , стохастичними  $P$  і нечіткими моделями  $\tilde{F}$ .

При оцінюванні об'єктів предметної області слід враховувати множину факторів  $\{\Phi_i\}$ ,  $i \in I$ , які функціонують в нечіткому просторі станів.

Підсумовуючи вищевикладене, можна зробити висновок, що розробка нових моделей, методів та інформаційних технологій оцінювання станів просторово розподілених об'єктів дозволить істотно підвищити ефективність та достовірність рішень, які приймаються.

Проведені дослідження були спрямовані на розробку підходів до побудови моделей прийняття рішень, визначення критеріїв адекватності процесів прийняття рішень та виділення домінуючих факторів, що визначають ступінь ранжирування процесів; розробку та обґрунтування гібридної моделі опису об'єктів (4.1) і процесів (4.2), а також визначення умов досяжності на множині компонент моделі; розробку і обґрунтування ефективного метода

моделювання (4.5) послідовно-паралельних процесів в просторово розподілених об'єктах; використовуючи існуючі підходи та нормативну базу, розробити і обґрунтувати інформаційну технологію оцінювання об'єктів (4.1) на множині процесів предметної області в задачах оцінювання земельних ресурсів; розробити методи і алгоритми рішення прикладних задач, розробити структуру інструментальних програмних засобів оцінювання станів території.

#### 4.2 Розробка фаззі моделей процесів оцінювання станів просторово розподілених об'єктів

У рамках проведених досліджень отримано і обґрунтовано нову гібридну модель (рис. 4.1) [142, 146], що є інтеграцією мережевої моделі (4.6) та фрагментів баз знань детермінованих, імовірностних та нечітких процесів.

Нечітка модель отримала подальший розвиток на основі розширення функції кольорових предикатних мереж Петрі та функції відображення просторових даних [143, 146, 147]

$$S_1 = \langle \tilde{P}, \tilde{T}, \tilde{F}, \tilde{M}_0, m_S, O, \tilde{M}_{0C}(f), \tilde{M}_C(f), \tilde{C}, \tilde{V}, \tilde{K}, L\{x_u\} \rangle, \quad (4.6)$$

де  $\tilde{P}$  – множина нечітких позицій;

$\tilde{T}$  – множина нечітких переходів;

$\tilde{F}$  – нечітка функція інцидентності –  $\tilde{F} : (\tilde{P} \times \tilde{T}) \cup (\tilde{T} \times \tilde{P})$ ;

$\tilde{M}_0$  – вектор нечіткого початкового маркірування нечітких позицій;

$m_S$  – ваги, які віднесені до вхідних  $\{p_i(in)\}$  і вихідних  $\{p_i(out)\}$  дуг деякого переходу  $t_i$ ;

$O$  – просторово розподілена компонента множини відображень координат

$$O = \langle X, Y, Z \rangle \quad (4.7)$$



в географічних інформаційних системах (ГІС);

$\tilde{M}_{0C}(f)$  – вектор початкового маркірування;

$\tilde{M}_C(f)$  – вектор поточного маркірування;

$\tilde{C}$  – функція кольору маркера, що визначає у даному випадку колір  $c$  кожного з маркерів  $\tilde{M}(\tilde{p}_j)$  для позицій мережі;

$\tilde{V}$  – умови здійснення переходів в залежності від кольору маркера;

$\tilde{K}$  – місткість маркерів в позиціях з урахуванням  $\tilde{C}$ ;

$L\{x_u\}, u \in U$  – деякий предикат, віднесений на моделі до множини позицій, переходів, функції інцидентності в просторі станів нечітких взаємодіючих процесів та який визначає додаткові умови виконання переходів.

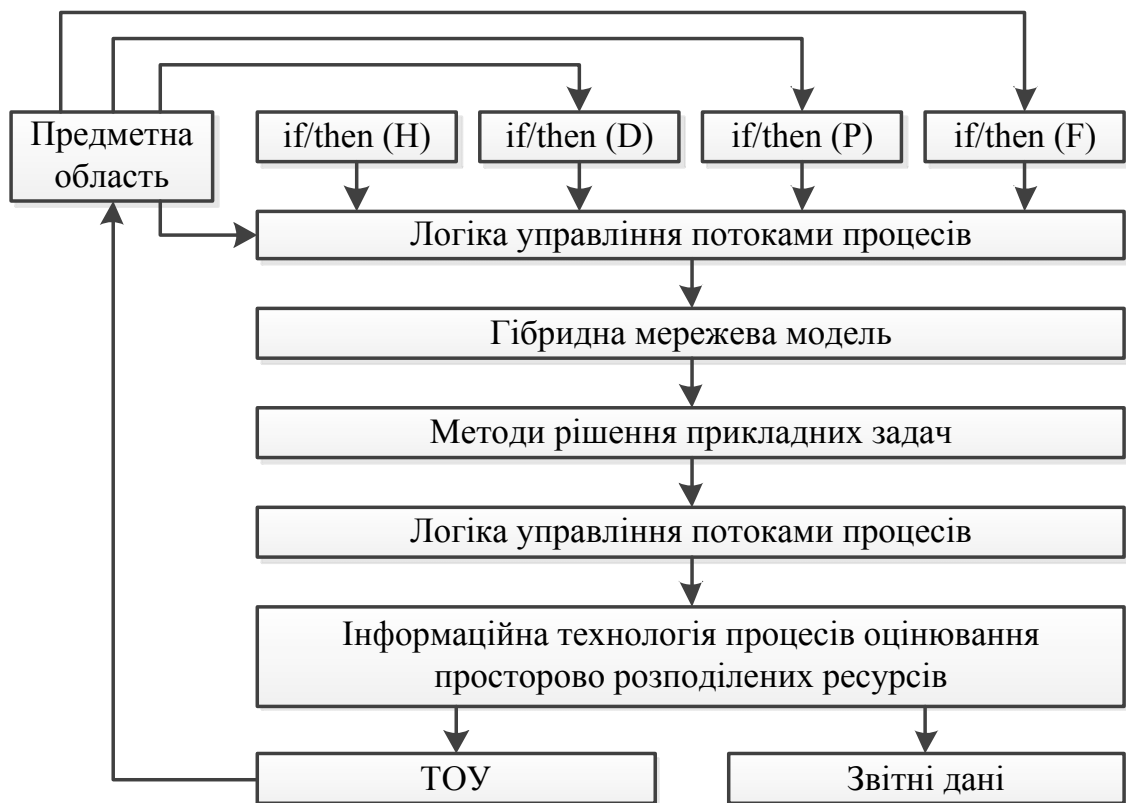


Рисунок 4.1 – Структура прийняття рішень

Враховуючи доцільність відображення деякої множини інших важливих складових, що характерні для заданої предметної області, розширимо модель

(4.6) шляхом додаткового введення наступних величин: множин нечітких позицій  $\tilde{P}$ , нечітких переходів  $\tilde{T}$ , нечіткої функції інцидентності  $\tilde{F}$ , нечіткої функції кольору  $\tilde{C}$ . Вектору нечіткого маркірування  $\tilde{M}_0$  поставлені у відповідність функції належності  $\mu(x)$ ,  $\mu(y)$  лінгвістичних термів.

При реалізації (4.7) застосування функції кольору  $C$  істотно зменшує розмірність моделі.

В обґрунтованих випадках у якості функції належності застосовують інтервальну логіку  $\Delta\mu(x)$  та її аналітичні форми [148].

Нечітке логічне виведення реалізується на основі підходів Заде-Мамдани з наступною дефаззіфікацією

$$y'_0 = \vee x'_0 \wedge \mu(x, y), \quad (4.8)$$

де  $y'_0$  – результуючий вектор;

$x'_0$  – чіткий вхідний вектор;

$\mu(x, y)$  – нечітке відношення Заде-Мамдани;

$\vee$  – оператор знаходження значення  $\max$ ;

$\wedge$  – оператор знаходження значення  $\min$ .

Вперше запропоновано гібридну модель  $S_\Sigma$  оцінювання станів складних просторово розподілених об'єктів (4.1), яка інтегрує розвинуті нечіткі кольорові мережі Петрі, моделі детермінованих, стохастичних та нечітких баз знань і логіку проявлення їх взаємодії

$$S_\Sigma = \bigcup_{k \in K} \{S_k\}, k \in K, \quad (4.9)$$

де  $K$  – множина індексів, віднесених до гібридної моделі;

$\bigcup_{k \in K}$  – операція об'єднання, яка у даному випадку відображає деяку додаткову функціональність, з відображенням типів взаємодіючих процесів, що

дозволяє варіювати розмірністю моделі та в повній мірі використовувати в задачах аналізу всю повноту типів характеристик баз даних і знань.

Введення деяких знань, які носять детермінований  $D$ , стохастичний  $P$  і нечіткий  $\tilde{F}$  характер у вигляді моделі бази знань

$$S_2 = \text{if / then}(D), S_3 = \text{if / then}(P), \tilde{S}_4 = \text{if / then}(\tilde{F}) \quad (4.10)$$

дозволяє у повній мірі використовувати в задачах аналізу всю повноту нормативної бази, стохастичних та нечітких характеристик бази знань, якщо прийняти у якості моделі  $\tilde{S}_4$  продукційну модель в нечіткому представленні  $\tilde{F}$  в задачах нечіткого логічного виведення Заде-Мамдані (Сугено).

Виходячи з особливостей предметної області дослідження, знання орієнтована модель для множини (4.1) має вигляд

$$\tilde{S}_\Sigma = \cup \{S_k\}, k \in K, \text{if / then} - D, P, \tilde{F}, \quad (4.11)$$

та враховує детермінізм об'єкта, стохастичні характеристики і показники нечіткості.

Якщо задана модель (4.6) і визначені деякі  $m_{i\alpha}$ , де  $\alpha$  – елемент множини  $A$  існуючих альтернатив  $Alt_\alpha$ , то шукана альтернатива  $Alt_\alpha \in \{Alt_\alpha\}$  може бути визначена як

$$\sum m_{i\alpha} \xrightarrow{\Omega} \text{extr}, \quad (4.12)$$

$$\Omega = \langle \tau_i < \tau, L_i = \text{true}, F_i = F^*, m_{i\alpha} = m^*, O = \text{true} \rangle,$$

де  $\Omega$  – деякі обмеження;

$\tau_i$  – час реалізації альтернативи;

$\tau^*$  – допустимий час реалізації альтернатив;

$L_i$  – множина предикатів предметної області;

$F_i$  – множина обмежень предметної області;

$F^*$  – множина допустимих значень предметної області;

$O$  – множина прямокутних координат;

$m^*$  – нижня межа допустимих значень вагових коефіцієнтів, що безпосередньо слідує з сутності множини альтернатив  $\{Alt_\alpha\}$  та діючих обмежень.

Сформульовано умови досяжності та несуперечності процесів, оснований на аналізі дозволеності переходів і виконання предикатів, віднесених до предметної області. Викладено прикладні аспекти досліджень, визначено актуальність та перспективність реалізації інформаційних технологій оцінювання земельних ресурсів.

### **4.3 Розвиток модифікованих методів налаштування та навчання функцій належності на основі багатозначної інтервальної логіки в знання орієнтованих технологіях**

У даному підрозділі розвинуто метод моделювання послідовно-паралельних процесів просторово розподілених об'єктів та удосконалено метод настроювання параметрів функцій належності нечітких баз знань, який на відміну від існуючих, ґрунтується на багатозначній інтервальній логіці, що дозволяє зменшити обчислювальну складність та час настроювання параметрів функцій за рахунок орієнтації обчислювальних процедур на виділені класи термів лінгвістичних змінних [149 – 151].

Наведемо основні етапи методу:

– фазифікація правила-продукції, вибір типів функції належності, виявлення і формулювання параметрів функцій належності;

– рішення задачі нечіткого логічного виведення Заде-Мамдані;

– дефазифікація нечіткого логічного виведення по центру мас;

- визначення оцінки очікуваного дефаззифікованого значення  $y^{(оч)}$ ;
- формування параметрів інтервалу значень для вибору знаменника дихотомії  $d2, \Delta d, \varepsilon$  і запуск процесу ітераційної апроксимації; зупинення процесу при досягненні необхідної точності  $\varepsilon$ ;
- визначення оптимальних значень множника дихотомії  $m$  для різних типів функцій належності;
- видача рекомендацій по оптимальному значенню множника для конкретного типу нечіткої величини функції належності з використанням критерію ранжирування;
- повторний прогін, уточнення параметрів дихотомії, останов.

Це не тягне за собою зміни колишніх властивостей алгоритму, за виключенням введення двох параметрів (верхня межа інтервалу дискретизації знаменника дихотомії  $d2$ , шаг дискретизації  $\Delta d$ ), але дозволяє прискорити збіжність при необхідності багатократного підбору параметра  $k$  в рамках деякої предметної області правил.

Встановлено, що така реалізація має обчислювальну складність, яка характеризується

$$O(n) \approx e^{1/\varepsilon} \quad (4.13)$$

де  $\varepsilon$  – параметр методу настроювання параметрів, що визначає норму точності апроксимації.

Обчислювальна складність запропонованого алгоритму не є оптимальною.

При розв'язанні задачі з використанням метода багатозначної логіки при деяких довільних вхідних даних виникала проблема стрибка обчислювальної складності, що приводило до недопустимо великому збільшенню часу роботи програми.

Було виконано модифікацію існуючих рішень, що дозволило істотно знизити обчислювальну складність водночас з підвищенням стабільності роботи.

Для кожного типу функції належності та кожного дільника  $d_i$  можна виміряти час роботи запропонованого алгоритму  $\tau_i$ , де  $\tau_i$  – модельний час.

Наведемо етапи модифікованого метода при настроюванні параметрів функцій належності [149]:

- фаззифікація правила продукції, вибір класів функції належності;
- формалізація параметрів функцій належності;
- рішення задачі нечіткого логічного виведення Заде-Мамдані;
- дефаззифікація нечіткого логічного виведення (центр мас);
- визначення оцінки очікуваного дефаззифікованого значення  $y^{(оч)}$ ;
- завдання дільника дихотомії  $d$  і точності обчислень  $\varepsilon$ , що вимагається;
- запуск процесу ітеративної апроксимації;
- зупинення при досягненні необхідної точності  $\varepsilon$ ;
- повторний прогін викладених вище процедур, модифікація параметрів класів функції належності;
- останов.

Вирішення даної задачі за допомогою використання методів багатозначної інтервальної логіки дозволяє визначати мінімальне значення модельного часу  $\tau$ . Проте, у ряді випадків, воно не є оптимальним. Це пов'язано з тим, що у ряді випадків не завжди можливо досягнути мінімальних значень модельного часу по причині використання у цьому підході значення  $\Delta y = |y^{(ф)} - y^{(оч)}| \leq \varepsilon$  у якості чисельника кроку апроксимації, що не завжди адекватно [149].

Знаменник кроку визначається як  $A \in (0, 1)$ ,  $A \in R$ . В запропонованому підході у ролі дільника виступає значення апріорно заданої точності обчислень  $\varepsilon$ , що дозволяє виконати умову для модельного часу  $\tau_k$ :

$$\tau_{\Delta y, A} < \tau_{\varepsilon, d} \quad (4.14)$$

де  $\tau_{\Delta y, A}$  – модельний час при використанні чисельника кроку  $\Delta y$  і знаменника  $A$ ;

$\tau_{\varepsilon, d}$  –  $\varepsilon$  і  $d$  – відповідно.

Запропоновано модифіковану програмну реалізацію викладеного рішення, що реалізує  $n$ -розрядну інтервальну настройку функцій, яку виконано мовою C++ [149].

#### **4.4 Розвиток інформаційних технологій на основі розроблених методів і моделей**

У даному підрозділі отримала подальший розвиток інформаційна технологія оцінювання станів просторово розподілених об'єктів, яка на відміну від існуючих, додатково включає моделі аналізу адекватності побудови процесів, модифікація яких дозволяє істотно знизити витрати часу на її реалізацію [143, 146, 150].

Запропоновано інформаційну технологію оцінювання станів об'єкту на основі моделі (4.14) (рис. 4.2) [143].

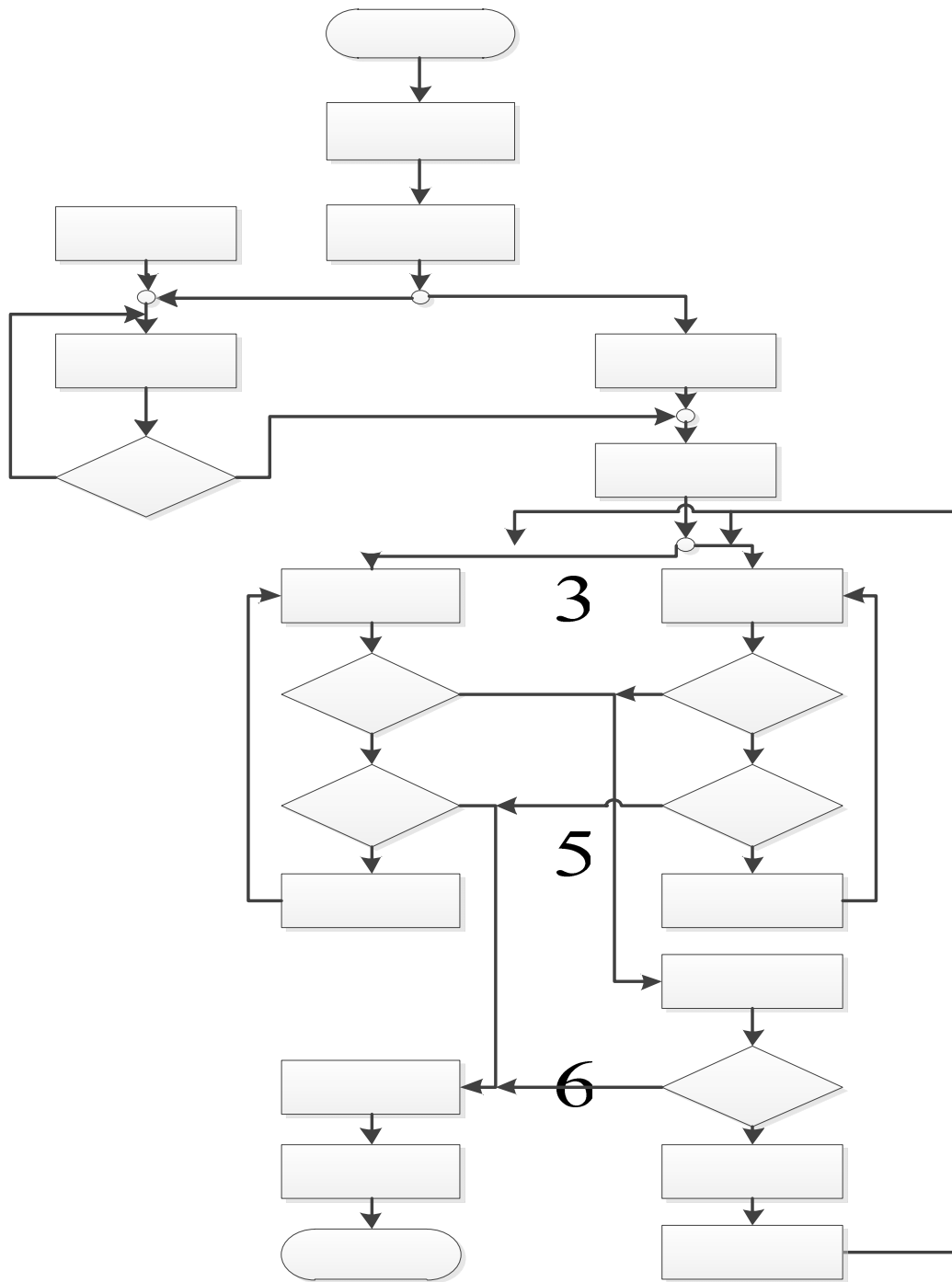


Рисунок 4.2 – Структурна схема узагальненої технології з використанням формальних моделей

Розглянемо поетапно узагальнені компоненти технології:

0 – інфологічний аналіз предметної області;

1 – на основі інфологічного аналізу здійснюється вибір об'єкта дослідження (4.1). Здійснюємо вибір території й призначення;



2 – розробка моделі об'єкта (4.11);

3 – вибір нормативної бази, яка розглядається сумісно як результат побудови моделі;

4 – визначення характеристик об'єкта. Змістовний аналіз об'єкту і предметної області дає змогу виявлення характеристик об'єкта

$$\{\text{Pr}_j\} \supseteq \{\text{Pr}_j^A\}, j \in J. \quad (4.15)$$

5 – навчання моделі об'єкта. Використовуючи знання орієнтованих технологій та моделі об'єкту, здійснюємо нечітке логічне виведення Заде-Мамдані з урахуванням норми  $|Y^{\phi} - Y^{оч}| \leq \varepsilon$ . Це потребує також застосування технологій навчання і настроювання параметрів функцій належності за критерієм мінімуму індексу нечіткості Хеммінга на множині обмежень  $F^*$  предметної області.

На основі нечіткої бази знань та знань експертів про важливість значення характеристик для даного об'єкту виконуємо ранжирування та вибираємо  $n$  найбільш значимих.

6 – перевірка логічної умови на достовірність настроювання функцій належності  $\mu$

$$\left| \mu_{(\alpha)}^{\phi} - \mu_{(\alpha)}^{оч} \right| \leq \varepsilon^*, \quad (4.16)$$

де  $\varepsilon^*$  – норма точності;

7 – після реалізації процесів навчання бази знань здійснюється визначення параметрів об'єкту  $\{X_i^P\}$ . Для найбільш значущих характеристик визначаємо параметри

$$\{X_l^P\} \supseteq \{X_l^{PA}\}, l \in L. \quad (4.17)$$

Для змістовного аналізу параметрів об'єкту оцінювання використовуємо розпаралелювання процесів, що відносяться до внутрішніх то зовнішніх взаємодій цих процесів;

8, 11, 12 – на множині об'єктів (4.1) виконується аналіз внутрішніх взаємодій за критерієм досяжності та відсутністю конфліктів з наступною модифікацією моделі на етапі 14;

9, 10, 13 – на множині об'єктів (4.1) виконується аналіз зовнішніх взаємодій за критерієм досяжності та відсутністю конфліктів з наступною модифікацією моделі на етапі 17;

15, 16, 18, 21 – аналіз та усунення властивостей неадекватності за критеріями повноти  $Pl$  та несуперечності  $Npt$

$$Pl = \begin{cases} 1, & true, \\ 0, & false, \end{cases} \quad Npt = \begin{cases} 1, & true, \\ 0, & false \end{cases} \quad (4.18)$$

з наступною модифікацією моделі згідно етапу 21.

Фактом задовільного рішення за результатами аналізу внутрішньої  $V$  та зовнішньої  $W$  взаємодії процесів звичайно є знаходження

$$\{AX^{V,W}\} = true, \quad (4.19)$$

в іншому випадку:

$$Pl, Npt = false \quad (4.20)$$

та реалізуються процедури модифікації внутрішніх та зовнішніх умов, уточнення множини параметрів виду

$$if \{A_N AX^{V,W}\} \rightarrow false, then U\{A_N AX^{V,W}\} \rightarrow true. \quad (4.21)$$

Після виконаних дій по модифікації, згідно етапу 21, умов (4.21) отримуємо значення  $\{A_N AX^{V,W}\} \rightarrow true$ .

19, 20, 22 – етапи прийняття рішень про стан об'єкту, формування звітів, останов.

Розглянуті прикладні аспекти реалізації методів та інформаційних технологій в задачах оцінювання складних просторово розподілених об'єктів [152] на прикладі оцінювання привабливості території з використанням геоінформаційних технологій [145].

Розроблені гібридні моделі та інформаційні технології є теоретичною базою практичних аспектів аналізу станів процесів в просторово розподілених об'єктах в знання орієнтованих технологіях оцінювання станів просторово розподілених об'єктів [142, 143, 150, 152 – 155].

Розроблено, досліджено та впроваджено наступні основні практичні результати:

- нові методи та інструментальні засоби оперативного аналізу в інформаційних технологіях оцінювання станів складних просторово розподілених об'єктів, що дозволило зменшити вплив суб'єктивного фактору на результати оцінювання та підвищити достовірність рішень, які приймаються, при скороченні витрат часу;

- на основі нових методів навчання функцій належності запропоновано та обґрунтовано алгоритми та інструментальні засоби інформаційних технологій прийняття рішень в знання орієнтованих технологіях з оцінювання територій, що дозволяє використовувати дані з існуючих електронних ресурсів (топографічні карти, кадастрові плани, матеріали нормативної грошової оцінки населених пунктів, генеральні плани розвитку територій та ін.) та знання експертів для оперативного аналізу та оцінювання станів земельних ресурсів.

Результати науково-дослідної роботи було впроваджено в роботах Державного підприємства «Східгеоінформ» (Харків, Україна) для оцінювання станів складних об'єктів, оперативного аналізу і оцінювання територій, прийняття рішень в знання орієнтованих технологіях, що дозволило скоротити

час виконання робіт (за даними експертів) до 12%; в роботах ТОВ «Українське агентство земельних ресурсів» (АР Крим, Україна) при оцінюванні станів складних просторово розподілених об'єктів, що дозволило скоротити час виконання робіт (за даними експертів) до 14%; у навчальному процесі кафедри геоінформаційних систем та геодезії Харківської національної академії міського господарства.

## 5 ДОСЛІДЖЕННЯ ТА РОЗРОБКА МЕТОДІВ ПОДАННЯ НЕЧІТКИХ МОДЕЛЕЙ ЗАСОБАМИ РЕЛЯЦІЙНИХ СИСТЕМ

Комп'ютерні технології разом з організацією інтелектуальних обчислень переживають свій розквіт. Це пов'язано, головним чином, з потоком нових ідей, породжених дослідженнями у порівняно новій галузі комп'ютерних наук, яка утворилася на перетині штучного інтелекту, статистики та теорії баз даних. Зараз відбувається стрімке зростання кількості програмних продуктів, що використовують нові технології, а також типів завдань, де їх застосування приносить значний економічний ефект. Елементи автоматичної обробки й аналізу даних стають невід'ємною частиною концепції електронних сховищ даних та організації інтелектуальних обчислень.

Системи, побудовані шляхом поєднання баз даних (БД) і нечіткої логіки, дозволяють істотно розширити функціональні можливості та коло задач, що розв'язуються. У той же час, нечітка база даних розглядається як узагальнена версія традиційної бази даних, інакше кажучи, остання є спеціальним випадком нечіткої бази даних.

Проблема проектування БД зараз є об'єктом зростаючого інтересу широкого кола фахівців у різних галузях науки і техніки. Проте, у багатьох випадках інформація, що зберігається у БД, не завжди точно відображає характеристики досліджуваного об'єкта. Це пояснюється низкою об'єктивних і суб'єктивних причин. По-перше, даний об'єкт може мати складну функціональну структуру, і дослідники не в змозі повністю її з'ясувати. По-друге, через різні об'єктивні причини фахівці часто-густо не мають можливості отримати точну інформацію, а отримують лише деяку її частину. По-третє, на практиці зустрічається досить багато атрибутів об'єкта, які можна оцінити лише якісними величинами, а не кількісними. Такі типи даних часто зустрічаються в експертних системах, системах прийняття рішень та інтелектуальних БД. Тому сьогодні проблема проектування нечіткої моделі БД

і створення методик обробки неточної та абстрактної інформації стає все більш актуальною.

Багато вчених розглядали проблеми проектування нечітких баз даних, особливо на основі реляційної моделі. Розроблено різні варіанти нечітких моделей реляційних баз даних. Результати досліджень викладені, наприклад, у роботах: Buckles B.P., Petty F.E., Shenoі S., Melton A., Vila M.A., Lipski W Jr., Prade H., Zemankova-Leech M., Kandel A., Umamo M., Ding-An Chiang, Lui R. Chow, Nan-Chen Hsien, Le Tien Vuong, Ho Thuan, Truong Due Hung та інших. Серед них нечіткі моделі Buckles B.P. і Petry F.E., Shenoі S. і Melton A.

Завдяки роботам в області баз даних таких вчених як Мартін Дж., Мейер Д., Дейт К.Дж, Codd E.F., Fagin R., Armstrong W.W. та ін., теорія класичних баз даних була достатньо розвинена з математичної точки зору. У той же час теорія нечітких баз даних ще не може вважатися закінченою, і в ній залишається чимало питань, що потребують вирішення.

### **5.1 Особливості нечітких моделей і нечіткого моделювання**

Визначимо деякі вимоги до нечітких моделей. Унікальною особливістю таких моделей є те, що вони повинні забезпечувати гнучку стратегію обробки різнорідних процесів, які представляють дані і знання в досить нечіткому просторі станів об'єктів аналізу.

У зв'язку з цим можна стверджувати, що нечіткі моделі орієнтовані на моделювання конструкцій, для яких характерно:

- функціонування на рівні лінгвістичних термів (нечітких множин);
- характеристики системи можуть бути зображені в такому самому лінгвістичному форматі;
- представлення й обробка даних в умовах невизначеності.

Нечітка система може бути представлена набором нечітких моделей в залежності від ступеня деталізації нечітких множин, що використовуються для представлення спеціалізованої моделі із середовищем моделювання. Ступінь

деталізації лінгвістичних представлень (міток) визначає об'єкт досліджень, а логіка лінгвістичного представлення процесів описується рівнем логічно-орієнтованих відношень [118].

Логічність запропонованої структури очевидна, однак вона не реалізує окремі моделі представлення середовища на різних ієрархічних рівнях з єдиною методологічною базою і з єдиним математичним апаратом, що не завжди розумно й істотно ускладнює обчислювальні процедури й інтерпретацію отриманих результатів.

Розглянемо деяку узагальнену структуру нечітких моделей. Нечітка модель може бути представлена на основі цілеспрямованої взаємодії трьох основних модулів: нечіткого кодера, модуля обробки, нечіткого декодера. Інструментом формування інтерфейсу між середовищем моделювання і власне обчислювальним модулем моделі є нечіткі множини.

Модуль обробки може істотно змінюватися в залежності від специфічної проблеми й особливостей предметної області. Зокрема, перспективним є відображення правил у формі нечітких нейронних мереж. У цій роботі також обґрунтовується структура моделювання предметних областей, що ґрунтується на нечітких даних і знаннях. Справедливість запропонованої структури очевидна, однак багато в чому не враховує вимог перевірки забезпечення адекватності даних і знань предметної області.

Розглянемо деякі особливості нечіткого кодування і декодування. Істотна роль нечіткого кодера і нечіткого декодера полягає в тому, що вони повинні кодувати / декодувати інформацію, що виходить з середовища або спрямовану до середовища, у якому відбувається моделювання. Інформація може бути різномірною за характером, включаючи точні числові дані, інтервали чітких і нечітких даних і знань, а також лінгвістичні змінні. Перетворення цієї зовнішньої форми інформації у внутрішній формат, що використовується в нечіткій моделі, реалізованій через різні процедури відповідності, є важливою задачею. Часто ці процедури залежать від можливості надбання даних і знань, а також потреб користувача.

Такі перетворення звичайно здійснюються за допомогою функцій належності і процедур дефазіфікації.

## 5.2 Нечіткі граматики

Нечіткі граматики і нечіткі мови – суть нечіткі символно-орієнтовані формалізми, що можуть досить просто використовуватися в описі різних систем. Особливий інтерес вони представляють у задачах аналізу часових рядів і розробки класифікаторів сигналу. У формальному представленні нечітка граматика заснована на класі породжуючих формальних граматик, які засновані на загальноприйнятій теорії Хомського, і визначена четвіркою

$$G = (V_N, V_T, P, \sigma), \quad (5.1)$$

де  $V_T$  - множина термінальних символів (алфавіт);

$V_N$  - множина нетермінальних символів, причому  $V_N \cap V_T = \emptyset$ ;

$P$  - список продукційних правил;

$\sigma$  - початковий символ.

Елементи  $P$  у (5.1) мають форму

$$a \xrightarrow{\beta} b,$$

де  $a, b$  - два рядки, що належать до  $V_T \cup V_N$ , а  $\beta$  визначає важливість правила.

У практичних реалізаціях нечіткі формальні граматики є громіздкими, їхнє застосування для цілей аналізу властивостей нечітких систем великої розмірності ускладнене.

Нечіткі моделі, що базуються на правилах, засновані на правилах обчислення з нечіткими множинами є ефективним засобом представлення



взаємодіючих динамічних процесів, що відображають дані і знання у нечіткому просторі станів.

У загальному вигляді продукція – це вирази такого вигляду:  $(i); Q; P; A \rightarrow B, N$ , де  $i$  - ім'я продукції. Як ім'я може виступати також порядковий номер. Елемент  $Q$  характеризує сферу застосування продукції. Важливою складовою продукції є їхнє ядро -  $A \rightarrow B$ , яке в загальному випадку містить *if / then* структури типу: *if A then B ELSE*. У реальних конструкціях ядра складова  $A$  характеризується складною структурою, що включає також деякі предикати, логічні оператори типу *and, or, not* та їхні похідні. Компонента  $P$  визначає умови застосовності ядра продукції, що часто представлено деяким предикатом. Компонента  $N$  визначає постумови продукції і звичайно актуалізується при виконанні правила.

У процедурах керування логічним виведенням часто використовують такі стратегії:

1. Принцип "купки книг" заснований на тому, що найбільш часто використовувані продукції є найбільш важливими і корисними. Продукції утворюють "купку", на самому верху якої знаходиться найбільш часто використовувана продукція. Цей принцип застосовують, якщо правила незалежні одне від одного або застосована така організація їхньої обробки, що вона не залежить від їхнього перебування. Підхід набув практичного застосування в деяких виробничих системах на основі робототехнічних комплексів.

2. Принцип "найбільш довгої умови". У цьому випадку з готових правил вибираємо те, у якого найбільш довга умова (антецедент) виконання ядра. Це обумовлено гіпотезою, що правила, які визначають рішення у вузьких класах ситуацій, є найбільш важливими, а значить і невідкладними. Вони, у свою чергу, враховують більшу кількість інформації стосовно більш загальних ситуацій. Очевидно, що підхід застосовують у випадку, якщо правила добре структуровані і прив'язані до типових ситуацій, що на практиці викликає певні труднощі.

3. Принцип "шкільної дошки", заснований на ідеях так званих спускових функцій. У цьому випадку в системі виділяється область пам'яті, що є деяким аналогом шкільної дошки, на якій можуть писати і при необхідності усувати деякі дані. У цій області паралельно виконувані процеси знаходять інформацію, що визначає умови їхнього запуску і застосовності ядра правил. На "дошці" можуть фіксуватися окремі фрагменти бази знань. З принципом "шкільної дошки" може бути також задіяна і навіть сполучена процедура метаправил для перевірки деяких умов.

4. Принцип "метапродукцій", заснований на ідеї введення в систему спеціальних метаправил, задачею яких є спеціальна організація і керування продукціями в умовах їхнього неоднозначного вибору і запуску на виконання. Природно, що база знань у цьому випадку також повинна бути добре структурованою.

5. Принцип "пріоритетного вибору". Цей принцип пов'язаний із процедурами організації статичних і динамічних пріоритетів. Статичні пріоритети формуються в процесі створення бази знань на основі даних і знань про відносну важливість виконання окремих правил або групи правил. Динамічні пріоритети формуються в процесі їхнього виконання за деякими заздалегідь визначеними критеріями. Найпростішим з них може бути максимально припустимий час чекання і запуску окремих правил на виконання.

6. Принцип "керування по іменах", заснований на процедурах, близьких до визначення динамічних пріоритетів. У цьому випадку для імен продукцій пропонуються деякі граматики або процедури, що визначають підмножину правил, а найчастіше одне з правил, яке запускається на виконання. Очевидно, що даний підхід громіздкий і вимагає гарної структурованості правил, а це викликає труднощі при розвитку бази знань на основі введення деякої підмножини нових правил.

На цей час існує велика кількість літератури по різних концептуальних і прикладних аспектах систем, заснованих на правилах, їх функціонуванню і можливій оптимізації. Більшість з них має справу з процесами надбання й

обробки знань. У зв'язку з цим нас можуть цікавити такі фундаментальні аспекти:

- надбання знань;
- процедури аналізу властивостей адекватності надбаних знань;
- розробка ефективних механізмів логічного виведення, які забезпечують обчислення і керування з нечіткими правилами.

Проблеми надбання знань і розробки ефективних механізмів логічного виведення загалом вже розроблено і вирішено, а їхні результати можуть бути використані в теоретичних і практичних розробках. Питання ж розробки ефективних формалізованих процедур аналізу властивостей адекватності набутих знань, представлених на основі нечітких правил продукції, в основному не досліджені і чекають свого подальшого дослідження.

### **5.3 Бази даних в системах нечіткого моделювання**

Останнім десятиріччям спостерігається гібридизація методів інтелектуальної обробки інформації. М'які обчислення поєднують такі області як нечітка логіка, штучні нейронні мережі, видобування знань, бази даних, імовірнісні міркування, еволюційні алгоритми та низку інших. Вони доповнюють одне одного й використовуються у різних комбінаціях для створення гібридних інтелектуальних систем.

Не залишилися осторонь від цікавого й сучасного напрямку дослідники в галузі баз даних. Розробляється нечітка реляційна алгебра й спеціальні розширення структурованої мови (SQL) для нечітких запитів. У цій області інтенсивні дослідження проводять європейські вчені Д.Дюбуа й Г.Праде. Формується перспективний напрямок у сучасних системах обробки інформації – нечіткі запити до баз даних (fuzzy queries).

У цьому контексті можна розглядати два основних питання, найбільш актуальних в цей час: як проектувати, де й у яких структурах зберігати нечіткі дані систем такого класу. Вирішення цих проблем відкриє шляхи інтеграції

реляційних баз даних, що накопичили колосальні обсяги інформації, і систем на основі нечіткої логіки.

Метою проведених досліджень є розробка методів зберігання й обробки нечітких даних засобами реляційної моделі, орієнтованої на реалізацію в середовищі сучасних систем управління базами даних. Особлива увага приділена обґрунтуванню вибору схеми реляційної моделі даних для подання функцій належності лінгвістичних змінних.

Розглянемо класичний підхід до побудови реляційного відношення, запропонований Е.Ф. Коддом, і визначимо основні властивості відношень при розширенні множини доменів.

Основним структурним компонентом даних у реляційній моделі даних (РМД) є  $n$ -арне відношення, що є підмножиною кортежів декартового добутку доменів, тобто множини значень елементів даних. Для заданих кінцевих множин  $D_1, \dots, D_n$  (не обов'язково різних по типу) декартовим добутком  $D_1 \times \dots \times D_n$  називається множина добутків виду  $d_1, \dots, d_n$ , де  $d_1 \in D_1, \dots, d_n \in D_n$ . Відношенням  $R$ , визначеним на множинах  $D_1, \dots, D_n$ , називається підмножина добутку (декартовий добуток)  $D_1 \times \dots \times D_n$ , тобто  $R \subseteq D_1 \times D_2 \times \dots \times D_n$ . Множина  $D = \{D_1, \dots, D_n\}$  називається доменом. Домени – це однотипні семантично однозначні (однакові за змістом) значення елементів даних. Елементи декартового добутку  $d_1, \dots, d_n$  називаються кортежами, число  $n$  визначає ступінь відношення; кількість кортежів визначає потужність відношення [156].

Нехай  $a = \{a_1, \dots, a_n\}$  – множина імен, тоді однозначне відображення виду  $\rho : A_i \rightarrow D_i$ , де пару  $A_i = \langle \rho(A_i), D_i \rangle$  називають атрибутами відношення. Схемою відношення  $R$  будемо називати вираз  $S(A_1, \dots, A_n)$  в якому всі атрибути  $A_i$  різні. При цьому екземпляр відношення  $R(S)$  визначається як підмножина декартового добутку доменів  $r_i \subseteq \rho(A_i) \times \dots \times \rho(A_n)$ . Екземпляр відношення зі схемою  $R_i$  будемо позначати як  $R_i(r_i)$ . Зазначимо, що перестановка атрибутів у схемі не породжує нового стану БД. Таким чином, множина атрибутів  $\{A_1, \dots, A_n\}$  завдає тип відношення й визначає його властивості.

Схему БД будемо позначати як множину схем відношень  $U = \{R_1, \dots, R_n\}$ , де  $R_i \in R$  і всі  $R_i$  різні. Відповідно, екземпляр БД будемо позначати множиною екземплярів відношень  $U(r_1, \dots, r_n)$ . Концептуально реляційна БД є інформаційною моделлю предметної області (ПО), такою, що кожний екземпляр відповідає стану ПО у певний момент часу. Кожний стан моделюється впорядкованою сукупністю значень елементів даних, що відповідають значенням властивостей об'єктів ПО. Об'єкту певного типу відповідає кортеж відношень.

Для задач аналізу даних визначимо додатковий тип відношень, що визначає приналежність існуючих даних до деякого заданого числового відрізка, що характеризує інформаційний об'єкт – відношення фаззифікації.

Будь-яку лінію на координатній площині можна подати у вигляді бінарного відношення, де  $Dom R$  визначається значеннями осі абсцис, а  $Im R$  визначається значеннями осі ординат. У задачі фаззифікації діаграма містить три показники, які необхідно враховувати при формуванні відношення.

Під нечіткою змінною будемо розуміти набір  $(N, X, Y)$ , де  $N$  – назва змінної,  $X$  – область міркувань,  $Y$  – нечітка множина на  $X$ . Використовуючи таке визначення, задамо три домена, що відповідають елементам змінної. Нехай  $N = \{n_1, \dots, n_m\}$ ,  $Y = \{0, 0.1, \dots, 1\}$ ,  $X = \{x_0, \dots, x_k\}$ . Значення  $X$  і  $Y$  відповідають обраній шкалі дискретизації координатних осей і описують область належності до параметра  $N$ . Для кожного параметра  $N$  робимо вибірку даних за значеннями  $X$  і будемо діаграму належності, наведену на рис. 5.1.

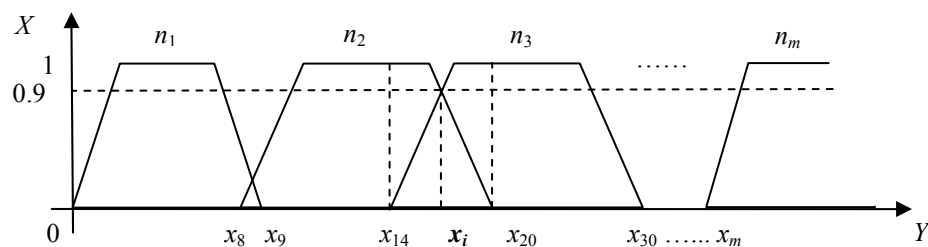


Рисунок 5.1 – Графік функції належності

Для розглянутого випадку визначимо відповідні домени для подання нечіткої змінної.

$$\begin{aligned} D_1 &= \{n_1, n_2, n_3, \dots, n_m\}; \\ D_2 &= \{x_0, \dots, x_8, x_9, \dots, x_{14}, \dots, x_i, \dots, x_{20}, \dots, x_{30}, \dots, x_m\}; \\ D_3 &= \{0, 0.1, \dots, 1\}. \end{aligned} \quad (5.2)$$

Задамо множину імен доменів і побудуємо відображення. Для множини імен  $A = \{A_1, A_2, A_3\}$  відображення  $\rho : (A_1 \rightarrow D_1; A_2 \rightarrow D_2; A_3 \rightarrow D_3)$  визначають множину атрибутів  $A = \{A_1, A_2, A_3\}$  і відповідну схему відношення  $S(A_1, A_2, A_3)$ . Структура відношення  $R^f$  у табличному вигляді буде називатися відношенням фаззифікації. Можливі перетини діаграм фаззифікації визначають той факт, що всі значення можуть повторюватися відносно одне одного [157, 158].

Наприклад, розглянемо фрагмент відношення для показників  $\{x_{14}, x_i, x_{20}\}$ , наведений на рис. 5.2.

Таким чином, можна говорити про відношення, що включає повний набір кортежів декартового добутку доменів  $D_1 \times D_2 \times D_3$  (повний декартовий добуток, у цьому випадку, визначається скінченністю значень системи координат). Виходячи із цього факту, можна визначити, що ключем відношення буде множина всіх атрибутів  $K = \{A_1, A_2, A_3\}$ .

$A_1$	$A_2$	$A_3$
$n_2$	$x_{14}$	0
$n_3$	$x_{14}$	1
$n_2$	$x_i$	0.9
$n_3$	$x_i$	0.9
$n_2$	$x_{20}$	1
$n_3$	$x_{20}$	0
...	...	...

Рисунок 5.2 – Фрагмент структури даних для обраних показників

Очевидно, що інформативність кортежів визначається значеннями на діаграмі фаззифікації. У розглянутій задачі, необхідно враховувати ще один показник – множину значень вибірки із БД, для якої будується діаграма. Тобто необхідно встановити зв'язок між доменом відношення  $R^f$ , що відображує значення осі абсцис і доменом із БД, що відображує значення параметрів фаззифікації.

Таким чином, наступна задача, яку необхідно розглянути, пов'язана з інтеграцією відношення  $R^f$  із БД, що зберігає основні показники.

Розглянемо задачу в загальному вигляді.

Нехай  $U(R_1, \dots, R_n)$  – БД, що зберігає основні дані,  $R^f(A_1, A_2, A_3)$  – відношення фаззифікації. Задача має сенс, якщо в БД  $U$  існує параметр, для якого виконана фаззифікація.

Щоб організувати спільну роботу з базами даних  $U$  і  $R$ , формалізуємо процедуру інтеграції, спираючись на поетапну нормалізацію. Структура БД  $U$  отримана на підставі функціональних залежностей  $F = \{M_i \rightarrow N_i\}$ , де  $M_i, N_i \in U$ .

Виділимо одну залежність, яка містить атрибут з параметрами фаззифікації, і позначимо її як  $W \rightarrow V$ , причому  $W$  і  $V$  можуть бути множинами. Відношення  $R^f$  містить одну залежність виду  $F' = \{A_1, A_2, A_3 \rightarrow A_1, A_2, A_3\}$ . Спираючись на аксіоми виводу, можна одержати еквівалентну множину

$$F' = \{A_1, A_2, A_3 \rightarrow A_1; A_1, A_2, A_3 \rightarrow A_2; A_1, A_2, A_3 \rightarrow A_3\}. \quad (5.3)$$

Нехай параметр фаззифікації відповідає атрибуту  $A_2$ , тоді для визначення типу зв'язку необхідно одержати множину  $F = F \cup F'$  і розглянути два випадки, що впливають на правила нормалізації.

1.  $A_2 \in W$  – пошук неповних залежностей: якщо виконуються функціональні залежності  $\xi \rightarrow \zeta$  і  $\omega \rightarrow \zeta$ , причому  $\omega \subseteq \xi$ , тоді залежність  $\omega \rightarrow \zeta$  є неповною.

2.  $A_2 \in V$  – пошук транзитивно залежних елементів: якщо виконуються функціональні залежності  $\xi \rightarrow \omega$  і  $\omega \rightarrow \zeta$ , тоді елемент  $\zeta$  є транзитивно залежним.

Існування таких залежностей дозволить виконати коректну декомпозицію й встановити зв'язок між базами даних  $U$  і  $R^f$ .

Якщо  $A_2 = W$  або  $A_2 = V$ , то процес декомпозиції призводить до другої або третьої нормальної форми. Якщо рівняння не виконуються, то неможливо організувати підтримку однозначності зв'язаних даних, тому що асоціація між відношеннями буде відповідати типу «багато-до-багатьох».

Як правило, на практиці умови рівняння не виконуються і для нормалізації необхідно виділити базис  $F$  і повторити процедуру декомпозиції.

Враховуючи той факт, що структура БД не повинна змінюватися, необхідно зв'язати відношення фаззифікації  $R^f$  і БД  $U$  без реструктуризації схеми даних. Використовуючи діаграму моделі «сутність-зв'язок», представимо  $R^f$  і  $U$  у вигляді сутностей (рис. 5.3).

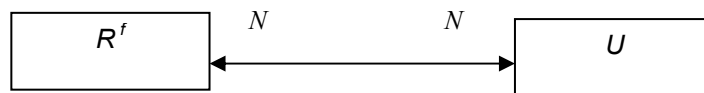


Рисунок 5.3 – Діаграма «сутність-зв'язок» між БД  $U$  і  $R^f$

Для усунення зв'язку «N:N» запровадимо додаткову сполучну сутність, яка вирішить проблему цілісності даних за рахунок визначення нових типів зв'язків. «Сутність-зв'язок» буде містити один атрибут – сполучний для  $R^f$  і  $U$ , причому з об'єктивних причин він буде ключовим. На рис. 5.4 показана діаграма інтеграції БД і відношення фаззифікації з підтримкою однозначності зв'язку.

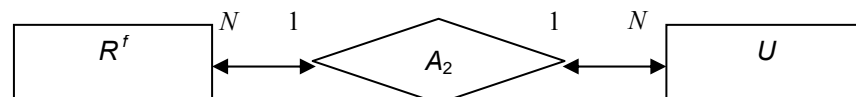


Рисунок 5.4 – Усунення зв'язку «багато-до-багатьох»



Виходячи з опису концептуальної схеми ПО можна бачити, що для коректного з'єднання  $R^f$  і  $U$  необхідно побудувати проміжну таблицю. Такий підхід гарантує погодженість даних для будь-яких параметрів фаззифікації.

Покажемо, що для даної задачі цілком коректні результати при виконанні з'єднання відношень з асоціацією типу «N:N». Можливі значення атрибута  $A_1 \in U$  можуть повторюватися стільки разів, скільки це значення перетинає границі діаграми фаззифікації по осі ординат. Тобто кожному значенню атрибута  $A_1$  відповідає рядок унікальних даних. Якщо  $A_1$  не є ключем, і значення повторюються, то, за визначенням множини, у рядку повинно бути хоча б одне відмінне значення. У термінах розв'язуваної задачі необхідно аналізувати всі такі рядки. В атрибуті  $A_1$  відношення  $R^f$  також повторюються значення, які необхідно проаналізувати, причому в різних комбінаціях [159].

Таким чином, у загальному вигляді для аналізу даних, що накопичуються в реляційних базах даних, досить побудувати відношення фаззифікації й встановити зв'язок з атрибутом (атрибутами), за значеннями якого необхідно провести відповідний аналіз.

## ВИСНОВКИ

Синтезовано еволюційні архітектури, де в якості вузлів використано введені нами орто-нейрони, Q-нейрони, нео-фаззі нейрони, адаптивні вейвлони, що дозволило підвищити як апроксимуючі властивості, так і швидкість навчання. З використанням цих типів нейронів синтезовано МГУА-вейвлет-нейро-фаззі мережі, що відрізняються обчислювальною простотою, але можуть оброблювати інформацію лише у пакетному режимі, тому було введено гібридні вейвлет-нейро-фаззі-мережі, що об'єднують переваги як каскадних, так і МГУА-систем. При цьому в якості функції активації-належності використовувалися ядерні конструкції, що забезпечило низку переваг у швидкості збіжності та якості апроксимації.

Запропоновано й теоретично обґрунтовано еволюційну нейро-систему, де в якості вузлів використовуються не нейрони, а нейромережі зниженої розмірності, засновані на різних принципах навчання: оптимізаційних, «нейрони в точках даних», опорних векторів. Доведена оптимальність таких мереж в порівнянні із багат шаровими МГУА-нейромережами. Обчислювальні експерименти підтвердили ефективність підходу, що розвивається.

Розроблено нові методи ітераційного аналізу індукованих відеоданими багатовимірних часових рядів.

Побудовано схему адаптивної еволюційної системи. Розроблено моделі та запропоновано адаптивні процедури контролю, ідентифікації та керування в еволюційних системах, що мають прямокутну функцію належності. Перевагою процедур є їх відносна простота при обчислювальній реалізації.

Розроблено еволюційні гібридні моделі ідентифікації нелінійних динамічних об'єктів зі змінною структурою на основі нечіткої нейронної мережі та штучних імунних систем. При вирішенні задач ідентифікації нелінійних об'єктів запропоновані моделі еволюціонують, змінюючи як свої параметри, так і свою структуру.

Запропоновано методи структурної та параметричної адаптації еволюційних гібридних моделей на основі штучних імунних систем;

Запропоновано нову модель кодування параметрів еволюційних гібридних моделей, що настроюються, у вигляді структурованого адаптивного мультиантитіла.

Розроблено методи прийняття рішень про стан об'єктів за умов невизначеності, що ґрунтуються на еволюційних нечітких інтервальних моделях, а також моделі в системах з інтервальними функціями належності та алгоритми настроювання функцій належності нечіткого логічного виведення Заде-Мамдані в задачах оперативної оцінки технічного стану просторово розподілених об'єктів з використанням нечіткої інтервальної логіки, що задовольняють критеріям точності і швидкодії.

Досліджено та теоретично обґрунтовано метод подання нечітких моделей засобами реляційних систем, які на відміну від існуючих підходів, дає змогу розв'язування завдань аналізу даних інтегрованих систем – баз даних та систем з нечіткою логікою.

**ПЕРЕЛІК ДЖЕРЕЛ**

1. Pham D. T. Neural Networks for Identification, Prediction and Control / Pham D. T., Liu X.. – London: Springer – Verlag, 1995. – 238 p.
2. Ivakhnenko G.A. Self-organizationalization of neuro net with active neurons for effects of nuclear test explosions forecasting / Ivakhnenko G.A. // System Analysis Modeling Simulation . – 1995. – 20. – P. 107-116.
3. Ivakhnenko A.G. Self-organizationalization of the neural networks with active neurons / Ivakhnenko A.G., Ivakhnenko G.A., Muller J.A. // Pattern Recognition and Image Analysis . – 1994 . - 4(2). – P. 177-188.
4. Ivakhnenko A.G. Inductive sorting-out GMDH algorithms with polynomial complexity for active neurons of neural network / Ivakhnenko A.G., Wuensch D., Ivakhnenko G.A. // Neural Networks. – 1999. – 2. – P. 1169-1173.
5. Kondo T. GMDH neural network algorithm using the heuristic self-organization method and its application to the pattern identification problem / Kondo T. // Proceedings of the 37th SICE Annual Conference. - 1998. - P. 1143-1148.
6. Kondo T. Identification of radial basis function networks by using revised GMDH-type neural networks with a feedback loop / KondoT. // Proc. of the SICE Annual Conference 2002, International Session Papers. 2002. - P. 2882-2887.
7. Kondo T. Structural Identification of the Multi-layered Neural Networks by using Revised GMDH-type Neural Network Algorithm with a Feedback Loop / Kondo T., Pandya A.S. // Proc. of the SICE Annual Conference 2003, International Session Papers. - 2003. - P. 2806-2811.
8. Kondo T. Revised GMDH-type Neural Networks using PSS Criterion / Kondo T. // Proc. of the 47th IEEE International Midwest Symposium on Circuits and System, - 2004. - P.81-84.
9. Kondo T. Feedback GMDH-type neural network algorithm using prediction error criterion for self-organization / Kondo T. // Proc. of SICE Annual Conference 2008. - P. 1044-1049.

10. Kondo T. Feedback GMDH-type neural network using prediction error criterion and its application to 3-dimensional medical image recognition / Kondo T // Proc of SICE Annual Conference. - 2008. - P.1050-1055.

11. Kondo T. Nonlinear system identification by feedback GMDH-type neural network with architecture self-selecting function / Kondo T., Ueno J. // Proc Intelligent Control (ISIC), 2010 IEEE International Symposium on Publication Year: - 2010. - P. 1521 - 1526.

12. Ichihashi H. Selection of the optimum number of hidden layers in neuro-fuzzy GMDH / Ichihashi H., Harada N., Nagasaka K. // Proc. of International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium. – 1995. - 3. - P. 1519 - 1526.

13. Ohtani T. Structural learning with M-apoptosis in neurofuzzy GMDH / Ohtani, T., Ichihashi H., Miyoshi T., Nagasaka K. // Proc. of IEEE World Congress on Computational Intelligence. – 1998. – 2. - P. 1265-1270.

14. Ohtani T. Structural learning of neurofuzzy GMDH with Minkowski norm / Ohtani T., Ichihashi H., Miyoshi T., Nagasaka K., Kanaumi Y. // Proc of Int. conf, Knowledge-Based Intelligent Electronic Systems – 1998. - 2. – P. 100 - 107.

15. Ohtani T. Automatic variable selection in RBF network and its application to neurofuzzy GMDH / Ohtani T. // Proc. Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies – 2000. – 2. – P. 840 - 843

16. Hwang H.-S. Container Terminal Demand Forecasting Framework Using Fuzzy-GMDH and Neural Network Method / Hwang H.-S., Bae S.-T., Cho G.-S.// Proc. of Second International Conference on Innovative Computing, Information and Control – 2007. – P. 119.

17. Mamdani E.H. Application of Fuzzy Logic to Approximate Reasoning Using Linguistic Synthesis / Mamdani E.H. // IEEE Trans. Computers. –1977. - 26(12). – P.1182-1191.

18. Zhao X. A novel NF-GMDH-IFL and its application to identification and prediction of nonlinear systems / Zhao X., Song Z., Li P. // Proc. IEEE Region 10

Conference on Computers, Communications, Control and Power Engineering. – 2002. - 3. – P. 1286-1289.

19. Bodyanskiy Ye. The neo-fuzzy neural network structure optimization using GMDH-for the solving forecasting and classification problems / Bodyanskiy Ye., Zaychenko Yu., Pavlikovskaya E., Samarina M., Viktorov Ye. // Proc. Int. Workshop on Inductive Modeling. – Krynice, Poland, 2009. – P.77-89.

20. Harris C. J. Intelligent Control. Aspects of Fuzzy Logic and Neural Nets / Harris C. J., Moore C. G., Brown M. – Singapore: World Scientific. – 1993. – 380 p

21. Brown M. Neural networks for modelling and control / Brown M., Harris C. J. / Ed. by C. J. Harris "Advances in Intelligent Control". – London: Taylor and Francis, 1994. – P. 17-55.

22. Ивахненко А.Г. Самообучающиеся системы распознавания и автоматического управления / Ивахненко А.Г. – Киев: Техніка, 1969 – 392 с.

23. Narendra K. S. Identification and control of dynamical systems using neural networks / Narendra K. S., Parthasarathy K. // IEEE Trans. on Neural Networks. – 1990. – 1. – P. 4-26.

24. Wang H. Advanced Adaptive Control / Wang H., Liu G. P., Harris C.J., Brown M.. – Oxford: Pergamon, 1995. – 262 p.

25. Billings S. A. A new class of wavelet networks for nonlinear system identification / Billings S. A., Wei H.-L. // IEEE Trans. on Neural Networks. - 16 (4). - 2005. - P. 862-874.

26. Haykin S. Neural Networks. A Comprehensive Foundation / Haykin S. – Upper Saddle River, N.J.: Prentice Hall, Inc., 1999. – 842 p.

27. Ljung L. System Identification: Theory for the User. / Ljung, L. - Prentice Hall PTR 1999. – 520 p.

28. Kaczmarz S. Angenaeherte Ausloesung von Systemen linearer Gleichungen / Kaczmarz S. // Bull. Int. Acad. Polon. Sci. – 1937. – Let.A. – S. 355-357.

29. Kaczmarz S. Approximate solution of systems of linear equations / Kaczmarz S. // Int. J. Control. – 1993 – 53. – P. 1269-1271.

30. Бодянский Е.В. Гибридные нейросетевые архитектуры на Q-нейронах и алгоритмы их обучения / Бодянский Е.В., Винокурова Е.А., Плисс И.П. // Матеріали міжнародної наукової конференції «Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту». – Херсон. – 2. – 2009. – С. 235-239.

31. Bodyanskiy Ye. Hybrid GMDH-neural network of computational intelligence / Bodyanskiy Ye., Pliss I., Vynokurova O. // Proc. 3rd International Workshop on Inductive Modelling, Poland, Krynica – 2009. – CD. – 8 p.

32. Бодянский Е. В. Синтез дуального адаптивного экстремального регулятора / Бодянский Е. В. // Докл. АН УССР. – 1988. – Сер. А. - №5. – С. 63-65.

33. Carotenuto L. On the minimization of quadratic functions with bilinear constraints via augmented Lagrangians / Carotenuto L., Raiconi G. // J. of Optimization Theory and Applications. – 1987. – 55(1). – P. 23-36.

34. Poggio T. A Theory of Networks for Approximation and Learning / Poggio T., Girosi F. – A.I.Memo 1140, C.B.I.P. Paper №31. – MIT, 1994. – 63 p.

35. Friedman J. The Elements of Statistical Learning. Data Mining, Inference and Prediction / Friedman J., Hastie T., Tibshirani R. – Berlin: Springer, 2003. – 552 p.

36. Bishop C. M. Neural Networks for Pattern Recognition / Bishop C. M. - Oxford: Clarendon Press, 1995. - 482 p.

37. Bodyanskiy Ye. Radial-basis-fuzzy-wavelet-neural network with adaptive activation-membership function / Bodyanskiy Ye., Vynokurova O., Yegorova E. // Int. Journal on Artificial Intelligence and Machine Learning. – 2008. – V.8. – II. – P. 9-15.

38. Sobol I. Points that uniformly fill multidimensional cube / Sobol I. – Moscow: Znanie, 1985. – 32 p. (in Russian)

39. Епанечников В.А. Непараметрическая оценка многомерной плотности вероятности / Епанечников В.А. // Теория вероятности и её применение. – 1969. – 14. – №1. – С.156-161

40. Bodyanskiy Ye. Adaptive quadratic radial basis function network for time series forecasting / Bodyanskiy Ye., Chaplanov O., Kolodyazhniy V., Otto P. // Proc. East West Fuzzy Coll. 2002.– Zittau/Gorlitz: HS, 2002.– P.164-172.
41. Bodyanskiy Ye. An adaptive learning algorithm for a neuro-fuzzy network / Bodyanskiy Ye., Kolodyazhniy V., Stephan A. / Ed. by B. Reusch "Computational Intelligence. Theory and Applications." - Berlin - Heidelberg - New York: Springer, 2001. - P. 68-75.
42. Otto P. A new learning algorithm for a forecasting neuro-fuzzy network / Otto P., Bodyanskiy Ye., Kolodyazhniy V. // Integrated Computer-Aided Engineering. – 10. – N4. – Amsterdam: IOS Press, 2003. – P. 399-409.
43. Chui C. K. An Introduction to Wavelets / Chui C. K. - New York: Academic, 1992. - 264 p.
44. Daubechies I. Ten Lectures on Wavelets / Daubechies I. - Philadelphia, PA: SIAM., 1992. - 228 p.
45. Meyer Y. Wavelets: Algorithms and Applications / Meyer Y. - Philadelphia, PA: SIAM., 1993. - 133 p.
46. Zhang Q. H. Wavelet networks / Zhang Q. H., Benveniste A. // IEEE Trans. on Neural Networks. - 3 (6). - 1992. - P. 889-898.
47. Zhang Q. H. Using wavelet network in nonparametric estimation / Zhang Q. H. // IEEE Trans. on Neural Networks. - 8(2). - 1997. - P. 227-236.
48. Bodyanskiy Ye. Adaptive wavelet-neuro-fuzzy network in the forecasting and emulation tasks / Bodyanskiy Ye., Pliss I., Vynokurova O. // Int. Journal on Information Theory and Applications. - 2008. - V.15. – 1. - P. 47-55
49. Bodyanskiy Ye. Hybrid wavelet-neuro-fuzzy system using adaptive W-neurons / Bodyanskiy Ye., Pliss I., Vynokurova O. // Wissenschaftliche Berichte, FH Zittau / Goerlitz. – 2010. - 106. – N 2454-2490 – S.301-308.
50. Reyneri L. M. Unification of neural and wavelet networks and fuzzy systems / Reyneri L. M. // IEEE Trans. on Neural Networks. - 1999. - 10. - P. 801-814.



51. Бодянский Е.В. МГУА-нейронная сеть на составных R-нейронах с активационными функциями типа многомерных ядер Епанечникова / Бодянский Е.В., Плисс И.П., Тесленко Н.А. // Праці 5 міжнар. школи-семінару «Теорія прийняття рішень». – Ужгород: УжНУ, 2010. – С.27-28.

52. Бодянский Е.В. Адаптивный вейвлон как узел штучних МГУА-нейронних мереж / Бодянский Е.В., Винокурова О.А // Моделирование та керування станом еколого-економічних систем регіонів. – Київ: МННЦ ІТС. - 2008. – 4. - С. 19-29.

53. Itakura F. Maximum prediction residual principle applied to speech recognition / Itakura F. // IEEE Trans. on Acoustics, Speech, and Signal Processing. – 1975. – 23. – P. 67-72.

54. Бодянский Е.В. Составной адаптивный вейвлон и алгоритм его обучения / Бодянский Е.В., Винокурова Е.А. // Управляющие системы и машины. – 2009. – 1 (219). – С.47-53

55. Rey W.J.J. Robust Statistical Methods / Rey W.J.J. - Lecture Notes in Mathematics. - Berlin-Heidelberg-New York: Springer-Verlag. - V. 690. - 1978. – 128 p.

56. Cichocki A. Neural Networks for Optimization and Signal Processing / Cichocki A., Unbehauen R. - Stuttgart: Teubner, 1993. – 544 p.

57. Holland P.W. Robust regression using iteratively reweighted least squares / Holland P.W., Welsh R.E. // Commun. Statist., Theory and Methods. – 46. – 1977. – P.813-828.

58. Бодянский Е.В. Функционально связанная МГУА-вейвлет-нейро-фаззи система и робастный алгоритм ее обучения / Бодянский Е.В., Винокурова Е.А. // Індуктивне моделювання складних систем. Збірник наукових праць. – К.: МННЦ ІТС, вип.2. – 2010. – С. 15-24

59. Jekabsons G. A software tool for performing regression modelling using various modelling methods / Jekabsons G. Точка доступа [<http://www.cs.rtu.lv/jekabsons/>]

60. Otto P. Fuzzy based time series forecasting of electrical load / Otto P., Schunk T. // Proc. of the European Control Conference ECC'99, Karlsruhe, Germany, 1999. – P. 178-185.
61. Chiu S. Fuzzy model identification based on cluster estimation / Chiu S. // Journal of Intelligent and Fuzzy Systems. - V.2. - 3. - 1994. - P. 267-278.
62. Клигенс Н., Тельскис Л. Методы обнаружения моментов изменения свойств случайных процессов / Клигенс Н., Тельскис Л. // Автоматика и телемеханика. – 1983. – № 10. – С.5-56.
63. Гребенюк Е.А. Методы анализа нестационарных временных рядов с неявными изменениями свойств / Гребенюк Е.А. // Автоматика и телемеханика. – 2005. – №12. – С. 3-30.
64. Машталир С.В. Исследование областей достаточности и необходимости в условиях решения задач альтернативного распознавания видеоданных / Машталир С.В., Щербинин К.С. / Прикладная радиоэлектроника. – 9 (4). – 2010. – С. 580-583.
65. Kinoshenko D. Changes search methods in video data / Kinoshenko D., Mashtalir S., Stephan A., Vinarski V. // Int. J. Information Technologies and Knowledge. – 2011. – V18. – 3. – P.232-242.
66. Блаттер К. Вейвлет-анализ. Основы теории / Блаттер К. - М: «Техносфера». - 2006. – 272 с.
67. Малла С. Вэйвлеты в обработке сигналов / Малла С. - М.: Мир. 2005. – 671 с.
68. Смоленцев Н.К. Основы теории вейвлетов. Вейвлеты в MATLAB. / Смоленцев Н.К. - М.: ДМК Пресс. 2005. – 304 с.
69. Дьяконов В. Вейвлеты. От теории к практике / Дьяконов В. - М.: СОЛОН-Р. - 2002. – 440 с.
70. Gray R. Entropy and Information Theory / Gray R. - New York: Springer. 1990. – 332 p.
71. Zunino L. Wavelet Entropy of Stochastic Processes / Zunino L., Perez D. G., Garavaglia M., Rosso O.A. // Physica A - 2007. - 379. - 2. - P. 503-512.

72. Rosso O.A. Wavelet entropy: a new tool for analysis of short duration brain electrical signals / Rosso O.A., Blanco S., Yordanova J., Kolev V., Figliola A., Schürmann M., Başar E. // *Journal of Neuroscience Methods*. - 2001. - 105. - P. 65–75.

73. Кириченко Л.О. Исследование вызванных потенциалов в ЭЭГ человека с помощью дискретного вейвлет-преобразования / Кириченко Л.О., Кротких С.С. // *Радіоелектроніка, інформатика, управління. Запоріжжя*. – 2011. – 2. - С.86-93.

74. Кириченко Л.О. Использование вейвлет-энтропии для анализа информационных потоков данных / Кириченко Л.О., Кротких С.С. // *Международная конференция «Проблемы и потенциал академической мобильности: пути к международной и междисциплинарной кооперации»*. - Казахстан, Алматы, - 2010. - С. 80-82.

75. Кириченко Л.О. Применение вейвлет анализа для исследования вызванных потенциалов в ЭЭГ человека / Кириченко Л.О., Кротких С.С. //: *Матеріали 1-ї Міжнародної науково-технічної конференції “Обчислювальний інтелект (результати, проблеми, перспективи)”*. – Черкаси: Маклаут, 2011. - С.196-197.

76. Bodyanskiy Ye. Feedforward Neural Network with a Specialized Architecture for Estimation of the Temperature Influence on the Electric Load / Bodyanskiy Ye., Popov S., Rybalchenko T. // *Proc. 4th Int. IEEE Conf. Intelligent Systems*. – Varna, Bulgaria, September 6-8, 2008. – I. – P. 7-14-7-18.

77. Bodyanskiy Ye. Multilayer neuro-fuzzy network for short term electric load forecasting / Bodyanskiy Ye., Popov S., Rybalchenko T. // *Lecture Notes in Computer Science* – Berlin, Heidelberg: Springer-Verlag, 2008. – 5010. – P. 339-348.

78. Bodyanskiy Ye. Neuro-fuzzy unit for real-time signal processing / Bodyanskiy Ye., Popov S. // *Proc. IEEE East-West Design & Test Workshop (EWDTW'06)*. – Sochi, Russia, September 15-19, 2006. – P. 403-406.

79. Bodyanskiy Ye. Multilayer Network of Neuro-Fuzzy Units in Forecasting Applications / Bodyanskiy Ye., Popov S. // Research Papers of Wroclaw University of Economics. Knowledge Acquisition and Management. – 2008. – N. 25. – P. 9-14.

80. Бодянский Е.В. Прогнозирующая адаптивная нейронная сеть с динамическими нейронами-фильтрами / Бодянский Е.В., Попов С.В., Чепенко Т.Е. // Радиоэлектроника и информатика. – 2003. – N. 2. – С. 48-51.

81. Бодянский Е.В. Алгоритм обучения локально рекуррентной нейронной сети в задаче идентификации нелинейных динамических объектов / Бодянский Е.В., Попов С.В., Чепенко Т.Е. // АСУ и приборы автоматики. – 2004. – 126. – С. 17-23.

82. Гибридная нейро-фаззи сеть для краткосрочного прогнозирования взаимосвязанных процессов потребления электроэнергии / Бодянский Е.В., Попов С.В., Рыбальченко Т.В., Титов Н.Н. // Праці Інституту електродинаміки НАН України. – 2008. – 21. – С. 13-22.

83. Попов С.В. Специализированные архитектуры искусственных нейронных сетей на базе гибридных нейроподобных элементов / Попов С.В. // Збірник наукових праць Національного гірничого університету. – 2009. – №33, Т. 2. – С. 76-82.

84. Попов С.В. Эволюционная нейро-фаззи сеть на базе гибридных нейроподобных элементов / Попов С.В., Шкуро К.А. // 17 міжнародна конференція з автоматичного управління «Автоматика-2010». Тези доповідей. Том 2. – Харків: ХНУРЕ, 2010. – С. 193-194.

85. Попов С.В. Прогнозирование динамики гололедной нагрузки на воздушных линиях электропередачи с использованием искусственных нейронных сетей / Попов С.В., Шкуро К.А., Пархоменко О.В. // Материалы Международной научно-практической конференции «Информационные системы и технологии в энергетике и жилищно-коммунальной сфере» ИСТЭ-2011. – Ялта, 1-6 октября, 2011. – С. 15-17.

86. Цыпкин Я.З. Основы информационной теории идентификации / Цыпкин Я.З. – М.: Наука, 1984. – 320 с.

87. Эйкхофф П. Основы идентификации систем управления / Эйкхофф П.. – М.: Мир, 1975. – 683 с.
88. Astrom K.J. Lectures on the identification problem - the least squares method / Astrom K.J. - Report 6806. Division on Automatic Control, Lund Institute Of Technology–Lund; Sweden, 1968. – 164 p.
89. Hsia T.C. Identification: Least Squares Methods / Hsia T.C. – Lexington.Mass; Lexington Books, 1977. – 342 p.
90. Mendel J.M. Discrete Techniques of Parameter Estimation. The Equation Error Formulation / Mendel J.M.. – New York: Marc Dekker, 1973. – 246 p.
91. Алберт А. Регрессия, псевдоинверсия и рекуррентное оценивание / Алберт А.: Пер. с англ. – М.: Наука, 1977. – 495 с.
92. Позняк А.С. Обобщенный метод инструментальных переменных в задачах идентификации линейных объектов / Позняк А.С., Цыпкин Я.З. // Докл. АН СССР. – 1989. – №5. – С. 1068-1072.
93. Schweppe F.C. Recursive state estimation: unknown but bounded errors and system inputs / Schweppe F.C. // IEEE Trans. on Autom. Contr. – 1968. – 13. – №2. – P. 22-29.
94. Fogel E. On the value of information in system identification - bounded noise case / Fogel E., Huang Y.F. // Automatica. – 1982. – 18. – №2. – P. 229- 238.
95. Norton J.P. An introduction to identification / Norton J.P. – London: Academic Press Inc., 1986. – 310 p.
96. Wittenmark B. Stochastic adaptive control methods: a survey / Wittenmark B. // Int. J.Control-1975. – 21. – №5. – P. 705-730.
97. Ortega R. A note on direct adaptive control of systems with bounded disturbances / Ortega R., Lozano-Leal R. // Automatica. – 1987. – 23. – №2. – P. 253-254.
98. Деревицкий Д. П. Прикладная теория дискретных адаптивных систем управления / Деревицкий Д. П., Фрадков А. Л. – М.: Наука, 1982. – 216 с.
99. Gharieb W. Fuzzy retiming PI controller / Gharieb W., Sheiran M. A. // Proc. IFAC-IFIP -IMACS Conf. –Belfort, France, 1997. – P. 167-171.

100. Справочник по теории автоматического управления / Под ред. А.А. Красовского. – М.: Наука, 1987. – 712 с.
101. Fogel E. On the value of information in system identification - bounded noise case / Fogel E., Huang Y.F. // Automatica. –1982. –18. –№2. –P. 229- 238.
102. Lozano-Leal R. Reformulation of the parameter identification problem for system with bounded disturbances / Lozano-Leal R., Ortega R. // Automatica. – 1987. – 23. – №2. – P. 245-257.
103. Ortega R. A note on direct adaptive control of systems with bounded disturbances / Ortega R., Lozano-Leal R. // Automatica. – 1987. – 23. – №2. – P. 253-254.
104. Canudas de Wit C.C. A modified EW-RLS algorithm for systems with bounded disturbances / Canudas de Wit C.C., Carrilo J. // Automatica. – 1990. – 26. – P. 599-606.
105. Тимофеев В.А. Модифицированный рекуррентный метод наименьших квадратов с супремальными свойствами / Тимофеев В.А. // Автоматика, автоматизация, электротехнические комплексы и системы. – 2008. – №2(12). – С. 38-45.
106. Борисов В.В. Нечеткие модели и сети / В.В. Борисов, В.В. Круглов, Ф.С. Федулов – М: Горячая линия. – Телеком, 2007. – 284 с.
107. Штовба С.Д. Проектирование нечетких систем средствами MATLAB / С.Д. Штовба– М.: Горячая линия – Телеком, 2007. – 288 с.
108. Ярушкина Н.Г. Основы теории нечетких и гибридных систем: учеб. пособие / Н.Г. Ярушкина – М.: Финансы и статистика, 2004.– 320 с.
109. Кораблев Н.М. Сравнительный анализ иммунного и генетического алгоритмов в задачах оптимизации нелинейных функций / Н.М. Кораблев, И.В. Овчаренко, В.В. Токарев // Бионика интеллекта.–2007.–№2(67).–С.56-60.
110. Корабльов М.М. Адаптація моделей нечіткого виведення з використанням штучних імунних систем / М.М. Корабльов, І.В. Овчаренко // Вісник Національного університету «Львівська політехніка».–2007.–№603.– С. 73–76.

111. Кораблев Н.М. Адаптивные нечеткие модели идентификации нелинейных объектов на основе искусственных иммунных систем / Н.М. Кораблев, И.В. Сорокина // Бионика интеллекта.–2008.–№2(69).–С.125-131.
112. Designing Ensembles of Fuzzy Classification Systems: An Immune-Inspired Approach / [P.A.D. Castro, G.P. Coelho, M. F. Caetano, F. J. Von Zuben] // Springer Lecture Notes in Computer Science. – 2005. – Vol. 3627. – P. 469–482.
113. Растрингин Л. А. Адаптация сложных систем. Методы и приложения / Растрингин Л. А. – Рига: Зинатне, 1981.— 375 с.
114. Цыпкин Я.З. Адаптация и обучение в автоматизированных системах / Цыпкин Я.З. – М., Наука, 1968.
115. Levenberg K. A Method for the Solution of Certain Non-Linear Problems in Least Squares / K. Levenberg // The Quarterly of Applied Mathematics – 1944. – V.2. – P.164 – 168.
116. Marquardt D. An Algorithm for Least-Squares Estimation of Nonlinear Parameters / D. Marquardt // SIAM Journal on Applied Mathematics. – 1963. – V.11. – P.431 – 441.
117. Гилл Ф. Практическая оптимизация / Ф. Гилл, У. Мюррей, М. Райт; пер. с англ. — М.: Мир, 1985. – 465 с.
118. Руденко О.Г. Искусственные нейронные сети: учебное пособие / О.Г. Руденко, Е.В. Бодянский– Харьков: ООО «Компания СМИТ», 2005. – 408 с.
119. Takagi T. Fuzzy identification of systems and its application to modeling and control / T. Takagi, M. Sugeno // IEEE Trans. On Systems, Man, and Cybernetics. – 1985. – Vol.15. – №1. – P.116-132.
120. Ротштейн А.П. Интеллектуальные технологии идентификации: нечеткая логика, генетические алгоритмы, нейронные сети / А.П. Ротштейн – Винница: УНІВЕРСУМ-Вінниця, 1999. – 320 с.
121. Воеводин Вл.В. Методы описания и классификации вычислительных систем / Вл.В. Воеводин, А.П. Капитонова. – М.:

Издательство МГУ, 1994. –79 с.

122. Кораблев Н.М. Параметрическая адаптация в адаптивных нечетких моделях на основе искусственных иммунных систем / Н.М. Кораблев, И.В. Овчаренко // Интеллектуальный анализ інформації: VII міжнар. конф., 15-18 травня 2007 р.: збірник праць. – К., 2007. – С.169-178.

123. Jerne N. K. The immune system / N. K. Jerne // Sci. Am. –1973.– V. 229, № 1. –P. 52-61.

124. Jerne N. K. Towards a network theory of the immune system / N. K. Jerne // Ann. Immunol. (Inst. Pasteur). – 1974. – V. 125C. – P. 373–389.

125. Круглов В.В. Искусственные нейронные сети. Теория и практика / В.В Круглов., В.В. Борисов – М.: Горячая линия – Телеком, 2001. – 382 с.

126. Jang J.-S. R. ANFIS: Adaptive-Network-Based Fuzzy Inference System / J.-S. R. Jang // IEEE Trans. Systems & Cybernetics. – 1993. – Vol. 23. – P. 665 – 685.

127. Кораблев Н.М. Иммунный алгоритм обучения адаптивных нечетких нейронных сетей / Н.М. Кораблев, И.В. Сорокина, А.И. Русецкий // Системи управління, навігації та зв'язку.–2008.– Вип.4(8).– С. 62-67.

128. Кораблев Н.М. Адаптация структуры и параметров нейросетевого нечеткого регулятора с использованием искусственных иммунных систем / Н.М. Кораблев, А.Э. Макогон, И.В. Сорокина // Системи управління, навігації та зв'язку.–2011.– Вип.1(17).– С. 89-93.

129. Korablev N. Immune Approach for Neuro-Fuzzy Systems Learning Using Multiantibody Model / Korablev N., Sorokina I. // ICARIS 2011, Springer Lecture Notes in Computer Science; P. Lit, G. Nicosia, and T. Stibor (Eds.) – 2011. – Vol.6825. – P. 395-405.

130. Бодянский Е.В. Искусственные нейронные сети: архитектуры, обучение, применения / Е.В. Бодянский, О.Г. Руденко – Харьков: ТЕЛЕТЕХ, 2004. – 372с.



131. Штовба С.Д. Обеспечение точности и прозрачности нечеткой модели Мамдани при обучении по экспериментальным данным / С.Д. Штовба // Проблемы управления и информатики. – 2007. – №4. – С. 102–114.

132. Takagi T. NN-driven fuzzy reasoning / T. Takagi, I. Hayashi // International Journal of Approximate reasoning. – 1991. – №5(3). – P. 191-212.

133. Takagi T. Fuzzy identification of systems and its application to modeling and control / T. Takagi, M. Sugeno // IEEE Trans. On Systems, Man, and Cybernetics. – 1985. – Vol.15. – №1. – P.116-132.

134. Сорокина И.В. Оценка сходимости иммунного алгоритма адаптации нечеткой нейронной сети / И.В. Сорокина, Е.В. Токарева // Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: друга наук.-техн.конференція, 15-16 грудня 2011 р.: тези доп. – Харків: ДП «ХНДІ ТМ»; К.: ДП «ЦНДІ НіУ», 2011. – С.40.

135. Кораблёв Н.М. Анализ сходимости иммунных алгоритмов / Кораблёв Н.М., Макогон А.Э., Фомичёв А.А. // Системы обработки информации: сборник научных работ. – 2011. – Вып. 2 (92). – С. 29-33.

136. Кораблёв Н.М. Определение вектора приоритетов признаков на основе процедуры неполных парных сравнений / Кораблёв Н.М. // Системы обработки информации. – 2011. – Вып. 4 (94). – С. 133-137.

137. Кораблёв Н.М. Обработка результатов тестирования обучающихся с использованием искусственных иммунных систем / Кораблёв Н.М., Фомичёв А.А. // Бионика интеллекта. – 2011. – № 1 (75). – С. 84-87.

138. Кораблев Н.М. Повышение качества дифференциальной диагностики аллергодерматозов с применением искусственных иммунных систем / Кораблев Н.М., Сорокина И.В., Солошенко Э.Н. // Сб. научн. Трудов VI-йой Междунар. Научно-практической конф. «Наука и социальные проблемы общества: Информатизация и информационные технологии». – Харьков, ХНУРЭ. – 2011. – С. 24-25.

139. Кучеренко, Є.І. Теоретичні основи та технології оцінки технічного стану просторово розподілених об'єктів: монографія, наукове видання /

Є.І. Кучеренко, Д.Є. Краснокутський, І.С. Глушенкова. – Х.: ХНАМГ, ХНУРЕ, 2011. – 167с.

140. Zadeh, L.A. Fuzzy Sets / L.A. Zadeh // Information and Control. – 1965. – № 8 (3). – P. 338 – 353.

141. Tsoukalas, L.H. Fuzzy and Neural Approaches in Engineering / L.H. Tsoukalas, R.E. Uhrig. – New York: John Wiley&Sons.Inc, 1997. – 587 p.

142. Кучеренко, Е.И. Модели процессов оценивания состояния сложных пространственно распределенных объектов / Е.И. Кучеренко, И.С. Глушенкова // Системи обробки інформації. – 2011. – Вип. 3 (92). – С. 93 – 101.

143. Кучеренко, Е.И. Информационные технологии оценивания состояния сложных пространственно распределенных объектов / Е.И. Кучеренко, И.С. Глушенкова // Системы управления, навигации и связи. – 2011. – Вып. 1 (27). – С. 144 – 150.

144. Кучеренко, Е.И. Проблемы анализа нечетких интеллектуальных систем на основе знание ориентированных технологий / Е.И. Кучеренко, И.С. Глушенкова // Материалы Шестой Международной научно-практической конференции «Наука и социальные проблемы общества: информатизация и информационные технологии». – Харьков: ХНУРЭ, 2011. – С. 106 – 107.

145. Глушенкова, І.С. Модифікація геодезичної опорної мережі регіону засобами ГІС / І.С. Глушенкова, О.В. Зарицький // Інформаційні технології: Наука, техніка, технологія, освіта, здоров'я: зб. тез допов. ХІХ міжнар. науково-практичн. конф., Ч. ІV (01 – 03 червня 2011 р., Харків). – Х.: НТУ «ХП», – 2011. – С. 16.

146. Глушенкова, І.С. Информационная технология оценивания состояния объекта на основе формальной модели / И.С. Глушенкова // Наукове забезпечення службово-бойової діяльності внутрішніх військ МВС України: зб. тез допов. ІІІ науково-практичн. конф. – Х.: Академія внутрішніх військ МВС України. – 2011 – С. 83-84.

147. Кучеренко, Є.І. Еволюційні інтервальні нечіткі моделі та методи оцінювання ієрархічних розподілених процесів / Є.І. Кучеренко,

І.С. Творошенко, Т.Д. Кучеренко // Искусственный интеллект. Интеллектуальные системы. ИИ – 2011: материалы Международной научно-технической конференции (Кацивели, АР Крым, 19-23 сент. 2011 г.). – Д.: ПШ «Наука і освіта», 2011. – Т. 1. – С. 190-193.

148. Кучеренко, Е.И. Оперативне оцінювання простору станів складних розподілених об'єктів з використанням нечіткої інтервальної логіки / Е.И.Кучеренко, І.С. Творошенко // Искусственный интеллект. – 2011. – № 3. – С. 382-387.

149. Кучеренко, Е.И. Развитие методов на основе многозначной интервальной логики в задачах настройки функций принадлежности / Е.И. Кучеренко, А.В. Корниловский, И.С. Глушенкова // Біоніка інтелекту. – 2011. – Вип. 1 (75). – С. 75 – 78.

150. Глушенкова, И.С. Прикладные аспекты реализации методов и информационных технологий оценивания сложных объектов / И.С. Глушенкова, Е.И. Кучеренко // Восточно-Европейский журнал передовых технологий. – 2011. – № 3/9 (51). – С. 35-38.

151. Кучеренко, Е.И. Метод декомпозиции безопасных сетей Петри для анализа производственных систем / Е.И. Кучеренко, А.Д. Дрюк, С.А. Нестерцова // Матеріали 1 Всеукраїнської та студентської конференції за напрямком «Проблеми розвитку та впровадження систем управління, стандартизації, сертифікації, метрології в регіонах України» (травень 2011). – Донецьк: ДНТУ, 2011. – С. 181 – 183.

152. Творошенко, І.С. Процессы принятия решений в практических реализациях оценивания состояний сложного объекта / І.С. Творошенко // Матеріали 5 Всеукраїнської науково-практичної конференції «Сучасні тенденції розвитку інформаційних технологій в науці, освіті та економіці» (квітень 2011). – Луганськ: Phoenix, 2011. – Т. 1. – С. 175 – 177.

153. Творошенко, І.С. Моделирование нечетких процессов в системах управления производством / І.С. Творошенко // Матеріали 15 Юбилейного

Международного молодежного форума «Радиоэлектроника и молодежь в 21 веке». – Харьков: ХНУРЭ, 2011. – Т. 7. – С. 78 – 79.

154. Корнеев, Е.О. Робототехнические комплексы на основе нечеткой интервальной логики / Е.О. Корнеев // Материалы 15 Юбилейного Международного молодежного форума «Радиоэлектроника и молодежь в 21 веке». – Харьков: ХНУРЭ, 2011. – Т. 9. – С. 22 – 23.

155. Дрюк, А.Д. Моделирование дискретных динамических процессов с использованием сетевых моделей / А.Д. Дрюк, Е.И. Кучеренко // 10 Всеукраинская научно-техническая конференция «Математическое моделирование и информационные технологии». – Одесса: ОДАХ, 2011. – С. 71 – 72.

156. Дейт, К. Введение в системы баз данных: пер. с англ.– М. : Издательский дом «Вильямс», 2001. – 1072 с.

157. Касаткіна, Н.В. Об одном подходе к классификации моделей данных информационных систем // Проблеми системного підходу в економіці. Збірник наукових праць: випуск 37 / Н.В. Касаткіна, В.А. Филатов, Е.Б. Чапланова . – К.: НАУ, 2011. – С. 3–9.

158. Филатов, В.А. Об одном подходе к интеллектуальному анализу реляционных данных / В.А. Филатов, Н.С. Дзюба, З.Л. Костина // Вестник Херсонского национального техн. ун-та. – Херсон, 2011. – вып.2 (41). – С. 212–216.

159. Филатов, В.А. Об одном подходе к интеллектуальному анализу реляционных данных / В.А. Филатов, Е.Б. Чапланова, С.С. Танянский, А.И. Сизов // УСиМ: Управляющие системы и машины, № 3, 2011. – С. 73-78.