

УДК 004.512

Л.О.Катеринич<sup>1</sup>, к.ф.-м.н.

### Адаптація системи керування нейронною мережею

*Розглянуто деякі підходи щодо можливості адаптації системи керування нейронною мережею, а саме параметрична адаптація, структурна адаптація, адаптація об'єкта та цілей керування.*

*Ключові слова: нейронні мережі; штучний інтелект; адаптація.*

<sup>1</sup> Київський національний університет імені Тараса Шевченка, 03680, м. Київ, пр-т. Глушкова 4д, e-mail: katerinich@rambler.ru

L.Katerynych<sup>1</sup>, PhD.

### Adaptation of neural network control

*Considers some approaches the possibility of adapting the control system of neural networks, namely parametric adaptation, structural adaptation, adaptation of the object and purpose of control.*

*Key words: neural networks, artificial intelligence, adaptation.*

<sup>1</sup> Taras Shevchenko National University of Kyiv, 03680, Kyiv, Glushkova st., 4d, e-mail: katerinich@rambler.ru

Статтю представив д. т. н., проф. Заславський В.А.

### Вступ

Адаптація або корекція всієї системи курування (всіх етапів) розглядається, як більш глибокий аналіз процесу керування складної системи. Поняття адаптації як інструмента цілеспрямованого впливу на об'єкт останніми роками стало фігурувати в математичній, технічній літератур [1,2,3].

У широкому розумінні адаптація нічим не відрізняється від управління, тобто, адаптація, як і будь-яке управління, є організація такого цілеспрямованого впливу на об'єкт, при якому досягаються задані цілі.

Під час проектування необхідність введення адаптації добре відчуває будь-який проектувальник, якому доводиться створювати систему при значній апріорній невизначеності про умови її функціонування. Осереднення по цій невизначеності рідко буває вдалим. З іншого боку, будь-яке осереднення поведінки середовища дозволяє спроектувати систему, оптимально працюючу тільки при середньому стані середовища. Усяке ж відхилення середовища від середнього призводить до не оптимальності функціонування системи.

Складні системи, як правило, не мають єдиного критерію функціонування. Такого роду системи функціонують в стані

багатокритеріальності, причому ці критерії можуть бути не тільки екстремальними, але й мати характер обмежень. Це вимагає формулювати відразу кілька критеріїв та варіювати їх вибір залежно від ситуації, що склалася і внутрішніх потреб самої системи. Таким чином, вже вибір критеріїв адаптації є процесом адаптивним і повинен враховуватися при визначенні адаптації.

Тому, з урахуванням особливостей складних систем, адаптацію в широкому сенсі можна визначити як процес цілеспрямованого зміни параметрів і структури системи, який полягає в визначенні критеріїв її функціонування і виконанні цих критеріїв. Виділяють наступні підсистеми адаптації: апаратний рівень, алгоритмічний рівень, програмний рівень, системний та мережевий рівень [4].

**Параметрична адаптація** пов'язана з корекцією та налаштуванням параметрів моделі. Така адаптація має місце при будь-якій змінні характеристик керуючого об'єкта. Адаптація дозволяє налаштовувати модель на кожному кроці керування. Вихідною інформацією для цього виду адаптації - є неспівпадання вихідної/вхідної інформації та необхідною вхідною для керуючого об'єкта.

Оскільки процес керування нейронною мережею з керуючим об'єктом не завжди здатен представити необхідну інформацію для найбільш точного остаточного результату, необхідно вводити додаткові елементи керування, а саме поповнювати вхідною інформацією початкову структуру вхідних даних нейронної мережі (НМ).

**Структурна адаптація** має місце при неспівпаданні структури моделі та структури НМ. За різних обставин структура НМ може змінитися, що в свою чергу буди вимагати зміни структури моделі, а саме до механізмів структурної адаптації. Перехід від однієї структури до іншої виконується процедурою вибору альтернативної структури моделі. Різниця може полягати як у кількості входів/виходів до моделі, варіантами декомпозиції або структурою елементів моделі. Слід зазначити, що альтернативні моделі при ідентифікації параметрів теж використовують методи параметричної адаптації. [5]

**Адаптація об'єкта** необхідна у тому випадку, коли структурна адаптація не ефективна, тобто деякі цілі керування не досягаються. Ця адаптація пов'язана зі зміною організації потоків інформації в НМ. Додавання нової інформації призводить до підвищення його керованості, але вимагає більше затрат при параметричній та структурній адаптації. Вибір найкращого варіанту розподілення та переміщення потоків інформації у НМ в процесі керування і є основою процесу адаптації на цьому етапі.

**Адаптація цілей керування.** Інколи виникає ситуація коли зв'язок об'єкта і середовища настільки сильний, що їх відокремити неможливо. Коли предметна область чітка, то така задача не виникає. Це буває у таких випадках, коли предметна область автономна. Однак у реальному житті предметні області настільки пов'язані, що визначити чітку межу майже не можливо. Ця обставина є поштовхом для виділення процесу визначення меж предметної області та/або об'єкта в окремий етап [5,6].

У середовищі, що має формальний опис виділення об'єкта не є проблемою. Обмежуючи певні фактори середовища ми

отримуємо певний екземпляр об'єкта і перевіряємо на моделі досяжність поставлених цілей. Також можливо оцінити чисельно ступінь некерованості цього варіанту об'єкта. Повторюючи цю процедуру для інших варіантів об'єктів буде отримано той екземпляр об'єкта на якому може бути досягнута максимальна керованість процесом [7].

Маючи формальний опис середовища це не було б проблемою. Оскільки існування такого опису майже неможливо звертаються до експертів. Для цього необхідно експертно синтезувати декілька варіантів об'єкта, а потім за допомогою експерта оцінити їх за критерієм і обрати найкращий.

Таким чином виникає потреба виділити процес визначення об'єкта (КЕ) в самостійний етап керування. Задача зводиться до того, щоб для заданої множини цілей  $\{Z^*\}$  і ресурсів  $R$  визначити такий КЕ, який по критеріям досяжності цілей буде найкращий.

В робочому режимі системи де використовується НМ може виникати ситуація, коли виникають нові цілі, або їх уточнення. В цьому випадку необхідно визначити нову множину цілей  $\{Z^*\}$ , адаптовану під поточні потреби. Це є основою цього етапу [8, 9].

Розглянемо задачу адаптації об'єкта  $F^0$ , як задачу керування. Ціль  $Z^*$  адаптації визначає вимоги до критеріїв. Ці потреби можуть бути записні наступним чином:

1. Критерій-нерівність  
 $H(U) = (h_1(U), \dots, h_p(U)) \geq 0$
2. Критерій -рівність:  
 $G(U) = (g_1(U), \dots, g_s(U)) = 0$
3. Критерії, що мінімізують:  
 $Q(U) = (q_1(U), \dots, q_l(U)) \rightarrow \min$ , де  
 $h_i = M_x h_i'(Y) = M_x h_i'(F^0(X, U))$  ( $i = 1, \dots, p$ );  
 $g_i = M_x g_j'(Y) = M_x g_j'(F^0(X, U))$  ( $j = 1, \dots, s$ );  
 $g_k(U) = M_x q_k'(Y) = M_x q_k'(Y) = M_x q_k'(F^0(X, U))$   
 $k = (1, \dots, l)$ ;

$M_x$  - оператор усереднення по  $X$ :  
 $M(\bullet) = \int (\bullet) \rho(X) dX$ , де  $\rho(X)$  - щільність розподілу стану  $X$  середовища.

Ціль адаптації це розв'язок наступної задачі:

$$Q(U) \rightarrow \min_{U \in S} \Rightarrow U^*, \quad (1)$$

$$\text{де } S: \begin{cases} H(U) \geq 0; \\ G(U) = 0 \end{cases}$$

Такі задачі при заданій моделі  $F$  об'єкта  $F^0$  і розподілу  $\rho(X)$  називають задачам стохастичного програмування [6]. Проблема розв'язку поставлених задач очевидна. Але задачі адаптації ускладнюються ще й тим фактором, що модель об'єкта  $F$  відсутня. Також відсутні інформація о розподілі  $\rho(X)$ . Більш того, ці фактори мають тенденцію змінюватися у часі.

Отже, маючи алгоритм  $\varphi$  розв'язку задачі (1) с обмеженнями у вигляді

$$U^* = \varphi(F, \rho(X)) \quad (2)$$

не можна вважати задачу адаптації розв'язаною. Необхідно додатково ідентифікувати об'єкт, що змінюється  $F^0$  і статистичні властивості середовища, щоб отримати

$$W = V(F_t, \rho_t(X)) \quad (3)$$

де  $F_t$  - модель об'єкта, а  $\rho_t$  - щільність розподілу  $X$ . Ці обставини змушують відмовитися від пошуку алгоритму  $\varphi$  розв'язку поставленої задачі адаптації., оскільки для складного об'єкта адаптації залежності  $F_t$  і  $\rho_t(X)$  достатньо складні.

Тому для розв'язку задачі (1) необхідно звертатися к алгоритмам адаптації, використовуючи лише значення функцій  $h_i(\bullet)$ ,  $g_j(\bullet)$  і  $q_k(\bullet)$  ( $i = 1, \dots, p$ ;  $j = 1, \dots, s$ ;  $k = 1, \dots, l$ ) у визначені моменти часу. У загальному випадку має такий вигляд:

$$U_{N+1} = \psi(\overrightarrow{U}_N, \overrightarrow{H}_N, \overrightarrow{G}_N, \overrightarrow{Q}_N), \quad (10)$$

де  $\psi$  - алгоритм рекурентної адаптації.

$$\overrightarrow{U}_N = (U_N, U_{N-1}, \dots, U_{N-W});$$

$$\overrightarrow{H}_N = (H'_N, H'_{N-1}, \dots, H'_{N-W});$$

$$\overrightarrow{G}_N = (G'_N, G'_{N-1}, \dots, G'_{N-W});$$

$$\overrightarrow{Q}_N = (Q'_N, Q'_{N-1}, \dots, Q'_{N-W});$$

$$H' = (h'_1, \dots, h'_p);$$

$$G' = (g'_1, \dots, g'_s);$$

$$Q' = (q'_1, \dots, q'_l)$$

де  $U'_N, H'_N, G'_N, Q'_N$  - значення параметрів, що адаптуються,  $N$  - момент часу,  $W$  - глибина пам'яті алгоритму  $\psi$ .

Як будь-яке керування, адаптацію необхідно розглядати з позиції функціонування об'єкта адаптації. Розглядати задачу глобальної адаптації усієї нейронної мережі (НМ) з повною системою критеріїв на предмет ефективного функціонування не має сенсу.

Розглядаються локальні задачі адаптації, які охоплюють лише окремі аспекти функціонування НМ, і лише з цих підсистем формується повна система адаптації, яка буде дозволяти оперативно пристосовувати НМ до нових станів середовища. Розглядаються відповідні підсистемі адаптації НМ.

1. Апаратний рівень - адаптація налаштувань, параметрів і структури апаратних засобів.
2. Алгоритмічний рівень - адаптація алгоритмів обробки інформації з метою урахування специфіки задач, що розв'язуються НМ.
3. Програмний рівень - процес адаптації програм роботи з НМ при виборі їх з альтернативної множини.
4. Системний рівень - покращення функціонування системного забезпечення для підтримки та роботи НМ.
5. Мережевий рівень - адаптація процесів передачі інформації по НМ.

Алгоритмічний рівень адаптації НМ - адаптація алгоритмів обробки інформації в НМ. Мета такої адаптації - пристосувати алгоритми до специфіки задач, що розв'язуються. Критеріями ефективності, зазвичай, є середній час розв'язання задачі,

точність цього розв'язку, об'єм займаної пам'яті, ймовірність помилки, тощо.

Однак адаптація алгоритму може мати і структурний характер. Це означає, що потрібно вміти переходити від одного альтернативного алгоритму пошуку до іншого, з тим щоб весь час підтримувати в процесі оптимізації той алгоритм, який найкращим чином здійснює розв'язання даної задачі.

Програмна адаптація пов'язана з маніпулюванням окремими програмами і вибором тієї, яка працює найкращим чином в сенсі заданого критерію. Алгоритмічно програмна адаптація реалізується у вигляді альтернативного варіанту структурної адаптації.

Розглянемо алгоритмічний рівень адаптації для НМ. Мета такої адаптації - пристосувати алгоритми до специфіки задач, що розв'язуються НМ. Критеріями ефективності є середній час розв'язання задачі, точність цього розв'язку, ймовірність помилки тощо.

Специфіка задач, що розв'язуються, проявляється у вигляді статистично стійкою протягом деякого часу особливості цих задач, облік якої дозволяє підвищити ефективність функціонування НМ за обраним критерієм. Зазначена особливість не ідентифікується, тобто не визначається в явній формі, а враховується в процесі адаптації.

Побудуємо відповідну модель. Нехай

$$P_i = P(\Omega_i, \omega_i) \quad (4)$$

задачі, що розв'язує НМ, відрізняються структурою  $(\Omega_i)$  і параметрами  $(\omega_i)$ . Розглянемо задачі однієї структури  $\Omega$ :

$$P_i = P(\Omega_i, \omega_i) \quad (5)$$

Нехай

$$q_i = q(P_i, C) \quad (6)$$

- значення критерію ефективності розв'язання задачі  $P_i$  алгоритмом, який має параметри  $C = (c_1, \dots, c_k)$ .

Для вибору оптимального значення цих параметрів необхідно осереднити

критерії (3) для всіх задачами, що розв'язує НМ зі структурою  $\Omega$  (інша структура задачі вимагає іншого алгоритму розв'язання):

$$Q(C) = \int q(P, C) p(P) dP \quad (7)$$

де  $p(P)$  - щільність розподілу задач, що розв'язуються. Очевидно, що такої інформації практично ніколи не буває. Ця обставина змушує звертатися до адаптації параметрів алгоритму за спостереженнями окремих значень критерію (6), що з'являються при розв'язанні різних задач.

Адаптація параметрів алгоритму розв'язання задачі можлива не тільки від завдання до завдання, але і в процесі розв'язання однієї задачі і застосовується для рекурентних методів, коли можлива оцінка ефективності одного циклу - наприклад, у процесах пошукової оптимізації. Покажемо це. Алгоритм оптимізації, тобто алгоритм розв'язання задачі

$$Q(X) \rightarrow \min_{x \in S} \Rightarrow X^* \quad (8)$$

являє собою оператор  $F$  переходу від одного стану  $X_N$  об'єкта, що оптимізуємо, до іншого  $X_{N+1}$ , більш кращого за заданим критерієм  $Q$ :

$$X_{N+1} = X_N + \Delta X_{N+1}, \quad (9)$$

де  $\Delta X_N = F(X_N, C)$ , а  $C$  - параметри оператора оптимізації. Очевидно, що параметри  $C$  повинні змінюватися так, щоб процес оптимізації протікав найбільш успішно. Інакше кажучи, необхідна адаптація цих параметрів:

$$C_{N+1} = C_N + \Delta C_{N+1}, \quad (10)$$

де  $\Delta C_{N+1} = \varphi(C_N, X_N)$  - алгоритм адаптації.

Однак адаптація алгоритму може мати і структурний характер. Це означає, що потрібно вміти переходити від одного альтернативного алгоритму пошуку до іншого, з тим щоб весь час підтримувати в процесі оптимізації той алгоритм, який найкращим чином здійснює розв'язання даної задачі. Такі алгоритми альтернативної

адаптації лежать в основі адаптивного пакета програм оптимізації.

Розглянемо альтернативну адаптацію в процесах пошукової оптимізації як приклад структурної адаптації. Проблема пошукової оптимізації виникла у зв'язку з необхідністю розв'язання складних задач математичного програмування:

$$Q(X) \rightarrow \min_{x \in \Omega} Q(X) \quad (11)$$

виникаючих при алгоритмізації процесів управління, прийнятті оптимальних рішень тощо. Тут  $Q(X)$  - скалярна функція якості векторного аргументу  $X = (x_1, \dots, x_n)$ , визначеного в області  $\Omega$ . Ця область може бути або континуальною і задаватися системою рівностей і нерівностей:

$$\Omega: \begin{cases} g_i(X) = 0 & (i = 1, \dots, l < n) \\ h_j(X) \geq 0 & (j = 1, \dots, p) \end{cases} \quad (12)$$

або кінцевою множиною, тобто визначати дискретні значення аргументу:

$$\Omega: \{X_i\} \quad (i = 1, \dots) \quad (13)$$

або поєднанням цих обмежень. Можливі й інші форми завдання області  $\Omega$ .

Складність розв'язку (8) для більш-менш реальних задач полягає насамперед у тому, що функція  $Q(X)$  і область  $\Omega$  задані не аналітично, а алгоритмічно, тобто у вигляді будь-якого роду правил, інструкцій і вказівок, що мають як формальний, так і неформальний (експертний) характер. Ця обставина практично виключає застосування стандартних методів математичного програмування, що спираються на відому структуру і вид функції  $Q(X)$  і області  $\Omega$ .

Задача, таким чином, полягає у відшуванні хоча б одного розв'язку  $X^*$ , що задовольняє очевидній умові

$$Q(X^*) = \min_{X \in \Omega} Q(X) \quad (14)$$

Суть пошукового методу відшукування

$$X^* \text{ зводиться до побудови послідовності} \\ X_0, X_1, \dots, X_N, \dots \quad (15)$$

яка повинна сходиться до досить малої околиці розв'язку  $X^*$ . Алгоритм пошуку  $F$  пов'язує один за одним наступні стани:

$$X_{N+1} = F(X_N, W_N) \quad (16)$$

де  $W_N$  - фактор передісторії, наприклад  $W_N = \{X_{N-1}, \dots, X_{N+l}\}$ ;  $l$  -глибина передісторії. Однак зручніше алгоритм  $F$  визначати для зміщення

$$\Delta X_{N+1} = F(X_N, W_N) \quad (17)$$

зв'язує два сусідніх стану  $X_{N+1} = X_N + \Delta X_{N+1}$  в просторі параметрів, що оптимізуються  $\{X\}$ .

Очевидно, що алгоритм  $F$  повинен задовольняти певним вимогам, що пред'являються до процесу пошуку розв'язку задачі. Ці вимоги можна представити у вигляді набору критеріїв, що характеризують ефективність процесу пошуку:

$$K = (k_1, \dots, k_m) \quad (18)$$

Кожен з критеріїв залежить від алгоритму  $F$  і ситуації  $L$ , що склалися в процесі оптимізації:

$$L = \langle Q(X), L \rangle \quad (19)$$

Таким чином, кожен критерій в (18) має вигляд

$$k_i = k_i(F, L) \quad (i = 1, \dots, m) \quad (20)$$

Задача вибору оптимального алгоритму зводиться тим самим до задачі оптимізації:

$$\Theta(F, L) \rightarrow \min_{F \in E} \Rightarrow F_L^* \quad (21)$$

де  $\Theta$  - критерій, що оптимізується, ефективності алгоритму  $F$ , обраний з множини (17);  $E$  - множина допустимих алгоритмів, що задовольняють іншим критеріям з (17). Розв'язання задачі (18) є

оптимальний алгоритм  $F_L^*$  залежний від ситуації  $l$ . Ввівши поняття класу ситуацій  $\{L\} = \{Q(X), L\}$  (19)

і правило оцінки критерію  $\Theta$  на цьому класі (наприклад, як максимуму або середнього на

класі), отримуємо оптимальні алгоритми пошуку.

Програмна адаптація пов'язана з маніпулюванням окремими програмами і вибором тієї, яка працює найкращим чином в сенсі заданого критерію. У експертній системі (ЕС) Н-Гомеопат відповідну роль виконує комутаційний елемент.[8, 9]

Алгоритмічно програмна адаптація реалізується у вигляді альтернативного варіанту структурної адаптації. Формалізуємо цю задачу. Нехай є альтернативні програми

$$A_1, \dots, A_m, \quad (22)$$

що розв'язують певний клас задач  $\Omega$  задач (5) у НМ. Критерій ефективності розв'язання конкретної задачі  $P_j$  за допомогою  $i$ -й програми:

$$q_{ij} = q(P_j, A_j) \quad (23)$$

Це означає, що, розв'язавши задачу, завжди можна оцінити критерій апостеріорно; апріорних міркувань звичайно не має. Задача полягає в тому, щоб для наявного потоку завдань  $P$  вибрати програму, що мінімізувала інтегральний критерій

$$Q(A) = \int q(P, A)p(P)dp \quad (24)$$

тобто вирішити задачу

$$Q(A) \rightarrow \min_{A_i (i=1, \dots, m)} \Rightarrow A^* \quad (25)$$

використовуючи тільки спостереження значень критерію (23).

Цю задачу слід вирішувати методами альтернативної адаптації, тобто так переходити від однієї програми (22) до іншої, щоб процес привів до програми  $A^*$ ,

оптимальної по обраному критерієм (24) в сформованій ситуації.

#### Список використаних джерел

1. Rutkovskaya D. Pylinsky M., L. Rutkovskyy. Neural networks, algorithms and henetycheskye nechetkye system. - M: Hotline - Telecom, 2006. - P.452. (in Russian).
2. Andon F., Balabanov A., Identification of knowledge and research in databases: approaches, models, methods and systems / / Problems of programming. -2000. - N1-2. - P.513-526. (in Russian).
3. Rutkovskaya D. Pilinsky M. Rutkowski L. Neural networks, genetic algorithms and fuzzy systems. - M: Hotline - Telecom, 2006. - 452p. (in Russian).
4. Winston P. Artificial Intelligence. - Academic Press, 1980. - 519s.
5. S. Haykin. Neural networks: a complete course. - Moscow: OOO " Williams ", 2006. - 220С. (in Russian).
6. Kruglov VV, The length MI, Golunov RY Fuzzy Logic and Artificial Neural Networks: A Handbook. allowance. - Moscow: Publishing House of Physical-Mathematical Literature, 2001. - 224.
7. Yudin D, Mathematical methods of control under incomplete information. Moscow, Sov. Radio, 1974. 400 p. . (in Russian).
8. Katerynych L., A. Provotar. Dyahnostyrovanye on neural setyah in the system Homeopath / / XIII-th International Conference: Knowledge Dialogue Solution. - Sofia, 2007. - V1. - P.64-68. (in Russian).
9. Katerynych L. Control synthesis of neural networks / / Problems of programming. 2009. Issue 1. P.53-59. (in Russian).

Надійшла до редколегії 04.02.13