

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Центр післядипломної освіти _____
(повна назва)

Кафедра _____ Програмної інженерії _____
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

_____ другий (магістерський) _____
(рівень вищої освіти)

Дослідження алгоритмів непараметричного керування в багатомірних
системах

(тема)

Виконав: студент 2 курсу, групи ІПЗмзд-18-1
спеціальності 121- Інженерія програмного
забезпечення _____

(код і повна назва спеціальності)

освітньо-наукової програми Інженерія
програмного забезпечення _____

(повна назва освітньої програми)

_____ Панфьоров О.Т. _____

(прізвище, ініціали)

Керівник _____ проф. Шостак І.В. _____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф. _____

З.В.Дудар

2020 р.

Харківський національний університет радіоелектроніки

Факультет Центр післядипломної освіти

Кафедра програмної інженерії

Рівень вищої освіти другий (магістерський)

Спеціальність 121 – Інженерія програмного забезпечення

(код і повна назва)

Освітньо-наукова програма Інженерія програмного забезпечення

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

Студентові Панфьорову Олегу Тимофійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження алгоритмів непараметричного керування в багатомірних системах

затверджена наказом по університету від « _____ » _____ 2020 р № _____

2. Термін подання студентом роботи до екзаменаційної комісії «21» травня 2020 р.

3. Вихідні дані до роботи Алгоритми керування, методи взаємодії багатомірних систем, методи обробки великих даних та пояснювальна записка. Використовувати ОС Windows, середовище об'єктно-орієнтованого проектування.

4. Перелік питань, що потрібно опрацювати в роботі мета роботи, аналіз проблемної галузі і постановка задачі, методи пошуку корисних даних, опис об'єктних моделей, використовувані методи та алгоритми, архітектура програмної системи, опис розробленої програмної системи, результати тестування програмної системи

5. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	проф. Шостак І.В.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1.	Аналіз предметної галузі	25 березня 2020 р.	
2.	Огляд існуючих методів	31 березня 2020 р.	
3.	Методи кооперації штучних агентів	15 квітня 2020 р.	
4.	Підготовка пояснювальної записки	20 квітня 2020 р.	
5.	Спецчастина	28 квітня 2020 р.	
6.	Підготовка презентації та доповіді	03 травня 2020 р.	
7.	Попередній захист	15 травня 2020 р.	
8.	Нормоконтроль, рецензування	17 травня 2020 р.	
9.	Занесення диплома в електронний архів	18 травня 2020 р.	
10.	Допуск до захисту у зав. кафедри	20 травня 2020 р.	

Дата видачі завдання _ « _____ » _____ 2020 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Шостак І.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Пояснювальна записка до атестаційної роботи: 77 с., 8 табл., 41 рис., 3 дод., 19 джерел.

ДИСКРЕТНО-БЕЗПЕРЕРВНІ ПРОЦЕСИ, СТРУКТУРИ З ЗАПІЗНЕННЯМ, ТЕОРІЯ ІДЕНТИФІКАЦІЇ, ПАРАМЕТРИ, РЕГУЛЯТОРИ

Мета роботи полягає в побудові й дослідженні непараметричних моделей і алгоритмів керування для багатомірних дискретно-безперервних процесів.

Метод – математичне та програмне моделювання

Об'єкт дослідження – методи параметричної теорії ідентифікації, теорії керування, теорії адаптивних та систем що навчаються.

В результаті розроблено алгоритми системи керування із зовнішнім контуром та тестове програмне забезпечення

DISCRETE-CONTINUOUS PROCESSES, DELAYED STRUCTURES, IDENTIFICATION THEORY, PARAMETERS, REGULATORS.

The purpose of the work is to build and study nonparametric models and control algorithms for multidimensional discrete-continuous processes.

Method – mathematical and software modeling

Object of research – methods of parametric theory of identification, control theory, theory of adaptive and learning systems.

As a result, algorithms for the control system with an external circuit and test software were developed

ЗМІСТ

Вступ	5
1 Аналіз моделей ідентифікації та постановка задач дослідження.....	8
1.1 Аналіз методів ідентифікації в умовах неповної інформації	8
1.2 Параметричні моделі безінерційних об'єктів з запізнюванням	15
1.3 Аналіз матриць спостереження	17
1.4 Алгоритми заповнення матриці спостережень із пропусками	18
1.5 Постановка задач дослідження	23
2 Опис проведених теоретичних досліджень	26
2.1 Аналіз «трубчастих» структур	26
2.2 Аналіз алгоритмів ідентифікації	31
2.3 Моделювання дискретно-безперервних процесів	32
3 Аналіз результатів досліджень.....	36
3.1 Опис алгоритмів ідентифікації «трубчастих» процесів	36
3.2 Обчислювальний експеримент	37
3.3 Оцінка похибок моделювання	43
3.4 Алгоритми керування макрооб'єктами	44
4 Опис функціонування розробленої програмної системи.....	46
4.1 Обчислювальні експерименти	46
4.2 Розрахунок варіантів функціонування	48
5 Опис можливості використання отриманих результатів.....	55
Висновки	58
Перелік джерел посилання	60
Додаток А Лістинг фрагментів коду програми	62
Додаток Б Слайди презентації	67
Додаток В Апробація результатів роботи.....	76

ВСТУП

Завдання керування складним промисловими об'єктами невід'ємне пов'язана з постановкою й рішенням завдання ідентифікації досліджуваного процесу. Це завдання є центральним у багатьох проблемах системного аналізу. Теорії ідентифікації на сьогоднішній день присвячена велика кількість праць як українських, так і закордонних авторів [1]. Залежно від рівня апріорної інформації про об'єкт дослідження виділяють методи параметричної ідентифікації (ідентифікація в «вузькому» змісті) і методи непараметричної ідентифікації (ідентифікація в «широкому» змісті). Методи параметричної ідентифікації припускають досить високий рівень апріорної інформації, тому що тут потрібне визначення параметричної структури об'єкта дослідження. Але на практиці виникають випадки, коли ми змушені працювати в умовах малої апріорної інформації. У цьому випадку доцільно використовувати методи непараметричної ідентифікації. У цьому зв'язку розвиток методів непараметричної ідентифікації представляється актуальним.

Постановка й рішення завдань ідентифікації й керування залежать від класу досліджуваного процесу (статичний, динамічний, лінійний, нелінійний і ін. типи процесів).

В даній роботі досліджується новий клас процесів, названих «трубчастими» (або Н-процесами). Перше згадування про процеси «трубчастої» структури з'являється в [2]. Ці процеси були помічені при моделюванні технологічних процесів у металургії. Було виявлено, що компоненти вектору входу досліджуваного процесу зв'язані стохастичною залежністю, внаслідок чого він протікає не у всій області, встановленої технологічним регламентом підприємства, а лише в деякій його підобласті. Моделювання подібного роду процесів зв'язане з багатьма складностями, зокрема, традиційні методи параметричної ідентифікації не дають задовільного результату. Н-процеси можна вважати новими й, на сьогоднішній день, маловивченими. Але процеси цього

класу все частіше виявляються на практиці, а виходить, вимагають подальшого вивчення.

Велику роль при рішенні завдань ідентифікації й керування різного роду процесами, у тому числі й «трубчастого» типу, відіграє первинна обробка вихідних даних, оскільки від їхньої якості багато в чому залежить і якість рішення завдання ідентифікації. Дані, у свою чергу, можуть мати багато недоліків, приміром, містити в собі пропуски, викиди і т.д. Пропуски й викиди у вибірках спостережень «вхідних-вихідних» змінних процесу знижують точність рішення завдання ідентифікації. Даній проблемі на сьогоднішній день присвячена велика кількість робіт [3]. Становить інтерес рішення проблеми первинної обробки даних в умовах непараметричної невизначеності.

Рівень апріорної інформації про об'єкт особливо важливий при математичному формулюванні завдання керування. У цьому зв'язку проблеми адаптивних систем, що й навчаються, відповідних до різних рівнів апріорної інформації, є на сьогоднішній день найважливішими в теорії автоматичного управління. Потреба в побудові подібних систем виникає в багатьох технологічних, виробничих процесах, а також в інших областях людської діяльності (економіка, соціологія й ін.).

.
. .
. .
. .

1 АНАЛІЗ МОДЕЛЕЙ ІДЕНТИФІКАЦІЇ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Аналіз методів ідентифікації в умовах неповної інформації

Завдання теорії ідентифікації полягає в побудові оптимальної, у деякому змісті, моделі за результатами спостережень над вхідними й вихідними змінними системи [4]. На рис 1.1 представлена загальна схему досліджуваного процесу, прийняту в теорії ідентифікації:

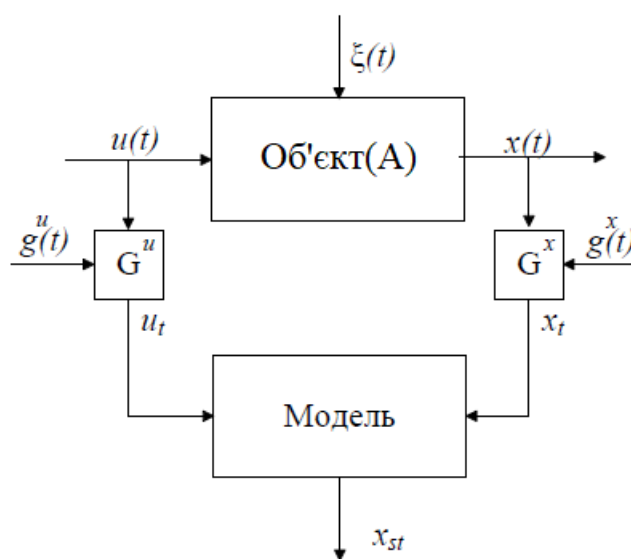


Рисунок 1.1 – Загальна схема досліджуваного процесу

Схема на рис. 1.1 складається із двох блоків: «Об'єкт» і «Модель». Тут прийняті наступні позначення: A – це невідомий оператор об'єкта; $u(t) = (u_1(t), u_2(t), \dots, u_m(t)) \in \Omega(u) \subset R^m$ – векторний вхідний вплив об'єкта розмірністю m ; $x(t) = (x_1(t), x_2(t), \dots, x_n(t)) \in \Omega(x) \subset R^n$ – векторна вихідна змінна об'єкта розмірністю n ; виконується умова $m \geq n$; (t) – безперервний час, Δt – дискретність контролю «вхідних-вихідних» змінних процесу; $\xi(t)$ – векторна випадкова перешкода; блоки контролю змінних G^u , G^x піддані впливу випадкових перешкод $g^u(t)$ і $g^x(t)$; u_t і x_t – виміри змінних $u(t)$ і $x(t)$ у дискретні моменти часу. Є вибірка вимірів «вхідних-

вихідних» змінних процесу $\{u_i, x_i, i = \overline{1, s}\}$, де s – обсяг вибірки. Вимір «вхідних-вихідних» змінних об'єкта надходять на блок «Модель», де на підставі заданого алгоритму перебувають значення виходу моделі x_{st} . Усі випадкові фактори, діючі в каналах виміри й на процес, мають нульові математичні очікування й обмежені дисперсії.

Розглянутий процес ставиться до класу дискретно-неперервних, тобто по своїй природі процес є безперервним, однак «вхідні-вихідні» змінні процесу контролюються через дискретні моменти часу.

Більшість систем, що зустрічаються на практиці, є динамічними, тобто стан такої системи в цей момент часу t залежить як від вхідних впливів, так і від її станів у минулому. Інакше кажучи, динамічна система може бути описана наступним рівнянням:

$$x(t) = A(u(y), x(t-1), x(t-2), \dots, x(t-k), \xi(t)), \quad (1.1)$$

де k – загальний час роботи системи.

Якщо ж стан системи $x(t)$ залежить тільки від вхідних змінних $u(t)$, те система називається статичної, або безінерційних. Така система описується рівнянням виду:

$$x(t) = A(u(t), \xi(t)). \quad (1.2)$$

У деяких випадках доводиться розглядати динамічну систему як статичну систему із запізнюванням. Запізнювання τ може суттєво перевищує постійну часу об'єкта θ ($\tau > 4\theta$). Як відомо, реакція системи на вхідний вплив практично закінчується через час, рівне $(4\div 5)\theta$ [6]. У зв'язку із цим значення $x(t-1), x(t-2), \dots, x(t-k)$ з (1.1) стають відомими лише тоді, коли перехідний процес уже завершений і їх використання не має змісту. У цих випадках система може бути описана як безінерційних система із запізнюванням:

$$x(t) = f(u(t-\tau), \xi(t)), \quad (1.3)$$

де τ – запізнювання, яке може відрізнитися по різних каналах зв'язку. Модель системи, описуваної (1.3), має вигляд:

$$\hat{x}(t) = f(u(t - \tau)), \quad (1.4)$$

Більшу роль при вивченні того або іншого процесу відіграють засоби контролю його «вхідних-вихідних» змінних [6]. Розглянуто наступну схему (рис. 1.2):

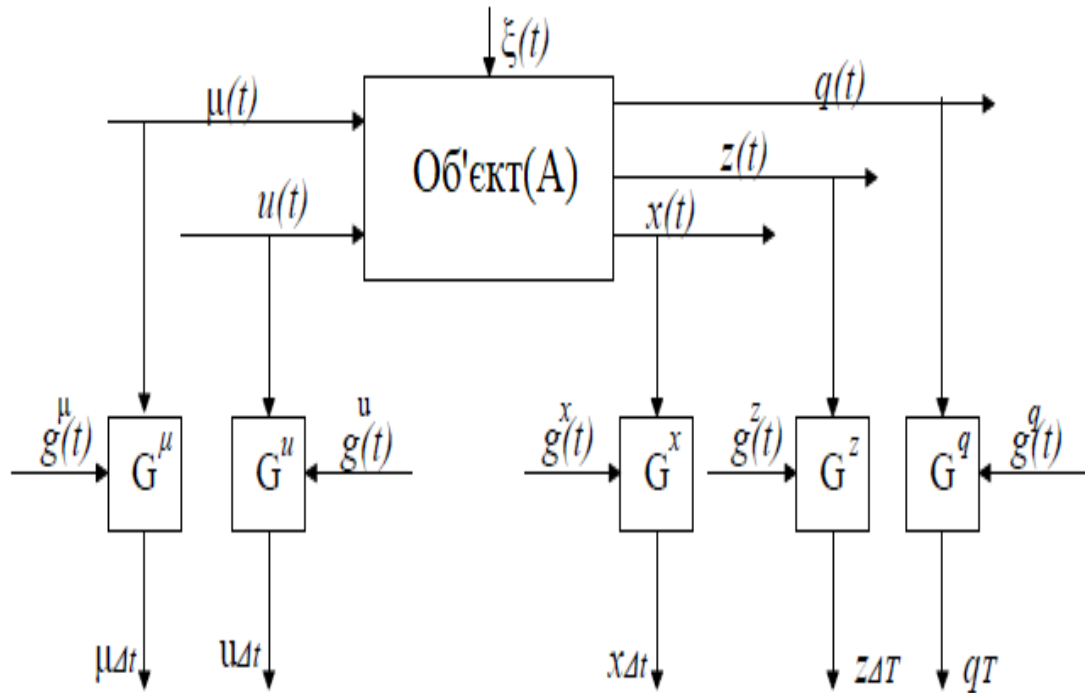


Рисунок 1.2 – Контроль «вхідних-вихідних» змінних процесу

Тут $\mu(t) = (\mu_1(t), \mu_2(t), \dots, \mu_r(t)) \in \Omega(\mu) \subset R^r$ – це вхідна вимірювана, але неконтрольована змінна процесу. Приміром, якщо $u(t)$ – це завантаження подаваного на вхід матеріалу, то $\mu(t)$ може бути фізико-хімічною або технологічною характеристикою цього матеріалу, що суттєво впливає на процес, який протікає в об'єкті, тобто на вихід. Змінні $z(t) = (z_1(t), z_2(t), \dots, z_l(t)) \in \Omega(z) \subset R^l$ і $q(t) = (q_1(t), q_2(t), \dots, q_v(t)) \in \Omega(q) \subset R^v$ – це векторні вихідні змінні процесу. Відзначимо істотну відмінність між вихідними змінними $x(t)$, $z(t)$ і $q(t)$. Вихідні змінні даного процесу контролюються з різною дискретністю: змінна $x(t)$ вимірюється через інтервал часу Δt , змінна $z(t)$ вимірюється через більший інтервал часу ΔT , а змінна $q(t)$ – через інтервал часу T , причому виконується наступна умова $\Delta t \ll \Delta T \ll T$.

Відмінність у дискретності виміру змінних процесу в цьому випадку обумовлене відмінністю в способах їх контролю. Приміром, одні величини можуть бути обмірювані електричними засобами. Подібний контроль не вимагає більших тимчасових витрат. Тут ми самі можемо задати дискретність виміру, що цікавить нас. Виміри інших змінних можуть бути отримані лише за допомогою лабораторного аналізу, або ж шляхом фізико-механічних, фізико-хімічних і іншого роду випробувань, що вимагає значно більшого часу. Слідє відзначити, що найчастіше змінна з найбільшою дискретністю контролю $q(t)$ є найбільш важливою, тобто вона визначає якість продукції, що випускається. Але використовувати її з метою керування в режимі реального часу неможливо. Тому завдання її прогнозування є актуальною.

У цьому випадку вихідна змінна $x(t)$, як і раніше, може бути описана рівнянням (1.4). Для опису змінних з більшою дискретністю виміру $z(t)$ і $q(t)$ доцільно використовувати весь набір змінних, що впливають на їхню поведінку:

$$\hat{z}(t) = A(u(t - \tau), \mu(t), \hat{x}(t)), \quad (1.5)$$

$$\hat{q}(t) = A(u(t - \tau), \mu(t), \hat{x}(t), \hat{z}(t)), \quad (1.6)$$

Ще одна особливість, що властива реальним об'єктам. Розглянуто схему, представлену на рис. 1.3. На вхід об'єкта надходить векторний вхідний вплив об'єкта – векторна вихідна змінна $u(t) = (u_1(t), u_2(t), \dots, u_m(t))$, на виході $x(t) = (x_1(t), x_2(t), \dots, x_n(t))$. На рис. показане, що усередині об'єкта присутні всілякі зв'язки. Це зв'язки як між «вхідними-вихідними» змінними процесу, так і між компонентами вектору входу $u(t)$ і компонентами вектору виходу $x(t)$. Процеси, для яких компоненти вектору входу $u(t)$ або вектору виходу $x(t)$ зв'язані деякої стохастичною залежністю називають Н-процесами (процеси «трубчастого» типу) [7].

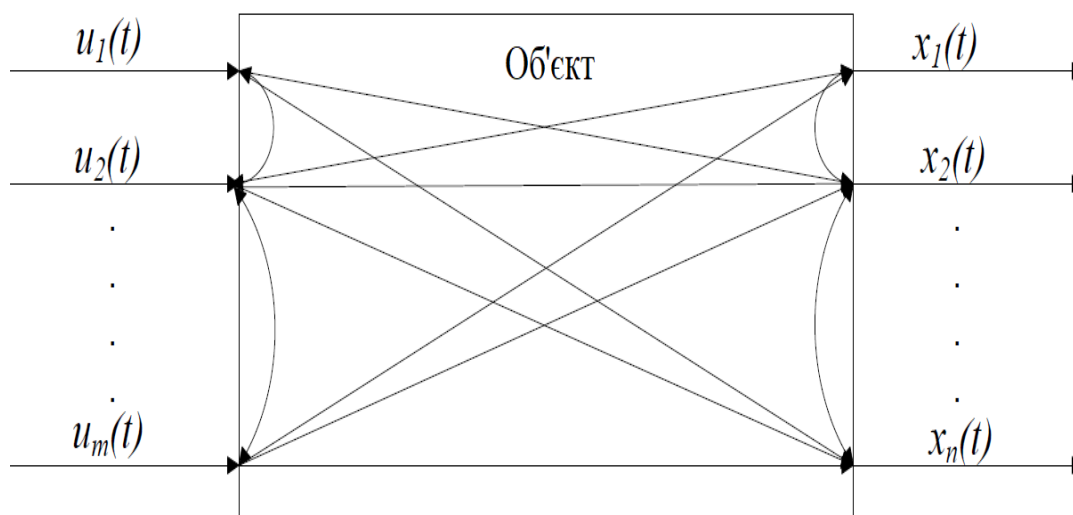


Рисунок 1.3 – Зв'язки між змінними процесу

Зв'язки між змінними процесу можуть мати різний характер. Одні зв'язки можуть бути добре вивчені й описуватися деяким фундаментальним законом (наприклад, рівнянням балансу, законом збереження імпульсу й ін.). Для інших зв'язків ми можемо на основі наявної апріорної інформації скласти параметричну структуру й звести завдання побудови моделі до оцінки невідомих коефіцієнтів. У випадку, коли апріорної інформації недостатньо для знаходження параметричної структури, необхідно використовувати інформацію якісного характеру, тобто методи непараметричної ідентифікації. Таким чином, можна говорити про тріаду, що описує поведінку об'єкта, а саме: фундаментальні закони, параметричні структури, якісні властивості [10].

Для математичного формулювання завдання ідентифікації необхідна апріорна інформація про об'єкт дослідження, яка складається з інформації про його оператора, випадкові перешкоди, критерій оптимальності й обмеження. Критерій оптимальності виражає собою ті вимоги, які повинні бути щонайкраще задоволені, а обмеження визначають наші можливості.

Апріорна інформація може базуватися на фундаментальних законах, що лежать в основі різноманітних фізичних, механічних, електротехнічних, хімічних, біофізичних і інших процесів і об'єктів або результатів попередніх досліджень,

об'єктів що нас цікавлять [11]. Априорні відомості про об'єкт не відновляються, і згодом можуть втратити своє значення (старіння встаткування, різного роду випадкові зміни й ін.).

Слід відрізнити априорну інформацію від поточної (апостеріорної) інформації. Поточна інформація витягає в результаті спостережень за ходом процесу або в результаті експериментів і являє собою вибірки спостережень «вхідних-вихідних» змінних процесу виду $\{u_i, x_i, i = \overline{1, s}\}$. Поточна інформація обновляється в кожний момент часу. Вона може використовуватися для нагромадження відповідної априорної інформації, але найбільш важлива її роль – компенсація недостатнього обсягу априорної інформації.

Повна априорна інформація про процес припускає абсолютно точне знання, якого ніколи немає. Крім того, кожний процес піддається впливу безлічі випадкових факторів. Внаслідок цього, усі випадки, з якими ми зустрічаємося, відповідають неповній априорній інформації.

Виділяють деякі рівні априорної інформації:

- байесів рівень априорної інформації. З точністю до параметрів відомі: параметрична модель досліджуваного об'єкта, закони розподілу випадкових перешкод і рівняння каналів зв'язків. Необхідно оцінити параметри параметричної моделі об'єкта;

- рівень параметричної невизначеності. Параметрична модель об'єкта дослідження відома з точністю до параметрів, які необхідно оцінити. Відомі деякі характеристики випадкових перешкод. Вирішується завдання ідентифікації в «вузькому» змісті;

- рівень непараметричної невизначеності. На цьому рівні априорної інформації відсутній етап визначення параметричної структури досліджуваного об'єкта, тому вимоги до рівня априорної інформації слабшають, але тут потрібна інформація якісного характеру (однозначність або неоднозначність характеристик, лінійність процесу або характер його нелінійності й ін.). Для рішення завдання ідентифікації в цьому випадку застосовують методи

непараметричної статистики [4];

– рівень параметричної й непараметричної невизначеності. Це випадок, коли завдання ідентифікації багато зв'язної системи формулюється в умовах і параметричної, і непараметричної апріорної інформації. Моделі тут являють собою взаємозалежну систему параметричних і непараметричних співвідношень.

Залежно від рівня апріорної інформації про процес розрізняють завдання ідентифікації в «вузькому» і «широкому» змістах.

При моделюванні різноманітних дискретно-безперервних процесів у цей час домінує теорія ідентифікації в «вузькому» змісті (параметрична ідентифікація) [5]. Її суть полягає в тому, що на першому етапі, на підставі наявної апріорної інформації, визначається параметричний клас оператора A^α , наприклад:

$$x_\alpha(t) = A^\alpha(u(t), \alpha), \quad (1.7)$$

де $x_\alpha(t)$ – параметрична модель,

A^α – параметрична структура моделі,

α – вектор параметрів.

На другому етапі здійснюється оцінка параметрів α на основі наявної вибірки $\{x_i, u_i, i = \overline{1, s}\}$, s – обсяг вибірки. Оцінка параметрів може здійснюватися за допомогою численних рекурентних процедур, зокрема, методом стохастичних апроксимацій, або методом найменших квадратів. Успіх рішення завдання ідентифікації в цьому випадку суттєво залежить від того, наскільки «вдало» визначений оператор (1.7). У цей час теорія параметричної ідентифікації є найбільш розвинутою.

Ідентифікація в «широкому» змісті (непараметрична ідентифікація) припускає відсутність етапу вибору параметричного класу оператора [12]. «Апріорна інформація про об'єкт при ідентифікації в «широкому» змісті відсутня або дуже бідна, тому доводиться попередньо вирішувати велика кількість додаткових завдань. До цих завдань ставляться: вибір структури системи й завдання класу моделей, оцінювання ступеня стаціонарності й лінійності об'єкта й

діючих змінних, оцінювання ступеня й форми впливу вхідних змінних на вихідні, вибір інформативних змінних і ін. До теперішнього часу накопичений великий досвід рішення завдань ідентифікації в «вузькому змісті». Методи ж рішення завдань ідентифікації в «широкому» змісті почали розроблятися тільки в останні роки, і тут результати значно скромніше, що в першу чергу можна пояснити надзвичайними труднощами завдання».

Часто виявляється значно простіше визначити клас операторів на основі відомостей якісного характеру, наприклад, лінійності процесу або типу нелінійності, однозначності або неоднозначності й ін. У цьому випадку завдання ідентифікації полягає в оцінюванні цього оператора на основі вибірки $\{x_i, u_i, i = \overline{1, s}\}$:

$$x_s(t) = A_s(u(t), \vec{x}_s, \vec{u}_s), \quad (1.8)$$

де $\vec{x}_s = (x_1, x_2, \dots, x_n)$, $\vec{u}_s = (u_1, u_2, \dots, u_n)$ – часові вектори.

Оцінка оператора A_s може бути здійснена засобами непараметричної статистики. Примітним тут є те, що при цьому виключається етап вибору параметричної структури. Тим самим, можна затверджувати, що ідентифікація в цьому випадку є більш адекватною для деяких реальних завдань. До методів ідентифікації в «широкому» змісті ставиться непараметрична оцінка функції регресії за спостереженнями.

1.2 Параметричні моделі безінерційних об'єктів з запізнюванням

Завдання параметричної ідентифікації полягає у визначенні структури й параметрів системи за спостереженнями. По спостережуваним вхідним впливам і вихідним величинам об'єкта підбираються параметри моделі, що настроюється, забезпечують екстремум деякого критерію, що характеризує якість ідентифікації.

При цьому передбачається, що структура досліджуваного об'єкта відома з точністю до параметрів.

При параметричному підході саме від вибору математичного опису об'єкта багато в чому залежить результат рішення завдання ідентифікації. Загальноприйнята схема параметричної ідентифікації представлена на рис. 1.4.

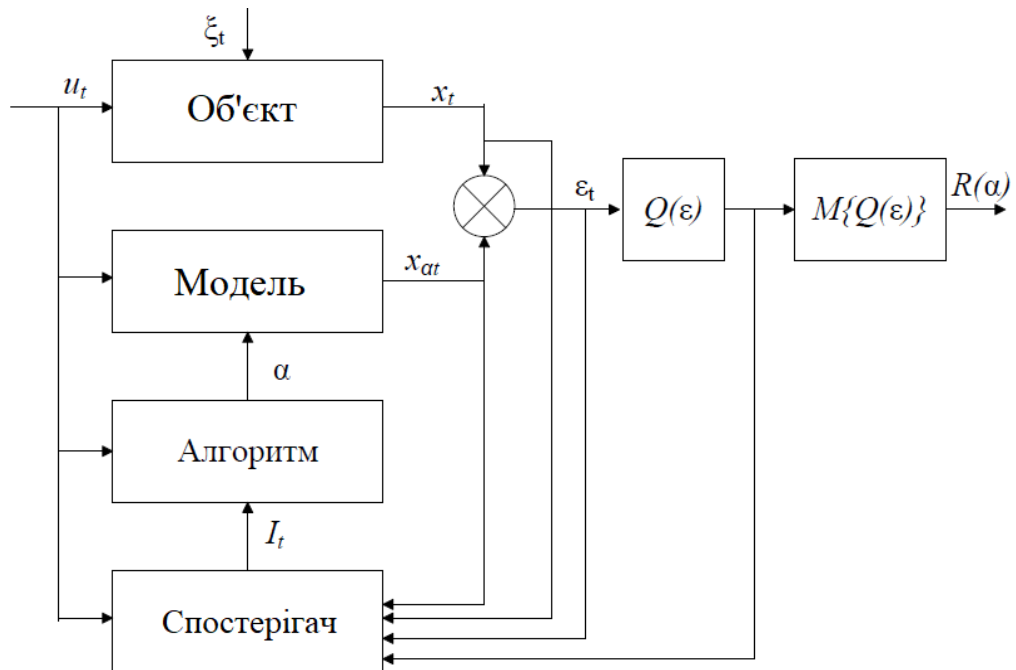


Рисунок 1.4 – Схема завдання ідентифікації

Тут, u_t і x_t значення вхідних і вихідних змінних процесу, відповідно, вимірювані в дискретні моменти часу t ; I_s – вектор параметрів моделі, що настраюється; I_t – вектор усіх спостережень до моменту часу t ; $x_t - x_{at}$ – помилка неузгодженості; $Q(x)$ випукла функція втрат; M – символ математичного очікування; $R(y)$ – критерій ідентифікації.

Ідентифікація за даною схемою здійснюється за допомогою моделі, що настраюється, заданої параметричної структури (блок «Модель»). Параметри моделі а коректуються в міру вступу спостережень «Спостерігач» у блок «Алгоритм». I_s від блоку Відповідність моделі, що настраюється, об'єкту оцінюється критерієм якості ідентифікації:

Блок «Алгоритм» містить у собі певний алгоритм ідентифікації, який дозволяє оцінити параметри моделі, його завдання – мінімізація критерію якості ідентифікації:

Для настроювання параметрів моделі використовують різні ітеративні (рекурентні) методи.

1.3 Аналіз матриць спостереження

Аналіз даних, представлених у матрицях спостережень, містить у собі рішення двох завдань [13]:

- виявлення закономірних зв'язків між елементами таблиці;
- використання виявлених закономірностей для прогнозування значень одних елементів таблиці за відомими значенням інших її елементів.

Ці завдання можуть вирішуватися як по роздільності, так і спільно. Завдання аналізу даних класифікуються залежно від розташування прогнозованих елементів і від їхньої кількості. Крім того, виділяють типи завдань у відповідності зі шкалами, у яких вимірюються значення елементів, що передвіщаються (абсолютна шкала, шкала відносин, інтервалів, порядку, найменувань).

До завдань аналізу даних ставиться й завдання первинної обробки даних. При рішенні завдання ідентифікації ми оперуємо зі спостереженнями «вхідних-вихідних» змінних процесу виду $\{x_i, u_i\}, i = \overline{1, s}$. Від якості цих даних багато в чому залежить і якість рішення поставленого завдання. Тому важливу роль придбає етап первинної обробки даних, що передує процесу моделювання. Даний етап спрямований на обробку деяких особливостей, що присутні у даних, які негативно впливають на процес ідентифікації. До таких особливостей можна віднести наявність у вихідних вибірках пропусків, наявність у вимірах викидів (промахів) і ін.. Викидом зазвичай назвемо аномальне значення вимірюваного

параметра, що не є особливістю досліджуваного процесу, викликане, приміром, збоями апаратури.



Рисунок 1.5 – Класифікація завдань аналізу даних

Викиди – це спостереження, що сильно відрізняються від основної маси елементів вибірки [14]. Також викид може бути наслідком неточних вхідних даних, погрішностей, внесених на окремих етапах вимірів, погрішностей самих методів обчислень, збоїти встаткування, помилки оператора й інших причин. Наявність подібних особливостей значно ускладнює процес моделювання. Більшість відомих методів аналізу даних не здатні обробити подібну інформацію.

1.4 Алгоритми заповнення матриці спостережень із пропусками

Для роботи з даними, що містять викиди, застосовують методи робасної статистики, які дозволяють згладити вплив «промаху» на результати моделювання [15]. Також існують приймання, що дозволяють зовсім виключити викид з вихідної вибірки.

Пропуски «вхідних-вихідних» змінних досліджуваного процесу можуть бути викликані причинами різного характеру й розташовуватися в матриці спостережень хаотично. Вони можуть бути наслідком причин як технічного характеру, наприклад, несправності вимірювального приладу, так і бути викликані помилкою в роботі оператора. Також, причиною може послужити різна дискретність контролю «вхідних-вихідних» змінних процесу.

Розглянуто об'єкт, представлений на рис. 1.6.

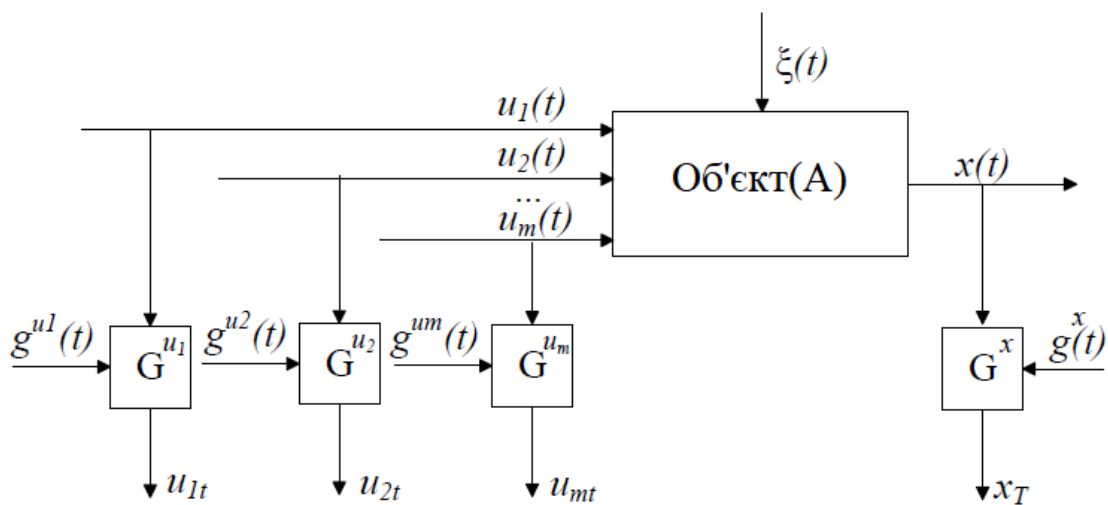


Рисунок 1.6 – Об'єкт із різною дискретністю виміру змінних

Для розглянутого об'єкта векторна вхідна змінна $u(t)$ вимірюється з дискретністю Dt , а вихідна змінна $x(t)$ с дискретністю DT , при цьому $DT = 3Dt$.

У цьому випадку пропуски розташовуються по вихідній змінній $x(t)$ і викликані тривалістю контролю змінної.

Найбільш простий і розповсюджений метод роботи з даними, утримуючими пропуски – виключення рядка із пробілом цілком. Але в цьому випадку разом із пропусками ми викреслюємо з матриці спостережень і заповнені гнізда, а це не розумно із практичної точки зору. До того ж цим скорочується обсяг вихідної вибірки, а для рішення завдань ідентифікації переважно мати вибірки більшого обсягу.

Тому що більшість існуючих методів аналізу даних не розраховане на роботу з матрицями спостережень, що містять пропуски, то пропонується заповнити відсутні значення змінних x у таблиці 1.3 їх оцінками x_s [16]:

При роботі з матрицями спостережень, що містять пропуски, велике значення має механізм породження цих пропусків. Виділяють кілька механізмів виникнення пропусків [15]:

- абсолютно випадкові пропуски (Missing Complete at Random). Дані є, але з якихось причин вони не внесені в матрицю спостережень. Поява пропуску не зв'язана ні із самою пропущеною ознакою, ні з іншими ознаками (приміром, збій у роботі вимірювального приладу);

- випадковий пропуск (Missing at Random). Пропущене значення ознаки теоретично існує, але з якихось причин не може бути заміряне в необхідний момент, поява пропуску зв'язана зі значеннями інших ознак, але не залежить від значення самого пропущеного ознаки (наприклад, занадто більша дискретність контролю змінної, внаслідок чого її значення нам відомо лише через певні проміжки часу);

- не випадковий пропуск (Missing Not at Random) – даних немає, тому що їх не існує – виникнення пропуску залежить як від самого пропущеного значення ознаки, так і від інших ознак (приміром, кількість дітей у жінки, що не родила).

Абсолютно випадкові пропуски (MCR) і випадкові пропуски (MAR) відносяться до припустимих для наближення, тобто тут можливе відновлення пропущеного значення. Не випадкові пропуски відновити не можна, так значення не існує.

Більшу роль при роботі з даними, що містять пропуски, відіграє характер досліджуваного процесу. У випадку, якщо процес є динамічним, коли справжній стан процесу залежить від його станів у минулому, то пропущене значення може бути заповнене середнім між попереднім і наступним спостереженнями. При цьому ми не одержимо грубих помилок. Якщо ж розглянутий процес статичний

(положення в справжньому не залежить від положення в минулому), то усереднення приведе до явних викривлень результатів.

Далі будуть розглядатися припущення, що досліджуваний процес ставиться до класу безінерційних із запізнюванням.

Найбільш простий і розповсюджений метод роботи з даними, що містять пропуски – виключення рядка із пробіл цілком з матриці спостережень. Перевагою тут є простота реалізації. При невеликій кількості пропусків у даних цей підхід може бути задовільним. Але звичайно він не дуже ефективний і приводить до серйозних зсувів [16]. У зв'язку із цим одержали свій розвиток чисельні методи заповнення пропусків у матрицях даних.

Згідно із методом заповнення середнім арифметичним пропущене значення в стовпці заповнюється середнім арифметичним значень цього стовпця (ознаки). Метод відрізняється своєю простотою, але приводить до значних неточностей.

Метод найближчого сусіда базується на припущенні про те, що якщо об'єкти схожі за значенням більшості своїх властивостей, то вони схожі за значенням усіх своїх властивостей. Мірою подібності тут може виступати декартова відстань між крапками. Нехай векторна ознака x' містить пропущене значення k -го компонента. Для того, щоб оцінити пропущене значення x'_k , знаходять декартову відстань між вектором x' і всіма повними векторами x_j :

$$r_j = \sqrt{\sum_{\substack{i=1 \\ i \neq k}}^s (x'_i - x_i^j)^2}. \quad (1.9)$$

Вибираються N векторів (число N задається самостійно) з найменшим значенням r_j і обчислюється оцінка пропущеного значення по формулі:

$$x'_k = \frac{\sum_{i=1}^N C_N x_k^i}{\sum_{i=1}^N C_N}, \quad (1.10)$$

де C_N – вага N -го сусіда, обернено пропорційний декартовій відстані r_N .

Даний метод є приватним варіантом методу заповнення середнім арифметичним і також приводить до неточностей.

Метод максимальної правдоподібності (МП метод). Пропуск замінюється значенням, найбільш імовірним з наявних. Передбачається, що є деяка модель для неповних даних X , якої відповідає щільність $f(X | \theta)$, де θ – невідомий вектор параметрів. Необхідно максимізувати деяку функцію правдоподібності $L(\theta) = f(X | \theta)$ [16].

Zet-Алгоритм, заснований на вивченні «подібності» об'єктів [17]. У його основі лежать три припущення:

- гіпотеза надмірності, яка полягає в тому, що матриця спостережень має надмірність, що проявляється в наявності схожих між собою об'єктів і залежних друг від друга властивостей, якщо це не так, то віддати перевагу одному прогнозу іншому неможливо;

- гіпотеза аналогічності: якщо деяка пара об'єктів близька за значеннями ($n-1$) властивостей, то вона близька й по n -ної властивості;

- гіпотеза локальної компетенції: надмірність носить локальний характер, у кожного об'єкта є підмножина об'єктів-аналогів і в кожній властивості є своя підмножина властивостей аналогів.

У роботі Zet-алгоритму такі етапи:

- для пробілу з вихідної матриці вибирається підмножина рядків-аналогів і підмножина стовпців-аналогів (перебувають компетентні стовпці й рядки). Компетентність j -ої рядка стосовно i -ої рядку визначається величиною K :

$$K_{ij}^{string} = v_{ij} \eta_{ij}, \quad (1.11)$$

де v_{ij} – величина, обернено пропорційна декартовій відстані між рядками i та j ,

η_{ij} – коефіцієнт комплектності,

Комплектність i -го стовпця стосовно j -му стовпцю обчислюється по формулі:

$$K_{ij}^{column} = \kappa_{ij} \eta_{ij}, \quad (1.12)$$

де κ_{ij} – коефіцієнт, який обернено пропорційний коефіцієнту кореляції між i -им і j -им стовпцями,

η_{ij} коефіцієнт комплектності, дорівнює числу властивостей, значення яких відомі як для i -го, так і для j -го стовпчиків.

Розміри комплектної матриці (кількість комплектних рядків і стовпців) задаються самостійно залежно від умов конкретного завдання.

1.5 Постановка задач дослідження

Проведений аналіз методів первинного аналізу даних показує, що, точність рішення завдання ідентифікації залежить від якості вихідних даних, які можуть містити в собі деякі недоліки, наприклад, пропуски й викиди. Пропуски й викиди в даних знижують точність рішення завдання ідентифікації. З метою підвищення точності рішення завдання ідентифікації розглянуто два завдання – завдання заповнення пропусків «вхідних-вихідних» змінних матриць спостережень і завдання видалення викидів з вихідної вибірки.

Завдання відновлення пропусків «вхідних-вихідних» змінних у матрицях спостережень вирішувалася в умовах малої апріорної інформації. Актуальною є методика відновлення пропусків, заснована на непараметричних алгоритмах оцінки функції регресії. Застосування запропонованої методики дозволяє підвищити точність рішення завдання ідентифікації. На точність відновлення впливають: обсяг вихідної вибірки, перешкоди, що діють у каналах вимірів, кількість пропусків у даних і якісні властивості досліджуваного процесу. Природно, що при малих обсягах вибірки, при наявності великої кількості пропусків ефективність застосування методики знижується. Однак, значення цих показників індивідуальні для кожного приватного завдання.

Викиди в даних також впливають на точність рішення завдання ідентифікації. Існуючі робастні алгоритми дозволяють лише послабити вплив викиду на результати оцінювання. Більшість існуючих методів видалення викидів вимагають нормального розподілу даних. У цьому зв'язку потрібно запропонувати алгоритми виключення викидів з вихідної вибірки спостережень, що заснована на непараметричній оцінці функції регресії за спостереженнями. Дана методика дозволяє виключати з вибірки спостережень викиди. Обчислювальні експерименти показали, що пропонована методика дозволяє підвищити точність рішення завдання ідентифікації.

Мета роботи полягає в дослідженні і побудові алгоритмів для непараметричних моделей і алгоритмів керування для багатомірних дискретно-безперервних процесів «трубчастої» структури із запізнюванням, які раніше не були досліджені.

Для досягнення поставленої мети необхідне рішення наступних основних завдань:

- досліджувати непараметричну методику відновлення пропусків «вхідних-вихідних» змінних матриці спостережень;
- проаналізувати непараметричну методику виключення викидів з вихідної матриці спостережень змінних процесу;
- розробити й досліджувати модифікований параметричний алгоритм ідентифікації для побудови моделей дискретно-безперервних безінерційних процесів «трубчастого» типу;
- дослідити непараметричний алгоритм дуального керування багатомірним безінерційним об'єктом із запізнюванням;

При виконанні роботи мають використовуватися методи параметричної теорії ідентифікації, непараметричної теорії ідентифікації, теорії керування, теорії адаптивних та систем, що навчаються, математичної статистики:

- проаналізувати математичну модель, що заснована на модифікації параметричного алгоритму ідентифікації процесів, що мають «трубчасту»

структуру в просторі «вхідних-вихідних» змінних, що відрізняється від відомих параметричних уведенням індикаторних функцій, що приводить до одержання більш точних моделей дискретно-безперервних безінерційних процесів із запізнюванням;

– запропонувати модифікацію алгоритму дуального керування дискретно-безперервними процесами «трубчастого» типу. Особливість даних моделей полягає в тому, що при керуванні багатомірним об'єктом кожний компонент вектора керуючого впливу формується з урахуванням значень попередніх компонентів, що підвищує точність керування.

Це дозволить створити програмні моделі для створення системи керування із зовнішнім контуром, який є надбудовою стосовно діючої системи «об'єкт-регулятор», що забезпечує більш якісне ведення технологічного процесу. При цьому раніше діюча система керування залишається незмінною. Отримані моделі й алгоритми керування можуть знайти широке застосування при автоматизації процесів дискретно-безперервного типу, які домінують у чорній і кольоровій металургії, нафтопереробці, й т.ін.;

2 ОПИС ПРОВЕДЕНИХ ТЕОРЕТИЧНИХ ДОСЛІДЖЕНЬ

2.1 Аналіз «трубчастих» структур

Представлені докази асимптотичної незміщеності й збіжності в середньому для непараметричної оцінки функції регресії по спостереженням, при цьому використовується теорема про середній, яка дозволяє виключити вимогу диференціювання функції регресії.

На практиці досить часто зустрічаються процеси, що мають стохастичну залежність між компонентами вектора як вхідних, так і вихідних змінних. При цьому ні характер, ні структура залежності нам невідомі. Об'єкти, що мають подібну особливість, називають «трубчастими» [17].

Перше згадування про процеси «трубчастої» структури (Н-процесах) з'являється в [17]. Ці процеси були помічені при моделюванні технологічних процесів у металургії. Було виявлено, що компоненти вектора входу досліджуваного процесу зв'язані стохастичною залежністю, внаслідок чого процес протікає не у всій області, установленій технологічним регламентом підприємства, а лише в деякій його подобласті. У цих умовах процес моделювання й керування ускладнюється.

Розглянуто стохастичний багатомірний об'єкт, представлений на рис. 2.1.

Тут прийняті наступні позначення: A – невідомий оператор об'єкта, $x(t) \in \Omega(x) \subset R^1$ – вихідна змінна процесу, $u(t) = (u_i(t), i = 1, m) \in \Omega(u) R^m$ – векторний вхідний вплив, $\xi(t)$ – векторний випадковий вплив, (t) – безперервний час, $G^{ui}, i = \overline{1, m}, G^x$ канали зв'язку, що відповідають різним змін засобу, що включають у себе, контролю, $g^{u1}(t), i = \overline{1, m}, g^x(t)$ – випадкові перешкоди вимірів відповідних змінних процесу з нульовими математичними очікуваннями й обмеженою дисперсією.

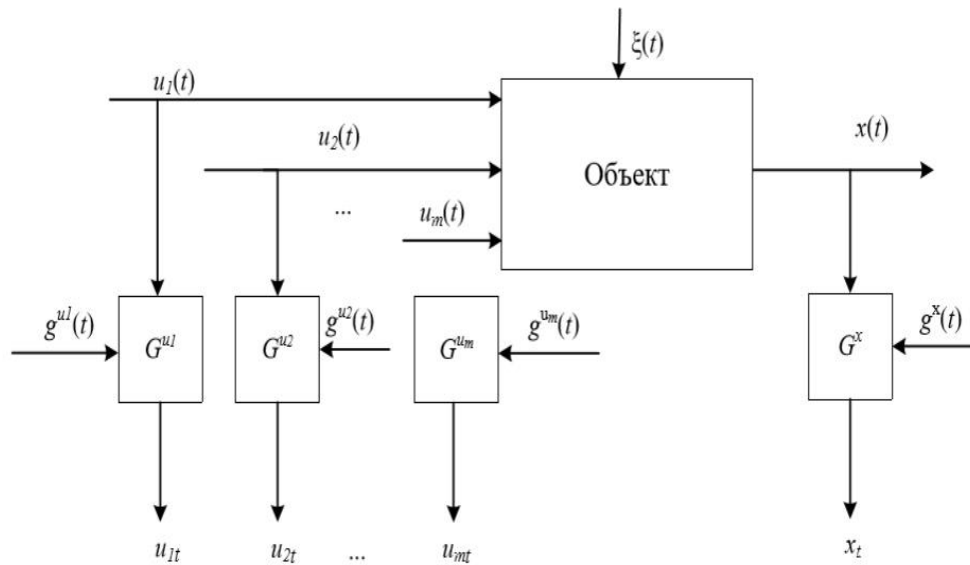


Рисунок 2.1 – Схема багатомірного стохастичного об'єкта

Вимірюючи значення вхідних і вихідних змінних через інтервал часу Δt , одержимо навчальну вибірку $\{u_i, x_i, i = \overline{1, s}\}$ де s – обсяг вибірки. Оборотною увагою на одну важливу обставину – компоненти вектора вхідних змінних $u(t)$ зв'язані стохастичною залежністю. Випадок, коли компоненти вектора входу проілюстрований рис.2.2:

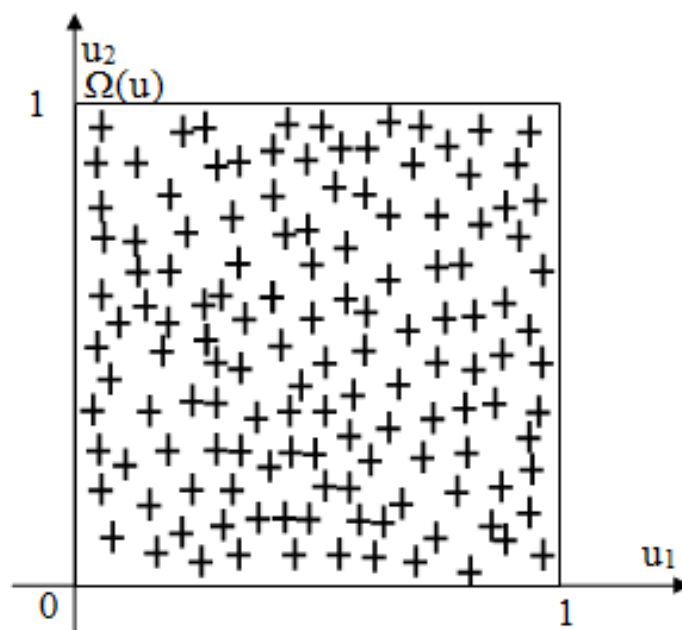


Рисунок 2.2 – Випадок незалежності компонентів вектора входу $u(t)$

Тут, з міркувань простоти, $u_1, u_2 \in [0;1]$. Знаком «+» позначені точки процесу. Як видно з рисунка, точки процесу можуть бути розташовані по всій площі даного одиничного квадрата. Одному значенню компонента $u_1(t)$ може відповідати безліч значень компоненти $u_2(t)$, і навпаки. Якщо між компонентами вектора входу існує деяка залежність, то в якості ілюстрації можна розглянути приклад, представлений на рис. 2.3, де компонента стохастичною залежністю. $u_1(t), u_2(t)$ зв'язані між собою

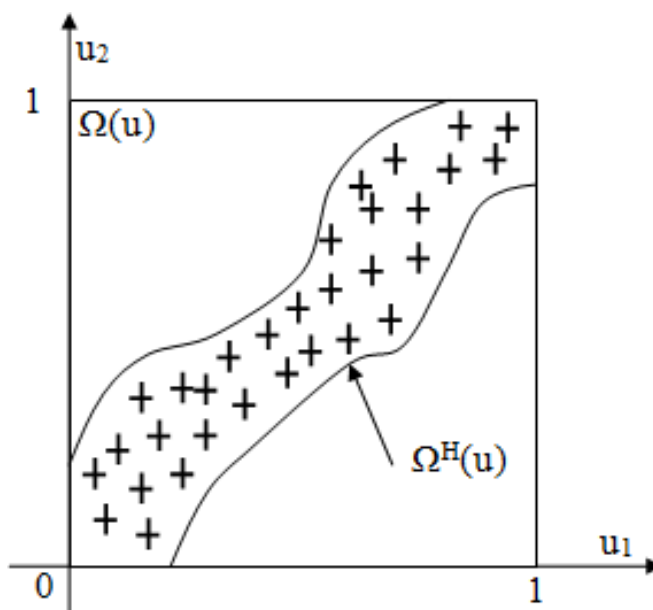


Рисунок 2.3 – Стохастичний зв'язок між вхідними змінними процесу

Як видно із представленого рисунка, точки процесу розташовуються не по усій площі одиничного квадрата $\Omega(u)$, а лише в його області $\Omega^H(u) \in \Omega(u)$. Одному певному значенню компонента u_1 відповідає тут уже значно менше число значень компоненти u_2 . Чим менше область $\Omega^H(u)$, тим більше однозначним стає відношення між компонентами u_1 і u_2 . Відзначимо, що області $\Omega^H(u)$ дослідникові завжди невідома, оскільки її розміри й положення визначаються видом зв'язку між змінними процесу, про яку немає певної інформації.

На рис. 2.4, на яким показаний процес «трубчастої» структури.

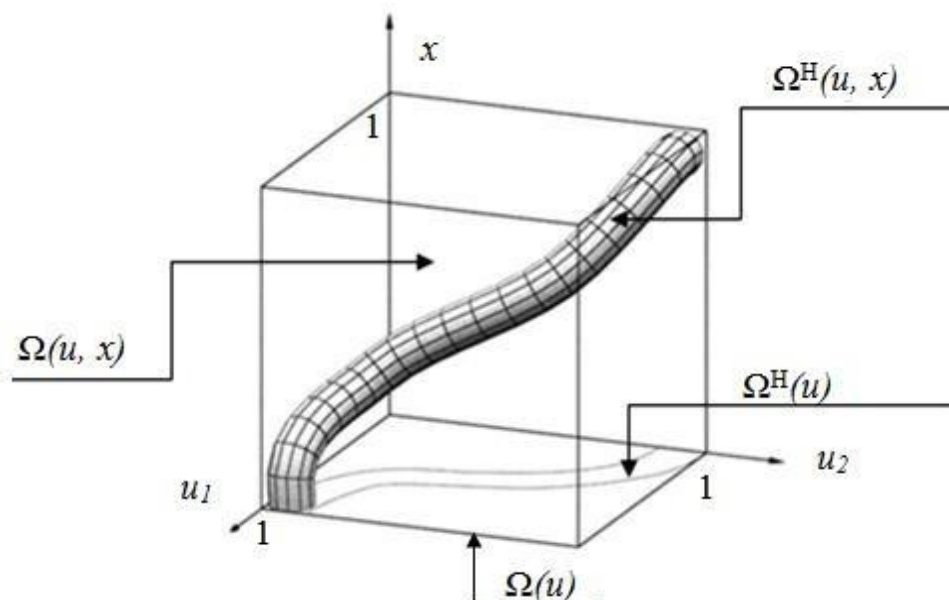


Рисунок 2.4 – Об'єкт із «трубчастою» структурою

Як видно з рис. 2.4, область протікання процесу $\Omega(u, x)$ представляє собою, без порушення спільності, одиничний гіперкуб, де $u = (u_1, u_2) \in R^2$, $x \in R^1$. Область протікання процесу $\Omega(u, x)$ на практиці відома завжди. Наприклад, для технологічного процесу значення «вхідних-вихідних» змінних визначаються рамками технологічного регламенту. Однак якщо досліджуваний процес має «трубчасту» структуру, тобто його вхідні змінні зв'язані між собою стохастичною залежністю, то область його протікання обмежується не тільки простором гіперкуба $\Omega(u, x)$, але й деякої його підобластю $\Omega^H(u, x) \in \Omega(u, x)$, яка нам ніколи не відома. Оскільки підобласть $\Omega^H(u, x)$ не відома, те й точно сказати про те, що досліджуваний об'єкт має дану особливість, не можна. У цьому полягає основна складність моделювання подібного роду процесів.

На практиці, при побудові моделі за наявним даними, використовують параметричні оцінки типу

$$X_\alpha(t) = A^\alpha(u(t), \alpha), \quad (2.1)$$

де A^α – параметрична структура моделі,

α – вектор параметрів.

При цьому для деяких значень вхідних змінних процесу $u \in \Omega(u) \subset R^m$, які одночасно задовольняють умові $u \notin \Omega^H(u)$, ми можемо одержати оцінки $x_\alpha \notin \Omega^H(u, x)$, або $x_\alpha \notin \Omega(u, x)$. Цей ефект продемонстрований на рис. 2.5.

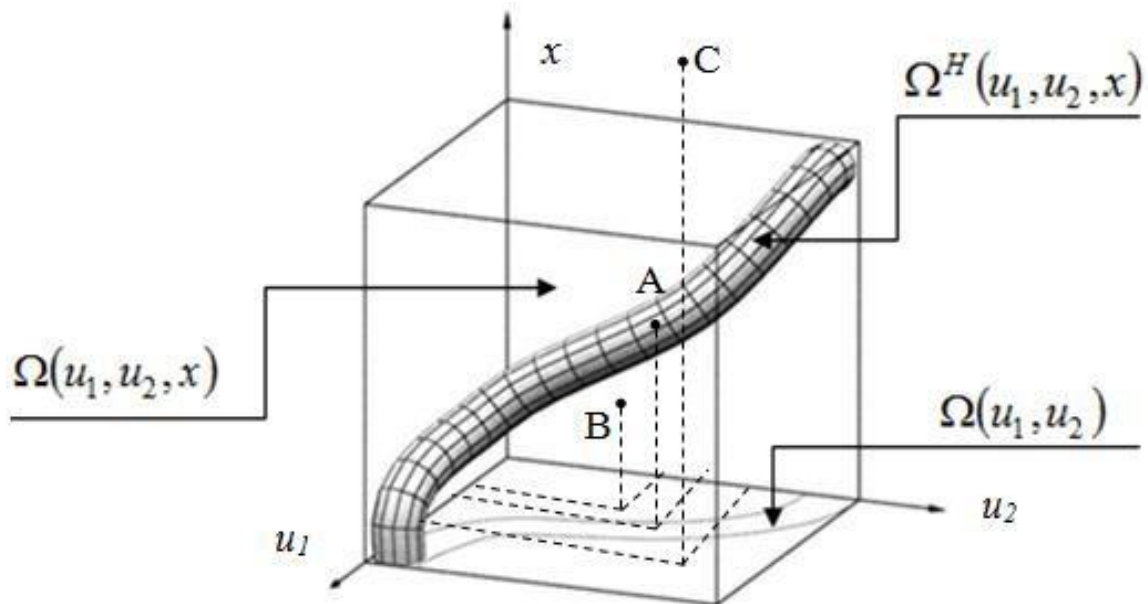


Рисунок 2.5 – Моделювання об'єкт із «трубчастою» структурою

Тут точки A , B , C – це різні значення вихідний змінної $x(t)$, отримані в результаті керування технологічним процесом. Як видно з малюнка, крапка A належить області «трубчастого» процесу, тобто $A \in \Omega^H(u, x)$, а крапка B не попадає в область «трубки», але перебуває в технологічному регламенті ($B \notin \Omega^H(u, x)$, но $B \in \Omega(u, x)$). Точка $C \notin \Omega(u, x)$, тобто вона виходить за рамки технологічного процесу. Якісний результат від ведення процесу можливий лише при одержанні точок, подібних точки A , тобто таких, які в підсумку потрапили в простір «трубки» $\Omega^H(u, x)$. Точки, що не потрапили в рамки технологічного регламенту, такі як крапка C , можна легко помітити й виключити з розгляду. Приміром, можна одержати негативні значення маси або долі речовини в розчині, що, природно, не є реальним.

Найбільш складний випадок представляють значення $x(t)$, які вписуються в обсяг гіперкуба $\Omega(u, x)$, але не належать простору «трубки» $\Omega^H(u, x)$ (наприклад,

точка B на рис. 2.5). Ці оцінки задовольняють умовам технологічного регламенту, однак, не має підстав для їхнього виключення, і враховується при керуванні. При цьому результат буде не задовільним.

2.2 Аналіз алгоритмів ідентифікації

Якщо необхідно відновити лінію у двовимірному просторі, то для якісного її відновлення буде потрібна вибірка з декількох десятків точок. Якщо робота проводиться в тривимірному просторі й об'єктом відновлення є площина, то мова йде про декілька сотень точок. Таким чином, зі збільшенням розмірності простору, обсяги вибірок s , необхідні для якісного оцінювання, збільшуються на порядки. Але на практиці не рідкі випадки, коли розмірність завдання k велика в порівнянні з обсягом наявної вибірки s . Наприклад, в практичних завданнях часто виникає ситуація, коли $k = 20 \div 30$, а $s = 900 \div 1000$. У цих ситуаціях математична статистика не може дати задовільне рішення завдання ідентифікації. Це означає, що отримані моделі будуть досить грубими.

У випадку наявності в об'єкта «трубчастої» структури компоненти його входу зв'язані стохастичною залежністю. Іншими словами, по суті, розмірність завдання в цьому випадку значно скорочується, і при малій кількості точок ми можемо одержати задовільні моделі. Розглянемо приклад. Нехай об'єкт описується рівнянням:

$$x(u) = f(u_1, u_2, u_3), \quad (2.2)$$

де $u = (u_1, u_2, u_3) \in R^3$ є вхідний змінна, а саме, тривимірний вектор,

$x \in R^1$ – вихідна змінна.

При параметричній ідентифікації досліджуваного процесу необхідно визначити клас параметричних залежностей $x^\alpha(u) = f^\alpha(u_1, u_2, u_3, \alpha)$ і оцінити тем

або іншим способом параметри α по вибірці спостережень (u_i, x_i) , $i = \overline{1, s}$, де s – обсяг вибірки.

Нехай компоненти вектора входу $u = (u_1, u_2, u_3)$ стохастично ніяк не зв'язані, тобто незалежні. У цьому випадку вихідна змінна $x(t)$ є функція від трьох змінних $u = (u_1, u_2, u_3)$. Припустимо, що компоненти вектора вхідних змінних зв'язані функціонально, наприклад

$$u_2 = \varphi_1(u_1), u_3 = \varphi_2(u_2) = \varphi_2(\varphi_1(u_1)), \quad (2.3)$$

Звичайно ж, дослідник не знає про наявність залежності (2.3). У зворотному випадку можна було б підстановкою і одержати:

$$x(u) = f(u_1, \varphi_1(u_1), \varphi_2(\varphi_1(u_1))). \quad (2.4)$$

Таким чином, тривимірна залежність (2.2) могла б бути зведена до одномірної залежності (2.4). Аналогічним образом залежність (2.2) може бути зведена й до двовимірної. Як ми бачимо, при наявності функціональної залежності між компонентами вектора входу $u = (u_1, u_2, u_3)$ розмірність розв'язуваного завдання скорочується. Аналогічно і при наявності не функціональної залежності, а стохастичною, але про наявність подібних залежностей дослідникові нічого не відомо.

2.3 Моделювання дискретно-безперервних процесів

При моделюванні дискретно-безперервних процесів традиційно прийнято використовувати параметричні алгоритми типу (2.1). Однак, при роботі із процесами «трубчастого» типу подібні алгоритми дають грубі, не точні результати. Розглянемо в якості ілюстрації наступний приклад.

Нехай досліджуваний процес описується наступною системою рівнянь:

$$\begin{cases} x = 0.5u_1 + 0.5u_2 + \psi, \\ u_2 = u_1 + \varphi \end{cases} \quad (2.5)$$

де $u_1, u_2 \in [0; 3]$, ξ и ψ – випадкові числа, розподілені по рівномірному закону в інтервалі $[-0.05; 0.05]$.

Процес у даному окремому випадку описується рівнянням прямої. Подібний опис процесу прийнятий з міркувань простоти. В обчислювальному експерименті рівняння об'єкта задане з метою одержання вибірок «вхідних-вихідних» змінних $\{x_i, u_{1i}, u_{2i}\}$, $i = 1, s$ для рішення завдання ідентифікації. При побудові моделі на основі отриманих вибірок, структура залежності вихідної змінної x від вхідних змінних u прийнята з точністю до параметрів. При оцінюванні параметрів використовується метод найменших квадратів (МНК). Параметрична модель досліджуваного об'єкта має вигляд:

$$x_\alpha = \alpha_1 u_1 + \alpha_2 u_2, \quad (2.6)$$

де α_1, α_2 параметри.

Згенеровано 3 статистично незалежні вибірки $\{x_i, u_{1i}, u_{2i}\}$, $i = \overline{1, s}$ обсягом $s = 100$. По кожній з вибірок оцінимо параметри моделі (2.6) з допомогою МНК. У результаті одержано три різні набори параметрів α_1, α_2 , які відповідають трьом різним моделям. Результати експерименту показано на рис. 2.6.

Отримані моделі не є адекватними – значення виходу моделі відновлюються в крапках, у яких процесу не існує. У даному прикладі був розглянутий окремий випадок, коли процес являє собою пряму. Відновлені моделі являють собою площини з різними оцінками параметрів. Відповідно через одну пряму ми можемо провести нескінченне число площин, при цьому кожна із цих площин буде вважатися моделлю даного процесу.

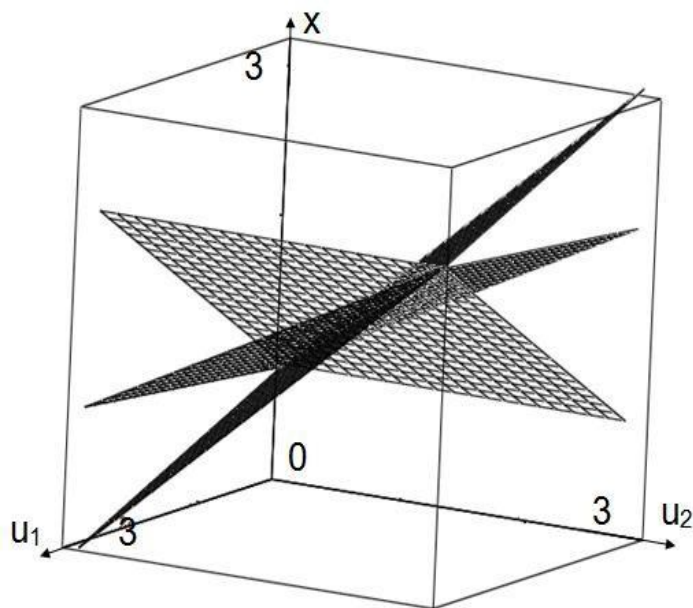


Рисунок 2.6 – Множина параметричних моделей процесу з «трубчастою» структурою

У тому випадку, коли компонента вектора вхідних змінних стохастично залежні, тобто отримано модель з «трубчастою» структурою процесу, необхідно ввести індикаторну функцію $I(u)$.

$$\hat{x}_\alpha(t) = I(u)A^\alpha(u(t), \alpha), \quad (2.7)$$

де індикатор $I(u)$ має вигляд:

$$I(u) = \begin{cases} 1, & \text{если } u \in \Omega^H(u); \\ 0, & \text{если } u \notin \Omega^H(u). \end{cases} \quad (2.8)$$

Область $\Omega^H(u)$ не відома, а відома лише вибірка $\{x_i, u_i\}, i = \overline{1, s}$. Якщо індикатор дорівнює нулю, то оцінка $\hat{x}_\alpha(u)$ не може бути обчислена, тобто при таких значеннях компонентів вектора $u \in \Omega(u)$ процес протікати не може. Якщо індикатор $I(u)$ при поданому на вхід значенні $u \in \Omega(u)$ дорівнює одиниці, то модель (2.3) збігається з (2.1).

Дійсний вид індикатору невідомий, тому потрібно використовувати його оцінку:

$$\widehat{x}_\alpha(t) = I_s(u)A^\alpha(u(t), \alpha_s), \quad (2.8)$$

де α_s - оцінки параметрів α .

Таким чином, при відомім значенні вхідний змінної $u = u'$ спочатку будується оцінка $x_s(u = u')$ по формулі (2.7), потім обчислюється оцінка індикатору $I_s(u)$, і тільки на наступному етапі використовується модель, якщо індикатор виявився рівним одиниці.

Якщо ж індикатор дорівнює нулю, то це означає, що хоча й $u \in \Omega(u)$, але $u \notin \Omega^H(u)$, тобто компоненти вектора $u = u' = (u'_1, \dots, u'_m)$ визначені не вірно, іншими словами, що реально протікає «трубчастий» процес не відповідає сукупності заданих значень компонент вектора $u = u'$. Причини цього можуть полягати в тому, що компоненти вектора $u = u' = (u'_1, \dots, u'_m)$ обрані невірно, або обмірювані зі значної погрішністю типу «викид». Це справедливо тільки за умови, якщо мається представницька вибірка $\{x_i, u_i\}, i = \overline{1, s}$. Використання традиційних моделей типу (2.1) дозволить одержати оцінку $\hat{x}_\alpha(u = u')$, яка, може бути далека від реальності.

3 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕНЬ

3.1 Опис алгоритмів ідентифікації «трубчастих» процесів

Таким чином, задача розробки та модифікації алгоритмів вивчення процесів зі стохастичною залежністю між компонентами вектора вхідних змінних (Н-процеси або процеси «трубчастого» типу) є актуальною та входить до класу задач, що вирішує інженерія програмного забезпечення. Процеси даного класу оточують повсюдно (медицина, хімія, промисловість, соціальні процеси й ін.), однак, на сьогоднішній день вони є маловивченими. Головна особливість Н-процесів полягає в тому, що вони протікають не у всій області визначення досліджуваного процесу Ω . У зв'язку з наявністю залежності між компонентами вектора входу Н-процес протікає лише в деякій підобласті $\Omega^H \in \Omega$, форма й обсяг якої дослідникові ніколи не відомі. При моделюванні об'єкта дослідження область протікання Н-процесу повинна підлягати визначенню.

До непараметричних методів ідентифікації, відноситься непараметрична оцінка функції регресії [19]. При роботі з подібного роду оцінками немає необхідності у використанні індикаторної функції (2.7), оскільки в них при відновленні точок процесу, використовується локальна апроксимація [4], [12]. Іншими словами, коли ми працюємо з методами параметричної ідентифікації, відновлення значення процесу в даній крапці відбувається з обліком усіх інших точок вибірки. При непараметричній ідентифікації для відновлення значення використовуються не всі точки вибірки, а лише ті, які перебувають в околиці відновлюваної точки процесу.

Питання, що стосується обсягу «трубчастого» процесу – обсяг області $\Omega(u, x)$ може бути легко знайдений, тому що значення змінних u і x заздалегідь визначені, наприклад, технологічним регламентом процесу. Обсяг області $\Omega^H(u, x)$ попередньо не відомий, оскільки про характер залежності між компонентами вектора входу u можна тільки догадуватися. Для обчислення обсягу

«трубчастого» процесу можна використовувати різні методи, у тому числі й метод Монте-Карло.

3.2 Обчислювальний експеримент

Нехай досліджуваний об'єкт описується залежністю виду:

$$\begin{cases} x = 2 \sin u_1 + u_2^2 + \psi, \\ u_2 = u_1 + \varphi; \end{cases} \quad (3.1)$$

де u_1, u_2 – компоненти вектора входу процесу, розподілені по рівномірному закону розподілу в інтервалі $[0; 3]$;

ξ і ψ – випадкові числа, розподілені за рівномірним законом в інтервалі $[-0.05; 0.05]$;

x – вихідна змінна процесу.

Компоненти вектора входу $u = (u_1, u_2)$ зв'язані між собою лінійною залежністю. Параметрична модель процесу (3.1) має бути прийнята в наступному виді:

$$x_\alpha = \alpha_1 \sin u_1 + \alpha_2 u_2^2, \quad (3.2)$$

де α_1, α_2 параметри моделі (3.1).

Параметрична модель процесу підібрана безпомилково, така можливість маємо тільки в рамках обчислювального експерименту, оскільки тут можна визначити характер процесу. Настроювання коефіцієнтів моделі (3.2) α_1, α_2 проводилася з допомогою МНК. Отримані результати, представлені на рис. 3.1.

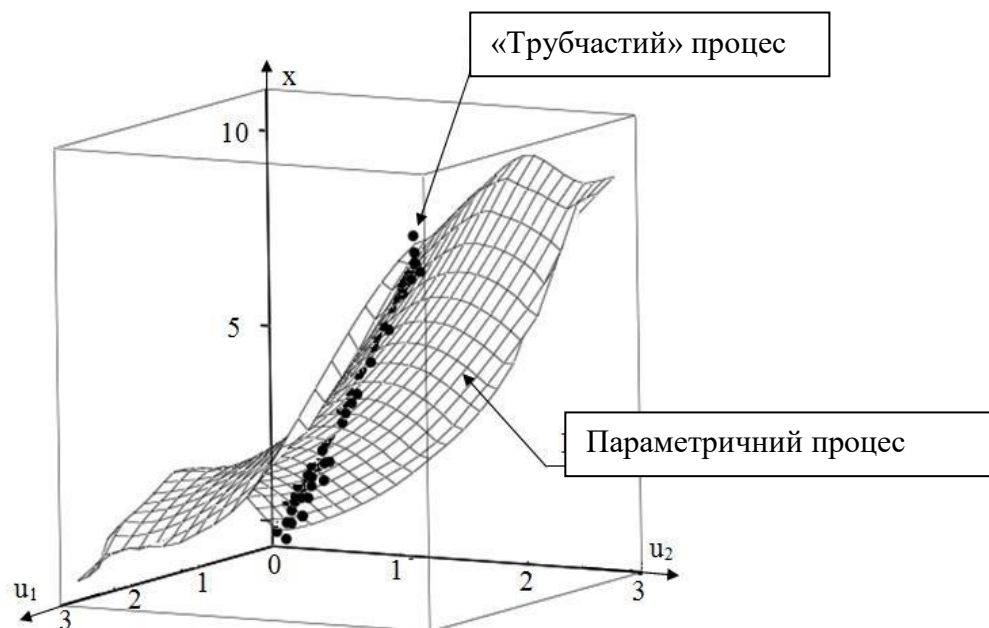


Рисунок 3.1 – Параметрична модель «трубчастого» процесу

Тут чорними крапками показані точки процесу (3.3). Як видно з рисунку, процес має «трубчастий» характер. Параметрична модель (3.2) являє собою поверхню, що лежить на крапках «трубчастого» процесу. Як видно з рисунку, отримана модель не є адекватною. Знайдено відносну помилку моделювання для даного випадку по наступній формулі:

$$E = \sqrt{\frac{1}{s} \sum_{i=1}^s (x_i - x_i^\alpha)^2 / \frac{1}{s-1} \sum_{i=1}^s (x_i - m_x)^2}, \quad (3.3)$$

Помилка моделювання $E = 3.87$. Це говорить про те, що отримана модель не є адекватною.

Модифікована модель (3.2), додавши до неї оцінку індикаторної функції:

$$x_\alpha = I_s(\alpha_1 \sin u_1 + \alpha_2 u_2^2). \quad (3.4)$$

У якості оцінки індикаторної функції I_s була обрана оцінка (3.2). Результати, отримані для моделі (3.3), представлені на рис. 3.2.

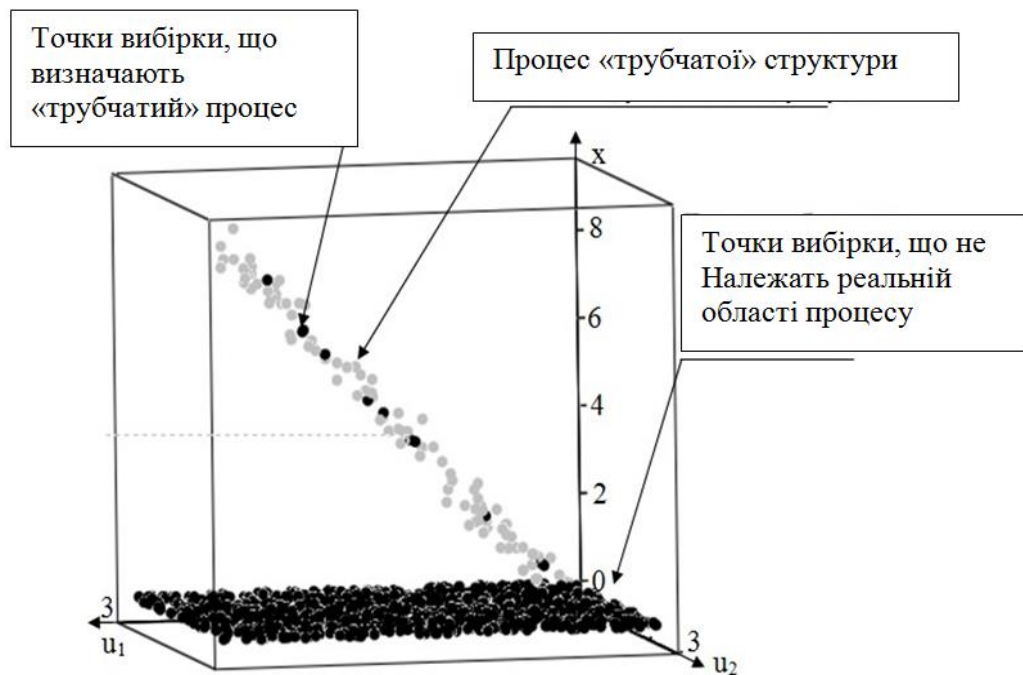


Рисунок 3.2 – Модифікована параметрична модель «тубчастого» процесу з використанням оцінки індикаторної функції

Параметрична модель являє собою поверхню, що проходить через «тубчастий» процес. Відносна помилка $E=3.73$. Використано для моделювання модифіковану параметричну модель (3.2).

У якості оцінки індикаторної функції I_s була обрана оцінка (3.4). Результати моделювання представлено на рис. 3.3.

Як видно з рис. 3.3, алгоритм залишив у моделі тільки ті точки, які визначають «тубчастий» процес. Відносна помилка моделювання склала $E=0.1$.

«Обсяг» процесу «тубчастої» структури – знаючи діапазони зміни «вхідних-вихідних» змінних процесу (наприклад, з технологічного регламенту підприємства), можна легко визначити «обсяг» області, у якій протікає досліджуваний процес. Цікавий з теоретичної й практичної точок зору питання про те, який «обсяг» від загальної області протікання займає процес «тубчастої» структури.



Рисунок 3.3 – Модифікована параметрична модель при нелінійній залежності компонентів вектора входу

У якості первісного наближення для розрахунку «обсягу» Н-процесу можна використовувати ідею методу Монте-Карло. Згідно наступною формулою:

$$V^H = \frac{q^H}{q} * 100\%, \quad (3.5)$$

де q – обсяг вибірки спостережень об'єкта дослідження,

qh – кількість точок, що визначають «трубчастий» процес у результаті моделювання.

Нехай досліджуваний процес описується системою рівнянь (3.3.5), розглянутої раніше. Тут $u_1, u_2 \in [0;3]$, ξ і ψ - випадкові числа, розподілені за рівномірним законом в інтервалі $[-0.05; 0.05]$.

Нехай обсяг вибірки рівний 1000, тобто $q=1000$. Побудуємо параметричну модель даного процесу. Тому що ми маємо справу з випадковими величинами, то проведемо усереднення за результатами 10 експериментів. У ході експериментів було отримано, що кількість точок, що потрапили в трубку $qh = 26$. Таким чином, «обсяг» процесу «трубчастої» структури займає 2,6% від загального «обсягу» процесу. Розглянуто результати аналогічних експериментів при різних значеннях q , представлені в таблиці 3.1.

Таблиця 3.1 – Опис структур, використуваних в обчислювальному експерименті

Змінна	Структура
x	$\sin(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12} + u_{13} + u_{14} + u_{15} + u_{16} + u_{17} + u_{18} + u_{19} + u_{20})$
u_1	$Random()$
u_2	$\sin u_1$
u_3	$u_1 u_2$
u_4	$0.3(u_1 + u_2 + u_3)$
u_5	$\sin(0.3(u_1 + u_2 + u_3 + u_4))$
u_6	$\sqrt{u_1 + u_2 + u_3 + u_4 + u_5 + u_6}$
u_7	$0.2(u_1 + u_2 + u_3 + u_4 + u_5 + u_6)$
u_8	$\sin(0.2(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7))$
u_9	$0.2(u_1 u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8)$
u_{10}	$0.2(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9)$
u_{11}	$2 \sin(0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10}))$
u_{12}	$2 \sin(0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11}))$
u_{13}	$2 \sin(0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12}))$
u_{14}	$2 \sin(0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12} + u_{13}))$
u_{15}	$0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} u_{11} + u_{12} + u_{13} + u_{14})$
u_{16}	$0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12} + u_{13} + u_{14} + u_{15} + u_{16})$
u_{17}	$\sqrt{0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12} + u_{13} + u_{14} + u_{15} + u_{16})}$
u_{18}	$\sqrt{0.1(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12} + u_{13} + u_{14} + u_{15} + u_{16} + u_{17})}$
u_{19}	$0.01(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12} + u_{13} + u_{14} + u_{15} + u_{16} + u_{17} + u_{18})$
u_{20}	$0.01(u_1 + u_2 + u_3 + u_4 + u_5 + u_6 + u_7 + u_8 + u_9 + u_{10} + u_{11} + u_{12} + u_{13} + u_{14} + u_{15} + u_{16} + u_{17})$

Результат чисельного моделювання представлено на рис. 3.4.

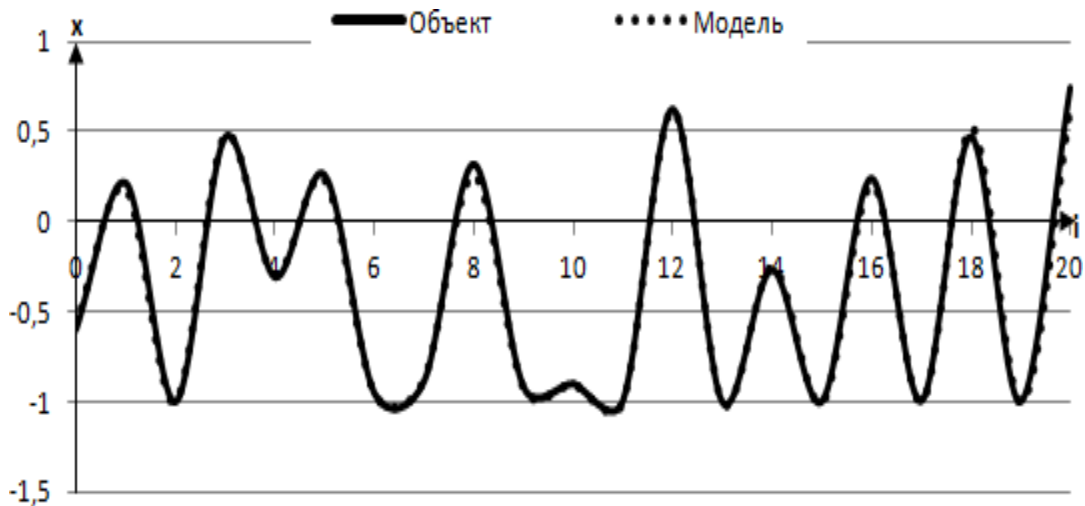


Рисунок 3.4 – Результат обчислювального експерименту в умовах без перешкод

При цьому відносна помилка моделювання склала $E=0.02$, оптимальне значення $cs=0.12$. Як видно, модель достатньо точна.

Додавши до виходу об'єкта x перешкоду 5%, то отримані результати графічно представлено на рис. 3.5.

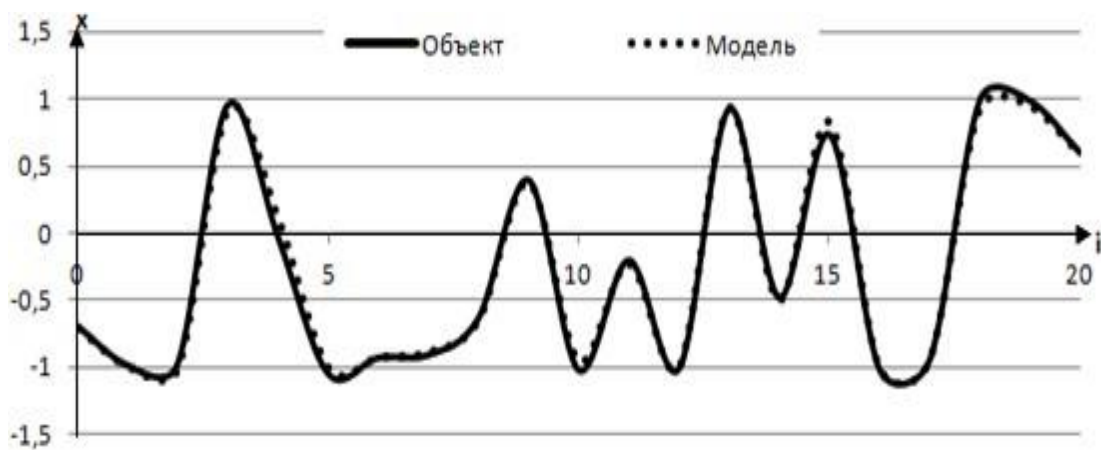


Рисунок 3.5 – Результат обчислювального експерименту при перешкоді в 5%

Відносна помилка моделювання рівна $E=0.09$, $cs=0.14$. Перешкода впливає на якість моделювання, але помилка, проте, не значна.

3.3 Оцінка похибок моделювання

Моделювання багатофакторних об'єктів в умовах малих обсягів вибірок дає задовільні результати лише за умови функціональної залежності вхідних змінних процесу. У зворотному випадку завдання або погано реалізована, або нереалізована зовсім.

Нехай всі компоненти вектору вхід являють собою незалежні величин, згенеровані за допомогою генератор випадкових чисел мовою C#. Розглянуто, як залежить відносна помилка моделювання від кількості функціонально залежних змінних вихідної вибірки. Обчислювальний експеримент складався з 20 етапів. На першому етапі всі вхідні змінні процесу були функціонально незалежними. Потім, на кожному наступному етапі, ми по черзі додавали функціонально залежну змінну. У підсумку, на останньому етапі, усі вхідні змінні, крім однієї, виявилися функціонально залежними. Результати експерименту ілюструється на нижченаведеному рис. 3.6.

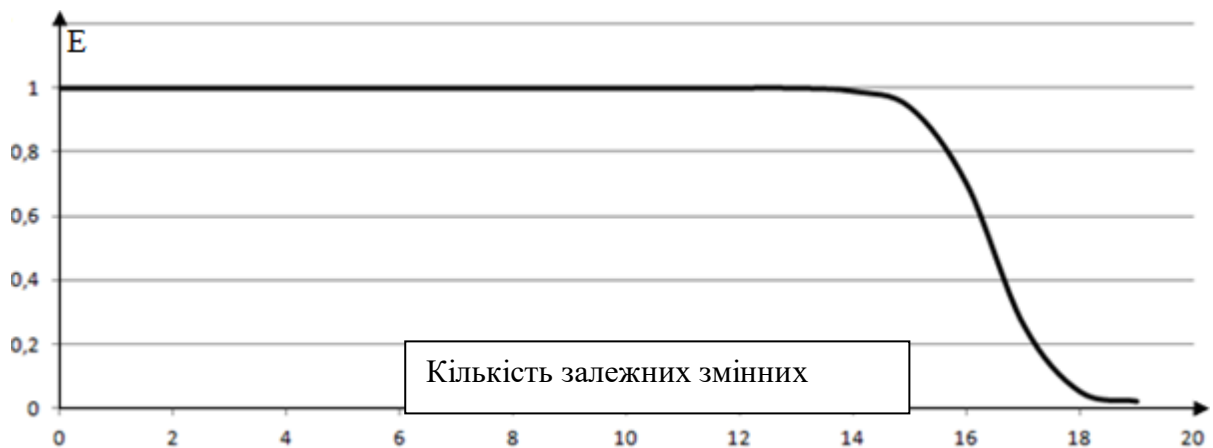


Рисунок 3.6 – Графік залежності відносної помилки моделювання від числа незалежних компонентів вектора входу

У роботі показано, що традиційні методи параметричної ідентифікації не прийнятні при роботі з N-процесами. У цьому зв'язку запропонована модифікація параметричного алгоритму ідентифікації з використанням індикаторної функції, яка визначає область протікання N-процесу. Запропоновано два види індикаторної функцій. Обчислювальні експерименти показали, що завдання

ідентифікації Н-процесів у цьому випадку вирішується досить точно. Методи непараметричної ідентифікації при роботі з Н-процесами в модифікації не мають потреби. Це обумовлене тим, що непараметричні методи ставляться до класу локальних апроксимацій.

Аналізується питання про те, який «обсяг» від загальної області протікання процесу займає процес «трубчастої» структури. У якості первісного наближення в роботі використовується ідея методу Монте-Карло.

Розглядається завдання моделювання процесу порівняно великої розмірності при наявності малого обсягу вибірки. Обчислювальні експерименти показали, що моделі Н-процесів дозволяють давати гарні прогнози при великій розмірності завдання, але при малому обсязі вихідних даних.

3.4 Алгоритми керування макрооб'єктами

Керування макрооб'єктом. У теперішні час, при керуванні дискретними безперервними процесами в різних галузях промисловості, часто використовують стандартні типові закони регулювання (П, ПІ, ПІД регулятори). Зокрема, вони ефективно використовуються при керуванні добре контрольованими технічними об'єктами, наприклад, плавильними печами, реакторами, турбінами й ін. Зовсім ясно, що якість регулювання залежить від налаштувань відповідних регуляторів, і в ряді випадків виявляється недостатньо ефективними. Відзначимо ще одна обставину – типові регулятори, використовувані в промисловості, не є навчальними (адаптивними). Це значить, що при повторному переключенні об'єкта з одного стану в інше, регулятор функціонує в такий же спосіб, як і раніше. Схема керування представлена на рис. 3.7.

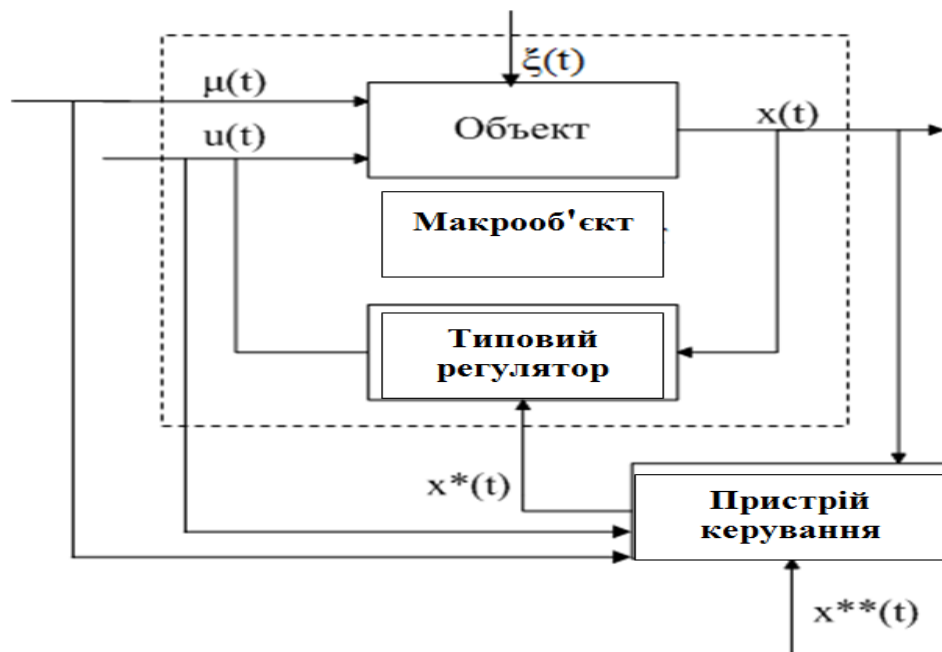


Рисунок 3.7 – Схема керування макрооб'єктом

У вищеписаній схемі вбудований у систему керування типовий регулятор збережений, але доданий зовнішній контур керування, що несе в собі риси навченості й дуалізму. Для нього об'єкт і регулятор є своєрідним макрооб'єктом. Тут x^{**} вплив, що задає, надходить на керуючий пристрій, який, у свою чергу, виробляє, що задає вплив x^* для типового регулятора.

4 ОПИС ФУНКЦІОНУВАННЯ РОЗРОБЛЕНОЇ ПРОГРАМНОЇ СИСТЕМИ

4.1 Обчислювальні експерименти

Перевірка працездатності викладеного алгоритму непараметричного дуального керування здійснювалася методами статистичного моделювання. Для обчислювального експерименту був обраний об'єкт, що описується рівнянням виду:

$$x(t) = 2\sqrt{u(t)} + \mu(t). \quad (4.1)$$

Дана залежність відома лише в рамках обчислювального експерименту. Вхідна керована змінна $u(t)$ генерувалася за рівномірним законом розподілу в заданих інтервалах $u(t) \in [0;3]$. Некерована, але контрольована змінна $\mu(t)$ описується (у рамках обчислювального експерименту) наступною залежністю:

$$\mu(t) = \sin(0.015i). \quad (4.2)$$

Графік, що ілюструє характеристику об'єкта (4.1), представлено на рис. 4.1.

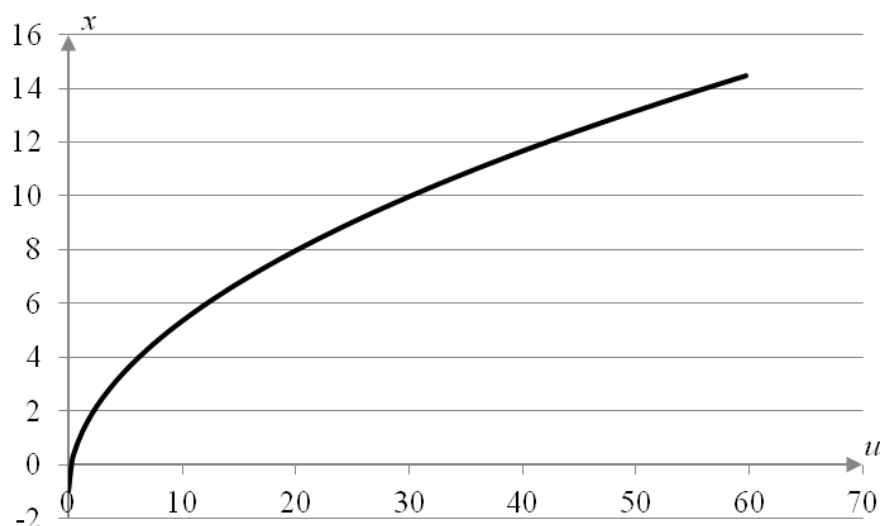


Рисунок 4.1 – Характеристика досліджуваного об'єкта

Як видно з рисунка, характеристика об'єкта взаємно однозначна. Розроблене програмне застосування реалізує вище вказані алгоритми.

Результати обчислювальних експериментів. вплив, що задається, є константою: $x^*(t) = 4$.

Параметри алгоритму містять наступні значення: $\alpha = 2.8$, $\beta = 1.1$, $\theta = 0.6$, $u_0 = 4$. Отримані результати представлено на рис. 4.2.

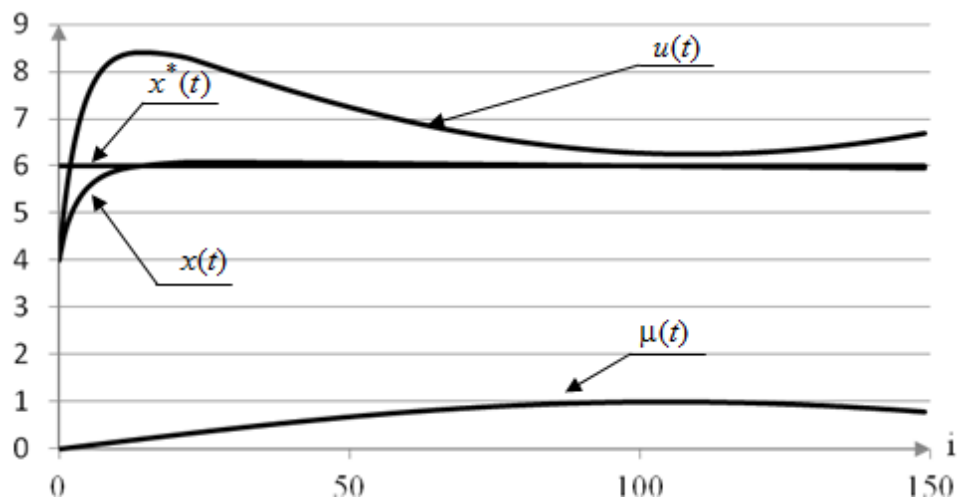


Рисунок 4.2 – Вплив, що задає, є константа

Програмний засіб моделювання ілюструє, як поводить ся алгоритм. На перших кроках алгоритм не «навчений». І тут основну роль відіграє пошукова складова Δu_{s+1} алгоритму, яка «тягне» вихід об'єкта $x(t)$ до завдання $x^*(t)$.

По ходу свого функціонування алгоритм накопичує «знання» про об'єкт. Роль пошукової складової знижується, і основний внесок в зроблений алгоритм вносить компонента u_s^* .

Якість роботи адаптивного алгоритму керування багато в чому залежить від характеру зміни вхідних змінних процесу й складності опису керованого об'єкта. Якщо змінні процесу мають просту (близьку до лінійної) характеристику, то алгоритм перебуває в «сприятливих» умовах, і процес його «навчання» вимагає меншого часу. І навпаки, якщо поведінка вхідних змінних процесу описується більш складними, нелінійними залежностями, то процес «навчання» може бути досить затягнутим.

Неконтрольована змінна $\mu(t)$ мала плавно мінливий характер.

Зміна рівняння неконтрольованої змінної $\mu(t)$, збільшує частоту коливань, зробивши тем самим її характеристику більш складної.

4.2 Розрахунок варіантів функціонування

Керуючі пристрій є зовнішнім контуром керування, який дозволяє поліпшити якість керування об'єктом. При цьому заміна типових регуляторів, які перебувають у виробництві (П, ПІ, ПІД), не передбачається. Це пов'язане з тим, що стандартні регулятори – аналогові, і часто більш надійні в роботі, чому цифрові, які більшою мірою піддані впливу зовнішнього середовища (різкий перепад температур, вібрації, агресивного середовища, впливу різного роду випромінюванням).

Навчання керуванню починається з першої тріади спостережень x_1, u_1, μ_1 . На початкової стадії керування необхідно якийсь час (нагромадження вибірки) для приведення об'єкта в заданий стан. Із графіків видно, що двоконтурна система керування приводить об'єкт до заданого стану швидше, чим П-регулятор.

Розглянуто випадок, представлений на рис. 4.3, коли вхідний вимірюваний некерований вплив має вигляд траєкторії $\mu(t) = 1.25 + \sin(0.5t)$. При цьому об'єкт описується рівнянням:

$$x(t) = 2u^2(t) + 1.5u(t) + \mu(t) + g^x(t). \quad (4.3)$$

Як можна помітити з рис. 4.3, непараметричний алгоритм дуального керування, при частішанні коливань вхідного контрольованого впливу $\mu(t)$, справляється із завданням керування краще, ніж П-регулятор, який у свою чергу видає керування поганої якості.

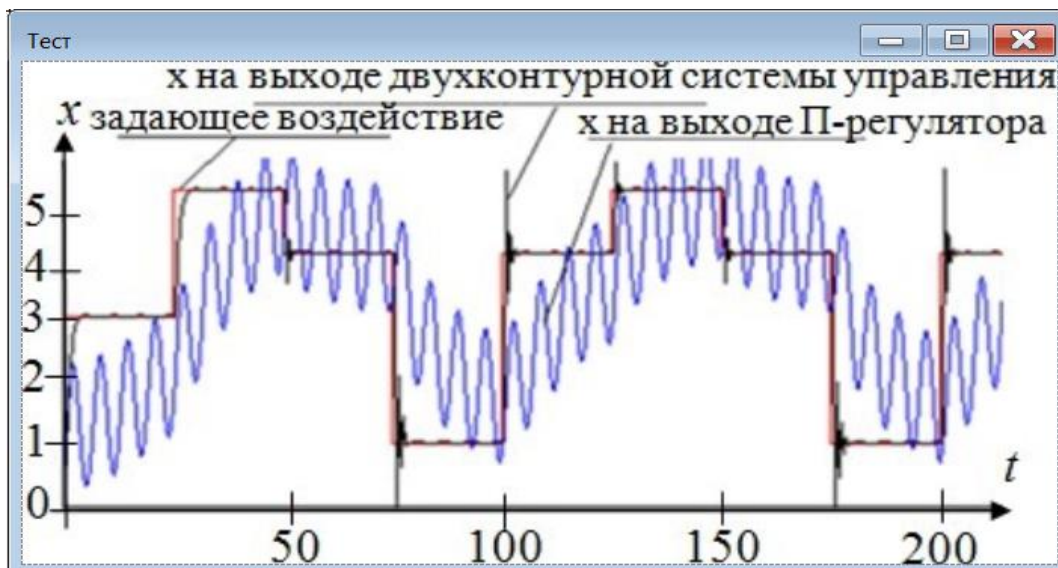


Рисунок 4.3 – Результат роботи системи при збільшенні частоти коливань невідомого вхідного впливу $\mu(t)$, що задають вплив – східчасте, перешкода відсутня

Представимо, що задають вплив, починаючи з 50-ої ітерації, у вигляді траєкторії, що має вид:

$$x^*(t) = 4 + 2\sin(0.1t), \quad (4.4)$$

Результат даного експерименту представлено на рис. 4.4:

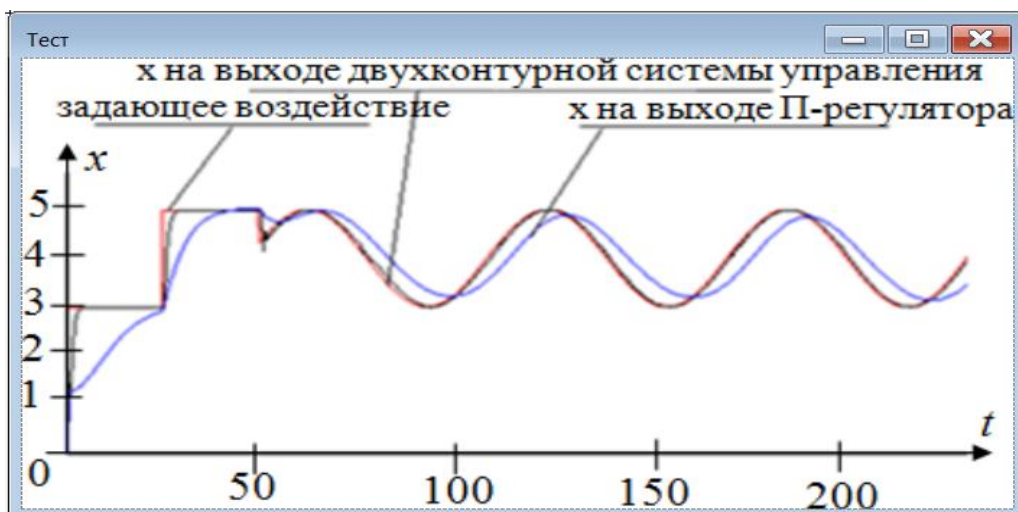


Рисунок 4.4 – Результат роботи системи при впливі, що задає, у вигляді траєкторії

П-регулятор, «не встигає» підлаштовуватися під x^* , внаслідок того, що він має вигляд траєкторії, але як тільки додається контур, що містить адаптивний

регулятор, то після «навчання», що відбувається на перших тактах, об'єкт миттєво переходить до завдання.

Розглянуто випадок, коли x^* починаючи з 20-ой ітерації задається випадковим чином (рис. 4.5). Тут представлено випадок, коли двоконтурна система керування, явно перевершує схему, що включає один П-регулятор.

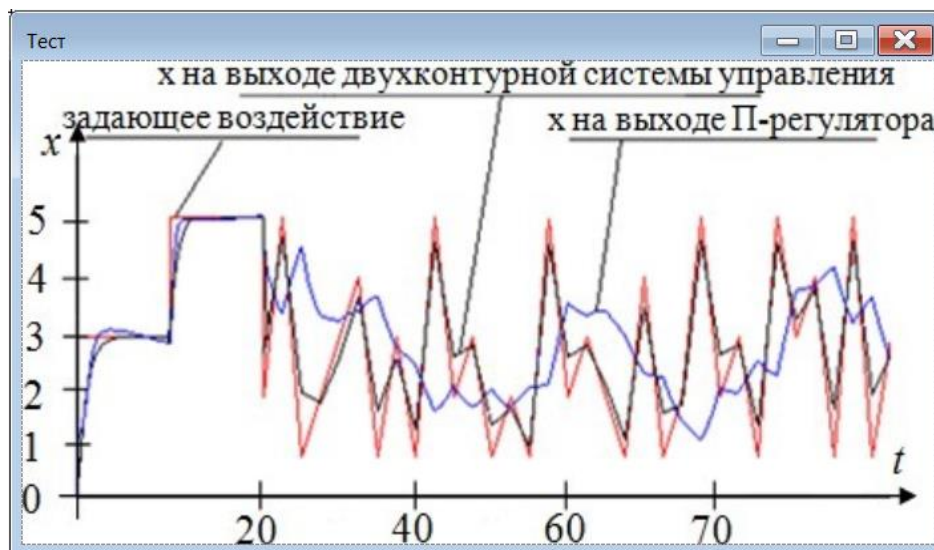


Рисунок 4.5 – Результат роботи системи при випадковій задачі

У даному прикладі x^* – задається випадково, і стандартні алгоритми регулювання не встигають підлаштуватися під завдання, у той час як адаптивний алгоритм, «запам'ятавши», як досягнути потрібний результат, справляється із завданням керування.

$$\mu(t) = \sin(0.04i), \quad (4.5)$$

інші параметри алгоритму залишимо без зміни. Результати обчислювального експерименту представлено на рис. 4.6.

Як видно з вищенаведеного рисунка, збільшення частоти характеристики $\Delta(t)$ призвело до збільшення часу «навчання» алгоритму.

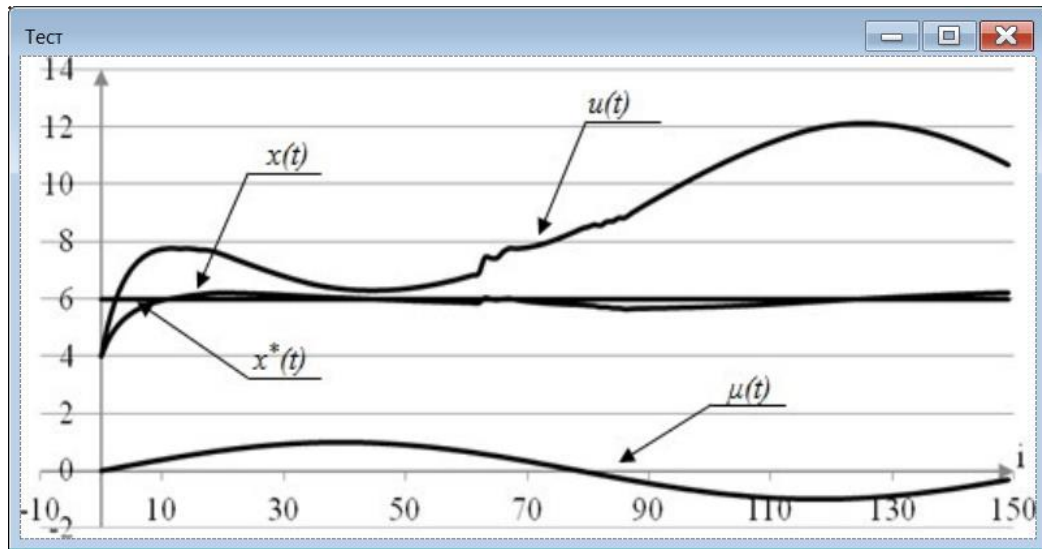


Рисунок 4.6 – Результати роботи алгоритму при збільшенні частоти зміни некерованої вхідної змінної $\mu(t)$

По порівнянню з результатом на рис. 4.5, процес «навчання» затягнутий. Розглянуто, як на роботу алгоритму впливає коефіцієнт пошукового кроку γ . вплив, що нехай задає $x^*(t)$ являє собою східчасте вплив і описується наступною системою рівнянь:

$$\begin{cases} x^*(t) = 5, & \text{якщо } 0 < i \leq 50; \\ x^*(t) = 2, & \text{якщо } 50 < i \leq 100; \\ x^*(t) = 6, & \text{якщо } 100 < i \leq 150; \\ x^*(t) = 3.5, & \text{якщо } 150 < i \leq 200. \end{cases} \quad (4.6)$$

Результати моделювання, при параметрах алгоритму $\alpha = 2.8$, $\beta = 1.1$, $\gamma = 0.6$, $u_0 = 4$. Алгоритм призводить вихід об'єкта $x(t)$ до бажаного значення $x^*(t)$ через деякий час «пошуку». Це відбувається щораз, коли завдання змінюється. Час «пошуку» суттєво залежить від коефіцієнта γ при пошуковому кроці Δu . Наприклад, $\gamma = 0.9$. Результати моделювання представлено на рис. 4.7.

При порівнянні результатів, показаних на рисунках 4.7 і 4.8, можна зробити висновок, «навчання» відбувається швидше. Алгоритм за менший час приводить вихід об'єкта $x(t)$ до бажаного значення $x^*(t)$. Як було відзначено раніше, на початковій стадії роботи алгоритму більша роль належить пошуковому кроку Δu .

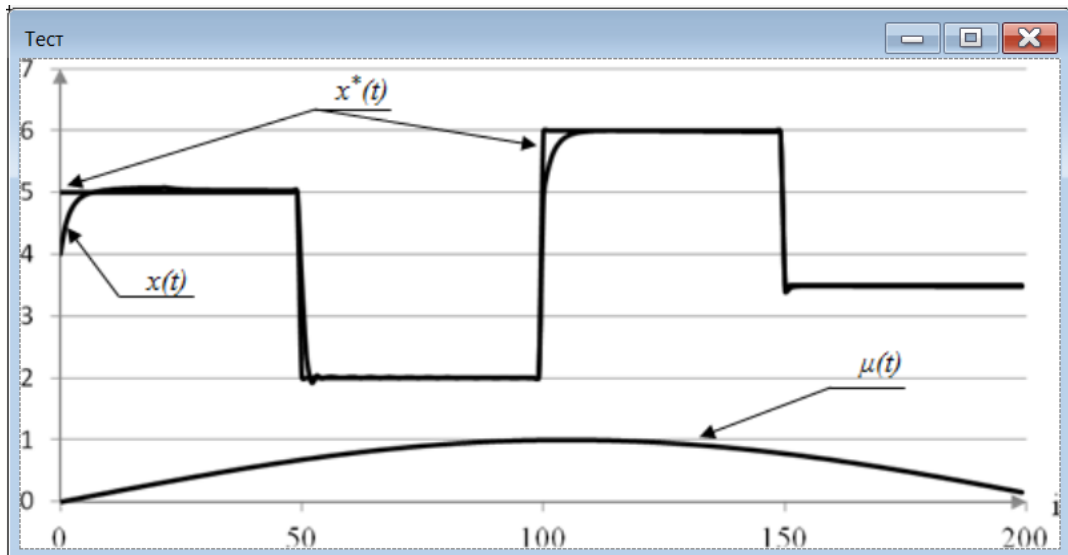


Рисунок 4.7 – Результати роботи алгоритму при пошуковому кроці $\gamma=0.9$

Для завданого впливу, це показано на рис. 4.8.

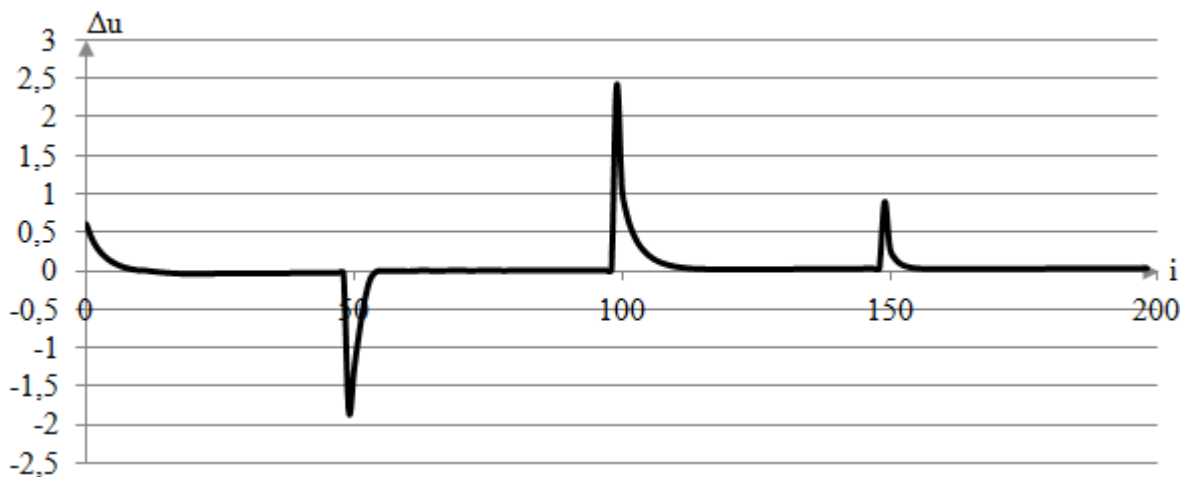


Рисунок 4.8 – Поведінка пошукового кроку Δu

З рис. 4.8 видно, що величина пошукового кроку Δu при зміні завдання $x^*(t)$ різко зростає. Це пов'язане з тим, що алгоритм попадає в невідомі йому умови. Але після деякого часу «навчання» значення компоненти Δu зменшується.

Розглянуто роботу алгоритму в умовах, коли на вихід об'єкта накладається перешкода. На рис. 4.9 показаний результат обчислювального експерименту, коли перешкода, прикладена до виходу, склала 5% (коефіцієнт $k=0.05$).

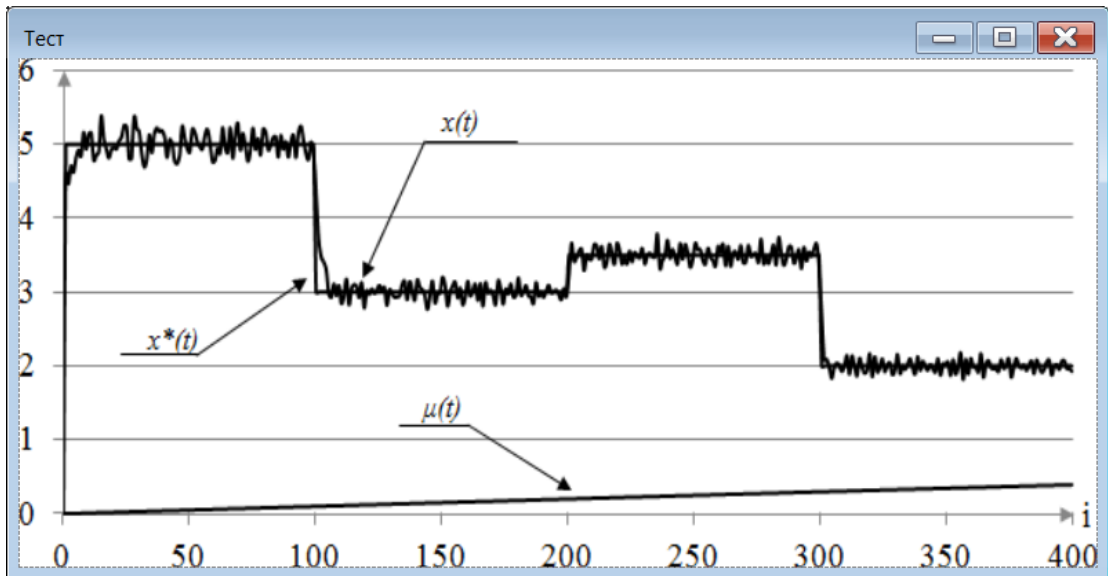


Рисунок 4.9 – Результати обчислювального експерименту при перешкоді в 5%

З рис. 4.9 видно, що перешкода впливає на значення вихідної змінної $x(t)$, але, незважаючи на це, алгоритм керування приводить об'єкт до бажаного значення $x^*(t)$. Розглянуто випадок, коли завдання $x^*(t)$ описується наступною системою:

$$\begin{cases} x^* = 5, & \text{якщо } 0 < i \leq 30; \\ x^* = 2, & \text{якщо } 30 < i \leq 60; \\ x^* = 5 + \cos(0.03i), & \text{якщо } 60 < i \leq 120; \\ x^* = \text{rand}(2; 3.5), & \text{якщо } 120 < i \leq 150. \end{cases} \quad (4.7)$$

Тут $\text{rand}(2; 3.5)$ – нормально розподілена випадкова величина в інтервалі $[2; 3.5]$. Параметри алгоритму: $\alpha = 2.8$, $\beta = 1.1$, $\gamma = 0.6$, $u_0 = 4$. Результати обчислювального експерименту представлено на рис. 4.10.

На рис. 4.10 показано, як поводить алгоритм при східчастім завданні, при завданні траєкторії й при випадковім завданні.

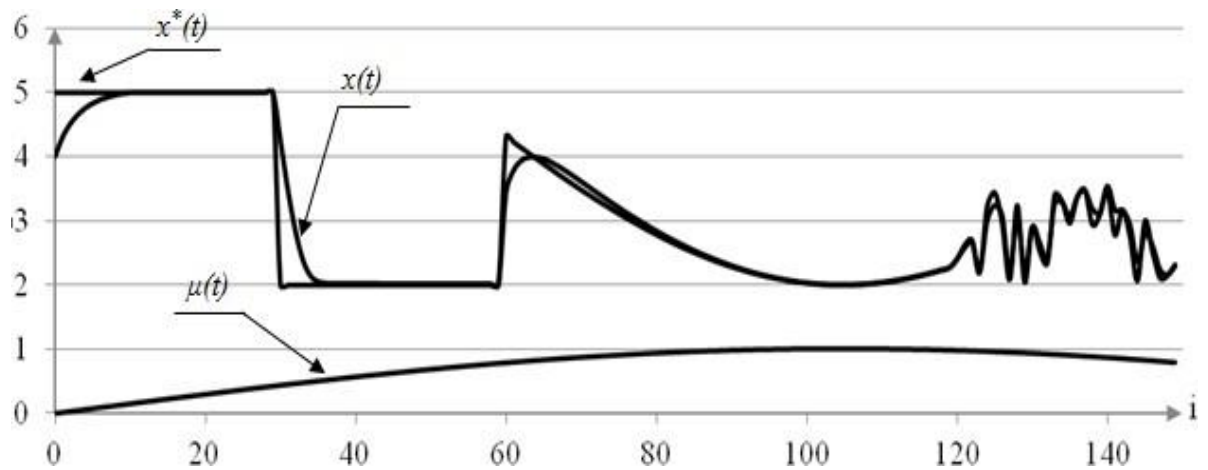


Рисунок 4.10 – Результати роботи алгоритму при різних впливах, що задають $x^*(t)$

З рис. 4.10 видна гарна якість керування навіть при такому варіанті, коли завдання носить випадковий характер.

З подібним завданням не впорається жоден з відомих регуляторів. На практиці такий варіант впливу, що задає, не зустрічається, однак, це становить інтерес із теоретичної точки зору.

5 ОПИС МОЖЛИВОСТІ ВИКОРИСТАННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Розглянуто в якості результату роботи програмного забезпечення наведено результати ще одного обчислювального експерименту. Нехай досліджуваний об'єкт керування описується залежністю виду:

$$x(t) = 4u(t) + g^x(t). \quad (5.1)$$

Порівняльні результати роботи П-регулятора й непараметричного регулятора – вплив, що заданий, має вигляд сходів. Коефіцієнт П-регулятора $K_p=0.1$.

Коефіцієнти непараметричного алгоритму керування $\gamma = 0.05$, $\alpha = 1.1$. Результати обчислювального експерименту представлено на рис. 5.1.

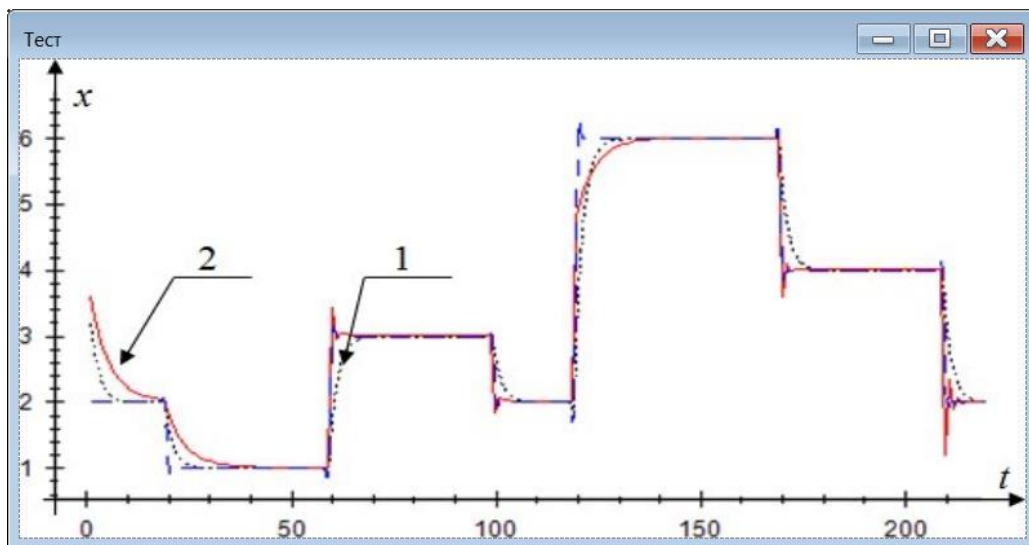


Рисунок 5.1 – Порівняння роботи П-регулятора й непараметричного регулятора

На рис. 5.1 графік «1» показує вихід $x(t)$ після П-регулятора, графік «2» – вихід $x(t)$ після непараметричного регулятора.

На перших тактах непараметричний регулятор збирає інформацію про об'єкт, і швидкість регулювання уступає швидкості регулювання в П-регулятора.

Однак після нагромадження інформації непараметричний регулятор дає практично миттєві результати.

Якщо досліджувані процес є «трубчастим» і описується системою

$$\begin{cases} x = 2 \sin u_1 + u_2^2 + \psi, \\ u_2 = u_1 + \varphi; \end{cases} \quad (5.2)$$

В системі завдано такі параметри u_1, u_2 – компоненти вектора входу процесу, розподілені по рівномірному закону розподілу в інтервалі $[0; 3]$; ξ і ψ – випадкові числа, розподілені за рівномірним законом в інтервалі $[-0.05; 0.05]$; x – вихідна змінна процесу.

Компоненти вектора входу $u = (u_1, u_2)$ зв'язані між собою лінійною залежністю.

Програмна система показує, як непараметричний регулятор впорається з керуванням Н- процесом (рис. 5.2).

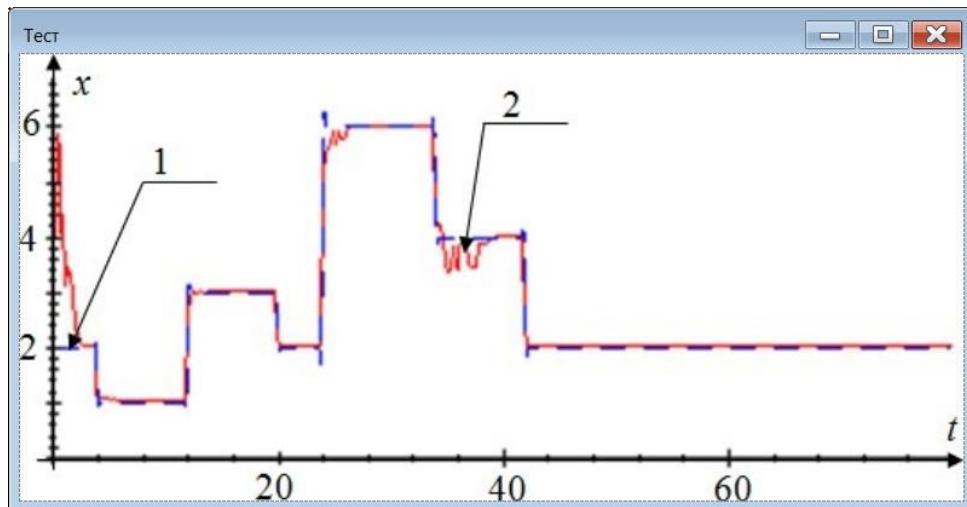


Рисунок 5.2 – Непараметричне керування процесом «трубчастого» типу

На рис. 5.2 графік «1» – вплив, що задає, обране сходишкою, графік «2» – результат непараметричного керування. З екранної форми видно, що непараметричний регулятор досить добре справляється з поставленим завданням. В вирішенні проблеми дуального керування багатомірними безінерційним процесами в умовах непараметричної невизначеності приводяться короткі

відомості з теорії дуального керування в байесовій, параметричній й непараметричній постановках. Слід зазначити, що в байесовій теорії дуального керування, процес одержання оптимальної стратегії керування виявляється досить складним, зате найбільш простим і коротким є власне процес навчання. У системах параметричного дуального керування, процес синтезу алгоритмів керування виявляється більш простим, але потрібен більш тривалий час для навчання системи керування. Нарешті, у системах непараметричного дуального керування процедура побудови системи ще більш проста, але час для навчання системи потрібен ще більший. Тут усе залежить від того, який рівень апріорної інформації був «вкладений» при формулюванні завдання.

При наявності великого обсягу апріорної інформації, як при байесовій постановці завдання, слід очікувати, що процес навчання не буде затягнутим у часі. Навпаки, якщо на початковій стадії формулювання завдання обсяг інформації не великий, то слід очікувати, що процес навчання може виявитися більш тривалим.

Запропоновано непараметричні алгоритми дуального керування, що представляють собою послідовність по-компонентного обчислення керуючих впливів при вимірюваних некерованих вхідних змінних. Розглядається випадок введення в систему керування зовнішнього контуру, який дозволяє поліпшити якість керування об'єктом.

ВИСНОВКИ

У ході виконання атестаційної роботи магістра розглянуті завдання ідентифікації й керування дискретно-безперервними безінерційними процесами «трубчастого» типу із запізнюванням в умовах параметричної й непараметричної невизначеності. Важливість побудови моделей і алгоритмів керування в умовах непараметричної невизначеності, доведено тим, оскільки вони в багатьох випадках є більш адекватними різноманітним завданням практики.

Розглянуте завдання аналізу даних, що містять у собі пропуски й викиди. Запропонована непараметрична методика відновлення пропусків «вхідних-вихідних» змінних матриці спостережень, заснована на непараметричній оцінці функції регресії за спостереженнями.

Результати обчислювальних експериментів показали, що завдання ідентифікації по заповненій за допомогою запропонованої методики матриці вирішується більш точно, ніж по вихідній матриці спостережень із пропусками.

Запропонована методика виключення викидів з вихідної вибірки спостережень. Дана методика дозволяє виключити викид з вибірки спостережень цілком, а не просто згладити його вплив на результати моделювання, як це роблять робастні алгоритми. При виключенні викиду з вибірки спостережень точність рішення завдання ідентифікації, як показують обчислювальні експерименти, підвищується.

У роботі розглядається проблема дуального керування багатомірними системами в умовах непараметричної невизначеності. Проведено порівняльний аналіз завдань дуального керування в байесовій, параметричній і непараметричній постановках.

Пропонується модифікований алгоритм непараметричного дуального керування, що продемонстрував свою ефективність у ряді обчислювальних експериментів. Особливістю непараметричних алгоритмів дуального керування є те, що при керуванні багатомірним об'єктом кожний компонент вектора

керуючого впливу формується з урахуванням значень попередніх компонентів, що значно підвищує точність керування.

Алгоритм використовує схему керування процесом, що включає в себе зовнішній контур керування, який, безумовно, дозволить більш якісно вести технологічний процес.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Глушков В.М., Цейтлин Г.Е., Ющенко Е.Л. Алгебра. Языки. Программирование. 3-е изд. – К.: Наукова думка, 1989. – 376 с.
2. Горин А.К. Автоматизация проектирования структур данных АИС // Кибернетика. – 1989. – №6. – С. 114–117.
3. Дал У., Дейкстра Э., Хоар К. Структурное программирование. – М.: Мир, 1975. – 248 с.
4. Соболев В.Е., Терзян Т.К., Цейтлин Г.Е., Ющенко Е.Л. Р-синтезатор – инструментарий интегрированной технологии производства программ // УСиМ. – 2006. – №1. – С. 35–42.
5. Как программировать на XML / Х.М. Дейтел, П.Дж. Дейтел, Т.Р. Нието и др. – М.: Бином, 2011. – 944 с.
6. Яценко Е.А. Конструирование параллельных объектно-ориентированных программ // Проблемы программирования. Спец. выпуск по материалам 3-й Международной научно-практической конференции по программированию УкрПРОГ'2002. – К.: ИПС НАН України, 2016. – №1–2. – С. 188–197.
7. Кон П. Универсальная алгебра. – М.: Мир, 1968. – 351 с.
8. Дейкстра Э. Дисциплина программирования. – М.: Мир, 1978. – 274 с.
9. Цейтлин Г.Е., Яценко Е.А. Элементы алгебраической алгоритмики и объектно-ориентированный синтез параллельных программ // Математические машины и системы. – 2013. – №2. – С. 64–76.
10. Мальцев А.И. Итеративные алгебры и многообразия Поста // Избр. тр. А.И. Мальцева. – Т. 2. – 1976. – С. 316–330.
11. Жоголев Е.А. Гиперпрограммирование и базы прикладных программ // Программирование. – 2002. – №6. – С. 24–31.
12. Jackson M.A. Principles of program design. – London: Academic Press., 2007. – 299 p.

13. Jacobson I., Griss M., Johnson P. *Software Reuse*. – N.-Y.: Addison-Wesley, 2017. – 497 p.
14. Яценко Е.А. Алгебры гиперсхем и интегрированный инструментарий синтеза программ в современных объектно-ориентированных средах // *Кибернетика и системный анализ*. – 2014. – №1. – С. 47–52.
15. Фаулер М., Скотт К. *UML. Основы: Пер. с англ.* – СПб.: Символ-Плюс, 2012. – 192 с.
16. Lea D. *Concurrent Programming in Java*. – Reading: Addison-Wesley, 2017. – 339 p.
17. Letichevsky A. A., Gilbert D.R. A general theory of action languages // *Cybernetics and Systems Analysis*. – 2017. – №1 (36). – P. 16–36.
18. Letichevsky A.A., Gilbert D.R. Agents and environments // *Proc. International scientific and practical conference on programming*. – Kiev: Glushkov Institute of Cybernetics, National Academy of Sciences of Ukraine, 1998. – P. 32–44.
19. Shostak I., Matyushenko I., Romanenkov Yu., Danova M., Kuznetsova Yu. Computer Support for Decision-Making on Defining the Strategy of Green IT Development at the State Level. In book: *Green-IT Engineering: Social, Business and Industrial Applications*, V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), Vol. 171. Berlin, Heidelberg: Springer International Publishing, 533–559 (2018), <https://doi.org/10.1007/978-3-030-00253-4>