

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Нейромережева ідентифікація нелінійних об'єктів

(тема)

Виконав:

студент II курсу, групи СПМ-20-3
Попова В.Ю.
(прізвище, ініціали)

Спеціальність 123 «Комп'ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системне програмування
(повна назва освітньої програми)

Керівник: проф. Руденко О.Г.
(посада, прізвище, ініціали)

Допускається до захисту

В. о. зав. кафедри ЕОМ

(підпис)

Волк М.О.

(прізвище, ініціали)

2022 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 123 «Комп'ютерна інженерія» _____
(код і повна назва)

Тип програми _____ освітньо-наукова _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Системне програмування _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ ____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту _____ Поповій Валерії Юріївні _____
(прізвище, ім'я, по батькові)

1. Тема роботи Нейромережева ідентифікація нелінійних об'єктів

затверджена наказом по університету від “ 24 ” березня 2022 р. № 413 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 18 травня 2022 р.

3. Вхідні дані до роботи 1)штучна нейронна мережа

4. Перелік питань, що потрібно опрацювати у роботі _____

1) аналіз предметної області;

2) опис нелінійних об'єктів;

3) методи ідентифікації нелінійних систем;

4) огляд математичних інструментів;

5)нелінійні динамічні системи ідентифікації штучною нейронною системою мережі;

б)висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) _____

Слайд-презентація - 14 слайдів _____

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Огляд існуючих методів ідентифікації нелінійних об'єктів	31.03.22-13.04.22	
2	Вибір та обґрунтування методики дослідження	14.04.22-21.04.22	
3	Вибір інструментальних засобів	22.04.22-27.04.22	
4	Проведення математичного аналізу	28.04.22-03.05.22	
5	Оформлення матеріалів кваліфікаційної роботи	04.05.22-06.05.22	
6	Подання кваліфікаційної роботи керівникові та її попередній захист	07.05.22-08.05.22	
7	Подання кваліфікаційної роботи на рецензування	09.05.22-15.05.22	

Дата видачі завдання 28 березня 2022 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

проф. Руденко О.Г. _____
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 108 с., 25 рис., 80 джерел.

ШТУЧНА НЕЙРОННА МЕРЕЖА, НЕЛІНІЙНИЙ ДИНАМІЧНИЙ ОБ'ЄКТ, ІДЕНТИФІКАЦІЯ, ANARX, MIMO, NN-SANARX.

Метою кваліфікаційної роботи є розглянення ідентифікації нелінійних систем на основі нейронних мереж: підхід на основі моделі ANARX

У ході виконання кваліфікаційної роботи було розглянено ідентифікації на основі нейронних мереж нелійними системами на основі моделі. Різні структури штучних нейронних мереж розглядаються як інструменти апроксимації для ідентифікації складних нелінійних систем і процесів. У дослідженні показано, що правильна залежна від програми архітектура мережі може значно покращити якість ідентифікації, що має вирішальне значення для проектування системи керування. Крім того, специфічні структури моделі на основі нейронної мережі дозволяють комбінувати класичні алгоритми керування з ідентифікацією та адаптацією на основі нейронної мережі.

Структура ANARX запропонована та застосована для ідентифікації нелінійних систем MIMO. Клас моделі Simplified ANARX (NN-SANARX) на основі NN представлений тут як альтернативний варіант, який дозволяє спростити обчислення зворотного зв'язку, доводячи його до розв'язку системи лінійних рівнянь. У роботі також розроблено адаптивний контролер для нелінійних систем MIMO, ідентифікованих структурою NN-SANARX.

ABSTRACT

Master's thesis: 108 pages, 25 figures, 80 sources.

ARTIFICIAL NEURAL NETWORK, NONLINEAR DYNAMIC OBJECT, IDENTIFICATION, ANARX, MIMO, NN-SANARX.

The major goal of this thesis is to consider the identification of nonlinear systems based on neural networks: an approach based on the ANARX model

In the course of the qualification work, identifications based on neural networks by nonlinear systems based on the model were considered. Different structures of artificial neural networks are considered as approximation tools for the identification of complex nonlinear systems and processes. The study shows that the correct program-dependent network architecture can significantly improve the quality of identification, which is crucial for the design of the control system. In addition, specific neural network-based model structures allow the combination of classical control algorithms with neural network-based identification and adaptation.

The ANARX structure is proposed and used to identify nonlinear MIMO systems. The NN-based Simplified ANARX (NN-SANARX) class is presented here as an alternative to simplify feedback calculations by solving a system of linear equations. The thesis also developed an adaptive controller for nonlinear MIMO systems identified by the NN-SANARX structure.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	12
1.1 Опис предметної області.....	12
1.2 Переваги та недоліки предметної області	13
1.3 Актуальність предметної області	14
2 ОПИС НЕЛІНІЙНИХ ОБ'ЄКТІВ.....	16
2.1 Нейромережевий підхід до ідентифікації.....	16
3 МЕТОДИ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ СИСТЕМ	18
3.1. Сучасні способи ідентифікації нелінійних систем.	18
3.2. Реалізація фільтру Вольтерра другого порядку для ідентифікації нелінійних систем	22
4 ОГЛЯД МАТЕМАТИЧНИХ ІНСТРУМЕНТІВ	26
4.1 Нелінійні системи.....	26
4.2 Математичні моделі нелінійних динамічних систем	26
4.2.1 Моделі введення-виведення.....	28
4.2.3 Моделі NARX.....	30
4.2.4 Моделі ANARX	31
4.3 Точність моделі	32
4.4 Штучні нейронні мережі	34
4.4.1 Штучні нейрони	34
4.4.2 Функції активації.....	36
4.4.3 Персептрон.....	37
4.4.4 Теорема Стоуна-Вейєрштрасса	39
4.4.5 Рекурентні нейронні мережі	40
4.4.6 Теорема Зонтага	42

4.5 Алгоритми навчання	43
4.5.1 Зворотне поширення помилки градієнтного спуску	45
4.5.2 Алгоритм Левенберга-Марквардта	45
5 НЕЛІНІЙНІ ДИНАМІЧНІ СИСТЕМИ ІДЕНТИФІКАЦІЇ ЗІ ШТУЧНОЮ НЕЙРОННОЮ СИСТЕМОЮ МЕРЕЖІ	47
5.1 Рекурентні моделі на основі нейронних мереж	47
5.2 Нейронні мережі прямого зв'язку із зовнішнім зворотним зв'язком для ідентифікації динамічних систем. Моделі NN-NARX	49
5.3.1 Числовий приклад 5.1	52
5.3 Модель Хаммерштейна на основі нейронної мережі	54
5.4 Числовий приклад 5.2	58
5.4.1 Числовий приклад 5.3 – Застосування моделі Хаммерштейна на основі нейронної мережі для ідентифікації серводвигуна постійного струму з нелінійним драйвером	61
5.5 Моделі ANARX на основі нейронних мереж	63
5.5.1 Реалізація простору стану за допомогою NN-ANARX	65
5.5.2 Числовий приклад 5.4	66
5.5.3 Числовий приклад 5.5	68
5.6 Модель Хаммерштейна на основі NN-ANARX	71
5.6.1 Числовий приклад 5.6	74
5.7 Моделі нелінійних систем MIMO	76
5.7.1 Моделі MIMO NARX і MIMO NN-NARX	76
5.7.2 Моделі MIMO ANARX і MIMO NN-ANARX	79
5.7.3 Числовий приклад 5.7 – ANARX-модель руки хірурга на основі NN для розпізнавання руху та прогнозування руху	81
ВИСНОВКИ	89
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	92
ДОДАТОК А Графічний матеріал кваліфікаційної роботи	101

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ

- БП – багатошаровий перцептрон
НДО – нелінійний динамічний об’єкт
ШНМ – штучна нейронна мережа
ADALINE – адаптивний лінійний елемент
ANARX – адитивний нелінійний авторегресивний екзогенний
BP BackPropagation – резервуарний реактор безперервного
перемішування
CSTR – постійний постійний струм
L. M. – Левенберг-Марквардт
MIMO – багато входів багато виходів
MSE – середньоквадратична помилка
MISO – багато входів один вихід
NARX – нелінійний авторегресивний екзогенний
NN – нейронна мережа
NN-ANARX – адитивна нелінійна авторегресія на основі нейронної
мережі
RL – підкріплення навчання
SIMO – один вхід і кілька виходів
SISO – один вхід один вихід
SNR – Scrub Nurse Robot

ВСТУП

Поява ШНМ пов'язані з розумінням того, що мозок живого організму працює інакше, ніж комп'ютер. Людський мозок – це дуже складна нелінійна паралельна інформаційно-керуюча система, здатна до мислення, накопичення та відновлення інформації, вирішення проблем. Ця система складається з досить однотипних будівельних блоків - нервових клітин, або нейронів, що є простими елементами обробки сигналів, які отримують і комбінують інформацію від інших нейронів через входи - дендрити.

З інженерної точки зору ШНМ – це паралельно розподілена система обробки інформації, утворена тісно пов'язаними простими обчислювальними вузлами (однотипними або різними), яка має властивість накопичувати експериментальні знання, узагальнювати їх та робити доступними для користувача у формі зручної для інтерпретації та прийняття рішень.

На сьогодні є всі підстави говорити про досягнення певних успіхів нейромережових технологій у вирішенні складних завдань як суто наукових, так і у сфері техніки, бізнесу, фінансів, медичної діагностики та інших галузей, пов'язаних з інтелектуальною діяльністю.

У цій роботі узагальнено досвід досліджень та основні результати, досягнуті у галузі ідентифікації систем на основі нейронних мереж моделі. Робота присвячена ідентифікації нелінійних систем за допомогою моделей на основі різних структур нейронних мереж. Основна увага приділяється ANARX структурі моделі.

Значна увага приділяється ідентифікації ANARX-моделей нелінійних систем через їх дуже важливі переваги перед класичними моделями NARX. ANARX — це підклас моделей NARX з відокремленими екземплярами часу. Обмеження, накладені цим підкласом, гарантують лінеаризованість за допомогою динамічного вихідного зворотного зв'язку, а також репрезентативності моделі в просторі станів. Ці переваги особливо важливі

для програм керування. Тому такий тип моделі є розумним вибором для керування широким класом нелінійних систем.

Робота також натхнена здатністю нейронних мереж моделювати складну поведінку нелінійних систем з одного боку та їх здатністю відтворювати довільну структуру системи, вибираючи відповідні функції та зв'язки між вузлами з іншого боку.

У цій роботі розглядається штучна нейронна мережа як інструмент ідентифікації нелінійних систем для управління на основі моделі. Завдяки тому, що нейронна мережа складається з простих взаємопов'язаних вузлів (штучних нейронів), нейронна мережа може представляти будь-яку структуру моделі, що відповідає вимогам алгоритму керування. Правильну структуру моделі можна отримати, вибравши зв'язки між нейронами (з'єднавши їх належним чином) і визначивши функції активації кожного нейрона в мережі відповідно до потреб системи управління.

Неаналітичні методи, що розглядаються в роботі, розглядають ідентифіковану систему як «чорний ящик» та ідентифікують параметри моделі за допомогою набору даних, зібраних із входу та виходу системи. Коли використовуються ці методи, структуру моделі необхідно вибрати перед початком процедури ідентифікації або повернути під час неї за попередньо визначеним алгоритмом. Структура моделі істотно залежить від її застосування.

Історія розвитку технічних систем, заснованих на взаємозв'язку вузлів, що представляють математичні моделі біологічних нейронів, бере свій початок з 1943 року, коли Мак-Калох і Піттс запропонували математичну модель нейронів [1]. Ця модель називається штучним нейроном і використовується в більшості додатків на основі штучних нейронних мереж дотепер. Ця модель, запропонована майже 65 років тому, також є основним базовим елементом систем, які розглядаються в цій кваліфікаційній роботі.

Навчальну машину, створену Едмондсом і Мінським у 1951 році, можна вважати першим симулятором штучної нейронної мережі. Це

навчання нейронної мережі машина під назвою SNARC була заснована на ідеях Хебба [48], які математично відтворюють те, що відбувається, коли синаптичні передачі відбуваються в мозку [2]. Проте справжній початок нейронних мереж (NN) і навчання на основі NN винахід Розенблаттом [49] простої нейроноподібної мережі навчання у 1962 році. Ця найпростіша багат шарова повнозв'язана нейронна мережа називається персептрон. Сьогодні багат шаровий персептрон залишається найпопулярнішою і найпоширенішою структурою нейронної мережі, тому що його дуже хороших і перевірених [44] можливостей апроксимації.

Слід зазначити, що приблизно до 1980-х років у цій області проводилося дуже мало досліджень, в основному через високу обчислювальну складність навчання мереж, які здатні вирішувати складні проблеми. Однак багато штучних нейронних мереж, які використовуються сьогодні, все ще базуються на ранніх досягненнях нейрона МакКалоха-Пітса і персептрона Розенблата. Численні приклади та результати досліджень можна знайти в літературі, що демонструють дуже хороші апроксимаційні, ідентифікаційні та адаптаційні здібності цього типу нейронних мереж та їхнє значення для проектування систем керування.

Більшість досліджень вказують на апроксимаційні можливості нейронних мереж та застосування цієї властивості в технічних системах. При цьому істотно менше уваги приділяється структурі нейронної мережі. Протягом останніх 20 років багат шаровий персептрон продемонстрував свої дуже хороші апроксимаційні здібності та застосовність для вирішення багатьох складних проблем із дуже різних областей, і тому він занадто загальний, щоб бути найкращим у кожному конкретному застосуванні.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Опис предметної області

Нейронні мережі, також відомі як штучні нейронні мережі (ANN) або змодельовані нейронні мережі (SNN), є підмножиною машинного навчання і лежать в основі алгоритмів глибокого навчання. Їх назва та структура натхненні людським мозком, імітуючи спосіб, яким біологічні нейрони сигналізують один одному.

ШНМ складаються з шарів вузлів, що містять вхідний шар, один або кілька прихованих шарів і вихідний шар. Кожен вузол або штучний нейрон з'єднується з іншим і має відповідну вагу та поріг. Якщо вихід будь-якого окремого вузла перевищує вказане порогове значення, цей вузол активується, надсилаючи дані на наступний рівень мережі. В іншому випадку дані не передаються на наступний рівень мережі.

Нейронні мережі покладаються на навчальні дані, щоб з часом вивчати та покращувати свою точність. Однак, як тільки ці алгоритми навчання будуть точно налаштовані на точність, вони стануть потужними інструментами в інформатиці та штучному інтелекті, що дозволить нам класифікувати та кластерувати дані з високою швидкістю. Завдання розпізнавання мовлення або зображення можуть займати хвилини та години, якщо порівнювати з ручною ідентифікацією експертів. Однією з найвідоміших нейронних мереж є пошуковий алгоритм Google.

Нейронні мережі можна розділити на різні типи, які використовуються для різних цілей.

Персептрон — найстаріша нейронна мережа, створена Френком Розенблатом у 1958 році. Вона має один нейрон і є найпростішою формою нейронної мережі.

Нейронні мережі прямого зв'язку або багат шарові перцептрони (MLP) – це те, на чому ми впершу чергу зосередилися в цій статті. Вони складаються з вхідного шару, прихованого шару або шарів і вихідного шару. Хоча ці нейронні мережі також зазвичай називають MLP, важливо зазначити, що вони насправді складаються з сигмовидних нейронів, а не з перцептронів, оскільки більшість проблем реального світу є нелінійними. Дані зазвичай подаються в ці моделі для їх навчання, і вони є основою для комп'ютерного зору, обробки природної мови та інших нейронних мереж.

Згорткові нейронні мережі (CNN) подібні до мереж з прямим зв'язком, але вони зазвичай використовуються для розпізнавання зображень, образів та/або комп'ютерного зору. Ці мережі використовують принципи лінійної алгебри, зокрема множення матриці, для визначення шаблонів у зображенні.

Рекурентні нейронні мережі (RNN) ідентифікуються за їх петлями зворотного зв'язку. Ці алгоритми навчання впершу чергу використовуються під час використання даних часових рядів для прогнозування майбутніх результатів, таких як прогнозування фондового ринку або прогнозування продажів.

1.2 Переваги та недоліки предметної області

Ми можемо виділити такі переваги для використання нейромережі:

- 1 здатність ігнорувати сторонню інформацію;
- 2 можливість зберігати працездатність у разі втрати окремих елементів;
- 3 висока швидкість роботи. Завдяки тому, що ШНМ складається з тисяч мікропроцесорів, що взаємодіють між собою, завдання вирішуються набагато швидше, ніж стандартними способами.

Проте повністю покладатися на нейромережі не можна. Їх можна використовувати як ефективне доповнення до інших методів, але не як єдиний варіант досягнення мети. Для пропонована ШНМ відповідь не буде

абсолютно точною, тільки зразковою. Отже, не можна повністю покладатися на результати роботи нейромережі, оскільки є ймовірність недостовірного рішення. Кожен штучний нейрон діє незалежно від сусідніх, не співвідносить свою поведінку з іншими мікропроцесорами. Специфіка ШНМ полягає у тому, що немає гарантії абсолютної правдивості результату.

1.3 Актуальність предметної області

Потенційними областями застосування штучних нейронних мереж є ті, де людський інтелект є малоефективним, а традиційні обчислення трудомісткі або фізично неадекватні, тобто не відображають або погано відображають реальні фізичні процеси та об'єкти. Актуальність застосування нейронних мереж багаторазово зростає, коли з'являється необхідність вирішення погано формалізованих завдань. Основні галузі застосування нейронних мереж: автоматизація процесу класифікації, автоматизація прогнозування, автоматизація процесу розпізнавання, автоматизація процесу прийняття рішень; управління, кодування та декодування інформації; апроксимація залежностей.

За допомогою нейронних мереж успішно вирішується важливе завдання у галузі телекомунікацій – проектування та оптимізація мереж зв'язку (знаходження оптимального шляху трафіку між вузлами). Крім управління маршрутизацією потоків, нейронні мережі використовуються для отримання ефективних рішень у галузі проектування нових телекомунікаційних мереж.

Розпізнавання мови – одна з найпопулярніших галузей застосування нейронних мереж.

Ще одна область – управління цінами та виробництвом (втрати від неоптимального планування виробництва часто недооцінюються). Оскільки попит та умови реалізації продукції залежать від часу, сезону, курсів валют та багатьох інших факторів, то і обсяг виробництва повинен гнучко

варіюватися з метою оптимального використання ресурсів (нейромережна система виявляє складні залежності між витратами на рекламу, обсягами продажів, ціною, цінами конкурентів, днем тижня, сезоном і т. д.). Через війну використання системи здійснюється вибір оптимальної стратегії виробництва з погляду максимізації обсягу продажу чи прибутку.

При аналізі споживчого ринку(маркетинг), коли звичайні (класичні) методи прогнозування відгуку споживачів може бути недостатньо точні, використовується прогноуюча нейромережна система з адаптивної архітектурою нейросимулятора.

Дослідження попиту дозволяє зберегти бізнес компанії в умовах конкуренції, тобто підтримувати постійний контакт із споживачами через «зворотний зв'язок». Великі компанії проводять опитування споживачів, дозволяють з'ясувати, які чинники є вирішальними при купівлі даного товару чи послуги, чому у деяких випадках перевагу надається конкурентам і які товари споживач хотів би побачити у майбутньому. Аналіз результатів такого опитування – досить складне завдання, оскільки існує велика кількість корелейованих параметрів. Нейромережева система дозволяє виявляти складні залежності між факторами попиту, прогнозувати поведінку споживачів при зміні маркетингової політики, знаходити найбільш значущі фактори та оптимальні стратегії реклами, а також окреслювати сегмент споживачів найбільш перспективний для даного товару.

2 ОПИС НЕЛІНІЙНИХ ОБ'ЄКТІВ

2.1 Нейромережевий підхід до ідентифікації

Ефективність створюємих систем управління реальними об'єктами в значній мірі залежить від якості використовуваних при цьому математичних моделей, які, з одного боку, повинні найбільш повно відображати властивості досліджуваного об'єкта, а з іншого – бути зручними для реалізації алгоритмів керування. Відсутність повної інформації про умови функціонування, властивості самих об'єктів і перешкод обумовлюють необхідність застосування при управлінні ними адаптивного підходу, що допускає можливість використання спрощених, зокрема, лінійних моделей. Хоча такий підхід і дозволяє в ряді випадків істотно зменшити апріорну невизначеність і реалізувати досить ефективну ідентифікацію, обмеження лінійними моделями не завжди забезпечує отримання необхідного результату. Тому більш ефективною є розробка систем управління на основі адаптивного підходу в поєднанні з методами теорії ШНМ.

Більшість реальних систем характеризуються нелінійними залежностями, складними для моделювання динамічними властивостями, наявністю неконтрольованих шумів і перешкод, що перешкоджають реалізації традиційних стратегій управління, оскільки, як сучасна (зокрема теорія адаптивного і оптимального управління), так і класична теорія управління в значній мірі базуються на ідеї лінеаризації систем [9].

Для практичного застосування даного підходу необхідна, перш за все, розробка математичних моделей. Однак математичне моделювання, що реалізовується на основі припущення про лінійність системи, може невідображати її дійсних фізичних властивостей. Навіть якщо вдається побудувати складні математичні моделі, точно відображають фізичні співвідношення між входом і виходом системи, вони можуть виявитися

марними для цілей управління. Практично прийнятними можуть бути тільки моделі з низькою чутливістю по параметрам, що для нелінійних систем вкрай складно забезпечити [10-12].

Для практичного застосування алгоритмів управління необхідно, щоб вони були адаптивними, стійкими, нелійними, а також простими для реалізації та розуміння. Саме з цих причин в даний час широке застосування в задачах управління отримали ШНМ, що володіють зазначеними вище властивостями.

Будучи альтернативою традиційним методам ідентифікації та управління, нейромережевий підхід заснований на повністю визначених ШНМ для вироблення необхідних сигналів управління. Високий інтерес доданого напрямку в теорії управління визначається наступними факторами: - ШНМ – адаптивні системи, здатні навчатися вирішення складних завдань. Необхідною умовою якісного навчання є правильний вибір структури НММ і достатній обсяг інформації, який використовується при навчанні; - нейромережеві технології управління дозволяють подолати багато труднощів, що виникають при роботі з нелійними об'єктами або з об'єктами невідомої структури і нерозв'язні за допомогою звичайних методів адаптивного управління. Здатність ШНМ реалізовувати складні нелінійні відображення обумовлена застосуванням сигмоїдальних функцій активації (або деяких нелінійних функцій загального вигляду); - високу швидкодію і надійність обумовлена високим ступенем паралельності ШНМ. Простота реалізації нейронних мереж і їх здатність до навчання роблять їх особливо привабливими при управлінні складними нелійними об'єктами в реальному часі.

3 МЕТОДИ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ СИСТЕМ

Для вирішення задачі ідентифікації нелінійних об'єктів розроблено досить багато підходів та методів [4, 7]. На сучасному етапі зросли вимоги до точності характеристик застосовуваних алгоритмів ідентифікації. У зв'язку з цим модифікуються класичні підходи до вирішення задачі ідентифікації нелінійних систем з метою підвищення їх точності та зменшення обмежень застосування [3], а також універсальні пошукові методи, які вимагають мінімальної апріорної інформації про ідентифіковані системи, але складні в реалізації.

До останніх відносяться генетичні алгоритми багатопараметричного пошуку рішень, що адекватно описують значення параметрів досліджуваної системи, а також різні методи самоорганізації та нейронні мережі [2, 5, 6, 11].

Представлені найбільш значущі методи ідентифікації нелінійних систем автоматичного управління, показані їх переваги та недоліки, запропонована модифікація класичного методу Вольтерра, заснована на спрощенні реалізації алгоритму

3.1. Сучасні способи ідентифікації нелінійних систем.

При проектуванні систем керування різними динамічними об'єктами часто необхідно мати нелінійну математичну модель досліджуваного процесу. Подібні нелінійні моделі використовуються також для прогнозування стану об'єкта та параметрів зовнішнього середовища, на основі яких здійснюється аналіз та вироблення управлінських впливів.

Розглянемо деякі методи ідентифікації нелінійних систем, що застосовуються на практиці. Одним із популярних методів ідентифікації є алгоритм ідентифікації на основі розкладання функціоналів Вінера [7]. В результаті реалізації цієї процедури визначаються набори оптимальних ядер

Вінера, що визначаються взаємокореляційним методом

$$h_n(\sigma_1, \dots, \sigma_n) = \frac{1}{K^n * n!} M[(y(t) - \sum_{i=0}^{n-1} G_i[h_i, x(t)]) * \prod_{r=1}^n x(t - \sigma_r)], \quad (3.1)$$

де $n=1, 2, \dots$.

Завдання визначення ядер Вінера можна вважати коректним лише якщо на вхід досліджуваної системи подається ідеальний білий шум. Такий процес реалізувати на практиці не є можливим, тому виникає проблема вибору оптимальних параметрів тестуючого сигналу, що забезпечують сприйняття цього сигналу системою, як білий шум. Для отримання точних оцінок характеристик досліджуваної системи необхідно вибрати частотний діапазон тестуючого сигналу. Але щоб вхідний гауссів процес для системи був білим шумом, частотний діапазон повинен повністю перекривати смугу пропускання системи, а він сам у межах цієї смуги повинен мати постійну спектральної густиною.

На практиці через обмеженість спектральної смуги тестуючого сигналу оцінки ядер Вінера є усунутими [3]. Вибрати оптимальний частотний діапазон тестуючого сигналу, орієнтуючись на смугу пропускання, можливо тільки для лінійних систем. Для нелінійних систем ця характеристика не є інформативною.

Для отримання уточнених оцінок ядер розроблено алгоритм диференційованого підходу вибору оптимального частотного діапазону тестуючого сигналу, тобто, для кожного порядку розрахованих ядер пропонується визначати свій оптимальний частотний діапазон тесту. Однак такий підхід складний у реалізації, особливо на борту динамічний об'єкт.

Сучасні пошукові алгоритми в практичних додатках представлені в здебільшого алгоритмами самоорганізації, нейронними мережами та генетичними алгоритми.

Генетичні алгоритми оперують із популяцією оцінок потенційних

рішень (індивідуумів), використовуючи принцип "виживає найбільш пристосований". На кожному кроці алгоритму утворюється нове безліч наближень, створюване за допомогою процесу відбору індивідуумів відповідно до їхнього рівня придатності. Операндом генетичного алгоритму є ген G – закодована оцінка того чи іншого ускладнення технологічного режиму, об'єднаний у "хромосому" - n -мірний вектор S . Область усіх значень генів G уніфікована – $[0...7]$, що забезпечує можливість застосування оператори генетичних алгоритмів між ними. Розмір коду залежить від необхідної точності оцінки прогнозованих параметрів. Значення коду G визначає кількісну оцінку ускладнення, яка може бути пропорційною чи прогресивною [9].

Проте, генетичні алгоритми погано масштабуються під складність вирішуваної проблеми. Це означає, що кількість елементів, схильних до мутації дуже велика, якщо великий розмір області пошуку рішень. Для того, щоб зробити так, щоб такі проблеми піддавалися еволюційним алгоритмам, вони мають бути поділені на найпростіші уявлення. Але тоді потрібна хороша сумісність цих уявлень з іншими частинами у процесі оцінки придатності [12].

Нейромережа є структурою, що складається з безлічі однотипних елементів – нейронів, поєднаних між собою синаптичними зв'язками. Нейронні мережі дозволяють побудувати моделі досліджуваних об'єктів з досить високою точністю, але вимагають у своїй тривалого часу реалізації процесу навчання. Основним завданням побудови та навчання нейронної мережі є апроксимація функції. Маючи навчальну вибірку вхідних даних та значень функції, потрібно визначити вагові коефіцієнти нейронної мережі так, щоб результат роботи мережі (значення вихідний функції) на векторі вхідних змінних був якомога ближче до заданого значення функції (навчального значення) для цього вектора.

Навчання нейронної мережі відбувається за наступним алгоритмом:

1 початкові ваги задаються випадковим чином;

- 2 реалізується епоха навчання;
- 3 перевіряються умови завершення роботи нейронної мережі.

У процесі реалізації епохи навчання нейронної мережі всім вхідних векторів по черзі здійснюються такі процедури:

- 1 значення вхідного вектора пропускаються через мережу та визначається результат роботи мережі;
- 2 визначається відхилення результату мережі від вихідного значення;
- 3 змінюються ваги зв'язків елементів мережі від останніх шарів до перших. Зміна відбувається відповідно до методу градієнтного спуску.

Метою є знайти мінімум помилки для кожного елемент. Після того як пройшов процес навчання, перевіряється умова закінчення функціонування алгоритму A саме, наскільки результати роботи нейронної мережі відрізняються від вихідних значень. Якщо умова ще не виконано, то алгоритм повертається до другого кроку. Якщо відхилення від вихідної вибірки задовольняє умовам, заданим в алгоритмі апріорно, то нейронна мережа вважається навченою.

В даний час коло вирішення завдань, що вирішуються однією окремо взятою нейронною мережею досить обмежений. Це пов'язано з тим, що для кожного можливого застосування нейромережа вибираються топологія, алгоритми та коефіцієнти нейромережі, найбільш придатні для даної задачі, і вони можуть бути неприйнятними для жодної іншої [1].

Алгоритми самоорганізації за класифікацією А.Г. Івахненка представлені цілим класом алгоритмів, що реалізують механізми спрямованих самовідбору [2]. Структура та базисні функції моделей алгоритму вибираються залежно від виду матриці моделі досліджуваного процесу. Набір базових функцій $F_p = \{b_i(x) | i = \overline{1, N}\}$, где $b_i(x)$ – нелінійні функції можуть змінюватися залежно від конкретних практичних додатків. Кожній базовій функції ставиться у відповідність двовимірний вектор параметрів $(a, f)^T$, де a – амплітуда, f – частота, що визначаються у процесі

функціонування алгоритму.

Шукана модель матиме вигляд:

$$\Phi(x) = \sum_{i=1}^N a_i b_{n_i}(f_i x) \quad (3.2)$$

Алгоритм самоорганізації ґрунтується на гіпотезі селекції моделей. Перший крок алгоритму полягає в ідентифікації базисних функцій за заданим критерієм. На наступних рівнях будується комбінація моделей попереднього рівня, що пройшли пороговий самовідбір з ансамблю критеріїв [10]. Проте методологічною основою використання підходу самоорганізації для побудови прогнозуючих моделей припущення про те, що вичерпна інформація, що характеризує динаміку досліджуваного об'єкта, міститься у вимірах (таблиці спостережень, вибірці даних) та ансамблі критеріїв селекції моделей. Вимірювальні вибірки повинні містити досить повну інформацію про об'єкт дослідження, відображати сучасний стан об'єкта, що функціонує у цих умовах. Якщо для побудови моделі використовувати застарілі або неповні інформаційні вибірки, модель вийде неадекватною, і прогноз із використанням такої моделі утримуватиме суттєві похибки.

Конструктивним підходом у вирішенні задачі ідентифікації нелінійних систем управління є використання фільтруючої структури у вигляді послідовності Вольтерра.

3.2 Реалізація фільтру Вольтерра другого порядку для ідентифікації нелінійних систем

З теоретичного погляду фільтр Вольтерра є привабливим, оскільки він може взаємодіяти із загальним класом нелінійних систем, у той час як його вихід залишається лінійним по відношенню до різних системних ядер високого порядку або імпульсних реакцій. Однак однією з головних причин

досить рідкісного застосування методики фільтрації Вольтерра на практиці є значна складність, пов'язана з реалізацією фільтрів Вольтерра. Наприклад, використання методики лінеаризації, якої фільтр Вольтерра розглядається як лінійний фільтр з мультирозмірним вхідним сигналом, призводить до серйозних обчислювальних проблем, пов'язаних з збільшенням кількості операцій зі збільшенням порядку фільтра. Таким чином, головним завданням є знаходження спрощень у розробці та реалізації фільтра Вольтерра.

Особливу увагу приділимо фільтру Вольтерра 2-го порядку (ФВ2), що складається з паралельної комбінації лінійного та квадратичного фільтрів, є прототипом нелінійного фільтра, за допомогою якого можна покращити характеристики лінійного фільтра з відносно малою помилкою обчислення. Визначимо ФВ2 як:

$$y(n) = h_0 + \sum_{j=0}^{N-1} a(j)x(n-j) + \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} b(j,k)x(n-j)x(n-k), \quad (3.3)$$

де $\{a(j)\}$ і $\{b(j,k)\}$ називаються лінійною та квадратичною вагою відповідно, а N показує довжину фільтра (передбачається симетричність квадратичних ваг фільтра, тобто. $b(j,k) = b(k,j)$) [8].

Потрібно мінімізувати середньоквадратичну помилку (СКОШ) між основним сигналом $s(n)$ та виходом фільтра $y(n)$, тобто.

$$\varepsilon = E \left[|s(n) - y(n)|^2 \right] \quad (3.4)$$

Першим кроком у визначенні мінімуму СКОШ фільтра Вольтерра 2-го порядку є вимога бездрейфового виходу фільтра. Іншими словами, має бути $E[y(n)] = 0$, так як основний сигнал має нульове математичне очікування. Тоді виходить наступне співвідношення між h_0 і $b(j,k)$:

$$h_0 = -\sum_{j=0}^{N-1} \sum_{k=0}^{N-1} b(j,k)r_x(j-k), \quad (3.5)$$

де $r_x(j) = E[x(n)x(n-j)]$ означає автокореляційну функцію $x(n)$. Важливо включення члена нульового порядку h_0 , так як без цього вихід мінімальної СКОШ ФВ2 не є обов'язково бездрейфовим і помилка буде мати, отже, тенденцію до збільшення. Отже, формула визначення ФВ2 виглядатиме так:

$$y(n) = \sum_{j=0}^{N-1} a(j)x(n-j) + \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} b(j,k)[x(n-j)x(n-k) - r_x(j-k)] \quad (3.6)$$

Наступний крок – визначення лінійної та квадратичного ваг фільтра, які визначають мінімум СКОШ. Для цього в Виведемо просте рішення для оптимального ФВ2 в припущенні, що на вході фільтра білий гаусовий шум.

Формулу для визначення ФВ2 можна переписати у матричному вигляді:

$$y(n) = A^T X(n) + t_r \{ B[X(n)X^T(n) - R_x] \},$$

$$X(n) = [x(n), \dots, x(n-N+1)]^T, A = [a(0), \dots, a(N-1)]^T,$$

$$B = \begin{bmatrix} b(0,0) & \dots & b(0,N-1) \\ \dots & \dots & \dots \\ b(N-1,0) & \dots & b(N-1,N-1) \end{bmatrix}, \quad (3.7)$$

а R_x вказує на $N \times N$ матрицю від $x(n)$, де $r_x(j, k) = r_x(j-k)$ автокореляційна функція вхідного сигналу $x(n)$. А і B – оператори лінійного та квадратичного фільтра відповідно [8].

Визначимо крос-кореляційну $r_{sx}(j)$ та крос-бікореляційну $t_{sx}(j,k)$ функції між $x(n)$ та $s(n)$ наступним чином:

$$r_{sx}(j) = E[s(n)x(n-j)],$$

$$t_{sx}(j,k) = E[s(n)x(n-j)x(n-k)]. \quad (3.8)$$

Оскільки передбачається, що $s(n)$ і $x(n)$ строго стаціонарні, як $r_{sx}(j)$, і $t_{sx}(j,k)$ є незалежними від змінної n . Крім того, крос-бікореляційна функція є симетричною: $t_{sx}(j,k) = t_{sx}(k,j)$. Також можна визначити кроскореляційну та кросс-бікореляційну функції в матричному вигляді:

$$R_{sx} [r_{sx}(0), \dots, r_{sx}(N-1)]^T$$

$$T_{sx} = \begin{bmatrix} t_{sx}(0,0) & \dots & t_{sx}(0,N-1) \\ \dots & \dots & \dots \\ t_{sx}(N-1,0) & \dots & t_{sx}(N-1,N-1) \end{bmatrix}$$

Отже, видно, що лінійний та квадратичний оператори ФВ2 з мінімальною СКОШ повинні задовольняти наступним математичним співвідношенням:

$$E[X(n)S(n)] = E[X(n)A^T X(n) + X(n)t_r \{B(X(n)X^T(n) - R_x)\}] \quad (3.9)$$

$$E[X(n)X^T(n)S(n)] = E[X(n)X^T(n)A^T X(n) + X(n)X^T(n)t_r \{B(X(n)X^T(n) - R_x)\}] \quad (3.10)$$

Після деяких перетворень виходить, що лінійний та квадратичний оператори фільтра визначаються так:

$$A_0 = R_x^{-1} R_{sx},$$

$$B_0 = \frac{1}{2} R_x^{-1} T_{sx} R_x^{-1}. \quad (3.11)$$

З (11) видно, що лінійний оператор оптимального ФВ2 це те саме, що і оптимальний лінійний фільтр. Отже, можна сконструювати ФВ2 просто за допомогою додавання квадратичного фільтра паралельно створеному лінійному фільтру без його зміни. Таким чином, процедура реалізації ФВ2 спрощується без суттєвої втрати точності.

4 ОГЛЯД МАТЕМАТИЧНИХ ІНСТРУМЕНТІВ

У цьому розділі наведено огляд основних інструментів, які використовуються в цій роботі. Основні інструменти для моделювання нелінійних систем з одним входом і одним виходом (SISO).

4.1 Нелінійні системи

Поняття системи можна визначити, як у [3].

Система - це комбінація компонентів, які діють разом для виконання певної мети.

Систему можна розуміти як ізольовану частину Всесвіту, яка цікавить нас, а інші частини Всесвіту, які взаємодіють з системою, складають середовище системи, або сусідню систему [3].

Більшість оточуючих нас систем нелінійні. Це означає, що відносини між цими компонентами мають нелінійний характер.

Усі існуючі системи змінюються в часі, і коли темпи змін значні, системи називаються динамічними [3]. Практично це означає, що поточна поведінка системи залежить від її попередньої поведінки або, іншими словами, залежить від стану системи. Більшість систем є нелінійними динамічними системами.

4.2 Математичні моделі нелінійних динамічних систем

Існує більшість методик математичного моделювання нелінійних систем. Математична модель має описати основні особливості системи. Для моделей, які розглядаються в роботі, це особливості, важливі для проектування систем керування.

Модель системи — це форма абстрактних описів зв'язків, що існують

між системними змінними [3].

Коли ці відносини математично виражені нелінійними функціями, модель називається нелінійною математичною моделлю.

Існують моделі безперервного і дискретного часу. У випадку моделей безперервного часу вихід і внутрішні стани залежать від часу і можуть бути розраховані в кожній точці безперервного часу. У випадку моделей з дискретним часом вихідні та/або внутрішні стани можуть бути розраховані лише в певні моменти часу. Період часу між цими випадками називається часом вибірки.

Моделі безперервного часу можна виразити як

$$\hat{y}(t) = f(x_1(t), \dots, x_n(t), u(t)), \quad (4.1)$$

де t є безперервний час;

$\hat{y}(t)$ - вихід моделі;

$u(t)$ - вхід моделі;

$x_1(t), \dots, x_n(t)$ - внутрішній стан моделі

n - порядок моделі і

f - нелінійна функція.

На відміну від безперервних сигналів, в системах з дискретним часом сигнали мають дискретні значення амплітуди в дискретний момент часу.

Зазвичай проводиться відбір проб періодично з часом відбору проб Δt [7]. У цій роботі розглядаються лише системи дискретного часу з постійним часом дискретизації.

Модель дискретного часу:

$$\hat{y}(t_{k+1}) = f(x_1(t_k), \dots, x_n(t_k), u(t_k)), \quad k = 0, 1, \dots \quad (4.2)$$

Тут

t_k - момент часу, вимірний часом вибірки

$$\Delta t = t_{k+1} - t_k \text{ для } k=0,1,\dots \quad (4.3)$$

Для простоти рівняння (2.2) можна переписати у вигляді

$$\hat{y}(k+1) = f(x_1(k), \dots, x_n(k), u(k)), k=0,1,\dots \quad (4.4)$$

або

$$\hat{y}(t+1) = f(x_1(t), \dots, x_n(t), u(t)), t=0,1,\dots \quad (4.5)$$

Іноді позначення (4.5) також використовується в літературі і буде використовуватися в цій роботі, коли ми говоримо лише про системи з дискретним часом, які необхідно згадати.

Існує кілька типів моделей безперервного та дискретного часу. Вибір типу моделі залежить від наявних даних та інформації про процес або систему, яку ми повинні моделювати. Це також дуже залежить від програми, де ця модель буде використовуватися, як це буде показано в цій роботі пізніше.

У наведених нижче системах, які можна моделювати за допомогою кінцевої кількості диференціальних або різницевих рівнянь, розглядаються. Найважливішими типами нелінійних моделей, які використовуються в керуючих програмах, є моделі введення-виведення та простору станів.

Для простоти розглянемо системи з одним входом і одним виходом (SISO). Системи множення вхідних і вихідних даних будуть розглянуті в кваліфікаційній роботі пізніше.

4.2.1 Моделі введення-виведення

Загалом, зв'язки між вхідними сигналами u та вихідними сигналами y

системи можуть бути представлені диференціальним рівнянням n -го порядку

$$f\left(y, \frac{dy}{dt}, \dots, \frac{dy}{dt}, u, \frac{du}{dt}, \dots, \frac{du}{dt}\right) = 0, \quad (4.6)$$

де f – нелінійна функція і $m \leq n$

або за різницеvim рівнянням n -го порядку

$$f(y(k), \dots, y(k-n), u(k), \dots, u(k-m)) = 0, \quad (4.7)$$

де f також нелінійна функція і $m \leq n$.

Рівняння (4.6) називають вхідно-вихідною моделлю нелінійної системи безперервного часу, а рівняння (4.7) — моделлю введення-виходу нелінійної системи з дискретним часом [3]-[6].

4.2.2 Моделі простору стану

Іншим важливим класом моделей є моделі простору станів. Багато методів керування засновані на цьому представленні простору станів, яке складається з рівнянь стану та вихідних рівнянь. [3]-[6]

У випадку нелінійної моделі простору станів безперервного часу динаміка система моделюється кінцевим числом зв'язаних звичайних диференціалів першого порядку рівняння

$$\begin{aligned} \dot{x}_1 &= f_1(t, x_1, \dots, x_n, u) \\ \dot{x}_2 &= f_2(t, x_1, \dots, x_n, u) \\ &\dots \\ \dot{x}_n &= f_n(t, x_1, \dots, x_n, u) \end{aligned} \quad (4.8)$$

Де \dot{x}_i позначає похідну від x_i щодо змінної часу t , а u - вхід системи,

f_1, f_2, \dots, f_n — нелінійні функції. Змінні x_1, x_2, \dots, x_n називаються змінними стану, а n є порядком моделі.

Вихідне рівняння є статичною нелінійною функцією, яку можна визначити наступним чином

$$y = h(t, x_1, \dots, x_n, u) \quad (4.9)$$

Рівняння (4.8) і (4.9) називаються нелінійною моделлю простору станів безперервного часу.

Модель простору станів із дискретним часом також складається з рівнянь стану та вихідного рівняння. Динаміка моделі представлена кінцевою кількістю різницевих рівнянь першого порядку

$$\begin{aligned} x_1(k+1) &= f_1(k, x_1(k), \dots, x_n(k), u(k)) \\ x_2(k+1) &= f_2(k, x_1(k), \dots, x_n(k), u(k)) \\ &\dots \\ x_n(k+1) &= f_n(k, x_1(k), \dots, x_n(k), u(k)) \end{aligned} \quad (4.10)$$

і вихід рівняння є

$$y(k) = h(k, x_1(k), \dots, x_n(k), u(k)) \quad (4.11)$$

Рівняння (4.10) і (4.11) називають нелінійною моделлю простору станів дискретного часу.

4.2.3 Моделі NARX

Нелінійні авторегресивні екзогенні моделі (NARX) є підкласом моделей введення-виведення дискретного часу (4.7). Модель NARX представлена різницеvim рівнянням високого порядку

$$y(k) = f(y(k-1), \dots, y(k-n), u(k-1), \dots, u(k-n)), \quad (4.12)$$

де n - порядок моделі.

Це дуже поширена модель, яка може моделювати дуже великий клас нелінійних систем з високим ступенем точності, але вона має ряд серйозних недоліків. А саме, це не завжди реалізується в класичній формі простору станів або це уявлення дуже складне. Його також не завжди можна лінеаризувати за допомогою зворотного зв'язку. [8], [9], [10] Друга властивість дуже важлива для проектування нелінійної системи керування і буде більш детально розглянута в розділі 4.4.3.

Адитивна модель NARX (ANARX) була запропонована в [11] для подолання розриву.

4.2.4 Моделі ANARX

На відміну від моделей NARX, моделі ANARX мають розділені всі екземпляри часу [8], [11], [33]. Ідея розділення екземплярів часу була запропонована в [12]. Модель ANARX описується наступним рівнянням

$$y(t) = f_1(y(t-1), u(t-1)) + \dots + f_n(y(t-n), u(t-n)) = \sum_{i=1}^n f_i(y(t-i), u(t-i)) \quad (4.13)$$

Тут f_1, \dots, f_n – нелінійні функції, а n – порядок нелінійної моделі (довжина пам'яті).

Однією з основних переваг цієї моделі є те, що вона добре підходить для представлення штучними нейронними мережами структури, зазначеної в [10], як показано в [13], [14]. Модель на основі нейронних мереж буде детально описана в розділі 4.5.2 цієї роботи.

Незалежно від того, як була отримана модель ANARX, шляхом

навчання нейронної мережі чи ні, її завжди можна безпосередньо перезаписати у вигляді моделі простору станів за допомогою рівнянь n станів x_1 до x_n , що відповідає порядку моделі, має бути введено.

$$\begin{aligned}
 x_1(t+1) &= x_2(t) + f_1(x_1(t), u(t)) \\
 x_2(t+1) &= x_3(t) + f_2(x_1(t), u(t)) \\
 &\dots \\
 x_{n-1}(t+1) &= x_n(t) + f_{n-1}(x_1(t), u(t)) \\
 x_n(t+1) &= f_n(x_1(t), u(t)) \\
 y(t) &= x_1(t)
 \end{aligned} \tag{4.14}$$

Представлення в просторі станів є дуже важливим для проектування та аналізу систем керування. Концепція простору станів надзвичайно корисна не лише для розробки конкретної оптимальної системи керування, але й для вдосконалення принципу, за яким буде працювати система.[15]

Іншою перевагою цієї моделі в системах керування є те, що вона завжди лінеаризується за допомогою лінеаризації динамічного вихідного зворотного зв'язку, як це буде показано в розділі 4.4.3.

Алгоритми керування на основі моделі припускають, що модель, отримана за допомогою процедури ідентифікації, є точною. Це означає, що модель з достатньо високим ступенем точності описує властивості системи, важливі для управління нею. Якість системи управління на основі моделі дуже залежить від якості моделі. Для оцінки точності моделі можна використовувати наступний підхід.

4.3 Точність моделі

Модель — це функція, яка відображає вхідне значення u з набору системних входів U у набір вихідних даних моделі \hat{Y} :

$$u \xrightarrow{\text{model}} \hat{y}, u \in U, \hat{y} \in \hat{Y} \quad (4.15)$$

Вихідне значення \hat{y} з множини \hat{Y} ставиться у відповідність кожному входу u з множини U з моделлю. У випадку систем SISO $U, \hat{Y} \subset \mathbb{R}$ і в разі системи MIMO $U \subset \mathbb{R}^r$ і $\hat{Y} \subset \mathbb{R}^m$, де r – кількість вхідних даних моделі та m – кількість вихідних даних моделі.

При цьому кожному входу u відповідає значення y з набору системних виходів Y :

$$u \xrightarrow{\text{system}} y, u \in U, y \in Y \quad (4.16)$$

Зв'язок між системою та моделлю схематично зображено на рисунку 4.1.

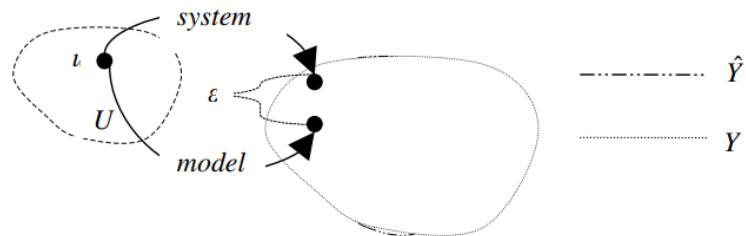


Рисунок 4.1 – Зв'язок між системою та моделлю

$y \neq \hat{y}$ через неточність моделі. Відстань між виходом системи y і виходом моделі \hat{y} є неточністю моделі в точці u і позначається як ε на рисунку 4.1.

$$\varepsilon = y - \hat{y} \quad (4.17)$$

Точність моделі можна виміряти на так званому наборі перевірки. Набір перевірки — це набір відомих пар системного входу та виходу системи. Вихідні дані моделі, що відповідають кожному вхідному значенню з

набору перевірки, обчислюються, і середня квадратична помилка (MSE) тепер може використовуватися як міра точності моделі.

$$MSE = \frac{1}{N} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (4.18)$$

де N – кількість пар введення-виведення в наборі перевірки.

Для систем MIMO MSE можна розрахувати як

$$MSE = \frac{1}{Nm} \sum_{i=1}^N \sum_{j=1}^m (y_{i,j} - \hat{y}_{i,j})^2, \quad (4.19)$$

де m - кількість виходів системи. Кількість виходів моделі дорівнює кількості системних виходів.

4.4 Штучні нейронні мережі

Штучні нейронні мережі складаються з основних вузлів – штучних нейронів. Щоб говорити про штучні нейронні мережі, необхідно визначити позначення штучного нейрона.

4.4.1 Штучні нейрони

У 1909 році Кахал [35] виявив, що мозок складається з великої кількості великої кількості тісно пов'язаних нейронів. У 1943 році Маккаллок і Піттс запропонували математичну модель нейронів і показали, як можна обчислювати нейронні мережі.[24] Спрощена математична модель нейрона, розроблена МакКаллохом і Піттсом, зазвичай називається штучним нейроном. Ця модель використовується в більшості програм на основі нейронних мереж і буде використана в цій роботі. Штучний нейрон

схематично зображений на рисунку 4.5.

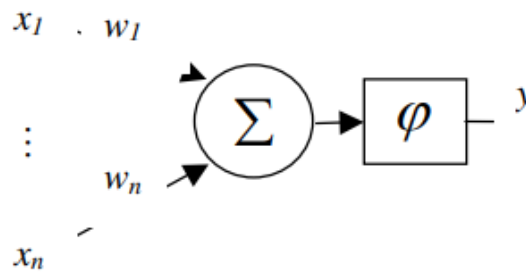


Рисунок 4.5 – Штучний нейрон

З рисунка видно, що штучний нейрон складається з двох частин. Перша частина — це зважена сума вхідних даних, а друга частина — нелінійний елемент називається функцією активації [36], [40], [24]. Вибір цієї функції $\varphi(\cdot)$ суттєво залежить від застосування, стратегії навчання та алгоритму [37].

Розглянемо штучний нейрон, який має n входів, як показано на рисунку. Штучний нейрон математично можна визначити наступною функцією

$$y = \varphi\left(\sum_{i=1}^n w_i \cdot x_i\right) \quad (4.46)$$

або в матричній формі

$$y = \varphi\left([w_1 \dots w_n] \begin{bmatrix} x_1 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix}\right) = \varphi(W \cdot X), \quad (4.47)$$

де X — вектор вхідних даних нейрона, а W — вектор вагових коефіцієнтів, які також називають синаптичними вагами.

Лінійний нейрон без функції активації, представлений рівнянням називається ADaptive LInear NEuron (тут і після ADALINE).

$$y = W \cdot X \quad (4.48)$$

Ще один параметр θ штучного нейрона, який називається зміщенням або порогом, що дає додатковий ступінь свободи мережі, може бути введено в деякі реалізації нейронних мереж, але це також додатковий параметр, який необхідно налаштувати під час навчання мережі. Функція нейрона зі зміщенням має вигляд (4.49)

$$y = \varphi(\sum_{i=1}^n w_i \cdot x_i + \theta) \quad (4.49)$$

4.4.2 Функції активації

Найпоширенішою функцією активації є безперервна функція, яка поступово змінюється між двома асимптотичними значеннями, як правило, 0 і 1, або -1 і 1. Ці функції називаються сигмоїдальними функціями [40].

Визначення сигмоподібних функцій можна дати таким чином

Сигмовидна м'яка являє собою плавну нелінійність з насиченням. Найпоширенішими функціями активації сигмовидної форми є логістична функція та гіперболічний тангенс, представлені рівняннями (4.50) та (4.51). Основна перевага використання цих функцій полягає в тому, що вони завжди диференційовані, і дуже легко та швидко (це дуже важливо для програм керування) обчислити похідні цих функцій [37], як показано нижче.

Логістична функція ($\varphi(I) \in [0;1]$)

$$\varphi(I) = \frac{1}{1+e^{-aI}} \Rightarrow \varphi'(I) = a \cdot \varphi(I) \cdot (1 - \varphi(I)) \quad (4.50)$$

Гіперболічний тангенс ($\varphi(I) \in [-1;1]$)

$$\varphi(I) = \frac{1-e^{-a \cdot I}}{1+e^{-a \cdot I}} \Rightarrow \varphi'(I) = \frac{a}{2} \cdot (1 - \varphi^2(I)) \quad (4.51)$$

Підраховано, що в мозку людини є понад 100 мільярдів (10¹¹) нейронів [41], [37]. У штучних нейронних мереж значно менше нейронів. Найбільші штучні нейронні мережі в керуючих програмах мають не більше кількох сотень нейронів.

Нейрони організовані як природна мережа для отримання інформації з реального середовища, а потім надання відповідної реакції, напр. рішення або дії. Однак з інженерної точки зору нейронну мережу можна розглядати як паралельну систему обробки інформації з деякою людською розумною поведінкою.[37]

Штучну нейронну мережу можна визначити як система обробки даних, що складається з великої кількості простих, тісно взаємопов'язаних елементів обробки (штучних нейронів) в архітектурі, натхненній структурою кори головного мозку [40].

Ці елементи обробки зазвичай організовані в послідовність шарів. Типова нейронна мережа «повністю зв'язана», що означає, що існує зв'язок між кожним із нейронів у будь-якому даному шарі з кожним із нейронів наступного шару.[40]

4.4.3 Персептрон

У 1962 році Розенблатт винайшов клас простих нейроноподібних мереж навчання, який називається персептрон [42], [24]. Персептрон був фактично цілим класом архітектури, який складався з процесорних блоків, які передавали сигнали та адаптували їх ваги з'єднання. Дослідження Розенבלата були орієнтовані на моделювання мозку в спробі зрозуміти пам'ять, навчання та когнітивний процес. Роботи Розенבלата були розширені багатьма вченими та інженерами, і низка машин була побудована на основі архітектури персептрона.[24]

Багатошарові перцептрони дуже швидко стали найбільш поширеними штучними нейронними мережами, особливо в області систем і управління [23]. Структура двошарового перцептрона зображена на рисунку 4.6.

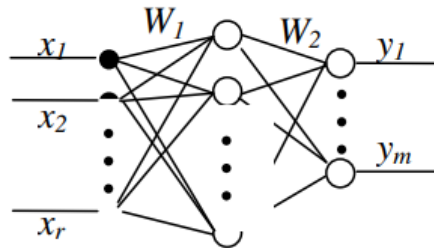


Рисунок 4.6 – Двохрівневий перцептрон

Перший шар (зліва направо) називається вхідним шаром. Метою елементів цього шару є прийом сигналів із середовища та розподіл його між нейронами нейронів шару. Усі шари, які підключені до середовища мережі, називаються прихованими. Приймають сигнали від нейронів попереднього шару і виробляють сигнали для нейронів наступного шару. У розглянутому прикладі другий шар є прихованим шаром. Нейрони останнього шару виробляють виходи мережі. Перцептрон, показаний на рисунку, називається двошаровим (а не тришаровим), оскільки в першому шарі (вхідному шарі) немає обчислень і він не враховується при обчисленні кількості шарів.

Якщо r — кількість мережевих входів, m — кількість мережевих виходів (кількість нейронів вихідного рівня), а l — кількість нейронів прихованого шару, потім $W_1 \in \mathcal{R}^{l \times r}$ є $l \times r$ матриця синаптичних ваг між входами та прихованими нейронів шару і $W_2 \in \mathcal{R}^{m \times l}$ є $m \times l$ матриця синаптичних ваг між прихованим шаром та вихідним. $\theta_1 \in \mathcal{R}^{l \times 1}$ і $\theta_2 \in \mathcal{R}^{m \times 1}$ є векторами зміщень нейронів прихованого вихідного шару. Тепер двошаровий перцептрон можна представити наступною математичною функцією.

$$Y = f_2 (W_2 \cdot f_1 (W_1 \cdot X + \theta_1) + \theta_2), \quad (4.52)$$

де Y — вектор мережових виходів, f_1 і f_2 — функції активації нейронів прихованого та вихідного шарів відповідно.

Слід зазначити, що не всі нейронні мережі (включаючи перцептрони) мають зміщення. Вони дають додаткові параметри і іноді роблять навчання швидшим, але нейронні мережі можуть представляти нелінійності і без них. У цьому випадку функція, що представляє перцептрон

$$Y = f_2 (W_2 \cdot f_1 (W_1 \cdot X)) \quad (4.53)$$

Згідно з теоремою Стоуна-Вейерштрасса, багатошарові перцептрони здатні апроксимувати будь-які безперервні функції.

4.4.4 Теорема Стоуна-Вейерштрасса

Вихідна теорема Вейерстаса показує, що довільна неперервна функція $f : [a, b] \rightarrow \mathfrak{R}$ може бути рівномірно апроксимована послідовністю поліномів $\{p(x)\}$ з точністю до потрібної. Цю теорему проаналізовано за Стоуна [44], який намагався знайти загальні властивості апроксимуючих функцій, а не необхідних поліномів [17]. Теорема Стоуна-Вейерштрасса актуальна для апроксимації неперервних функцій штучними нейронними мережами. Щоб сформулювати результати Стоуна, необхідно дати деякі визначення.

Початкове формулювання теореми для $f : \mathfrak{R}^r \rightarrow \mathfrak{R}$ через умову (II) в (4.54), але результат залишається справедливим і для $r > 1$, оскільки кодобласть векторнозначної функції є декартовим добутком її компонентів.[17]

Двошаровий перцептрон описується рівнянням (4.52) або (4.53). Для апроксимації нелінійних функцій нейрони прихованого шару повинні мати нелінійну функцію активації. Це означає, що принаймні функція f_1 в (4.52) і (4.53) є нелінійною функцією. Функція f_2 зазвичай є лінійною функцією (у

цьому випадку виходи мережі не обмежені). Коли f_1 є сигмоїдною функцією, а f_2 є ADALINE, як визначено в (4.48), функція мережі є лінійною комбінацією сигмоїдних кісток.

Множина всіх лінійних комбінацій сигмоїдів є незникаючою алгеброю роздільна точка на компактi $K \subset \mathcal{R}^r$ [17] і відповідно до теореми 4.1 це тип нейронних мереж підходить для рівномірної апроксимації довільної безперервної карти. Це означає, що будь-яку безперервну нелінійну функцію можна апроксимувати двошаровим персептроном із сигмоподібними функціями активації його прихованого шару та нейронів лінійного вихідного шару з точністю до бажаної точності.

У структурі багатошарового персептрона немає зворотних зв'язків. Такі мережі без внутрішнього зворотного зв'язку між нейронами та/або шарами називаються мережами прямого зв'язку або статичними мережами. Такі мережі можуть апроксимувати лише статичні функції. Для моделювання динамічних систем за допомогою цього типу нейронних мереж необхідний зовнішній зворотний зв'язок.

Рекурентні (також звані динамічними або зворотними) нейронними мережами є альтернативою статичним нейронним мережам.

4.4.5 Рекурентні нейронні мережі

Нейронні мережі цього типу мають внутрішні зворотні зв'язки від виходів нейронів до входів нейронів того ж (або попереднього) шару. Ці мережі не потребують жодних додаткових зовнішніх зворотних зв'язків для моделювання динамічних процесів. Кількість мережевих входів дорівнює кількості входів процесу. Мережа Елмана [45] розглянемо як приклад рекурентних мереж. Можливості цієї мережі досліджуються в [46].

Мережа Елмана зазвичай являє собою двошарову мережу зі зворотним зв'язком від виходу першого рівня до входу першого рівня. Це періодичне з'єднання дозволяє мережі Elman виявляти та генерувати змінювані у часі

шаблони. Двошарова мережа Elman зображена на рисунку 4.7.

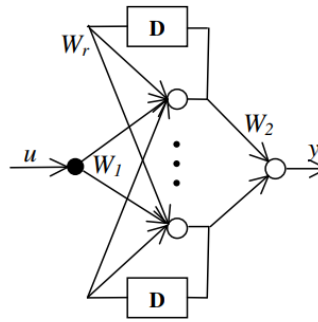


Рисунок 4.7 – Рекурентна мережа Елмана

Мережа Elman, показана в мережі, має один прихований шар, який називається рекурентним. Функція мережі з одним входом і одним виходом задовольняє наступним рівнянням

$$\begin{cases} X(t) = f(X(t-1), u(t)) \\ y(t) = h(X(t)) \end{cases}, \quad (4.55)$$

де $u(t)$ – вхід мережі, $y(t)$ – вихід мережі, а $X(t)$ – вектор вихідних даних повторюваного рівня. З рівняння і рисунка представлених вище видно, що мережа Elman є динамічною мережею. Вихідні значення прихованого (рекурентного) шару зберігаються в пам'яті на один такт (затримується на один часовий крок) і виробляють вектор станів мережі, який впливає на вихід і вектор стану мережі на наступному кроці часу.

Розглянемо мережу, що має r входів, m виходів і l нейронів у своєму повторюваному шар. Тоді мережа має такі параметри: $W_1 \in \mathcal{R}^{l \times r}$ — матриця синаптичних ваг між входами мережі та нейронами повторюваного шару, $W_2 \in \mathcal{R}^{m \times l}$ — це матриця синаптичних ваг між прихованим шаром мережі нейрони та нейрони вихідного шару, $W_3 \in \mathcal{R}^{l \times l}$ — квадратна матриця синаптичних ваг між виходами прихованого шару та входами того самого шару (вага коефіцієнти зворотного зв'язку), $\theta_1 \in \mathcal{R}^{1 \times l}$ — вектор зміщень прихованого шару

і $\theta_2 \in \mathbb{R}^{1 \times m}$ – вектор зміщень вихідного шару. При повторному шарі нейронів мають нелінійну сигмовидну функцію активації $f(\cdot) = \sigma(\cdot)$ і нейрони вихідного шару є лінійними ($h(\cdot)$ — лінійна функція), функцію мережі Елмана можна визначити як (4.56) або (4.57), впливаючи з рівняння (4.55), вводячи параметри мережі.

$$\begin{cases} X(t) = f(W_1 U(t) + W_r \cdot X(t-1) + \theta_1) \\ y(t) = W_2 \cdot X(t) + \theta_2 \end{cases} \quad (4.56)$$

Коли в мережі немає зміщень (або вони дорівнюють нулю), це рівняння приймає такий вигляд

$$\begin{cases} X(t) = f(W_1 U(t) + W_r \cdot X(t-1)) \\ y(t) = W_2 \cdot X(t) \end{cases} \quad (4.57)$$

Мережа Елман здатна апроксимувати будь-які безперервно диференційовані динамічні функції, як це було доведено Sontag у [47].

4.4.6 Теорема Зонтага

Тут коротко викладено основний результат роботи [47].

Теорема 4.2 (Зонтаг) Нехай система задається з $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$, $y \in \mathbb{R}^q$ з f і h безперервно диференційований, такий, що для всіх $u : [0, T] \rightarrow \mathbb{R}^p$ існує розв'язок (4.57) і є єдиним для всіх $t \in [0, T]$ і деяких компактів $K_1 \subset \mathbb{R}^n$, $K_2 \subset \mathbb{R}^p$, тоді як $x \in K_1$, $u \in K_2$.

$$\begin{aligned} \dot{x} &= f(x, u); x(t_0) = x^0 \\ y &= h(x) \end{aligned} \quad (4.58)$$

Тоді існує рекурентна нейронна мережа виду

$$\begin{aligned} \dot{\chi} &= \sigma(A\chi + Bu) \\ y &= C\chi \end{aligned} \quad (4.59)$$

де $\chi \in \mathbb{R}^N$, $N \geq n$, $u \in \mathbb{R}^q$, σ — вектор сигмоїдів, а A , B — матриці такі, що на $K_1 \times K_2$,

$$\forall \varepsilon > 0 \quad \forall t \in [0, T] \quad \|x(t) - M(\chi(t))\| < \varepsilon \quad \text{і} \quad h(x(t)) - C\chi(t) < \varepsilon$$

де M — диференційована карта.[17]

Оскільки рівняння (4.56) і (4.57) задовольняють умову (4.59), можна зробити висновок, що мережа Elman із сигмоїдними функціями рекурентного шару та лінійної

Нейрони вихідного шару здатні апроксимувати безперервно диференційовані динамічні функції, якщо кількість нейронів повторюваного шару більше, ніж порядок системи, яка ідентифікує мережу.

Апроксимація функції нейронною мережею (статична або динамічна) означає обчислення (або коригування) параметрів (синаптичних ваг і зміщень, якщо такі є) мережі. Цей процес називається мережевим навчанням (або навчанням). Розрізняють контрольовані та неконтрольовані алгоритми навчання. Методики навчання під наглядом потребують набору еталонних карт для представлення мережі. У разі неконтрольованого навчання мережа здатна регулювати свої параметри за деякими критеріями. У цій роботі буде розглянуто лише контрольований підхід, оскільки він більше стосується завдань моделювання та контролю.

4.5 Алгоритми навчання

Перший набір ідей навчання в нейронних мережах міститься в книзі Хебба [48]. У 1951 році Едмундс і Мінські побудували свою навчальну машину, використовуючи ідею Хебба. Хоча Мінський був першою людиною, яка запропонувала навчальну машину, справжній початок значущого нейроноподібного мережевого навчання можна простежити в роботі

Розенבלата [49] у 1962 році.[24]

Кероване навчання, яке розглядається в цьому розділі, вимагає «вчителя» або іншого джерела інформації, яке безпосередньо вказує мережі правильну відповідь на кожен визначений стимул, з яким вона зіткнеться. Наприклад, викладач може бути людиною або реальним процесом, який генерує приклади необхідних мережевих виходів, що відповідають визначеному набору вхідних даних.[39]

Навчання — це процес адаптації ваг зв'язків у штучній нейронній мережі для отримання бажаного вихідного вектора у відповідь на вектор стимулу, представлений у вхідному буфері.[40]

Навчання базується на наборі вхідно-вихідних даних

$$Z_N = \{[u(t), y(t)], t = 1, \dots, N\}, \quad (4.60)$$

де N — розмір навчальної множини.

Хоча існує безліч різних методів навчання, у цій роботі використовуються два основних алгоритми навчання. Перший з них — навчальний алгоритм зворотного поширення помилки градієнтного спуску (BP). Розрахунок оновлень мережевих параметрів на кожній ітерації швидкий. Він потребує менших обчислювальних ресурсів, ніж другий, і може використовуватися для он-лайн адаптації моделі. Левенберг-Марквардт (LM) має кращі можливості наближення, йому потрібно менше ітерацій для зближення, але кожна ітерація займає набагато більше часу та обчислювальних ресурсів. Його можна використовувати для офлайнового моделювання.

В обох випадках метою навчання є мінімізувати функцію вартості $F(\Theta, Z^N)$, де Θ — вектор параметрів мережі (синаптичні ваги та деякі програми також зміщення). Середньоквадратична помилка зазвичай використовується як критерій мінімізації.

$$F(\theta, Z_N) = \frac{1}{2N} \sum_{t=1}^N (y(t) - \hat{y}(t))^2 = \frac{1}{2N} \sum_{t=1}^N e^2(t, \theta) \quad (4.61)$$

Тут $y(t)$ – відоме вихідне значення з навчальної множини (4.60), що відповідає вхід $u(t)$. Використовується як значення еталону («учитель»). $\hat{y}(t)$ є результатом мережа (оцінений вихід), яка має бути якомога ближче до $y(t)$.

Під час тренувальних процедур показані в двох наступних розділах

$$F(\Theta, Z_N) \rightarrow 0 \quad (4.62)$$

4.5.1 Зворотне поширення помилки градієнтного спуску

Цей алгоритм є методом першого порядку. Градієнт визначається як

$$G(\theta) = \frac{\partial F_n(\theta, Z_n)}{\partial \theta} = \frac{1}{N} \sum_{t=1}^N \frac{\partial e(t, \theta)}{\partial \theta} e(t, \theta) \quad (4.63)$$

При застосуванні алгоритму ВР [17] вектор параметрів мережі оновлюється на кожній ітерації як

$$\Theta(k+1) = \Theta(k) - \lambda(k) \cdot G(\Theta), \quad (4.64)$$

де $\lambda(k)$ – це адаптивний адаптивний коефіцієнт, який називається коефіцієнтом навчання або швидкість навчання.

Одну ітерацію навчального алгоритму також називають епохою або епохою навчання.

4.5.2 Алгоритм Левенберга-Марквардта

Алгоритм LM [50], [51] є сумішшю градієнтного спуску та ітерації

Гауса-Ньютона. Він також надає розв'язок нелінійної задачі мінімізації найменших квадратів (4.61), (4.62). Гессен визначається за

$$H(\theta) = \frac{\partial^2 F(\theta, Z_n)}{\partial \theta \partial \theta^T} = \frac{1}{N} \sum_{t=1}^N \frac{\partial e(t, \theta)}{\partial \theta} \left[\frac{\partial e(t, \theta)}{\partial \theta} \right]^T - \frac{1}{N} \sum_{t=1}^N \frac{\partial^2 e(t, \theta)}{\partial \theta \partial \theta^T} e(t, \theta) \quad (4.65)$$

Правило оновлення в методі LM є

$$\Theta(k+1) = \Theta(k) + \Delta \Theta(k) \quad (4.66)$$

Де

$$[R(\Theta(k)) + \lambda I] \Delta \Theta(k) = -G(\Theta(k)) \quad (4.67)$$

Тут $G(\Theta(k))$ — градієнт, визначений (4.63), I — ідентична матриця, а λ — коефіцієнт адаптивного навчання. Цей коефіцієнт врівноважує поведінку методу між другим і першим порядком. [52], [53]

У [54] було детально вивчено, що швидкість навчання цього алгоритму набагато вища, і нейронна мережа з прямим зв'язком, навчена за алгоритмом LM, може краще моделювати нелінійності. У [53] також встановлено, що цей алгоритм є набагато ефективнішим, ніж будь-який з інших методів, коли мережа містить не більше кількох сотень ваг.

5 НЕЛІНІЙНІ ДИНАМІЧНІ СИСТЕМИ ІДЕНТИФІКАЦІЇ ЗІ ШТУЧНОЮ НЕЙРОННОЮ СИСТЕМОЮ МЕРЕЖІ

Ідентифікація нелінійних динамічних систем зі штучними нейронними мережами

З математичної точки зору ідентифікація нелінійних систем означає апроксимацію нелінійної карти (нелінійної функції або функцій). Стоуном і Зонтагом було доведено, що штучні нейронні мережі здатні апроксимувати будь-яку безперервну нелінійність. Саме тому нейронні мережі стали привабливим інструментом у побудові моделей складних нелінійних процесів і було запропоновано велику кількість ідентифікаційних структур на основі нейронних мереж.

Застосування рекурентних (динамічних) нейронних мереж та структур на основі нейронних мереж із прямим зв'язком із зовнішнім зворотним зв'язком для ідентифікації нелінійних динамічних систем буде розглянуто в цьому розділі. Також будуть показані деякі специфічні для програми структури мережі з прямим зв'язком. Буде розглянуто ідентифікацію нелінійних систем SISO та MIMO.

5.1 Рекурентні моделі на основі нейронних мереж

Більшість систем і процесів реального світу мають динамічну природу, і тому динамічна (рекурентна) нейронна мережа видається розумним вибором для ідентифікації нелінійних динамічних систем. Мережа Елмана, є однією з найпростіших і поширених рекурентних повнозв'язаних мережевих структур.

Модель мережі Елмана, може моделювати нелінійну систему після навчання на відповідному навчальному наборі, що складається з пар вхід-вихід. Набір вхідних даних, наданих системі, визначається як

$$U_N = [U(t - N), U(t - N + 1), \dots, U(t)], \quad (5.1)$$

де $U(t)$ — вектор вхідних даних системи. Кількість входів мережі дорівнює кількості входів системи r .

$$U(t) = \begin{bmatrix} u_1(t) \\ \vdots \\ u_r(t) \end{bmatrix} \quad (5.2)$$

Вхідні дані повинні вимірюватися безперервно з постійним часом вибірки, а також відповідні вихідні дані.

$$Y_N = [Y(t - N), Y(t - N + 1), \dots, Y(t)], \quad (5.3)$$

де $Y(t)$ - вихідний вектор на етапі часу t

$$Y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_m(t) \end{bmatrix}, \quad (5.4)$$

де m - кількість системних виходів.

Усі значення з необхідні і мають бути представлені мережі в правильному порядку під час процедури навчання.

В першу чергу параметрам мережі задаються випадкові значення. Застосовується алгоритм ітераційного навчання і на кожній ітерації k обчислюється вихід мережі (вихід моделі) $\hat{Y}(k)$ за (2.57) і порівнюється з виходом еталону Y_N . Середньоквадратична похибка обчислюється за (2.61) і процес повторюється доти менше заданого числа (бажана точність моделі).

$$\frac{1}{2N \cdot m} \sum_{i=1}^m \sum_{t=1}^N (y_i(t) - \hat{y}_i(t))^2 < \varepsilon \quad (5.5)$$

Відповідно до теореми Зонтага (п. 2.5.6) нейронна мережа кількість прихованих нейрони шару $n > r$ необхідно вибрати для отримання моделі нелінійної динаміки системи з будь-якою бажаною точністю. Практичні проблеми, які виникають при застосуванні цього підходу, описані в [26] та [58].

Повторювані мережі не настільки надійні, як мережі з прямим зв'язком, оскільки навчання відбувається з використанням наближення градієнта помилки. Внесок ваг і зміщень до помилок через відкладене повторне з'єднання ігнорується. В іншому випадку навчання рекурентних мереж стає надзвичайно складним.

Мережа прямого зворотного зв'язку із зовнішніми зворотними зв'язками є хорошою альтернативою рекурентним мережам.

5.2 Нейронні мережі прямого зв'язку із зовнішнім зворотним зв'язком для ідентифікації динамічних систем. Моделі NN-NARX

Зовнішнього зворотного зв'язку достатньо для представлення всіх динамічних систем.[38]

Модель введення-виведення дискретного часу (2.7) може бути реалізована у вигляді прямої нейронної мережі із зовнішнім зворотним зв'язком і затримками. Статична функція $f(\cdot)$ можна апроксимувати мережею прямого зв'язку з точністю до будь-якого ступеня точність за теоремою Стоуна-Вейерштрасса і динамічні аргументи цієї функції повинні бути подані на входи мережі. Відкладені вхідні та вихідні значення (від зовнішнього зворотного зв'язку) надходять на додаткові входи мережі. Для реалізації

моделі (2.7) мережа повинна мати $n+m+1$ входів. Приклад динамічної моделі SISO, реалізованої у вигляді прямої нейронної мережі із зовнішнім зворотним зв'язком, зображено на рисунку 5.1.

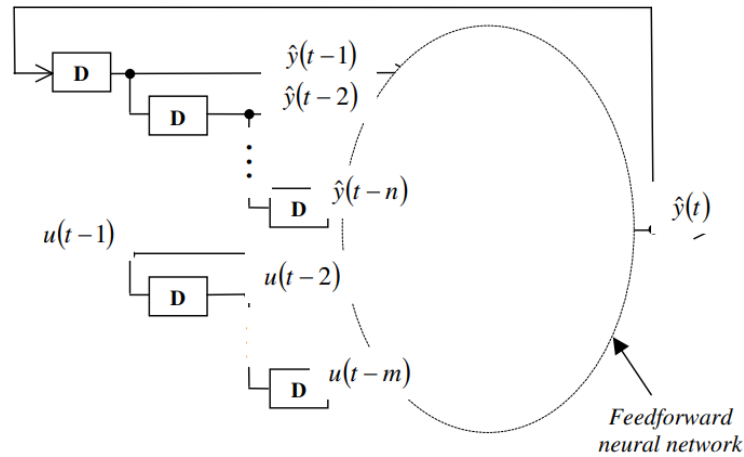


Рисунок 5.1 – Представлення динамічної моделі за допомогою прямої нейронної мережі

Нейронні мережі з обмеженим зворотним зв'язком, який надходить лише від вихідного нейрона, а не від прихованих станів, називаються нелінійними авторегресивними екзогенними мережами або мережами NARX [61], [62], [63]. Вони формалізовані шляхом

$$y(t) = \psi (y(t - 1), \dots, y(t - n), u(t - 1), \dots, u(t - m)), \quad (5.6)$$

де Ψ – відображення, що виконується нейронною мережею з прямим зв'язком (наприклад, багатошаровим персептроном), n – вихідний порядок, а m – порядок введення.

У [64] доведено, що мережі NARX зі скінченною кількістю параметрів є такими ж міцними, як і повністю зв'язані рекурентні мережі, а отже, і машини Тьюрінга.

Моделі нелінійних систем на основі цих мереж називаються моделями

NARX на основі нейронних мереж або моделями NN-NARX. У [64] також показано, що теоретично можна використовувати моделі NARX, а не звичайні рекурентні мережі без будь-яких обчислювальних втрат, навіть якщо їх зворотний зв'язок обмежений.

Коли двошаровий перцептрон з нелінійними функціями активації нейронів прихованого шару та lin-rear виходом використовується як пряма мережа в цій моделі, це може бути виражено наступним рівнянням

$$\hat{y}(t) = C \cdot \varphi(W \cdot [\hat{y}(t-1), \dots, \hat{y}(t-n), u(t-1), \dots, u(t-m)]), \quad (5.7)$$

де $\hat{y}(t)$ – вихід мережі (оцінений вихід), розрахований за допомогою моделі на кроці часу t ; $\varphi(\cdot)$ – нелінійна функція активації нейронів прихованого шару; C — вектор синаптичних ваг нейрона лінійного виходу, а W — матриця синаптичних ваг нейронів прихованого шару.

Коли мережа має 1 нейронів у прихованому шарі, $W \in \mathcal{R}^{1 \times (n+m)}$ і $C \in \mathcal{R}^{1 \times 1}$. Це означає, що параметри $1 \cdot (n + m) + 1 = 1 \cdot (n + m + 1)$ мають бути налаштовані за допомогою навчального алгоритму для моделювання динамічної системи.

Якщо система МІМО з r входами та k виходами моделюється двошаровим перцептроном із зовнішнім зворотним зв'язком, то $W \in \mathcal{R}^{1 \times (k \cdot n + r \cdot m)}$, $C \in \mathcal{R}^{k \times 1}$ і вже $1 \cdot (k \cdot n + r \cdot m) + k \cdot 1 = 1 \cdot (k \cdot (n + 1) + r \cdot m)$ синаптичні ваги мають бути налаштовані.

На відміну від ідентифікації за допомогою повторюваних нейронних мереж, перестановки в навчальному наборі дозволені під час навчання, коли використовується мережа прямого зв'язку. Можливі перестановки між вхідними векторами (не між елементами в кожному векторі), оскільки вся інформація про динаміку процесу представлена в кожному вхідному шаблоні.

Використовуючи цей підхід, модель нелінійної динамічної системи на

основі нейронної мережі можна дуже легко отримати шляхом навчання мережі з прямим зв'язком. Розглянемо наступний приклад.

5.3.1 Числовий приклад 5.1

Нелінійна система [59], [60], яку потрібно ідентифікувати, представлена наступним рівнянням введення-виведення дискретного часу

$$y(t) = \frac{1.5y(t-1)y(t-2)}{1+y^2(t-1)+y^2(t-2)} + 0.3 \cos(0.5(y(t-1) + y(t-2))) + 1.2u(y-1) + e(t), \quad (5.8)$$

де системний вхід $u(t)$ є рівномірно розподіленим сигналом у діапазоні $[-1, 1]$, і послідовність шумів $e(t) \sim N(0, 0,12)$. Послідовність даних із 1000 введених даних

Були створені вибірки $u(t)$ і вихідних $y(t)$ і навчальна множина (5.9), (5.10) що складається з вхідних векторів U_t і відповідні виходи Y_t було вироблено.

$$U_t = \begin{bmatrix} y(999) & y(998) & y(2) \\ y(998) & y(997) & y(1) \\ u(999) & u(998) & u(2) \end{bmatrix} \quad (5.9)$$

$$Y_t = [y(1000) \ y(999) \ \dots \ y(3)] \quad (5.10)$$

Цей набір даних $\{U_t, Y_t\}$ був використаний для навчання двошарового перцептрона з трьома входами та одним вихідом. Найкраща точність моделі була досягнута, коли прихований шар складався з 15 нейронів. Таким чином, 60 синаптичних ваг і 16 зміщень (взято разом 76 параметрів) були налаштовані за допомогою алгоритму навчання АТ. Після навчання модель виходить у

наступному вигляді

$$\hat{y}(t) = C \cdot \varphi (W \cdot [\hat{y}(t-1), \hat{y}(t-2), u(t-1)]), \quad (5.11)$$

де $\varphi(\cdot)$ – функція активації гіперболічного тангенса (2.51) нейронів прихованого шару.

Для перевірки моделі було створено інший набір даних $\{U_v, Y_v\}$. Мережа моделюється за допомогою входів U_v і виходів, розрахованих мережею \hat{y} порівняно з відповідними виходами з набору перевірки Y_v . Для перевірки моделі використовувався синусоїдальний вхідний сигнал $u_v(t) = \sin(0.2t)$. Порівняння виходу системи з виходом моделі (нейронної мережі) показано на рисунку 5.2.

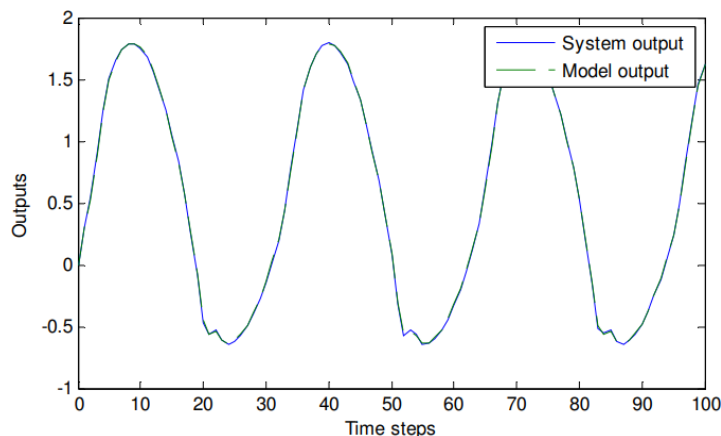


Рисунок 5.2 – Вихід системи та вихід моделі

У цьому експерименті система моделюється без шуму, і можна побачити, що вихідні дані збігаються, навіть якщо модель не отримує жодної інформації від реальної системи. Середньоквадратична помилка моделі (2.18) дорівнює $MSE \approx 1 \cdot 10^{-4}$. Отже, двошаровий перцептрон із зовнішньою динамікою (зворотний зв'язок і затримки) здатний представляти цю нелінійну систему з дуже високим ступенем точності. Він також здатний прибрати шум

із встановленої дати. Ось чому моделі на основі нейронних мереж стали настільки популярними в ідентифікації нелінійних систем і достатньо точними, щоб їх використовувати в структурах керування на основі моделі. Хоча моделі NARX, засновані на багат шарових перцептронах, мають дуже хороші можливості апроксимації, практика показує, що більш конкретні структури залежно від конкретних застосувань також можуть мати велику цінність. При побудові нейронної мережі людина має свободу у виборі функцій активації нейронів і у визначенні зв'язків між нейронами. Для кращої якості ідентифікації деяких конкретних нелінійних систем можна використовувати нейронну мережу зі структурою, яка краще представляє структуру системи. Структура мережі також може залежати від потреб алгоритму керування. Приклади таких мережевих структур будуть розглянуті в наступних частинах цієї глави.

Специфічна структура нейронної мережі для ідентифікації нелінійних систем з нелінійністю актуаторів [55].

5.3 Модель Хаммерштейна на основі нейронної мережі

Одним з найпоширеніших класів моделей є моделі Гаммерштейна [65], [66], представлені на рисунку 5.3.

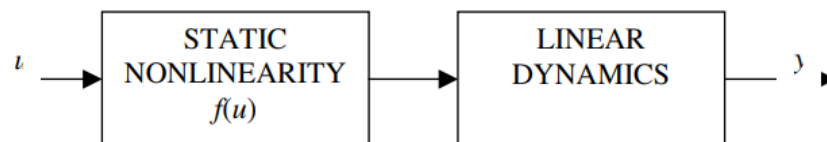


Рисунок 5.3 – Модель Гаммерштейна

Моделі лінійних динамічних систем зі статичними нелінійністю приводів належать до класу моделей Гаммерштейна. Статичні нелінійності актуаторів суттєво впливають на продуктивність адаптивних та навчальних

систем керування. У багатьох випадках керовані системи можна апроксимувати лінійними динамічними функціями з високою точністю, а керування на основі моделі може бути дуже легко реалізовано. Тим не менш, актуатори роблять ці системи нелінійними через неточність механічних компонентів і природи фізичних законів. Таким чином, актуатори вносять нелінійність у систему, коли вона складається з двох частин: пристрою, яким ми повинні керувати, і приводу. Статичні нелінійності приводу, такі як межі насичення та швидкості погіршують продуктивність систем керування, особливо систем керування на основі моделей, коли вимога керування є високою.[67]

Хоча нелінійність приводу часто присутня, більшість методів проектування керування ігнорують їх. Це особливо проблематично в адаптивному управлінні.[68] Нелінійність приводу не можна скасувати або компенсувати за допомогою методів лінеаризації зворотного зв'язку, оскільки вони не з'являються в шляху зворотного зв'язку.[69] Це головне, чому ці нелінійності не можуть бути апроксимовані стандартними методами. Точність моделей, отриманих шляхом навчання стандартної єдиної нейронної мережі, також є низькою. Мертва зона та насичення називаються жорсткими нелінійністю.[69] Це ускладнює впровадження алгоритмів керування на основі моделі в системах з нелінійністю приводу. Ця проблема була вирішена шляхом введення спеціального типу нейронних мереж для ідентифікації цього типу нелінійності в [55].

Розглянуто системи, що мають лінійну динамічну поведінку з приводними пристроями, що мають статичну нелінійність. Іншим важливим фактом, який ускладнює ідентифікацію таких систем, є те, що продуктивність приводу не піддається вимірюванню. Ось чому привод і систему не можна ідентифікувати окремо.

У зазначеному вище випадку ми маємо вимірювані вхідні та вихідні дані приводу системи. Система між ними складається з двох частин: статичної нелінійної частини та динамічної лінійної частини. Вхідна

нелінійність є статичною і не відображається у зворотному зв'язку. Отже, під час процедури ідентифікації важливо розділити ці дві частини.

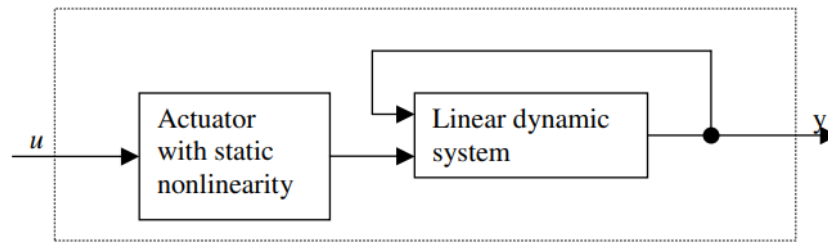


Рисунок 5.4 – Система з нелінійністю статичного приводу

Для ідентифікації систем, що мають цей тип нелінійності, запропоновано структуру нейронної мережі, що складається з двох підмереж. Структура мережі показана на рисунку 5.5. Мережа складається з двох підмереж. Перша підмережа - це статична мережа прямого зв'язку. Він має один вхід і один вихід, але він повинен наближатися до нелінійної функції актуатора і, отже, має нелінійні функції активації нейронів у своєму прихованому шарі. Відповідно до теореми Стоуна-Вейрштрассе вона здатна апроксимувати будь-яку безперервну нелінійність приводу. Вихід першої підмережі служить входом для другої підмережі. Другий підмережа динамічна. Він має зовнішній зворотний зв'язок від свого виходу, а також минулі вхідні та вихідні значення на своїх додаткових входах, як показано на рисунку 5.4. Кількість додаткових входів (довжина пам'яті) залежить від порядку моделі. Друга підмережа використовується для ідентифікації лінійної динамічної частини і, отже, має функції лінійної активації всіх її нейронів. Він здатний апроксимувати будь-яку лінійну динамічну систему. Цю мережу можна розглядати як систему, що складається з двох персептронів, які необхідно навчати разом, щоб розділити лінійну динаміку та нелінійну статику для кращої якості ідентифікації систем типу Гаммерштейна.

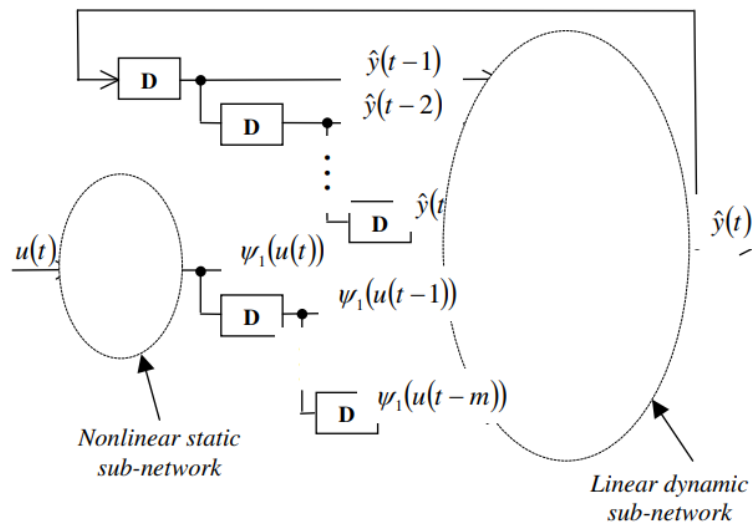


Рисунок 5.5 – Структура моделі Хаммерштейна на основі нейронної мережі

Модель на основі нейронної мережі, показана на рисунку 5.5, належить до класу моделей Хаммерштейна. Її можна формалізувати за допомогою

$$y(t) = \psi_2 (y(t-1), \dots, y(t-n), \psi_1 (u(t-1)), \dots, \psi_1 (u(t-m))), \quad (5.12)$$

де Ψ_1 – відображення, що виконується нелінійною підмережею, а Ψ_2 – це відображення, що виконується лінійною підмережею. Коли для виконання цих відображень використовуються двошарові перцептрони, модель (5.12) виражається наступним рівнянням

$$\hat{y}(t) = C_2 \cdot W_2 \cdot [\hat{y}(t-1), \dots, \hat{y}(t-n), C_1 \varphi (W_1 \cdot u(t-1)), \dots, C_1 \varphi (W_1 \cdot u(t-m))]^T, \quad (5.13)$$

де W_1 — вектор синаптичних ваг прихованого шару першої (нелінійної) підмережі, C_1 — вектор синаптичних ваг вихідного шару першої підмережі, W_2 — матриця синаптичних ваг прихованого шару другої (лінійної) підмережі, C_2 є вектором синаптичних ваг вихідний шар другої підмережі і φ

є нелінійною функцією активації нейронів прихованого шару першої підмережі.

Ідентифікаційні можливості цієї структури продемонстровані на наступних прикладах.

5.4 Числовий приклад 5.2

Розглянемо наступну лінійну установку з дискретним часом [7]

$$H(z) = \frac{z+0.5}{z^2-1.5z+0.7} \quad (5.14)$$

Лінійна система дискретного часу (5.14) з несиметричним насиченням і нелінійністю мертвої зони (5.15) на її вході була використана як тестова система в [55].

Нелінійності мертвої зони та насичення визначаються наступними рівняннями

$$s(u) = \begin{cases} s_{\max}, & \text{if } K \cdot (u - u_{dz_max}) \geq s_{\max} \\ K \cdot (u - u_{dz_max}) & \text{if } (K \cdot (u - u_{dz_max}) \leq s_{\max}) \& (u \geq u_{dz_max}) \\ K \cdot (u - u_{dz_min}) & \text{if } (K \cdot (u - u_{dz_min}) \geq s_{\min}) \& (u \leq u_{dz_min}) \\ d & \text{if } u_{dz_min} < u < u_{dz_max} \\ s_{\min}, & \text{if } K \cdot (u - u_{dz_min}) \leq s_{\min} \end{cases} \quad (5.15)$$

Параметри вхідної нелінійності (5.15), використані в експерименті, є $S_{\max} = 1.1$, $S_{\min} = -0.8$, $u_{dz_min} = -0.3$, $u_{dz_max} = 0.2$, $d = 0.1$ та $K = 1$.

Ця система, що складається з лінійної установки та нелінійного приводу, була змодельована з рівномірним розподіленим сигналом у

діапазоні $[-1.5, 1.5]$ і була створена послідовність даних із 1000 вибірок введення-виведення $\{u(t), y(t)\}$ для отримання тренувальний набір.

Перш за все, модель системи (5.14), (5.15) була отримана шляхом навчання звичайної двошарової мережі NARX на основі персептрона (5.11). Найкраща точність моделі була отримана, коли прихований шар складався з 20 нейронів з функціями активації гіперболічного тангенса (2.51). Оскільки модель є модель SISO другого порядку (порядок входу 2, а порядок виходу 2) модель NARX має один зовнішній зворотний зв'язок від виходу і Нейронна мережа прямого зв'язку має 4 входи. Це означає, що $4 \times 20 + 20 = 100$ синаптичні ваги потрібно було регулювати за допомогою тренувального алгоритму. Через велику кількість параметрів, які потрібно перевернути, мережу навчали за алгоритмом зворотного поширення помилки градієнтного спуску.

Після навчання мережі модель була протестована на наборі даних, що складається з 500 елементів, який не використовувався для навчання. Лінійно зростаючий в діапазоні $[-3, 3]$ вхідний сигнал $u_v(t) = -3 + .012t$, $t = 0, 1, \dots, 500$ використовувався для генерування набору перевірки $\{u_v(t), y_v(t)\}$. Результат перевірки моделі наведено в рисунок 5.6.

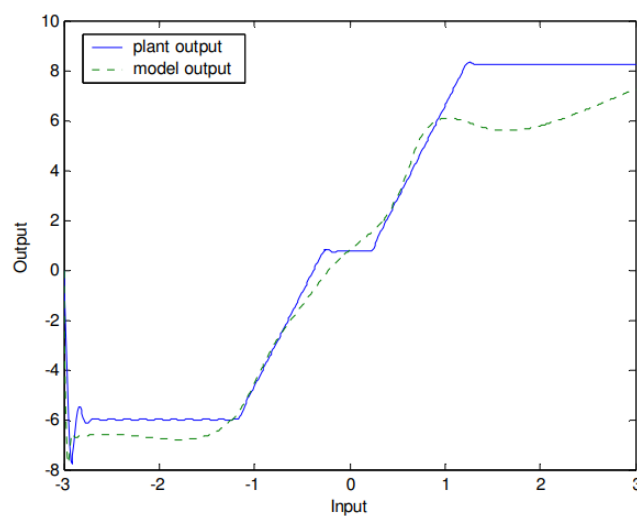


Рисунок 5.6 – Ідентифікація за допомогою одного двошарового персептрона

Середньоквадратична помилка на наборі перевірки в цьому експерименті становить $MSE \approx 1,749$.

Після цього модель була отримана шляхом навчання мережі (5.13), що складається з нелінійних статичних і лінійних динамічних підмереж, запропонованих у [55] і зображених на рисунку 5.5. Був застосований той самий алгоритм навчання.

Найкраща точність була досягнута шляхом навчання мережі з 6 нелінійними нейронами в прихованому шарі першої підмережі та 3 нейронами в прихованому шарі другої підмережі. Таким чином, кількість синаптичних ваг мережі становить $6 + 6 + 4 \times 3 + 3 = 27$, що майже в 4 рази менше, ніж у попередньому експерименті. Отримана модель була перевірена на тому ж наборі даних, що й попередній. Результат перевірки зображено на рисунку 5.7. У разі використання запропонованої структури мережі для ідентифікації цієї системи $MSE \approx 0,013$.

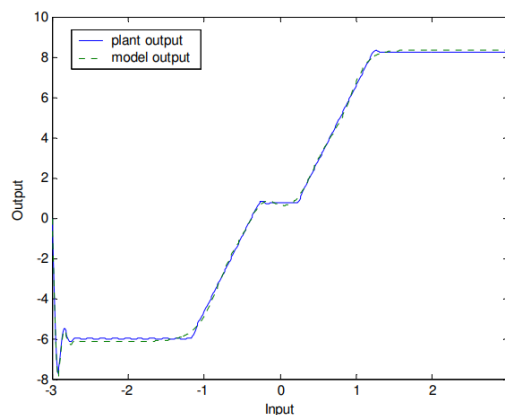


Рисунок 5.7 – Ідентифікація з мережею, що розділяє нелінійну статичну та лінійну динамічну частини

Значне покращення якості ідентифікації можна побачити, порівнявши середньоквадратичні помилки та рисунки 5.6 та 5.7. Значно більш точна модель була отримана шляхом навчання нейронної мережі з майже в 4 рази меншою кількістю синаптичних ваг. Це покращення можна пояснити

розділенням складної моделі на дві прості частини. Отже, в динамічній частині немає нелінійності і в нелінійній частині немає динаміки. Запропонована мережева структура може мати велике значення для ідентифікації та керування на основі моделі систем з лінійною динамічною поведінкою та статичною нелінійністю в приводі.

5.4.1 Числовий приклад 5.3 – Застосування моделі Хаммерштейна на основі нейронної мережі для ідентифікації серводвигуна постійного струму з нелінійним драйвером

Задача керування положенням серводвигуна була сформульована в [70]. Двигун має двонаправлений драйвер з нелінійною характеристикою входу/виходу, що має мертву зону та насичення

$$\begin{cases} \text{if } |V_i| \geq V_{dz}, V_o = 0V \\ \text{if } (V_i > V_{dz}) \& (V_i < V_{\max}), V_o = (V_i - V_{dz}) \cdot \text{gain} \\ \text{if } (V_i < -V_{dz}) \& (V_i > -V_{\max}), V_o = (V_i + V_{dz}) \cdot \text{gain} \\ \text{f } V_i \geq V_{\max}, V_o = V_{cc} \\ \text{f } V_i \leq -V_{\max}, V_o = -V_{cc} \end{cases} \quad (5.16)$$

Тут V_i – це вхід драйвера (керуючий сигнал), а V_o – вихід драйвера (вхідна напруга двигуна). Драйвер має такі параметри: $V_{dz} = 0,25$, $V_{\max} = 0,5$, посилення = 40, $V_{cc} = 10V$. Двигун постійного струму має лінійну динаміку [71] і коли вплив крутного моменту механічного навантаження не враховується, це може бути представлено наступною моделлю безперервного часу в просторі станів другого порядку.

$$\frac{d}{dt} \begin{bmatrix} i_a \\ \omega_a \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{k_v}{L_a} \\ \frac{k_t}{J} & -\frac{B}{J} \end{bmatrix} \begin{bmatrix} i_a \\ \omega_a \end{bmatrix} + \begin{bmatrix} -\frac{1}{L_a} \\ 0 \end{bmatrix} V_a$$

$$y = [0 \ 1] \begin{bmatrix} i_a \\ \omega_a \end{bmatrix} + 0 \cdot V_a \quad (5.17)$$

Двома внутрішніми станами є струм якоря i_a і швидкість обертання ω_a , один вхід - це напруга V_a , а один вихід - швидкість обертання ω_a (другий внутрішній стан).

Параметри цього двигуна наступні

$R_a = 1,75$ Ом – опір якоря двигуна;

$L_a = 2,83 \cdot 10^{-3}$ Н - – індуктивність якоря двигуна;

$k_v = 0,093$ В с/рад – постійна швидкості;

$k_t = 0,0924$ Нм / А — константа крутного моменту;

$J = 3 \cdot 10^{-5}$ кгм² - це інерція, яку бачить двигун. Вона включає в себе інерцію навантаження

$B = 5 \cdot 10^{-3}$ Нмс – коефіцієнт механічного гасіння, пов'язаний з обертанням.

Проведено моделювання моделі двигуна з нелінійним приводом (5.16), (5.17). Змодельовано роботу цієї системи безперервного часу протягом 2 секунд. Оскільки моделі на основі нейронних мереж є моделями дискретного часу, вибірку даних проводили з інтервалами в 1 мс, а набір вибірових вхідно-вихідних даних використовували для навчання мережі запропонованої структури (5.13). Як і в попередньому прикладі, мережа була навчена з 6 нелінійними нейронами в прихованому шарі першої підмережі та 3 лінійними нейронами в прихованому шарі другої підмережі.

Отримана модель була перевірена на валідаційному наборі, що складається з введення-виведення пари $\{u_v(t), y_v(t)\}$, де $u_v(t)$ — лінійно зростаюча в діапазоні $[-1.5, 1.5]$ вхідний сигнал $u_v(t) = -1.5 + 60t$, $t = 0,1, \dots, 50$ мс або в термінах дискретного часу система з часом вибірки 1 мс u_v

$(k) = -1.5 + 0.06k$, де $k = 0, 1, \dots, 50$. Перевірка моделі зображена на рисунку 5.8.

З рисунка видно, що вихід двигуна і вихід запропонованої архітектури на основі дискретної моделі часу практично збігаються. Середня абсолютна похибка цієї моделі становить близько 0.56 рад/с, що становить приблизно 1% від максимальної швидкості двигуна.

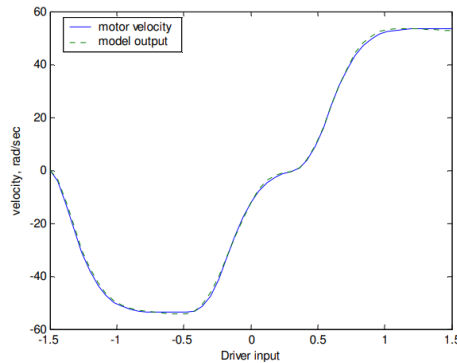


Рисунок 5.8 – Ідентифікація системи, що складається з двигуна постійного струму та нелінійного двонаправленого драйвера двома підмережами

Застосування запропонованої структури нейронної мережі замість звичайних багатошарових моделей NARX на основі персептрона для ідентифікації систем зі статичними вхідними нелінійністю та лінійною динамікою може значно покращити якість схем керування на основі моделі. Це особливо важливо, коли вимога контролю висока.

Тут показана одна з можливостей підвищення якості моделей, що використовуються в управлінні на основі моделі для одного класу нелінійних систем. Реалізація цих моделей у схемах керування на основі моделі буде розглядатися у розділі 4.2.2 цієї роботи.

5.5 Моделі ANARX на основі нейронних мереж

У [32] доведено, що модель ANARX (2.13) лінеаризується за

допомогою алгоритму лінеаризації динамічного вихідного зворотного зв'язку (2.36)-(2.38). У [8], [9] також показано, що модель ANARX завжди реалізована в статистиці форма електронного простору. Структура на основі нейронних мереж для представлення моделі ANARX була показана в [8], [10], [11] та використана для ідентифікації та керування нелінійними системами в [13] та [14]. Ця модель називається адитивною моделлю NARX на основі нейронних мереж (або NN-ANARX). Структура мережі для представлення моделі ANARX зображена на рисунку 5.9.

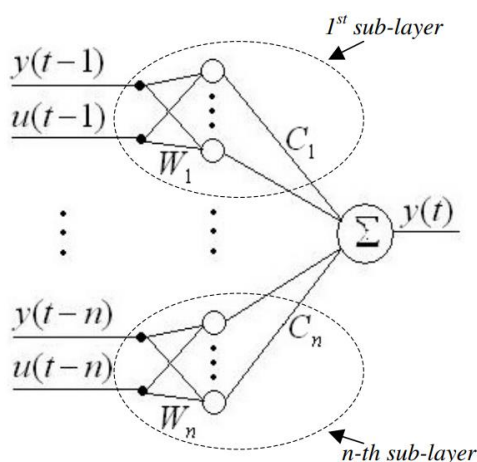


Рисунок 5.9 – Адитивна модель NARX, представлена нейронною мережею

Ця мережа складається з вихідного шару з нейронами типу ADALINE (Adaptive Linear Neuron) і прихованого шару, що складається з n паралельних підшарів, що відповідають n -му порядку моделі. Кожен i -й підрівень апроксимує функцію f_i з (2.13). Цю модель на основі нейронної мережі можна формалізувати за допомогою наступного рівняння

$$y(t) = \sum_{i=1}^n C_i \varphi_i(W_i \cdot Z(t-i)), \quad (5.18)$$

де $Z(t) = [y(t), u(t)]^T$, $\varphi_i(\cdot)$ – функція активації нейронів i -го підшару, C_i — матриця вихідних синаптичних ваг i -го підшару, W_i є матрицею i -х вхідні

синаптичні ваги i -го підшару.

Ця нейронна мережа має обмежене підключення. Таким чином, якщо i -й підрівень прихованого шару складається з l_i нейронів, то кількість параметрів, які необхідно налаштувати за допомогою навчального алгоритму,

$$\text{size}(W_1, \dots, W_n, C_1, \dots, C_n) = 3 \sum_{i=1}^n l_i, \quad (5.19)$$

що набагато менше, ніж у випадку повноз'єданого багатошарового персептрона з

$l_1 + \dots + l_n$ нейронів у прихованому шарі. Наприклад, якщо двошаровий персептрон з 9 нейронами в прихованому шарі використовується для ідентифікації моделі 3-го порядку ($n = 3$), він має 6 входів $[y(t-1), u(t-1), y(t-2), u(t-2), y(t-3), u(t-3)]$ і таким чином $6 \times 9 + 9 = 60$ синаптичні ваги повинні бути скориговані. Коли NN-ANARX модель (5.18) з 3 нейронами в кожному з 3 підшарів використовується для ідентифікації тієї ж системи, тоді згідно з рівнянням (5.19) лише $27 \cdot 3 = 81$ синаптичних ваг потрібно відрегулювати.

Цей факт дозволяє використовувати більш точні алгоритми другого порядку для навчання мережі. Наприклад, навчальний алгоритм Левенберга-Марквардта можна використовувати замість зворотного поширення помилки градієнтного спуску, як це було показано в [13], [14].

5.5.1 Реалізація простору стану за допомогою NN-ANARX

Структура ANARX завжди реалізована в класичній формі простору станів (2.14). Коли модель отримана у вигляді нейронної мережі (5.18) шляхом навчання мережі структури, показаної на рисунку 5.9, реалізацію простору станів можна сформулювати за допомогою матриць синаптичних ваг мережі [8] як

$$\begin{aligned}
x_1(t+1) &= x_2(t) + C_1 \varphi_1(W_1(x_1(t), u(t))^T) \\
x_2(t+1) &= x_3(t) + C_2 \varphi_2(W_2(x_1(t), u(t))^T) \\
&\dots \\
x_{n-1}(t+1) &= x_n(t) + C_{n-1} \varphi_{n-1}(W_{n-1}(x_1(t), u(t))^T) \\
x_n(t+1) &= C_n \varphi_n(W_n(x_1(t), u(t))^T),
\end{aligned} \tag{5.20}$$

де W_i і C_i — це матриці вхідних і вихідних синаптичних ваг кожного під-шару, отриманий за допомогою навчального алгоритму і $\varphi(\cdot)$ є нелінійною активацією функції. Модель простору станів n -го відповідає кожній моделі ANARX і NN-ANARX n -го порядку.

У наступних трьох числових прикладах буде показано ідентифікацію та реалізацію в просторі станів нелінійних систем SISO шляхом навчання нейронних мереж відповідної структури.

5.5.2 Числовий приклад 5.4

Модель системи рівня рідини з'єднаних між собою резервуарів [72] представлена таким рівнянням введення-виведення

$$\begin{aligned}
y(t+3) &= 0,43y(t+2) + 0,681y(t+1) - 0,149y(t) + 0,396u(t+2) + \\
&+ 0,014u(t+1) - 0,071u(t) - 0,351y(t+2)u(t+2) - 0,03y^2(t+1) - \\
&- 0,135y(t+1)u(t+1) - 0,027y^3(t+1) - 0,108y^2(t+1)u(t+1) - \\
&- 0,099u^3(t+1)
\end{aligned} \tag{5.21}$$

Ідентифікація цієї системи шляхом навчання структури NN-ANARX продемонстрована в [14].

Рівняння (5.21) було використано як невідома нелінійна установка, яка

моделюється структурою ANARX на основі NN

$$y(t + 3) = \sum_{i=1}^3 C_i \varphi_i(W_i \cdot [y(t + i - 1), u(t + i - 1)]^T) \quad (5.22)$$

Для отримання вхідно-вихідних даних об'єкт (5.21) моделювався з синусоїдальним вхідним сигналом $u(t) = \sin(0.05t) + 0.05e(t)$, де шум $e(t)$ нормально розподілений із середнім 0 та дисперсією $\sigma^2 = 1$: $e(t) \sim N(0, 1^2)$. Нейронна мережа з трьома підшарами (порядок моделі $n = 3$) і з трьома нейронами на кожному підшарі прихованого шару ($l_1 = l_2 = l_3 = 3$) з логістичними функціями активації сигмовидної форми (2.50) пройшов підготовку.

Алгоритм навчання LM був обраний для виконання офлайнного навчання, оскільки він набагато ефективніший порівняно з іншими методами, коли мережа містить не більше кількох сотень ваг [53], а в нашому випадку відповідно до У рівнянні (5.19) нейронна мережа має всього 27 синаптичних ваг. Крім того, швидкість навчання алгоритму LM набагато вища, і нейронна мережа прямої подачі, навчена за допомогою нього, може краще моделювати нелінійність [54]. Навчання займало близько 600 ітерацій (epoch) для зближення. Виявлені параметри моделі (5.22) мають такі значення

$$W_1 = \begin{bmatrix} 0.9132 & 0.2356 \\ 0.3809 & -1.3663 \\ 0.8472 & 0.0455 \end{bmatrix}, W_2 = \begin{bmatrix} -0.2946 & -4.4628 \\ 1.2427 & -0.5516 \\ 1.2093 & -0.6216 \end{bmatrix},$$

$$W_3 = \begin{bmatrix} 1.2788 & -3.0221 \\ 0.4240 & 0.4244 \\ 0.4339 & 0.5441 \end{bmatrix}, \quad (5.23)$$

$$C_1 = [-24.7605 \quad -4.5011 \quad 34.0981],$$

$$C_2 = [-0.2364 \quad -26.4052 \quad 27.2547],$$

$$C_3 = [1.6433 \quad -71.6573 \quad 64.4851]$$

Для перевірки моделі набір даних $\{u_v(t), y_v(t)\}$ був згенерований за допомогою вхідний сигнал з удвічі більшою частотою: $u_v(t) = \cos(0.1t) + 0,05e(t)$. Відповідні виходи системи (5.21) і моделі (5.22) з параметрами (5.23) зображені на рисунку 5.10.

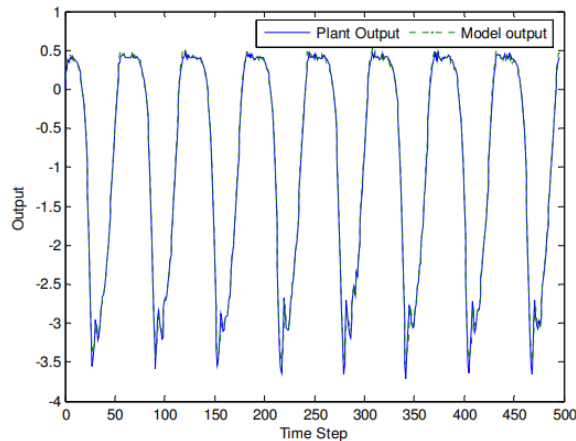


Рисунок 5.10 – Перевірка моделі системи взаємопов'язаних резервуарів на основі структури NN-ANARX

Перевірка моделі показує майже чудове перекриття моделі та заводу виходи. Середньоквадратична помилка на наборі для перевірки була настільки низькою як $MSE \approx 4 \cdot 10^{-3}$. Видно, що ця система може бути представлена моделлю NN-ANARX з високим ступенем точності.

У наступному прикладі буде показано ідентифікацію та представлення в просторі станів системи, модель введення-виведення якої не має структури ANARX. Це можна зробити шляхом навчання нейронної мережі з архітектурою ANARX, показаною на рисунку 5.9.

5.5.3 Числовий приклад 5.5

Модель дискретного часу другого порядку реактора безперервного перемішування (CSTR) з сорочкою [74], [75], [13], [14] представлена таким

рівнянням введення-виведення.

$$\begin{aligned}
 y(t+2) = & 0,7653y(t+1) - 0,231y(t) + 0,4801u(t+1) - \\
 & - 0,6407 y^2(t+1) + 1,014 y(t) y(t+1) - 0,3921y^2(t+1) + \\
 & + 0,592 y(t+1)u(t+1) - 0,5611y(t)u(t+1)
 \end{aligned} \quad (5.24)$$

Зауважте, що через останній член у (5.24) модель не має структури ANARX. Щоб отримати вхідно-вихідні дані з «невідомого» заводу, рівняння (5.24) було змодельовано з синусоїдальним вхідним сигналом $u(t) = \sin(0.05t) + 0.05e(t)$, де $e(t) \sim N(0, 1^2)$. Це вхідний сигнал і відповідний вихідний сигнал використовувалися як навчальний набір для ідентифікації шляхом навчання структури ANARX на основі нейронної мережі (5.18). Ця рослина була змодельована моделлю NN-ANARX з двома підшарами, що відповідають другому порядку моделі ($n = 2$), і трьома нейронами з логістичними функціями активації сигмовидної форми (2.50) на кожному підшарі прихованого шару ($l_1 = l_2 = 3$). Таким чином, 18 параметрів були налаштовані за допомогою алгоритму навчання LM, на зближення якого знадобилося близько 600 епох, і модель, яка має таке рівняння було отримано.

$$y(t+2) = C_1 \varphi_1(W_1 \cdot [y(t+1), u(t+1)]^T) + C_2 \varphi_2(W_2 \cdot [y(t), u(t)]^T) \quad (5.25)$$

Параметри моделі (5.25) мають значення вказані нижче.

$$\begin{aligned}
 W_1 &= \begin{bmatrix} 0.1793 & 0.2616 \\ -23.0273 & 22.2099 \\ -0.0785 & -0.4576 \end{bmatrix}, \\
 W_2 &= \begin{bmatrix} -23.4344 & 29.5285 \\ 1.5266 & -2.0768 \\ -1.0887 & 1.6082 \end{bmatrix},
 \end{aligned} \quad (5.26)$$

$$C_1 = [31.9678 \quad -0.0166 \quad 13.8712],$$

$$C_2 = [-0.0128 \quad -19.8288 \quad -25.9744]$$

Для перевірки моделі набір даних $\{u_v(t), y_v(t)\}$ був згенерований за допомогою вхідний сигнал з удвічі більшою частотою: $u_v(t) = \cos(0.1t) + 0.05e(t)$. Відповідні виходи системи (5.24) і моделі (5.25) з параметрами (5.26) зображені на рисунку 5.11.

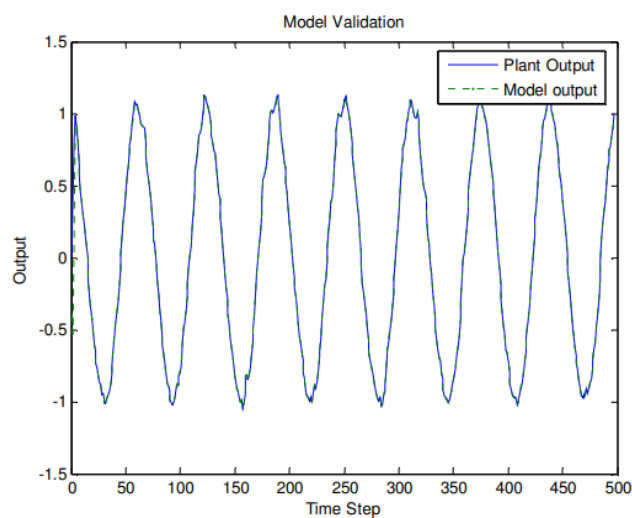


Рисунок 5.11 – Перевірка моделі CSTR з оболонкою на основі структури NN-ANARX

Перевірка моделі показує, що, незважаючи на обмеження, накладені структурою NN-ANARX, ідентифікована модель пояснює вхідно-вихідні дані з високим ступенем точності. З рисунка видно, що вихід системи та моделі практично збігаються. Середньоквадратична помилка була настільки низькою як $MSE \approx 3,5 \cdot 10^{-3}$.

Хоча структурне обмеження, накладене NN-ANARX на матричну структуру підключення, може здатися занадто сильним, результати навчання [13], [14], [76] показують, що в багатьох випадках модель ANARX на основі NN здатна представляти оригінальну модель з високим ступенем точності і

не викликає серйозних недоліків у якості ідентифікації.

Відповідно до (5.20), модель введення-виведення (5.24) тепер може бути представлена у вигляді моделі простору станів. Модель простору станів може бути формально записана як

$$x_1(t+1) = x_2(t) + C_1 \varphi_1(W_1(x_1(t), u(t))^T) \quad (5.27)$$

$$x_2(t+1) = x_3(t) + C_2 \varphi_2(W_2(x_1(t), u(t))^T)$$

де W_1 , W_2 , C_1 і C_2 – параметри (5.26) мережі (5.25) представлення системи у вигляді моделі NN-ANARX; φ_1 і φ_2 є логістичні функції сигмовидної активації (2.50) відповідних підшарів прихованого шару.

5.6 Модель Хаммерштейна на основі NN-ANARX

У розділі 5.4 цієї роботи пропонується структура нейронної мережі, що складається з двох повнозв'язаних багатошарових персептронів, для представлення систем типу Гаммерштейна – систем, що мають лінійну динаміку та нелінійну статичну нелінійність входу. Запропонована мережа (5.12), зображена на рисунку 5.5, здатна розділяти статичну нелінійність і лінійну динаміку. Такий же підхід можна використовувати для розробки моделі Хаммерштейна на основі нейронної мережі, яка належить до класу моделей ANARX.

Для ідентифікації лінійної динамічної частини моделі використовується нейронна мережа (5.18) структури, показаної на рисунку 5.9, з усіма лінійними нейронами. У динамічній частині замість нелінійних нейронів необхідно використовувати ADALINE. Для ідентифікації вхідної нелінійності використовується статична нейронна мережа з одним входом і одним виходом і нелінійними функціями активації нелінійності прихованого шару. Вихід цієї мережі потім передається до підшарів лінійної мережі

ANARX з відповідними затримками. Структура мережі представлена на рисунку 5.12.

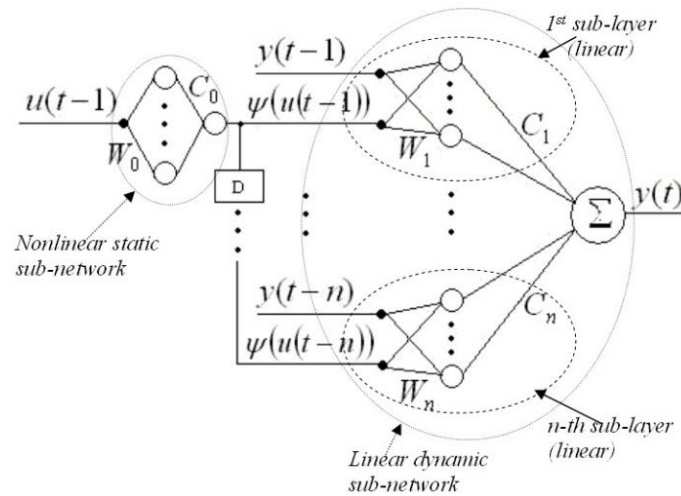


Рисунок 5.12 – Модель Хаммерштейна на основі NN-ANARX

Запропоновану тут модель можна формалізувати як

$$y(t) = \sum_{i=1}^n C_i \left(W_i [y(t-i), \psi(u(t-i))]^T \right), \quad (5.28)$$

де $\psi(u(t))$ – функція першої нелінійної статичної нейронної мережі

$$\psi(u(t)) = C_0 \varphi(W_0 \cdot u(t)) \quad (5.29)$$

Тут $\varphi(\cdot)$ — нелінійна функція активації нейронів прихованого шару статична підмережа, $W_0 \in \mathcal{R}^{l_0 \times 1}$ і $C_0 \in \mathcal{R}^{1 \times l_0}$ є векторами синаптичних ваг статична підмережа, $W_i \in \mathcal{R}^{l_i \times 2}$ і $C_i \in \mathcal{R}^{1 \times l_i}$ де $i = 1, \dots, n$ є матрицями і вектори лінійної динамічної підмережі. l_i , $i = 0, \dots, n$ є числа нейронів у відповідних підшарах. Таким чином, з рівнянь (5.28) і (5.29) отримуємо

$$y(t) = \sum_{i=1}^n C_i \left(W_i [y(t-i), C_0 \varphi(W_0 u(t-i))]^T \right) \quad (5.30)$$

Як $C_i \in \mathcal{R}^{1 \times l_i}$ і $W_i \in \mathcal{R}^{l_i \times 2}$, вектори двох коефіцієнтів $G_i = [g_{1i} \ g_{2i}]$ може бути визначено для кожного підшару лінійної динамічної підмережі ANARX як

$$G_i = [g_{1i} \ g_{2i}] = C_i \cdot W_i, \forall i = 1, \dots, n. \quad (5.31)$$

За допомогою цих коефіцієнтів модель Хаммерштейна на основі NN-ANARX тепер можна формалізувати як

$$y(t) = \sum_{i=1}^n g_{1i} \cdot y(t-i) + g_{2i} \cdot C_0 \cdot \varphi(W_0 \cdot u(t-i)). \quad (5.32)$$

Ця модель належить до обох класів: до класу моделей ANARX на основі нейронних мереж (5.18), оскільки в ній всі екземпляри часу розділені, і до класу моделей Гаммерштейна оскільки має лінійну динаміку та нелінійний статичний вхід. Ця модель здатна з високим ступенем точності представляти системи зі статичними вхідними нелінійністю (наприклад, з нелінійними приводами). Завдяки моделі, заснованій на навчальній нейронній мережі, вона здатна моделювати системи з невідомими статичними вхідними нелінійністю шляхом автоматичного розділення лінійної динаміки та нелінійної статичності під час процедури навчання. Завдяки властивостям моделей ANARX (2.14), модель n-го порядку (5.32) може бути дуже легко представлена у класичній формі простору станів, яка підходить для різноманітних алгоритмів керування, як

$$x_1(t+1) = x_2(t+1) + g_{11} \cdot y(t) + g_{21} \cdot C_0 \cdot \varphi(W_0 \cdot u(t))$$

$$x_2(t+1) = x_3(t+1) + g_{12} \cdot y(t) + g_{22} \cdot C_0 \cdot \varphi(W_0 \cdot u(t))$$

$$x_n(t+1) = g_{1n} \cdot y(t) + g_{2n} \cdot C_0 \cdot \varphi(W_0 \cdot u(t)) \quad (5.33)$$

$$y(t) = x_1(t)$$

Для порівняння різних підходів моделювання систем типу Гаммерштейна на основі нейронних мереж розглянемо наступний приклад.

5.6.1 Числовий приклад 5.6

У [11] модель Хаммерштейна на основі NN-ANARX (5.32) була використана для ідентифікації серводвигуна постійного струму (DC) з нелінійним драйвером (5.16)-(5.17) і порівнювалася з підходом, запропонованим у [51], який базується на навчання двох персептронів (5.12)-(5.13), що розглядається в розділі 5.4 цієї роботи.

Проведено навчання нейронної мережі з лінійною динамічною підмережею, що має два підшари свого прихованого шару ($n = 2, l_1 = l_2 = 3$), що відповідають другому порядку моделі. У статичній частині моделі використано шість нейронів нелінійного прихованого шару ($l_0 = 6$). Таким чином, лише $6 + 6 + 2 \cdot 2 \cdot 3 + 3 \cdot 2 = 30$ синаптичних ваг довелося відкоригувати за допомогою навчального алгоритму.

Обидві моделі Хаммерштейна на основі NN-NARX, показані в числовому прикладі 5.4, і модель на основі NN-ANARX, обговорювані в цьому розділі, були навчені на одному наборі даних і перевірені на вхідному сигналі, що зростає лінійно від -1.5 до +1.5. Відповідні виходи системи та відповідної моделі ANARX зображені на рисунку 5.13.

Середня абсолютна похибка цієї моделі двигуна постійного струму і нелінійного двонаправленого драйвера на основі NN-ANARX становить близько 0.8 рад/с, що становить приблизно 2% від максимальної швидкості двигуна в рамках цього експерименту. Оскільки якість ідентифікації в певному сенсі залежить від конкретного навчального експерименту

(залежить від випадкових початкових значень синаптичних ваг), можна сказати, що точність цієї моделі приблизно така ж, як і у випадку моделі NARX Hammerstein на основі нейронних мереж, яка обговорюється в розділ 5.4.

Порівнявши рисунки 5.13 і 5.8, можна також побачити, що обидві моделі здатні представити цю систему з нелінійним приводом з високим ступенем точності і обмеження, накладені структурою ANARX на топологію мережі, не викликають серйозного зниження якості ідентифікація.

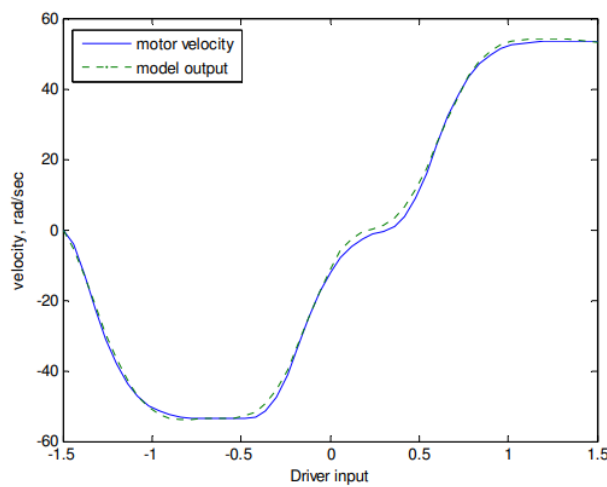


Рисунок 5.13 – Ідентифікація системи, що складається з двигуна постійного струму та нелінійного двонаправленого драйвера за моделлю NN-ANARX

У той же час, коли для моделювання систем зі статичними вхідними нелінійністю використовуються специфічні для прикладних нейронних мереж (5.12)- (5.13) або (5.16)-(5.17), якість ідентифікації значно вища в порівнянні з ідентифікацією шляхом навчання а. одношаровий персептрон. Таким чином, незважаючи на те, що згідно з теоремою Стоуна-Вейерштрасса двошарові персептрони здатні апроксимувати будь-які безперервні нелінійності і за допомогою додавання зовнішнього зворотного зв'язку можна використовувати для ідентифікації нелінійної динаміки. систем, практично реалізація прикладних нейромережових структур, запропонованих

в [11] та [51], дає значне покращення якості ідентифікації динамічних систем зі статичними вхідними (актуаторними) нелінійністю.

Обидві моделі NN-NARX і NN-ANARX Hammerstein забезпечують приблизно однакову хорошу якість ідентифікації. Основною перевагою використання моделей Хаммерштейна на основі NN-NARX є більш проста структура моделі, але, з іншого боку, моделі Хаммерштейна на основі NN-ANARX з такою ж кількістю нейронів, що й мережа NN-NARX, мають меншу кількість синаптичних ваг через обмежене підключення. Це робить навчання швидшим, дозволяє використовувати складніші алгоритми навчання та можливу швидку адаптацію моделі в адаптивних схемах реального часу. Ще одна перевага моделей Хаммерштейна на основі NN-ANARX полягає в тому, що вони лінеаризуються за допомогою динамічного вихідного зворотного зв'язку (2.36)-(2.38) і репрезентуються в класичній формі простору станів (5.33), що робить їх придатними для більш широкого діапазону моделей на основі алгоритми управління.

5.7 Моделі нелінійних систем MIMO

Штучні нейронні мережі також є дуже зручним інструментом для ідентифікації систем Multiply Input – Multiply Output (MIMO). У цьому розділі будуть розглянуті обидві моделі нелінійних систем MIMO на основі нейронних мереж NARX і ANARX.

5.7.1 Моделі MIMO NARX і MIMO NN-NARX

З літератури (наприклад, [77]) відомо, що широкий клас нелінійних систем MIMO може бути представлений нелінійною дискретною моделлю в описі введення-виведення відомої в літературі як моделі NARX, як послідовно-паралельна модель або як одно- провісник кроку вперед.

Нехай $U(t) = [u_1(t), \dots, u_r(t)] \in \mathcal{R}^r$ — вектор системних входів і $Y(t) =$

$[y_1(t), \dots, y_m(t)] \in \mathcal{R}^m$ — вектор вихідних даних системи на етапі часу t , r — кількість входів і m - кількість виходів моделі. Тоді модель MIMO NARX можна визначити як

$$Y(t) = f(Y(t-1), \dots, Y(t-n_y), U(t-1), \dots, U(t-n_u)), \quad (5.34)$$

де n_u і n_y – порядок введення та виведення та $f(\cdot)$ — нелінійна функція з $n_u + n_y$ аргументи.

Дуже зручно отримати модель (5.34) шляхом навчання багат шарового перцептрона апроксимуюча нелінійна функція $f(\cdot)$. Щоб отримати динамічний MIMO системи зовнішні зворотні зв'язки від кожного виходу моделі $y_1(t), \dots, y_m(t)$ є необхідні, а також відкладені значення всіх входів $u_1(t), \dots, u_r(t)$. Структура відповідної нейронної мережі на основі моделі MIMO NARX зображена на рисунку 5.14.

Відповідно до теореми Стоуна-Вейерштрасса будь-яка безперервна нелінійна функція $f(\cdot)$ може бути апроксимована двошаровою перцептрон. Отже, двошаровий перцептрон із зовнішніми зворотними зв'язками та затримками також може бути використаний для ідентифікації нелінійних систем MIMO. Моделі, засновані на цій структурі, називаються моделлю NARX на основі нейронної мережі або моделями NN-NARX.

Модель MIMO NARX, отримана шляхом навчання двошарового перцептрона, має в якості параметрів дві матриці синаптичних ваг. Перший є вхідною матрицею синаптичні ваги $W \in \mathcal{R}^{(r \cdot n_u + m \cdot n_y) \times 1}$, де 1 – кількість нейронів у прихований шар, а другий є вихідною матрицею $C \in \mathcal{R}^{m \times 1}$.

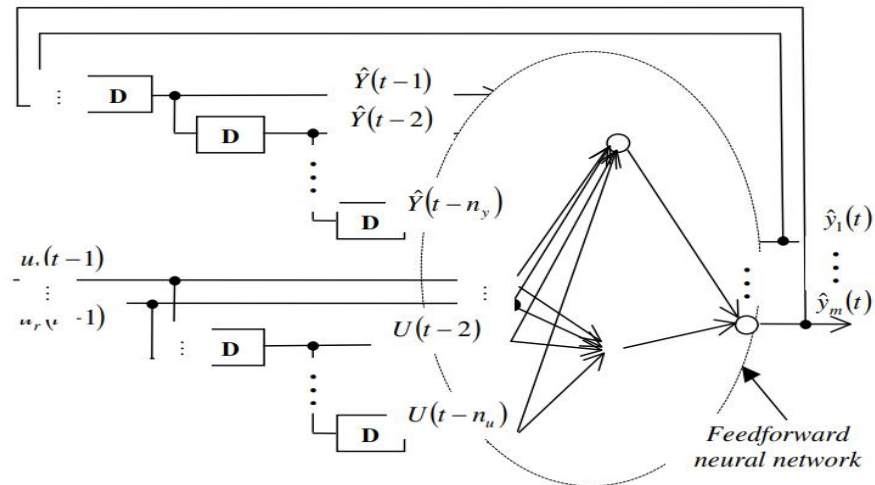


Рисунок 5.14 – Подання моделі MIMO NARX на основі нейронної мережі

Коли мережа навчена і отримано оптимальні значення її синаптичних ваг, вектор провісників $Y(t)$ на крок вперед можна розрахувати за допомогою такого рівняння:

$$Y(t) = C \cdot \varphi(W \cdot [Y(t-1), \dots, Y(t-n_y), U(t-1), \dots, U(t-n_u)]^T) \quad (5.35)$$

У випадку цієї моделі кількість параметрів, які ми повинні розрахувати під час навчання мережі, різко зростає за рахунок збільшення кількості виходів та/або входів та/або порядку моделі. Це робить практичну ідентифікацію дуже повільною або навіть неможливою. Експерименти показують, що в деяких випадках навчальні алгоритми можуть не сходитися через величезну кількість регульованих параметрів (синаптичних ваг). Наприклад, коли двошаровий перцептрон з 9 нейронами в прихованому шарі використовується для отримання моделі системи MIMO третього порядку, що має 2 входи і 2 виходи, значення 126 синаптичних ваг мають бути розраховані за вибраним алгоритмом навчання.

Для подолання розриву у [56] була запропонована модель MIMO NN-ANARX.

5.7.2 Моделі MIMO ANARX і MIMO NN-ANARX

Модель ANARX — це підклас моделей NARX, які мають розділені всі екземпляри часу. Для систем MIMO модель ANARX можна визначити як

$$Y(t) = \sum_{i=1}^n f_i(Y(t-i), U(t-i)), \quad (5.36)$$

де n - порядок моделі і $f_1(\cdot), \dots, f_n(\cdot)$ — нелінійні функції.

Модель ANARX на основі нейронних мереж для систем MIMO визначена в [56] як

$$Y(t) = \sum_{i=1}^n C_i \cdot \varphi_i(W_i \cdot [Y(t-i), U(t-i)]^T), \quad (5.37)$$

де $\varphi_1(\cdot), \dots, \varphi_n(\cdot)$, W_1, \dots, W_n , C_1, \dots, C_n є нелінійними функціями активації і матриці синаптичних ваг підшарів мережевої структури, показаних на рисунку 5.15.

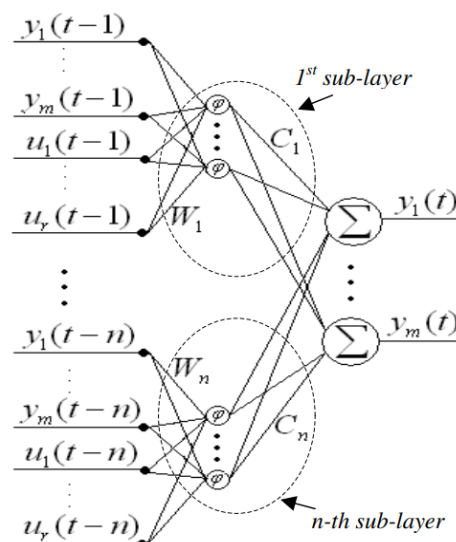


Рисунок 5.15 – Структура нейронної мережі, що представляє модель ANARX для системи MIMO

Матриці синаптичних ваг цієї моделі мають такі розміри: $W_i \in \mathbb{R}^{l_i \times (m+r)}$ і $C_i \in \mathbb{R}^{m \times l_i}$, де l_i – кількість нейронів в i -му підшарі ($i = 1, \dots, n$).

Щоб порівняти цю структуру з моделями NN-NARX, які розглядалися в попередньому розділі, давайте також розглянемо систему MIMO третього порядку ($n = 3$), що має 2 входи і два виходи ($m = r = 2$), отриману шляхом навчання нейронної мережі з 9 нейронів у його прихованому шарі ($\sum_{i=1}^3 l_i = 9$). У випадку MIMO на основі нейронної мережі модель ANARX (5.37), структура відповідної нейронної мережі, показана на рисунку 5.15, повинна складатися з 3 підшарів. 3 нейрони можна використовувати в кожному підшарі для 9 нейронів у прихованому шарі ($l_1 = l_2 = l_3 = 3$). Отже, навчаючи цю структуру, потрібно розрахувати лише 54 синаптичні ваги. Це більш ніж у 2 рази менше в порівнянні зі структурою MIMO NN-NARX з такою ж кількістю входів, виходів і нейронів.

Експерименти показали, що сильні обмеження зв'язності структури мережі не викликають недоліків у якості ідентифікації. Навіть більше, це дає деяке поліпшення, оскільки кількість підключень не стає занадто великою. Менша кількість параметрів робить навчання швидшим і дозволяє реалізувати більш широкий спектр алгоритмів навчання. Наприклад, навчальні алгоритми другого порядку, такі як LM-алгоритм.

Проблеми управління нелінійними системами MIMO на основі структури MIMO NN-ANARX будуть детально розглянуті в розділі 5 цієї роботи.

Для кращої точності моделі у разі ідентифікації складних багатовимірних процесів, модель MIMO може бути замінена набором r моделей MISO, де r - кількість виходів процесу, або кількома системами MIMO, де виходи розділені між кількома окремими моделі. Цей підхід реалізовано для моделювання кисті хірурга. Цей проект буде розглянуто в наступному прикладі.

5.7.3 Числовий приклад 5.7 – ANARX-модель руки хірурга на основі NN для розпізнавання руху та прогнозування руху

Основною метою проекту є виконання можливостей навчання та адаптації робота-медсестри, щоб мати можливість передбачати запити хірурга під час хірургічної операції та реагувати на них належними допоміжними діями. Однією з дуже важливих частин «мозку» робота є модель руки хірурга, яка необхідна для прогнозування координат зап'ястя хірурга та визначення типу руху. Для отримання моделі було використано набір структур MIMO NN-ANARX. Ці результати опубліковані в [73]. Зображено структуру потоку даних робота-медсестри на рисунку 5.16, а проект показаний на схемі штриховим овалом.

Перш за все, дозвольте мені дати короткий огляд цього проекту [73]. Однією з останніх тенденцій розвитку медичних роботів можна вважати тісну взаємодію з людиною. Така тісна взаємодія викликає необхідність виявлення дій людини і модель поведінки людини.[78] У роботі [79] основна увага приділена побудові цілісної моделі медичної сестри, яка включає розширені часові автомати для моделювання спільних дій усього хірургічного персоналу та динамічні моделі, що описують фізичний рух хірургічного персоналу.

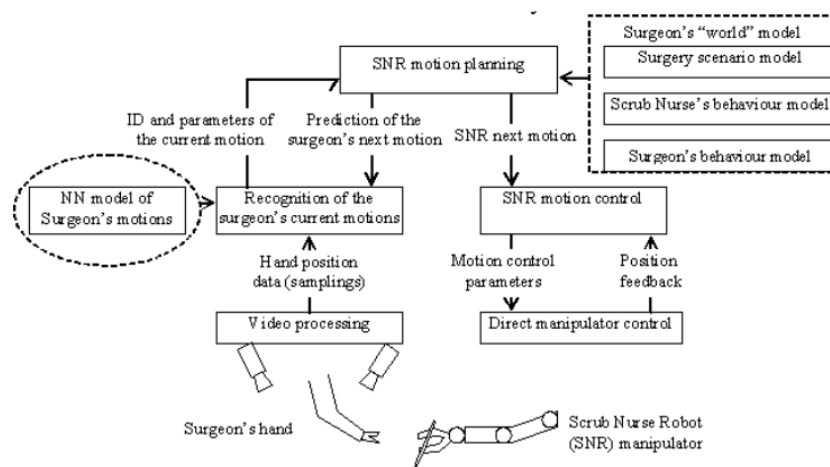


Рисунок 5.16 – Потік керуючих даних медсестри-робота

Основна мета робота-медсестри (SNR) — розпізнати наміри хірурга та надати допомогу, яку він очікує від SNR для цього. Допомогою може бути просто тримання хірургічного інструменту, але в більш складних випадках це послідовність дій, включаючи, наприклад, видихання інструменту з лотка, очікування, передачу його хірургу, отримання іншого інструменту та повернення його на лоток, тощо. Прийняття людиною SNR означає не тільки правильне розташування інструментів і правильний час виконання рухів, але також коригування прискорення та траєкторії рухів відповідно до особистих уподобань хірурга. Єдиний практичний спосіб досягти усиювлення людини в даному сенсі — це вчитися у вправних медсестер і наслідувати їх поведінку в подібних ситуаціях. Однак правильна поведінка медичної сестри залежить також від контекстуальних знань, які не можна спостерігати безпосередньо: як поводитися в надзвичайних ситуаціях, який узгоджений сценарій хірургічної процедури, в якому порядку інструменти розміщуються на хірургічному лотку тощо. Коли SNR навчається на основі пасивного спостереження за хірургічними процедурами, щоб помістити дані спостереження в правильний контекст, деякі базові знання вже мають бути там. Таким чином, «модель світу» повинна бути центральною одиницею знання SNR. Фактично, «модель світу» — це набір моделей частково жорстко запрограмований під час проектування, частково створений і створений за допомогою он-лайн навчання. Наразі ми накопичуємо п'ять типів знань у моделі світу роботів: сценарій операції, реактивна поведінка медсестри та хірурга, а також моделі (безперервного) руху обох. Коли перші три кодуються як моделі розширених тимчасових автоматів [80], останні дві є моделями нейронних мереж, які детальніше обговорюються нижче в цьому розділі кваліфікаційній роботі. Для прийняття управлінських рішень використовується «модель світу» робота разом з даними онлайн-спостереження. Для реалізації згаданих вище функцій SNR пропонується архітектура керування роботом з потоком даних, зображена на рисунку 5.16.

Відповідно до заданої архітектури, контур управління включає відеомоніторинг та обробку сигналів, розпізнавання поточного руху хірурга, вибір відповідної реакції SNR та передачу команди керування з необхідними параметрами на блок керування маніпулятором.

Робота присвячена моделюванню руху руки хірурга моделлю ANARX на основі нейронних мереж. За допомогою цієї моделі можна вирішити проблему виявлення поточного руху хірурга, тобто виявити його поточну дію, та передбачення координат точки, де має відбуватися обмін приладами.

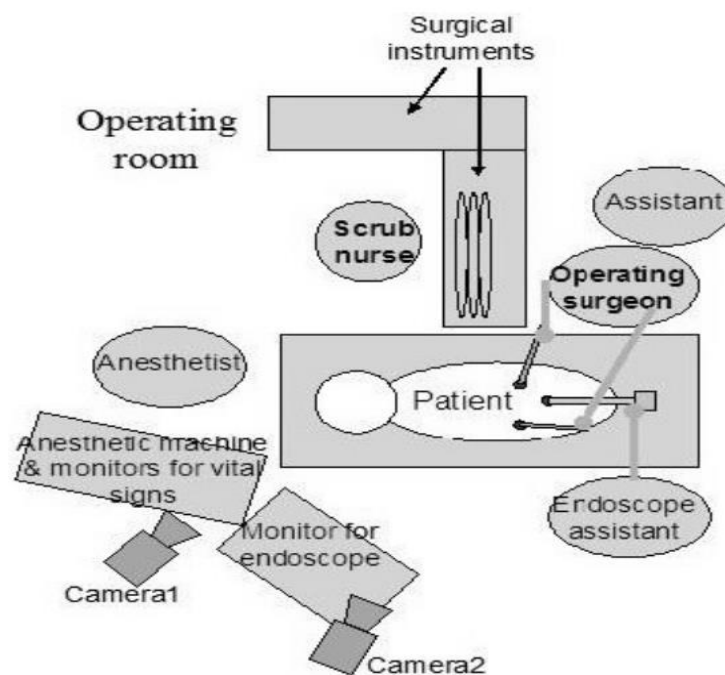


Рисунок 5.17 – Принципова схема операційної

Отримання моделі руху хірурга означає побудову вхідно-вихідної моделі, де вхід – це вектор спостережуваних координат грудної клітки, ліктя та зап'ястя хірурга, а вихід – тривимірний вектор, елементами якого є прогнозування грудної клітки, ліктя та зап'ястя. координати за певну кількість кроків часу.

Для отримання експериментальних даних від реальної хірургічної операції операційна була обладнана двома камерами. Положення медичного

персоналу під час хірургічної операції та розміщення камер зображені на рисунку 5.17. Для спрощення процесу моніторингу кольорові маркери були розміщені на грудях, лікті та зап'ясті хірурга.

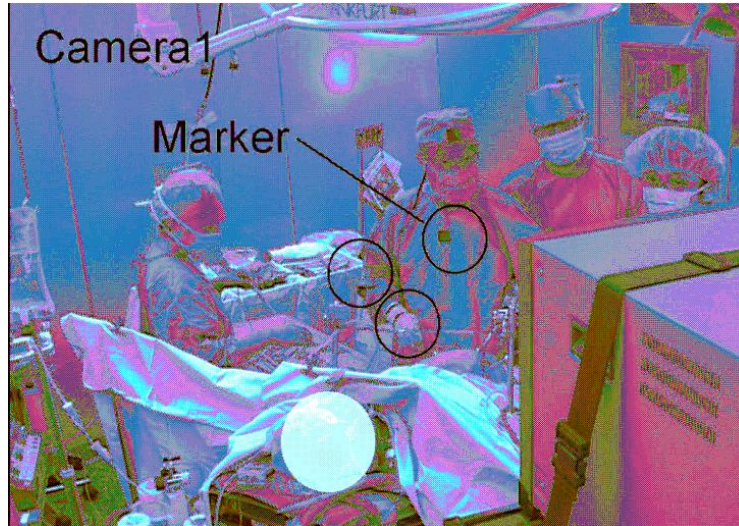


Рисунок 5.18 – Операційна

Камери були налаштовані на зйомку операції зі швидкістю 30 кадрів на секунду. Пізніше на основі записаної операції була витягнута інформація про координати грудної клітки, ліктя та зап'ястя. Частота дискретизації зібраних даних становить 0,033 сек. Дані вимірювання потім зберігалися у файлі MS Excel для навчання моделей. Короткий приклад послідовності даних (2 с), використаної для навчання, показано на рисунку 5.19.

	A	B	C	D	E	F	G	H	I	J	K
1	Time [sec]	chest: X [cm]	right_elbow	right_wrist: X	chest: Y [cm]	right_elbow: Y	right_wrist: Y	chest: Z [cm]	right_elbow: Z	right_wrist: Z	motion_name
2	0	26,032	52,352	58,253	44,762	32,819	24,629	16,948	-8,589	-17,068	working
3	0,0333	26,02	52,378	58,192	44,81	32,769	24,681	16,957	-8,616	-16,966	
4	0,0667	26,016	52,387	58,101	44,826	32,718	24,649	16,96	-8,625	-16,817	
5	0,1	26,016	52,387	57,909	44,826	32,698	24,723	16,96	-8,625	-16,434	
6	0,1333	26,016	52,388	57,637	44,826	32,697	24,921	16,96	-8,677	-15,85	
7	0,1667	26,016	52,483	57,514	44,826	32,653	25,01	16,96	-8,78	-15,622	
8	0,2	26,016	52,67	57,453	44,826	32,568	25,093	16,96	-8,781	-15,571	

Рисунок 5.19 – Виміряні навчальні дані

Проведено навчання моделей грудної клітки, ліктя та зап'ястя на основі

структури NN-ANARX. В якості вхідних даних моделей були використані координати X, Y і Z грудної клітки, ліктя та зап'ястя хірурга на 3 різних часових кроках, що відповідають третьому порядку моделі. 3 різні нейронні мережі зі структурою, показаною на рисунку 5.15 з трьома виходами, були навчені для отримання моделей руху координат X, Y і Z грудної клітки, ліктя та зап'ястя хірурга. Таким чином, кожна з 3 моделей NN-ANARX MIMO мала 27 входів (координати X, Y і Z грудей, ліктя та зап'ястя в 3 моменти часу).

Для ідентифікації використовувалися реальні вимірювання з часом вибірки 0,033 сек. Набір даних, що складається з 1227 вимірювань кожної координати, був використаний як навчальний набір. Модель MIMO NN-ANARX з трьома прихованими підшарами тренувалися шари, що відповідають третьому порядку моделі ($n = 3$). Наступні три моделі були отримані наступним чином.

Модель грудної клітки хірурга:

$$[x_c(t), y_c(t), z_c(t)]^T = \sum_{i=1}^3 C_c^i \cdot \varphi_c^i(W_c^i[X(t-i), Y(t-i), Z(t-i)]^T), \quad (5.38)$$

де $x_c(t)$, $y_c(t)$, $z_c(t)$ є відповідними координатами маркера, розміщеного на грудна клітка хірурга;

C_c^i , W_c^i , $i = 1, \dots, 3$ є матрицями синаптичної ваги MIMO NN-ANARX заснована модель грудної клітки хірурга;

$\varphi_c^i(\cdot)$, $i = 1, \dots, 3$ – нелінійні функції активації відповідного підшари нейронів прихованого шару цієї моделі;

$X(t)$, $Y(t)$, $Z(t)$ — вектори відповідних координат грудей, ліктя та зап'ястя:

$$X(t) = [x_c(t), x_e(t), x_w(t)] \quad (5.39)$$

$$Y(t) = [y_c(t), y_e(t), y_w(t)] \quad (5.40)$$

$$Z(t) = [z_c(t), z_e(t), z_w(t)] \quad (5.41)$$

Аналогічно моделі грудної клітки хірурга, модель ліктя хірурга

$$[x_e(t), y_e(t), z_e(t)]^T = \sum_{i=1}^3 C_e^i \cdot \varphi_e^i(W_e^i[X(t-i), Y(t-i), Z(t-i)]^T) \quad (5.42)$$

і модель зап'ястя хірурга

$$[x_w(t), y_w(t), z_w(t)]^T = \sum_{i=1}^3 C_w^i \cdot \varphi_w^i(W_w^i[X(t-i), Y(t-i), Z(t-i)]^T) \quad (5.43)$$

Зверніть увагу, що введіть $[X(t-i), Y(t-i), Z(t-i)]^T$ однакова для всіх трьох моделей (5.38), (5.42) і (5.43).

Для навчання мережі було обрано алгоритм Левенберга-Марквардта. Після навчання мережі різні набори даних (отримані під час іншої хірургічної операції) були використані для перевірки моделі. На рисунках 5.20-5.22 представлені результати моделювання прогнозу для 5 часових кроків.

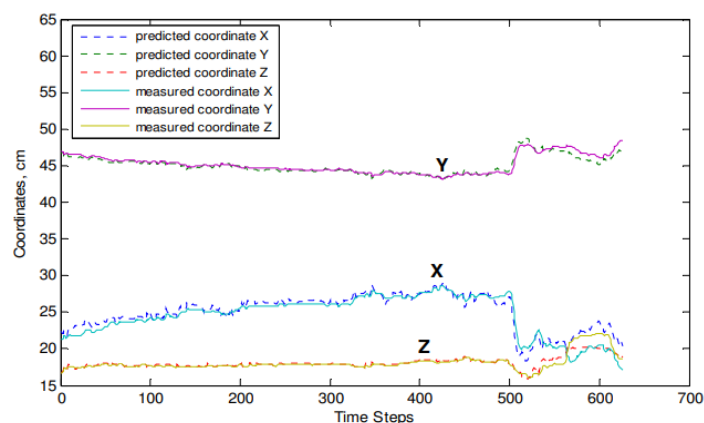


Рисунок 5.20 – Прогноз координат грудної клітки хірурга за 5 часових кроків

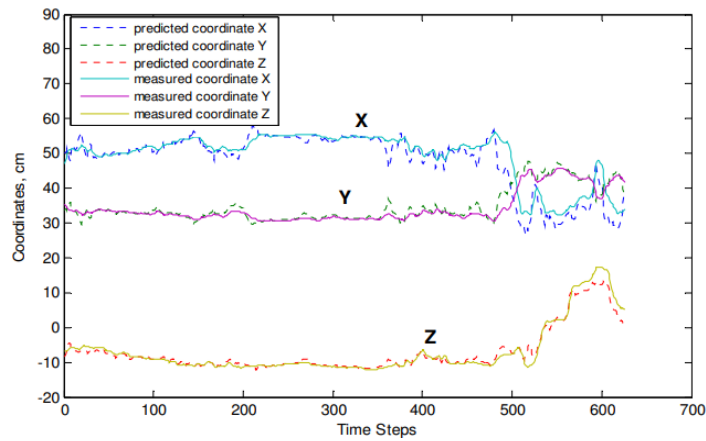


Рисунок 5.21 – Прогноз координат ліктя хірурга за 5 часових кроків

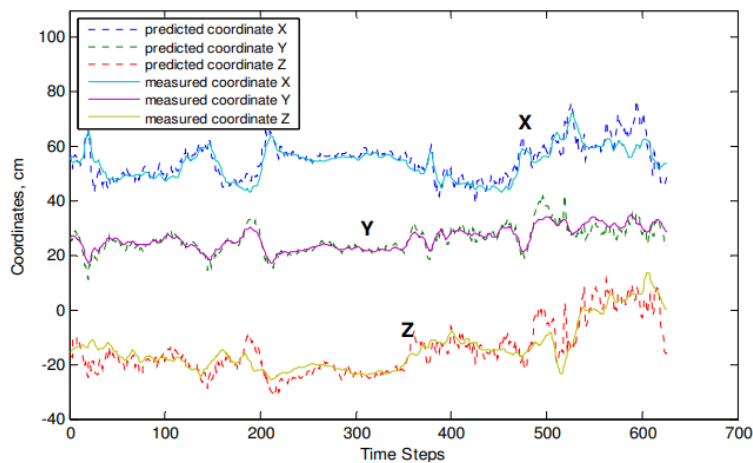


Рисунок 5.22 – Прогноз координат зап'ястя хірурга за 5 кроків часу

З рисунків видно, що модель здатна з високою точністю передбачати координати грудної клітки, ліктя та зап'ястя до 5 часових кроків. Для валідаційного експерименту, показано на цих рисунках, середні абсолютні похибки прогнозу грудних координат X, Y і Z, координат ліктя X, Y і Z і координат X, Y і Z зап'ястя становлять 7,3 мм, 2,6 мм, 2,5 мм, 1,68 см, 8,8 мм, 8,4 мм, 2,3 см, 1,5 см, 3,4 см відповідно. Це дуже хороша точність, особливо якщо врахувати той факт, що похибки вимірювання системи обробки зображень зберігаються як у наборі перевірки, так і в наборі для навчання. Координати були отримані системою обробки зображень, яка

відстежувала положення 3 кольорових маркерів діаметром близько 3 см. Таким чином, похибки прогнозування не перевищують діаметри маркерів. Можна також згадати, що під час навчання нейронної мережі ми фільтруємо випадковий шум, який присутній у навчальній сукупності, як це було продемонстровано в числовому прикладі 5.1. Отже, ми припускаємо, що помилки вимірювань фільтруються під час процедури ідентифікації. Порівняння вихідних даних моделі з відфільтрованими помилками вимірювань із шумним набором даних перевірки також викликає певні помилки перевірки.

Ці прогнозовані координати були успішно використані для виявлення поточного руху хірурга, як описано в [73]. Прогнозування 5 часових кроків (0,166 секунди) достатньо для виявлення руху та алгоритмів обробки зображень. Це також дуже хороший прогноз, якщо врахувати, що деякі дії під час реальної хірургічної операції, як-от передача або вилучення інструменту, займають не більше 1 або максимум 2 секунд.

Модель навчалася на даних, отриманих для конкретного хірурга. Модель може дещо відрізнятись для іншого хірурга з іншою довжиною руки і так далі. Отже, дуже важливо, щоб модель була отримана у вигляді а нейронної мережі і тому може бути дуже легко адаптований для іншого хірурга, незначна зміна положень камер спостереження і так далі. Це можна зробити на робочому етапі, який на початку операції триває не менше 10 секунд.

Цей приклад демонструє, що структура MIMO ANARX на основі NN, яка має обмежену зв'язність через розділення різних екземплярів часу, може бути дуже успішно використана для отримання досить складних моделей, коли використання класичних структур NARX на основі NN практично неможливе через велику кількість зв'язків і, відповідно, параметрів, які потрібно розрахувати за алгоритмом навчання. У деяких випадках необхідність відокремлення екземплярів часу також дуже очевидна через природу змодельованого процесу.

ВИСНОВКИ

У кваліфікаційній роботі розглядаються проблеми ідентифікації нелінійних систем штучними нейронними мережами з різною структурою та застосовність цих моделей для модельних алгоритмів керування.

Хоча добре відомо, що двошаровий перцептрон здатний апроксимувати будь-яку безперервну нелінійність, що означає, що більшість нелінійних динамічних систем можна ідентифікувати шляхом навчання багат шарового перцептрона із зовнішнім зворотним зв'язком на набір вхідно-вихідних даних, результати цього дослідження демонструють важливість вибору конструкції. Структура моделі на основі нейронної мережі має бути обрана відповідно до вимог конкретного алгоритму керування або відповідної структурним особливостям керованої системи. Штучна нейронна мережа, що складається з простих вузлів (штучних нейронів), з'єднаних між собою таким чином, який можна вибрати відповідно до наших потреб, є дуже хорошим, якщо не найкращим інструментом для отримання моделі попередньо визначеної структури.

По-перше, вибір належної структури моделі, що повторює структуру фізичної системи, може значно покращити якість моделі, що призведе до підвищення якості алгоритму керування на основі моделі. У цій кваліфікаційній роботі було продемонстровано ідентифікацію та керування нелійними системами типу Гаммерштейна. Відповідна структура нейронної мережі, що представляє модель Хаммерштейна, запропонована та застосована для управління нелійними системами цього класу на основі моделі. Насамперед це системи зі статичними нелінійністю приводу.

По-друге, зміна структури мережі розширює кількість алгоритмів, які можна комбінувати з моделюванням на основі нейронних мереж та адаптацією на основі навчання. У цій роботі параметри адитивних моделей NARX для різних нелінійних систем були ідентифіковані шляхом навчання

нейронної мережі відповідної структури (модель NN-ANARX), що дозволяє комбінувати ідентифікацію та адаптацію на основі нейронної мережі з динамічним виходом лінеаризація зворотного зв'язку та застосування цього комбінованого методу керування для керування широким класом нелінійних систем.

Якість незалежних від моделі алгоритмів можна покращити шляхом підвищення точності моделі. У кваліфікаційній роботі показано, що запропонована модель Хаммерштейна на основі нейронних мереж значною мірою якістю моделі прогностного керування системами зі статичними нелінійністю приводу.

У цій кваліфікаційній роботі досліджуються та порівнюються методи управління прогностичним керуванням, керування на основі зворотної моделі та динамічною лінеаризацією на основі моделі NN-ANARX. Їх переваги та недоліки виведені з результатів експерименту. Методика керування, заснована на лінеаризації динамічного вихідного зворотного зв'язку моделі NN-ANARX, може допомогти подолати декілька недоліків інших методів. Цей метод заснований лише на прямій динамічній моделі керованої системи. Модель повинна працювати тільки в замкнутому циклі. Не потрібні прогнози, розраховані за допомогою моделювання окремої моделі. Отже, помилки моделі не накопичуються. У той же час обмеження, накладені структурою ANARX на матриці зв'язності нейронної мережі, що представляють цю модель, не викликають серйозних недоліків у якості ідентифікації та дають значні переваги в програмах керування. Тому значна увага в даному дослідженні приділяється ідентифікації нелінійних систем за моделями ANARX на основі NN та їх контролю шляхом лінеаризації динамічної вихідної зворотної зв'язку цієї моделі.

Запропоновано структури ANARX на основі NN для ідентифікації систем МІМО. Також розроблено два методи для лінеаризації нелінійних систем МІМО на основі динамічного вихідного зворотного зв'язку на основі моделі NN-ANARX.

Спрощена модель ANARX (NN-SANARX) на основі NN була запропонована для ідентифікації нелінійних систем SISO та MIMO. Така структура, що має один лінійний підрівень прихованого шару, значно спрощує обчислення точних значень керуючих сигналів (не оцінених чисельним методом) для лінеаризації динамічного зворотного зв'язку цієї моделі. Доходить до рішення лінійного рівняння (у випадку систем SISO) або системи лінійних рівнянь (у випадку системи). Модель NN-SANARX була успішно застосована для управління динамічним зворотним зв'язком на основі лінеаризації нелінійних систем MIMO з однаковою кількістю входів і виходів. Також наведено критерії застосовності конкретної моделі MIMO NN-SANARX для управління на основі динамічної лінеаризації.

Другий підхід базується на повністю нелінійній, а не спрощеній (без лінійних підшарів) моделі NN-ANARX нелінійної системи MIMO, і тому цей метод можна використовувати для керування більш широким класом нелінійних систем. У той же час у кваліфікаційній роботі показано, що перша методика контролю має більшу надійність. Обидва методи накладають ряд обмежень на онлайн-адаптацію моделі. Синаптичні ваги мають бути зафіксовані. Відповідні поправки внесено в адаптацію стека історії та розроблено адаптивні контролери для нелінійних систем MIMO. Показано, що ці контролери здатні одночасно відстежувати декілька бажаних опорних сигналів, компенсуючи небажаний вплив вхідних і вихідних збурень, а також зміни параметрів керованої системи.

Ефективність запропонованих методик демонструється на численних чисельних прикладах.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. W. McCulloch and W. Pitts, "A Logical Calculus of Ideas Immanent in Nervous Activity," *Bulletin of Mathematical Biophysics*, 1943, vol. 5, pp. 115-133.
2. D. Cleveland, *How Do We Know How the Brain Works*, Rosen Publishing Group, 2005.
3. J. L. Shearer, B. T. Kulakovski, J. F. Gardner, *Dynamic Modeling and Control of Engineering Systems*, Prentice Hall, Upper Saddle River, New Jersey, 1997.
4. H. K. Khalil, *Nonlinear Systems*, Macmillan Publishing Company, New York, 1992.
5. H. Nijmeier, A. J. van der Schaft, *Nonlinear Dynamical Control Systems*, Springer-Verlag, New-York, 1990.
6. H. J. Marquez, *Nonlinear Control Systems*, John Wiley & Sons, New Jersey, 2003.
7. R. Iserman, *Digital Control Systems*, Springer-Verlag Berlin, Heidelberg, 1981.
8. Ü. Kotta, F. Chowdhury, and S. Nõmm, "On realizability of neural networks-based input-output models in the classical state space form," *Automatica*, 2006, vol. 42, no. 7, pp. 1211-1216.
9. Ü. Kotta and N. Sadegh, "Two approaches for state space realization of NARMA models: bridging the gap," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 8, no. 1, pp. 21-32, 2002.
10. F. N. Chowdhury, Ü. Kotta and S. Nõmm, "On realizability of neuralnetworks-based input-output models," *Proc. of the 3rd Int. Conf. on Differential Equations and Applications*, St. Petersburg, Russia, vol 6., pp. 47-51, 2000.
11. Ü. Kotta, S. Nõmm, and F. Chowdhury, "On a new type of neural

- network-based input-output model: The ANARMA structure,” Proc. of the 5th IFAC Symposium on nonlinear control systems NOLCOS, St. Petersburg, Russia, July 2001.
12. F. N. Chowdhury. “Input-output modeling of nonlinear systems with time-varying linear models. IEEE Transactions on Automatic Control, vol. 7, pp. 1355-1358, 2000. 156
 13. E. Petlenkov, S. Nõmm, and Ü. Kotta, “NN-based ANARX structure for identification and model-based control,” Proc: 9th International Conference on Control Automation Robotics & Vision (ICARCV 2006), Singapore, 2006, pp. 2284-2288.
 14. E. Petlenkov, S. Nõmm, and Ü. Kotta, “Adaptive Output Feedback Linearization for a Class of NN-based ANARX Models,” Proc: 6th IEEE International Conference on Control and Automation, Guangzhou, China, May 30 – June 1, 2007, pp. 3173-3178.
 15. K. Ogata, State Space Analysis of Control Systems, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1967.
 16. M. Mrabet, F. Fnaiech, A. Chaari, and K. Al-Haddad, “Nonlinear Predictive Control Based on NARX Models with Structure Identification,” in Proc. of the 28th Annual IEEE Conference of the Industrial Electronics Society (IECON02), vol. 3, Sevilla, Spain, November 2002, pp. 1757-1762.
 17. K. J. Hunt, G. R. Irwin and K. Warwick, Neural Network Engineering in Dynamic Control Systems: Advances in Industrial Control, Springer-Verlag, London, 1995.
 18. J. B. Rowlings, “Tutorial: Model Predictive Control Technology,” in Proc. of the American Control Conference, 1999, Vol. 1, San Diego, California, USA, June 1999, pp 662-676.
 19. V. Peterka, “Predictor Based Self-Tuning Control,” Automatica, vol. 20, no. 1, pp 39-50, 1984.
 20. D. W. Clarke, C. Mohtadi and P. S. Tuffs, “Generalized Predictive

- Control – Part 1. The basic algorithm,” *Automatica*, 1987.
21. C. Kambhupati, K. Warwick and C. S. Berger, “A Comparative Study of Multilayered and Single Layered Neural Network Based Predictive Controllers,” *IEE International Conference on Intelligent Systems Engineering*, pp. 293-298, 1992.
 22. H. Chen and F. Allgöwer, “A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability,” *Automatica*, vol. 34, no. 10, pp. 1205-1217, 1998.
 23. G. W. Irwin., K. Warwick., K. J. Hunt., *Neural network applications in control*, The Institution of Electrical Engineers, 1995.
 24. S. Omatu, M. Khalid and R. Yusof, *Neuro-Control and its Applications: Advances in Industrial Control*, Springer-Verlag, London, 1996.
 25. F. Garces, V. M. Becerra, C. Kambhampai and K. Warwick, *Strategies for Feedback linearization. A Dynamic Neural Network Approach*, Springer-Verlag, London, 2003. 157
 26. E. Petlenkov and E. Rüstern, “Experiments in Neural Network Inverse Modelling Based Control for a Class of Nonlinear Systems,” *Proc. of the 9th Biennial Baltic Electronics Conference: BEC2004*, Tallinn, Estonia, October 3-6, 2004, pp. 145-148.
 27. M.V. Chen, M. S. Zalzal and N. E. Sharkey, “Towards a Comparative Study of Neural Networks in Inverse Model Learning and Compensation Applied to Dynamic Robot Control,” *Proc of the Fifth International Conference on Artificial Neural Networks (Conf. Publ. No. 440)* , 7-9 July 1997, pp. 146-151.
 28. E. Colina-Morles and N. Mort, “Inverse Model Neural Network-Based Control of Dynamic Systems,” *Control '94. International Conference*, vol. 2, 21-24 March 1994, pp.955-960.
 29. G. P. Plett, “Adaptive Inverse Control of Linear and Nonlinear Systems Using Dynamic Neural Networks,” *IEEE Trans. on Neural Networks*, vol.14, no.2, March 2003, pp.360-376.

30. S. Ushida and H. Kimura, "FEL and JIT Approaches to Tracking Adaptive Control Based on the Internal Inverse Models," Proc. Of 42nd IEEE Conference on Decision and Control, Vol. 6 , December 9- 12, 2003, pp. 6363 – 6368.
31. R. Boukezzoula, S. Galichet, and L. Foulloy, "Nonlinear Internal Model Control: Application of Inverse Model Based Fuzzy Control," IEEE Transactions on Fuzzy Systems, Vol. 11, Issue 6, December, 2003, pp. 814-829.
32. R. Pothin, Ü . Kotta, and C. Moog, "Output feedback linearization of nonlinear discrete-time systems," Proc. of the IFAC Conf. on Control system design, Bratislava, 2000, pp. 174–179.
33. S. Nõmm, Realization and Identification of Discrete-time Nonlinear Systems, PhD thesis, Tallinn University of Technology, Tallinn, 2004.
34. W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, Numerical Recipes in C: The Art of Scientific Computing, Cambridge University Press, 1992.
35. S. R. Cajal, "Histologie du systeme nerveux de l-homme et des vertebres", Maloine, Paris, 1909.
36. S. Haykin, Neural Networks, Prentice-Hall international (UK) Limited, London, 1994.
37. Y-Z. Lu, Industrial intelligent control. Fundamentals and applications. Chicher: Wiley, 1996.
38. T. Hrycey, Neurocontrol. Towards an Industrial Control Methodology, John Wiley & Sons, Inc., Toronto, 1997. 158
39. P. M. Mills, A. Z. Zomaya and M. O. Tade, Neuro-Adaptive Process Control, John Wiley & Sons, Inc., England, 1996.
40. L. H. Tsoukalas and R. E. Uhrig, Fuzzy and neural approaches in engineering. Wiley-Interscience, New York, 1996.
41. D. E. Rumelhart and J. L. McClelland, Parallel Distributed Processing, vol. 1 Foundation, MIT Press, Cambridge, 1988.

42. K. J. Åström and B. Wittenmark, "On Self-Turning Regulators," *Automatica*, 1973, vol. 9, pp. 185-199.
43. K. J. Åström and B. Wittenmark. *Adaptive control*. Addison-Wesley Publishing Company, 1989.
44. M. H. Stone, "The generalized Weierstrass approximation theorem," *Mathematics Magazine*, vol. 21, 1948, pp. 167-184, 237-254.
45. J. L. Elman, "Finding Structure in Time," *Cognitive Science*, vol. 14, pp. 179-211, 1990.
46. E. Haselsteiner, "What Elman networks cannot do," *Proc. Of The 1998 IEEE International Joint Conference on Neural Networks. IEEE World Congress on Computational Intelligence*, vol. 2, pp. 1245-1249, Anchorage, Alaska, USA May 1998.
47. E. Sontag, "Neural nets as systems models and controllers," *Proc. Of the Seventh Yale Workshop on Adaptive and Learning Systems*, pp. 73- 79, Yale University, 1992.
48. D. O. Hebb, *The Organization of behavior*, Wiley, New York, 1949
49. F. Rosenblatt, "Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms," Spartan Book, Washington, 1961.
50. K. Levenberg, "A method for the solution of certain problems in least squares, *Quart. Appl. Math.*, 1944, Vol. 2, pp. 164–168.
51. D. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *SIAM J. Appl. Math.*, 1963, Vol. 11, pp. 431–441.
52. A. Toledo, M. Pinzolas, J. J. Ibarrola and G. Lera, "Improvement of the neighborhood based Levenberg-Marquardt algorithm by local adaptation of the learning coefficient," *Neural Networks, IEEE Transactions on*, Vol. 16, Issue 4, July 2005, pp. 988-992.
53. M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *Neural Networks, IEEE Transactions on*, Vol. 5, Issue 6, November 1994, pp. 989-993.
54. F. Declercq and R. De Keyser, "Comparative study of neural predictors

- in model based predictive control” Proc. of The International Workshop on Neural Networks for Identification, Control, Robotics, 159 and Signal/Image Processing, Venice, Italy, 21-23 Aug. 1996, pp. 20- 28.
55. E. Petlenkov and E. Rüstern, “Linear Dynamic Systems with Static Actuator Nonlinearities Identification for Control,” Proceedings of the ICGST Automatic Control and System Engineering Conference: ACSE-05, December 19-21, 2005, Cairo, Egypt, pp. 41-46.
 56. E. Petlenkov, „NN-ANARX Structure Based Dynamic Output Feedback Linearization for Control of Nonlinear MIMO Systems,“ In Proc. Of The 15th Mediterranean Conference on Control and Automation, MED'07, Athena, Greece, June 2007, pp. 1-6.
 57. E. Petlenkov, and J. Belikov, “NN-ANARX Structure for Dynamic Output Feedback Linearization of Nonlinear SISO and MIMO Systems: Neural Networks Based Approach,” In Proc. Of the 26th Chinese Control Conference, Zhangjiajie, China , July 2007, vol. 4, pp. 138-145.
 58. H. Demuth, M. Beale, Neural network toolbox: for use with MATLAB, The MathWork, Inc., 1998.
 59. C. L. Harris, and X. Hong, “Neurofuzzy mixture of experts network parallel learning and model construction algorithms,” IEE Proc.- Control Theory Applications, vol. 148, num. 6, November, 2001, pp. 456-466.
 60. F. C. Chen and H. K. Khalil, “Adaptive Control of a class of nonlinear discrete-time systems using neural networks,” IEEE transactions on Automatic Control, 1995, vol. 40, num. 5, pp. 791-801.
 61. S. Chen, S. Billings, and P. Grant, “Non-linear system identification using neural networks,” International Journal on Control, 1990, vol. 51, no. 6, pp. 1191–1214.
 62. K. Narendra, and K. Parthasarathy, “Identification and control of dynamical systems using neural networks,” IEEE Transactions on Neural Networks, March 1990, vol. 1, pp. 4–27.
 63. K. Narendra, and S. Mukhopadhyay, “Adaptive Control Using Neural

- Networks and Approximate Models,” IEEE Transactions on Neural Networks, vol. 8, no. 3, pp. 475-485, 1997.
64. H. T. Siegelmann, B. G. Horne and C. L. Giles, “Computational Capabilities of Recurrent NARX Neural Networks,” IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics, vol. 27, no. 2, April 1997, pp. 208-215.
 65. S. Billings, “Identification of Nonlinear Systems - A survey,” Proceedings of IEE, Part D, 1980, pp. 127-272. 160
 66. G. A. Pajunen, “Application of a Model Reference Adaptive Technique to the Identification and Control of Wiener Type Nonlinear Processes,” Thesis for the degree of Doctor of Technology, Helsinki University of Technology, Espo, Finland, 1984.
 67. B.-J. Yang, A. J. Calise, and N. Hovakimyan, “Augmenting adaptive output feedback control of uncertain nonlinear systems with actuator nonlinearities,” Proceedings of the 2004 American Control Conference, Boston, Massachusetts, Vol. 5, June 30 - July 2, 2004, pp. 4675-4680.
 68. M. Polycarpou, J. Farrell and M. Sharma, “On-line Approximation Control of Uncertain Nonlinear Systems: Issues with Control Input Saturation,” Proceedings of the 2003 American Control Conference, Denver, Colorado, Vol. 1, 4-6 June 2003, pp 543-548.
 69. R. R. Selmic, V. V. Phoha, and F. L. Lewis, “Intelligent Compensation of Actuator Nonlinearities,” Proc. Of the 42nd IEEE Conference on Decision and Control, Maui, Hawaii, Vol. 4, 9-12 Dec, 2003, pp 4327-4332.
 70. C. K. Lee, T. H. M. Chow and D. K. W. Cheng, “DC Servo Motor Controllers Based on a Neural Network,” Proc. of the First International Conference on Intelligent Systems Engineering, Conf. Publ. No. 360, 19-21 Aug 1992, pp. 287-292.
 71. Laboratoire de Meteorologie Dynamique. Chemical and nuclear engineering. System dynamics. Mathematical Modeling of Engineering

- Systems, “Dynamic Model of a Permanent Magnet DC Motor,” <http://gershwin.ens.fr/vdaniel/Doc-Locale/Cours-Mirrored/Methodes-Maths/white/sdyn/s6/s6fmathm/s6fmathm.html> [February, 2007].
72. S. A. Billings and M. B. Fadzil, “The practical identification of systems with nonlinearities,” Proc. of the 7th IFAC/IFORS Symposium on Identification and System Parameter Estimation, 1985, pp. 155–160.
 73. S. Nömm, E. Petlenkov, J. Vain, K. Yoshimitsu, K. Ohnuma, T. Sadahiro and F. Miyawaki, „NN-based ANARX Model of the Surgeon’s Hand for the Motion Recognition,” Proc. of the 4th COE Workshop on Human Adaptive Mechatronics (HAM), Tokyo Denki University, Japan, March 2007, pp. 19-24.
 74. R. Pearson, Ü. Kotta, and S. Nömm, “Systems with associative dynamics,” *Kybernetika*, vol. 38, no. 5, pp. 585–600, 2002.
 75. T. Knapp and H. Budman, “Robust control design of non-linear processes using empirical state-affine models,” *Int. J. Control*, no. 73 pp. 1525–1535, 2000. 161
 76. J. Belikov, E. Petlenkov and S. Nömm, „Application of Neural Networks based ANARX structure to backing up control of a trucktrailer,“ in Proc. of the 6th IFAC Symposium on Intelligent Autonomous Vehicles, Toulouse, France, September 2007.
 77. T. F. Junge and H. Unbehauen, “Recursive Identification of a TurboGenerator Plant Using Structurally Adaptive Neural Networks,” in Proc. of IEEE International Conference on Industrial Technology 2000, Goa, India, vol. 1, pp. 572-577, January 2000.
 78. F. Miyawaki, K. Masamune, S. Suzuki, K. Yoshimitsu, and J. Vain, “Tscrub nurse robot system – intraoperative motion analysis of a scrub nurse and timed-automatabased model for surgery,” *IEEE Transaction on Industrial Electronics*, vol. 5 no. 52, pp. 1227-1235, 2005.
 79. K. Ohnuma, K. Masamune, K. Yoshimitsu, F. Miyawaki, J. Vain, and Y. Fukui, “Analysis and recognition of a surgeon’s motions in laparoscopic

cholecystectomy giving a scrub nurse robot suitable timings for instrument exchange,” In Proceedings of the 3rd COE Workshop on Human Adaptive Mechatronics (HAM), Tokyo Denki University, Japan, pp. 1–5, March 2006.

80. R. Alur, and D. L. D. Gerd, “Theory of timed automata,” TCS, vol. 2, no. 126, pp. 183–235, 2004.