

## НАВЧАННЯ РЕКУРЕНТНИХ НЕЙРОННИХ МЕРЕЖ МЕТОДОМ ПСЕВДОРЕГУЛЯРИЗАЦІЇ ДЛЯ БАГАТОКРОКОВОГО ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

---

**Анотація.** Розглядається задача багатокрокового прогнозування часових рядів. Представлені існуючі моделі рекурентних та нерекурентних нейронних мереж, що застосовуються для задач прогнозування. Описується спосіб навчання нейромереж методом розширеного фільтра Калмана з обчисленням похідних методом зворотного поширення в часі. Пропонується метод псевдорегуляризації для зменшення ефекту зникнення градієнтів, що приводить до підвищення якості прогнозування в багатокроковому випадку. Наводяться результати чисельних експериментів на прикладі прогнозування кількості сонячних плям за рік і хаотичного процесу Маккея-Гласса.

**Ключові слова:** псевдорегуляризація, рекурентні нейромережі, зворотне поширення в часі, розширений фільтр Калмана.

**Аннотация.** Рассматривается задача многошагового прогнозирования временных рядов. Представлены существующие модели рекуррентных и нерекуррентных нейронных сетей, которые применяются для задач прогнозирования. Описывается способ обучения нейросетей методом расширенного фильтра Калмана с вычислением производных методом обратного распространения во времени. Предлагается метод псевдорегуляризации для уменьшения эффекта исчезновения градиентов, что приводит к повышению качества прогнозирования в многошаговом случае. Приводятся результаты численных экспериментов на примере прогнозирования годового количества солнечных пятен и хаотического процесса Маккея-Гласса.

**Ключевые слова:** псевдорегуляризація, рекуррентные нейронные сети, обратное распространение во времени, расширенный фильтр Калмана.

**Abstract.** The problem of time series multi-step prediction is considered. The existing models of recurrent and nonrecurrent neural networks which are used for prediction problems were represented. Efficient training method of neural networks using Extended Kalman Filter and Backpropagation Through Time technique of dynamic derivatives calculation is described. Pseudoregularization method for vanishing gradients effect reducing that leads to prediction accuracy in multi-step case improvement is proposed. The results of numerical experiments on the example of yearly sunspots number and Chaotic Mackey-Glass Time Series predicting are presented.

**Keywords:** Pseudoregularization, Recurrent Neural Networks, Backpropagation Through Time, Extended Kalman Filter.

### 1. Вступ

Прогнозування часових рядів є актуальною науковою проблемою, що має безліч застосувань у теорії управління, в економіці, медицині, фізиці та інших галузях. Нейромережеві методи добре себе зарекомендували як засіб моделювання динамічних систем при невідомій апріорі математичній моделі динамічної системи [1]. Нейронна мережа може бути навчена на відомих прикладах реалізацій динамічного процесу і потім використовуватися для прогнозування на нових даних.

Існують два базових підходи для надання нейронним мережам на базі багатопереданих перцептронів властивостей, необхідних для обробки динамічних даних: додавання ліній затримок на вхід мережі та додавання рекурентних зв'язків до внутрішньої структури нейронної мережі. В першому випадку, при використанні «методу часового вікна» (Time Delay Neural Networks, Time-Window Neural Networks, Finite Impulse Response Neural Networks, Time-Lagged Neural Networks, Focused Multilayer Perceptrons, Dynamic Multilayer Perceptrons) [1–3], нейронна мережа отримує на вхід разом з поточним вхідним сигналом

затримані в часі минулі значення вхідних сигналів. Навчання нейромережі виконується з обчисленням похідних за відомим методом зворотного поширення (Backpropagation, BP) і з застосуванням градієнтного методу оптимізації першого або другого порядку. Плюсами цього підходу є його простота й технологічність: за нашими оцінками, зараз більш ніж в 90% випадків для прогнозування часових рядів використовується саме така схема. З іншого боку, при застосуванні такої схеми кількість і порядок затримок часового вікна мають бути встановлені апіорі. Якщо ці величини не будуть відповідати порядку динамічного процесу, нейромережа буде погано навчатися або зовсім не навчиться. Іншим мінусом цього підходу є погана якість багатокрокових прогнозів, коли як вхідні дані для прогнозу на новий крок використовуються власні прогнозні дані нейромережі, отримані для попередніх кроків.

Другим базовим підходом введення динаміки в нейромережі прямого поширення є додавання внутрішніх рекурентних зв'язків у приховані, вхідні або вихідні шари нейромережі (Recurrent Multilayered Perceptrons, Simple Recurrent Neural Networks, NARX Neural Networks) [1, 3]. Для визначення впливу минулих тактів на поточний результат вводиться обчислення спеціальних динамічних похідних. Динамічні похідні належать до одного з двох видів за способом обчислення: «рекурентне навчання в реальному часі» (Real-Time Recurrent Learning, RTRL) [1, 4] або «зворотне поширення в часі» (Backpropagation Through Time, BPTT) [1, 4, 5]. Як оптимізаційний метод для налаштування ваг нейромережі може використовуватися, як і в першому випадку, будь-який градієнтний оптимізаційний алгоритм. Такі рекурентні мережі за своєю структурою більш відповідають динамічним процесам, що моделюються, і тому краще показують себе в задачах керування і багатокрокового прогнозування. Разом з тим навчання таких мереж є більш важким завданням через додаткові ступені вільності в таких мережах, на практиці процес навчання часто не сходиться. Крім того, при розрахунках динамічних похідних у перцептронно-подібних мережах має місце ефект зникнення градієнта (gradient vanishing) [1, 6], що додатково ускладнює для алгоритму навчання виявлення кореляцій між минулими входами і поточними цільовими виходами мережі.

Ми пропонуємо оригінальний підхід для навчання рекурентних нейромереж шляхом накладення додаткових обмежень на величину модуля градієнта при зворотному поширенні в часі в цільову функцію оптимізації навчання нейромережі. Як функція оптимізації використовується метод розширеного фільтра Калмана. Проводиться експериментальне порівняння запропонованого методу навчання зі звичайним методом навчання зворотного поширення в часі BPTT(h) для рекурентних нейромереж, а також багат шарових перцептронів з лінією затримок на прикладі багатокрокового прогнозування кількості плям на сонці за рік (числа Вольфа) [7], а також хаотичного процесу Маккея-Гласса [7–8].

## 2. Багат шаровий перцептрон з лінією затримок

Схема багат шарового перцептрона з лінією затримок порядку  $N$  показана на рис. 1. Перцептрон містить нейрони з лінійною функцією активації для нейронів вхідного шару і сигмоїдальної функції активації для нейронів прихованого і вихідного шарів (у наших експериментах ми використовували функції активації гіперболічного тангенса). Вагові значення між шарами нейронів задаються матрицями  $W^{(1)}$  й  $W^{(2)}$ . Перцептрон отримує на вхід  $x(k)$  поточне значення часового ряду  $y(k)$ , а також затримані минулі значення  $y(k-1)$ ,  $y(k-2)$ , ...,  $y(k-N)$ , які реалізуються за допомогою елементів затримки  $z^{-1}$ ,  $z^{-2}$ , ...,  $z^{-N}$ , і за цими даними навчається робити прогноз наступного значення  $\hat{y}(k+1)$ .

Розрахунок вихідного значення нейромережі  $\hat{y}$  виконується за формулою

$$\tilde{y} = g\left(\sum_j w_j^{(2)} f\left(\sum_i w_{ji}^{(1)} x_i\right)\right), \quad (1)$$

де  $w^{(1)}$  – ваги нейронів прихованого шару,  $f(\cdot)$  – активаційні функції нейронів прихованого шару,  $w^{(2)}$  – ваги нейронів вихідного шару,  $g(\cdot)$  – активаційні функції нейронів вихідного шару.

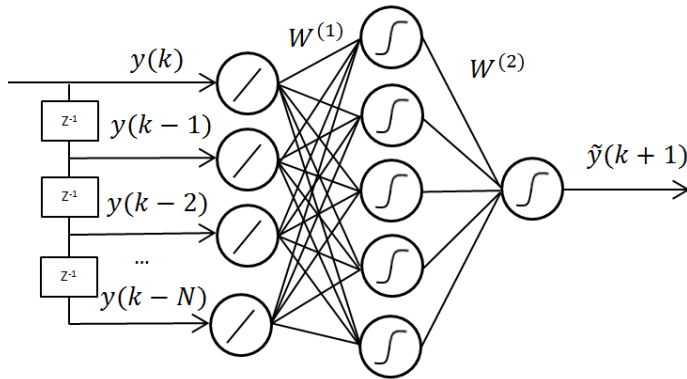


Рис. 1. Схема багатозарового перцептрона з лінією затримок на вході

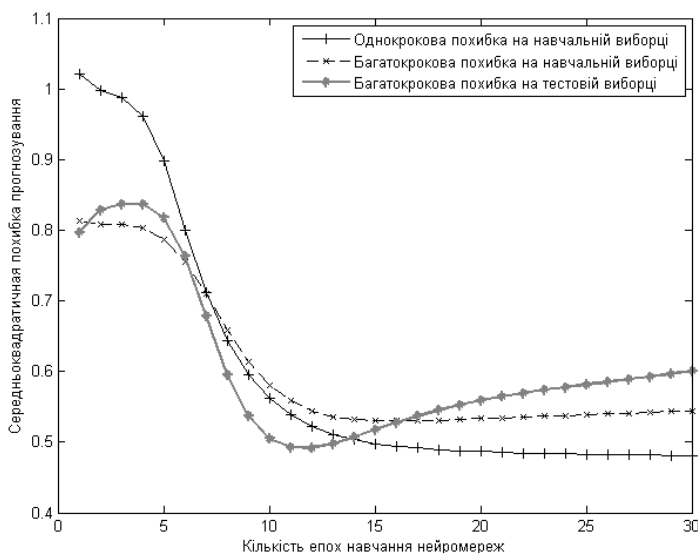


Рис. 2. Залежність якості однокрокового та багатокрокового прогнозування від кількості епох навчання

На рис. 1 показано нейромережу з п'ятьма нейронами у прихованому шарі, який отримує на вхід стан порядку  $N$ . Похідні для навчання обчислюються звичайним (статичним) методом зворотного поширення. В нашій роботі як градієнтний метод оптимізації ми використовували метод глобального розширеного фільтра Калмана (Global Extended Kalman Filter, GEKF) [1, 8–9], тому, замість

градієнтів  $\frac{\partial [e(k)^2]}{\partial w}$ , ми обчислюємо якобіани  $\frac{\partial \tilde{y}}{\partial w}$ . Це робиться шляхом пропускання значення 1 на кожному зворотному проході замість пропускання поточної помилки навчання  $e(k)$ , що приводить до отримання якобіанів замість градієнтів при тих же обчисленнях, оскільки

$$\frac{\partial [e(k)^2]}{\partial w} = 2e(k) \frac{\partial y}{\partial w}.$$

Для виконання прогнозування на  $H$  кроків вперед проводиться така ітеративна процедура:

1. Перцептрон виконує прогноз на один крок вперед, отримує оцінку наступної величини часового ряду  $\hat{y}(k+1)$ .

2. На основі наявних реальних минулих входів і отриманої оцінки формується новий вхідний вектор для нейромережі

$$x(k+1) = [\hat{y}(k+1) \quad y(k) \quad \dots \quad y(k-N+1)]^T.$$

3. Прогнозується нове значення  $\hat{y}(k+2)$ .

Кроки 2 – 3 виконуються, поки не буде отримана необхідна оцінка  $\hat{y}(k+H)$ .

На рис. 2 показано результати навчання багатозарового перцептрона з лінією затримок задачі прогнозування на один крок вперед на процесі Маккея-Гласса. Після кожної епохи навчання нейромережа також тестувалася на здатність виконувати прогноз на  $H = 14$  кроків вперед.

Як впливає з графіків на рис. 2, якість однокрокового прогнозування прямо не пов'язана з якістю багатокрокового прогнозування. При збільшенні кількості епох навчання

ня якість однокрокового прогнозування монотонно зростає, а якість багатокрокового прогнозування має певний локальний мінімум.

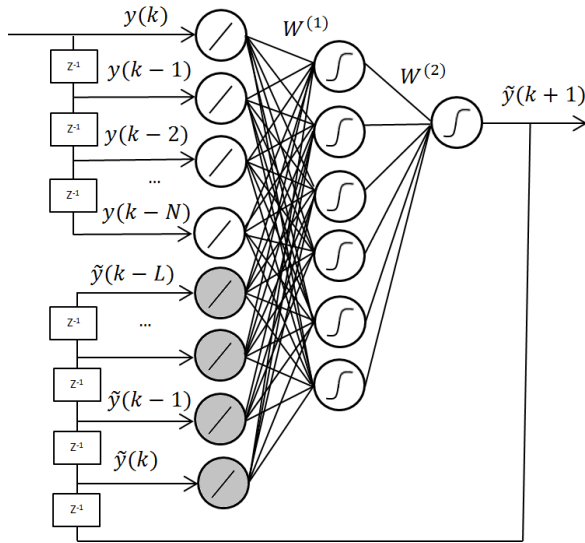


Рис. 3. Схема рекурентної нейромережі NARX. Сірим кольором позначені входні нейрони для рекурентних зв'язків

### 3. Рекурентна нейронна мережа NARX

З метою підвищення якості багатокрокового прогнозування використовують рекурентні нейронні мережі, що мають додаткові зворотні зв'язки та за рахунок специфічного способу навчання дозволяють розрахувати вплив власних похибок нейромережі на якість багатокрокового прогнозу.

Рекурентна нейронна мережа NARX (Nonlinear AutoRegression with eXternal inputs) [1, 3] (рис. 3) є модифікацією багат шарового перцептрона з додаванням зворотних зв'язків до входного шару нейронів, реалізованих затримками  $z^{-1}$ . Розрахунки вихідного значення мережі  $\hat{y}$  виконуються аналогічно (1), відмінність полягає тільки в тому, що в мат-

риці ваг прихованого шару  $W^{(1)}$  тепер також зберігаються ваги для рекурентних зв'язків.

Головною відмінністю в навчанні рекурентної нейромережі NARX від багат шарового перцептрона з лінією затримок на вході є використання динамічних похідних, що в нашому випадку розраховуються методом усіченого зворотного поширення в часі ВРТТ(h) замість статичних похідних, як для навчання багат шарового перцептрона з лінією затримок. Схема обчислення похідних під час зворотного проходу представлена на рис. 4. На такті  $k$ , після обчислення вихідного значення  $\hat{y}(k+1)$ , мережа розгортається назад у часі на глибину усікання  $h$ .

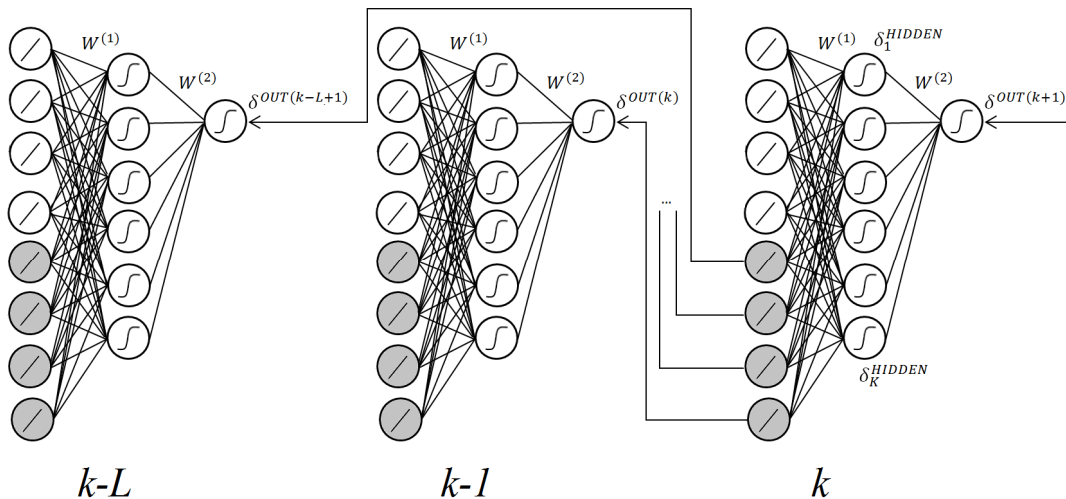


Рис. 4. Схема обчислення динамічних похідних методом зворотного поширення в часі для рекурентної нейромережі NARX

При цьому рекурентна мережа може бути представлена у вигляді багат шарового перцептрона прямого поширення з більшою кількістю шарів, де кожний шар відповідає одному з минулих тактів  $k-1, k-2, \dots, k-h$ . Похідні обчислюються методом зворотного

поширення, який застосовується для отриманої розгорнутої в часі нейромережі прямого розповсюдження. Локальні градієнти, що обчислюються у процесі процедури зворотного поширення, визначаються такими співвідношеннями:

$$\delta_j^{HD} = f_j'(k)w_j^{(2)}\delta^{OUT(k+1)}, \quad (2)$$

$$\delta_j^{IN} = f_j'(k-1)\sum_{i=1}^K w_{ij}^{(1)}\delta_i^{HD}, \quad (3)$$

$$\delta^{OUT(k-l+1)} = \delta_{N+l}^{IN}, \quad (4)$$

де  $w^{(1)}$  – ваги нейронів прихованого шару,  $f_j(k)$  – активаційна функція  $j$ -го нейрона прихованого шару на такті  $k$ ,  $w^{(2)}$  – ваги нейронів вихідного шару,  $\delta_j^{HD}$  – локальний градієнт для  $j$ -го нейрона прихованого шару,  $\delta_j^{IN}$  – локальний градієнт для  $j$ -го нейрона вхідного шару,  $1 \leq l \leq L$ ,  $L$  – порядок лінії затримки для рекурентних зв'язків,  $\delta_j^{OUT(n)}$  – локальний градієнт вихідного шару для  $n$ -го кроку розгортання у часі назад.

#### 4. Навчання нейромереж методом розширеного фільтра Калмана з псевдорегуляризацією

При обчисленні динамічних похідних методом ВРТТ(h) для персептроно-подібних нейромереж значну роль відіграє ефект зникнення градієнта, що погіршує якість навчання нейромереж для моделювання довготривалих залежностей. Зміст цього ефекту можна інтуїтивно зрозуміти, глянувши на формули обчислення локальних градієнтів (2)–(4). Абсолютне значення кожного нового локального градієнта  $\delta_j^{IN}$  на практиці виходить менше, ніж попереднього градієнта  $\delta_i^{HD}$ , оскільки воно є добутком функцій по модулю менших числа 1: початкові градієнти  $\delta^{OUT}$  рівні або менші за 1, ваги нейронної мережі  $w_{ij}^{(1)}$  й  $w_i^{(2)}$  не можуть бути більше 1, оскільки це приводить до випадку перенавчання, похідні активаційних функцій  $f(\cdot)$  завжди менше 1. Більш строгий доказ ефекту зникнення градієнта і дослідження на цю тему містяться в [6].

У нашій роботі ми пропонуємо метод навчання нейромереж з оптимізацією локальних градієнтів, названий нами «псевдорегуляризація». Суть псевдорегуляризації полягає в мінімізації у процесі навчання нейромережі функціонала, що містить в собі функцію від величин локальних градієнтів, наприклад, типу  $Q(w)$ :

$$Q(w) = e(w) + \lambda \sum_{j=1}^K (1 - \delta_j^{IN}(w))^2, \quad (5)$$

де  $e(w)$  – середньоквадратична похибка моделювання динамічного процесу,  $\delta_j^{IN}$  – локальні градієнти (3),  $\lambda$  – задана апіорі константа псевдорегуляризації. Нашою метою є навчання нейромереж динаміці деякого процесу з максимізацією абсолютних значень градієнтів вхідних нейронів у середньому:

$$\Omega = \frac{1}{N_D} \sum_{j=1}^K (\delta_j^{IN}(k))^2. \quad (6)$$

При цьому враховується, що в реальних системах, внаслідок ефекту зникнення градієнта,  $\delta(w) \ll 1$ . У формулі (6)  $K$  – кількість нейронів у прихованому шарі,  $N_D$  – кіль-

кість елементів навчальної вибірки. На інші локальні градієнти (4) обмеження, аналогічні другому доданку (5), не накладаються, оскільки нам невідомий ступінь впливу минулих тактів роботи системи на поточний такт. Проте накладання таких обмежень можливо: якщо це зробити, воно буде за змістом аналогічно апріорному визначенню часових затримок у багатошарових мережах прямого поширення з лінією затримок на вході.

Другий доданок функціонала (5) може бути мінімізований з використанням звичайних градієнтних методів. Для цього обчислимо похідні квадрата локального градієнта (2) по вагах прихованих і вихідного шарів. Враховуючи, що  $\delta_j^{HID}(k)$  є функцією тільки від ваг вихідного шару  $w^{(2)}$ ,

$$\frac{\partial}{\partial w_{\alpha\beta}^{(1)}} (\delta_j^{HID}(k))^2 = 0, \quad (7)$$

$$\begin{cases} \frac{\partial}{\partial w_{\alpha}^{(2)}} (\delta_j^{HID}(k))^2 = 2\delta_j^{HID}(k) f_j'(k-1) w_{\alpha}^{(2)} \delta^{OUT}, & j = \alpha, \\ \frac{\partial}{\partial w_{\alpha}^{(2)}} (\delta_j^{HID}(k))^2 = 0, & j \neq \alpha. \end{cases} \quad (8)$$

Аналогічно, обчислимо похідні для квадрата локального градієнта (3) по вагах прихованого і вихідного шарів:

$$\begin{cases} \frac{\partial}{\partial w_{\alpha\beta}^{(1)}} (\delta_j^{IN}(k))^2 = 2\delta_j^{IN}(k) f_j'(k-1) \delta_{\alpha}^{HID}, & j = \beta, \\ \frac{\partial}{\partial w_{\alpha\beta}^{(1)}} (\delta_j^{IN}(k))^2 = 0, & j \neq \beta, \end{cases} \quad (9)$$

$$\frac{\partial}{\partial w_{\alpha}^{(2)}} (\delta_j^{IN}(k))^2 = 2\delta_j^{IN}(k) f_j'(k-1) \sum_{i=1}^K w_{ij}^{(1)} \frac{\partial}{\partial w_{\alpha}^{(2)}} (\delta_i^{HID}(k)). \quad (10)$$

## 5. Навчання нейромереж методом розширеного фільтра Калмана

Фільтр Калмана є ефективним рекурсивним фільтром, який за рядом зашумлених і неповних вимірів дозволяє оцінити внутрішній стан динамічної системи. Він застосовується в широкому спектрі технічних пристроїв, від спідометрів автомобіля до радіоприймачів і радарів. Навчання нейронних мереж є досить несподіваним застосуванням теорії фільтрації Калмана [1, 4, 9] і, разом з тим, дуже ефективним: з одного боку, якість такого навчання перебуває на рівні кращих пакетних алгоритмів другого порядку, таких як метод Левенберга-Марквардта або квазі-ньютонівських методів [8], а з іншого – навчання проводиться в режимі он-лайн, що є актуальним у випадку вибірок даних великого обсягу і задач керування.

Навчання нейронної мережі методом розширеного фільтра Калмана розглядається як спроба виконання оцінки дійсного стану деякої невідомої «ідеальної» нейромережі, що забезпечує нульову нев'язку. Під станами в цьому випадку розглядаються значення ваг нейромережі  $w(k)$ , а під нев'язкою – поточна помилка навчання  $e(k)$ . Цей динамічний процес навчання може бути описаний парою рівнянь у просторі станів (11) і (12). Рівняння стану (11) являють собою модель процесу, що представляє собою еволюцію вектора ваг під впливом випадкового процесу  $\xi(k)$ , який вважається білим шумом з нульовим математичним очікуванням і відомою діагональною коваріаційною матрицею  $Q$ :

$$w(k+1) = w(k) + \xi(k). \quad (11)$$

Рівняння виходу (12) являє собою лінеаризовану модель нейромережі (1) на такті  $k$ , зашумлену випадковим процесом  $\zeta(k)$ , який вважається білим шумом з нульовим математичним очікуванням, і відомою діагональною коваріаційною матрицею  $R$ :

$$h(k) = \frac{\partial y(w(k), v(k), x(k))}{\partial w} + \zeta(k), \quad (12)$$

де  $w(k)$  – ваги нейромережі,  $v(k)$  – постсинаптичні потенціали нейронів,  $x(k)$  – вхідні значення мережі. Обчислення миттєвих значень похідних  $\frac{\partial y}{\partial w}$  проводиться методом зворотного поширення у випадку багатосарового персептрона з лінією затримок або методом зворотного поширення в часі у випадку рекурентного персептрона. Помилка  $e(k)$  розраховується за формулою

$$e(k) = t(k) - \tilde{y}(k), \quad (13)$$

де  $t(k)$  – цільове значення для нейромережі,  $\tilde{y}(k)$  – реальний вихід нейромережі, що розраховується по (1).

Перед навчанням нейромережі виконується етап ініціалізації. Задаються коваріаційні матриці шуму вимірювань  $R = \eta I$  і динамічного шуму навчання  $Q = \mu I$ . Розмір матриці  $R$  в нашому випадку дорівнює  $L_w \times L_w$ , розмір матриці  $Q$  дорівнює  $N_w \times N_w$  відповідно, де  $L_w$  – кількість вихідних нейронів,  $N_w$  – кількість вагових коефіцієнтів нейромережі. Коефіцієнт  $\eta$  є швидкістю навчання, у нас  $\eta = 0,001$ , коефіцієнт  $\mu$  визначає шум вимірювання, у нас  $\mu = 10^{-5}$ . Також задаються одинична на етапі ініціалізації матриця коваріації  $P$ , що має розмір  $N_w \times N_w$ , і нульова матриця вимірювань  $H$ , що має розмір  $L_w \times N_w$ .

На такті  $k$  виконуються такі дії.

1) Виконується прямий прохід нейромережі, обчислюється вихід нейромережі  $\tilde{y}(k)$  згідно з формулою (1).

2) Виконується зворотний прохід нейромережі: методом зворотного поширення обчислюються якобіани  $\frac{\partial \tilde{y}}{\partial w}$ . Формується багатокритеріальна матриця спостережень  $H(k)$ :

$$H(k) = \left[ \begin{array}{cccc} \frac{\partial \tilde{y}}{\partial w_1} & \frac{\partial \tilde{y}}{\partial w_2} & \dots & \frac{\partial \tilde{y}}{\partial w_{N_w}} \end{array} \right]^T. \quad (14)$$

3) Формується матриця відхилень  $E(k)$ :

$$E(k) = [e(k)]. \quad (15)$$

4) Обчислюються нові значення ваг нейромережі  $w(k+1)$  і матриці кореляції  $P(k+1)$  за формулами (16)–(18):

$$K(k) = P(k)H(k)^T [H(k)P(k)H(k)^T + R]^{-1}, \quad (16)$$

$$P(k+1) = P(k) - K(k)H(k)P(k) + Q, \quad (17)$$

$$w(k+1) = w(k) + K(k)e(k). \quad (18)$$

Дії 1 – 4 виконуються для всіх елементів навчальної вибірки.

## 6. Навчання нейромереж з використанням псевдорегуляризації

Для максимізації величин локальних градієнтів для вхідних нейронів (6) ми пропонуємо

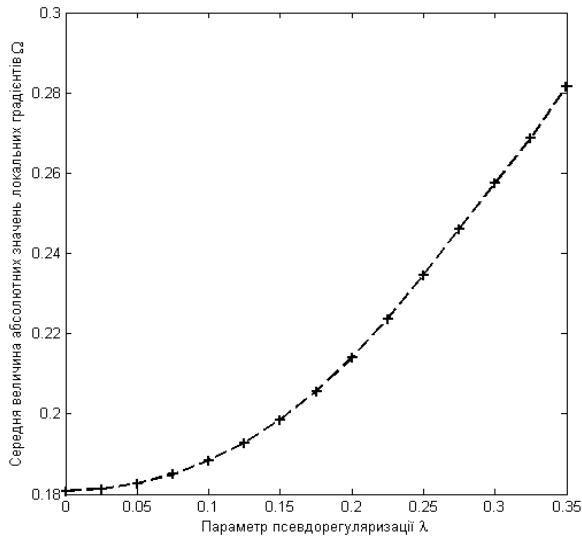


Рис. 5. Залежність величин локальних градієнтів для вхідних нейронів від параметра  $\lambda$

використовувати додатковий розширений фільтр Калмана, що працює одночасно з першим фільтром, але має іншу цільову функцію оптимізації. Будемо вважати, що виходом моделі динамічного процесу навчання на кожному такті є величини  $\delta_j^{IN}(k)^2$ ,  $j = 1, 2, \dots, L$ , які обчислюються під час виконання зворотного проходу,  $L$  – кількість рекурентних зв'язків нейромережі. Відповідно, для додаткового виходу розраховуються яacobіани  $\frac{\partial[(\delta_j^{IN}(k))^2]}{\partial w}$ ,  $j = 1, 2, \dots, L$  для формування матриці спостережень  $\tilde{H}$ , а другий доданок функціонала  $Q(w)$  формує матрицю відхилень  $\tilde{E}$ :

$$\tilde{H}(k) = \begin{bmatrix} \lambda \frac{\partial[(\delta_1^{IN})^2]}{\partial w_1} & \lambda \frac{\partial[(\delta_1^{IN})^2]}{\partial w_1} & \dots & \lambda \frac{\partial[(\delta_1^{IN})^2]}{\partial w_1} \\ \lambda \frac{\partial[(\delta_2^{IN})^2]}{\partial w_1} & \lambda \frac{\partial[(\delta_2^{IN})^2]}{\partial w_2} & \dots & \lambda \frac{\partial[(\delta_2^{IN})^2]}{\partial w_{N_w}} \\ \dots & \dots & \dots & \dots \\ \lambda \frac{\partial[(\delta_L^{IN})^2]}{\partial w_1} & \lambda \frac{\partial[(\delta_L^{IN})^2]}{\partial w_2} & \dots & \lambda \frac{\partial[(\delta_L^{IN})^2]}{\partial w_{N_w}} \end{bmatrix}^T. \quad (19)$$

$$\tilde{E}(k) = [\lambda(1 - \delta_1^{IN}(w)^2) \quad \lambda(1 - \delta_2^{IN}(w)^2) \quad \dots \quad \lambda(1 - \delta_L^{IN}(w)^2)]^T. \quad (20)$$

Корекція ваг нейромережі відбувається на кожному такті за формулами, аналогічними (16)–(18). На рис. 5 показано результати навчання рекурентних нейромереж на процесі Маккея-Гласса при різних значеннях параметра  $\lambda$ .

## 7. Послідовності для прогнозування

Для проведення експериментів було взято два відомі приклади числових рядів, що використовуються для порівняння якості прогнозування різних алгоритмів: числа Вольфа і процес Маккея-Гласса (рис. 6).

Числа Вольфа – це кількість плям на Сонці за рік, ці дані фіксувалися астрономами, починаючи з 1700 року. В наших експериментах навчальна вибірка містить дані з 1700 по 1920 рік, а тестова вибірка – з 1921 по 1979 рік, як і в [7]. Другою числовою послідовністю було взято процес Маккея-Гласса. Послідовність Маккея-Гласса задано рівнянням

$$x_{t+1} = (1 - b)x_t + a \frac{x_{t-\tau}}{1 + (x_{t-\tau})^{10}}, t = \tau, \tau + 1, \dots,$$



де  $\tau \geq 1$  – цілі числа. Для проведення експериментів була сгенерована послідовність з використанням таких значень параметрів:  $a = 0,2$ ;  $b = 0,1$ ;  $\tau = 17$ , як в [7] і [8]. Перші 1000 значень сгенерованої послідовності були проігноровані, 500 наступних значень використані як навчальна вибірка, наступні 150 – використані як текстова вибірка.

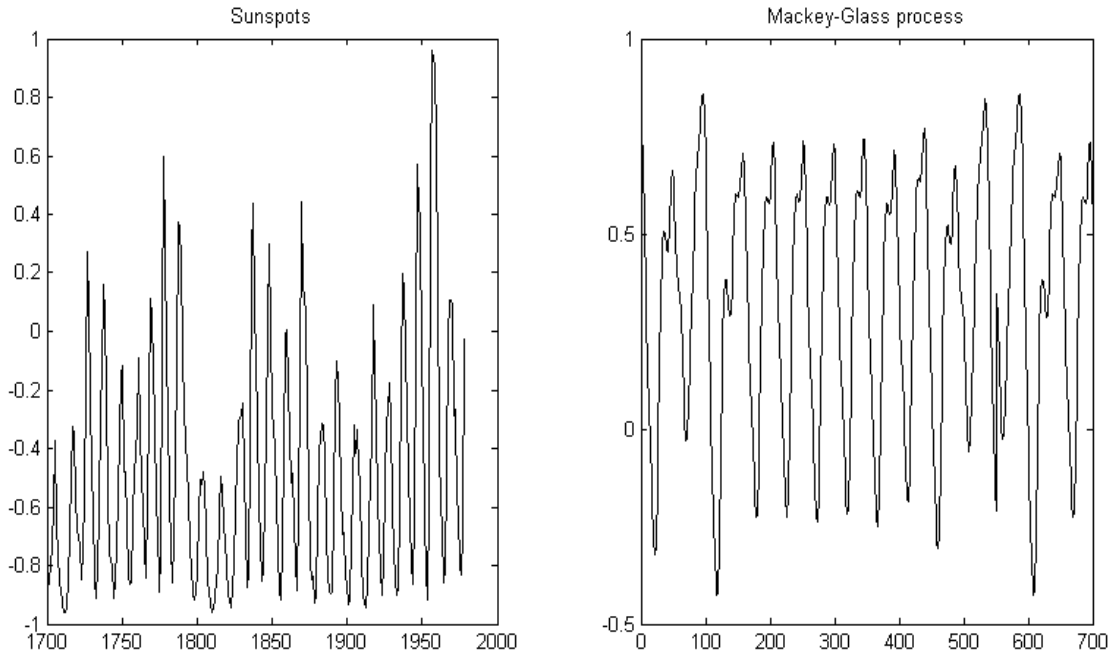


Рис. 6. Числові ряди для навчання нейромереж: числа Вольфа (справа), процес Маккея-Гласса

Якість багатокрокового прогнозування оцінювалася за критерієм середньоквадратичної помилки з обрієм,  $MSE(H)$ , за формулою

$$MSE(H) = \frac{1}{H} \sum_{k=1}^H \langle (\hat{y}_k - t_k)^2 \rangle^{\frac{1}{2}}, \quad (17)$$

де  $\hat{y}_k$  – прогнозні значення,  $t_k$  – цільові значення,  $H$  – обрій прогнозування.

## 8. Експериментальні результати

Спочатку, для вирішення задачі прогнозування, було навчено 100 багаточарових перцептронів з лінією затримок (MLP). Задача навчання для MLP ставилася фактично в такий спосіб: за  $N$  відомими минулими значеннями послідовності  $(y_k, y_{k-1}, \dots, y_{k-N})$  дати оцінку  $\hat{y}_{k+1}$  наступного значення  $y_{k+1}$ , в наших експериментах  $N=5$ . Кількість нейронів у прихованому шарі варіювалася від 3 до 8, навчання велося методом глобального розширеного фільтра Калмана. У процесі тренування мережі навчалися протягом 50 епох на навчальній послідовності і після кожної епохи перевірялися на якість багатокрокового прогнозування на навчальній вибірці. Для чисел Вольфа обрій прогнозування становив  $H = 10$ , для процесу Маккея-Гласса  $H = 14$ . Мережа, що показала кращий результат на тестовій послідовності після 50 епох, тестувалася на тестовій вибірці. Цей результат вважається остаточним і представлений у таблицях і на графіках.

Потім було навчено по 100 рекурентних нейромереж NARX, у наших експериментах  $N=5$  і  $L=5$ , з обчисленням динамічних похідних методом зворотного поширення в часі. Кількість нейронів у прихованому шарі для цих мереж коливалася від 3 до 7, навчання велося також методом глобального розширеного фільтра Калмана протягом 50 епох.

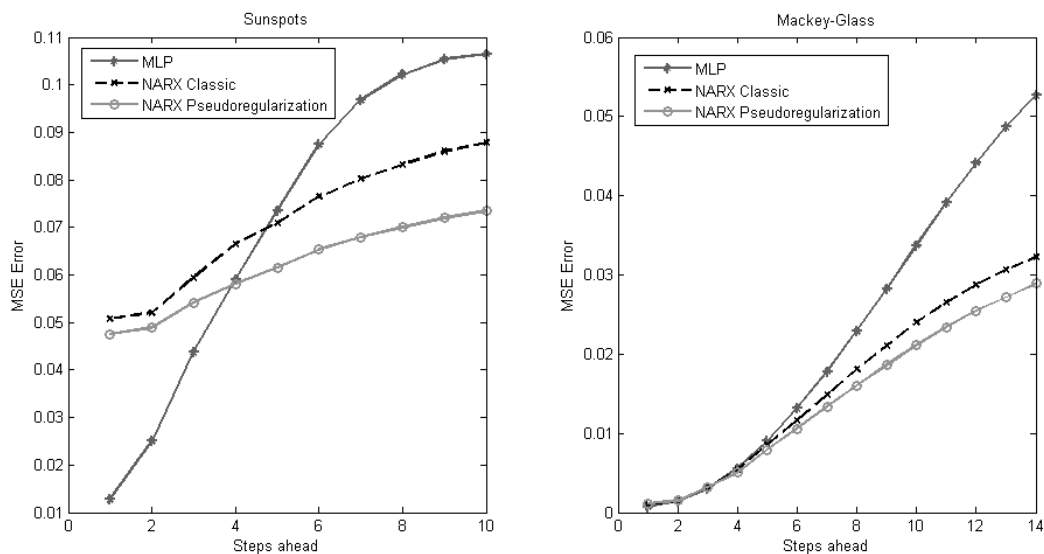


Рис. 7. Залежність помилки багатокрокового прогнозування від об'єму прогнозування

Нарешті було навчено по 100 рекурентних мереж протягом 50 епох з використанням псевдорегуляризації з використанням додаткового розширеного фільтра Калмана, параметра величини псевдорегуляризації  $\lambda = 0,1$ . Методи відбору мереж і тестування аналогічні. Результати представлені в табл. 1, 2 і на рис. 7.

Таблиця 1. Результати багатокрокового прогнозування чисел Вольфа

Назва підходу/Н	1	2	3	4	5	6	10
MLP	0,0128	0,0250	0,0437	0,0589	0,0736	0,0874	0,1065
NARX Classic	0,0507	0,0520	0,0594	0,0664	0,0709	0,0764	0,0879
NARX Pseudoregularization	0,0475	0,0489	0,0542	0,0581	0,0616	0,0653	0,0735

Таблиця 2. Результати багатокрокового прогнозування процесу Маккея-Гласса

Назва підходу/Н	1	2	3	4	5	10	14
MLP	0,0007	0,0015	0,0030	0,0055	0,0090	0,0337	0,0526
NARX Classic	0,0010	0,0014	0,0033	0,0054	0,0086	0,0240	0,0323
NARX Pseudoregularization	0,0012	0,0015	0,0032	0,0050	0,0079	0,0211	0,0289

## 9. Висновки

У роботі запропоновано новий метод навчання рекурентних нейромереж із застосуванням псевдорегуляризації для зменшення ефекту зникнення градієнта. Було проведено експериментальне дослідження запропонованого методу на прикладах числових послідовностей чисел Вольфа і процесу Маккея-Гласса, що показало ефективність нового методу для багатокрокового прогнозування часових рядів у порівнянні із традиційним навчанням рекурентних нейромереж методом розширеного фільтра Калмана. Використання псевдорегуляризації забезпечило зменшення помилки багатокрокового прогнозування для рекурентних нейромереж в середньому на 10–30%.

## СПИСОК ЛІТЕРАТУРИ

1. Хайкин С. Нейронные сети: полный курс / Хайкин С.; пер. с англ. – [2-е изд., испр.]. – М.: Вильямс, 2006. – 1104 с.
2. Gers F.A. Applying LSTM to Time Series Predictable Through Time-Window Approaches / F.A. Gers, D. Eck, J. Schmidhuber // Proc. of International Conference on Artificial Neural Networks. – Berlin, 2001. – P. 669 – 676.
3. Осовский С. Нейронные сети для обработки информации / Осовский С. – М.: Финансы и статистика, 2002. – 344 с.
4. Cernansky M. Simple recurrent network trained by RTRL and Extended Kalman Filter algorithms / M. Cernansky, L. Benuskova // Neural Network World. – 2003. – N 13, Vol. 3. – P. 223 – 234.
5. De Jesus O. Backpropagation: Algorithms for a Broad Class of Dynamic Networks / O. de Jesus, M.T. Hagan // IEEE Transactions on Neural Networks. – 2007. – N 1, Vol. 18. – P. 14 – 27.
6. Hochreiter S. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies / S. A Field Guide to Dynamical Recurrent Neural Networks / S. Hochreiter, Y. Bengio, P. Frasconi [et al.] // IEEE Press. – 2001. – 421 p.
7. Bone R. Advanced Methods for Time Series Prediction Using Recurrent Neural Networks / R. Bone, H. Cardot // Recurrent Neural Networks for Temporal Data Processing, Chapter 2, Intech. – Croatia. – 2011. – P. 15 – 36.
8. On the convergence of EKF-based parameters optimization for Neural Networks / Alessandri A., Cuneo M., Pagnan S. [et al.] // Proc. of the 42nd IEEE Conference on Decision and Control. – Maui, Hawaii, USA, 2001. – Vol. 6. – P. 6181 – 6186.
9. Prokhorov D.V. Toyota Prius HEV Neurocontrol and Diagnostics / D.V. Prokhorov // Neural Networks. – 2008. – N 21. – P. 458 – 465.

*Стаття надійшла до редакції 30.07.2012*