

УДК 681.51

*В.А. Жеребко, В.В. Афанасьєв, А.С. Лясковський*

## РОЗРОБКА ПРОГРАМНОЇ МОДЕЛІ НЕЙРОМЕРЕЖЕВОГО ПІД-РЕГУЛЯТОРА В СЕРЕДОВИЩІ LABVIEW

У даній роботі представлена методика синтезу та моделювання нейромережевого ПІД-регулятора засобами LabVIEW для побудови адаптивної системи автоматичного керування. Як об'єкт керування було обрано двигун постійного струму. Синтез та моделювання нейромережевого ПІД-регулятора виконані за функціональними властивостями класичного ПІД-регулятора. Структура нейромережевого ПІД-регулятора – перцептрон із одним прихованим шаром нейронів, а навчання нейромережі відбувається за алгоритмом зворотного поширення помилки.

### Вступ

При побудові робототехнічних або інших складних систем керування часто виникає потреба керування об'єктом, чия математична модель дуже складна або невідома. Підхід до синтезу таких систем автоматичного керування (САК) класичними методами буває або вкрай трудомісткий або неможливий взагалі.

Класичні закони керування, наприклад, ПІД-регулювання, не здатні враховувати структурних та параметричних змін, що відбуваються в об'єкті керування (ОК) з плином часу, тому необхідно використовувати адаптивні механізми. САК [1, 2], що побудовані з використанням нейромережевих підходів [3] можна класифікувати як адаптивні САК.

Нейромережеві методи особливо актуальні для завдань керування ОК зі змінною структурою. У такому разі використання нейромережевих регуляторів здатне помітно скоротити терміни і вартість проектування САК оскільки немає необхідності вирішувати задачу ідентифікації ОК та усувати вплив невизначеності за умови відповідного навчання нейронної мережі регулятора.

Нейромережеві структури (штучні нейронні мережі) використовуються для побудови ПІД-регулятора двома шляхами [4]: для побудови власне регулятора, або для побудови блоку настроювання його коефіцієнтів. Такий ПІД-регулятор схожий на регулятор із табличним керуванням (так зване «програмне керування»), тобто виконується апроксимація функції сигналу керування нелінійним ОК.

Сучасний спектр універсальних програмних засобів, що орієнтовані на повний цикл як інженерних досліджень, так і кінцеву реалізацію спроектованих нейромережевих алгоритмів у прикладних програмах для цільових апаратних платформ не викликає захоплення. Загалом всі інструментальні середовища чітко спеціалізовані за напрямками: моделювання, проектування, тестування, симуляція тощо.

Популярним програмним пакетом для моделювання та синтезу нейромережевих систем керування є MATLAB, але певні обмеження не дозволяють використовувати його для кінцевої програмної реалізації нейромережевих алгоритмів керування на цільових платформах. Тому вибір було зупинено на пакеті LabVIEW [5]. Цей пакет дозволяє виконати всі етапи розробки від моделювання та синтезу до побудови прикладної програми на цільовій апаратній платформі.

У даній роботі запропоновано програмну реалізацію моделі нейромережевого ПІД-регулятора на цільовій апаратній платформі NI CompactRIO [6] засобами LabVIEW. Синтез регулятора та САК у цілому також виконані в середовищі LabVIEW. Розробка, синтез та реалізація програмної моделі нейромережевого ПІД-регулятора складається із окремих етапів:

1. Вибір неперервної моделі ОК.
2. Вибір структури неперервної моделі ПІД-регулятора.
3. Переведення неперервних моделей у цифрову форму.
4. Розрахунок коефіцієнтів налаштування ПІД-регулятора.

5. Вибір структури штучної нейронної мережі для побудови регулятора.

6. Генерування навчальних вибірок на базі функціонування класичного ПД-регулятора в замкненій САК із ОК.

7. Виконання алгоритму навчання нейронної мережі за базою навчальних вибірок.

8. Порівняльний аналіз реалізованої моделі нейронно-регуляторного та класичного ПД-регуляторів.

Тож розглянемо кожний із цих етапів синтезу й розробки моделі нейронно-регуляторного ПД-регулятора більш докладно.

Структурна схема САК з класичним ПД-регулятором та зі зворотним зв'язком показана на рис. 1. У ній блок *PID* називають ПД-регулятором, *P* – об'єктом керування, *r* – задавальний вплив, або уставкою, *e* – сигналом розузгодження або помилки, *u* – керуючий вплив (вихід регулятора), *y* – регульованою величиною. На САК можуть впливати зовнішні обурення *d* і шум вимірювань *n*.

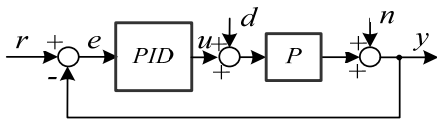


Рис. 1. Класичний ПД-регулятор в САК зі зворотним зв'язком

На рис. 2 показана типова структура САК з нейронно-регуляторним ПД-регулятором.

Нейронна мережа в даній структурі грає роль функціонального перетворювача, котрий для кожного набору сигналів *r*, *e*, *u*, у виробляє сигнал, що замінює компоненти класичного ПД-регулятора.

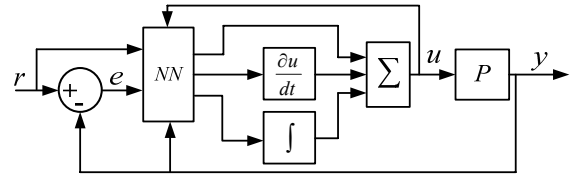


Рис. 2. САК з нейронно-регуляторним ПД-регулятором (*NN* – нейронна мережа)

### 1. Математична модель об'єкта керування

Як ОК використовується двигун постійного струму (ДПС) типу 2ПН132ЛУХЛ4 [7] з характеристиками, що наведені у табл. 1.

Математична модель динаміки ДПС описується диференціальним рівнянням:

$$(T_y \cdot T_m) \frac{d^2 \omega}{dt^2} + T_m \cdot \frac{d\omega}{dt} + \omega = k_o u, \quad (1)$$

де  $T_y$  та  $T_m$  сталі часу.

$$T_y = \frac{L_y}{R_y}, \quad T_m = \frac{J_{np} \cdot R_y}{(c_e \cdot \Phi)^2},$$

де  $J_{np}$  – сумарний момент інерції, приведений до вісі двигуна,  $c_e$  – конструктивна стала,  $\Phi$  – потік збудження ДПС.

$$c_e \cdot \Phi = \frac{U_n - I_y \cdot R_y}{\omega_n},$$

Таблиця 1. Характеристики ДПС

Назва характеристики	Позначення	Номінальне значення	Одиниці вимірювання
Номінальна потужність	$P_n$	1.9	кВт
Номінальна напруга	$U_n$	220	В
Номінальні оберти	$n_n$	750	об/хв.
Коефіцієнт корисної дії	$\eta$	76.6	%
Опір обмотки якоря	$R_y$	0.836	Ом
Опір обмотки додаткових полюсів	$R_{дп}$	0.647	Ом
Опір обмотки збудження	$R_{оз}$	148	Ом
Індуктивність якоря	$L_y$	23	мГн
Момент інерції двигуна	$J_n$	0.0836	кг/м <sup>2</sup>

де  $U_n$  – номінальна напруга,  $I_a$  – струм в обмотці якоря,  $\omega_n$  – номінальна швидкість обертання валу ДПС.

$$I_a = \frac{P_n}{\eta \cdot U_n},$$

$$\omega_n = \frac{2 \cdot \pi \cdot n_n}{60},$$

$$J_{np} = 3 \cdot J_n,$$

$$k_\delta = \frac{1}{c_e \cdot \Phi},$$

де  $k_\delta$  – коефіцієнт передачі ДПС.

Підставимо числові значення величин характеристик ДПС, що наведені в табл. 1, у відповідні формули і отримаємо диференційне рівняння ДПС (1) у наступному вигляді:

$$0.00093 \frac{d^2 \omega}{dt^2} + 0.03 \frac{d\omega}{dt} + \omega = 0.379 \cdot u. \quad (2)$$

Запишемо неперервну передатну функцію (ПФ) для ДПС, застосувавши перетворення Лапласа до диференційного рівняння (2):

$$W(p) = \frac{0.379}{0.00093 p^2 + 0.03 p + 1}.$$

Перейдемо до цифрової ПФ приведеної неперервної частини (ПБЧ) з частотою дискретизації 0.1 с, використавши команду *c2d* в пакеті MATLAB.

Наведемо команди переходу від неперервної ПФ до цифрової ПФ:

- 1) `>> W=tf(0.379,[0.00093 0.03 1]);`
- 2) `>> W(p) = \frac{0.379}{0.00093 p^2 + 0.03 p + 1};`
- 3) `>> c2d(W,0.1);`
- 4) `>> W(p) = \frac{0.4394z + 0.0996}{z^2 + 0.3824z + 0.0397}.`

## 2. Математична модель ПІД-регулятора

ПІД-регулятор (PID-regulator) видає вихідний сигнал, що є сумою трьох складових компонент: пропорційного регулювання (*P*), регулювання по інтегралу (*I*) та регулювання за похідною (*D*). Перша час-

тина  $u_p(t)$  пропорційна помилці вихідної величини, а сама різниця між вихідною величиною та задавальним впливом:  $e(t) = r(t) - y(t)$ ; друга частина  $u_i(t)$  – інтегралу за часом помилки вихідної величини, а третя частина  $u_d(t)$  – похідній помилки.

Математичне рівняння класичного ПІД-регулятора має вигляд

$$u(t) = K_p \cdot e(t) + K_i \cdot \int_0^t e(\tau) dt + K_d \cdot \frac{de(t)}{dt}.$$

Призначення ПІД-регулятора полягає в підтриманні заданого значення  $r$  деякої величини  $y$  за допомогою зміни керуючої величини  $u$ .

ПФ цифрового ПІД-регулятора має наступний вигляд:

$$W_{ПІД}(z) = K_p + \frac{K_i T_0}{2(z-1)} + \frac{K_d(z-1)}{T_0 z}, \quad (3)$$

де  $T_0$  – період дискретизації,  $K_i$ ,  $K_d$  і  $K_p$  – параметри інтегральної, диференціальної і пропорційної складових ПІД-закону регулювання  $u$ .

Звівши до спільного знаменника формулу (3), отримуємо кінцеву ПФ цифрового ПІД-регулятора:

$$\frac{(K_i T_0 + 2K_d + 2K_p T_0)z^2 + (K_i T_0^2 - 2K_p T_0 - 4K_d) + 2K_p}{2T_0 z(z-1)}.$$

Тоді ПФ розімкнутої системи з корекцією буде мати вигляд:

$$W_{ск}(z) = W_{ПІД}(z) \cdot W_{ПБЧ}(z),$$

де  $W_{ПБЧ}(z)$  – цифрова ПФ ПБЧ даної моделі САК.

## 3. Розрахунок коефіцієнтів ПІД-регулятора

**3.1. Визначення коефіцієнта інтегральної складової  $K_i$ .** Для визначення  $K_i$  використовуємо вираз для коефіцієнта помилки за швидкістю:

$$K_v = \frac{1}{T_0} \lim_{z \rightarrow 1} (z-1)W_{ск}(z). \quad (4)$$

*Зауваження.* Для того, щоб вираз (4) був справедливим, необхідно, щоб функція, яка входить під знак *lim* не мала полюсів на одиничному колі.

$K_v$  – вибираємо самостійно ( $K_v = 3 \div 8$ ). Помітимо, що  $K_v$  визначається

тільки коефіцієнтом  $K_i$  і параметрами САК та залежить від  $K_\delta$  і  $K_p$ . Виберемо  $K_v = 5$ .

$$K_v = \frac{1}{T_0} \lim_{z \rightarrow 1} (z-1)W_{ck}(z) = K_i \cdot W_{пбч}(z-1) = 59.191K_i \rightarrow K_i = 0.094.$$

**3.2. Визначення коефіцієнтів  $K_\delta$  і  $K_p$ .** Вибираючи два нулі регулятора так, щоб вони компенсували полюси САК, отримуємо:

$$z^2 + \frac{K_i T_0^2 - 2K_p T_0 - 4K_\delta}{K_i T_0^2 + 2K_\delta + 2K_p T_0} z + \frac{2K_\delta}{K_i T_0^2 + 2K_\delta + 2K_p T_0} = (z-z_1)(z-z_2) = z^2 - z(z_1+z_2) + z_1 z_2.$$

Прирівнюючи коефіцієнти при однакових степенях  $z$ , отримуємо систему рівнянь, з якої знаходимо  $K_\delta$  і  $K_p$ .

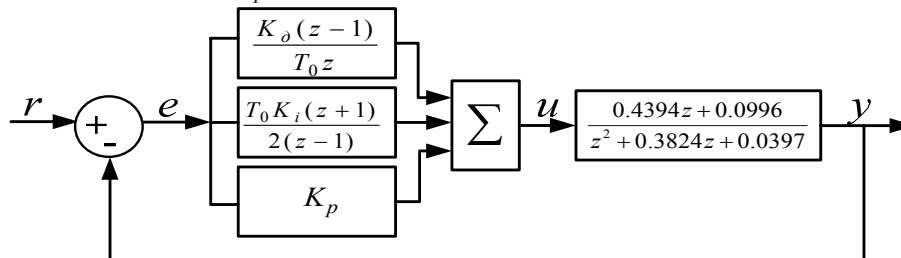


Рис. 3. Модель САК з цифровим ПІД-регулятором

#### 4. Вибір структури нейромережевого ПІД-регулятора

В якості структури нейромережевого ПІД-регулятора була вибрана математична структура типу перцептрон [8].

Багатошарові перцептрони успішно застосовуються для вирішення різноманітних складних завдань. Багатошарові перцептрони мають три відмінні ознаки:

- кожен нейрон нейронної мережі має нелінійну функцію активації (nonlinear activation). Важливо підкреслити, що дана нелінійна функція є гладкою (тобто всюди диференційованою), на відміну від жорсткої порогової функції, використовуваної у перцептроні Розенблатта;

- нейромережа містить один або кілька шарів прихованих нейронів, які не є частиною входу або виходу мережі. Ці нейрони дозволяють мережі навчатися рішенням складних завдань, послідовно

$$\begin{cases} \frac{K_i T_0^2 - 2K_p T_0 - 4K_\delta}{K_i T_0^2 + 2K_\delta + 2K_p T_0} = 0.3824 \\ \frac{2K_\delta}{K_i T_0^2 + 2K_\delta + 2K_p T_0} = 0.03972 \end{cases} \Leftrightarrow \begin{cases} \frac{0.1319 - 0.2K_p - 4K_\delta}{0.1319 + 2K_\delta + 0.2K_p} = -1.844 \\ \frac{2K_\delta}{0.1319 + 2K_\delta + 0.2K_p} = 0.85 \end{cases}$$

Розв'язавши систему рівнянь, отримуємо коефіцієнти ПІД-регулятора:  $K_\delta = 0.2022$ ,  $K_p = 0.00157$ . Остаточний вигляд моделі САК з ПІД-регулятором показано на рис. 3.

витагуючи найбільш важливі ознаки з вхідного образу (вектора);

- нейромережа має високий ступінь зв'язності, реалізованої за допомогою синаптичних з'єднань. Зміна рівня зв'язності мережі вимагає зміну безлічі синаптичних з'єднань або їхніх вагових коефіцієнтів. Комбінація всіх цих властивостей разом із здатністю до навчання на власному досвіді забезпечує обчислювальну потужність багатошарового перцептрона.

Найпопулярнішою формою функції, що задовольняє цій вимозі, є сигмоїдальна (sigmoidal nonlinearity), обумовлена наступною логістичною функцією.

$$y(x) = \frac{1}{1 + e^{-x}} \quad (5)$$

Схема перцептрона, який був обраний для вирішення нашої задачі показаний на рис. 4, що складається з 4 входів ( $r, e, u, y$ ), 3 виходів (компоненти  $P, I, D$ ) та одного внутрішнього шару нейронів  $n_1..n_N$ , де  $N$  - кількість нейронів внутрішнього шару

неймережі;  $w_{ij}$  – масив ваг, що з'єднує вхідні сигнали та внутрішній шар нейронів,  $w_{jk}$  – масив ваг, що з'єднує внутрішній шар нейронів з виходами мережі. Значення  $i$  змінюється в межах від 1 до 4,  $k$  – від 1 до 3, а  $j$  – від 1 до  $N$ .

Щоб визначитися з кількістю нейронів внутрішнього шару було проведено декілька дослідів і вибрано  $N=200$ .

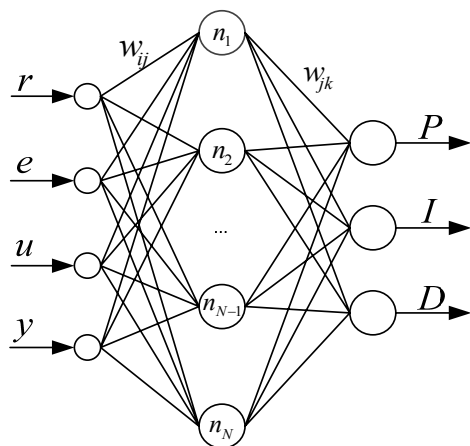


Рис. 4. Структурна схема нейронної мережі, що використовується як неймережевий ПІД-регулятор

За результатами дослідів було виявлено, що при меншій кількості нейронів, навчання нейронної мережі проходить швидше, але результати моделювання при цьому є досить неточними, а при більшій кількості нейронів значно спадає швидкість навчання, при цьому точність практично не змінюється.

Таблиця 2. Функціональні блоки

Зовнішній вигляд	Призначення	Зовнішній вигляд	Призначення
	Вхідний сигнал		Блок запису передатної функції
	Підсилювач		Осцилограф
	Запис в масиви поточного часу симуляції та значення сигналу		Суматор
	Інтегратор		Кластер масивів поточного часу симуляції та значення сигналу
	Диференціатор		Функція для запису навчальних вибірок у файл

## 5. Генерування навчальних вибірок для неймережевого ПІД-регулятора

Генерування навчальних вибірок для неймережевого ПІД-регулятора здійснено в середовищі розробки LabVIEW з допомогою модуля Control Design & Simulation (CDS) [9]. На рис. 5 показана схема генерування навчальних вибірок. Для цього було здійснено симуляцію переходного процесу САК із цифровим ПІД-регулятором, і запису бажаних входів та виходів неймережевого ПІД-регулятора в масиви з частотою, рівною частоті дискретизації (0.1 с).

В табл. 2 наведені та описані функціональні блоки, що були використані для генерування навчальних вибірок для неймережевого ПІД-регулятора.

Значення керуючого впливу ( $u$ ) та виходу системи ( $y$ ) на початку ітерації невідомі, тому як вхідні параметри  $u$  та  $y$  беруться значення попередньої ітерації, отже для створення коректної навчальної вибірки кожен  $i$ -й елемент масивів  $u$  та  $y$  записується під номером  $i+1$ .

Оскільки в модулі CDS непередбачено функціонального блоку для цифрового диференціатора, то було використано функціональний блок *Derivative x(t) PtByPt*, який знаходиться в розділі *Signal Processing-Point By Point*.

Симуляція перехідного процесу САК була здійснена за допомогою структури *Control & Simulation Loop* з часом симуляції 40 с, з кроком 0.001 с, та з часом дискретизації рівним 0.1 с. Далі відбувається окремо конкатенація масивів

входів та виходів та об'єднання їх в єдиний кластер, який і є навчальними вибірками, що записується в файл (рис. 6) і використовується в подальшому для навчання нейронної мережі.

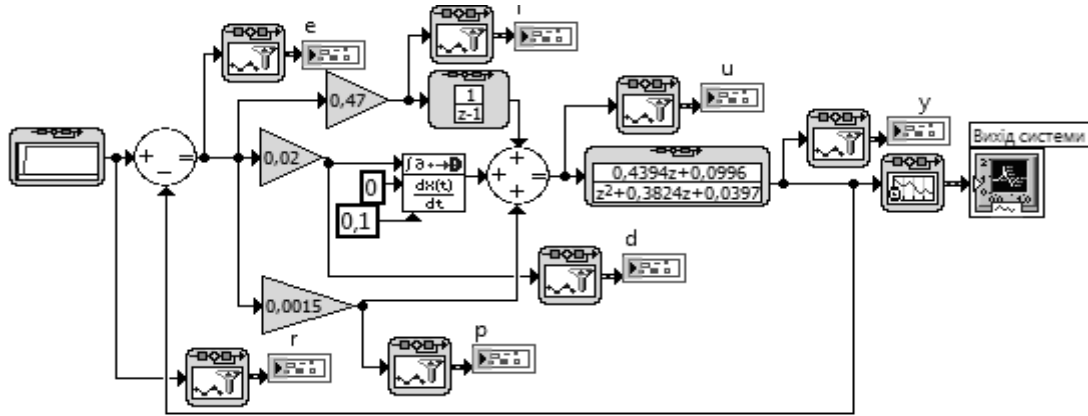


Рис. 5. Блок-діаграма генерування навчальних вибірок

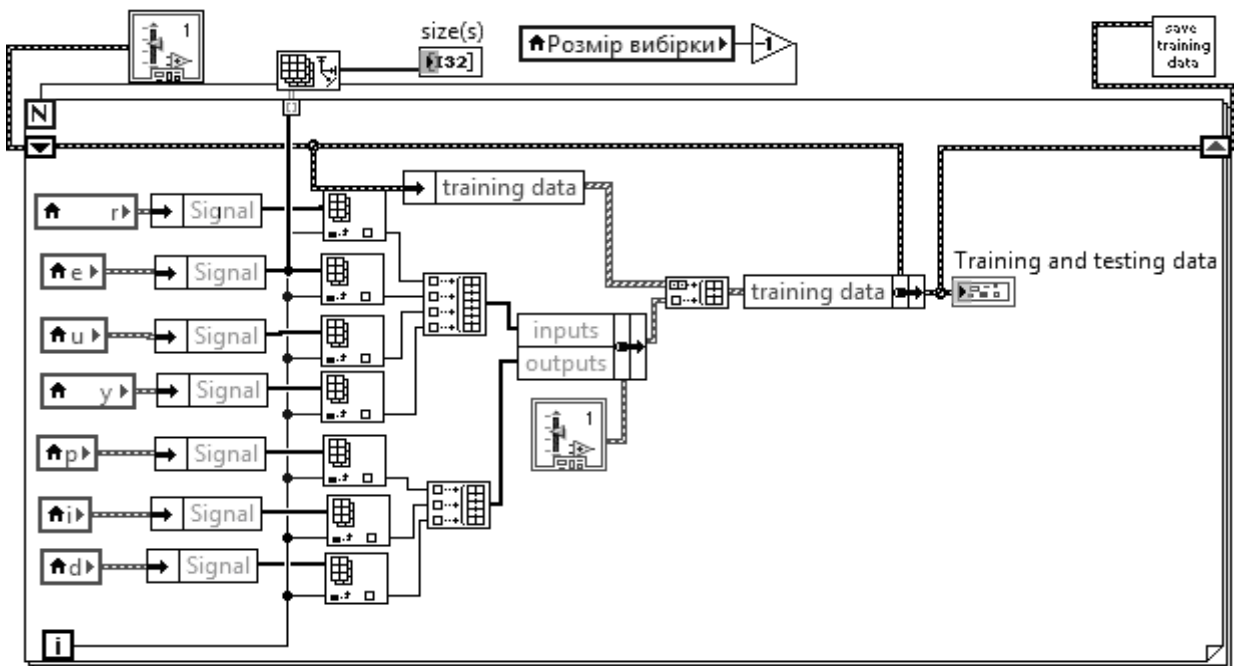


Рис. 6. Блок-діаграма збереження навчальних вибірок у XML-файл в LabVIEW

## 6. Алгоритм навчання нейронної мережі

Навчання нейронної мережі з учителем виконується за допомогою такого популярного алгоритму, як алгоритм зворотного поширення помилки (ЗПП). Цей алгоритм ґрунтується на корекції помилок виходу нейромережі. Його можна розглядати як узагальнення популярного алгоритму

адаптивної фільтрації – алгоритму мінімізації середньоквадратичної помилки. Як функцію активації було використано сигмоїд (5).

Сигнал помилки вихідного нейрона  $j$  на ітерації  $n$  визначається співвідношенням:

$$e_j(n) = d_j(n) - y_j(n),$$

де  $d_j(n)$  – бажане значення,  $y_j(n)$  – реальне.

Поточне значення енергії помилки нейрона  $j$  визначимо як  $\frac{1}{2}e_j^2(n)$ . Відповідно поточне значення  $E(n)$  загальної енергії помилки нейронної мережі визначається шляхом додавання величин  $\frac{1}{2}e_j^2(n)$  за всіма нейронами вихідного шару. Це «видимі» нейрони, для яких сигнал помилки може бути розрахований безпосередньо. Таким чином, можна записати:

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

де множина  $C$  включає всі нейрони вхідного шару.

Основними перевагами функції сигмоїд (5) є проста похідна, що використовується при реалізації алгоритму ЗПП та автоматичний контроль підсилення – для слабких сигналів крива вхід-вихід має сильний нахил, що дає більше підсилення, коли величина сигналу стає більшою, підсилення послаблюється. Таким чином великі сигнали сприймаються нейромережею

без насичення, а слабкі сигнали проходять по нейромережі без сильного ослаблення.

Основними характеристиками, які характеризують ефективність навчання нейронної мережі є «швидкість навчання» та «навчальний імпульс», які контролюють хід процесу навчання.

Для запобігання «паралічу» та нестійкості нейромережі швидкість навчання була взята невелика (0.2), навчальний імпульс взятий рівним 0.1. Середньоквадратична помилка, згідно якої мережа вважається навченою, була взята рівною 0.0001. При менших значеннях похибки суттєво збільшується тривалість процесу навчання, при менших значеннях спостерігається великий відсоток значень, що не задовольняють бажаним виходам.

На рис. 7 показана лицьова панель віртуального приладу навчання нейронної мережі. Процес навчання було здійснено в LabVIEW за допомогою спеціального набору функцій Neural Net [10] та модулю CDS [9].

На рис. 8 показана частина блок-діаграми з циклом навчання нейронної мережі реалізації ПІД-регулятора.

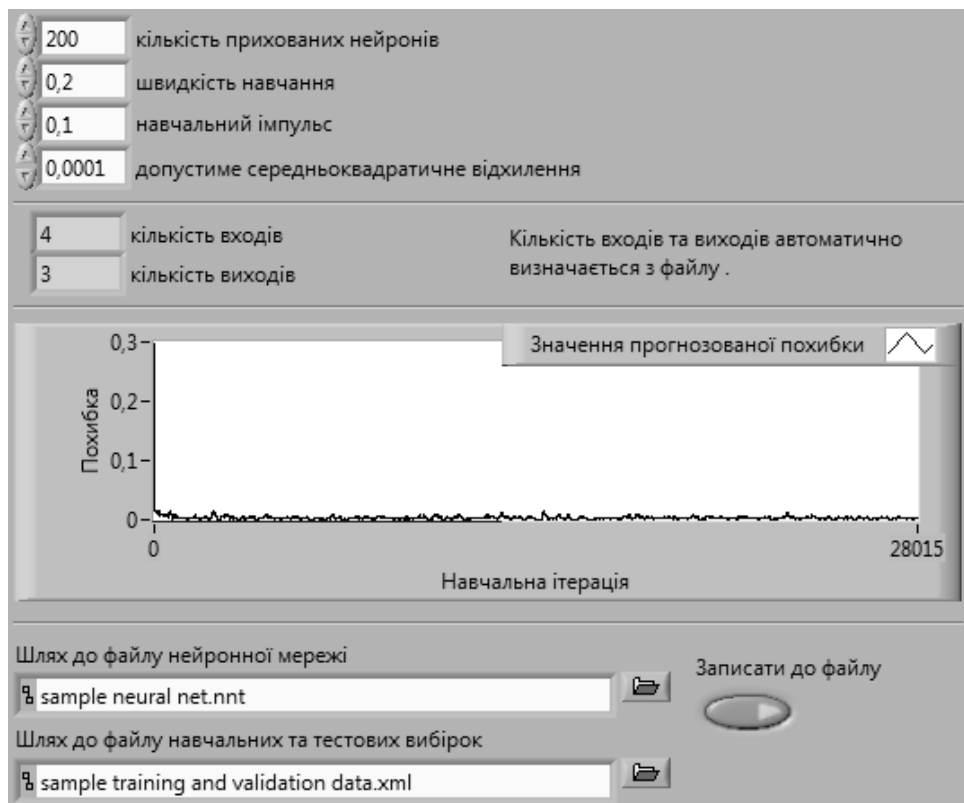


Рис. 7. Лицьова панель віртуального приладу навчання нейронної мережі в LabVIEW

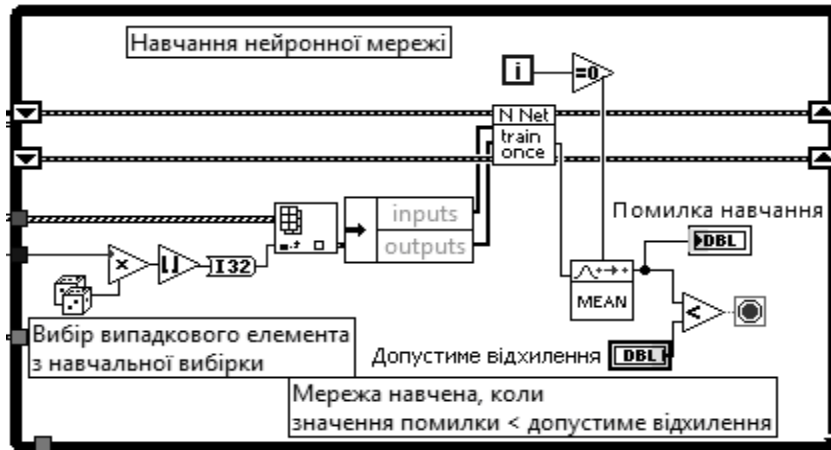


Рис. 8. Частина блок-діаграми навчання нейронної мережі в LabVIEW

При запуску віртуального приладу виконується цикл навчання нейронної мережі, який зупиняється за умов, що значення середньоквадратичної похибки стає менше за допустиме відхилення. При цьому на виході отримуємо масив скорегованих ваг, які записуються у файл нейронної мережі.

### 7. Порівняння перехідного процесу системи з нейромережовим та цифровим ПІД-регуляторами

Після навчання нейромережового ПІД-регулятора його було включено в САК замість класичного цифрового ПІД-

регулятора. Для збереження значень  $u$  та  $v$  з попередньої ітерації використано запам'ятовуючі пристрої (рис. 9). Проведено порівняння перехідних процесів системи з нейромережовим ПІД-регулятором, цифровим ПІД-регулятором та системи без регулятора. В результаті використання ПІД-регулятора помилка системи зменшилась до нуля, зникло перерегулювання, що загалом значно покращило перехідний процес системи (рис. 10). Завдяки вибраній низькій середньоквадратичній помилці при навчанні, перехідні процеси з нейромережовим ПІД-регулятором та цифровим ПІД-регулятором майже збігаються.

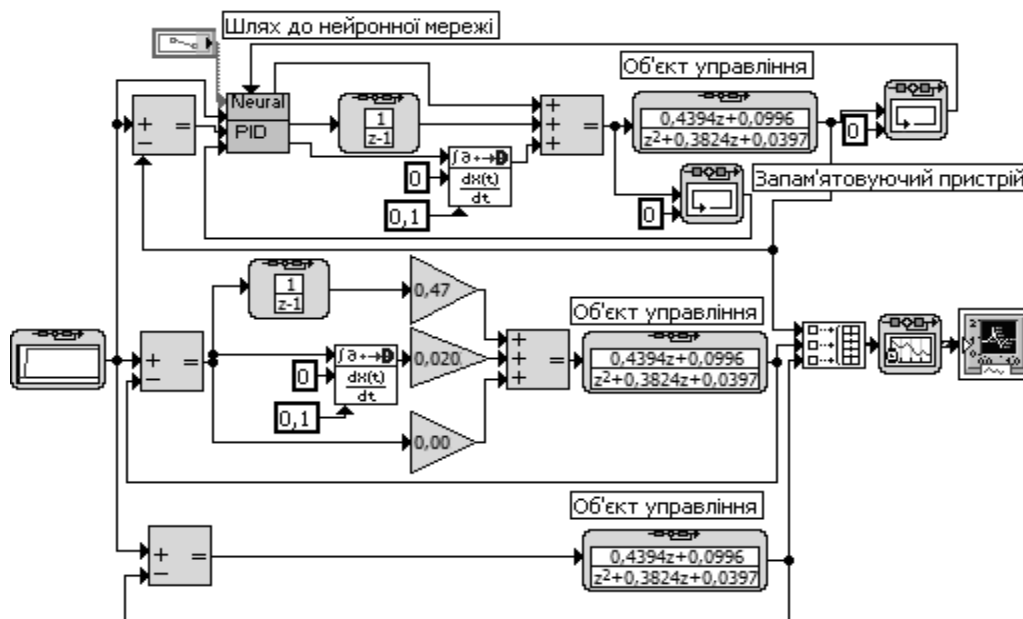


Рис. 9. Блок-діаграма порівняння перехідних процесів САК без регулятора, з цифровим ПІД-регулятором та нейромережовим ПІД-регулятором в LabVIEW



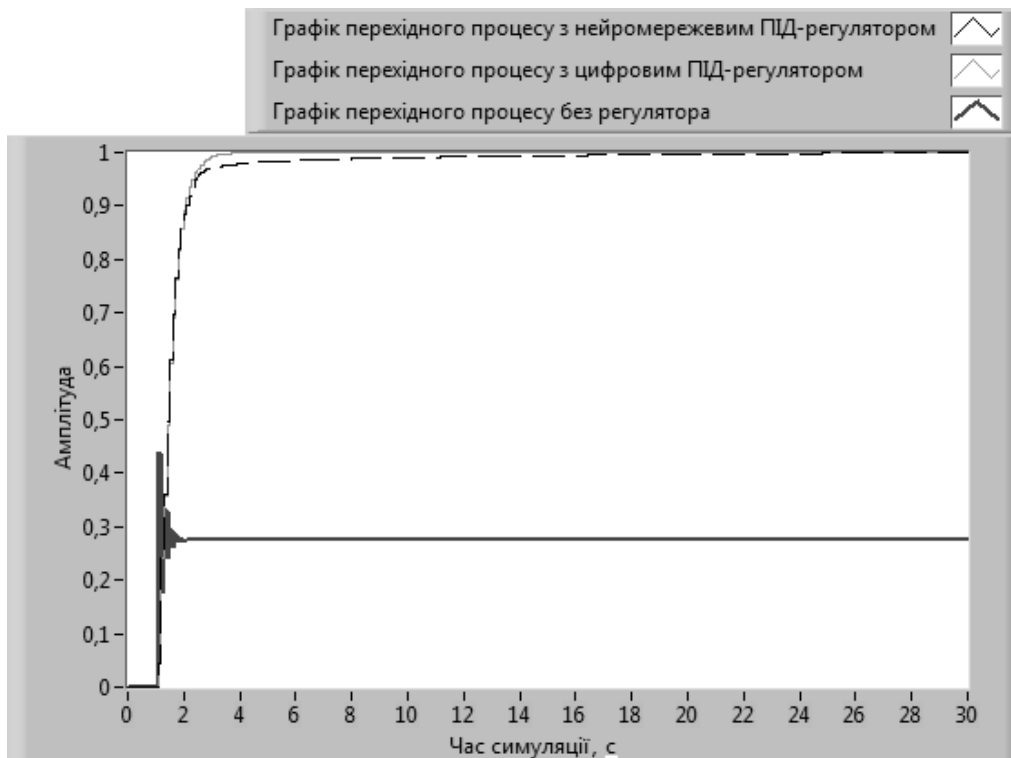


Рис. 10. Лицьова панель порівняння перехідних процесів системи без регулятора, з цифровим ПІД-регулятором та нейромережним ПІД-регулятором в LabVIEW

### Висновки

У роботі представлена покрокова методологія синтезу й реалізації нейромережевого ПІД-регулятора графічною мовою програмування G засобами LabVIEW. Були побудовані моделі ОК, та класичного ПІД-регулятора, розраховані коефіцієнти настроювання класичного ПІД-регулятора.

Для програмної реалізації нейромережевого ПІД-регулятора було обрано структуру та архітектуру штучної нейронної мережі із певним набором нейронів, шарів та активаційною функцією.

Настроювання нейронної мережі відбувалось шляхом процедури навчання за результатами роботи класичного ПІД-регулятора в САК. Завдяки вибраній низькій середньоквадратичній помилці, перехідні процеси з нейромережним ПІД-регулятором та класичним цифровим ПІД-регулятором майже збігаються.

Головні переваги нейромережевої реалізації ПІД-регулятора наступні:

- використовуючи здатність на-

- вчання на великій кількості навчальних вибірок, нейронна мережа має можливість вирішувати задачі, в яких невідомі закономірності між вхідними і вихідними даними ОК;

- можливість стійкого функціонування САК при наявності більшої кількості неінформативних сигналів шляхом адаптування до змін зовнішнього середовища;

- потенційно більша швидкодія, що досягається за рахунок організації масової квазіпаралельної обробки даних із використанням програмних технологій LabVIEW та цільової апаратної платформи CompaqтRIO на базі FPGA;

- гарантована відмовостійкість при апаратній реалізації нейромережевого ПІД-регулятора. Це означає, що при несприятливих умовах продуктивність мережі падатиме незначно.

Програмне моделювання інших інтелектуальних регуляторів в САК засобами LabVIEW є перспективним та актуальним для даного наукового напрямку.

1. Жеребко В.А., Лукіна Т.Й., Ляковський А.С. Методичне забезпечення проектування адаптивного регулятора засобами LabVIEW // Проблеми програмування. – 2009. – № 2-3. – С. 444–451.
2. Александров А. Г. Оптимальные и адаптивные системы: учеб. пособие для вузов по спец. «Автоматика и управление в технических системах». – 2-е изд. – М.: Высш. шк., 2003. – 262 с.
3. Хайкин С. Нейронные сети: полный курс. : 2-е изд. [пер. с англ.]. – М.: Издательский дом «Вильямс», 2006. – 1104 с.
4. Пилюхов В. Х., Шанин Д.А., Медведев М.Ю. Построение нейросетевых регуляторов для синтеза адаптивных систем управления // Информационно-измерительные и управляющие системы. – 2008. – № 3, 6. – С. 48–52.
5. Трэвис Дж., Кринг Дж. LabVIEW для всех – 3-е изд. – М.: ДМК Пресс, 2008. – 880 с.
6. LabVIEW и CompactRIO: основы разработки приложений // National Instruments, 2008. – 305 с.
7. Бесекерский В.А. Сборник задач по теории автоматического регулирования и управления – М.: Наука, 1978. – 587 с.
8. Галушкин А. Нейронные сети. Основы теории. – М.: Высш. шк., 2010. – 496 с.
9. NI LabVIEW Control Design Toolkit. User Manual [Електронний ресурс]. <http://www.ni.com/pdf/manuals/371057g.pdf>.
10. NI LabVIEW Neural Network Toolkit. User Manual [Електронний ресурс]. <http://www.ni.com/pdf/manuals/370765c.pdf>

Отримано 03.12.2010

### **Про авторів:**

*Жеребко Валерій Анатолійович,*  
молодший науковий співробітник,  
викладач кафедри АУТС,

*Афанасьєв Віталій Вікторович,*  
магістрант кафедри АУТС,

*Ляковський Антон Сергійович,*  
магістрант кафедри АУТС.

### **Місце роботи авторів:**

Національний технічний університет  
України «КПІ»,  
03056, Київ,

Проспект Перемоги 37,  
корпус 18, к. 522.

Тел.: +380(44) 406 8346,  
e-mail: [zherebko@kpi.ua](mailto:zherebko@kpi.ua)