

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

На правах рукопису

КОПАЛІАНІ Дар'я Сергіївна

УДК 004.032.26

**ЕВОЛЮЦІЙНІ НЕЙРО-ФАЗЗИ МЕРЕЖІ З КАСКАДНОЮ
СТРУКТУРОЮ ДЛЯ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАННИХ**

05.13.23 — системи та засоби штучного інтелекту

Дисертація на здобуття наукового ступеня
кандидата технічних наук

Науковий керівник
Бодянський Євгеній Володимирович,
доктор технічних наук, професор

Харків — 2015

ЗМІСТ

Розділ 1. Гібридна каскадна нейро-фаззі мережа з оптимізацією пулу нейронів	3
1.1. Архітектура оптимізованої каскадної нейронної мережі	5
1.2. Навчання елементарних персептронів Розенблатта у каскадній оптимізованій системі	7
1.3. Навчання нео-фаззі нейронів у оптимізованій каскадній нейронній мережі	10
1.4. Розширенні нео-фаззі нейрони в якості елементів гібридної каскадної мережі, що еволюціонує	17
1.5. Оптимізація пулу нео-фаззі нейронів	20
Висновки до розділу 1	23
Розділ 2. Багатовимірна каскадна нео-фаззі система, що еволюціонує	24
2.1. Багатовимірна каскадна система, що еволюціонує, побудована на нео-фаззі нейронах	25
2.1.1. Оптимізація пулу нео-фаззі нейронів багатовимірної каскадної системи, що еволюціонує	30
2.2. Багатовимірна каскадна система, що еволюціонує, побудована на багатовимірних нео-фаззі нейронах	33
2.2.1. Багатовимірний нео-фаззі нейрон	33
2.2.2. Метод визначення поточно оптимального вихідного сигналу пулу багатовимірних нео-фаззі нейронів каскадної системи, що еволюціонує	35
Список використаних джерел	38

РОЗДІЛ 1

ГІБРИДНА КАСКАДНА НЕЙРО-ФАЗЗИ МЕРЕЖА З ОПТИМІЗАЦІЄЮ ПУЛУ НЕЙРОНІВ

Зазвичай під «навчанням» розуміють процес коригування синаптичних вагових коефіцієнтів за допомогою певної процедури оптимізації, що ґрунтується на пошуці екстремуму заданого критерію навчання. Якість процесу навчання може бути поліпшена шляхом коригування топології мережі поспіль з синаптичними вагами [11, 13]. Ця ідея лежить в основі систем обчислювального інтелекту, що еволюціонують [17, 23].

Мабуть, найбільш відомою реалізацією цього підходу є каскадно-кореляційні нейронні мережі [12, 25, 27], привабливі високою ефективністю та простотою налаштування як синаптичних вагових коефіцієнтів, так і топології мережі. Така мережа напочатку містить лише один пул (ансамбль) нейронів, які навчаються незалежно один від іншого (перший каскад). Кожен нейрон у пулі може відмінні функції активації та метод навчання. Доки навчання триває, нейрони у пулі не взаємодіють один з одним. Після того як процес налаштування вагових коефіцієнтів завершився для всіх нейронів пулу першого каскаду, кращий нейрон відповідно до обраного критерію навчання формує перший каскад і коефіцієнти його синаптичних ваг більше не коригуються. Далі формується другий каскад зазвичай з нейронів, подібних до нейронів першого каскаду. Різниця лише в тому, що нейрони, які навчаються в пулі другого каскаду мають додатковий вхід (і, отже, додатковий синаптичний ваговий коефіцієнт) - вихід першого каскаду. Подібно до першого каскаду, у другому каскаді залишиться лише один найбільш продуктивний нейрон і його синаптичні вагові коефіцієнти зафіксуються. Аналогічним чином нейрони третього каскаду матимуть два додаткових входи, а саме виходи першого та другого каскадів. Еволюційна мережа продовжуватиме розширяти свою ар-

хітектуру новими каскадами, доки вона не досягне бажаної якості вирішення завдання для заданого набору даних.

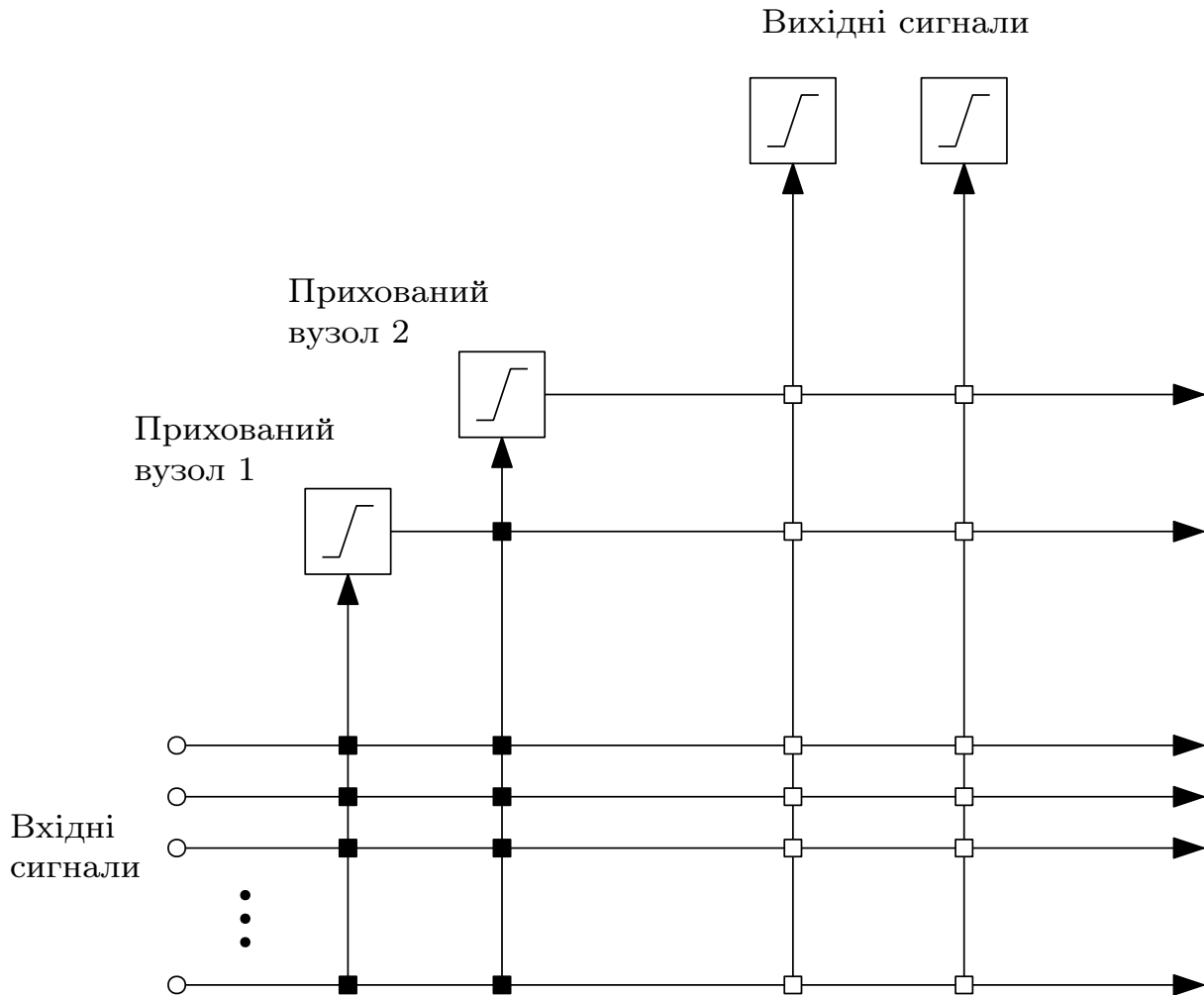


Рис. 1.1. Архітектура каскадної системи (за Фалманом та Ліб'єром) після додавання двох прихованих вузлів. Ухідні сигнали, що надходять до вертикальних ліній, сумуються; вагові коефіцієнти, позначені □, – зафіксовані, позначені ■, – налаштовуються

Автори найпопулярнішої каскадної нейронної мережі, що еволюціонує, CasCorLA (схему наведено на рис. 1.1), Фалман та Ліб'єр, використовували елементарні перцептрони Розенблатта з традиційними сигмоїдальними функціями активації і коригували синаптичні вагові коефіцієнти за допомогою QuickProp-алгоритму [12], що є модифікацією δ -правила. Оскільки вихідний сигнал таких нейронів нелінійно залежить від синапсових ваг, швидкість навчання не може бути суттєво збільшена для таких нейронів.

Для уникнення багатоепохового навчання [3, 4, 6–8, 14, 19, 35] доцільно в якості вузлів системи використовувати такі типи нейронів, що їх виходи лінійно залежать від синаптичних ваг, що дозволить використовувати оптимальні за швидкістю методи навчання та обробляти дані в онлайн режимі.

Проте варто зазначити, що у випадку послідовного навчання системи неможливо визначити найкращий нейрон у пулі, адже при оброблянні нестационарних об'єктів певний нейрон може бути кращим для однієї частини тренувальної вибірки, проте поступатися у точності іншому нейрону на іншій частині вибірки. Отже доцільно зберегти усі нейрони пулу та використовувати певну оптимізуючу процедуру (відповідну обраному критерію якості) задля визначення нейрона-переможця на кожному кроці оброблення даних.

1.1. Архітектура оптимізованої каскадної нейронної мережі

Архітектура пропонованої гібридної системи з оптимізованим пулом нейронів у кожному каскаді наведена на рис. 1.2.

На вхід такої системи (так званий «рецептивний» шар) подається векторний сигнал

$$x(k) = (x_1(k), x_2(k), \dots, x_n(k))^T, \quad (1.1)$$

де $k = 1, 2, \dots$, – кількість обраців у таблиці «об'єкт - властивість» або поточний дискретний час.

Ці сигнали подаються на входи кожного нейрона в мережі $N_j^{[m]}$ ($j = 1, 2, \dots, q$ – кількість нейронів у тренувальному пулі, $m = 1, 2, \dots$ – номер каскаду) з вихідним сигналом $\hat{y}_j^{[m]}(k)$. Далі вихідні сигнали кожного каскаду $\hat{y}_j^{[m]}(k)$ надходять до «узагальнюючого» вузлу $GN^{[m]}$, який генерує поточно-оптимальний вихідний сигнал відповідного каскаду $\hat{y}^{*[m]}$. Слід зауважити, що вхідними сигналами першого каскаду є вектор $x(k)$ (що може містити опціональне порогове значення $x_0(k) \equiv 1$), другий каскад має додатковий вхід для

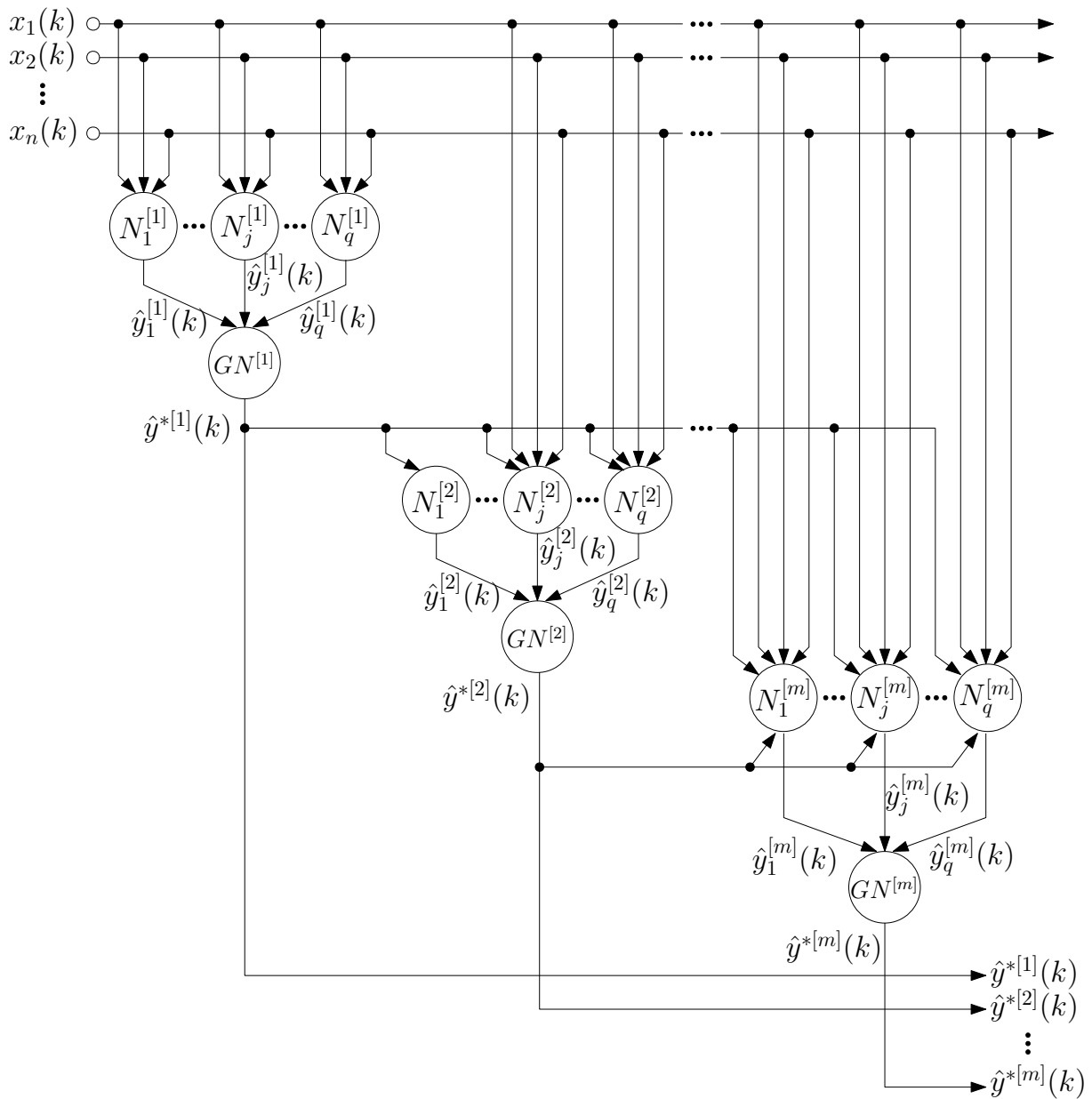


Рис. 1.2. Архітектура гібридної системи з оптимізованим пулом нейронів

сгенерованого першим каскадом вихідного сигналу $\hat{y}^{*[1]}(k)$, нейрони третього каскаду оброблятимуть два додаткових сигнали $\hat{y}^{*[1]}(k)$, $\hat{y}^{*[2]}(k)$, нейрони m -ого каскаду матимуть $(m - 1)$ додаткових вхідних сигналів: $\hat{y}^{*[1]}(k)$, $\hat{y}^{*[2]}(k)$, ..., $\hat{y}^{*[m-1]}(k)$. Під час тренування системи нові каскади додаються доки не буде досягнута бажана точність.

1.2. Навчання елементарних персептронів Розенблатта у каскадній оптимізованій системі

Наразі вважатимемо j -й вузол m -ого каскаду елементарним персепторном Розенблату з активаційною функцією

$$0 > \sigma_j^m \left(\gamma_j^{[m]} u_j^{[m]} \right) = \frac{1}{1 + e^{-\gamma_j^{[m]} u_j^{[m]}}} < 1 \quad (1.2)$$

де $u_j^{[m]}$ – внутрішній активаційний сингнал j -ого нейрону m -ого каскаду, $\gamma_j^{[m]}$ – параметр посилення.

У такому випадку вихідні сигнали нейронів тренувального пулу першого каскаду матимуть вигляд

$$\hat{y}_j^{[1]} = \sigma_j^{[1]} \left(\gamma_j^{[1]} \sum_{i=0}^n w_{ji}^{[1]} x_i \right) = \sigma_j^{[1]} \left(\gamma_j^{[1]} w_j^{[1]T} x \right), \quad (1.3)$$

де $w_{ji}^{[1]}$ – i -й ваговий коефіцієнт j -ого нейрону першого каскаду.

Вихідні сигнали другого каскаду дорівнюватимуть

$$\hat{y}_j^{[2]} = \sigma_j^{[2]} \left(\gamma_j^{[2]} \left(\sum_{i=0}^n w_{ji}^{[2]} x_i + w_{j,n+1}^{[2]} \hat{y}^{*[1]} \right) \right), \quad (1.4)$$

вихідні сигнали m -ого каскаду матимуть вигляд

$$\begin{aligned} \hat{y}_j^{[m]} &= \sigma_j^{[m]} \left(\gamma_j^{[m]} \left(\sum_{i=0}^n w_{ji}^{[m]} x_i + w_{j,n+1}^{[m]} \hat{y}^{*[1]} \right. \right. \\ &\quad \left. \left. + w_{j,n+2}^{[m]} \hat{y}^{*[2]} + \dots + w_{j,n+m-1}^{[m]} \hat{y}^{*[m-1]} \right) \right) \\ &= \sigma_j^{[m]} \left(\gamma_j^{[m]} \sum_{i=0}^{n+m-1} w_{ji}^{[m]} x_j^{[m]} \right) = \sigma_j^{[m]} \left(w_j^{[m]T} x^{[m]} \right), \end{aligned} \quad (1.5)$$

де $x^{[m]} = (x^T, \hat{y}^{*[1]}, \hat{y}^{*[m-1]})^T$.

Таким чином, нейронна мережа з персептронами Розенблатта у якості вузлів, що містить m каскадів, залежить від $\left(m(n+2) + \sum_{p=1}^{m-1} p\right)$ параметрів, у тому числі від параметрів посилення $\gamma_j^{[p]}$, $p = 1, 2, \dots, m$.

У якості критерію навчання можна використовувати загальноприйнятну квадратичну функцію

$$\begin{aligned} E_j^{[m]} &= \frac{1}{2} \left(e_j^{[m]}(k) \right)^2 \\ &= \frac{1}{2} \left(y(k) - \hat{y}_j^{[m]}(k) \right)^2 \\ &= \frac{1}{2} \left(y(k) - \sigma_j^{[m]} \left(\gamma_j^{[m]} w_j^{[m]T} x^{[m]}(k) \right) \right)^2, \end{aligned} \quad (1.6)$$

де $y(k)$ – шуканий сигнал.

Градiєнтну оптимізацію критерію (1.6) відносно $w_j^{[m]}$ можна записати у вигляді

$$\begin{aligned} w_j^{[m]}(k+1) &= w_j^{[m]} + \eta_j^{[m]}(k+1) e_j^{[m]}(k+1) \gamma_j^{[m]} \hat{y}_j^{[m]}(k+1) \\ &\quad \times \left(1 - \hat{y}_j^{[m]}(k+1) \right) x^{[m]}(k+1) \\ &= w_j^{[m]}(k) + \eta_j^{[m]}(k+1) e_j^{[m]}(k+1) \gamma_j^{[m]} J_j^{[m]}(k+1) \end{aligned} \quad (1.7)$$

де $\eta_j^{[m](k+1)}$ – параметр швидкості навчання.

Мінімізувати критерій (1.6) відносно $\gamma_j^{[m]}$ можна за допомогою алгоритму Крушке-Мовелана [20]

$$\begin{aligned} \gamma_j^{[m]}(k+1) &= \gamma_j^{[m]}(k) + \eta_j^{[m]}(k+1) e_j^{[m]}(k+1) \hat{y}_j^{[m]}(k+1) \\ &\quad \times \left(1 - \hat{y}_j^{[m]}(k+1) \right) u_j^{[m]}(k+1). \end{aligned} \quad (1.8)$$

Поеднуючи (1.7) та (1.6) отримаємо алгоритм навчання для j -ого нейрону m -ого каскаду

$$\begin{aligned} \frac{w_j^{[m]}(k+1)}{\gamma_j^{[m]}(k+1)} &= \frac{w_j^{[m]}(k)}{\gamma_j^{[m]}(k)} + \eta_j^{[m]}(k+1)e_j^{[m]}(k+1)\hat{y}_j^{[m]}(k+1) \\ &\times \left(1 - \hat{y}_j^{[m]}(k+1)\right) \left(\frac{\gamma_j^{[m]}x^{[m]}(k+1)}{u_j^{[m]}(k+1)}\right), \end{aligned} \quad (1.9)$$

Або, вводячи нові змінні, у більш компактній формі

$$\begin{aligned} \tilde{w}_j^{[m]}(k+1) &= \tilde{w}_j^{[m]}(k) + \eta_j^{[m]}(k+1)e_j^{[m]}(k+1)\hat{y}_j^{[m]}(k+1)\tilde{x}^{[m]}(k+1) \\ &= \tilde{w}_j^{[m]}(k) + \eta_j^{[m]}(k+1)e_j^{[m]}(k+1)\tilde{J}_j^{[m]}(k+1). \end{aligned} \quad (1.10)$$

Використовуючи регулюючий параметр (momentum term) [10, 28, 29] можна удосконалити процес корегування синаптичних вагових коефіцієнтів під час навчання. Тоді, замість критерію (1.6) слід використовувати функцію

$$\begin{aligned} E_j^{[m]}(k) &= \frac{\eta}{2} \left(e_j^{[m]}(k)\right)^2 \\ &+ \frac{1 - \eta}{2} \left\| \tilde{w}_j^{[m]}(k) - \tilde{w}_j^{[m]}(k-1) \right\|^2, \quad 0 < \eta \leq 1. \end{aligned} \quad (1.11)$$

Тоді алгоритм навчання приймає вигляд

$$\begin{aligned} \tilde{w}_j^{[m]}(k+1) &= \tilde{w}_j^{[m]}(k) \\ &+ \eta_j^{[m]}(k+1) \left(\eta e_j^{[m]}(k+1) \tilde{J}_j^{[m]}(k+1) \right. \\ &\left. + (1 - \eta) \left(\tilde{w}_j^{[m]}(k) - \tilde{w}_j^{[m]}(k+1) \right) \right), \end{aligned} \quad (1.12)$$

що є модифікацією процедури Сільва-Альмеїда [28].

Доцільно вдосконалити алгоритм використовуючи підхід, запропонований у [5], тоді алгоритм (1.12) набуває слідкуючих та фільтруючих властивостей. Так, кінцева модифікація алгоритму приймає вигляд

$$\begin{cases} \tilde{w}_j^{[m]}(k+1) = \tilde{w}_j^{[m]}(k) + \frac{\eta e_j^{[m]}(k+1) \tilde{J}_j^{[m]}(k+1)}{r_j^{[m]}(k+1)} \\ \quad + \frac{(1-\eta)(\tilde{w}_j^{[m]}(k) - \tilde{w}_j^{[m]}(k-1))}{r_j^{[m]}(k+1)}, \\ r_j^{[m]}(k+1) = r_j^{[m]}(k) + \left\| \tilde{J}_j^{[m]}(k+1) \right\|^2 - \left\| \tilde{J}_j^{[m]}(k-s) \right\|^2, \end{cases} \quad (1.13)$$

де s – розмір «плаваючого» вікна.

Цікаво, що при $s = 1$ та $\eta = 1$ отримуємо нелінійну версію загальновідомого алгоритму Качмажа-Уїдроу-Хоффа [16, 33]:

$$\tilde{w}_j^{[m]}(k+1) = \tilde{w}_j^{[m]}(k) + \frac{e_j^{[m]}(k+1) \tilde{J}_j^{[m]}(k+1)}{\left\| \tilde{J}_j^{[m]}(k+1) \right\|^2}, \quad (1.14)$$

який широко використовується для навчання штучних нейронних мереж і відомий високою швидкістю збіжності.

1.3. Навчання нео-фаззі нейронів у оптимізованій каскадній нейронній мережі

Низька швидкість навчання персептронів Розенблатта у поєднанні з труднощами інтерпретації результатів (властиві всім ІНС в цілому) спонукає нас шукати альтернативні підходи до синтезу еволюційних нейронних мереж. Як зазначається у [15], нейро-фаззі системи відомі високою інтерпритуємістю і прозорістю, а також добрими апроксимаційними властивостями, та є основою гібридних систем штучного інтелекту. У [6, 19] розглядаються гібридні каскадні системи штучного інтелекту побудовані на нео-фаззі нейронах [24, 34], що дозволяє їм суттєво підвищити швидкість корегування синаптичних вагових коефіцієнтів. Нео-фаззі нейрон (NFN), що його архітектуру наведено на рис. 1.3, – це нелінійна система, що реалізує висновування

$$\hat{y} = \sum_{i=1}^n f_i(x_i) \quad (1.15)$$

де x_i – i -й вхідний сигнал ($i = 1, 2, \dots, n$),

\hat{y} – вихідний сигнал нео-фаззі нейрону.

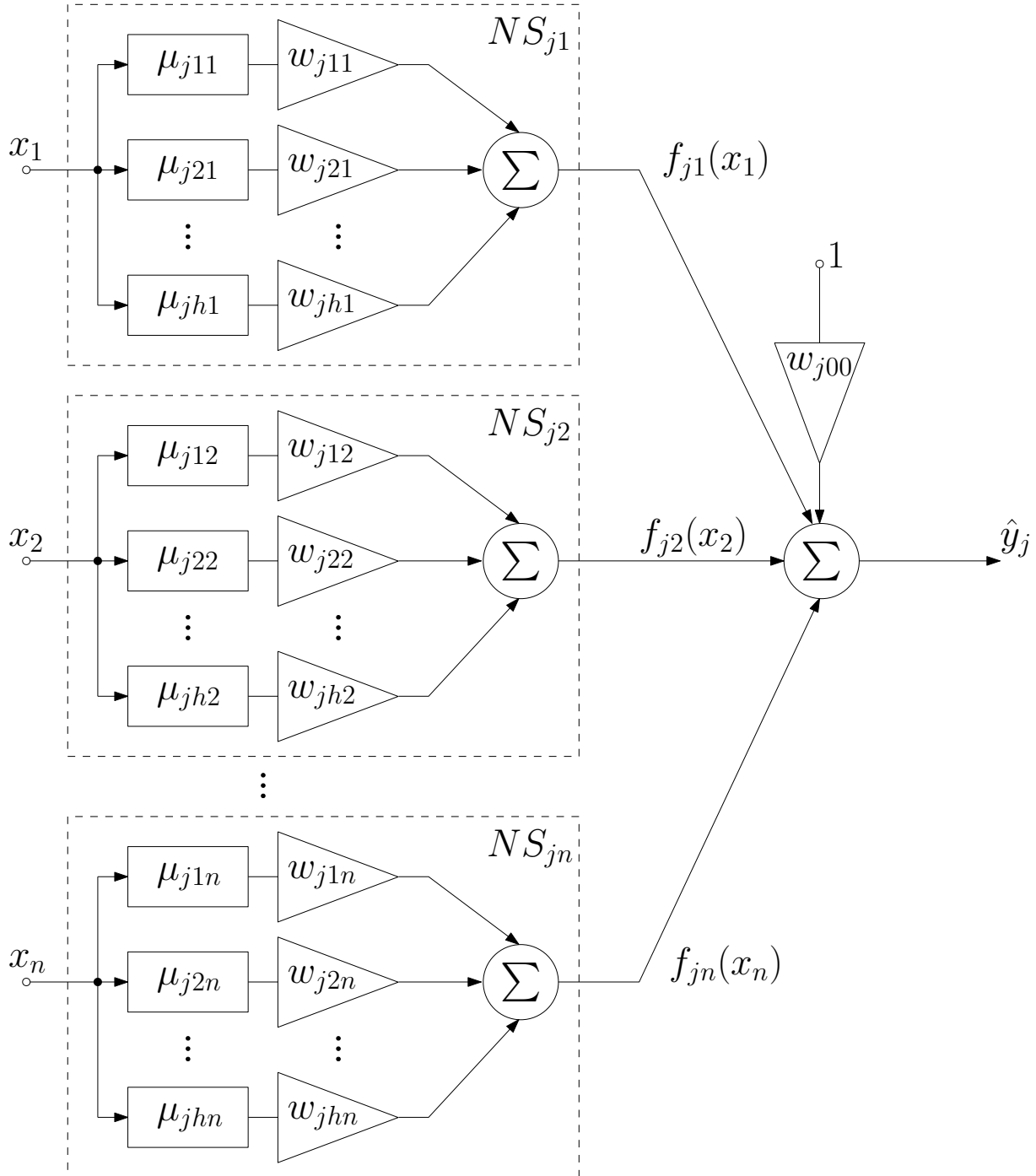


Рис. 1.3. Архітектура нео-фаззі нейрону

Структурними елементами нео-фаззі нейрона є нелінійні синапси NS_i , які

трансформують вхідні сигнали в такий спосіб:

$$f_i(x_i) = \sum_{l=1}^h w_{li} \mu_{li}(x_i), \quad (1.16)$$

де w_{li} – l -й ваговий коефіцієнт i -ого нелінійного синапсу,

$l = 1, 2, \dots, h$ – кількість синаптичних ваг, а отже і функцій належності $\mu_{li}(x_i)$ у синапсі.

Таким чином, нелінійний синапс NS_i реалізує нечітке висновування

$$\text{IF } x_i \text{ IS } X_{li} \text{ THEN THE OUTPUT IS } w_{li}, \quad (1.17)$$

де X_{li} – нечітка множина з функцією належності μ_{li} ,

w_{li} – сінглтон (синаптичний ваговий коефіцієнт у консиквенті).

Тобто нелінійний синапс фактично є системою висновування Такагі-Сугено нульового порядку [15].

Запишемо вихідні сигнали для нейронів першого каскаду у наступному вигляді:

$$\begin{cases} \hat{y}_j^{[1]}(k) = \sum_{i=1}^n f_{ji}^{[1]}(x(k)) = \sum_{i=1}^n \sum_{l=1}^h w_{jli}^{[1]} \mu_{jli}^{[1]}(x_i(k)), \\ \text{IF } x_i \text{ IS } X_{li} \text{ THEN THE OUTPUT IS } w_{li} \end{cases} \quad (1.18)$$

J -й нео-фаззі нейрон другого каскаду зображено на рис. 1.4 згідно топології нейронної мережі, зображеної на рис. 1.1)

Автори нео-фаззі нейрону [24, 34] в якості функцій належності використовували традиційні трикутні структури, які задовольняють умови розбиття Руспіні:

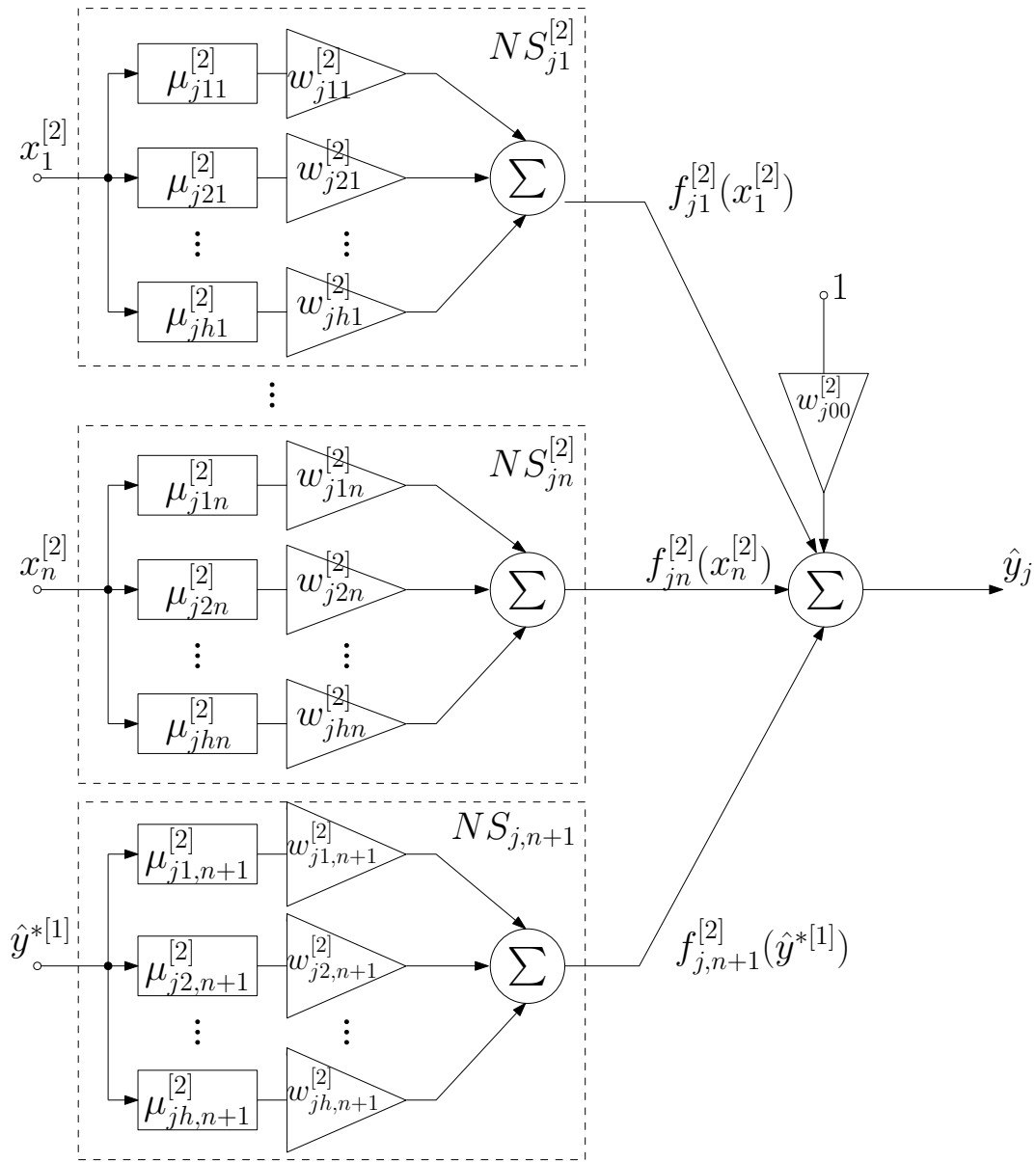


Рис. 1.4. Нео-фаззі нейрон другого каскаду запропонованої каскадної системи

$$\mu_{jli}^{[1]}(x_i) = \begin{cases} \frac{x_i - c_{j,l-1,i}^{[1]}}{c_{jli}^{[1]} - c_{j,l-1,i}^{[1]}} & \text{if } x_i \in [c_{j,l-1,i}^{[1]}, c_{jli}^{[1]}], \\ \frac{c_{j,l+1,i}^{[1]} - x_i}{c_{j,l+1,i}^{[1]} - c_{jli}^{[1]}} & \text{if } x_i \in [c_{jli}^{[1]}, c_{j,l+1,i}^{[1]}], \\ 0 & \text{otherwise,} \end{cases} \quad (1.19)$$

де $c_{jli}^{[1]}$ – довільно обрані центри параметрів функцій належності на інтервалі $[0, 1]$, зазвичай рівномірно розподілені.

Такий вибір функцій належності гарантує, що вхідний сигнал x_i активує

лише два сусідні функції, а сума їх значень завжди дорівнюватиме 1:

$$\mu_{jli}^{[1]}(x_i) + \mu_{j,l+1,i}^{[1]}(x_i) = 1, \quad (1.20)$$

$$f_{jl}^{[1]}(x_i) = w_{jli}^{[1]} \mu_{jli}^{[1]}(x_i) + w_{j,l+1,i}^{[1]} \mu_{j,l+1,i}^{[1]}(x_i). \quad (1.21)$$

Аппроксимуючі властивості системи можна поліпшити використовуючи кубічні сплайни [6] замість трикутних функцій належності:

$$\mu_{jli}^{[1]}(x_i) = \begin{cases} \frac{1}{4} \left(2 + 3 \frac{2x_i - c_{jli}^{[1]} - c_{j,l-1,i}^{[1]}}{c_{jli}^{[1]} - c_{j,l-1,i}^{[1]}} - \left(\frac{2x_i - c_{jli}^{[1]} - c_{j,l-1,i}^{[1]}}{c_{jli}^{[1]} - c_{j,l-1,i}^{[1]}} \right)^3 \right), \\ \text{if } x \in [c_{j,l-1,i}^{[1]}, c_{jli}^{[1]}], \\ \frac{1}{4} \left(2 - 3 \frac{2x_i - c_{j,l+1,i}^{[1]} - c_{jli}^{[1]}}{c_{j,l+1,i}^{[1]} - c_{jli}^{[1]}} + \left(\frac{2x_i - c_{j,l+1,i}^{[1]} - c_{jli}^{[1]}}{c_{j,l+1,i}^{[1]} - c_{jli}^{[1]}} \right)^3 \right), \\ \text{if } x \in [c_{jli}^{[1]}, c_{j,l+1,i}^{[1]}], \\ 0, \text{ otherwise,} \end{cases} \quad (1.22)$$

або B -сплайни [19]:

$$\mu_{jli}^{g[1]} = \begin{cases} 1, \text{ if } x_i \in [c_{jli}^{[1]}, c_{j,l+1,i}^{[1]}], \\ 0, \text{ otherwise} \end{cases} \text{ for } g = 1, \quad (1.23)$$

$$\begin{cases} \frac{x_i - c_{jli}^{[1]}}{c_{j,l+g-1,i}^{[1]} - c_{jli}^{[1]}} \mu_{jli}^{g-1,[1]}(x_i) + \frac{c_{j,l+g,i}^{[1]} - x_i}{c_{j,l+g,i}^{[1]} - c_{j,l+g-1,i}^{[1]}} \mu_{j,l+1,i}^{g-1,[1]}(x_i), \\ \text{for } g > 1, \end{cases}$$

де $\mu_{jli}^{g[1]}(x_i)$ – l -й сплайн g -ого порядку.

Нескладно помітити, що при $g = 2$ отримуємо трикутні функції належності (1.19).

B -сплайни, як і трикутні функції належності, забезпечують розбиття Руспіні, але в загальному випадку вони можуть активувати довільне число функцій належності, за межами інтервалу $[0, 1]$, що може стати у нагоді для подальших каскадів.

Також у якості функцій належності нелінійних синапсів можна використовувати інші структури, такі як поліноміальні гармонійні функції, вейвлети, ортогональні функції, тощо. Проте не можна сказати наперед, які з функцій забезпечать кращі результати, тому ідея використання не одного нейрона, а пулу нейронів з різними функціями належності та активації виглядає доречною та перспективною.

За аналогією до (1.18) визначаймо вихідні сигнали інших каскадів. Так, для другого каскаду можемо записати вихідні сигнали у формі

$$\hat{y}_j^{[2]} = \sum_{i=1}^n \sum_{l=1}^h w_{jli}^{[2]} \mu_{jli}^{[2]}(x_i) + \sum_{l=1}^h w_{j,l,n+1}^{[2]} \mu_{j,l,n+1}^{[2]}(\hat{y}^{*[1]}), \quad (1.24)$$

вихідні сигнали для нейронів m -ого каскаду

$$\hat{y}_j^{[m]} = \sum_{i=1}^n \sum_{l=1}^h w_{jli}^{[m]} \mu_{jli}^{[m]}(x_i) + \sum_{p=n+1}^{n+m-1} \sum_{l=1}^h w_{jlp}^{[m]} \mu_{jlp}^{[m]}(\hat{y}^{*[p-n]}). \quad (1.25)$$

Таким чином, каскадна нейронна мережа з нео-фаззі нейронів, що сформована m каскадами, містить $h \left(\sum_{p=1}^{m-1} p \right)$ параметрів.

Введемо вектор функцій належності для j -ого нео-фаззі нейрону m -ого каскаду

$$\begin{aligned} \mu_j^{[m]}(k) = & \left(\mu_{j11}^{[m]}(x_1(k)), \dots, \mu_{jh1}^{[m]}(x_1(k)), \mu_{j12}^{[m]}(x_2(k)), \right. \\ & \dots, \mu_{jh2}^{[m]}(x_2(k)), \dots, \mu_{jli}^{[m]}(x_i(k)), \dots, \mu_{jhn}^{[m]}(x_n(k)), \\ & \left. \dots, \mu_{j1,n+1}^{[m]}(\hat{y}^{*[1]}(k)), \dots, \mu_{jh,n+m-1}^{[m]}(\hat{y}^{*[m-1]}(k)) \right)^T \end{aligned} \quad (1.26)$$

та відповідний вектор синаптичних вагових коефіцієнтів

$$w_j^{[m]} = \left(w_{j11}^{[m]}, \dots, w_{jh1}^{[m]}, w_{j12}^{[m]}, \dots, w_{jh2}^{[m]}, \dots, w_{jli}^{[m]}, \right. \\ \left. \dots, w_{jhn}^{[m]}, w_{j1,n+1}^{[m]}, \dots, w_{jh,n+m-1}^{[m]}, \right)^T. \quad (1.27)$$

Тоді можемо компактно записати вихідні сигнали для j -ого нейрону m -ого каскаду

$$\hat{y}_j^{[m]}(k) = w_j^{[m]T} \mu_j^{[m]}(k). \quad (1.28)$$

У такому разі критерій навчання (1.6) приймає вигляд

$$E_j^{[m]}(k) = \frac{1}{2} (e_j^m(k))^2 = \frac{1}{2} \left(y(k) - w_j^{[m]T} \mu_j^{[m]}(k) \right), \quad (1.29)$$

а мінімізувати його можна використавши модифікацію процедури [9] для «плаваючого» вікна

$$\begin{cases} w_j^{[m]}(k+1) = w_j^{[m]}(k) + \frac{e_j^{[m]}(k+1) \mu_j^{[m]}(k+1)}{r_j^{[m]}(k+1)}, \\ r_j^{[m]}(k+1) = r_j^{[m]}(k) + \left\| \mu_j^{[m]}(k+1) \right\|^2 - \left\| \mu_j^{[m]}(k-s) \right\|^2, + \end{cases} \quad (1.30)$$

або для випадку, коли $s = 1$,

$$w_j^{[m]}(k+1) = w_j^{[m]}(k) + \frac{e_j^{[m]}(k+1) \mu_j^{[m]}(k+1)}{\left\| \mu_j^{[m]}(k+1) \right\|^2}, \quad (1.31)$$

що збігається з одношаговим оптимальним алгоритмом Качмаж-Уідроу-Гоффа.

Вочевидь, замість (1.30) можна скористатись іншими алгоритмами, як-от експоненційно зважений рекурентний метод найменших квадратів (EWRLSM), що використовується у DENFIS [18], ETS [1] та FLEXFIX [2, 22]. Та варто зауважити, що EWRLSM може бути нестійким при невисокому коефіцієнті забувкуватості.

При використанні критерію навчання з **регулюючим параметром** (momentum term) (1.6) замість (1.29) отримаємо остаточний метод навчання нео-фаззі нейрона

$$\begin{cases} w_j^{[m]}(k+1) = w_j^{[m]}(k) + \frac{\eta e_j^{[m]}(k+1) \mu_j^{[m]}(k+1)}{r_j^{[m]}(k+1)} \\ \quad + \frac{(1-\eta)(w_j^{[m]}(k) - w_j^{[m]}(k-1))}{r_j^{[m]}(k+1)}, \\ r_j^{[m]}(k+1) = r_j^{[m]}(k) + \left\| \mu_j^{[m]}(k+1) \right\|^2 - \left\| \mu_j^{[m]}(k-s) \right\|^2. \end{cases} \quad (1.32)$$

Варто наголосити, що оскільки вихідні сигнали нео-фаззі нейрону лінійно залежать від його синаптичних вагових коефіцієнтів, можна використовувати будь-які методи адаптивної лінійної ідентифікації [21] (наприклад, рекурсивний метод найменших квадратів другого порядку, робастні методи, методи, що ігнорують застарілі данні, тощо), що дозволяє обробляти нестационарні сигнали в онлайн режимі.

1.4. Розширенні нео-фаззі нейрони в якості елементів гібридної каскадної мережі, що еволюціонує

Як зазначалося вище, розглядаючи нелінійний синапс нео-фаззі нейрону з позицій нечіткої логіки, нескладно бачити, що він є вельми схожим на шар фазифікування таких нейро-фаззі систем як мережі Такагі-Сугено-Канґа, Дженґа, Ванґа-Менделя, і, фактично реалізує нечітке висновування

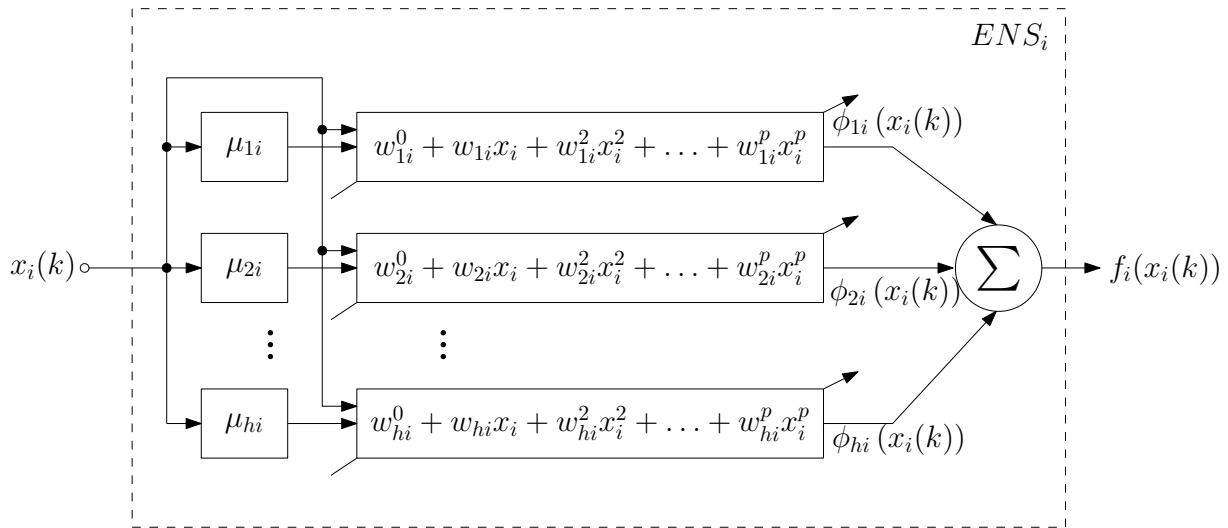


Рис. 1.5. Синапс розширеного нео-фаззі нейрону

Такагі-Сугено нульового порядку [31, 32]. Та задля поліпшення апроксимуючих властивостей таких систем видається доцільним запропонувати удосконалений нелінійний синапс, такий, що реалізує нечітке висновування довільного порядку, далі «розширений нелінійний синапс» (ENS), та зсинтезувати «розширений нео-фаззі нейрон» (ENFN), що містить такі структури замість традиційних нелінійних синапсів NS_i . Архітектури розширеного нелінійного синапсу та розширеного нео-фаззі нейрону наведено на рис. 1.5 та рис. 1.6 відповідно.

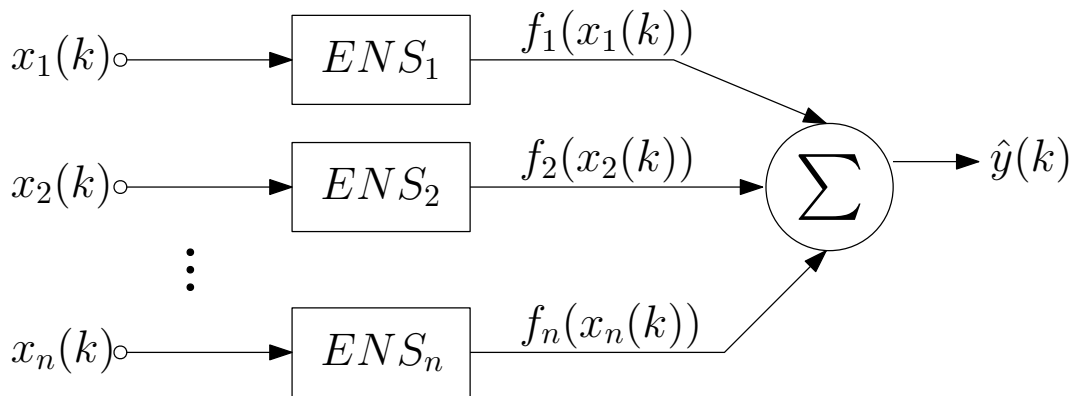


Рис. 1.6. Розширений нео-фаззі нейрон

Вводячі нові змінні

$$\phi_{li}(x_i) = \mu_{li}(x_i) (w_{li}^0 + w_{li}^1 x_i + w_{li}^2 x_i^2 + \dots + w_{li}^p x_i^p), \quad (1.33)$$

$$\begin{aligned}
f_i(x_i) &= \sum_{l=1}^h \mu_{li}(x_i) (w_{li}^0 + w_{li}^1 x_i + w_{li}^2 x_i^2 + \dots + w_{li}^p x_i^p) \\
&= w_{li}^0 \mu_{li}(x_i) + w_{li}^1 x_i \mu_{li}(x_i) + \dots + w_{li}^p x_i^p \mu_{li}(x_i)
\end{aligned} \tag{1.34}$$

$$\begin{aligned}
&+ w_{2i}^0 \mu_{2i}(x_i) + \dots + w_{2i}^p x_i^p \mu_{2i}(x_i) + \dots + w_{hi}^p x_i^p \mu_{hi}(x_i), \\
w_i &= (w_{1i}^0, w_{1i}^1, \dots, w_{1i}^p, w_{2i}^0, \dots, w_{2i}^p, \dots, w_{hi}^p)^T,
\end{aligned} \tag{1.35}$$

$$\begin{aligned}
\tilde{\mu}_i(x_i) &= \left(\mu_{1i}(x_i), x_i(\mu_{1i}(x_i)), \dots, x_i^p(\mu_{1i}(x_i)), \right. \\
&\quad \left. \mu_{2i}(x_i), \dots, x_i^p \mu_{2i}(x_i), \dots, x_i^p \mu_{hi}(x_i) \right)^T
\end{aligned} \tag{1.36}$$

можна представити вихідні сигнали розширеного нео-фаззі нейрона у вигляді

$$f_i(x_i) = w_i^T \tilde{\mu}_i(x_i), \tag{1.37}$$

$$\begin{aligned}
\hat{y} &= \sum_{i=1}^n f_i(x_i) \\
&= \sum_{i=1}^n w_i^T \tilde{\mu}(x_i) \\
&= \tilde{w}^T \tilde{\mu}(x),
\end{aligned} \tag{1.38}$$

де

$$\tilde{w}^T = (w_1^T, \dots, w_i^T, \dots, w_n^T)^T, \tag{1.39}$$

$$\tilde{\mu}(x) = (\tilde{\mu}_1^T(x_1), \dots, \tilde{\mu}_i^T(x_i), \dots, \tilde{\mu}_n^T(x_n))^T, \tag{1.40}$$

Таким чином, ENFN містить $(p+1)hn$ вагових коефіцієнтів та реалізує нечітке висновування Такагі-Сугено p -ого порядку, а висновування, що його реалізує кожний розширений нелінійний синапс ENS_i можна записати у формі

$$\begin{aligned} \text{IF } x_i \text{ IS } X_{li} \text{ THEN THE OUTPUT IS} \\ w_{li}^0 + w_{li}^1 x_i + \dots + w_{li}^p x_p, \quad l = 1, 2, \dots, h, \end{aligned} \quad (1.41)$$

що збігається з нечітким висновуванням Такагі-Сугено p -ого порядку.

Коли подати векторний сигнал $x(k)$ на вхід $ENFN$ першого каскаду, на виході отримуємо скалярне значення

$$\hat{y}^{[1]}(k) = \tilde{w}^{[1]T}(k-1) \tilde{\mu}^{[1]}(x(k)), \quad (1.42)$$

що відрізняється від виразу (1.28) для звичайних NFN тим, що містить у $p+1$ більше параметрів, що корегуються.

Вочевидь, будь-які методи навчання нео-фаззі нейронів підійдуть і для розширених нео-фаззі нейронів. Так вирази (1.30) та (1.31) для j -ого нейрону m -ого каскаду приймають вигляд

$$\begin{cases} \tilde{w}_j^{[m]}(k+1) = \tilde{w}_j^{[m]}(k) + \frac{e_j^{[m]}(k+1) \tilde{\mu}_j^{[m]}(k+1)}{\tilde{r}_j^{[m]}(k+1)}, \\ \tilde{r}_j^{[m]}(k+1) = \tilde{r}_j^{[m]}(k) + \left\| \tilde{\mu}_j^{[m]}(k+1) \right\|^2 - \left\| \tilde{\mu}_j^{[m]}(k-s) \right\|^2 \end{cases} \quad (1.43)$$

та

$$\tilde{w}_j^{[m]}(k+1) = \tilde{w}_j^{[m]}(k) + \frac{e_j^{[m]}(k+1) \tilde{\mu}_j^{[m]}(k+1)}{\left\| \tilde{\mu}_j^{[m]}(k+1) \right\|^2} \quad (1.44)$$

відповідно.

1.5. Оптимізація пулу нео-фаззі нейронів

Вихідні сигнали, згенеровані нейронами пулу кожного з каскадів, можна об'єднати у окремому вузлі, нейроні $GN^{[m]}$, з точністю $\hat{y}^{*[m]}(k)$, не меншою

від точності будь-якого нейрону пулу $\hat{y}_j^{[m]}(k)$. Це завдання можна вирішити за допомогою підходу ансамблей нейронних мереж. Хоча відомі алгоритми не призначені для роботи в онлайн-режимі, варто розглянути методи адаптивного узагальнюючого прогнозування [26, 30].

Введемо вектор ухідних сигналів для m -ого каскаду:

$$\hat{y}^{[m]}(k) = \left(\hat{y}_1^{[m]}(k), \hat{y}_2^{[m]}(k), \dots, \hat{y}_q^{[m]}(k) \right)^T; \quad (1.45)$$

тоді оптимальний вихідний сигнал, що його генерує нейрон $GN^{[m]}$ (що, власне, є адаптивним лінійним асоціатором [11, 13]), можна записати у формі

$$\hat{y}^{*[m]}(k) = \sum_{j=1}^1 c_j^{[m]} \hat{y}_j^{[m]}(k) = c^{[m]T} \hat{y}^{[m]}(k) \quad (1.46)$$

з обмеженнями на «незміщенність» **unbiasedness - незсунутість?**

$$\sum_{j=1}^q c_j^{[m]} = E^T c^{[m]} = 1, \quad (1.47)$$

де $c^{[m]} = (c_1^{[m]}, c_2^{[m]}, \dots, c_q^{[m]})^T$ та $E = (1, 1, \dots, 1)^T$ – $(q \times 1)$ -вектори.

Введемо критерій навчання на «плаваючому» вікні

$$\begin{aligned} E^{[m]}(k) &= \frac{1}{2} \sum_{\tau=k-s+1}^k (y(\tau) - \hat{y}^{*[m]}(\tau))^2 \\ &= \frac{1}{2} \sum_{\tau=k-s+1}^k (y(\tau) - c^{[m]T} \hat{y}^{[m]}(\tau))^2, \end{aligned} \quad (1.48)$$

зважаючи на обмеження (1.47), функція Лагранжа матиме вигляд

$$L^{[m]}(k) = E^{[m]}(k) - \lambda(1 - E^T c^{[m]}), \quad (1.49)$$

де λ – невизначений Лагранжів множник.

Мінімізуючи (1.49) відносно $c^{[m]}$, отримуємо

$$\begin{cases} \hat{y}^{*[m]}(k+1) = \frac{\hat{y}^{[m]T}(k+1)P^{[m]}(k+1)E}{E^T P^{[m]}(k+1)E}, \\ P^{[m]}(k+1) = \left(\sum_{\tau=k-s+2}^{k+1} \hat{y}^{[m]}(\tau)\hat{y}^{[m]T}(\tau) \right)^{-1} \end{cases} \quad (1.50)$$

або у рекурентній формі

$$\begin{cases} \tilde{P}^{[m]}(k+1) = P^{[m]}(k) - \frac{P^{[m]}(k)\hat{y}^{[m]}(k+1)\hat{y}^{[m]T}(k+1)P^{[m]}(k)}{1 + \hat{y}^{[m]T}(k+1)P^{[m]}(k)\hat{y}^{[m]}(k+1)}, \\ P^{[m]}(k+1) = \tilde{P}^{[m]}(k+1) + \\ \quad \frac{\tilde{P}^{[m]}(k+1)\hat{y}(k-s+1)\hat{y}^{[m]T}(k-s+1)\tilde{P}^{[m]}(k+1)}{1 - \hat{y}^{[m]T}(k-s+1)\tilde{P}^{[m]}(k+1)\hat{y}^{[m]}(k-s+1)}, \\ \hat{y}^{*[m]}(k+1) = \frac{\hat{y}^{[m]T}(k+1)P^{[m]}(k+1)E}{E^T P^{[m]}(k+1)E}. \end{cases} \quad (1.51)$$

У випадку, коли $s = 1$ (1.50) та (2.17) приймають вельми простий вигляд:

$$\begin{aligned} \hat{y}^{*[m]}(k+1) &= \frac{\hat{y}^{[m]T}(k+1)\hat{y}^{[m]}(k+1)}{E^T \hat{y}^{[m]}(k+1)} \\ &= \frac{\|\hat{y}^{[m]}(k+1)\|^2}{E^T \hat{y}^{[m]}(k+1)} \\ &= \frac{\sum_{j=1}^q (\hat{y}^{[m]}(k+1))^2}{\sum_{j=1}^q \hat{y}^{[m]}(k+1)}. \end{aligned} \quad (1.52)$$

Важливо зазначити, що навчання як нео-фаззі нейронів, так і нейронів-узгалгальнювачів можна організовувати в онлайн-режимі. Таким чином, вагові коефіцієнти нейронів попередніх каскадів (на відміну від CasCorLA) можна не заморозжувати, а постійно корегувати. Так само, число каскадів не має бути фіксованим і може змінюватись у часі, що відрізняє пропоновану нейронної мережі від інших відомих каскадних систем.

Висновки до розділу 1

1. Розглянуті інсууючі гібридні системи, що еволюціонують, та визначені потенційні модифікації, що їх варто привнести аби такі системи можна було застосувати у режимі послідовного надхоження даних на обробку.
2. Зсинетзована варіація каскадної системи, що еволюціонує, побудована на персептронах Розенблата, для послідовного обробляння ухідних сигналів, що дозволило сформулювати вимоги до вузлів шуканої гідбридної системи.
3. Запропонована архітектура та методи навчання гібридної каскадної системи, що еволюціонує, заснованої на нео-фаззі нейронах. Пропонованій система притаманні усі переваги нео-фаззі нейронів (інтерпритуємість та прозорість поряд з хорошими апроксимаційними властивостями), а також, зрештою, вона забезпечує модель адекватної складності для кожного поставленого завдання.
4. Запропонована архітектура та методи навчання гібридної каскадної нейронної мережі, що еволюціонує, з оптимізацією пулу нейронів у кожному каскаді, що реалізують оптимальний за точністю прогноз нелінійних стохастичних і хаотичних сигналів у онлайн режимі. Варто зазначити, що оптимізіція пулу нейронів дуже доречна саме у разі застосування системи для аналізу даних в онлайн режимі, адже використання узагальнюючих нейронів дозволяє визначати оптимальний нейрон на кожному етапі функціонування системи, який з високою верогідністю може змінюватися у випадку послідовного обробляння нестационарних об'єктів.
5. Запропонований розширений нео-фаззі нейрон, який дозволяє реалізовувати нечітке висновуння за Такагі-Сугено довільного порядку, що має покращенні апроксимуючі властивості. Зсинтезована архітекутра гібридної системи, що ґрунтується на розширених нео-фаззі нейронах.

РОЗДІЛ 2

БАГАТОВИМІРНА КАСКАДНА НЕО-ФАЗЗІ СИСТЕМА, ЩО ЕВОЛЮЦІОНУЄ

Задача апроксимації та екстраполяції багатовимірних часових рядів доволі часто виникає у багатьох технічних, медико-біологічних та інших дослідженнях, де якість прийнятих рішень істотно залежить від точності синтезованих прогнозів. У багатьох реальних задачах часові ряди характеризуються високим рівнем нелінійності та нестаціонарності своїх параметрів, наявністю аномальних викидів. Зрозуміло, що традиційні методи аналізу часових рядів, засновані на регресійному, кореляційному та інших подібних підходах, що мають на меті апріорну наявність доволі великої вибірки спостережень, є неефективними. Альтернативою традиційним статистичним методам може слугувати математичний апарат обчислювального інтелекту, зокрема штучні нейронні мережі [1, 2] та нейро-фаззі-системи [3], завдяки своїм універсальним апроксимувальним властивостям. Водночас з апроксимувальних властивостей зовсім не витікають екстраполуючі, оскільки врахування давньої передісторії для побудови прогнозувальної моделі може погіршити якість прогнозу. У зв'язку з цим під час оброблення нестаціонарних процесів треба відмовитися від процедур навчання, що базуються на зворотному поширенні помилок (багатошарові персептрони, рекурентні нейронні мережі, адаптивні нейромережеві системи нечіткого виведення – ANFIS) або методі найменших квадратів (радіально-базисні та функціонально пов'язані нейронні мережі) та скористатися процедурами на основі локальних критеріїв та «короткої» пам'яті типу алгоритма Качмажа-Уїдроу-Хоффа. При цьому використанні алгоритми навчання мусять забезпечувати не лише високу швидкодію, але й фільтруючі якості для придушення стохастичної «шумової» компоненти в оброблюваному сигналі. У зв'язку з цим синтез спеціалізованих гібридних

систем обчислювального інтелекту для розв'язання задач прогнозування істотно нестаціонарних часових рядів за умов невизначеності, що забезпечують разом з високою швидкістю навчання і фільтрацію завад, є досить цікавою та перспективною задачею.

Таким чином, цей розділ присвячено синтезу багатовимірної гібридної системи обчислювального інтелекту, що здатна реалізувати нелінійне відображення $R^n \rightarrow R^g$ у режимі реального часу.

2.1. Багатовимірна каскадна система, що еволюціонує, побудована на нео-фаззі нейронах

Для вирішення задачі прогнозування та ідентифікації багатовимірних даних в умовах апріорної і поточної структурної та параметричної невизначеності як ніколи доречні переваги каскадно-кореляційної архітектури, адже системи з такою архітектурою успадковують всі переваги елементів, які використовуються в їх вузлах, а в процесі навчання автоматично підбирається необхідна кількість каскадів для того, щоб отримати модель адекватної складності для вирішення поставленого завдання [12]. Однак, слід зазначити, що каскадно-кореляційна мережа у формі, що її запропонували С. Фальман і К. Ліб'єр, є системою з одним виходом, тобто не здатна реалізувати нелінійне відображення $R^n \rightarrow R^g$. Це досить серйозне обмеження, оскільки більшість практичних завдань містять кілька вихідних сигналів. Тож пропонуємо такі модифікації до архітектури каскадно-кореляційної мережі CasCorLA:

1. замість елементарних персептронів Розенблата використовувати нео-фаззі нейрони (доцільність такого рішення було детально показано у розділі 1),
2. кількість нейронів у кожному каскаді відтепер має дорівнювати розмірності вектору вихідного сигналу системи.

Схему пропонованої архітектури наведено на рис. 2.1.

Тоді вихідний сигнал системи формується з векторів, що його складають

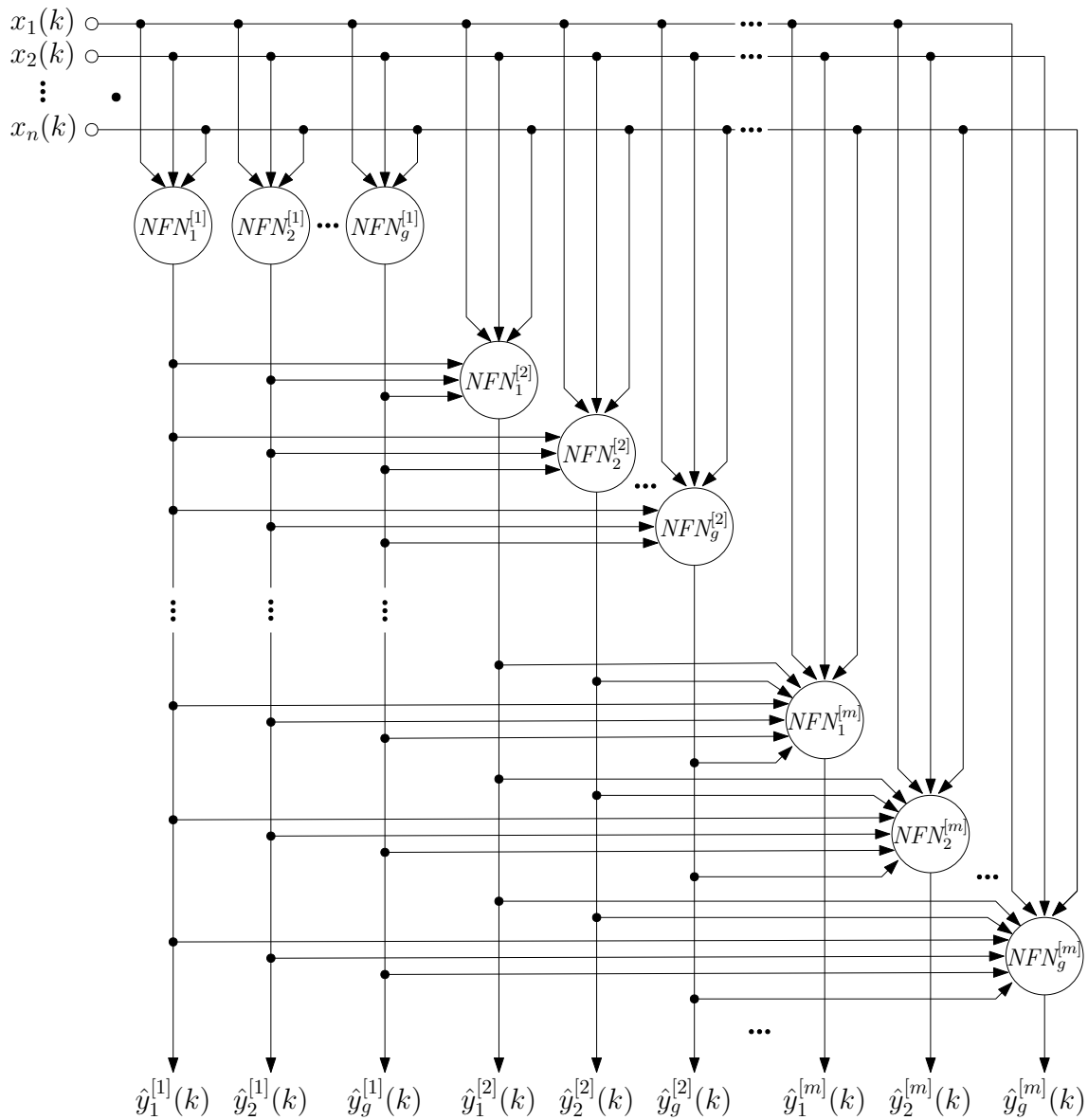


Рис. 2.1. Архітектура гібридної МІМО системи, побудованої на нео-фаззі нейронах

вихідні сигнали кращих нейронів останнього каскаду:

$$\hat{y}(k) = \left(\hat{y}_1^{*[m]}(k), \hat{y}_2^{*[m]}(k), \dots, \hat{y}_g^{*[m]}(k) \right)^T, \quad (2.1)$$

де g - кількість елементів вихідного вектору даних, що їх треба спрогнозувати чи ідентифікувати.

Для кожного з нео-фаззі нейронів системи в якості функцій належності можна використовувати трикутні конструкції:

$$\mu_{jli}^{[1]}(x_i) = \begin{cases} \frac{x_i - c_{d,l-1,i}^{[1]j}}{c_{dli}^{[1]j} - c_{d,l-1,i}^{[1]j}} \text{ якщо } x_i \in [c_{d,l-1,i}^{[1]j}, c_{dli}^{[1]j}], \\ \frac{c_{d,l+1,i}^{[1]j} - x_i}{c_{d,l+1,i}^{[1]j} - c_{dli}^{[1]j}} \text{ якщо } x_i \in [c_{dli}^{[1]j}, c_{d,l+1,i}^{[1]j}], \\ 0 \text{ у протилежному випадку,} \end{cases} \quad (2.2)$$

кубічні сплайни:

$$\mu_{jli}^{[1]}(x_i) = \begin{cases} \frac{1}{4} \left(2 + 3 \frac{2x_i - c_{dli}^{[1]j} - c_{d,l-1,i}^{[1]j}}{c_{dli}^{[1]j} - c_{d,l-1,i}^{[1]j}} - \left(\frac{2x_i - c_{dli}^{[1]j} - c_{d,l-1,i}^{[1]j}}{c_{dli}^{[1]j} - c_{d,l-1,i}^{[1]j}} \right)^3 \right), \\ \text{якщо } x \in [c_{d,l-1,i}^{[1]j}, c_{dli}^{[1]j}], \\ \frac{1}{4} \left(2 - 3 \frac{2x_i - c_{d,l+1,i}^{[1]j} - c_{dli}^{[1]j}}{c_{d,l+1,i}^{[1]j} - c_{dli}^{[1]j}} + \left(\frac{2x_i - c_{d,l+1,i}^{[1]j} - c_{dli}^{[1]j}}{c_{d,l+1,i}^{[1]j} - c_{dli}^{[1]j}} \right)^3 \right), \\ \text{якщо } x \in [c_{dli}^{[1]j}, c_{d,l+1,i}^{[1]j}], \\ 0 \text{ у протилежному випадку,} \end{cases} \quad (2.3)$$

або B -сплайни:

$$\mu_{jli}^{g[1]} = \begin{cases} \left. \begin{aligned} &1 \text{ якщо } x_i \in [c_{dli}^{[1]j}, c_{d,l+1,i}^{[1]j}], \\ &0 \text{ у протилежному випадку} \end{aligned} \right\} \text{ якщо } g = 1, \\ \frac{x_i - c_{dli}^{[1]j}}{c_{d,l+g-1,i}^{[1]j} - c_{dli}^{[1]j}} \mu_{dli}^{g-1,[1]j}(x_i) + \frac{c_{d,l+g,i}^{[1]j} - x_i}{c_{d,l+g,i}^{[1]j} - c_{d,l+1,i}^{[1]j}} \mu_{d,l+1,i}^{g-1,[1]j}(x_i), \\ \text{якщо } g > 1, \end{cases} \quad (2.4)$$

де $\mu_{dli}^{g[1]j}(x_i)$ – l -й сплайн g -ого порядку. Варто зауважити, що всі ці конструкції задовольняють умови одиничного розбиття Руспіні.

Запишемо вихідний сингнал j -ого нео-фаззі нейрону d -ого виходу першого каскаду у вигляді

$$\begin{cases} \hat{y}_d^{[1]j}(k) = \sum_{i=1}^n f_{di}^{[1]j}(x_i(k)) = \sum_{i=1}^n \sum_{l=1}^h w_{dli}^{[1]j} \mu_{dli}^{[1]j}(x_i(k)), \\ \text{ЯКЩО } x_i(k) \in X_{li}^j, \text{ ТОДІ ВИХІД } w_{dli}^{[1]j}. \end{cases} \quad (2.5)$$

вихідні сигнали нео-фаззі нейронів другого каскаду:

$$\begin{aligned} \hat{y}_d^{[2]j} = & \sum_{i=1}^n \sum_{l=1}^h w_{dli}^{[2]j} \mu_{dli}^{[2]j}(x_i) + \\ & \sum_{d=1}^g \sum_{l=1}^h w_{dl,n+1}^{[2]j} \mu_{dl,n+1}^{[2]j}(\hat{y}_d^{*[1]}) \quad \forall \quad d = 1, 2, \dots, g \end{aligned} \quad (2.6)$$

вихідні сигнали m -ого каскаду:

$$\begin{aligned} \hat{y}_d^{[2]j} = & \sum_{i=1}^n \sum_{l=1}^h w_{dli}^{[2]j} \mu_{dli}^{[2]j}(x_i) + \\ & \sum_{d=1}^g \sum_{p=n+1}^{n+m-1} \sum_{l=1}^h w_{dlp}^{[m]j} \mu_{dlp}^{[m]j}(\hat{y}_d^{*[p-n]}) \quad \forall \quad d = 1, 2, \dots, g \end{aligned} \quad (2.7)$$

Введемо до розгляду надалі вектор функцій належності j -ого нейрону d -ого виходу m -ого каскаду:

$$\begin{aligned} \mu_d^{[m]j}(k) = & \left(\mu_{d11}^{[m]j}(x_1(k)), \dots, \mu_{dh1}^{[m]j}(x_1(k)), \mu_{d12}^{[m]j}(x_2(k)), \right. \\ & \dots, \mu_{dh2}^{[m]j}(x_2(k)), \dots, \mu_{dli}^{[m]j}(x_i(k)), \dots, \mu_{dhn}^{[m]j}(x_n(k)), \\ & \left. \dots, \mu_{d1,n+1}^{[m]j}(\hat{y}^{*[1]}(k)), \dots, \mu_{dh,n+m-1}^{[m]j}(\hat{y}^{*[m-1]}(k)) \right)^T \end{aligned} \quad (2.8)$$

та відповідний йому вектор синаптичних вагових коефіцієнтів

$$\begin{aligned} w_d^{[m]j} = & \left(w_{d11}^{[m]j}, \dots, w_{dh1}^{[m]j}, w_{d12}^{[m]j}, \dots, w_{dh2}^{[m]j}, \dots, w_{dli}^{[m]j}, \right. \\ & \left. \dots, w_{dhn}^{[m]j}, w_{d1,n+1}^{[m]j}, \dots, w_{dh,n+m-1}^{[m]j} \right)^T, \end{aligned} \quad (2.9)$$

щоб записати вихідний сигнал системи у компактній формі:

$$\hat{y}_d^{[m]j}(k) = \left(w_d^{[m]j}\right)^T \mu_d^{[m]j}(k). \quad (2.10)$$

Для навчання нео-фаззі нейронів може бути використаний будь-який з методів адаптивної ідентифікації, що ми пропонували використовувати для навчання вузлів одновимірної нео-фаззі системи у першому розділі. Так корегувати вагові кофіцієнти можна за допомогою експоненційно зваженого рекурентного методу найменших квадратів:

$$\begin{cases} w_d^{[m]j}(k+1) = w_d^{[m]j}(k) + \\ \frac{P_d^{[m]j}(k) \left(y^d(k+1) - \left(w_d^{[m]j}(k) \right)^T \mu_d^{[m]j}(k+1) \right)}{\alpha + \left(\mu_d^{[m]j}(k+1) \right)^T P_d^{[m]j}(k) \mu_d^{[m]j}(k+1)} \mu_d^{[m]j}(k+1), \\ P_d^{[m]j}(k+1) = \frac{1}{\alpha} \left(P_d^{[m]j}(k) - \frac{P_d^{[m]j}(k) \mu_d^{[m]j}(k+1) \left(\mu_d^{[m]j}(k+1) \right)^T P_d^{[m]j}(k)}{\alpha + \left(\mu_d^{[m]j}(k+1) \right)^T P_d^{[m]j}(k) \mu_d^{[m]j}(k+1)} \right), \end{cases} \quad (2.11)$$

де $y^d(k+1)$, $d = 1, 2, \dots, g$ – зовнішній навчальний сигнал,

$0 < \alpha \leq 1$ – фактор забування;

або градієнтний метод навчання, що, як зазначалося, відрізняється як згладжувальними, так і слідкуючими властивостями:

$$\begin{cases} w_d^{[m]j}(k+1) = w_d^{[m]j}(k) + \frac{y^d(k+1) - \left(w_d^{[m]j}(k) \right)^T \mu_d^{[m]j}(k+1)}{r_d^{[m]j}(k+1)} \mu_d^{[m]j}(k+1), \\ r_d^{[m]j}(k+1) = \alpha r_d^{[m]j} + \left\| \mu_d^{[m]j}(k+1) \right\|^2, 0 \leq \alpha \leq 1. \end{cases} \quad (2.12)$$

2.1.1. Оптимізація пулу нео-фаззі нейронів багатовимірної каскадної системи, що еволюціонує. Оскільки за мету було поставлено синтез такої багатовимірної каскадної системи, що б могла працювати саме в режимі реального часу, було б дуже доречно, якби система могла самостійно визначати найліпшу кількість функцій належності та їх форму, адже ці параметри також можуть змінюватися у часі. Тому у цьому підрозділі пропонується у кожному каскаді збільшити кількість нео-фаззі нейронів до такої, що є кратною (а не дорівнює, як пропонуувалося у попередньому підрозділі) розмірності вектору вихідного сигналу ввести узагальнюючі нейрони, що для пулу кожного каскаду визначатимуть локально оптимальні вихідні сигнали (тут під «локально оптимальними вихідними сигналами» слід розуміти вихідні сигнали оптимальні у конкретний момент часу). Таким чином, коли g – розмірність вихідного векторного сигналу, а z – кількість відмінних типів нейронів (що відрізняються за кількістю чи характером функцій належності) системи, у пулі першого каскаду знаходиться zg нео-фаззі нейронів та g нейронів-узагальнювачів $GN^{[1]}$, пул другого каскаду містить $z(g+1)$ нейронів та $g+1$ нейронів $GN^{[2]}$, останній каскад - $z(g+m-1)$ нейронів та $g+m-1$ нейронів $GN^{[m]}$.

Схему такої оптимізованої MIMO (Multiple Input Multiple Output) архітектури зображено на рис. 2.1.

2.1.1.1. Метод визначення поточно оптимальних вихідних сигналів пулу нео-фаззі нейронів багатовимірної каскадної системи, що еволюціонує. Вихідні сигнали нейронів пулу кожного каскаду пропонується об'єднати узагальнюючим нейроном $GN_d^{[m]}$, що його було введено у ??? розділі.

Таким чином, у кожному каскаді системи маємо g $GN^{[m]}$ елементів, що узагальнюють вихідні сигнали нейронів пулу для кожного елементу вихідного вектору:

$$\hat{y}^{*[m]}(k) = \left(\hat{y}_1^{*[m]}(k), \hat{y}_2^{*[m]}(k), \dots, \hat{y}_q^{*[m]}(k) \right)^T; \quad (2.13)$$

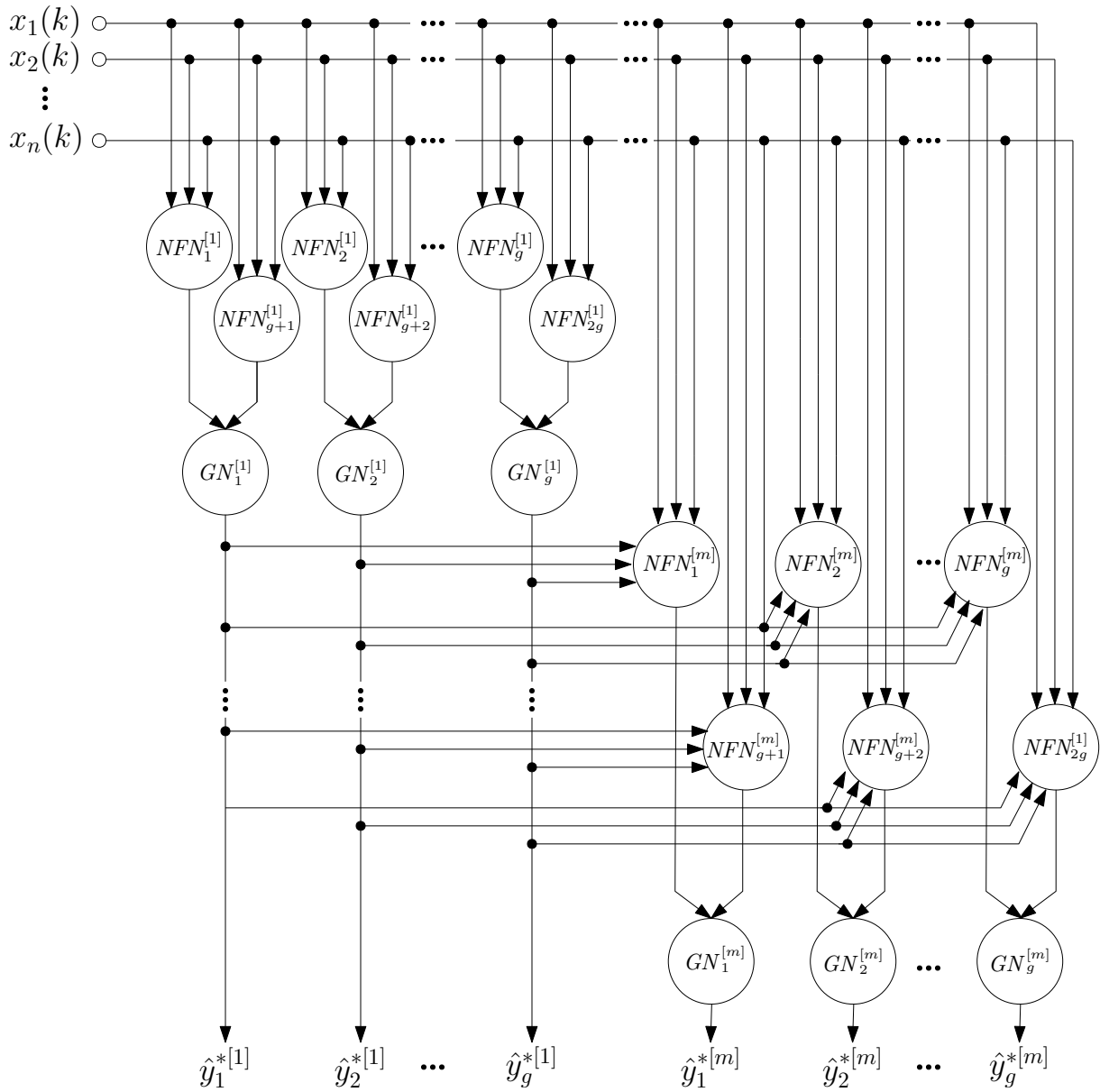


Рис. 2.2. Архітектура гібридної оптимізованої MIMO системи, побудованої на нео-фаззі нейронах

Так, до узагальнюючого елементу першого каскаду $GN_1^{[1]}$ подаються сигнали

$$\left(\hat{y}_1^{[1]}, \hat{y}_{g+1}^{[1]}, \dots, \hat{y}_{2g+1}^{[1]}, \dots, \hat{y}_{(z-1)(g+1)}^{[1]} \right)^T \quad (2.14)$$

до друго узагальнювача $GN_2^{[1]}$

$$\left(\hat{y}_2^{[1]}, \hat{y}_{g+2}^{[1]}, \dots, \hat{y}_{2g+2}^{[1]}, \dots, \hat{y}_{(z-1)(g+2)}^{[1]} \right)^T \quad (2.15)$$

і, нарешті, вектор ухідних сигналів останнього узагальнюючого елементу першого каскаду $GN_g^{[1]}$

$$\left(\hat{y}_1^{[g]}, \hat{y}_{2g}^{[1]}, \dots, \hat{y}_{(z-1)g}^{[1]} \right)^T \quad (2.16)$$

Нагадаймо, точність вихідного сигналу узагальнюючих елементів має бути не меншою від точності будь-якого сингналу, що узагальнюється (подається на вхід до $GN_d^{[m]}$). Рекурента форма метод навчання «на плаваючому вікні» елементів $GN_d^{[m]}$ кожного каскаду має вигляд

$$\left\{ \begin{array}{l} \tilde{P}_d^{[m]}(k+1) = P_d^{[m]}(k) - \frac{P_d^{[m]}(k) \hat{y}_d^{[m]}(k+1) \hat{y}_d^{[m]T}(k+1) P_d^{[m]}(k)}{1 + \hat{y}_d^{[m]T}(k+1) P_d^{[m]}(k) \hat{y}_d^{[m]}(k+1)}, \\ P_d^{[m]}(k+1) = \tilde{P}_d^{[m]}(k+1) + \\ \quad \frac{\tilde{P}_d^{[m]}(k+1) \hat{y}_d(k-s+1) \hat{y}_d^{[m]T}(k-s+1) \tilde{P}_d^{[m]}(k+1)}{1 - \hat{y}_d^{[m]T}(k-s+1) \tilde{P}_d^{[m]}(k+1) \hat{y}_d^{[m]}(k-s+1)}, \\ \hat{y}_d^{*[m]}(k+1) = \frac{\hat{y}_d^{[m]T}(k+1) P_d^{[m]}(k+1) E}{E^T P_d^{[m]}(k+1) E}, \end{array} \right. \quad (2.17)$$

а у випадку, коли $s = 1$:

$$\begin{aligned} \hat{y}_d^{*[m]}(k+1) &= \frac{\hat{y}_d^{[m]T}(k+1) \hat{y}_d^{[m]}(k+1)}{E^T \hat{y}_d^{[m]}(k+1)} \\ &= \frac{\left\| \hat{y}_d^{[m]}(k+1) \right\|^2}{E^T \hat{y}_d^{[m]}(k+1)} \\ &= \frac{\sum_{j=1}^q \left(\hat{y}_d^{[m]}(k+1) \right)^2}{\sum_{j=1}^q \hat{y}_d^{[m]}(k+1)}. \end{aligned} \quad (2.18)$$

2.2. Багатовимірна каскадна система, що еволюціонує, побудована на багатовимірних нео-фаззі нейронах

Архітектура багатовимірної каскадної системи, яка ґрунтується на звичайних нео-фаззі нейронах, що її описано у ?? є надмірною, адже вектор вхідних сигналів $x(k)$ (для першого каскаду) подається на однотипні нелінійні синапси $NS_{di}^{[1]j}$ нео-фаззі нейронів, кожен з яких на виході генерує сигнал $\hat{y}_d^{[1]j}(k)$, $d = 1, 2, \dots, g$. У результаті компоненти вихідного вектора

$$\hat{y}^{[1]j}(k) = \left(\hat{y}_1^{[1]j}(k), \hat{y}_2^{[1]j}(k), \dots, \hat{y}_g^{[1]j}(k) \right)^T \quad (2.19)$$

обчислюються незалежно один від одного, хоча $\mu_{1il}(x_i(k)) = \mu_{2il}(x_i(k)) = \mu_{jil}(x_i(k)) = \mu_{nil}(x_i(k))$. Уникнути цього можна, якщо ввести до розгляду багатовимірний нео-фаззі нейрон, що є модифікацією систем, запропонованих у [??,??].

2.2.1. Багатовимірний нео-фаззі нейрон. Вузлами багатовимірного нео-фаззі нейрону є складені нелінійні синапси $MNS_i^{[1]j}$, кожен з яких має h функцій належності $\mu_{li}^{[1]j}$ та gh настроюваних синаптичних вагових коефіцієнтів, але тільки hn функцій належності, що в g разів менше, ніж у випадку, коли каскад сформований із звичайних нео-фаззі нейронів.

Введемо надалі до розгляду $(hn \times 1)$ - вектор функцій належності

$$\mu^{[1]j}(k) = \left(\mu_{11}^{[1]j}(x_1(k)), \mu_{21}^{[1]j}(x_1(k)), \dots, \mu_{h1}^{[1]j}(x_1(k)), \dots, \mu_{hn}^{[1]j}(x_n(k)) \right)^T \quad (2.20)$$

та $(g \times hn)$ - матрицю синаптичних вагових коефіцієнтів

$$W^{[1]j} = \begin{pmatrix} w_{111}^{[1]j} & w_{112}^{[1]j} & \dots & w_{1li}^{[1]j} & \dots & w_{1hn}^{[1]j} \\ w_{211}^{[1]j} & w_{212}^{[1]j} & \dots & w_{2li}^{[1]j} & \dots & w_{2hn}^{[1]j} \\ \vdots & \vdots & & \vdots & & \vdots \\ w_{g11}^{[1]j} & w_{g12}^{[1]j} & \dots & w_{gli}^{[1]j} & \dots & w_{ghn}^{[1]j} \end{pmatrix}, \quad (2.21)$$

і запишемо сигнал на виході $MN_j^{[1]}$ у k -й момент часу у вигляді

$$\hat{y}^{[1]j}(k) = W^{[1]j} \mu^{[1]j}(k). \quad (2.22)$$

Навчання багатовимірного нео-фаззі нейрону можна реалізувати за допомогою матричної модифікації експоненційно-зваженого рекурентного методу найменших квадратів (2.11) у формі

$$\left\{ \begin{array}{l} W^{[1]j}(k+1) = W^{[1]j}(k) + \frac{(y(k+1) - W^{[1]j}(k)\mu^{[1]j}(k+1))(\mu^{[1]j}(k+1))^T P^{[1]j}(k)}{\alpha + (\mu^{[1]j}(k+1))^T P^{[1]j}(k)\mu^{[1]j}(k+1)}, \\ P^{[1]j}(k+1) = \frac{1}{\alpha} \left(P^{[1]j}(k) - \frac{P^{[1]j}(k)\mu^{[1]j}(k+1)(\mu^{[1]j}(k+1))^T P^{[1]j}(k)}{\alpha + (\mu^{[1]j}(k+1))^T P^{[1]j}(k)\mu^{[1]j}(k+1)} \right), \\ 0 < \alpha \leq 1 \end{array} \right. \quad (2.23)$$

або багатовимірного варіанту методу (2.12)

$$\left\{ \begin{array}{l} W^{[1]j}(k+1) = W^{[1]j}(k) + \frac{y(k+1) - W^{[1]j}(k)\mu^{[1]j}(k+1)}{r^{[1]j}(k+1)} \\ \quad \times (\mu^{[1]j}(k+1))^T, \\ r^{[1]j}(k+1) = \alpha r^{[1]j}(k) + \|\mu^{[1]j}(k+1)\|^2, \\ 0 \leq \alpha \leq 1, \end{array} \right. \quad (2.24)$$

де $y(k+1) = (y^1(k+1), y^2(k+1), \dots, y^g(k+1))^T$.

Аналогічним чином проводиться навчання інших каскадів, при цьому вектор функцій належності m -го каскаду $\mu^{[m]j}(k+1)$ збільшує свою розмірність на $(m-1)g$ компоненти, що їх утворили виходи попередніх каскадів.

2.2.2. Метод визначення поточно оптимального вихідного сигналу пулу багатовимірних нео-фаззі нейронів каскадної системи, що еволюціонує. У цьому підрозділі буде запропоновано узагальнюючий нейрон $GMN^{[m]}$ та рекурентний метод його навчання, щоб він об'єднував усі вихідні сигнали нейронів $MN[m]$ пулу каскаду у сигнал

$$\hat{y}^{*[m]}(k) = \left(\hat{y}_1^{*[m]}(k), \hat{y}_2^{*[m]}(k), \dots, \hat{y}_g^{*[m]}(k) \right)^T, \quad (2.25)$$

з точністю не меншою від точності будь-якого з сигналів $\hat{y}_j^{[m]}(k)$.

Розв'язати це завдання можна, знову скориставшись апаратом невизначених множників Лагранжа та адаптивного багатовимірного узагальненого прогнозування

Введемо до розгляду вихідний сигнал нейрону $GMN^{[m]}$ у вигляді

$$\hat{y}^{*[m]}(k) = \sum_{j=1}^q c_j^{[m]} \hat{y}_j^{[m]}(k) = \hat{y}^{[m]}(k) c^{[m]}, \quad (2.26)$$

де $\hat{y}^{[m]}(k) = \left(\hat{y}_1^{[m]}(k), \hat{y}_2^{[m]}(k), \dots, \hat{y}_g^{[m]}(k) \right)^T$ – $(g \times q)$ -матриця

$c^{[m]}$ – $(q \times 1)$ -вектор коефіцієнтів узагальнення, що відповідають умовам незміщеності

$$\sum_{j=1}^q c_j^{[m]} = E^T c^{[m]} = 1, \quad (2.27)$$

$E = (1, 1, \dots, 1)^T$ – вектор, утворений одиницями.

Введемо критерій навчання

$$\begin{aligned}
 E^{[m]}(k) &= \sum_{\tau=1}^k \|y(\tau) - \hat{y}^{[m]}(\tau)c^{[m]}\|^2 \\
 &= Tr\left(\left(Y(k) - \hat{Y}^{[m]}(k)I \otimes c^{[m]}\right)^T \left(Y(k) - \hat{Y}^{[m]}(k)I \otimes c^{[m]}\right)\right)
 \end{aligned} \tag{2.28}$$

де $Y(k) = (y^T(1), y^T(2), \dots, y^T(k))^T$ – $(k \times s)$ матриця спостережень,

$$\hat{Y}^{[m]}(k) = \begin{pmatrix} \hat{y}_1^{[m]T}(1) & \hat{y}_2^{[m]T}(1) & \dots & \hat{y}_q^{[m]T}(1) \\ \hat{y}_1^{[m]T}(2) & \hat{y}_2^{[m]T}(2) & \dots & \hat{y}_q^{[m]T}(2) \\ \vdots & \vdots & & \vdots \\ \hat{y}_1^{[m]T}(k) & \hat{y}_2^{[m]T}(k) & \dots & \hat{y}_q^{[m]T}(k) \end{pmatrix}, \tag{2.29}$$

де I – одинична $(g \times g)$ матриця,

\otimes – символ тензорного добутку.

З урахуванням ?? обмежень запишемо функцію Лагранжа

$$\begin{aligned}
 L^{[m]}(k) &= E^{[m]}(k) + \lambda(E^T c^{[m]} - 1) \\
 &= \sum_{\tau=1}^k \|y(\tau) - \hat{y}^{[m]}(\tau)c^{[m]}\|^2 + \lambda(E^T c^{[m]} - 1) \\
 &= Tr\left(\left(Y(k) - \hat{Y}^{[m]}(k)I \otimes c^{[m]}\right)^T \left(Y(k) - \hat{Y}^{[m]}(k)I \otimes c^{[m]}\right)\right) \\
 &\quad + \lambda(E^T c^{[m]} - 1) \\
 &= Tr(V^{[m]T}(k)V^{[m]}(k)) + \lambda(E^T c^{[m]} - 1),
 \end{aligned} \tag{2.30}$$

де $V^{[m]}(k) = Y(k) - \hat{Y}^{[m]}(k)I \otimes c^{[m]}$ – $(k \times g)$ матриця оновлень.

Розв'язання системи рівнянь Каруша-Куна-Таккера

$$\begin{cases} \nabla_{c^{[m]}} L^{[m]}(k) = \vec{0}, \\ \frac{\partial L^{[m]}(k)}{\partial \lambda} = 0 \end{cases} \tag{2.31}$$

призводить до очевидного результату

$$\begin{cases} c^{[m]} = (R^{[m]}(k))^{-1} E (E^T (R^{[m]}(k))^{-1})^{-1} \\ \lambda = -2 E^T (R^{[m]}(k))^{-1} E, \end{cases} \quad (2.32)$$

де $R^{[m]}(k) = V^{[m]T}(k) V^{[m]}(k)$.

Таким чином, можна організувати оптимальне об'єднання виходів усіх нейронів пулу кожного каскаду. Зрозуміло, що в якості таких нейронів можуть використовуватися не тільки багатовимірні нео-фаззі нейрони, але й будь-які інші конструкції, що реалізують нелінійне відображення $R^{n+(m-1)g} \rightarrow R^g$.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Angelov Plamen P, Filev Dimitar P. An approach to online identification of Takagi-Sugeno fuzzy models // Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on. — 2004. — Vol. 34, no. 1. — P. 484–498.
- [2] Angelov Plamen P., Lughofer Edwin. Data-driven evolving fuzzy systems using eTS and FLEXFIS: comparative analysis // Int. J. General Systems. — 2008. — Vol. 37, no. 1. — P. 45–67. — URL: <http://dx.doi.org/10.1080/03081070701500059>.
- [3] Bodyanskiy Yevgeniy, Grimm Paul, Teslenko Nataliya. Evolving cascaded neural network based on multidimensional Epanechnikov's kernels and its learning algorithm // Int. J. Information Technologies and Knowledge. — 2011. — Vol. 5, no. 1. — P. 25–30.
- [4] Bodyanskiy Yevgeniy, Kharchenko Oleksandra, Vynokurova Olena. Hybrid cascade neural network based on wavelet-neuron // Information Theories and Application. — 2011. — Vol. 18, no. 4. — P. 335–343.
- [5] Bodyanskiy Yevgeniy, Kokshenev Illya, Kolodyazhnyi Vitaliy. An adaptive learning algorithm for a neo fuzzy neuron. // EUSFLAT Conf. — 2003. — P. 375–379.
- [6] Bodyanskiy Yevgeniy, Viktorov Yevgen. The cascaded neo-fuzzy architecture using cubic-spline activation functions // Inf Theor Appl. — 2009. — Vol. 16, no. 3. — P. 245–259.
- [7] Bodyanskiy Yevgeniy, Viktorov Yevgen. 110 9 – Intelligent Processing THE CASCADE NEO-FUZZY ARCHITECTURE AND ITS ONLINE LEARNING ALGORITHM. — 2013. — URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.386.7492>.
- [8] Bodyanskiy Yevgeniy, Viktorov Yevgen, Pliss Iryna. International Book Series "Information Science and Computing "

25 THE CASCADE GROWING NEURAL NETWORK USING QUADRATIC NEURONS AND ITS LEARNING ALGORITHMS FOR ON-LINE INFORMATION PROCESSING. — 2013. — URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.386.9313>.

- [9] Bodyanskiy Ye V, Pliss IP, Solovyova T. Multistep optimal predictors of multidimensional non-stationary stochastic processes // Doklady AN USSR, series A. — 1986. — Vol. 12. — P. 47–49.
- [10] Chen L H, Chang S. An adaptive learning algorithm for principal component analysis. // IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council. — 1995. — Vol. 6, no. 5. — P. 1255–63.
- [11] Cichocki A., Unbehauen R. Neural networks for optimization and signal processing // Journal of Signal Processing. — 1998. — Vol. 2. — P. 62–63.
- [12] Fahlman Scott E., Lebiere Christian. The cascade-correlation learning architecture // Advances in Neural Information Processing Systems 2. — Morgan Kaufmann, 1990. — P. 524–532.
- [13] Haykin S. Neural Networks: a Comprehensive Foundation. — New York, NY : Macmillan, 1994.
- [14] Bodyanskiy Yevgeniy, Dolotov Artem, Pliss Iryna, Viktorov Yevgen. International Book Series "Information Science and Computing" 13 THE CASCADE ORTHOGONAL NEURAL NETWORK. — 2008.
- [15] Jang Jyh-Shing Roger, Sun Chuen-Tsai, Mizutani Eiji. Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence. MATLAB curriculum series. — pub-PH:adr : Prentice-Hall, 1997. — P. xxvi + 614. — ISBN: 0-13-261066-3.
- [16] Kaczmarz S. Approximate solution of systems of linear equations // Int. J. Control,. — 1993. — Vol. vol. 53. — P. 1269–1271.
- [17] Kasabov Nikola. Evolving connectionist systems - the knowledge engineering approach (2. ed.). — Springer, 2007. — P. I–XXI, 1–457. — URL: <http://dx.doi.org/10.1007/978-1-84628-347-5>.
- [18] Kasabov Nikola K, Song Qun. DENFIS: dynamic evolving neural-fuzzy

- inference system and its application for time-series prediction // *Fuzzy Systems, IEEE Transactions on.* — 2002. — Vol. 10, no. 2. — P. 144–154.
- [19] Kolodyazhniy V, Bodyanskiy Ye. Cascaded multiresolution spline-based fuzzy neural network // *Proc. Int. Symp. on Evolving Intelligent Systems.* — 2010. — P. 26–29.
- [20] Kruschke John K, Movellan Javier R. Benefits of gain: Speeded learning and minimal hidden layers in back-propagation networks // *Systems, Man and Cybernetics, IEEE Transactions on.* — 1991. — Vol. 21, no. 1. — P. 273–280.
- [21] Ljung Lennart. *System Identification: Theory for the User.* — Prentice-Hall, Inc., 1999.
- [22] Lughofer Edwin. FLEXFIS: A Robust Incremental Learning Approach for Evolving Takagi-Sugeno Fuzzy Models // *IEEE T. Fuzzy Systems.* — 2008. — Vol. 16, no. 6. — P. 1393–1410. — URL: <http://dx.doi.org/10.1109/TFUZZ.2008.925908>.
- [23] Lughofer Edwin. *Evolving Fuzzy Systems - Methodologies, Advanced Concepts and Applications.* — Springer, 2011. — Vol. 266 of *Studies in Fuzziness and Soft Computing.* — P. 1–410. — ISBN: 978-3-642-18086-6. — URL: <http://dx.doi.org/10.1007/978-3-642-18087-3>; <http://dx.doi.org/10.1007/978-3-642-18087-3>.
- [24] Miki T., (Japan) T. Yamakawa. *Analog Implementation of Neo-Fuzzy Neuron and Its On-board Learning.* — WSEAS, 1999. — URL: <http://www.worldses.org/online/>.
- [25] Nechyba Michael C., Xu Yangsheng. *Cascade Neural Networks with Node-Decoupled Extended Kalman Filtering.* — 1997.
- [26] Otto Peter, Bodyanskiy Yevgeniy, Kolodyazhniy Vitaliy. A new learning algorithm for a forecasting neuro-fuzzy network // *Integrated Computer-Aided Engineering.* — 2003. — Vol. 10, no. 4. — P. 399–409. — URL: <http://content.iospress.com/articles/integrated-computer-aided-engineering/ica00163>.
- [27] Prechelt L. *Investigation of the CasCor Family of Learning Algorithms* //

- Neural Networks. — 1997. — Vol. 10, no. 5. — P. 885–896.
- [28] Silva F. M., Almeida L. B. Speeding up back-propagation // Advanced Neural Computers / Ed. by R. Eckmiller. — Amsterdam : Elsevier, 1990. — P. 151–158.
- [29] Veitch A. C., Holmes G. A Modified Quickprop Algorithm // Neural Computation. — 1991. — Vol. 3, no. 3. — P. 310–311.
- [30] Vorobyov Sergiy A., Cichocki Andrzej, Bodyanskiy Yevgeniy V. Adaptive Noise Cancellation For Multi-Sensory Signals. — 2001. — URL: <http://citeseer.ist.psu.edu/404753.html>.
- [31] Wang Li-Xin. Adaptive fuzzy systems and control - design and stability analysis. — Prentice Hall, 1994. — P. I–XVII, 1–232. — ISBN: 978-0-13-099631-2.
- [32] Wang Li-Xin, Mendel Jerry M. Fuzzy basis functions, universal approximation, and orthogonal least-squares learning // IEEE Transactions on Neural Networks. — 1992. — Vol. 3, no. 5. — P. 807–814. — URL: <http://dx.doi.org/10.1109/72.159070>.
- [33] Widrow B., Hoff M. E. Adaptive Switching Circuits // Proceedings WESCON. — 1960. — P. 96–104.
- [34] Yamakawa T., Tomoda S. A Neo Fuzzy Neuron and its Applications to System Identification and Prediction of the System Behaviour // Proc. 2nd Int. Conf. on Fuzzy Logic and Neural Networks (IIZUKA'92). — Iizuka, 1992. — P. 477–483.
- [35] Ye Bodyanskiy, Vynokurova E, Teslenko N. Cascade GMDH-wavelet-neuro-fuzzy network // Proc. of the 4th Intern. Workshop on Inductive Modelling IWIM 2011. — 2011. — P. 22–30.