

Б. І. Мокін, В. Б. Мокін, О. Б. Мокін

МАТЕМАТИЧНІ МЕТОДИ ІДЕНТИФІКАЦІЇ ТА МОДЕЛЮВАННЯ ДИНАМІЧНИХ СИСТЕМ

Міністерство освіти і науки України
Вінницький національний технічний університет

Б. І. Мокін, В. Б. Мокін, О. Б. Мокін

**МАТЕМАТИЧНІ МЕТОДИ
ІДЕНТИФІКАЦІЇ ТА МОДЕЛЮВАННЯ
ДИНАМІЧНИХ СИСТЕМ**

Навчальний посібник

Вінниця
ВНТУ
2026

УДК 004.681
М74

Рекомендовано до видання Вченою радою Вінницького національного технічного університету Міністерства освіти і науки України як навчальний посібник для студентів і аспірантів закладів вищої освіти, що спеціалізуються в галузі інформаційних технологій за спеціальностями F4 (124) «Системний аналіз та наука про дані» і F6 (126) «Інформаційні системи та технології» (протокол № 10 від «26» лютого 2026 р.)

Рецензенти:

В. Я. Данилов, доктор технічних наук, професор (НТУУ «КПІ»)

А. М. Сільвестров, доктор технічних наук, професор (НТУУ «КПІ»)

М. С. Юхимчук, доктор технічних наук, професор (ВНТУ)

Мокін, Б. І.

М74 Математичні методи ідентифікації та моделювання динамічних систем : навчальний посібник – вид. 2-е, змін. та доп. [Електронний ресурс] / Б. І. Мокін, В. Б. Мокін, О. Б. Мокін. Електрон. текст. дані. Вінниця : ВНТУ, 2026. PDF, 246 с.

ISBN 978-617-8163-97-6 (PDF)

В навчальному посібнику викладено математичні методи ідентифікації динамічних систем (як лінійних, так і нелінійних, як неперервних, так і дискретних, як у часовому просторі, так і на комплексній площині, як в детермінованому, так і в стохастичному варіантах, як з урахуванням зосереджених, так і розподілених параметрів) та програми комп'ютерного моделювання цих систем, складені з використанням мови Python.

Навчальний посібник є другим виданням навчального посібника «Математичні методи ідентифікації динамічних систем» цих же авторів, доповненого програмами комп'ютерного моделювання цього класу систем для врахування сучасних вимог методології ідентифікації та моделювання в галузі інформаційних технологій.

УДК 004.681

ISBN 978-617-8163-97-6 (PDF)

ISBN 978-966-641-988-3 (Друк.)

© ВНТУ, 2026

ЗМІСТ

ВСТУП	6
Частина I ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ.....	10
Розділ 1 МАТЕМАТИЧНІ МОДЕЛІ НЕПЕРЕРВНИХ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ.....	10
1.1 Математичні моделі неперервних детермінованих ЛДС ЗП у часовому просторі	10
1.1.1 Математичні моделі неперервних детермінованих ЛДС ЗП у часовому просторі у вигляді диференціальних рівнянь.....	10
1.1.2 Математичні моделі неперервних детермінованих ЛДС ЗП в просторі змінних стану	17
1.2 Математичні моделі неперервних детермінованих ЛДС ЗП на комплексній площині	20
1.3 Математичні моделі неперервних детермінованих ЛДС ЗП в частотній області	27
1.4. Ідентифікація моделей неперервних детермінованих ЛДС ЗП з використанням експериментальних даних	32
1.5 Завдання для самоперевірки.....	39
Розділ 2 МАТЕМАТИЧНІ МОДЕЛІ ДИСКРЕТНИХ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ.....	40
2.1 Решітчасті функції та скінченні різниці.....	40
2.2 Рівняння в скінченних різницях та різницеві рівняння.....	46
2.3 Дискретне перетворення Лапласа та Z-перетворення	49
2.4 Математична модель комп'ютера як елемента дискретної детермінованої ЛДС ЗП	62
2.5 Приклад побудови математичної моделі детермінованої дискретної ЛДС ЗП.....	66
2.6 Завдання для самоперевірки.....	71
Розділ 3 ФУР'Є-ІНТЕГРАЛЬНИЙ МЕТОД ІДЕНТИФІКАЦІЇ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ.....	72
3.1 Вихідні передумови Фур'є-інтегрального методу ідентифікації	72
3.2 Синтез алгоритмів параметричної ідентифікації сигналів на вході лінійної вимірювальної системи	73
3.3 Побудова алгоритму параметричної ідентифікації лінійної динамічної системи за допомогою ФІМІ	86
3.4 Екскурс у метод найменших квадратів	89
3.5 Завдання для самоперевірки.....	97

Частина 2 ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ СТОХАСТИЧНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ.....	99
Розділ 4 СТОХАСТИЧНІ МОДЕЛІ НЕПЕРЕРВНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ НА ОСНОВІ СТАЦІОНАРНИХ ВИПАДКОВИХ ПРОЦЕСІВ	99
4.1 Основні характеристики стаціонарних випадкових процесів	99
4.2 Кореляційні функції та спектральні густини стаціонарних випадкових процесів	109
4.3 Ідентифікація неперервних стохастичних ЛДС ЗП за допомогою рівняння Вінера-Хопфа.....	117
4.4 Фур'є-інтегральний метод ідентифікації стохастичних неперервних ЛДС ЗП	121
4.4.1 Отримання основних розрахункових співвідношень та побудова алгоритму ідентифікації.....	121
4.4.2 Обґрунтування регуляризованих властивостей ФІМІ.....	123
4.5 Еквівалентні моделі функцій та густин розподілу випадкових величин	123
4.6 Завдання для самоперевірки.....	136
Розділ 5 СТОХАСТИЧНІ МОДЕЛІ ДИСКРЕТНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ НА ОСНОВІ ЧАСОВИХ РЯДІВ.....	139
5.1 Часові ряди та їх основні характеристики	139
5.2 Синтез моделі стаціонарного часового ряду на основі лінійного фільтра	142
5.3 Модель стаціонарного часового ряду на основі авторегресії	143
5.4 Модель стаціонарного часового ряду на основі ковзного середнього	144
5.5 Комбінована модель стаціонарного часового ряду на основі авторегресії – ковзного середнього	145
5.6 Модель нестаціонарного часового ряду на основі авторегресії – проінтегрованого ковзного середнього	146
5.7 Автоковаріація та автокореляція часового ряду	149
5.8 Рівняння Юла-Уокера	150
5.9 Узагальнений метод ідентифікації моделі авторегресії-ковзного середнього	152
5.10 Приклади ідентифікації дискретної ЛДС ЗП на основі моделей часових рядів.....	156
5.11 Завдання для самоперевірки.....	173

Частина 3 ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З РОЗПОДІЛЕНИМИ ПАРАМЕТРАМИ ТА НЕЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ.....	174
Розділ 6 МАТЕМАТИЧНІ МОДЕЛІ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З РОЗПОДІЛЕНИМИ ПАРАМЕТРАМИ.....	174
6.1 Загальна характеристика лінійних динамічних систем з розподіленими параметрами	174
6.2 Рівняння математичної фізики як моделі елементів систем з розподіленими параметрами	176
6.3 Аналіз математичних моделей систем з розподіленими параметрами	183
6.4 Математичні моделі систем з чистим запізненням	191
6.5 Завдання для самоперевірки.....	198
Розділ 7 МАТЕМАТИЧНІ МОДЕЛІ НЕЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З АНАЛІТИЧНИМИ НЕЛІНІЙНОСТЯМИ	200
7.1 Математичні моделі нелінійних динамічних систем, статичні характеристики яких допускають лінеаризацію	200
7.2 Метод ідентифікації нелінійних динамічних систем з екстремальними статичними характеристиками.....	203
7.2.1 Вихідні умови та постановка задачі	203
7.2.2 Ідентифікація нелінійної статичної характеристики	204
7.2.3 Ідентифікація лінійної інерційної частини нелінійної динамічної системи	209
7.3 Завдання для самоперевірки.....	212
Розділ 8 МАТЕМАТИЧНІ МОДЕЛІ НЕЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З РЕЛЕЙНИМИ ХАРАКТЕРИСТИКАМИ	213
8.1 Математичні моделі релейних елементів з симетричними характеристиками	213
8.2 Математичні моделі нелінійних динамічних систем з релейними елементами	216
8.3 Аналіз процесів в нелінійних динамічних системах з релейними елементами	227
8.4 Завдання для самоперевірки.....	240
ПІДСУМКИ.....	242
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	244

ВСТУП

У будь-яких об'єктах під дією внутрішніх і зовнішніх впливів з часом виникають зміни станів, які прийнято називати динамічними процесами. Складні об'єкти, в яких протікають динамічні процеси, відносять до класу динамічних систем. В одній і тій же динамічній системі можуть протікати одночасно кілька динамічних процесів, до того ж різного характеру, і за тим, який із цих процесів для нас є домінуючим, ми можемо відносити одну й ту ж динамічну систему до різних класів, найбільшими з яких є: клас неперервних систем, в яких процеси протікають у часі неперервно; клас дискретних систем, в яких процеси формуються лише в окремі моменти часу; клас детермінованих систем, в яких внутрішні і зовнішні впливи підпорядковані відомим детермінованим залежностям; клас стохастичних систем, в яких внутрішні чи зовнішні впливи мають імовірнісний характер; клас лінійних систем, в яких реакція системи в усталеному режимі є пропорційною вхідному впливу; клас нелінійних систем, в яких реакція системи в усталеному режимі не є пропорційною вхідному впливу; клас систем з зосередженими параметрами, в яких реакція системи на зовнішній вплив є лише функцією часу, і клас систем з розподіленими параметрами, в яких реакція системи на зовнішній вплив є функцією не лише часу, а і однієї чи кількох просторових координат. Так, наприклад, за процесом споживання електричної енергії комп'ютер є неперервною динамічною системою, а за обробкою цифрової інформації – дискретною; резервуар з насосом для закачування рідини є лінійною динамічною системою за рівнем рідини в резервуарі, а цей же резервуар з компресором для закачування газу є нелінійною динамічною системою за тиском газу в резервуарі; електропривод компресора, який має зворотний зв'язок за тиском газу на початку трубопроводу і стабілізує цей тиск, є динамічною системою з зосередженими параметрами; а цей же електропривод компресора, який має зворотний зв'язок за тиском газу в кінці трубопроводу і стабілізує саме цей тиск, є динамічною системою з розподіленими параметрами; радіолокаційна станція за каналом реалізації обертання антен в горизонтальній і вертикальній площинах є системою детермінованою, а за каналом обробки сигналів, прийнятих антеною, є системою стохастичною. Очевидно, що перелік таких прикладів можна продовжувати і продовжувати.

При протіканні процесів в динамічних системах внаслідок взаємної невідповідності окремих або всіх характеристик основних елементів систем виникають втрати, які можуть досягати значень вихідного потоку.

Експериментальний пошук найкращих умов припасування елементів приводить до значних затрат коштів та часу, а у випадках великих потужностей і значних габаритів може не бути здійсненим взагалі.

Дуже часто динамічні системи мають у своїй структурі об'єкти з областями, забороненими для функціонування, оскільки в них виникають умови для їх руйнації внаслідок поломок або вибухів. Визначення границь таких областей є окремою важливою проблемою, для розв'язання якої експериментальний спосіб є неприйнятним взагалі.

Ще більше 200 років тому назад вчені різних країн звернули увагу на те, що різноманітні фізичні процеси можуть бути адекватно описані функціональними, диференціальними або інтегральними рівняннями, що їх з часом почали називати математичними моделями цих процесів.

Оскільки для визначення найкращих умов припасування розв'язків рівнянь, котрі адекватно описують процеси у динамічних системах, часу, людських та матеріальних ресурсів треба набагато менше у порівнянні з експериментальним їх визначенням на самих системах, то це об'єктивно спричинило стрімкий злет зацікавленості у математиків та фізиків до проблем побудови математичних моделей таких процесів, особливо після створення методів синтезу систем управління і оптимізації цих процесів з застосуванням математичних моделей. Станом на сьогодні таких моделей і методів їх побудови розроблено надзвичайно багато, але вони «розкидані», в основному, по статтях наукових журналів і спеціалізованих монографіях, а тому є необхідність в написанні навчального посібника, в якому в концентрованому вигляді у зрозумілому для студентів інженерних спеціальностей викладі основні з цих методів і моделей були б представлені. Саме цю мету і переслідував задуманий та створений нами і опублікований у 2010 році навчальний посібник: «Математичні методи ідентифікації динамічних систем», який отримав відповідний гриф міністерства і на який ми в подальшому будемо посилались як на посібник з ММІДС.

Очевидно, що в одному, обмеженому за кількістю сторінок, посібнику з ММІДС зібрати усі основні методи ідентифікації довільних динамічних систем було неможливо, тому ми зосередили свою увагу лише на ідентифікації тих динамічних систем, параметри структури яких в часі не змінюються – такі системи називають стаціонарними. А розгляд методів ідентифікації нестаціонарних динамічних систем відклали на майбутнє.

У першій частині посібника з ММІДС показано, як будувати математичні моделі неперервних та дискретних динамічних систем з зосередженими параметрами за умов, коли рівень завад та внутрішніх шумів у системах є суттєво нижчим від рівня основних процесів на всіх стадіях їх перетворення, що дає право відносити такі системи до класу детермінованих.

Але можна навести досить багато прикладів, коли ця умова не виконується. Наприклад, режим роботи електропривода екскаватора ніяк не можна віднести до детермінованих, оскільки навіть досвідчений машиніст екскаватора не здатний однозначно спрогнозувати скільки ґрунту прихопить за цикл ківш його машини, особливо коли є небезпека наштовхнутись на міцне коріння від якогось пенька або зачепитись за підземний виступ скельного каменю. Що ж стосується електроприводів міських водогінних станцій, то значення моментів навантаження на їх валах у довільний момент часу є випадковим принципово, оскільки ніхто не може наперед точно визначити скільки водозабірних кранів, підключених до водогону, буде у даний момент часу відкрито і на яку пропускну здатність. Випадковими, також, є і режими роботи дробарок природних рудних матеріалів. Приклади можна наводити і далі, але мабуть і цього досить, аби дійти висновку, що для практичного аналізу актуальними є

задачі побудови математичних моделей динамічних процесів, що їх відносять до класу стохастичних, головною особливістю яких є, як відомо, випадковість значень параметрів їх режимів у кожний момент часу.

Розгляду математичних методів ідентифікації стохастичних динамічних систем із зосередженими параметрами присвячена друга частина посібника з ММІДС.

Часто найбільш ефективне функціонування динамічних систем досягається лише на нелінійних ділянках характеристик перетворення енергії, а тому виникає необхідність у синтезі математичних моделей цих систем з врахуванням нелінійності характеристик елементів, одним із найпоширеніших класів яких є релейні.

Дуже важливо також при математичному моделюванні динамічних систем вміти враховувати запізнення проходження сигналів через елементи з розподіленими у просторі параметрами.

Розгляду математичних методів ідентифікації нелінійних динамічних систем та динамічних систем з розподіленими параметрами присвячена третя частина навчального посібника з ММІДС.

На завершення вступу як і у посібнику з ММІДС відзначимо, що ідентифікацією при побудові математичних моделей динамічних систем одні фахівці називають визначення числових значень коефіцієнтів синтезованих моделей, а інші – весь процес побудови математичної моделі: від вибору типу і форми залежності між вхідною і вихідною величинами до визначення числових значень її коефіцієнтів.

Ми будемо використовувати термін «ідентифікація» в найбільш широкому його значенні, вважаючи тотожними словосполучення “синтез математичної моделі, ідентичної характеристикам системи, що моделюється” та “ідентифікація системи”. Одночасно нагадаємо, що процес ідентифікації динамічної системи містить у собі три складових, першою з яких є вибір відповідної математичної залежності, що пов’язує між собою параметри системи та параметри її режиму, тобто, вибір структури математичної моделі системи; другою складовою є вибір критерію, мінімізацією якого досягається визначення оптимальних значень параметрів математичної моделі; а третьою складовою є вибір алгоритму пошуку значень параметрів математичної моделі, оптимальних за вибраним критерієм.

Пам’ятаючи про те, що математичні моделі для оцінювання стану, управління і прогнозу – це моделі, різні за своєю структурою та множиною вхідних змінних, ми будемо розкривати ці відмінності в процесі викладення матеріалу, не роблячи окремих узагальнень за цією ознакою.

Даний навчальний посібник є другим виданням нашого навчального посібника з ММІДС, перше видання якого побачило світ у 2010 році і отримало широку популярність в студентській та аспірантській аудиторіях. У другому виданні враховані побажання студентів стосовно змін і доповнень його змісту до сучасних вимог методології розв’язання задач ідентифікації в галузі інформаційних технологій, обумовлених необхідністю використання у цьому процесі комп’ютерних програм, створених з використанням сучасних мов

програмування, серед яких лідирує нині мова програмування Python. Тож, доповнивши у другому виданні посібник з ММІДС комп'ютерними програмами, створеними з використанням мови Python, та прикладами моделювання процесів в динамічних системах з використанням цих комп'ютерних програм, тобто, розширивши зміст навчального посібника з ММІДС, ми вирішили для орієнтації читачів підправити у відповідності з новими змістовними реаліями і його назву, назвавши його друге видання «Математичні методи ідентифікації та моделювання динамічних систем» і у такий спосіб трансформували навчальний посібник з ММІДС в навчальний посібник з ММІМДС.

Оскільки при написанні як посібника з ММІДС так і посібника з ММІМДС використані лише відомі та опубліковані результати, то, навівши список літератури, з якої взяті ці результати, ми при викладенні конкретного матеріалу не стали робити посилань на те, звідки що взято, – це буде стимулювати студентів до ознайомлення з усіма першоджерелами.

Частина I

ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ

Розділ 1 МАТЕМАТИЧНІ МОДЕЛІ НЕПЕРЕРВНИХ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ

У цій частині навчального посібника охарактеризуємо і покажемо як побудувати математичні моделі детермінованих лінійних динамічних систем з зосередженими параметрами (ЛДС ЗП) – як неперервних, так і дискретних – в часовій та частотній областях, а також на комплексній площині. Покажемо також, як, використовуючи ці моделі та комп'ютерні програми, складені на мові Python, досліджувати процеси в динамічних системах, що моделюються.

Але спочатку нагадаємо, що для того, щоб динамічна система мала зосереджені параметри, її структура не повинна мати елементів, проходження сигналів у яких характеризується помітним запізненням.

Щоб динамічна система була лінійною, вона повинна забезпечувати в усталеному режимі пропорційність між значеннями вхідних сигналів та реакціями системи на них.

Щоб динамічну систему можна було вважати детермінованою, достатньо забезпечити в ній суттєво менший рівень збурень у порівнянні із рівнем основних функціональних сигналів.

1.1 Математичні моделі неперервних детермінованих ЛДС ЗП у часовому просторі

1.1.1 Математичні моделі неперервних детермінованих ЛДС ЗП у часовому просторі у вигляді диференціальних рівнянь

Ще з шкільного курсу фізики нам відомо, що напруга $U(t)$, яка прикладається до котушки з індуктивністю L та внутрішнім опором r (рис. 1.1), урівноважується сумою падіння напруги $r \cdot I(t)$, обумовленого протіканням струму $I(t)$ по внутрішньому опору r , та електрорушійною силою самоіндукції $L \frac{dI(t)}{dt}$, тобто має місце рівність

$$L \frac{dI(t)}{dt} + r \cdot I(t) = U(t), \quad (1.1)$$

яка являє собою диференціальне рівняння 1-го порядку, оскільки одна із змінних, які воно зв'язує між собою, входить в нього разом зі своєю похідною.

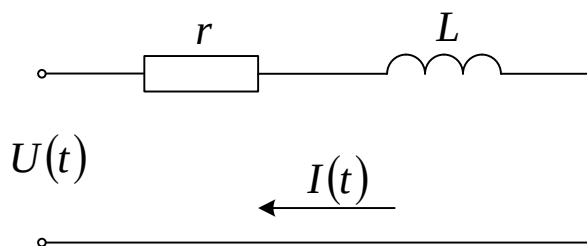


Рисунок 1.1 – Електрична схема підключення котушки індуктивності під напругу

Якщо котушку індуктивності розглядати як динамічну систему (ДС), на вхід якої надходить сигнал $U(t)$, котрий викликає у ній реакцію у вигляді $I(t)$ (рис. 1.2), то диференціальне рівняння (1.1) буде відігравати роль математичної моделі цієї системи, оскільки його розв'язок $I(t) = f(U, t)$ однозначно віддзеркалюватиме процес зміни в часі її реакції на даний вхідний сигнал.



Рисунок 1.2 – Структурна схема котушки індуктивності як динамічної системи

А тепер подамо ту ж саму напругу $U(t)$ на послідовне з'єднання тієї ж самої котушки індуктивності з конденсатором ємністю C (рис. 1.3), напруга на якому, як відомо з курсу фізики, дорівнює $\frac{1}{C} \int I(t) dt$

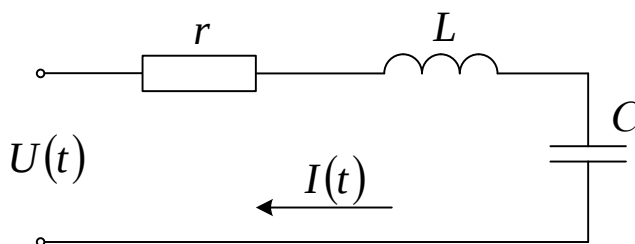


Рисунок 1.3 – Електрична схема підключення під напругу послідовного з'єднання котушки індуктивності та конденсатора

У цьому випадку прикладена до послідовного з'єднання котушки індуктивності та конденсатора напруга буде врівноважуватись сумою падіння напруги, обумовленого протіканням струму по внутрішньому опорі котушки, електрорушійної сили самоіндукції котушки і напруги на конденсаторі, тобто матиме місце рівність

$$L \frac{dI(t)}{dt} + r \cdot I(t) + \frac{1}{C} \int I(t) dt = U(t), \quad (1.2)$$

яка являє собою інтегро-диференціальне рівняння, оскільки одна із змінних, які воно зв'язує, входить в нього не лише зі своєю похідною, а й зі своїм інтегралом.

Для того, щоб рівняння (1.2) перетворити в чисто диференціальне, продиференціюємо обидві його частини. В результаті цього диференціювання з урахуванням того, що похідна від невизначеного інтеграла дорівнює підінтегральній функції, отримаємо рівність

$$L \frac{d^2 I}{dt^2} + r \frac{dI}{dt} + \frac{1}{C} I(t) = \frac{dU}{dt}, \quad (1.3)$$

яка являє собою диференціальне рівняння 2-го порядку, оскільки одна із змінних, які воно зв'язує, входить в нього не лише зі своєю першою, а й зі своєю другою похідною, що має найвищий порядок у цьому рівнянні, а тому і визначає порядок рівняння.

На підставі міркувань, аналогічних наведеним вище, під час розгляду котушки індуктивності як динамічної системи, в разі розгляду послідовного з'єднання котушки індуктивності та конденсатора як динамічної системи з вхідним сигналом $U(t)$ та реакцією на нього $I(t)$ (рис. 1.2), можна стверджувати, що диференціальне рівняння (1.3) відіграє роль математичної моделі цієї динамічної системи.

Звертаємо увагу на те, що коли система мала у своїй структурі лише один елемент (катушку індуктивності), здатний запасати на якийсь час енергію (індуктивну), то для моделювання процесу зміни станів у ній достатньо було диференціального рівняння 1-го порядку, а коли у структурі системи з'явився ще й другий елемент (конденсатор), здатний запасати на якийсь час енергію (ємнісну), то для моделювання зміни станів у ній уже знадобилось диференціальне рівняння 2-го порядку.

Легко показати, користуючись лише знаннями розділу «Механіка» шкільного курсу фізики, що аналогічна ситуація матиме місце і в динамічних системах з двома елементами, здатними запасати на якийсь час кінетичну та потенціальну енергію.

А у зв'язку з тим, що структура довільної ЛДС ЗП з вхідним сигналом $x(t)$ і реакцією на нього $y(t)$ може мати n елементів, здатних запасати на якийсь час якийсь вид енергії, то адекватно моделювати зміни станів у ній можна лише за допомогою диференціального рівняння n -го порядку

$$\sum_{i=0}^n a_i \frac{d^i y}{dt^i} = \sum_{l=0}^m b_l \frac{d^l x}{dt^l}, \quad (1.4)$$

для якого виконується умова

$$m \leq n, \quad (1.5)$$

що обумовлена здатністю системи фізично реалізуватись під таку модель.

Для отримання розв'язку диференціального рівняння 1-го порядку (1.1) це рівняння необхідно один раз проінтегрувати. В результаті цього інтегрування в розв'язку з'явиться стала інтегрування, для конкретизації якої необхідно задати початкову умову для вихідної координати $I(t)$, тобто задати умову

$$I(t)|_{t=0} = I(0). \quad (1.6)$$

Це означає, що з множини траєкторій руху, які задаються загальним розв'язком диференціального рівняння 1-го порядку, потрібно вибрати ту, яка в початковий момент часу ($t=0$) проходить через точку $I(0)$.

З фізичної точки зору це означає, що, моделюючи процес зміни струму $I(t)$ в котушці індуктивності після подачі на її затискачі напруги $U(t)$, потрібно врахувати в момент подачі напруги той залишковий струм, який мав місце у цій котушці індуктивності у цей момент в результаті її звільнення від енергії, привнесеної попереднім вхідним сигналом.

Оскільки для отримання розв'язку диференціального рівняння 2-го порядку його необхідно проінтегрувати двічі, що викликає появу одна за одною двох сталих інтегрування, то для їх конкретизації уже потрібно знати в початковий момент часу ($t=0$) не лише значення вихідної координати $I(t)$,

але і значення її першої похідної $\frac{dI}{dt}$, тобто початкова умова для рівняння (1.3)

набуває вигляду

$$\left. \begin{aligned} I(t)|_{t=0} &= I(0), \\ \frac{dI}{dt}|_{t=0} &= I^1(0). \end{aligned} \right\} \quad (1.7)$$

Цілком очевидно, що для отримання розв'язку диференціального рівняння n -го порядку його необхідно проінтегрувати n раз, що викликає появу n сталих інтегрування, для конкретизації яких потрібно знати в початковий момент часу не лише значення вихідної координати $y(t)$, але і значення у цей початковий момент усіх її похідних до $(n-1)$ порядку включно, тобто початкова умова для рівняння (1.4) має вигляд

$$\left. \begin{aligned} y(t)|_{t=0} &= y(0), \\ \frac{dy}{dt}|_{t=0} &= y^1(0), \\ &\dots\dots\dots, \\ \frac{d^{n-1}y}{dt^{n-1}}|_{t=0} &= y^{n-1}(0). \end{aligned} \right\} \quad (1.8)$$

Слід відзначити, що для значної кількості ЛДС ЗП характерним є те, що всі елементи їхніх структур, які здатні запасати енергію, після відключення системи цю енергію втрачають, що дає підставу перед новим включенням у роботу системи для її моделі у вигляді (1.4) вважати початкові умови нульовими, тобто в системі рівнянь (1.8) вважати усі праві частини такими, що дорівнюють нулю. Це суттєво спрощує подальший процес моделювання.

А далі на прикладах покажемо, як використовуючи математичні моделі динамічних систем у вигляді диференціальних рівнянь та комп'ютерні програми, складені на мові Python, досліджувати процеси у цих системах.

Отже, нехай математична модель динамічної системи задана у формі диференціального рівняння (ДР)

$$\frac{d^2u}{dt^2} + 3 \frac{du}{dt} + 2u = 2t \quad (1.9)$$

при початкових умовах: $u(0) = 0$; $u'(0) = 1$.

Розв'язуючи диференціальне рівняння (1.9) при заданих початкових умовах, ми отримаємо функцію $u(t)$, яка описуватиме процес змін вихідної координати динамічної системи з математичною моделлю у вигляді диференціального рівняння (1.9), викликаний лінійно наростаючим вхідним сигналом $2t$.

Приклад № 1, присвячений розв'язанню диференціального рівняння (ДР) з використанням комп'ютерної програми, складеної на мові Python в рамках **ППП sympy**

In [1]: import sympy	# Виклик ППП sympy	
In [2]: from sympy import *	# Виклик із sympy усіх функцій	
In [3]: from IPython.display import *	# Виклик із IPython.display усіх функцій	
In [4]: init_printing(use_latex=True)	# Запуск режиму «красивого» друку	
In [5]: t=symbols('t')	# Символізація змінної t	
In [6]: u=Function('u')(t)	# Символізація функції u(t)	
In [7]: de=Eq(u.diff(t,2)+3*u.diff(t)+2*u,2*t)	# Запис ДР у формі Eq(ДР)	
In [8]: display(de)	# «Красива» візуалізація ДР	
		$2u(t) + 3 \frac{d}{dt} u(t) + \frac{d^2}{dt^2} u(t) = 2t$
In [9]: des=dsolve(de,u)	# Розв'язання ДР	
In [10]: display(des)	# Візуалізація розв'язку ДР	
		$u(t) = C_1 e^{-2t} + C_2 e^{-t} + t - \frac{3}{2}$
In [11]: var('C1 C2')	# Символізація сталих C1,C2	
Out[11]: (C1, C2)		
In [12]: eq1=des.rhs.subs(t,0);eq1	# Формування правої частини функції u(t) у формі u(0)	
Out[12]:		$C_1 + C_2 - \frac{3}{2}$
In [13]: eq2=des.rhs.diff(t).subs(t,0);eq2	# Формування правої частини похідної u'(t) у формі u'(0)	
Out[13]:		

```

                                 $-2C_1 - C_2 + 1$ 
In [14]: seq=solve([eq1,eq2-1],C1,C2);seq      # Розв'язання системи рівнянь
Out[14]:                                     eq1,eq2 для  $u(0) = 0, u'(0) = 1$ 
                                 $\left\{C_1: -\frac{3}{2}, C_2: 3\right\}$ 
In [15]: rez=des.rhs.subs([(C1,seq[C1]),      # Підстановка значень констант
(C2,seq[C2])])                                $C_1, C_2$  в розв'язок ДР des
In [16]: display(rez)                        # Друк розв'язку ДР  $u(t)$ 
                                 $t - \frac{3}{2} + 3e^{-t} - \frac{3}{2}e^{-2t}$ 
In [17]: from sympy.plotting import plot      # Виклик із sympy.plotting функції plot
In [18]: plot(rez,(t,0,1))                   # Побудова графіка розв'язку ДР

```

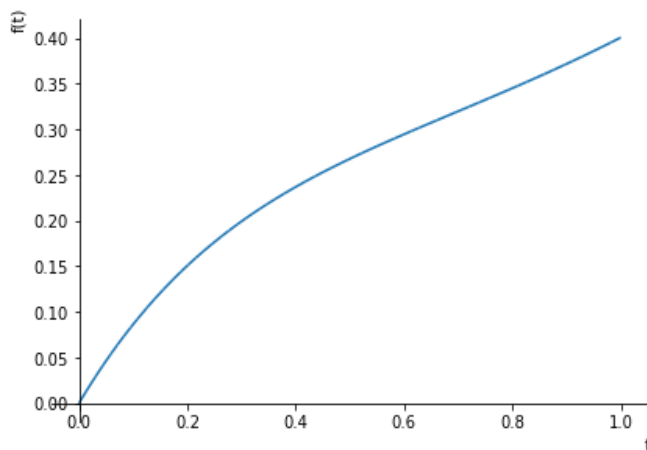


Рисунок 1.4 – Графік розв'язку ДР (1.9) для прикладу №1

А тепер нехай математична модель динамічної системи задана у формі системи ДР

$$\left. \begin{aligned} \frac{dx}{dt} &= -x(t) + y(t) + z(t), \\ \frac{dy}{dt} &= -x(t) + y(t) - z(t), \\ \frac{dz}{dt} &= x(t) - y(t) + z(t), \end{aligned} \right\} \quad (1.10)$$

при початкових умовах: $x(0) = 1$; $y(0) = -1$; $z(0) = 0$.

Розв'язуючи систему диференціальних рівнянь (1.10) при заданих початкових умовах, ми отримаємо функції $x(t), y(t), z(t)$, які описуватимуть процес змін внутрішніх координат багатозв'язної динамічної системи з математичною моделлю у вигляді системи ДР (1.10), викликаний ненульовими початковими умовами.

Приклад № 2, присвячений розв'язанню системи ДР (1.10) з використанням комп'ютерної програми, складеної на мові Python в рамках **ППП sympy**

```

In [1]: import sympy                # Виклик ППП sympy
In [2]: from sympy import*          # Виклик із sympy усіх функцій
In [3]: from IPython.display import* # Виклик із IPython.display усіх функцій

```



```

In [4]: init_printing(use_latex=True)          # Запуск режиму «красивого» друку
In [5]: t=symbols('t')                        # Символізація змінної t
In [6]: x,y,z=symbols('x y z', cls = Function) # Символізація функцій x,y,z
In [7]: eq1=Eq(x(t).diff(t),-x(t)+y(t)+z(t)) # Формування 1-го рівняння системи ДР
In [8]: eq2=Eq(y(t).diff(t),-x(t)+y(t)-z(t)) # Формування 2-го рівняння системи ДР
In [9]: eq3=Eq(z(t).diff(t), x(t)-y(t)+z(t)) # Формування 3-го рівняння системи ДР
In [10]: rez=dsolve((eq1,eq2,eq3))           # Розв'язання системи ДР
In [11]: display(rez[0])                      # Друк складової x(t) розв'язку

$$x(t) = -3C_1 e^{-t} + C_2$$

In [12]: display(rez[1])                      # Друк складової y(t) розв'язку

$$y(t) = -C_1 e^{-t} + C_2 - C_3 e^{2t}$$

In [13]: display(rez[2])                      # Друк складової z(t) розв'язку

$$z(t) = C_1 e^{-t} + C_3 e^{2t}$$

In [14]: eq4=rez[0].rhs.subs(t,0);eq4        # Формування правої частини
Out[14]:                                     функції x(t) у формі x(0)

$$-3C_1 + C_2$$

In [15]: eq5=rez[1].rhs.subs(t,0);eq5        # Формування правої частини
Out[15]:                                     функції y(t) у формі y(0)

$$-C_1 + C_2 - C_3$$

In [16]: eq6=rez[2].rhs.subs(t,0);eq6        # Формування правої частини
Out[16]:                                     функції z(t) у формі z(0)

$$C_1 + C_3$$

In [17]: var('C1 C2 C3')                     # Символізація сталих C1,C2,C3
Out[17]:                                     (C1, C2, C3)
In [18]: seq1=solve([eq4+1,eq5-1,eq6],      # Визначення сталих C1,C2,C3
C1,C2,C3);seq1                               для умов: x(0)=-1, y(0)=1, z(0)=0
Out[18]:                                      $\left\{C_1: \frac{2}{3}, C_2: 1, C_3: \frac{1}{3}\right\}$ 
In [19]: rez1=rez[0].rhs.subs([(C1,seq1[C1]), # Підстановка сталих C1, C2
(C2,seq1[C2])]);rez1                          із словника в розв'язок rez[0]
Out[19]:                                      $1 - 2e^{-t}$ 
In [20]: rez2=rez[1].rhs.subs([(C1,seq1[C1]), # Підстановка сталих C1, C2, C3
(C2,seq1[C2]),(C3,seq1[C3])]);rez2            із словника в розв'язок rez[1]
Out[20]:                                      $1 - \frac{1}{3}e^{2t} - \frac{2}{3}e^{-t}$ 
In [21]: rez3=rez[2].rhs.subs([(C1,seq1[C1]), # Підстановка сталих C1, C3
(C3,seq1[C3])]);rez3                          із словника в розв'язок rez[2]
Out[21]:                                      $\frac{1}{3}e^{2t} + \frac{2}{3}e^{-t}$ 
In [22]: from sympy.plotting import plot      # Виклик із sympy.plotting функції plot
In [23]: gr1=plot(rez1,(t,0,1),show=         # Побудова графіка розв'язку x(t)
False, line_color='k')                       без виведення на екран

```

```

In [24]: gr2=plot(rez2,(t,0,1),show=
False, line_color='g')
In [25]: gr3=plot(rez3,(t,0,1),show=
False, line_color='r')
In [26]: gr1.extend(gr2)
In [27]: gr1.extend(gr3)
In [28]: gr1.show()

```

```

# Побудова графіка розв'язку y(t)
без виведення на екран
# Побудова графіка розв'язку z(t)
без виведення на екран
# Приєднання графіка y(t) до x(t)
# Приєднання графіка z(t) до x(t)
# Побудова графіків x(t), y(t), z(t)
на одному рисунку

```

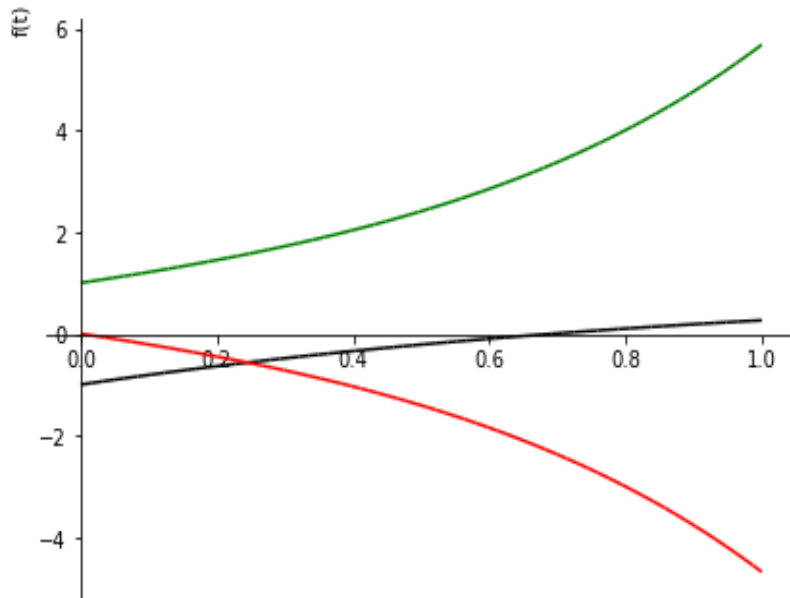


Рисунок 1.5 – Графіки розв'язків системи ДР (1.10) для прикладу № 2

1.1.2 Математичні моделі неперервних детермінованих ЛДС ЗП в просторі змінних стану

Нехай відомо, що процеси у динамічній системі описуються диференціальним рівнянням

$$a_3 \frac{d^3 y}{dt^3} + a_2 \frac{d^2 y}{dt^2} + a_1 \frac{dy}{dt} + a_0 y = bu, \quad (1.11)$$

де u – сигнал управління, а y – реакція системи на нього.

Позначимо

$$y = y_1. \quad (1.12)$$

Тоді

$$\frac{dy}{dt} = \frac{dy_1}{dt}. \quad (1.13)$$

Введемо заміну

$$\begin{cases} \frac{dy_1}{dt} = y_2, \\ \frac{dy_2}{dt} = y_3. \end{cases} \quad (1.14)$$

Перепишемо рівняння (1.11) у формі:

$$\frac{d^3 y}{dt^3} = -a_0 y - a_1 \frac{dy}{dt} - a_2 \frac{d^2 y}{dt^2} + bu. \quad (1.15)$$

З урахуванням (1.12) – (1.14) рівняння (1.15) можна записати у вигляді

$$\frac{dy_3}{dt} = -a_0 y_1 - a_1 y_2 - a_2 y_3 + bu. \quad (1.16)$$

У результаті цих перетворень диференціальне рівняння 3-го порядку (1.11) відносно координати y приведено до системи трьох диференціальних рівнянь 1-го порядку відносно координат y_1, y_2, y_3 , тобто еквівалентом для (1.11) є система диференціальних рівнянь:

$$\begin{cases} \frac{dy_1}{dt} = y_2, \\ \frac{dy_2}{dt} = y_3, \\ \frac{dy_3}{dt} = -a_0 y_1 - a_1 y_2 - a_2 y_3 + bu, \end{cases} \quad (1.17)$$

яку можна переписати у матричній формі у вигляді

$$\dot{Y} = AY + BU, \quad (1.18)$$

де

$$\begin{aligned} \dot{Y}_{(3 \times 1)} &= \begin{Bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{Bmatrix}, \quad U_{(3 \times 1)} = \begin{Bmatrix} 0 \\ 0 \\ u \end{Bmatrix}, \quad Y_{(3 \times 1)} = \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \end{Bmatrix}, \\ A_{(3 \times 3)} &= \begin{Bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 \end{Bmatrix}, \quad B_{(3 \times 3)} = \begin{Bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{Bmatrix}. \end{aligned} \quad (1.19)$$

Матричне рівняння (1.18) легко узагальнюється на довільний порядок вектора $\dot{Y}$, який характеризує стан системи, та вектора управління U за умови лише, що виконується умова (1.5).

Вирази (1.17) або (1.18) та (1.19) задають математичну модель динамічної системи у просторі змінних її стану, до яких відносять саму вихідну координату y та її $(n-1)$ похідну.

Запис математичної моделі динамічної системи у просторі змінних стану широко використовується в теорії автоматичного управління під час розв'язання задач синтезу систем оптимального керування.

Варто нагадати, що в літературі до 80-х років при $n=3$ простір змінних стану було прийнято називати фазовим простором, а при $n=2$ – фазовою площиною.

А далі на прикладі покажемо, як, використовуючи математичну модель динамічної систем у вигляді диференціального рівняння 2-го порядку та його трансформацію у простір змінних стану, за допомогою комп'ютерної програми, складеної на мові Python, досліджувати процес у цій системі.

Отже, нехай математична модель динамічної системи задана у формі диференціального рівняння (ДР)

$$\frac{d^2y}{dt^2} + t \frac{dy}{dt} + \sqrt{y} = 2\cos(3t) \quad (1.20)$$

при початкових умовах:

$$y(0) = 0.0; \quad y'(0) = 1.5. \quad (1.21)$$

Аналізуючи ДР (1.20), бачимо, що воно є нелінійним за рахунок кореня квадратного з координати y і відноситься до класу нестационарних внаслідок залежності коефіцієнта при першій похідній y' від незалежної змінної t .

Диференціальне рівняння такого класу є непосильним для символічного **Python sympy**, тому для його розв'язання необхідно застосовувати метод послідовних наближень, для чого це рівняння необхідно спочатку трансформувати у простір змінних стану, а потім для розв'язання системи ДР у просторі змінних стану використати **Python scipy** у поєднанні з **Python numpy** для представлення масиву вхідних даних та **Python matplotlib** для реалізації графіки.

Отже спочатку ДР (1.20) трансформуємо у простір змінних стану. Використовуючи викладену вище методику, матимемо:

$$\begin{cases} \frac{dy_1}{dt} = y_2, \\ \frac{dy_2}{dt} = -t y_2 - \sqrt{y_1} + 2\cos(3t), \end{cases} \quad (1.22)$$

де $y = y_1$, а початкові умови (1.21) набувають вигляду

$$y_1(0) = 0.0; \quad y_2(0) = 1.5. \quad (1.23)$$

Розв'язуючи систему диференціальних рівнянь 1-го порядку (1.22) при заданих початкових умовах (1.23), ми отримаємо функцію $y_1(t)$, яка описуватиме процес змін вихідної координати $y(t)$ динамічної системи з математичною моделлю у вигляді диференціального рівняння (1.20), викликаний вхідним сигналом $2\cos(3t)$ та початковими умовами (1.21).

Приклад № 3, присвячений розв'язанню системи диференціальних рівнянь 1-го порядку на відрізку $t \in [0,3]$ з використанням комп'ютерної

програми, складеної на мові Python в рамках **ПППP scipy** з залученням **ПППP numpy** та **ПППP matplotlib**

In [1]: import numpy as np	# Виклик ППП numpy як <i>np</i>
In [2]: import matplotlib	# Виклик matplotlib
In [3]: import matplotlib.pyplot as plt	# Виклик matplotlib. pyplot як <i>plt</i>
In [4]: import scipy	# Виклик ППП scipy
In [5]: from scipy.integrate import odeint	# Виклик функції <i>odeint</i>
In [6]: def f(y,t):	# Формування системи ДР 1-го
y1,y2=y	порядку після трансформації ДР
return [y2,-t*y2-	2-го порядку у цю систему
-y1**0.5+2*np.cos(3*t)]	
....	
In [7]: t=np.linspace(0.0,3.0,31)	# Формування масиву для <i>t</i> в <i>numpy</i>
In [8]: y0=[0.0,1.5]	# Внесення початкових умов
In [9]: w=odeint(f,y0,t)	# Розв'язання системи ДР
In [10]: y1=np.array(w[:,0]).flatten()	# Трансформація розв'язку в масив
In [11]: fig=plt.figure(facecolor='white')	# Формування поля <i>рисунку</i>
In [12]: plt.plot(t,y1,'-k',linewidth=4)	# Графік розв'язку

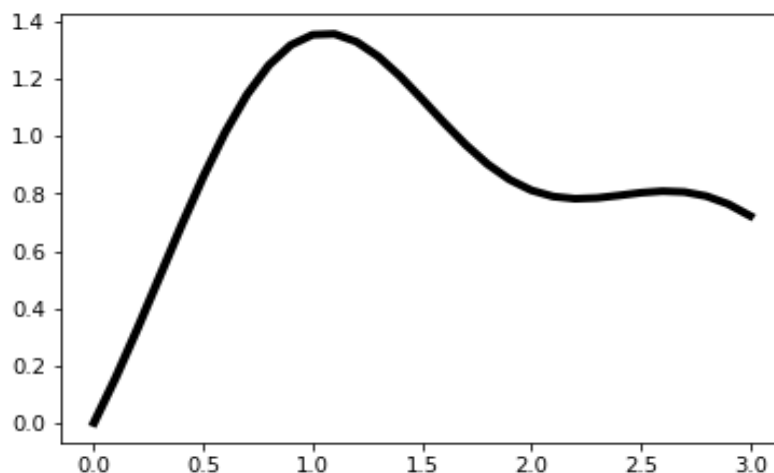


Рисунок 1.6 – Графік розв'язку ДР 2-го порядку для прикладу № 3

1.2 Математичні моделі неперервних детермінованих ЛДС ЗП на комплексній площині

З курсу операційного числення, що є одним з прикладних розділів загального курсу функціонального аналізу, відомо, що за допомогою оператора Лапласа

$$L\{f(t)\} = F(p) = \int_0^{\infty} f(t)e^{-pt} dt \quad (1.24)$$

кожній функції f часу t , яка задовольняє умову $f(t)=0$ при $t < 0$, умови Діріхле і називається *оригіналом*, можна поставити у відповідність функцію F

комплексної змінної p , яка називається зображенням оригіналу на комплексній площині. Ця відповідність записується у такий спосіб:

$$f(t) \Leftrightarrow F(p). \quad (1.25)$$

Наприклад, часовій функції одиничного стрибка

$$1(t) = \begin{cases} 1, & t \geq 0, \\ 0, & t < 0 \end{cases} \quad (1.26)$$

на комплексній площині відповідає зображення

$$F(p) = \int_0^{\infty} 1(t) e^{-pt} dt = \int_0^{\infty} e^{-pt} dt = \frac{1}{-p} \cdot e^{-pt} \Big|_0^{\infty} = \frac{1}{p}. \quad (1.27)$$

Ще один приклад – експоненті $e^{-\alpha t}$ при $t \geq 0$ на комплексній площині відповідає зображення

$$F(p) = \int_0^{\infty} e^{-\alpha t} \cdot e^{-pt} dt = \int_0^{\infty} e^{-(p+\alpha)t} dt = \frac{1}{-(p+\alpha)} e^{-(p+\alpha)t} \Big|_0^{\infty} = \frac{1}{p+\alpha} \quad (1.28)$$

що у відповідності з виразом (1.25) можна подати як

$$e^{-\alpha t} \Leftrightarrow \frac{1}{p+\alpha}. \quad (1.29)$$

Для більш складних функцій $f(t)$ взяття інтегралу в операторі Лапласа (1.24) для отримання їх зображень $F(p)$ може бути ускладненим, тому на прикладі лінійно наростаючої функції $f(t) = t$ покажемо як це зображення можна віднайти за допомогою комп'ютерної програми, складеної на мові Python.

Приклад № 4, присвячений визначенню зображення $F(p)$ за Лапласом функції $f(t) = t$ з використанням комп'ютерної програми, складеної на мові Python в рамках *ППП sympy*

In [1]: import sympy	# Виклик ППП sympy
In [2]: from sympy import *	# Виклик із sympy усіх функцій
In [3]: t = symbols('t')	# Символізація змінної t
In [4]: p = symbols('p')	# Символізація змінної p
In [5]: f = Function('f')(t)	# Символізація функції f(t)
In [6]: F = Function('F')(p)	# Символізація функції F(p)
In [7]: f = t	# Формування функції f(t)
In [8]: f1 = f*exp(-p*t)	# Формування ядра інтеграла
In [9]: F1 = integrate(f1, (t, 0, oo))	# Взяття інтеграла Лапласа
In [10]: F1	# Виклик на екран результату
Out[10]:	інтегрування у вигляді
Piecewise((p**(-2), Abs(arg(p)) < pi/2),	складеної функції у формі кортежу
(Integral(t*exp(-p*t), (t, 0, oo)), True))	
In [11]: K = ((p**(-2), Abs(arg(p)) < pi/2),	# Виклик на екран кортежу

```
(Integral(t*exp(-p*t), (t, 0, oo)), True))
In [12]: K[0][0]
Out [12]: p**(-2)
In [13]: F = K[0][0]
In [14]: F
Out[14]: p**(-2)
```

складеної функції
 # **Виклик** на екран елемента
 кортежу з індексом [0][0]
 # **Трансформація** елемента
 кортежу в зображення $F(p)$
 по Лапласу функції $f(t)$

Головною перевагою аналізу в області зображень $F(p)$, тобто на комплексній площині, у порівнянні з аналізом в області оригіналів $f(t)$, тобто у часовому просторі, є те, що за нульових початкових умов операції диференціювання $\frac{d}{dt}$ оригіналу $f(t)$ у часовому просторі відповідає операція перемноження на комплексну змінну p його зображення $F(p)$ на комплексній площині, тобто

$$\frac{df}{dt} = \dot{f}(t) \Leftrightarrow p \cdot F(p). \quad (1.30)$$

Як наслідок цієї властивості маємо:

$$\frac{d^2 f}{dt^2} = \ddot{f}(t) \Leftrightarrow p^2 \cdot F(p) \quad (1.31)$$

та

$$\frac{d^n f}{dt^n} = f^{(n)}(t) \Leftrightarrow p^n \cdot F(p), \quad (1.32)$$

а також

$$\int_0^t f(\tau) d\tau \Leftrightarrow \frac{F(p)}{p} \quad (1.33)$$

та

$$\underbrace{\int_0^t \dots \int_0^t}_{n} f(\tau_1, \tau_2, \dots, \tau_n) d\tau_1 d\tau_2 \dots d\tau_n \Leftrightarrow \frac{F(p)}{p^n}. \quad (1.34)$$

Завдяки властивості (1.30) та її наслідкам (1.31) – (1.34) диференціальним та інтегральним рівнянням, записаним у часовому просторі, відповідають алгебраїчні рівняння на комплексній площині, розв'язувати які набагато простіше, оскільки цьому вчать ще у школі.

Наприклад, диференціальному рівнянню в області оригіналів $x(t)$, $y(t)$

$$a_2 \frac{d^2 y(t)}{dt^2} + a_1 \frac{dy(t)}{dt} + a_0 y(t) = b_1 \frac{dx(t)}{dt} + b_0 x(t) \quad (1.35)$$

на комплексній площині відповідає алгебраїчне рівняння

$$a_2 p^2 Y(p) + a_1 p Y(p) + a_0 Y(p) = b_1 p X(p) + b_0 X(p) \quad (1.36)$$

відносно зображень $X(p)$ та $Y(p)$. Його розв'язком є функція $Y(p)$, яку із рівняння (1.36) можна визначити у такий спосіб:

$$Y(p) = \frac{b_1 p + b_0}{a_2 p^2 + a_1 p + a_0} X(p) \quad (1.37)$$

або

$$Y(p) = W(p) \cdot X(p), \quad (1.38)$$

де

$$W(p) = \frac{b_1 p + b_0}{a_2 p^2 + a_1 p + a_0}. \quad (1.39)$$

Якщо $X(p)$ вважати зображенням сигналу $x(t)$, який надходить на вхід динамічної системи (рис. 1.7), математичну модель якої можна записати у вигляді (1.38), де $y(t)$ є реакцією цієї системи на вхідний сигнал $x(t)$, то функцію $W(p)$ можна інтерпретувати як передаточну (або передатну) функцію системи.

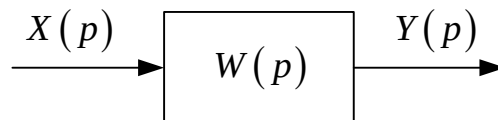


Рисунок 1.7 – Узагальнена структурна схема динамічної системи в області зображень

Як видно із виразів (1.37) – (1.39), передаточна функція $W(p)$ не залежить від зовнішніх сигналів, які діють на систему, та однозначно характеризує внутрішню структуру і властивості лише самої системи.

А тому ця функція є однією із найважливіших математичних моделей лінійних систем, для яких перетворення Лапласа (1.24) задає взаємно однозначну відповідність між оригіналами та їх зображеннями.

Із виразу (1.38) випливає, що, знаючи зображення $X(p)$ вхідного сигналу $x(t)$ та зображення $Y(p)$ реакції системи $y(t)$ на цей сигнал, передаточну функцію можна отримати, взявши їх відношення, тобто

$$W(p) = \frac{Y(p)}{X(p)}. \quad (1.40)$$

Варто нагадати, що обернене перетворення Лапласа

$$L^{-1}\{F(p)\} = f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(p) e^{pt} dp, \quad (1.41)$$

згідно з яким за відомим зображенням $F(p)$ можна знайти оригінал $f(t)$, як правило, використовується лише для побудови таблиць відповідності між $f(t)$ та $F(p)$, а в практиці аналізу частіше використовуються формули розкладання, одна з яких – для некратних полюсів p_i зображення

$$Y(p) = \frac{C(p)}{D(p)}, \quad (1.42)$$

де $C(p)$, $D(p)$ – багаточлени за степенями p порядків m та n , відповідно, – має вигляд

$$y(t) = \sum_{i=1}^n \frac{C(p_i)}{D'(p_i)} e^{p_i t}. \quad (1.43)$$

Нагадаємо, що p_i – це корені рівняння

$$D(p) = 0, \quad (1.44)$$

а

$$D'(p_i) = \left. \frac{dD}{dp} \right|_{p=p_i}. \quad (1.45)$$

Наведемо приклад використання формули розкладання (1.43). Нехай маємо зображення невідомого нам оригіналу у вигляді

$$F(p) = \frac{2p+1}{p^3+5p^2+6p}. \quad (1.46)$$

Треба визначити його оригінал $f(t)$.

Очевидно, що для нашого прикладу:

$$\begin{aligned} C(p) &= 2p+1, \\ D(p) &= p^3+5p^2+6p, \\ \frac{dD}{dp} &= 3p^2+10p+6. \end{aligned} \quad (1.47)$$

Знайдемо полюси зображення (1.46), тобто корені рівняння

$$p^3+5p^2+6p=0. \quad (1.48)$$

Привівши рівняння (1.48) до вигляду

$$p(p^2+5p+6)=0, \quad (1.49)$$

легко бачити, що полюсами зображення (1.46) є:

$$\begin{cases} p_1 = 0, \\ p_2 = -2, \\ p_3 = -3. \end{cases} \quad (1.50)$$

Підставляючи (1.47) та (1.50) у формулу розкладання (1.43), отримаємо:

$$\begin{aligned} f(t) &= \frac{C(p_1)}{D'(p_1)} e^{p_1 t} + \frac{C(p_2)}{D'(p_2)} e^{p_2 t} + \frac{C(p_3)}{D'(p_3)} e^{p_3 t} = \\ &= \frac{2 \cdot (0) + 1}{3 \cdot (0)^2 + 10 \cdot (0) + 6} e^{0 \cdot t} + \frac{2 \cdot (-2) + 1}{3 \cdot (-2)^2 + 10 \cdot (-2) + 6} e^{-2t} + \\ &+ \frac{2 \cdot (-3) + 1}{3 \cdot (-3)^2 + 10 \cdot (-3) + 6} e^{-3t} = \frac{1}{6} + \frac{3}{2} e^{-2t} - \frac{5}{3} e^{-3t}. \end{aligned}$$

Функція

$$f(t) = \frac{1}{6} + \frac{3}{2} e^{-2t} - \frac{5}{3} e^{-3t} \quad (1.51)$$

і є оригіналом зображення виразу (1.46). Ще раз нагадуємо, що оригінал визначається лише при значеннях $t \geq 0$.

А наступну, більш складну, задачу сформулюємо так: нехай математична модель динамічної системи у формі передаточної функції має вигляд

$$W(p) = \frac{2p+4}{p^2+7p+12}. \quad (1.52)$$

На вхід цієї системи подається сигнал

$$x(t) = t. \quad (1.53)$$

Необхідно визначити реакцію $y(t)$ цієї динамічної системи на даний вхідний сигнал.

Розв'язуватимемо цю задачу з використанням комп'ютерної програми, складеної на мові Python.

Приклад № 5, присвячений визначенню реакції $y(t)$ динамічної системи з передаточною функцією (1.52) на вхідний сигнал (1.53) з використанням комп'ютерної програми, складеної на мові Python в рамках **ППП sympy**

In [1]: import sympy	# Виклик ППП sympy
In [2]: from sympy import *	# Виклик усіх функцій sympy
In [3]: t = symbols('t')	# Символізація змінної t
In [4]: p = symbols('p')	# Символізація змінної p
In [5]: x = Function('x')(t)	# Символізація функції x(t)
In [6]: y = Function('y')(t)	# Символізація функції y(t)
In [7]: C = Function('C')(p)	# Символізація функції C(p)

```

In [8]: D = Function ('D')(p)
In [9]: W = Function ('W')(p)
In [10]: X = Function ('X')(p)
In [11]: Y = Function ('Y')(p)
In [12]: D1 = Function ('D1')(p)
In [13]: Y1 = Function ('Y1')(p,t)
In [14]: Y2 = Function ('Y2')(p,t)
In [15]: C = 2*p+4
In [16]: D = p**2+7*p+12
In [17]: W = C/D, W
Out[17]:
(2*p + 4)/(p**2 + 7*p + 12)
In [18]: x = t
In [19]: x1 = x*exp(-p*t)

In [20]: X1 = Function ('X1')(p)
In [21]: X1 = integrate (x1,(t,0,oo));X1
Out[21]:
Piecewise((p**(-2), Abs(arg(p)) < pi/2),
(Integral(t*exp(-p*t), (t, 0, oo)), True))
In [22]: K=(p**(-2), Abs(arg(p)) < pi/2),
(Integral(t*exp(-p*t), (t, 0, oo)), True)
In [23]: X2=Function ('X2')(p)
In [24]: X2 = K[0][0], X2
Out[24]:
p**(-2)
In [25]: Y = W*X2, Y
Out[25]:
(2*p + 4)/(p**2*(p**2 + 7*p + 12))
In [26]: C1 = Function ('C1')(p)
In [27]: expr = Y
In [28]: C1, D1 = fraction (expr)
In [29]: D2 = Function ('D2')(p)
In [30]: D2 = D1.diff(p)
In [31]: roots (Eq(D1,0), p)
Out[31]:
{-3: 1, -4: 1, 0: 2}
In [32]: d0 = { }
In [33]: d0["a"]=-3
In [34]: d0["b"]=-4
In [35]: d0["c"]=0
In [36]: p1,p2,p3 = symbols('p1 p2 p3')
In [37]: p1 = d0['a'], p1
Out[37]: -3
In [38]: p2 = d0['b'], p2
Out[38]: -4
In [39]: p3 = d0['c'], p3
Out[39]: 0

```

```

# Символізація функції D(p)
# Символізація функції W(p)
# Символізація функції X(p)
# Символізація функції Y(p)
# Символізація функції D1(p)
# Символізація функції Y1(p,t)
# Символізація функції Y2(p,t)
# Формування функції C(p)
# Формування функції D(p)
# Формування функції W(p)

# Внесення функції x(t)
# Формування ядра x1
інтеграла Лапласа
# Символізація функції X1(p)
# Інтегрування функції x1

# Інтеграл як складена функція
у формі кортежу
# Кортеж складеної функції

# Символізація функції X2(p)
#Обчислення оператора
Лапласа  $L\{x(t)\} = X2(p)$ 

# Визначення зображення Y(p)

# Символізація функції C1(p)
# Створення виразу expr
# Фракціонування виразу expr
# Символізація функції D2(p)
# D2(p) як похідна від D1(p)
# Визначення полюсів Y(p)
з рівняння D1(p)=0

# Створення пустого словника d0
# Внесення в d0 полюсів Y(p)
з ключами "a", "b", "c"

# Оголошення символічними p1,p2,p3
# Формування полюса p1
зображення Y(p)
# Формування полюса p2
зображення Y(p)
# Формування полюса p3
зображення Y(p)

```

```
In [40]: Y1 = C1*exp(p*t)/D2
```

```
In [41]: Y2 = diff(C1*(p-p3)**2*
exp(p*t)/D1,p)
```

```
In [42]: y = Y1.subs(p,p1)+\
Y1.subs(p,p2)+Y2.subs(p,p3), y
```

```
Out[42]:
```

```
t/3 - 1/36 - 2*exp(-3*t)/9 + exp(-4*t)/4
```

```
In [43]: p11 = plot(x,(t,0,2),show=\
False, line_color = 'c')
```

```
In [44]: p22 = plot(y,(t,0,2),show=\
False, line_color = 'r')
```

```
In [45]: p11.extend(p22)
```

```
In [46]: p11.show( )
```

```
# Формування складової функції  $y(t)$ ,
обумовленої простим полюсом  $Y(p)$ 
```

```
# Формування складової функції  $y(t)$ ,
обумовленої кратним полюсом  $Y(p)$ 
```

```
# Визначення  $y(t)$  як  $L^{-1}\{Y(p)\}$ 
за теоремою розкладу
```

```
# Побудова графіка  $x(t)$  без виведення
на екран
```

```
# Побудова графіка  $y(t)$  без виведення
на екран
```

```
# Поєднання на одному рисунку обох
графіків
```

```
# Виведення на екран рисунка з графіками
```

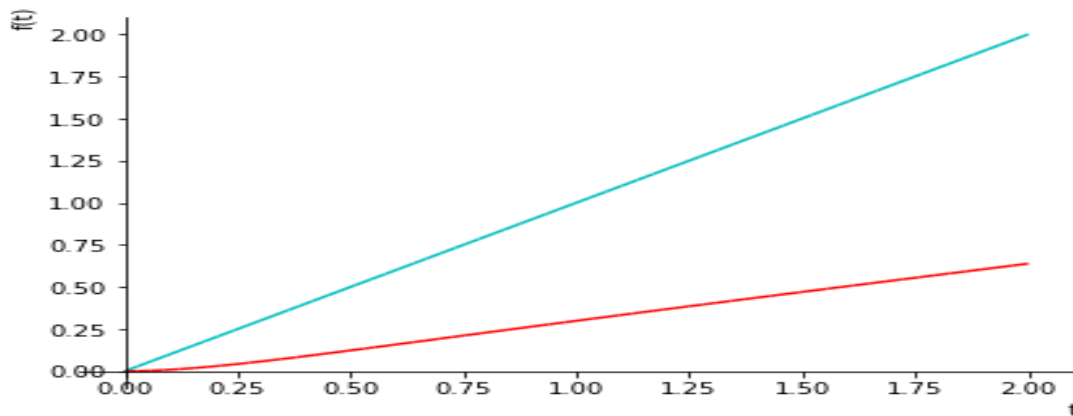


Рисунок 1.8 – Графіки вхідного сигналу $x(t)=t$, що діє на динамічну систему з заданою передаточною функцією $W(p)$, та її вихідного сигналу $y(t)$ на відрізок часу $t \in [0,2]$

1.3 Математичні моделі неперервних детермінованих ЛДС ЗП в частотній області

Як показано у попередньому підрозділі, достатньо повну інформацію про динаміку лінійної системи несе її передаточна функція, яка є функцією комплексної змінної $p = \sigma + j\omega$.

Якщо накласти умову $\sigma = 0$, то передаточна функція $W(p)$ вироджується у функцію уявної частини $j\omega$ комплексної змінної P , тобто в $W(j\omega)$.

Для прикладу, нехай

$$W_1(p) = \frac{10}{5p+1}. \quad (1.54)$$

За виконання умови $\sigma = 0$ із (1.54) маємо:

$$W_1(j\omega) = \frac{10}{1 + j5\omega}. \quad (1.55)$$

Функцію $W_1(j\omega)$ шляхом очевидних перетворень можна привести до такої форми

$$\begin{aligned} W_1(j\omega) &= \frac{10 \cdot (1 - j5\omega)}{(1 + j5\omega) \cdot (1 - j5\omega)} = \frac{10 - j50\omega}{1 + (5\omega)^2} = \\ &= \frac{10}{1 + 25\omega^2} - j \frac{50\omega}{1 + 25\omega^2} = P_1(\omega) + jQ_1(\omega), \end{aligned} \quad (1.56)$$

де

$$P_1(\omega) = \frac{10}{1 + 25\omega^2}, \quad (1.57)$$

$$Q_1(\omega) = -\frac{50\omega}{1 + 25\omega^2}. \quad (1.58)$$

Вирази (1.55) – (1.58) за своєю структурою справедливі для довільної функції $W(p)$. Оскільки як функція $W(j\omega)$, так і її складові $P(\omega)$ та $Q(\omega)$ у своїй структурі містять всі коефіцієнти передаточної функції $W(p)$, то можна стверджувати, що кожна з них несе однакову кількість інформації про динаміку системи, яка характеризується цією передаточною функцією. У цьому легко переконатись на наведеному прикладі.

З теорії комплексних чисел відомо, що при $\omega = \omega^*$ комплексне число

$$W(j\omega^*) = P(\omega^*) + jQ(\omega^*) \quad (1.59)$$

можна записати не тільки в алгебраїчній формі (1.59), а й у показниковій:

$$W(j\omega^*) = A(\omega^*) \cdot e^{j\varphi(\omega^*)}, \quad (1.60)$$

де

$$A(\omega^*) = \sqrt{P^2(\omega^*) + Q^2(\omega^*)}, \quad (1.61)$$

$$\varphi(\omega^*) = \arctg \frac{Q(\omega^*)}{P(\omega^*)}. \quad (1.62)$$

Оскільки параметр ω в усіх наведених вище виразах має чітко визначений фізичний сенс – це кругова частота, то всі функції, що залежать від нього, називаються частотними характеристиками, з яких:

$P(\omega)$ – дійсна частотна характеристика,

$Q(\omega)$ – уявна частотна характеристика,

$A(\omega)$ – амплітудна частотна характеристика,

$\varphi(\omega)$ – фазова частотна характеристика,

$W(j\omega)$ – амплітудно-фазова частотна характеристика.

Для співвідношень (1.59) – (1.62), які пов'язують між собою частотні характеристики, слід додати ще два, які є очевидними для всіх, хто знайомий з теорією комплексних чисел, а саме:

$$P(\omega) = A(\omega) \cdot \cos \varphi(\omega), \quad (1.63)$$

$$Q(\omega) = A(\omega) \cdot \sin \varphi(\omega). \quad (1.64)$$

Ввівши умову $\sigma = 0$, від перетворення за Лапласом, яке кожній функції, визначеній на часовій осі як оригінал, ставить у відповідність функцію на комплексній площині, переходимо до перетворення Фур'є, яке відображає оригінали на частотну вісь комплексної площини. Саме такий зміст вкладено у назву підрозділу, що впливає із всього викладеного вище.

А на завершення цього підрозділу у прикладі № 6 складемо комп'ютерну програму на мові Python для визначення частотних характеристик ЛДС ЗП з передаточною функцією

$$W(p) = \frac{4p^2 + p + 5}{p^3 + 2p^2 + 5p + 1}, \quad (1.65)$$

в якій комплексна змінна p задається виразом $p = u + jv$, де $j = \sqrt{-1}$ – уявна одиниця.

Отже, далі подаємо як **приклад № 6** комп'ютерну програму визначення частотних характеристик ЛДС ЗП з передаточною функцією (1.65), складену на мові Python в рамках *IIII sympy*

```
In [1]: import sympy
In [2]: from sympy import*
In [3]: p = symbols('p')
In [4]: a = symbols('a:3');a
Out[4]: (a0, a1, a2)
In [5]: b = symbols('b:4');b
Out[5]: (b0, b1, b2, b3)
In [6]: f1 = Function('f1')(p)
In [7]: f2 = Function('f2')(p)
In [8]: W = Function('W')(p)
In [9]: f1 = sum([a[i]*p**i for i in
range(3)]);f1
Out[9]:
a0 + a1*p + a2*p**2
In [10]: f2 = sum([b[i]*p**i for i in
range(4)]);f2
Out[10]:
b0 + b1*p + b2*p**2 + b3*p**3
In [11]: W = f1/f2;W
Out[11]:
(a0 + a1*p + a2*p**2)/(b0 + b1*p +
b2*p**2 + b3*p**3)
In [12]: f11 = Function('f11')(p)
In [13]: f22 = Function('f22')(p)
In [14]: f11 = f1.subs([(a[0],5),(a[1],\
1),(a[2],4)]);f11
Out[14]:
4*p**2 + p + 5
In [15]: f22 = f2.subs([(b[0],1),(b[1],\
5),(b[2],2),(b[3],1)]);f22
Out[15]:
p**3 + 2*p**2 + 5*p + 1
In [16]: W = f11/f22;W
Out[16]:
```

```

(4*p**2 + p + 5)/(p**3 + 2*p**2 + \
5*p + 1
In [17]: v = symbols('v',real=True)
In [18]: f111 = Function('f111')(v)
In [19]: f222 = Function('f222')(v)
In [20]: expr1=f11
In [21]: f111 = expr1.subs(p,v*I);\
f111
Out[21]:
-4*v**2 + I*v + 5
In [22]: expr2=f22
In [23]: f222 = expr2.subs(p,v*I);\
f222
Out[23]:
-I*v**3 - 2*v**2 + 5*I*v + 1
In [24]: W1 = Function('W1')(v)
In [25]: W1 =f111/f222;W1
Out[25]:
(-4*v**2 + I*v + 5)/(-I*v**3 - \
2*v**2 +5*I*v + 1)
In [26]: f1111 = Function('f1111')(v)
In [27]: f1111 = collect(f111,I);f1111
Out[27]:
-4*v**2 + I*v + 5
In [28]: f2222 = Function('f2222')(v)
In [29]: f2222 = collect(f222,I);f2222
Out[29]:
-2*v**2 + I*(-v**3 + 5*v) + 1
In [30]: h2222 = conjugate(f2222);\
h2222
Out[30]:
-2*v**2 - I*(-v**3 + 5*v) + 1
In [31]: fh = Function('fh')(v)
In [32]: fh = f2222*h2222;fh
Out[32]:
(-2*v**2 - I*(-v**3 + 5*v) + 1)*\
(-2*v**2 + I*(-v**3 + 5*v) + 1)
In [33]: ffhh = Function('ffhh')(v)
In [34]: ffhh = expand(fh);ffhh
Out[34]:
v**6 - 6*v**4 + 21*v**2 + 1
In [35]: fhf = Function('fhf')(v)
In [36]: fhf = f1111*h2222;fhf
Out[36]:

```

```

(-4*v**2 + I*v + 5)*(-2*v**2 - \
I*(-v**3 + 5*v) + 1)
In [37]: fhfh = Function('fhfh')(v)
In [38]: fhfh = expand (fhf);fhfh
Out[38]:
-4*I*v**5 + 7*v**4 + 23*I*v**3 - \
9*v**2 - 24*I*v + 5
In [39]: fhhf = Function('fhhf')(v)
In [40]: fhhf = collect (fhfh,I);fhhf
Out[40]:
7*v**4 - 9*v**2 + I*(-4*v**5 + \
23*v**3 - 24*v) + 5
In [41]: W1 = fhhf/ffhh;W1
Out[41]:
(7*v**4 - 9*v**2 + I*(-4*v**5 + \
23*v**3 - 24*v) + 5)/(v**6 - \
6*v**4 + 21*v**2 + 1)
In [42]: D = Function('D')(v)
In [43]: D = re(W1);D
Out[43]:
(7*v**4 - 9*v**2 + 5)/(v**6 - \
6*v**4 + 21*v**2 + 1)
In [44]: U = Function('U')(v)
In [45]: U = im(W1);U
Out[45]:
(-4*v**5 + 23*v**3 - 24*v)/(v**6 \
-6*v**4 +21*v**2 + 1)
In [46]: A = Function('A')(v)
In [47]: A = Abs(W1);A
Out[47]:
sqrt(16*v**10 - 135*v**8 + \
595*v**6 - 953*v**4 + 486*v**2 \
+25)*Abs(1/(v**6 - 6*v**4 + \
21*v**2 + 1))
In [48]: Faz = Function('Faz')(v)
In [49]: Faz = arg(W1);Faz
Out[49]:
arg((7*v**4 - 9*v**2 + I*v*(-4*\
v**4 + 23*v**2 - 24) + 5)/(v**6 \
-6*v**4 + 21*v**2 + 1))
In [50]: Faz = atan(U/D);Faz
Out[50]:
atan((-4*v**5 + 23*v**3 - \
24*v)/(7*v**4 -9*v**2 + 5))

```

```

In [51]: from sympy.plotting import plot
In [52]: plot(A,(v,0,10))
Out[52]: <sympy.plotting.plot.Plot at 0x
15bd2205310>

```

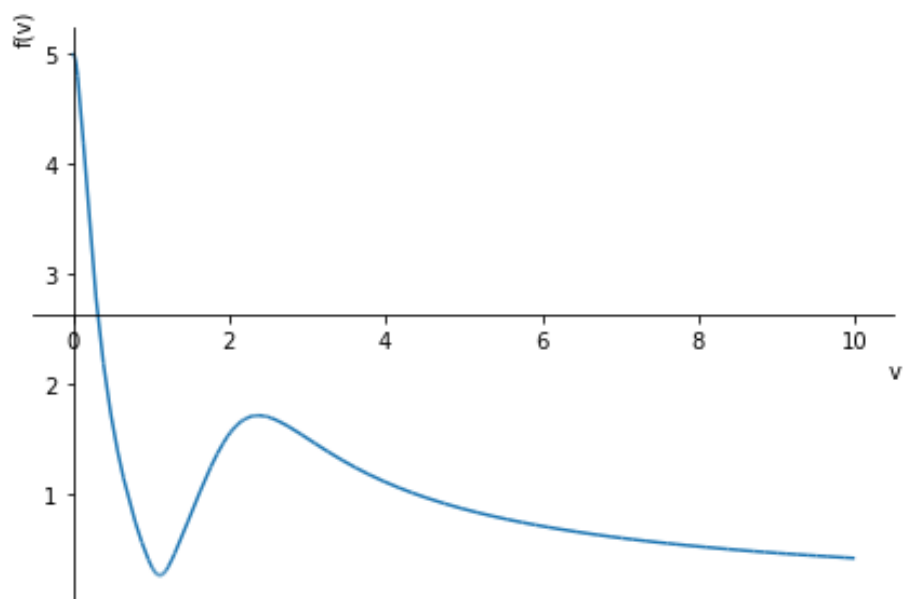


Рисунок 1.9 – Графік амплітудної частотної характеристики ЛДС ЗП

```
In [53]: plot(Faz,(v,0,10))
Out[53]: <sympy.plotting.plot.Plot at 0x15bd2205760>
```

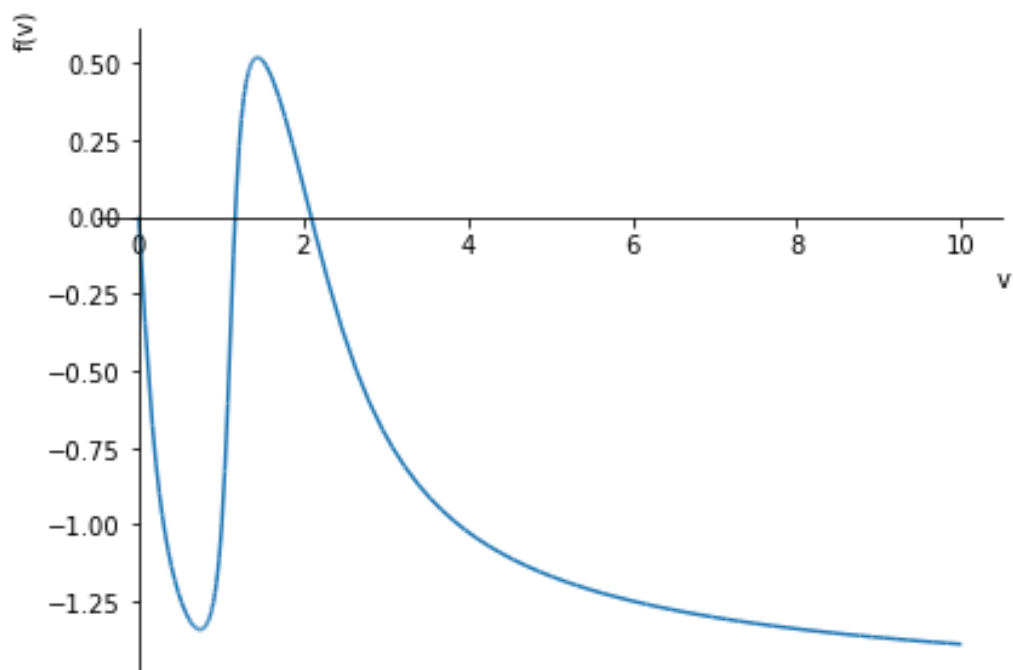


Рисунок 1.10 – Графік фазової частотної характеристики ЛДС ЗП

1.4. Ідентифікація моделей неперервних детермінованих ЛДС ЗП з використанням експериментальних даних

Із виразу (1.40), приведеного вище, витікає, що математичну модель ЛДС ЗП у вигляді передаточної функції $W(p)$ можна визначити, поділивши перетворену за Лапласом реакцію системи $y(t)$ на перетворений за Лапласом вхідний сигнал $x(t)$. Цілком очевидно, що до перетворення за Лапласом і експериментально зафіксований вхідний сигнал $x(t)$, і експериментально зафіксовану реакцію $y(t)$ системи на цей сигнал потрібно апроксимувати відповідними функціями аргументу t . Згідно з теоремою Вейерштрасса майже завжди це можна зробити за допомогою поліномів за степенями аргументу t , перетворення за Лапласом яких приводить до відношення поліномів за степенями аргументу p .

Особливо просто задача ідентифікації такої системи за цим алгоритмом буде розв'язуватись, якщо вхідним сигналом $x(t)$ динамічної системи є одиничний стрибок $1(t)$

$$x(t) = 1(t) = \begin{cases} 1, & t \geq 0, \\ 0, & t < 0, \end{cases} \quad (1.66)$$

або одиничний імпульсний вхідний сигнал $x(t)$ у вигляді дельта-функції $\delta(t)$, для якої справедливо:

$$\delta(t) = \begin{cases} \infty, & t = 0, \\ 0, & t \neq 0, \end{cases} \quad (1.67)$$

$$\int_{-\infty}^{\infty} \delta(t) dt = \int_0^{\infty} \delta(t) dt = 1, \quad (1.68)$$

з перетворених за Лапласом реакцій системи на кожен із яких у вигляді відповідно перехідної $h(t)$ та імпульсної перехідної $g(t)$ характеристик, одразу отримуємо передаточну функцію згідно зі співвідношеннями, для отримання яких розглянемо ці перехідну та імпульсну перехідну характеристики системи більш детально.

Перехідна характеристика $h(t)$ системи є її реакцією на вхідний сигнал $x(t)$ у вигляді одиничного стрибка (1.66). У цьому випадку маємо:

$$y(t) = h(t). \quad (1.69)$$

Графічна інтерпретація цього означення показана на рис. 1.11.

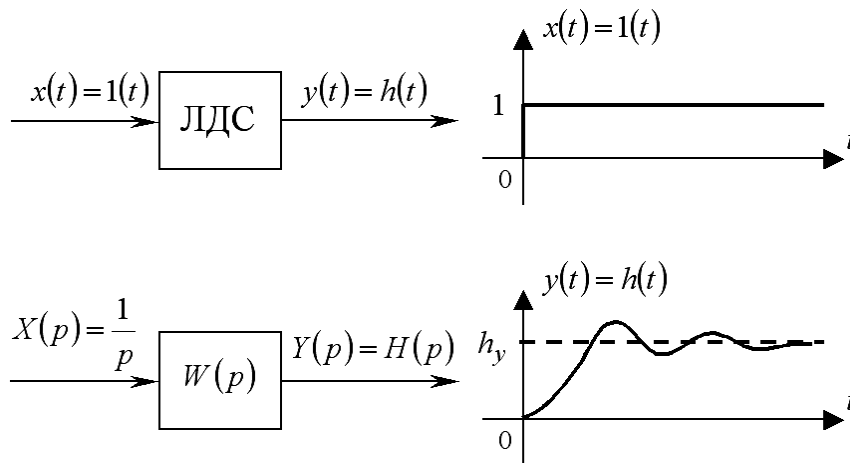


Рисунок 1.11 – Графік реакції $h(t)$ лінійної динамічної системи ЛДС на одиничний стрибок $1(t)$

Імпульсна перехідна або вагова характеристика $g(t)$ системи є її реакцією на одиничний імпульсний вхідний сигнал $x(t)$ у вигляді дельта-функції $\delta(t)$, для якої справедливими є вирази (1.67), (1.68), із яких випливає, що дельта-функція – це ідеалізація імпульсу одиничної площі з надзвичайно великою висотою і надзвичайно малою протяжністю, як показано на рис. 1.12.

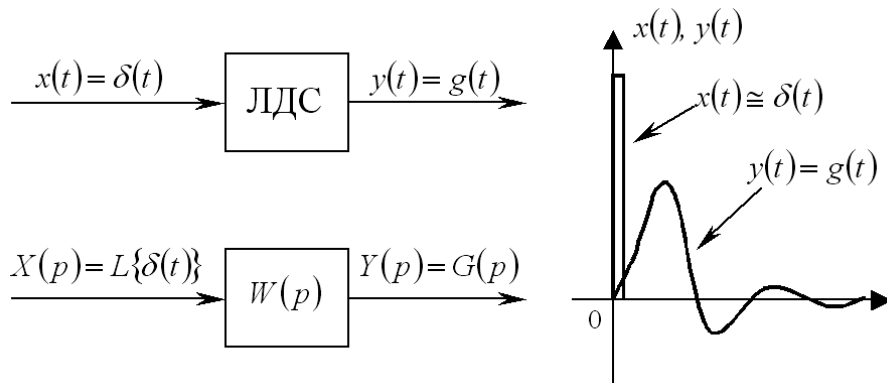


Рисунок 1.12 – Графік реакції $g(t)$ лінійної динамічної системи ЛДС на одиничний імпульс $\delta(t)$

Дуже важливим є те, що сигнал $x(t)$, який діє на вході лінійної динамічної системи з імпульсною перехідною характеристикою $g(t)$ (рис. 1.13), та реакція системи $y(t)$ на цей сигнал пов'язані між собою інтегралом згортки

$$y(t) = \int_0^{\infty} x(t-\tau) g(\tau) d\tau, \quad (1.70)$$

який має надзвичайно прозорий зміст – вихідний сигнал динамічної системи формується сумою реакцій на кожний імпульс вхідного сигналу під час подання цього вхідного сигналу у вигляді послідовності імпульсів з висотою, що дорівнює значенню вхідного сигналу у відповідний момент часу. Цілком

очевидно, що вираз (1.70) теж є математичною моделлю ЛДС ЗП, але в класі інтегральних рівнянь

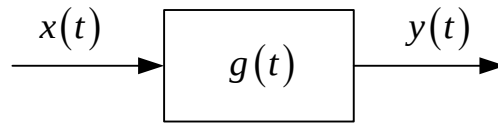


Рисунок 1.13 – Узагальнена структурна схема лінійної динамічної системи

Зв'яжемо передаточну функцію $W(p)$ ЛДС ЗП з її перехідною $h(t)$ та імпульсною перехідною $g(t)$ характеристиками.

Оскільки згідно зі співвідношенням (1.38) зображення за Лапласом вихідного сигналу $Y(p)$ лінійної динамічної системи є добутком передаточної функції цієї системи $W(p)$ та зображення за Лапласом вхідного сигналу системи $X(p)$, яке в разі виконання (1.26), згідно з (1.27), дорівнюватиме $\frac{1}{p}$, то для зображення за Лапласом перехідної характеристики $H(p)$ матимемо:

$$H(p) = \frac{W(p)}{p}, \quad (1.71)$$

що, в свою чергу, дає нам право записати

$$h(t) = L^{-1} \left\{ \frac{W(p)}{p} \right\}. \quad (1.72)$$

Легко бачити, що зображення за Лапласом дельта-функції дорівнює

$$L\{\delta(t)\} = \int_0^{\infty} \delta(t) e^{-pt} dt = e^{-p \cdot (t=0)} \int_0^{\infty} \delta(t) dt = e^{-p \cdot 0} \cdot 1 = 1. \quad (1.73)$$

Тож, якщо при

$$x(t) = \delta(t), \quad (1.74)$$

за визначенням

$$y(t) = g(t), \quad (1.75)$$

то із (1.38) випливає, що

$$G(p) = W(p). \quad (1.76)$$

Тобто передаточна функція лінійної динамічної системи є зображенням за Лапласом її імпульсної перехідної характеристики, і навпаки

$$g(t) = L^{-1}\{W(p)\}. \quad (1.77)$$

Із співвідношень (1.71) і (1.76) випливає, що

$$G(p) = p \cdot H(p). \quad (1.78)$$

А це, в свою чергу, означає, що в області оригіналів справедливим є рівняння

$$g(t) = \frac{dh(t)}{dt}, \quad (1.79)$$

тобто, що імпульсну перехідну характеристику $g(t)$ системи можна отримати диференціюванням її перехідної характеристики $h(t)$.

Аналізуючи вирази (1.74)-(1.78), бачимо, що для розв'язання задачі ідентифікації моделі ЛДС ЗП у формі передаточної функції необхідно на вхід цієї системи подати або сигнал у вигляді одиничного стрибка, або у вигляді одиничного імпульсу, зафіксувати реакцію системи на ці сигнали у вигляді перехідної або імпульсної перехідної характеристик і перетворити їх за Лапасом згідно з виразом (1.24), а потім застосувати співвідношення (1.76), (1.78).

А далі покажемо, як, отримавши передаточну функцію $W(p)$ ЛДС ЗП, побудувати математичну модель цієї системи у вигляді диференціального рівняння.

Нехай математична модель ЛДС ЗП на комплексній площині має вигляд

$$W(p) = \frac{b_m p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0}{a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0}. \quad (1.80)$$

Підставивши значення $W(p)$ із (1.80) в (1.38) та перенісши знаменник в ліву частину рівності, отримаємо

$$\begin{aligned} (a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0) \cdot Y(p) = \\ = (b_m p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0) \cdot X(p). \end{aligned} \quad (1.81)$$

Розкриваючи дужки в рівності (1.81) і враховуючи те, що перемноженню зображення на p відповідає диференціювання оригіналу за t , приходимо до диференціального рівняння (1.4).

А далі продемонструємо, як можна здійснити ідентифікацію ЛДС ЗП з використанням її амплітудної $A(\omega)$ і фазової $\varphi(\omega)$ частотних характеристик, які легко отримати експериментально за допомогою спеціального комплексу, який випускається серійно і містить у собі генератор синусоїдальних коливань змінної частоти, подвійний піковий вольтметр, фазометр-частотомір та пристрій для перетворення фізичних параметрів процесів на вході та виході системи в електричний сигнал і для їх узгодження. Зазначимо, що амплітудно-фазову частотну характеристику з експерименту можна побудувати тільки для стійких об'єктів.

Експеримент з отримання частотних характеристик $A(\omega)$ та $\varphi(\omega)$ полягає в тому, що на вхід системи від генератора синусоїдальних коливань подаємо сигнал

$$x_1(t) = X_1(\omega_1) \sin \omega_1 t \quad (1.82)$$

з фіксованими частотою $\omega_1 = 2\pi f_1$, де f_1 – реальна частота у герцах, та амплітудою $X_1(\omega_1)$.

Після закінчення перехідного процесу, тривалість якого визначається чотирма еквівалентними сталими часу цієї системи, на її виході встановлюються коливання

$$y_1(t) = Y_1(\omega_1) \cdot \sin(\omega_1 t + \varphi(\omega_1)). \quad (1.83)$$

Замірявши подвійним піковим вольтметром одночасно значення амплітуди $X_1(\omega_1)$ сигналу на вході системи та $Y_1(\omega_1)$ на її виході та взявши їх відношення, отримаємо значення амплітудної частотної характеристики системи на частоті ω_1 , тобто

$$A(\omega_1) = \frac{Y_1(\omega_1)}{X_1(\omega_1)}. \quad (1.84)$$

За допомогою фазометра-частотоміра на цій ж частоті ω_1 заміряємо зсув фаз $\varphi(\omega_1)$, який виникає між сигналами $x_1(t)$ та $y_1(t)$ – він безпосередньо задає значення фазової частотної характеристики $\varphi(\omega_1)$ на частоті ω_1 .

Дискретно змінюючи частоту вхідного сигналу від нуля до значень ω_n , за яких можна вважати, що

$$Y(\omega_n) \cong 0, \quad (1.85)$$

і здійснюючи дії, що описані вище, отримаємо скінченну множину чисел

$$\{(\omega_0, A(\omega_0), \varphi(\omega_0)), (\omega_1, A(\omega_1), \varphi(\omega_1)), \dots, (\omega_n, A(\omega_n), \varphi(\omega_n))\}, \quad (1.86)$$

яка дискретно задає амплітудну та фазову частотні характеристики $A(\omega)$ і $\varphi(\omega)$ системи.

Для отримання неперервних функцій $A(\omega)$ та $\varphi(\omega)$ залишається тільки здійснити апроксимацію послідовностей чисел

$$A(\omega_0), A(\omega_1), \dots, A(\omega_n), \quad (1.87)$$

$$\varphi(\omega_0), \varphi(\omega_1), \dots, \varphi(\omega_n) \quad (1.88)$$

за допомогою відомих інтерполяційних процедур, або використавши спеціально підібрані формули для $A(\omega)$ і $\varphi(\omega)$ та суму квадратів відхилень як критерій оптимізації процедури.

У разі використання першого шляху для отримання $W(j\omega)$ необхідно підставити інтерполяційні вирази для $A(\omega)$ та $\varphi(\omega)$ у формулу (1.60), а потім за допомогою іншої стандартної процедури перейти від $W(j\omega)$ до $W(p)$.

Другий шлях дозволяє одразу отримати формулу для $W(p)$.

Алгоритм синтезу $W(p)$ за цим підходом такий: спочатку висуваємо гіпотезу про те, що динаміка системи, яка розглядається, описується передаточною функцією

$$W_1(p) = \frac{b_1}{a_1 p + 1}. \quad (1.89)$$

Із формули (1.89) видно, що для ідентифікації системи необхідно знайти лише два коефіцієнти a_1 та b_1 .

Використовуючи формули (1.59) – (1.62) та (1.89), знаходимо, що

$$W_1(j\omega) = \frac{b_1}{a_1 j\omega + 1} = \frac{b_1 \cdot (1 - a_1 j\omega)}{(1 + a_1 j\omega) \cdot (1 - a_1 j\omega)}, \quad (1.90)$$

$$P_1(\omega) = \frac{b_1}{1 + a_1^2 \omega^2}, \quad (1.91)$$

$$Q_1(\omega) = -\frac{a_1 b_1 \omega}{1 + a_1^2 \omega^2}, \quad (1.92)$$

$$A_1(\omega) = \frac{b_1}{\sqrt{1 + a_1^2 \omega^2}}, \quad (1.93)$$

$$\varphi_1(\omega) = -\arctg(a_1 \omega). \quad (1.94)$$

Підставляючи у формулу (1.93) значення $\omega_0, A(\omega_0); \omega_1, A(\omega_1)$, чи якісь інші дві пари із (1.86), складаємо систему двох рівнянь з двома невідомими a_1, b_1 , розв'язавши яку чисельними методами, отримаємо оцінки коефіцієнтів a_1, b_1 .

А використовуючи значення $\omega_0, \varphi(\omega_0); \omega_1, \varphi(\omega_1) \dots \omega_n, \varphi(\omega_n)$ із (1.88) та значення коефіцієнта a_1 , знайденого на попередньому етапі, вчислимо оцінку

$$\sigma_1^2 = \sum_{i=0}^n (\varphi(\omega_i) + \arctg(a_1 \omega_i))^2 \quad (1.95)$$

суми квадратів відхилень експериментальних значень фазової частотної характеристики від її значень, що визначаються за вибраною моделлю (1.89).

Після цього висуваємо гіпотезу, що динаміка системи описується якоюсь іншою передаточною функцією і для її конкретного виду повторюємо всю описану вище процедуру.

У результаті отримаємо оцінку σ_2^2 .

Якщо

$$\sigma_2^2 > \sigma_1^2, \quad (1.96)$$

то найкращу апроксимацію передаточної функції системи за експериментальними даними (1.87), (1.88) забезпечує формула (1.89). Якщо ж

$$\sigma_2^2 < \sigma_1^2, \quad (1.97)$$

то пошук треба продовжувати.

Дві практичні поради:

1). Якщо умова (1.85) виконується, то це означає, що порядок m полінома у чисельнику передаточної функції системи менший за порядок n полінома в її знаменнику.

У цьому легко переконатись, проаналізувавши вираз для $A(\omega)$, записаний у загальному вигляді.

2). а). Якщо в (1.88) є таке ω_m , для якого

$$\varphi(\omega_m) \rightarrow -\frac{\pi}{2}, \quad (1.98)$$

то в формулі для передаточної функції (1.80) $n=1$, а згідно з умовою фізичної реалізуємості матимемо в (1.80) $m \leq 1$.

б). Якщо в (1.88) є таке ω_m , для якого

$$\varphi(\omega_m) \rightarrow -2 \cdot \frac{\pi}{2}, \quad (1.99)$$

то в (1.88) $n=2$, а $m \leq 2$

в). Якщо в (1.88) є таке ω_m , для якого

$$\varphi(\omega_m) \rightarrow -k \cdot \frac{\pi}{2}, \quad (1.100)$$

то в (1.88) $n=k$, а $m \leq k$.

Тож, отримавши експериментально значення множини (1.88), одразу можемо визначити порядок n системи і загальний вигляд знаменника її передаточної функції.

А оскільки для реальних систем завжди виконується умова фізичної реалізуємості $m \leq n$, то таке попереднє визначення порядку полінома знаменника передаточної функції суттєво скорочує кількість можливих варіантів в задачі синтезу моделі системи за даною процедурою.

1.5 Завдання для самоперевірки

За яких умов динамічна система може вважатись безперервною детермінованою лінійною динамічною системою з зосередженими параметрами?

Що собою являють математичні моделі ЛДС ЗП у просторі часу?

Що собою являє перетворення за Лапасом? Які головні переваги аналізу в області зображень?

Як визначити передаточну функцію системи, якщо відоме диференціальне рівняння, котре описує процеси у цій системі?

Як визначити оригінал за відомим його зображенням?

Побудуйте структурну схему ЛДС ЗП в області оригіналів і в області зображень.

Як, маючи передаточну функцію ЛДС ЗП, відтворити диференціальне рівняння, яке описує процеси у цій системі?

Яким чином, маючи передаточну функцію ЛДС ЗП, отримати частотні характеристики системи: дійсну, уявну, амплітудну, фазову, амплітудно-фазову?

Які частотні характеристики і за допомогою яких приладів можна знайти експериментально?

Як, маючи експериментально визначені окремі значення частотних характеристик, синтезувати передаточну функцію системи?

Як за експериментальними значеннями фазової частотної характеристики системи визначити її порядок?

Що собою являє простір змінних стану системи та як побудувати її модель у цьому просторі?

Дайте означення перехідної та імпульсної перехідної характеристик лінійної динамічної системи.

Що собою являє одиничний стрибок, і який графік він має?

Що собою являє одиничний імпульс і які його властивості ви знаєте?

Як зв'язані зображення за Лапасом перехідної та імпульсної перехідної характеристик системи з передаточною функцією цієї системи?

Доведіть, що імпульсна перехідна характеристика системи є похідною від її перехідної характеристики.

Які математичні умови фізичної реалізуємості ЛДС ЗП ви знаєте?

Як здійснити ідентифікацію ЛДС ЗП за експериментальними даними, отриманими в часовій області?

Як здійснити ідентифікацію ЛДС ЗП за експериментальними даними, отриманими в частотній області?

Розділ 2 МАТЕМАТИЧНІ МОДЕЛІ ДИСКРЕТНИХ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ

Як уже було відзначено раніше, до дискретних динамічних систем будемо відносити такі динамічні системи, в яких процеси формуються лише в окремі моменти часу хоча б в одному із елементів їх структури. Очевидно, що будь-яка лінійна автоматична система з комп'ютером у замкнутому контурі, який в принципі працює виключно лише з дискретними величинами, належить до класу дискретних динамічних систем.

Для аналізу процесів у дискретних системах нам потрібен математичний апарат, який виходить за рамки того, що вивчається у стандартному курсі вищої математики для студентів інженерних спеціальностей, а тому розпочнемо цей розділ викладенням основ цього апарату.

2.1 Решітчасті функції та скінченні різниці

Нехай $f(t)$ – неперервна функція, графік якої має вигляд, наведений на рис. 2.1, а.

Виберемо інтервал дискретності T і визначимо значення функції $f(t)$ лише при значеннях аргументу t , кратних T (рис. 2.1, б).

У результаті цих дій отримаємо функцію $f[kT]$ дискретного аргументу kT ($k = 0, 1, 2, \dots$) (рис. 2.1, в), яку математики домовились називати «*решітчастою*».

Записується алгоритм породження решітчастої функції $f[kT]$ із неперервної $f(t)$ у такий спосіб:

$$\begin{aligned} f[kT] &= f(t)|_{t=kT}, \\ T > 0, \quad k &= 0, 1, 2, \dots \end{aligned} \quad (2.1)$$

Нехай

$$f_1(t) = 2t - 4, \quad t \geq 0, \quad (2.2)$$

$$T = 0,5. \quad (2.3)$$

Згідно з алгоритмом (2.1) решітчаста функція $f_1[kT]$, яка породжується функцією (2.2) за умови (2.3), має вигляд:

$$\begin{aligned} f_1[kT] &= f_1(t)|_{t=0,5k} = 2 \cdot [0,5k] - 4, \\ k &= 0, 1, 2, \dots \end{aligned} \quad (2.4)$$

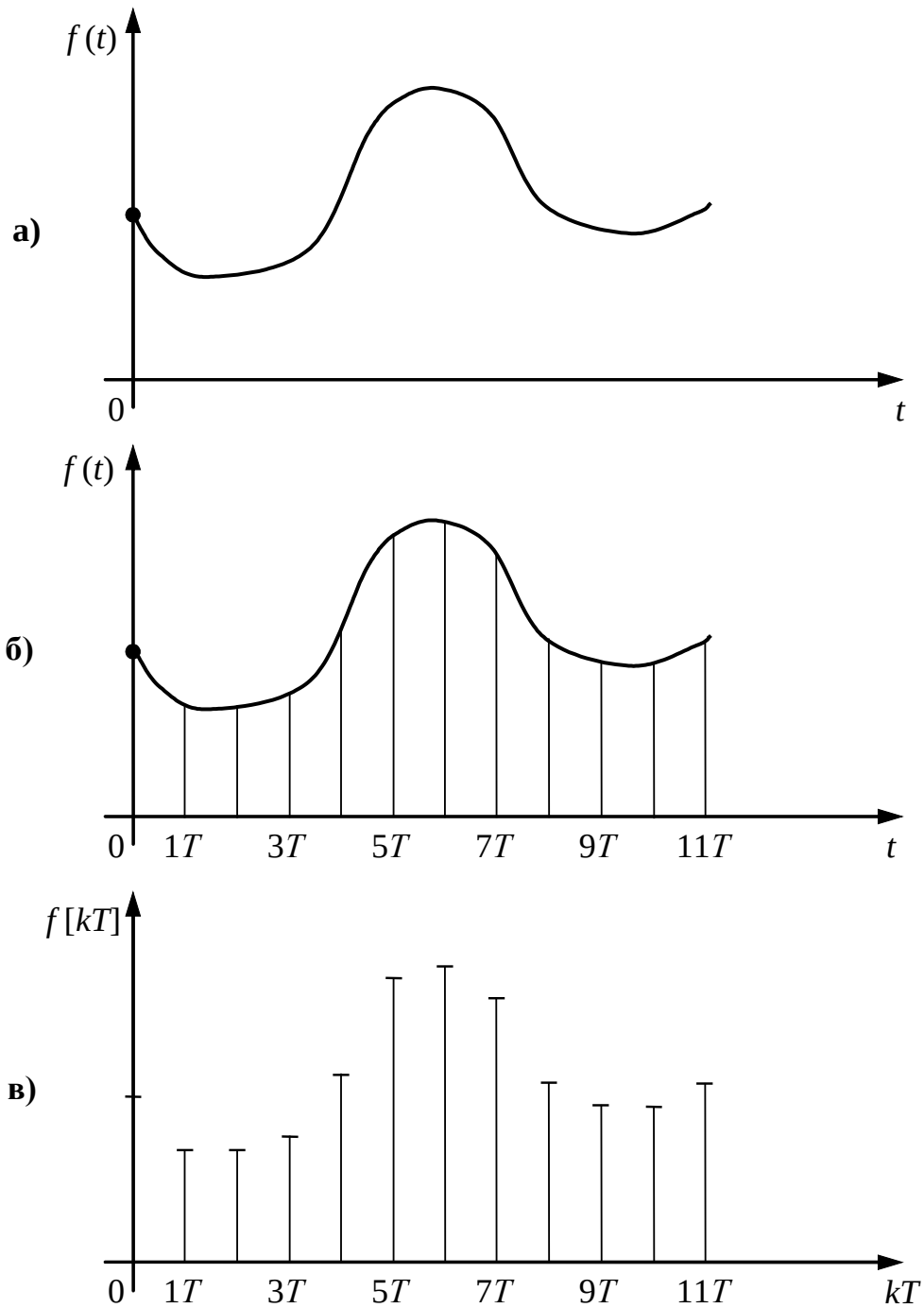


Рисунок 2.1 – Графіки неперервної $f(t)$ і породженої нею решітчастої $f[kT]$ функцій

А тепер нехай

$$f_2(t) = 3t^2 + 2, \quad t \geq 0, \quad (2.5)$$

$$T = 0,5. \quad (2.6)$$

Цій неперервній функції відповідає решітчаста, яка визначається у такий спосіб:

$$f_2[kT] = f_2(t)|_{t=0,5k} = 3 \cdot [0,5k]^2 + 2, \quad k = 0, 1, 2, \dots \quad (2.7)$$

Графіки решітчастих функцій (2.4), (2.7) наведені на рис. 2.2.

Із формул (2.4) та (2.7) легко бачити, що після вибору числового значення інтервалу дискретності T решітчаста функція $f[kT]$ стає функцією лише аргументу k , а тому автори багатьох книг записують її у вигляді не $f[kT]$, а $f[k]$. Роблячи так, слід пам'ятати, по-перше, що коефіцієнти у функціях $f[k]$ і $f[kT]$ не збігаються, а по-друге, що для відтворення їх породної неперервної функції $f(t)$ необхідно порядковий номер k , який виступає як аргумент решітчастої функції $f[k]$, помножити на числове значення інтервалу дискретності T до виконання операції над ним. Для того, щоб не тримати цього у пам'яті, будемо зберігати період дискретності T в аргументі решітчастої функції $f[kT]$.

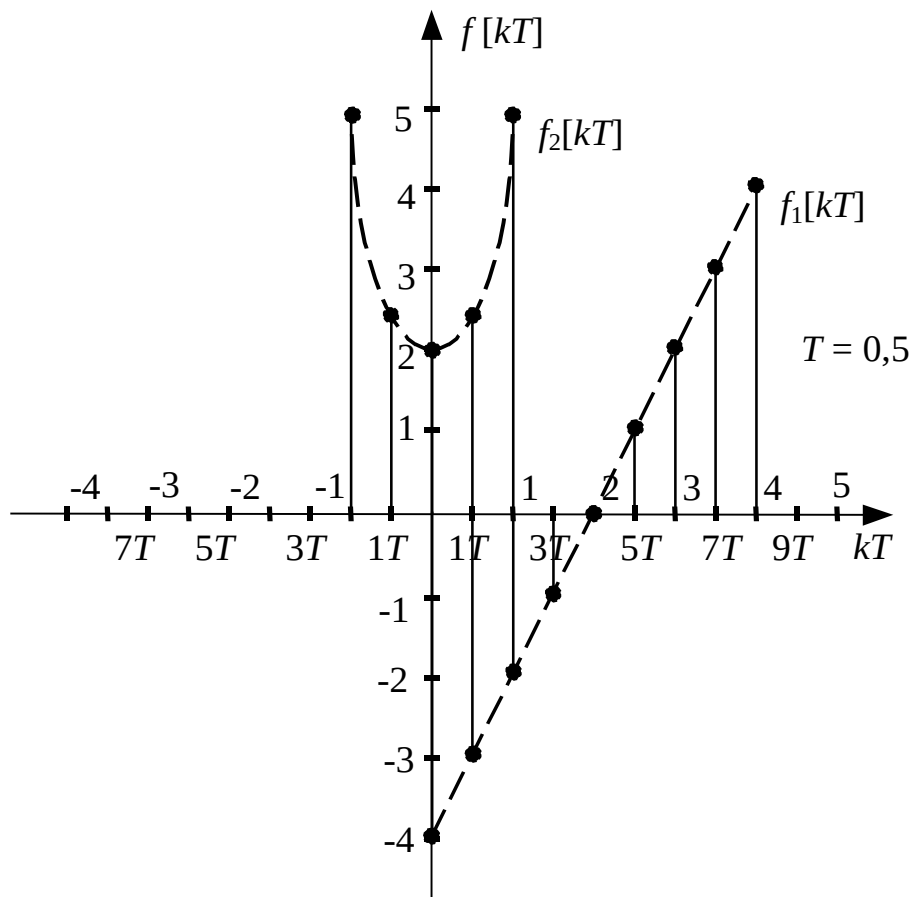


Рисунок 2.2 – Графіки решітчастих функцій $f_1[kT] = 2[0,5k] - 4$ та $f_2[kT] = 3[0,5k]^2 + 2$ при $T = 0,5$

Як відомо з курсу математичного аналізу, швидкість зміни неперервної функції $f(t)$ у кожній точці графіка характеризується значенням її похідної $\frac{df}{dt}$ у цій точці, числове значення якої дорівнює тангенсу кута α нахилу дотичної до графіка, проведеної через цю ж точку (рис. 2.3).

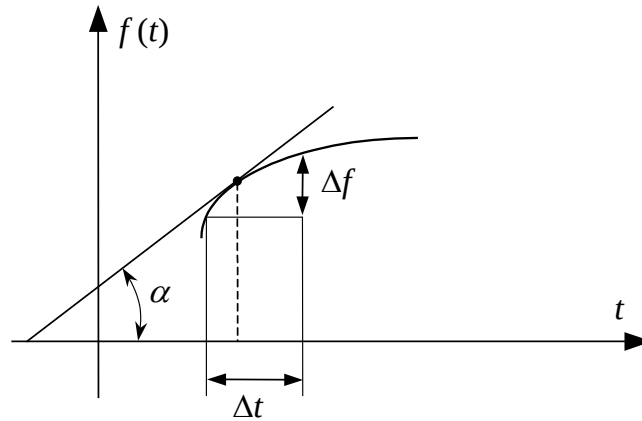


Рисунок 2.3 – Геометрична інтерпретація похідної $\dot{f}(t)$ від функції $f(t)$

Відомо, що похідна $\frac{df}{dt}$ є функцією $\dot{f}(t)$, яка визначається із співвідношення:

$$\dot{f}(t) = \frac{df}{dt} = \lim_{\Delta t \rightarrow 0} \frac{\Delta f}{\Delta t}, \quad (2.8)$$

яке не може мати місця для решітчастих функцій.

Але, оскільки сусідні значення аргументу kT решітчастої функції $f[kT]$ відрізняються на T , то, сформувавши різницю

$$\begin{aligned} \Delta f[kT] &= f[(k+1)T] - f[kT], \\ k &= 0, 1, 2, \dots, \end{aligned} \quad (2.9)$$

ми можемо формально використати цю різницю $\Delta f[kT]$ для характеристики швидкості зміни решітчастої функції $f[kT]$ у точці з аргументом kT .

Різницю $\Delta f[kT]$ називають *прямою скінченною різницею першого порядку* решітчастої функції $f[kT]$.

За аналогією з (2.9) можна визначити пряму скінченну різницю другого порядку

$$\begin{aligned} \Delta^2 f[kT] &= \Delta f[(k+1)T] - \Delta f[kT], \\ k &= 0, 1, 2, \dots \end{aligned} \quad (2.10)$$

Якщо у вираз (2.10) підставити значення $\Delta f[kT]$ та $\Delta f[(k+1)T]$, визначені за формулою (2.9), отримаємо:

$$\Delta^2 f[kT] = f[(k+2)T] - 2f[(k+1)T] + f[kT],$$

$$k = 0, 1, 2, \dots \quad (2.11)$$

Узагальнюючи (2.9) та (2.10), маємо:

$$\Delta^n f[kT] = \Delta^{n-1} f[(k+1)T] - \Delta^{n-1} f[kT],$$

$$n = 1, 2, \dots; \quad k = 0, 1, 2, \dots \quad (2.12)$$

При розкритті формули (2.12) для конкретних n варто пам'ятати, що

$$\begin{cases} \Delta^0 f[kT] = f[kT], \\ \Delta^0 f[(k+1)T] = f[(k+1)T]. \end{cases} \quad (2.13)$$

Відомо, що значення похідної $\dot{f}(t)$ від неперервної функції $f(t)$ у точці з аргументом t не залежить від того зліва чи справа наближається Δt до нуля (див. формулу (2.8)). Інша ситуація виникає при наближенні до точки з аргументом kT під час аналізу решітчастої функції $f[kT]$, а тому для характеристики швидкості її зміни крім прямої скінченної різниці $\Delta f[kT]$ першого порядку вводять ще й обернену скінченну різницю першого порядку за формулою

$$\nabla f[kT] = f[kT] - f[(k-1)T],$$

$$k = 1, 2, 3, \dots \quad (2.14)$$

Нагадаємо, що символи « Δ » та « ∇ » є грецькими літерами, відповідно, «дельта» та «набла».

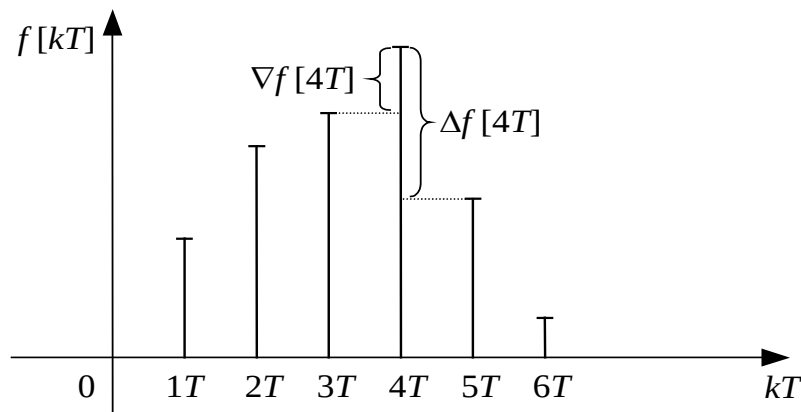


Рисунок 2.4 – Геометрична інтерпретація відмінності між прямою $\Delta f[kT]$ та оберненою $\nabla f[kT]$ скінченними різницями першого порядку у точці з аргументом $kT = 4T$

На рис. 2.4 показано, наскільки суттєво можуть відрізнятися пряма $\Delta f[kT]$ та обернена $\nabla f[kT]$ скінченні різниці першого порядку решітчастої функції $f[kT]$ при значенні аргументу $kT = 4T$.

За аналогією з (2.12) визначається і обернена скінченна різниця $\nabla^n f[kT]$ порядку n :

$$\begin{aligned}\nabla^n f[kT] &= \nabla^{n-1} f[kT] - \nabla^{n-1} f[(k-1)T], \\ n &= 1, 2, \dots; \quad k = 1, 2, \dots,\end{aligned}\tag{2.15}$$

де, як і в (2.13),

$$\begin{cases} \nabla^0 f[kT] = f[kT], \\ \nabla^0 f[(k-1)T] = f[(k-1)T]. \end{cases}\tag{2.16}$$

При $n = 2$ із формули (2.15) отримаємо

$$\begin{aligned}\nabla^2 f[kT] &= \nabla f[kT] - \nabla f[(k-1)T], \\ k &= 2, 3, 4, \dots,\end{aligned}\tag{2.17}$$

або, з урахуванням (2.14):

$$\begin{aligned}\nabla^2 f[kT] &= f[kT] - 2f[(k-1)T] + f[(k-2)T], \\ k &= 2, 3, 4, \dots\end{aligned}\tag{2.18}$$

Як приклад знайдемо $\Delta f[kT]$, $\Delta^2 f[kT]$ та $\nabla f[kT]$, $\nabla^2 f[kT]$ для решітчастих функцій $f_1[kT]$, $f_2[kT]$, визначених формулами (2.4) та (2.7).

Для функції $f_1[kT]$:

$$\begin{aligned}\Delta f_1[kT] &= f_1[(k+1)T] - f_1[kT] = k+1-4-(k-4)=1, \\ k &= 0, 1, 2, \dots,\end{aligned}\tag{2.19}$$

$$\begin{aligned}\Delta^2 f_1[kT] &= f_1[(k+2)T] - 2f_1[(k+1)T] + f_1[kT] = \\ &= k+2-4-2((k+1)-4)+(k-4)= \\ &= k+2-4-2k-2+8+k-4=0, \\ k &= 0, 1, 2, \dots,\end{aligned}\tag{2.20}$$

$$\begin{aligned}\nabla f_1[kT] &= f_1[kT] - f_1[(k-1)T] = k-4-(k-1-4)=1, \\ k &= 1, 2, 3, \dots,\end{aligned}\tag{2.21}$$

$$\begin{aligned}
\nabla^2 f_1[kT] &= f_1[kT] - 2f_1[(k-1)T] + f_1[(k-2)T] = \\
&= k - 4 - 2((k-1) - 4) + (k-2 - 4) = \\
&= k - 4 - 2k + 2 + 8 + k - 2 - 4 = 0, \\
&k = 2, 3, 4, \dots
\end{aligned} \tag{2.22}$$

Для функції $f_2[kT]$:

$$\begin{aligned}
\Delta f_2[kT] &= f_2[(k+1)T] - f_2[kT] = 0,75(k+1)^2 + 2 - (0,75k^2 + 2) = \\
&= 0,75k^2 + 1,5k + 0,75 + 2 - 0,75k^2 - 2 = 1,5k + 0,75, \\
&k = 0, 1, 2, \dots,
\end{aligned} \tag{2.23}$$

$$\begin{aligned}
\Delta^2 f_2[kT] &= f_2[(k+2)T] - 2f_2[(k+1)T] + f_2[kT] = \\
&= 0,75(k+2)^2 + 2 - 2(0,75(k+1)^2 + 2) + 0,75k^2 + 2 = \\
&= 0,75k^2 + 3k + 3 + 2 - 1,5k^2 - 3k - 1,5 - 4 + 0,75k^2 + 2 = 1,5, \\
&k = 0, 1, 2, \dots,
\end{aligned} \tag{2.24}$$

$$\begin{aligned}
\nabla f_2[kT] &= f_2[kT] - f_2[(k-1)T] = 0,75k^2 + 2 - (0,75(k-1)^2 + 2) = \\
&= 0,75k^2 + 2 - 0,75k^2 + 1,5k - 0,75 - 2 = 1,5k - 0,75, \\
&k = 1, 2, 3, \dots,
\end{aligned} \tag{2.25}$$

$$\begin{aligned}
\nabla^2 f_2[kT] &= f_2[kT] - 2f_2[(k-1)T] + f_2[(k-2)T] = \\
&= 0,75k^2 + 2 - 2(0,75(k-1)^2 + 2) + 0,75(k-2)^2 + 2 = \\
&= 0,75k^2 + 2 - 1,5k^2 + 3k - 1,5 - 4 + 0,75k^2 - 3k + 3 + 2 = 1,5, \\
&k = 2, 3, 4, \dots
\end{aligned} \tag{2.26}$$

Як видно з отриманих результатів, скінченні різниці першого порядку від лінійної решітчастої функції є константами, а другого порядку дорівнюють нулю.

Нагадаємо, що і перша похідна від лінійної неперервної функції є константою, а друга похідна теж дорівнює нулю. Така ж аналогія спостерігається і між решітчастою та неперервною квадратичними функціями.

2.2 Рівняння в скінченних різницях та різницеві рівняння

В попередньому підрозділі встановлено, що аналогом похідної l -го порядку, де $l = 1, 2, \dots, n$, для решітчастої функції $f[kT]$ є пряма $\Delta^l f[kT]$ та обернена $\nabla^l f[kT]$ скінченні різниці того ж порядку.

Очевидним наслідком цієї аналогії є те, що для решітчастих функцій $x[kT]$ та $y[kT]$ можна сконструювати рівняння в скінченних різницях $\Delta^l y[kT]$ ($l = 0, 1, 2, \dots, n$), $\Delta^q x[kT]$ ($q = 0, 1, 2, \dots, m$) або $\nabla^l y[kT]$ ($l = 0, 1, 2, \dots, n$), $\nabla^q x[kT]$ ($q = 0, 1, 2, \dots, m$), яке буде аналогом диференціального рівняння відносно породних функцій $y(t)$, $x(t)$ та їх похідних $y^{(l)}(t)$ ($l = 1, 2, \dots, n$), $x^{(q)}(t)$ ($q = 1, 2, \dots, m$).

Покажемо на прикладі як можна побудувати дискретний аналог диференціального рівняння.

Нехай маємо диференціальне рівняння

$$\frac{d^2 y}{dt^2} + 4 \frac{dy}{dt} + 2y = 2x \quad (2.27)$$

з початковими умовами

$$\begin{cases} y(0) = 0, \\ y'(0) = 1. \end{cases} \quad (2.28)$$

Нехай T – період дискретності функцій $x(t)$, $y(t)$ для породження решітчастих функцій $x[kT]$, $y[kT]$.

Нагадаємо, що аргументи t і kT пов'язані між собою співвідношенням:

$$t_k = kT. \quad (2.29)$$

Зрозуміло, що похідній $\frac{dy}{dt}$, яку маємо у рівнянні (2.27), в його дискретному аналізі відповідатиме вираз

$$\frac{\Delta y[kT]}{\Delta t_k} = \frac{\Delta y[kT]}{(k+1)T - kT} = \frac{\Delta y[kT]}{T}, \quad k = 0, 1, 2, \dots, \quad (2.30)$$

а другій похідній – вираз

$$\frac{\Delta^2 y[kT]}{(\Delta t_k)^2} = \frac{\Delta^2 y[kT]}{T^2}, \quad k = 0, 1, 2, \dots \quad (2.31)$$

Підставляючи (2.29) – (2.31) у рівняння (2.27), (2.28), отримаємо:

$$\frac{\Delta^2 y[kT]}{T^2} + 4 \frac{\Delta y[kT]}{T} + 2y[kT] = 2x[kT], \quad k = 0, 1, 2, \dots, \quad (2.32)$$

$$\begin{cases} y[0] = 0, \\ \frac{\Delta y[0]}{T} = 1, \end{cases} \quad (2.33)$$

або

$$\Delta^2 y[kT] + 4T \Delta y[kT] + 2T^2 y[kT] = 2T^2 x[kT], \quad k = 0, 1, 2, \dots, \quad (2.34)$$

$$\begin{cases} y[0] = 0, \\ \Delta y[0] = T. \end{cases} \quad (2.35)$$

Зрозуміло, що чим меншим буде значення періоду дискретності T , тим менше будуть відрізнятися один від одного значення t_k та t_{k+1} або t_{k-1} і, як наслідок, ближчими до розв'язку $y(t)$ диференціального рівняння (2.27) у точках t_k будуть розв'язки $y[kT]$ рівняння в скінченних різницях (2.34).

Найбільш просто рівняння (2.34) розв'язується шляхом перетворення його в різницеве, яке містить у собі не скінченні різниці, а значення решітчастих функцій, взятих при декількох значеннях аргументу.

Для здійснення цього перетворення необхідно у рівняннях (2.34), (2.35) замість скінченних різниць $\Delta y[kT]$, $\Delta^2 y[kT]$ підставити їхні значення, взяті з формул (2.9), (2.11).

Здійснивши це, отримаємо:

$$\begin{aligned} y[(k+2)T] - 2y[(k+1)T] + y[kT] + 4T(y[(k+1)T] - y[kT]) + \\ + 2T^2 y[kT] = 2T^2 x[kT], \end{aligned} \quad (2.36)$$

$$k = 0, 1, 2, \dots,$$

$$\begin{cases} y[0] = 0, \\ y[1 \cdot T] - y[0] = T, \end{cases} \quad (2.37)$$

або

$$y[(k+2)T] = 2T^2 x[kT] - (4T - 2) y[(k+1)T] - (2T^2 - 4T + 1) y[kT], \quad k = 0, 1, 2, \dots, \quad (2.38)$$

$$\begin{cases} y[0] = 0, \\ y[1 \cdot T] = T. \end{cases} \quad (2.39)$$

Вибравши конкретне значення T та сформувавши із заданої функції $x(t)$ решітчасту $x[kT]$, шляхом підстановки по черзі $k=0$, потім $k=1$ і наступні

його значення у рівняння (2.38) отримаємо стільки значень розв'язку $y[kT]$, скільки нам треба.

Отримати різницеве рівняння через обернені скінченні різниці $\nabla y[kT]$ та $\nabla^2 y[kT]$ пропонуємо самотійно як завдання на закріплення матеріалу. Також пропонуємо перетворити у різницеве довільне диференціальне рівняння третього порядку із самотійно заданими коефіцієнтами, початковими умовами та правою частиною.

2.3 Дискретне перетворення Лапласа та Z-перетворення

У першому розділі було показано, що лінійні диференціальні рівняння можна розв'язувати не лише безпосередньо, але й завдяки попередньому їх перетворенню в алгебраїчні рівняння комплексної змінної із застосуванням прямого та оберненого перетворень Лапласа.

Покажемо, що аналогічну процедуру можна застосувати і до різницевого рівнянь.

Нехай $y[kT]$ ($k = 0, 1, 2, \dots$) є решітчастою функцією, породженою неперервною функцією-оригіналом $y(t)$ з періодом дискретності T , для якої, як відомо, справедливо:

$$y(t) = \begin{cases} y(t), & t \geq 0, \\ 0, & t < 0, \end{cases} \quad (2.40)$$

$$Y(p) = \int_0^{\infty} y(t) e^{-pt} dt. \quad (2.41)$$

Оскільки

$$\begin{cases} t_k = kT, \\ dt \cong T, \end{cases} \quad (2.42)$$

то інтеграл у виразі (2.41) можна з деякою похибкою, яка зменшується із зменшенням значення T , записати у вигляді суми

$$Y^*(p) \cong \sum_{k=0}^{\infty} y(t_k) e^{-pt_k} T, \quad (2.43)$$

або

$$Y^*(p) \cong T \sum_{k=0}^{\infty} y[kT] e^{-pkT}. \quad (2.44)$$

Введемо нову комплексну змінну q у такий спосіб

$$q = pT \quad (2.45)$$

і запишемо суму, яка стоїть у правій частини виразу (2.44), у вигляді:

$$Y(q) = \sum_{k=0}^{\infty} y[kT] e^{-qk}. \quad (2.46)$$

Формула (2.46) задає дискретне перетворення Лапласа $Y(q)$ решітчастої функції $y[kT]$.

У попередньому розділі показано, що зображенням за Лапласом експоненціальної функції

$$f(t) = e^{-\alpha t} \quad (2.47)$$

є функція

$$F(p) = \frac{1}{p + \alpha}. \quad (2.48)$$

Знайдемо дискретне зображення за Лапласом решітчастого аналога

$$f[kT] = e^{-\alpha kT} \quad (2.49)$$

цієї експоненти, використавши формулу (2.46).

$$\begin{aligned} F(q) &= \sum_{k=0}^{\infty} f[kT] e^{-qk} = \sum_{k=0}^{\infty} e^{-\alpha kT} \cdot e^{-qk} = \sum_{k=0}^{\infty} e^{-(\alpha T + q)k} = \\ &= 1 + e^{-(\alpha T + q)} + e^{-2(\alpha T + q)} + \dots = \\ &= \frac{1}{1 - e^{-(\alpha T + q)}} = \frac{1}{1 - e^{-\alpha T} \cdot e^{-q}} = \frac{e^q}{e^q - e^{-\alpha T}}. \end{aligned} \quad (2.50)$$

Аналізуючи перетворення (2.50), бачимо, що в ньому використано формулу суми членів нескінченно спадної геометричної прогресії зі знаменником $e^{-(\alpha T + q)}$.

Порівнюючи формули (2.48) та (2.50), бачимо, що зображення за Лапласом неперервної експоненти є набагато простішим за своєю структурою, ніж аналогічне зображення її решітчастого аналога. Для більш складних функцій громіздкість зображень за Лапласом їх решітчастих аналогів суттєво наростає.

Ось чому сьогодні дискретне перетворення за Лапласом повністю витіснене з практики аналізу дискретних систем більш простим Z-перетворенням, яке отримується шляхом заміни у формулі (2.46)

$$z = e^q, \quad (2.51)$$

тобто

$$Z \{y[kT]\} = Y(z) = \sum_{k=0}^{\infty} y[kT] z^{-k}. \quad (2.52)$$

Прямою підстановкою (2.51) у (2.50) знайдемо, що Z-перетворення решітчастої експоненти має вигляд:

$$F(z) = \frac{z}{z - e^{-\alpha T}}. \quad (2.53)$$

Порівнюючи (2.53) з (2.48), бачимо, що вони мають однакову структуру – і у першому, і у другому випадках маємо дробово-раціональні функції, які є відношеннями поліномів у одному випадку за степенями p , а у другому – за степенями z .

Слід пам'ятати, що $e^{-\alpha T}$ за заданих α і T – це є число, наприклад при $\alpha = 1$ і $T = 2$ – це 0,135.

Як ще один приклад знайдемо Z-перетворення одиничної решітчастої функції $1[kT]$, $k = 0, 1, \dots$.

Оскільки одиничну функцію $1(t)$ можна отримати з експоненти $e^{-\alpha T}$, поклавши $\alpha = 0$, то одинична решітчаста функція $1[kT]$ є теж окремим випадком решітчастої експоненти при $\alpha = 0$. А тому для отримання Z-перетворення $1[kT]$ достатньо покласти у формулі (2.53) $\alpha = 0$. Тож матимемо

$$Z \{1[kT]\} = 1(z) = \frac{z}{z - 1}. \quad (2.54)$$

У справедливості формули (2.54) легко пересвідчитись і за допомогою основної формули (2.52) Z-перетворення, підставляючи $y[kT] = 1[kT]$ і знаходячи суму членів нескінченно спадної геометричної прогресії із знаменником z^{-1} .

Оскільки оператор Z-перетворення отримано з перетворення Лапласа шляхом двох лінійних підстановок, всі властивості перетворення Лапласа справедливі і для Z-перетворення, наприклад:

$$Z \{f_1[kT] + f_2[kT]\} = Z \{f_1[kT]\} + Z \{f_2[kT]\}, \quad (2.55)$$

$$Z \{\alpha f_1[kT]\} = \alpha Z \{f_1[kT]\}, \quad (2.56)$$

де α – число, значення якого не залежить від аргументу kT функції $f_1[kT]$.

Під час Z-перетворення різницевих рівнянь доводиться знаходити Z-зображення зсунутих вправо чи вліво значень решітчастих функцій, відносно яких ці рівняння складені, а тому побудуємо алгоритми їх знаходження.

Спочатку розглянемо зсув вправо:

$$Z \{y[(k+1)T]\} = \sum_{k=0}^{\infty} y[(k+1)T] z^{-k}. \quad (2.57)$$

Нехай $k + 1 = m$, тоді:

$$\begin{aligned} Z \{y[(k+1)T]\} &= \sum_{m=1}^{\infty} y[mT] z^{-(m-1)} = z \cdot \sum_{m=1}^{\infty} y[mT] z^{-m} = \\ &= z \cdot \left(\sum_{m=0}^{\infty} y[mT] z^{-m} - y[0] \right) = z \cdot (Y(z) - y[0]). \end{aligned} \quad (2.58)$$

Аналогічно:

$$Z \{y[(k+2)T]\} = \sum_{k=0}^{\infty} y[(k+2)T] z^{-k}. \quad (2.59)$$

Тепер нехай $m = k + 2$, тоді:

$$\begin{aligned} Z \{y[(k+2)T]\} &= \sum_{m=2}^{\infty} y[mT] z^{-(m-2)} = \\ &= z^2 \cdot \left(\sum_{m=0}^{\infty} y[mT] z^{-m} - y[0] - y[1 \cdot T] z^{-1} \right) = \\ &= z^2 \cdot (Y(z) - y[0] - y[1 \cdot T] z^{-1}). \end{aligned} \quad (2.60)$$

Узагальнюючи (2.58) та (2.60), отримуємо для довільного n :

$$\begin{aligned} Z \{y[(k+n)T]\} &= z^n \cdot (Y(z) - y[0] - y[1 \cdot T] z^{-1} - \dots - \\ &\quad - y[(n-1)T] z^{-(n-1)}). \end{aligned} \quad (2.61)$$

Тепер розглянемо зсув вліво:

$$Z \{y[(k-1)T]\} = \sum_{k=0}^{\infty} y[(k-1)T] z^{-k}. \quad (2.62)$$

Нехай $k - 1 = m$, тоді:

$$\begin{aligned} Z \{y[(k-1)T]\} &= \sum_{m=-1}^{\infty} y[mT] z^{-(m+1)} = \\ &= z^{-1} \cdot \left(y[-1 \cdot T] \cdot z + \sum_{m=0}^{\infty} y[mT] z^{-m} \right) = z^{-1} \cdot (0 + Y(z)) = z^{-1} \cdot Y(z). \end{aligned} \quad (2.63)$$

Виводячи формулу (2.63), використано властивість оригіналу бути рівним нулю за від'ємних значень аргументу.

Аналогічно:

$$Z \{y[(k-2)T]\} = \sum_{k=0}^{\infty} y[(k-2)T] z^{-k}. \quad (2.64)$$

Нехай $k-2=m$, тоді:

$$\begin{aligned} Z \{y[(k-2)T]\} &= \sum_{m=-2}^{\infty} y[mT] z^{-(m+2)} = \\ &= z^{-2} \cdot \left(y[-2 \cdot T] \cdot z^2 + y[-1 \cdot T] \cdot z^1 + \sum_{m=0}^{\infty} y[mT] z^{-m} \right) = \\ &= z^{-2} \cdot (0 + 0 + Y(z)) = z^{-2} \cdot Y(z). \end{aligned} \quad (2.65)$$

Узагальнюючи (2.63) та (2.65), отримуємо для довільного n :

$$Z \{y[(k-n)T]\} = z^{-n} \cdot Y(z). \quad (2.66)$$

Порівнюючи вирази (2.61) і (2.66), приходимо до висновку, що за допомогою Z -перетворення доцільніше розв'язувати різницеві рівняння, які конструюються із диференціальних шляхом застосування обернених скінченних різниць.

Якщо ж різницеве рівняння сконструйоване із диференціального за допомогою прямих скінченних різниць, то його доцільніше розв'язувати рекурентною (крок за кроком) процедурою, аналогічною (2.38).

На конкретному прикладі розглянемо процедуру Z -перетворення різницевого рівняння та алгоритм його розв'язання у цій області.

Нехай математична модель дискретної системи має вигляд:

$$\begin{aligned} y[kT] + 2y[(k-1)T] + 4y[(k-2)T] &= 2x[kT] - x[(k-1)T], \\ k &= 2, 3, 4, \dots, \end{aligned} \quad (2.67)$$

$$\begin{cases} y[-1 \cdot T] = 0, \\ y[-2 \cdot T] = 0. \end{cases} \quad (2.68)$$

Знайдемо Z -перетворення рівняння (2.67) за початкових умов (2.68).

Записуючи, спочатку формально,

$$Z \{y[kT] + 2y[(k-1)T] + 4y[(k-2)T]\} = Z \{2x[kT] - x[(k-1)T]\}, \quad (2.69)$$

та враховуючи лінійні властивості Z -перетворення (2.55), (2.56), отримаємо:

$$\begin{aligned} Z \{y[kT]\} + 2Z \{y[(k-1)T]\} + 4Z \{y[(k-2)T]\} &= \\ &= 2Z \{x[kT]\} - Z \{x[(k-1)T]\}. \end{aligned} \quad (2.70)$$

Із рівняння (2.70) з урахуванням виразів (2.52), (2.63), (2.65) випливає:

$$Y(z) + 2z^{-1}Y(z) + 4z^{-2}Y(z) = 2X(z) - z^{-1}X(z), \quad (2.71)$$

або

$$Y(z) = \frac{2 - z^{-1}}{1 + 2z^{-1} + 4z^{-2}} X(z), \quad (2.72)$$

або

$$Y(z) = W(z) X(z), \quad (2.73)$$

де

$$W(z) = \frac{2 - z^{-1}}{1 + 2z^{-1} + 4z^{-2}} \quad (2.74)$$

– дискретна передаточна функція динамічної системи, що описується різницевим рівнянням (2.67).

Як автоматизувати процес перетворень, приведених вище, покажемо на прикладі комп'ютерної програми, складеної на мові Python, для розв'язання задач, пов'язаних з перетворенням диференціального рівняння

$$a_3 \frac{d^3 y}{dt^3} + a_2 \frac{d^2 y}{dt^2} + a_1 \frac{dy}{dt} + a_0 y = b_2 \frac{d^2 x}{dt^2} + b_1 \frac{dx}{dt} + b_0 x \quad (2.75)$$

в різницеве рівняння

$$\begin{aligned} c_0 y[k] + c_1 y[k-1] + c_2 y[k-2] + c_3 y[k-3] = \\ = d_0 x[k] + d_1 x[k-1] + d_2 x[k-2] \end{aligned} \quad (2.76)$$

та Z-зображення цього різницевого рівняння у вигляді

$$Y(z) = \frac{d_0 + d_1 z^{-1} + d_2 z^{-2}}{c_0 + c_1 z^{-1} + c_2 z^{-2} + c_3 z^{-3}} X(z). \quad (2.77)$$

Цю програму ми подамо як **приклад № 7**, в якому вже не будемо розписувати зміст кожної команди, вважаючи, що ті читачі, які дочитають наш навчальний посібник до цього прикладу, уже оволоділи технікою програмування на мові Python в обсязі достатньому для розуміння приведених нами програм

```
In [1]: import sympy
In [2]: from sympy import*
In [3]: y = symbols('y:4')
In [4]: L1 = list(y);L1
Out[4]: [y0, y1, y2, y3]
In [5]: y0=L1[0];y1=L1[1];y2=L1[2];\
y3=L1[3]
In [6]: T = symbols('T')
In [7]: h = Function('h:6')(y)
In [8]: h0 = y0-y1
In [9]: h1 = h0/T
In [10]: h2 = y0-2*y1+y2
In [11]: h3 = h2/T**2
In [12]: h4 = y0-3*y1+3*y2-y3
In [13]: h5 = h4/T**3
```

```
In [14]: a = symbols('a:4')
In [15]: L2 = list(a);L2
Out[15]: [a0, a1, a2, a3]
In [16]: a0 = L2[0]; a1 = L2[1];\
a2 = L2[2]; a3 = L2[3]
In [17]: lsr = Function('lsr:5')(y)
In [18]: lsr0 = a3*h5+a2*h3+a1*h1+\
a0*y0; lsr0
Out[18]:
a0*y0 + a1*(y0 - y1)/T +\
a2*(y0 - 2*y1 + y2)/T**2 + a3*(y0 - \
3*y1 + 3*y2 - y3)/T**3
In [19]: lsr1=lsr0.expand();lsr1
Out[19]:
a0*y0 + a1*y0/T - a1*y1/T + \
```

```

a2*y0/T**2 - 2*a2*y1/T**2 + \
a2*y2/T**2 + a3*y0/T**3 - \
3*a3*y1/T**3 + 3*a3*y2/T**3 - \
a3*y3/T**3
In [20]: lsr2=lsr1.collect(y0);lsr2
Out[20]:
y0*(a0 + a1/T + a2/T**2 + \
a3/T**3) - a1*y1/T - 2*a2*y1/T**2 + \
a2*y2/T**2 - 3*a3*y1/T**3 + \
3*a3*y2/T**3 - a3*y3/T**3
In [21]: lsr3=lsr2.collect(y1);lsr3
Out[21]:
y0*(a0 + a1/T + a2/T**2 + a3/T**3) + \
y1*(-a1/T - 2*a2/T**2 - 3*a3/T**3) + \
a2*y2/T**2 + 3*a3*y2/T**3 - \
a3*y3/T**3
In [22]: lsr4=lsr3.collect(y2);lsr4
Out[22]:
y0*(a0 + a1/T + a2/T**2 + a3/T**3) + \
y1*(-a1/T - 2*a2/T**2 - 3*a3/T**3) + \
y2*(a2/T**2 + 3*a3/T**3) - \
a3*y3/T**3
In [23]: c = symbols('c:4')
In [24]: L11 = lsr4.as_ordered_terms(); \
L11
Out[24]:
[y0*(a0 + a1/T + a2/T**2 + \
a3/T**3), y1*(-a1/T - 2*a2/T**2 - \
3*a3/T**3), y2*(a2/T**2 + \
3*a3/T**3), -a3*y3/T**3]
In [25]: c0 = L11[0]/y0;c0
Out[25]:
a0 + a1/T + a2/T**2 + a3/T**3
In [26]: c1 = L11[1]/y1;c1
Out[26]:
-a1/T - 2*a2/T**2 - 3*a3/T**3
In [27]: c2 = L11[2]/y2;c2
Out[27]:
a2/T**2 + 3*a3/T**3
In [28]: c3 = L11[3]/y3;c3
Out[28]:
-a3/T**3
In [29]: Y = symbols('Y:4');Y
Out[29]: (Y0, Y1, Y2, Y3)
In [30]: L111=list(Y);L111
Out[30]: [Y0, Y1, Y2, Y3]
In [31]: Y0=L111[0];Y1=L111[1]; \
Y2=L111[2];Y3=L111[3]

```

```

In [32]: z = symbols('z')
In [33]: Y1=z**(-1)*Y0
In [34]: Y2=z**(-2)*Y0
In [35]: Y3=z**(-3)*Y0
In [36]: Q = Function('Q')(z)
In [37]: Q = c0*Y0+c1*Y1+c2*Y2+\
c3*Y3;Q
Out[37]:
Y0*(a0 + a1/T + a2/T**2 + \
a3/T**3) + Y0*(-a1/T - 2*a2/T**2 - \
3*a3/T**3)/z + Y0*(a2/T**2 + \
3*a3/T**3)/z**2 - \
Y0*a3/(T**3*z**3)
In [38]: F2=Function('F2')(z)
In [39]: F2=Q.subs(Y0,1);F2
Out[39]:
a0 + (-a1/T - 2*a2/T**2 - \
3*a3/T**3)/z + (a2/T**2 + \
3*a3/T**3)/z**2 + a1/T + a2/T**2 \
+ a3/T**3 - a3/(T**3*z**3)
In [40]: a0=4;a1=8;a2=4;a3=1
In [41]: T=0.5
In [42]: d11=dict({'T':0.5});d11
Out[42]: {'T': 0.5}
In [43]: F21=F2.subs([(L2[0],4), \
(L2[1],8),(L2[2],4), \
(L2[3],1)]);F21
Out[43]:
4 + (-8/T - 8/T**2 - 3/T**3)/z + \
(4/T**2 + 3/T**3)/z**2 + 8/T + \
4/T**2 + T**(-3) - 1/(T**3*z**3)
In [44]: F22=F21.subs({'T':0.5}); \
F22
Out[44]:
44.0 - 72.0/z + 40.0/z**2 - 8.0/z**3
In [45]: x = symbols('x:3')
In [46]: L3 = list(x);L3
Out[46]: [x0, x1, x2]
In [47]: x0=L3[0];x1=L3[1]; \
x2=L3[2]
In [48]: T1 = symbols('T1')
In [49]: m = Function('m:4')(x)
In [50]: m0 = x0-x1
In [51]: m1 = m0/T1
In [52]: m2 = x0-2*x1+x2
In [53]: m3 = m2/T1**2
In [54]: b = symbols('b:3')
In [55]: L4 = list(b);L4

```



```

Out[55]: [b0, b1, b2]
In [56]: b0 = L4[0]; b1 = L4[1]; \
b2 = L4[2]
In [57]: psr = Function('psr:4')(x)
In [58]: psr0 = b2*m3+b1*m1+\
b0*x0; psr0
Out[58]:
b0*x0 + b1*(x0 - x1)/T1 + \
b2*(x0 - 2*x1 + x2)/T1**2
In [59]: psr1=psr0.expand();psr1
Out[59]:
b0*x0 + b1*x0/T1 - b1*x1/T1 + \
b2*x0/T1**2 - 2*b2*x1/T1**2 +\
b2*x2/T1**2
In [60]: psr2=psr1.collect(x0);psr2
Out[60]:
x0*(b0 + b1/T1 + b2/T1**2) - \
b1*x1/T1 - 2*b2*x1/T1**2 +\
b2*x2/T1**2
In [61]: psr3=psr2.collect(x1);psr3
Out[61]:
x0*(b0 + b1/T1 + b2/T1**2) + \
x1*(-b1/T1 - 2*b2/T1**2) +\
b2*x2/T1**2
In [62]: d = symbols('d:3')
In [63]: L33=psr3.as_ordered_
terms( ); L33
Out[63]: [x0*(b0 + b1/T1 + \
lb2/T1**2), x1*(-b1/T1 - \
2*b2/T1**2), b2*x2/T1**2]
In [64]: d0 = L33[0]/x0;d0
Out[64]:
b0 + b1/T1 + b2/T1**2
In [65]: d1 = L33[1]/x1;d1
Out[65]:
-b1/T1 - 2*b2/T1**2
In [66]: d2 = L33[2]/x2;d2
Out[66]:
b2/T1**2
In [67]: X = symbols ('X:3');X

```

```

Out[67]: (X0, X1, X2)
In [68]: L333=list(X);\
L333
Out[68]: [X0, X1, X2]
In [69]: X0=L333[0];X1=L333[1];\
X2=L333[2]
In [70]: X1=z**(-1)*X0
In [71]: X2=z**(-2)*X0
In [72]: Q1 = Function ('Q1')(z)
In [73]: Q1=d0*X0+d1*X1+d2*X2;\
Q1
Out[73]:
X0*(b0 + b1/T1 + b2/T1**2) + \
X0*(-b1/T1 - 2*b2/T1**2)/z +\
X0*b2/(T1**2*z**2)
In [74]: F222=Function('F222')(z)
In [75]: F222=Q1.subs(X0,1);F222
Out[75]:
b0 + (-b1/T1 - 2*b2/T1**2)/z +\
b1/T1+b2/T1**2+b2/(T1**2*z**2)
In [76]: b0=4;b1=2;b2=1
In [77]: T1=0.5
In [78]: d22=dict({"T1":0.5});d22
Out[78]: {'T1': 0.5}
In [79]: F2222=\
F222.subs([(L4[0],4),\
(L4[1],2),(L4[2],1)]);F2222
Out[79]:
4 + (-2/T1 - 2/T1**2)/z + 2/T1 + \
T1**(-2) +1/(T1**2*z**2)
In [80]: F22222=F2222.subs({"T1":\
0.5});F22222
Out[80]:
12.0 - 12.0/z + 4.0/z**2
In [81]: W = Function ('W')(z)
In [82]: W = F22222/F1111;W
Out[82]:
(12.0 - 12.0/z + 4.0/z**2)/(44.0 - \
72.0/z+ 40.0/z**2 - 8.0/z**3)

```

А далі покладемо, що $x[kT] = 1[kT]$.

Тоді згідно з (2.54) матимемо

$$X(z) = \frac{z}{z-1} = \frac{1}{1-z^{-1}}. \quad (2.78)$$

Підставляючи (2.78) у (2.72), отримаємо:

$$Y(z) = \frac{2 - z^{-1}}{(1 + 2z^{-1} + 4z^{-2}) \cdot (1 - z^{-1})} = \frac{2 - z^{-1}}{1 + z^{-1} + 2z^{-2} - 4z^{-3}}. \quad (2.79)$$

Як і під час використання звичайного перетворення Лапласа, де для знаходження оригіналу за відомим зображенням використовується або процедура взяття оберненого інтеграла на основі теореми про лишки, або одна із формул розкладання, існує декілька процедур знаходження решітчастих оригіналів за їх відомим Z -зображенням. Але у практиці перевагу віддають найпростішій процедурі, яка полягає у діленні багаточлена на багаточлен.

Якщо розписати суму у формулі Z -перетворення в (2.52), то отримаємо:

$$Y(z) = y[0] + y[1 \cdot T]z^{-1} + y[2 \cdot T]z^{-2} + y[3 \cdot T]z^{-3} + \dots \quad (2.80)$$

Із виразу (2.80) видно, що коефіцієнти при степенях z^{-k} ($k = 0, 1, \dots$) і є значеннями решітчастої функції $y[kT]$, яка є оригіналом для Z -зображення $Y(z)$. А до форми (2.80) дробово-раціональна функція, яка є відношенням двох багаточленів, приводиться діленням чисельника на знаменник і взяттям такої кількості членів ряду частки від ділення, яка необхідна за умовами розв'язання.

Застосуємо описану процедуру до виразу (2.79):

$$\begin{array}{r|l}
 2 - z^{-1} & 1 + z^{-1} + 2z^{-2} - 4z^{-3} \\
 (-) & \hline
 2 + 2z^{-1} + 4z^{-2} - 8z^{-3} & 2 - 3z^{-1} - z^{-2} + 15z^{-3} - 25z^{-4} - \dots \\
 \hline
 -3z^{-1} - 4z^{-2} + 8z^{-3} & \\
 (-) & \\
 -3z^{-1} - 3z^{-2} - 6z^{-3} + 12z^{-4} & \\
 \hline
 -z^{-2} + 14z^{-3} - 12z^{-4} & \\
 (-) & \\
 -z^{-2} - z^{-3} - 2z^{-4} + 4z^{-5} & \\
 \hline
 15z^{-3} - 10z^{-4} - 4z^{-5} & \\
 (-) & \\
 15z^{-3} + 15z^{-4} + 30z^{-5} - 60z^{-6} & \\
 \hline
 -25z^{-4} - 34z^{-5} + 60z^{-6} & \\
 (-) & \\
 -25z^{-4} - 25z^{-5} - 50z^{-6} + 100z^{-7} & \\
 \hline
 -9z^{-5} + 110z^{-6} - 100z^{-7} &
 \end{array}$$

Зупинившись на цьому кроці процедури ділення многочлена на многочлен, вираз $Y(z)$, який задано у формі (2.76), можна записати у такий спосіб:

$$Y(z) = 2 - 3z^{-1} - z^{-2} + 15z^{-3} - 25z^{-4} + \frac{-9z^{-5} + 110z^{-6} - 100z^{-7}}{1 + z^{-1} + 2z^{-2} - 4z^{-3}}. \quad (2.81)$$

Порівнюючи вирази (2.80) та (2.81), знаходимо п'ять перших значень решітчастої функції $y[kT]$, яка є оригіналом для Z-зображення $Y(z)$, визначеного у формі (2.79):

$$\begin{aligned} y[0] &= 2, \\ y[1 \cdot T] &= -3, \\ y[2 \cdot T] &= -1, \\ y[3 \cdot T] &= 15, \\ y[4 \cdot T] &= -25. \end{aligned} \quad (2.82)$$

Оскільки процес ділення багаточлена на багаточлен легко алгоритмізується, то на комп'ютері дуже швидко можна отримати яку завгодно сукупність значень решітчастого оригіналу, час отримання якої обмежується, фактично, лише часом виведення цієї сукупності значень за допомогою принтера на папір.

Продемонструємо це на **прикладі № 8**, в якому приведена комп'ютерна програма, складена на мові Python для розв'язання задач, пов'язаних з визначенням послідовності значень $\{y[0], y[T], y[2T], y[3T], y[4T], \dots\}$ реакції $y[kT]$, $k=0,1,2,\dots$, $T>0$ об'єкта дослідження з дискретною передаточною функцією

$$W(z) = \frac{12 - 12z^{-1} + 4z^{-2}}{44 - 72z^{-1} + 40z^{-2} - 8z^{-3}} \quad (2.83)$$

на дискретний вхідний сигнал

$$x[kT] = kT, \quad k=0,1,2,\dots, \quad T=0,5, \quad (2.84)$$

In [1]: import sympy	In [13]: Q = Function('Q')(z)
In [2]: from sympy import*	In [14]: W = (12.0 - 12.0/z + \
In [3]: z = symbols('z')	4.0/z**2)/(44.0 - 72.0/z + \
In [4]: T = symbols('T')	40.0/z**2 - 8.0/z**3)
In [5]: X = Function('X')(z)	In [15]: X = T*z**(-1)/(1 - \
In [6]: Y = Function('Y')(z)	2*z**(-1) + z**(-2))
In [7]: W = Function('W')(z)	In [16]: Y = W*X; Y
In [8]: C = Function('C')(z)	Out[16]:
In [9]: D = Function('D')(z)	T*(12.0 - 12.0/z + 4.0/z**2)/(z*(1 - \
In [10]: A = Function('A:5')(z)	2/z + z**(-2))*(44.0 - 72.0/z + \
In [11]: y = symbols('y:5')	40.0/z**2 - 8.0/z**3))
In [12]: R = Function('R')(z)	In [17]: Q = Y.expand();Q

```

Out[17]:
4.0*T/(44.0*z**3 - 160.0*z**2 + \
228.0*z - 160.0 + 56.0/z - \
8.0/z**2) - 12.0*T/(44.0*z**2 - \
160.0*z + 228.0 - 160.0/z + \
56.0/z**2 - 8.0/z**3) + \
12.0*T/(44.0*z - 160.0 + \
228.0/z - 160.0/z**2 + \
56.0/z**3 - 8.0/z**4)
In [18]: T = 0.5
In [19]: d = dict({'T':0.5});d
Out[19]: {'T': 0.5}
In [20]: M = Function('M')(z)
In [21]: M = Q.subs({'T':0.5});M
Out[21]:
2.0/(44.0*z**3 - 160.0*z**2 + \
228.0*z - 160.0 + 56.0/z - \
8.0/z**2) - 6.0/(44.0*z**2 - \
160.0*z + 228.0 - 160.0/z + \
56.0/z**2 - 8.0/z**3) + \
6.0/(44.0*z - 160.0 + 228.0/z - \
160.0/z**2 + 56.0/z**3 - \
8.0/z**4)
In [22]: M1= Function('M1')(z)
In [23]: M1 = cancel(M);M1
Out[23]:
1.0*(11616.0*z**4 - 11616.0*z**3 + \
3872.0*z**2)/(85184.0*z**5 - \
309760.0*z**4 + 441408.0*z**3 - \
309760.0*z**2 + 108416.0*z - \
15488.0)
In [24]: expr = M1
In [25]: C,D = fraction(expr)
In [26]: print(C)
11616.0*z**4 - 11616.0*z**3 + \
3872.0*z**2
In [27]: print(D)
85184.0*z**5 - 309760.0*z**4 + \
441408.0*z**3 - 309760.0*z**2 + \
108416.0*z - 15488.0
In [28]: L1 = list(y);L1
Out[28]: [y0, y1, y2, y3, y4]
In [29]: f = Function('f.16')(z)
In [30]: A0 = L1[0];A0
Out[30]:
y0
In [31]: A1 =L1[1]*z**(-1);A1
Out[31]:

```

```

y1/z
In [32]: A2 = L1[2]*z**(-2);A2
Out[32]:
y2/z**2
In [33]: A3 = L1[3]*z**(-3);A3
Out[33]:
y3/z**3
In [34]: A4 =L1[4]*z**(-4);A4
Out[34]:
y4/z**4
In [35]: r = Function('r')(z)
In [36]: R = A0+A1+A2+A3+\
A4+r;R
Out[36]:
y0 + y1/z + y2/z**2 + y3/z**3 + \
y4/z**4 + r(z)
In [37]: L3 = C.as_ordered_terms( );\
L3
Out[37]: [11616.0*z**4, -\
11616.0*z**3, 3872.0*z**2]
In [38]: L4 = D.as_ordered_terms( );\
L4
Out[38]:
[85184.0*z**5, -309760.0*z**4,\
441408.0*z**3,-309760.0*z**2,\
108416.0*z, -15488.0000000000]
In [39]: L3[0]
Out[39]:
11616.0*z**4
In [40]: L4[0]
Out[40]:
85184.0*z**5
In [41]: L3[0]/L4[0]
Out[41]:
0.136363636363636/z
In [42]: A1 = L3[0]/L4[0];A1
Out[42]:
0.136363636363636/z
In [43]: y1 = A1*z;y1
Out[43]:
0.136363636363636
In [44]: A1*D
Out[44]:
0.136363636363636*(85184.0*\
z**5 - 309760.0*z**4 + 441408.0*\
z**3 - 309760.0*z**2 + \
108416.0*z - 15488.0)/z
In [45]: f0 = C-A1*D;f0

```

```

Out[45]:
11616.0*z**4 - 11616.0*z**3 + \
3872.0*z**2 - 0.1363636363636363*\
(85184.0*z**5 - 309760.0*z**4 + \
441408.0*z**3 - 309760.0*z**2 + \
108416.0*z - 15488.0)/z
In [46]: f1 = f0.expand ();f1
Out[46]:
-1.81898940354586e-12*z**4 + \
30624.0*z**3 - 56320.0*z**2 + \
42240.0*z - 14784.0 + 2112.0/z
In [47]: L6 = f1.as_ordered_terms();\
L6
Out[47]:
[-1.81898940354586e-12*z**4,
30624.0*z**3, -56320.0*z**2,\
42240.0*z,-14784.0000000000,\
2112.0/z]
In [48]: L6.pop(0)
Out[48]:
-1.81898940354586e-12*z**4
In [49]: L6
Out[49]: [30624.0*z**3,\
-56320.0*z**2, 42240.0*z,\
-14784.0000000000, 2112.0/z]\
In [50]: f4 = L6[0]+L6[1]+L6[2]+\
L6[3]+L6[4];f4
Out[50]:
30624.0*z**3 - 56320.0*z**2 + \
42240.0*z - 14784.0 + 2112.0/z
In [51]: A2=L6[0]/L4[0];A2
Out[51]:
0.359504132231405/z**2
In [52]: y2=A2*z**2;y2
Out[52]:
0.359504132231405
In [53]: A2*D
Out[53]:
0.359504132231405*\
(85184.0*z**5 - 309760.0*z**4 + \
441408.0*z**3 - 309760.0*z**2 + \
108416.0*z - 15488.0)/z**2
In [54]: f5=f4-A2*D;f5
Out[54]:
30624.0*z**3 - 56320.0*z**2 + \
42240.0*z -14784.0 + 2112.0/z - \
0.359504132231405*(85184.0*\
z**5 - 309760.0*z**4 + \

```

```

441408.0*z**3 - 309760.0*z**2 + \
108416.0*z - 15488.0)/z**2
In [55]: f6=f5.expand();f6
Out[55]:
-3.63797880709171e-12*z**3 + \
55040.0*z**2 - 116448.0*z + \
96576.0 - 36864.0/z + \
5568.0/z**2)
In [56]: L7 = \
f6.as_ordered_terms(); L7
Out[56]:
[-3.63797880709171e-12*z**3,\
55040.0*z**2, -116448.0*z,\
96576.0000000000, -36864.0/z,\
5568.0/z**2]
In [57]: L7.pop(0)
Out[57]:
-3.63797880709171e-12*z**3
In [58]: L7
Out[58]: [55040.0*z**2, -\
116448.0*z, 96576.0000000000,\
-36864.0/z, 5568.0/z**2]
In [59]: f7 =L7[0]+L7[1]+L7[2]+\
L7[3]+L7[4];f7
Out[59]:
55040.0*z**2 - 116448.0*z + \
96576.0 - 36864.0/z + \
5568.0/z**2
In [60]: A3 =L7[0]/L4[0];A3
Out[60]:
0.646130728775357/z**3
In [61]: y3=A3*z**3;y3
Out[61]:
0.646130728775357
In [62]: A3*D
Out[62]:
0.646130728775357*\
(85184.0*z**5 - 309760.0*z**4 + \
441408.0*z**3 - 309760.0*z**2 + \
108416.0*z - 15488.0)/z**3
In [63]: f8 = f7-A3*D;f8
Out[63]:
55040.0*z**2 - 116448.0*z + \
96576.0 - 36864.0/z + \
5568.0/z**2 - 0.6461307287*\
(85184.0*z**5 - 309760.0*z**4 + \
441408.0*z**3 - 309760.0*z**2 + \
108416.0*z - 15488.0)/z**3

```

```

In [64]: f9 = f8.expand();f9
Out[64]:
83697.45454545456*z - \
188631.27272727273 +\
163281.454545455/z - \
64482.9090909091/z**2 +\
10007.2727272727/z**3
In [65]: L8 =\
f9.as_ordered_terms( ); L8
Out[65]:
[83697.45454545456*z,\
-188631.272727273,\
163281.454545455/z,\
-64482.9090909091/z**2,\
10007.2727272727/z**3]
In [66]: A4 =L8[0]/L4[0];A4
Out[66]:
0.982549006215423/z**4
In [67]: y4 = A4*z**4;y4
Out[67]:
0.982549006215423
In [68]: A4*D
Out[68]:
0.982549006215423*\
(85184.0*z**5 - 309760.0*z**4 +\
441408.0*z**3 - 309760.0*z**2 +\
108416.0*z - 15488.0)/z**4
In [69]: f10=f9-A4*D;f10
Out[69]:
83697.45454545456*z -\
188631.27272727273 +\
163281.454545455/z - \
64482.9090909091/z**2 + \
10007.2727272727/z**3 - \
0.982549006215423*(85184.0*\
z**5 - 309760.0*z**4 + \
441408.0*z**3 - 309760.0*z**2 +\
108416.0*z - 15488.0)/z**4
In [70]: f11=f10.expand();f11

```

```

Out[70]:
115723.107438017 - \
270423.537190083/z +\
239871.47107438/z**2 - \
96516.7603305786/z**3 + \
15217.7190082645/z**4
In [71]: r=f11/D;r
Out[71]:
(115723.107438017 - \
270423.537190083/z +\
239871.47107438/z**2 - \
96516.7603305786/z**3 + \
15217.71900826/z**4)/(85184.0*\
z**5-309760.0*z**4 + 441408.0*\
z**3-309760.0*z**2 + 108416.0*\
z - 15488.0)
In [72]: L1[0]=0;L1
Out[72]: [0, y1, y2, y3, y4]
In [73]: L1[1]=y1;L1
Out[73]: [0, 0.13636363636363636, \
y2, y3, y4]
In [74]: L1[2]=y2;L1
Out[74]: [0, 0.13636363636363636, \
0.359504132231405, y3, y4]
In [75]: L1[3]=y3;L1
Out[75]: [0, 0.13636363636363636, \
0.359504132231405, \
0.646130728775357, y4]
In [76]: L1[4]=y4;L1
Out[76]: [0, 0.13636363636363636, \
0.359504132231405, \
0.646130728775357, \
0.982549006215423]
In [77]: import matplotlib
In [78]: import matplotlib.pyplot\
as plt
In [79]: plt.plot(L1)
Out[79]: [<matplotlib.lines.\
Line2D at 0x22cea175100>]

```

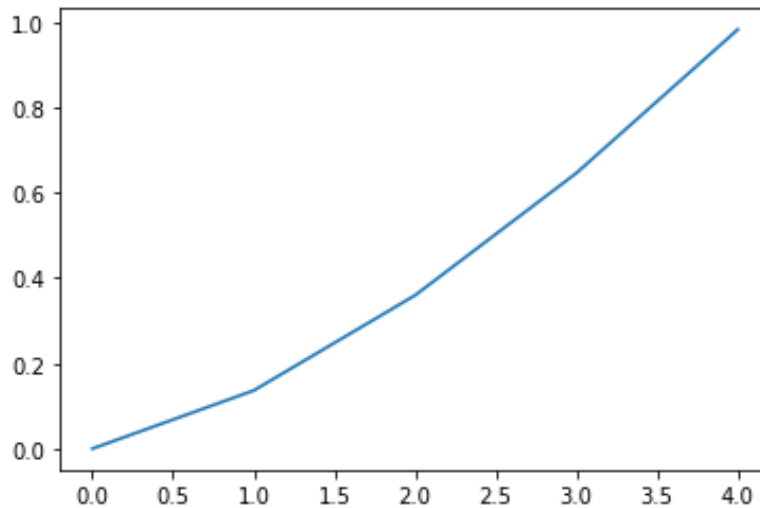


Рисунок 2.5 – Графік реакції об'єкта дослідження $y[k]$, $k=0,1,2,3,4$; $T=0,5$ у відносному часі k з лінійною графічною інтерполяцією між точками для кращої демонстрації характеру цієї реакції

2.4 Математична модель комп'ютера як елемента дискретної детермінованої ЛДС ЗП

Відомо, що при складних законах управління навіть лінійними динамічними об'єктами стає доцільним використання комп'ютера в замкнутому контурі для формування цих законів. Найчастіше комп'ютери, на яких покладаються такі задачі, є спеціалізованими і реалізованими в мікропроцесорному варіанті.

Як відомо, цифри у мікропроцесорі формуються комбінаціями імпульсів, а сигнали до і після мікропроцесора є неперервними, а точніше – кусково-неперервними часовими функціями. Для узгодження характеристик неперервних та імпульсних процесів на вході мікропроцесора ставлять пристрій, котрий неперервні сигнали перетворює в комбінації імпульсів і називається *аналого-цифровим перетворювачем* (АЦП), а на виході мікропроцесора обов'язковим під час його використання в мікропроцесорній системі є підключення пристрою, який комбінації імпульсів перетворює в кусково-неперервний сигнал і називається *цифроаналоговим перетворювачем* (ЦАП). Спрощені функціональні схеми АЦП та ЦАП зображені на рис. 2.6 та 2.7.

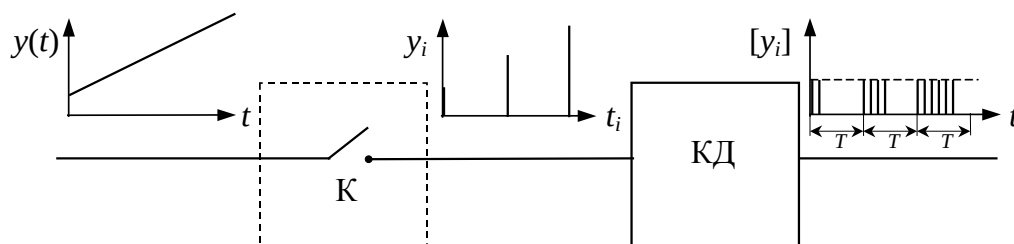


Рисунок 2.6 – Спрощена функціональна схема АЦП

Принципово АЦП (рис. 2.6) складається тільки з двох функціональних блоків. Перший блок – це ключ К, який, лише на мить один раз за період часу T вмикаючись, перетворює неперервний сигнал $y(t)$ у послідовність імпульсів y_i , для яких справедливо:

$$y_i = y[iT], \quad i = 0, 1, 2, \dots \quad (2.85)$$

Другий блок – це кодер КД, який кожному імпульсу y_i ставить у відповідність пачку імпульсів однакової амплітуди, кількість « n » яких у пачці однозначно залежить від значення амплітуди імпульсу y_i , тобто

$$n = n(y_i). \quad (2.86)$$

З такими параметрами імпульсів однакової амплітуди вже може працювати мікропроцесор.

В результаті виконання мікропроцесором заданих операцій на його виході з'являються комбінації інших пачок імпульсів, котрі надходять на вхід цифро-аналогового перетворювача ЦАП (рис. 2.7), який принципово складається лише з трьох функціональних блоків.

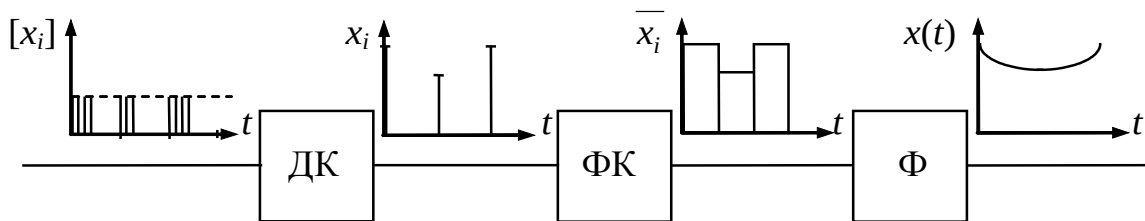


Рисунок 2.7 – Спрощена функціональна схема ЦАП

Перший блок – це декодер ДК, котрий кожній пачці з « m » імпульсів однакової амплітуди ставить у відповідність один імпульс x_i , амплітуда якого однозначно відображає кількість « m » імпульсів у пачці, тобто

$$x_i = x_i(m). \quad (2.87)$$

Другий блок – фіксатор ФК, який запам'ятовує амплітуду x_i імпульсу, що надходить з декодера ДК, на весь період T аж до появи наступного імпульсу x_{i+1} , перетворюючи послідовність імпульсів x_i у кусково-сталу східчасту функцію $\bar{x}_i$. Динамічна характеристика фіксатора має вигляд, наведений на рис. 2.8, а математично може бути записаною у вигляді

$$x_i(t) = x[iT] \cdot \{1(t - iT) - 1(t - [i+1]T)\}, \quad i = 0, 1, 2, 3, \dots, \quad (2.88)$$

де, нагадаємо, $1(t)$ – одинична східчаста функція.

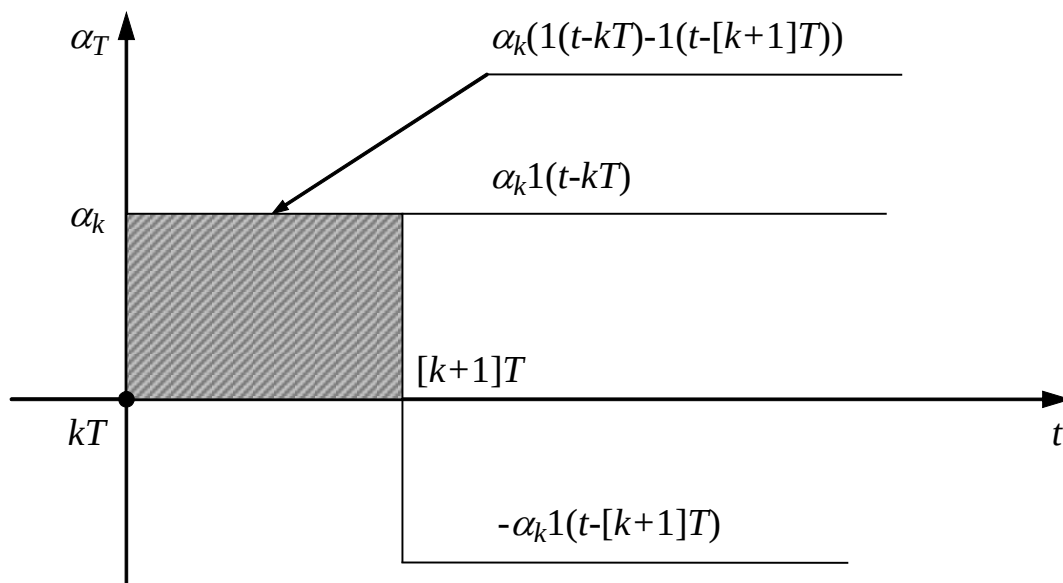


Рисунок 2.8 – Графік динамічної характеристики фіксатора

Третій блок ЦАП – фільтр Φ згладжує кусково-сталу східчасту функцію $\bar{x}_i$ і перетворює її в неперервну функцію $x(t)$.

Якщо використати алгоритм експоненціального згладжування, то модель згладженого сигналу $x(t)$ на виході фільтра матиме вигляд

$$x(t) = x[iT] \cdot (1(t-iT) - 1(t-[i+1]T)) + \{x([i-1]T) \times (1(t-[i-1]T) - 1(t-iT)) - x[iT] \cdot (1(t-iT) - 1(t-[i+1]T))\} \times e^{-\alpha(t-[i-1]T)}, \quad i=1,2,3,\dots \quad (2.89)$$

Що стосується безпосередньо мікропроцесора, то він протягом відрізка часу τ реалізує задану програму, завдяки якій довільна функція $y(t)$ перетворюється у функцію $x(t)$ з заданими характеристиками.

Для прикладу розглянемо випадок, коли мікропроцесор формує сигнал $x(t)$, який містить три складові: перша пропорційна координаті $y(t)$, друга пропорційна її першій похідній $\dot{y}(t)$, а третя – її другій похідній $\ddot{y}(t)$, тобто

$$x(t) = c_2 \frac{d^2 y(t)}{dt^2} + c_1 \frac{dy(t)}{dt} + c_0 y(t). \quad (2.90)$$

Але, оскільки мікропроцесору для реалізації залежності (2.90) потрібен певний час τ , протягом якого він отримує результат і не видає нічого на ЦАП, то рівняння (2.90) недостатньо для того, щоб задати модель мікропроцесора у загальній математичній моделі динамічної системи. Його потрібно доповнити ще рівнянням блока затримки сигналу на час τ , тобто рівнянням

$$x_{лз}(t) = x(t - \tau) \cdot 1(t - \tau). \quad (2.91)$$

Після цього зробимо два важливі зауваження. Перше з них полягає в тому, що не треба думати начебто якимсь одним рівнянням, наприклад (2.90), можна раз і назавжди задати математичну модель мікропроцесора. Вона змінюватиметься кожний раз, коли зазнаватиме змін та функція, реалізація якої покладатиметься на мікропроцесор.

Друге зауваження полягає в тому, що рівняння (2.90) стає непридатним для моделювання процесів у системі, якщо розглядається не лише характер процесів у її безперервній частині, а й характер процесів в імпульсних елементах системи, наприклад у ЦАП чи АЦП.

У цьому випадку від диференціального рівняння як математичної моделі мікропроцесора слід переходити до рівняння в скінченних різницях, що для прикладу зробимо саме для моделі (2.90), використавши обернені скінченні різниці.

Використовуючи для перетворення диференціального рівняння (2.90) на основі обернених скінченних різниць процедуру, викладену у співвідношеннях (2.27) – (2.32) для прямих скінченних різниць, отримаємо

$$x[kT] = c_2 \frac{\nabla^2 y}{T^2} + c_1 \frac{\nabla y}{T} + c_0 y[kT], \quad (2.92)$$

або

$$\begin{aligned} x[kT] = & \frac{c_2}{T^2} (y[kT] - 2y[(k-1)T] + y[(k-2)T]) + \\ & + \frac{c_1}{T} (y[kT] - y[(k-1)T]) + c_0 y[kT]. \end{aligned} \quad (2.93)$$

або

$$x[kT] = \left(\frac{c_2}{T^2} + \frac{c_1}{T} + c_0 \right) y[kT] - \left(\frac{2c_2}{T^2} + \frac{c_1}{T} \right) y[(k-1)T] + \frac{c_2}{T^2} y[(k-2)T]. \quad (2.94)$$

Якщо покласти, що для часу τ , який використовується мікропроцесором для розрахунку вихідної координати закладеної в нього моделі, наприклад, (2.90), справедливою є рівність

$$\tau = lT, \quad (2.95)$$

де l – ціле число, то за дискретний аналог рівняння блока затримки (2.91) матимемо

$$x_{лз}[kT] = x[(k-l)T] \cdot 1[(k-l)T]. \quad (2.96)$$

На завершення цього підрозділу зауважимо і звернемо увагу на те, що у функціональній схемі динамічної системи, яка використовує мікропроцесор, кодер АЦП і декодер ЦАП слід відносити до функціональної схеми мікропроцесора, за межами якої виноситься від АЦП лише ключ, котрий із

неперервної вхідної функції формує решітчасту, а від ЦАП за межі функціональної схеми мікропроцесора виноситься фіксатор, який із решітчастої вихідної функції формує неперервну кусково-сталу, та фільтр, який неперервну кусково-сталу функцію перетворює у неперервну гладку.

У загальній математичній моделі такої динамічної системи ці блоки враховуються їх моделями, визначеними виразами (2.85), (2.88) та (2.89).

Але якщо кінцевою моделлю динамічної системи є сукупність різницевих рівнянь, що описують усі структурні блоки цієї системи, то враховувати у цій сукупності різницевих рівнянь вирази (2.85), (2.88) та (2.89) немає потреби.

2.5 Приклад побудови математичної моделі детермінованої дискретної ЛДС ЗП

Нехай за допомогою мікропроцесорного регулятора (МПР), який реалізує закон регулювання, що представлений залежністю (2.90), здійснюється управління детермінованим лінійним динамічним об'єктом (ЛДО), про який відомо лише те, що його реакцією на вхідний сигнал $x(t)$ у вигляді одиничної східчастої функції (1.66) є вихідний сигнал $y(t)$ у вигляді залежності (1.51), в якій покладемо, що $f(t) = y(t)$. Функціональна схема цієї системи управління наведена на рис. 2.9.

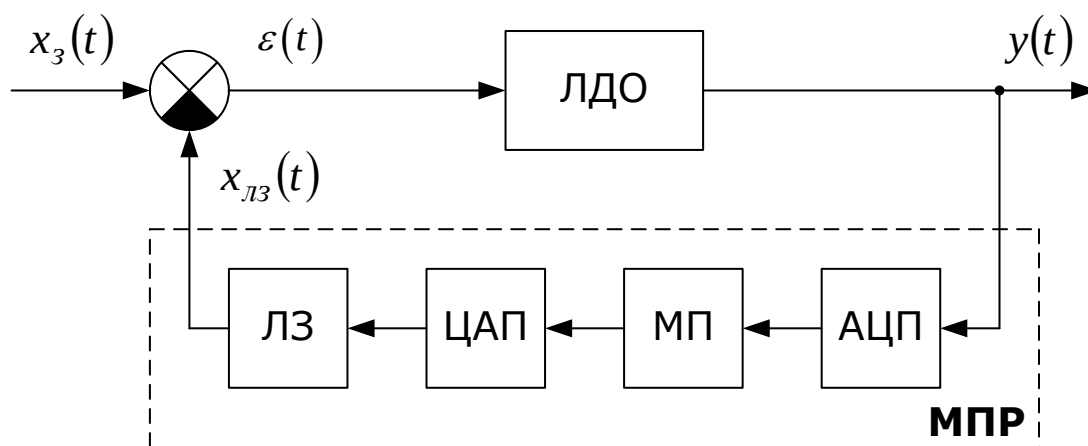


Рисунок 2.9 – Функціональна схема мікропроцесорної системи управління детермінованим лінійним динамічним об'єктом

На цій схемі для функціональних блоків АЦП і ЦАП прийняті такі ж позначення, як і на схемах рисунків 2.6 і 2.7, а символом ЛЗ позначена лінія затримки на час τ , потрібний мікропроцесору для визначення координати управління згідно з закладеним в нього законом (2.90).

Побудуємо математичну модель цієї дискретної детермінованої ЛДС ЗП як в часовій області у вигляді системи різницевих рівнянь, так і на комплексній площині у вигляді дискретної передаточної функції.

Почнемо з побудови моделей об'єкта управління ЛДО, передаточну функцію якого $W(p)$ знайдемо за допомогою виразу (1.40), підставивши в нього перетворені за Лапласом вхідний та вихідний сигнали,

$$\begin{aligned} W(p) &= \frac{L\{y(t)\}}{L\{x(t)\}} = \frac{L\left\{6 + \frac{3}{2}e^{-2t} - \frac{5}{3}e^{-3t}\right\}}{L\{1(t)\}} = \\ &= \frac{2p+1}{\frac{p^3+5p^2+6p}{\frac{1}{p}}} = \frac{2p+1}{p^2+5p+6}. \end{aligned} \quad (2.97)$$

Зауважимо, що при отриманні виразу (2.97) використано отримані раніше зображення функцій (1.26) та (1.51) у вигляді (1.27) та (1.46).

Підставляючи (2.97) в (1.38), після нескладних перетворень отримаємо:

$$(p^2 + 5p + 6)Y(p) = (2p + 1)X(p). \quad (2.98)$$

З викладеного раніше випливає, що рівнянню (2.98) в області зображень в області оригіналів буде відповідати диференціальне рівняння

$$\frac{d^2y}{dt^2} + 5\frac{dy}{dt} + 6y(t) = 2\frac{dx}{dt} + x(t), \quad (2.99)$$

якому, у свою чергу, відповідатиме рівняння в кінцевих обернених різницях

$$\frac{\nabla^2 y}{T^2} + 5\frac{\nabla y}{T} + 6y[kT] = 2\frac{\nabla x}{T} + x[kT], \quad (2.100)$$

де T – період, з яким ключ АЦП із неперервної функції $y(t)$ формує решітчасту $y[kT]$, а декодер ЦАП із пачок імпульсів, що надходять з мікропроцесора, формує решітчасту функцію $x[kT]$.

Використовуючи раніше викладену методику, тобто розписуючи в рівнянні (2.100) обернені різниці через значення решітчастої функції та впорядковуючи отримані результати, різницеvim аналогом цього рівняння матимемо

$$\begin{aligned} (1 + 5T + 6T^2)y[kT] - (2 + 5T)y[(k-1)T] + y[(k-2)T] = \\ = (1 + 2T)x[kT] - 2Tx[(k-1)T]. \end{aligned} \quad (2.101)$$

Різницеve рівняння (2.101) задає дискретну математичну модель об'єкта управління дискретної динамічної системи, а різницеvi рівняння (2.94) та (2.96), побудовані у попередньому підрозділі, задають дискретну математичну модель мікропроцесорного регулятора цієї системи, включеного в контур

зворотного зв'язку. Очевидно, що для побудови замкнутої математичної моделі динамічної системи, що розглядається, придатної для її аналізу і оптимізації, необхідно систему різницевих рівнянь (2.101), (2.94), (2.96) доповнити ще одним різницеvim рівнянням, яке задасть дискретну математичну модель компаратора на вході ЛДО, котрий має два входи: один для заданого сигналу $x_3(t)$, котрий об'єкт управління повинен відпрацьовувати, а другий для сигналу $x_{лз}(t)$, котрий надходить по контуру зворотного зв'язку з мікропроцесорного регулятора, та один вихід, з якого на об'єкт управління подається відхилення $\varepsilon(t)$, яке виникає в процесі роботи системи між сигналами $x_3(t)$ та $x_{лз}(t)$.

Виходячи з логіки роботи компаратора, можна одразу записати, що

$$\varepsilon[kT] = x_3[kT] - x_{лз}[kT], \quad k = 0, 1, 2, \dots \quad (2.102)$$

Цілком очевидно, що для замикання математичної моделі дискретної динамічної системи, заданої різницеvими рівняннями (2.94), (2.96), (2.101), (2.102), у правій частині рівняння (2.101) необхідно замість $x[kT]$, $x[(k-1)T]$ підставити відповідно $\varepsilon[kT]$, $\varepsilon[(k-1)T]$.

Тепер побудуємо математичну модель цієї ж дискретної динамічної системи на комплексній площині. Для цього здійcнимо Z-перетворення різницеvих рівнянь (2.94), (2.96), (2.101), (2.102).

Перетворюючи (2.94), отримаємо

$$X(z) = \left(\frac{c_2}{T^2} + \frac{c_1}{T} + c_0 \right) Y(z) - \left(\frac{2c_2}{T^2} + \frac{c_1}{T} \right) z^{-1} Y(z) + \frac{c_2}{T^2} z^{-2} Y(z), \quad (2.103)$$

або

$$X(z) = W_{mn}(z) \cdot Y(z), \quad (2.104)$$

де

$$W_{mn}(z) = \left(\frac{c_2}{T^2} + \frac{c_1}{T} + c_0 \right) - \left(\frac{2c_2}{T^2} + \frac{c_1}{T} \right) z^{-1} + \frac{c_2}{T^2} z^{-2} \quad (2.105)$$

дискретна передаточна функція мікропроцесора МП.

Перетворюючи (2.96), отримаємо

$$X_{лз}(z) = z^{-1} X(z), \quad (2.106)$$

або

$$X_{лз}(z) = W_{лз}(z) \cdot X(z), \quad (2.107)$$

де

$$W_{лз}(z) = z^{-1} \quad (2.108)$$

дискретна передаточна функція лінії затримки ЛЗ.

Перетворюючи (2.101) з врахуванням зробленого вище зауваження стосовно його правої частини, отримаємо

$$\begin{aligned} (1 + 5T + 6T^2)Y(z) - (2 + 5T)z^{-1}Y(z) + z^{-2}Y(z) = \\ = (1 + 2T)E(z) - 2Tz^{-1}E(z), \end{aligned} \quad (2.109)$$

або

$$Y(z) = W(z) \cdot E(z), \quad (2.110)$$

де

$$W(z) = \frac{(1 + 2T) - 2Tz^{-1}}{(1 + 5T + 6T^2) - (2 + 5T)z^{-1} + z^{-2}} \quad (2.111)$$

– дискретна передаточна функція об'єкта управління ЛДО.

Перетворюючи (2.102), отримаємо

$$E(z) = X_3(z) - X_{лз}(z). \quad (2.112)$$

Структурна схема динамічної системи, дискретні передаточні функції структурних блоків якої задані виразами (2.105), (2.108), (2.111), (2.112), наведена на рис. 2.10.

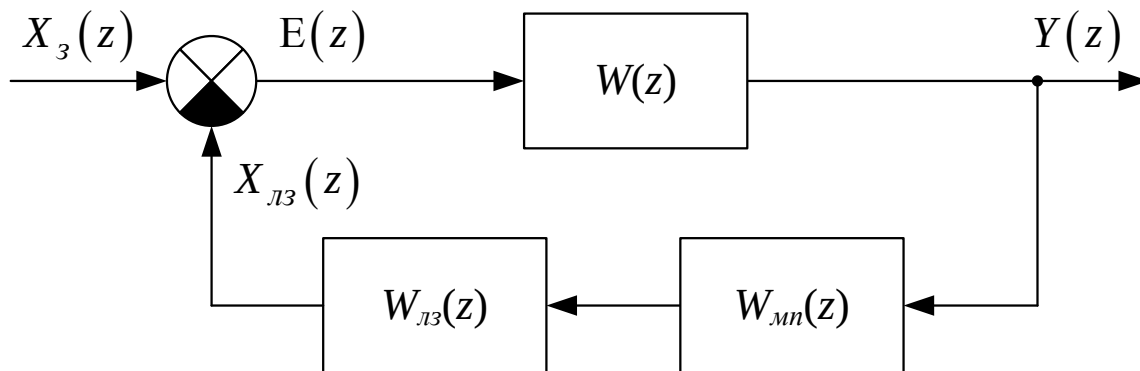


Рисунок 2.10 – Структурна схема детермінованої дискретної ЛДС ЗП, взятої для прикладу, на комплексній площині

Якщо потрібно отримати математичну модель даної системи як перетворювача заданого сигналу $x_3(t)$ у вихідний сигнал $y(t)$ у вигляді

$$Y(z) = W_{nc}(z) \cdot X_3(z), \quad (2.113)$$

де $W_{nc}(z)$ – дискретна передаточна функція системи як перетворювача (рис. 2.11), то з рівнянь (2.104), (2.107), (2.110), (2.112) необхідно вилучити шляхом послідовної підстановки усі проміжні змінні, залишивши лише вхідну координату $X_3(z)$ та вихідну координату $Y(z)$.

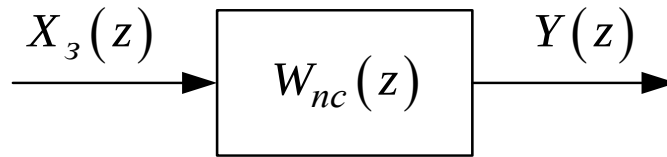


Рисунок 2.11 – Структурна схема взятої для прикладу детермінованої дискретної ЛДС ЗП як перетворювача сигналу на комплексній площині

В результаті цих перетворень отримаємо

$$Y(z) = \frac{W(z)}{1 + W(z) \cdot W_{mn}(z) \cdot W_{лз}(z)} X_3(z). \quad (2.114)$$

Підставляючи в (2.114) вирази для передаточних функцій із (2.105), (2.108), (2.111), виконуючи необхідні перетворення та порівнюючи отриманий результат з виразом (2.113), отримаємо

$$W_{nc}(z) = \frac{b_0 + b_1 z^{-1}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-l} + a_4 z^{-(1+l)} + a_5 z^{-(2+l)} + a_6 z^{-(3+l)}}, \quad (2.115)$$

де:

$$\begin{cases} b_0 = 1 + 2T, \\ b_1 = -2T, \\ a_0 = 1 + 5T + 6T^2, \\ a_1 = -(2 + 5T), \\ a_2 = 1, \\ a_3 = (1 + 2T) \cdot \left(\frac{c_2}{T^2} + \frac{c_1}{T} + c_0 \right), \\ a_4 = - \left(2T \cdot \left(\frac{c_2}{T^2} + \frac{c_1}{T} + c_0 \right) + (1 + 2T) \cdot \left(\frac{2c_2}{T^2} + \frac{c_1}{T} \right) \right), \\ a_5 = \frac{c_2}{T^2} \cdot (1 + 2T) + 2T \cdot \left(\frac{2c_2}{T^2} + \frac{c_1}{T} \right), \\ a_6 = -\frac{2c_2}{T}. \end{cases} \quad (2.116)$$

2.6 Завдання для самоперевірки

Дайте означення решітчастої функції, наведіть приклад із побудовою графіка.

Що є аналогом похідної для решітчастих функцій?

Чому недостатньо лише прямої скінченної різниці для характеристики швидкості зміни решітчастої функції? Який вихід?

Чим відрізняється рівняння в скінченних різницях від різницевого рівняння?

Як за диференціальним рівнянням побудувати різницеве? Наведіть приклад.

Що собою являє дискретне перетворення Лапласа решітчастої функції?

Що таке Z-перетворення решітчастих функцій і чому виникає в ньому потреба? Наведіть приклад Z-перетворення решітчастої функції.

Які є способи отримання решітчастого оригіналу за заданим його Z-зображенням? Наведіть приклад отримання решітчастого оригіналу.

У чому полягає перевага використання обернених скінченних різниць під час застосування Z-перетворення для аналізу дискретних систем?

Для чого потрібен АЦП? Які функціональні блоки входять до його схеми?

Яку функцію виконує ЦАП? Поясніть принцип його дії.

Як отримати математичну модель мікропроцесора? Що слід врахувати під час дослідження мікропроцесорної системи з АЦП та ЦАП?

Побудуйте функціональну схему мікропроцесорної системи управління детермінованим лінійним динамічним об'єктом з віднесенням її до класу дискретних систем.

Як знайти математичну модель неперервного лінійного динамічного об'єкта дискретної системи на комплексній площині?

Як знайти дискретну передаточну функцію неперервного лінійного динамічного об'єкта дискретної системи?

Як побудувати математичну модель мікропроцесорного каналу дискретної системи управління?

Що собою являють математичні моделі АЦП і ЦАП в мікропроцесорній системі?

Що собою являє повна математична модель дискретної детермінованої ЛДС ЗП в часовій області?

Як здійснити аналіз процесів у дискретній динамічній системі за допомогою математичної моделі?

Як отримати дискретну передаточну функцію системи, що в часовій області описується різницеви рівняннями?

Як отримати дискретну передаточну функцію системи, частина елементів якої є неперервними, а частина – дискретними?

Як отримати дискретну передаточну функцію мікропроцесорної системи управління як перетворювача вхідного сигналу у вихідний.

Розділ 3 ФУР'Є-ІНТЕГРАЛЬНИЙ МЕТОД ІДЕНТИФІКАЦІЇ ДЕТЕРМІНОВАНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ

3.1 Вихідні передумови Фур'є-інтегрального методу ідентифікації

Відомо, що сигнал $x(t)$, який діє на вході лінійної динамічної системи з імпульсною перехідною характеристикою $g(t)$ (див. рис. 1.13), та реакція системи $y(t)$ на цей сигнал пов'язані між собою інтегралом згортки (1.70), який для зручності посилення наведемо й у цьому розділі під номером (3.1)

$$y(t) = \int_0^{\infty} x(t-\tau) g(\tau) d\tau \quad (3.1)$$

або у більш загальному вигляді:

$$y = Ax, \quad (3.2)$$

де A – оператор системи.

Якщо моделі сигналів подати у вигляді рядів Фур'є з одним і тим же спектром частот, то кожен гармонічний складову сигналу $x(t)$ можна однозначно пов'язати із гармонічною складовою тієї ж частоти сигналу $y(t)$ алгебраїчним виразом, який містить у собі тільки коефіцієнти Фур'є сигналів $x(t)$, $y(t)$ та спектральні складові АФЧХ $W(j\omega)$ системи на цій же частоті.

Які ж задачі можна розв'язувати за допомогою отриманих алгебраїчних виразів, використання яких у певних комбінаціях та у певній послідовності і являє собою Фур'є-інтегральний метод ідентифікації (ФІМІ) сигналів та систем?

Таких задач дві.

Задача ідентифікації системи. У цій задачі, відштовхнувшись від параметричного аналізу відомих сигналів $x(t)$ та $y(t)$, синтезується такий оператор системи A , який оптимальним чином (за деяким критерієм) віддзеркалює властивості системи як перетворювача сигналу $x(t)$ у сигнал $y(t)$.

Задача ідентифікації (відновлення) вхідного сигналу. Для розв'язання цієї задачі за допомогою ФІМІ синтезуються такі співвідношення, виходячи з яких за відомих вихідним сигналом $y(t)$ і заданим оператором A системи ідентифікуються (відновлюються) характеристики сигналу $x(t)$, який, поділявши на вхід цієї системи, викликав реакцію $y(t)$.

Виконанням же яких умов повинно супроводжуватись використання ФІМІ, або, що одне і те ж, якими ж повинні бути вихідні передумови?

Для лінійних динамічних систем така передумова лише одна – сигнали $x(t)$, $y(t)$ повинні задовольняти умови Діріхле, щоб їх можна було розкласти у ряди Фур'є.

Ця передумова у реальних динамічних системах, запаси енергії в яких завжди обмежені, а тому амплітуди сигналів не можуть бути нескінченними, виконується завжди.

3.2 Синтез алгоритмів параметричної ідентифікації сигналів на вході лінійної вимірювальної системи

У більшості вимірювальних систем мають місце процеси перетворення енергії, якщо не впродовж усього каналу вимірювання, то, принаймні, хоча б на одній з його ділянок, а тому, розв'язуючи задачу синтезу алгоритмів параметричної ідентифікації сигналів на вході вимірювальної системи, залишаємось у визначеній у цьому посібнику області ідентифікації динамічних систем.

Як відомо, у вимірювальних системах використовується лише лінійний відрізок характеристики «вхід – вихід», тобто їх без будь-яких натяжок відносять до класу лінійних динамічних систем.

Зосередженість параметрів вимірювальних систем теж не викликає сумнівів за умови, що до їх складу не відносять канал передачі вимірювальної інформації на значну відстань (про такі канали мова піде в 3-й частині посібника).

Для вимірювальних систем визначеного класу справедливим є інтегральне рівняння згортки (3.1).

Якщо відрізок часу спостереження вихідного сигналу $y(t)$ дорівнює T , його можна подати на цьому відрізку у вигляді ряду Фур'є

$$y(t) = \frac{m_0}{2} + \sum_{i=1}^{\infty} m_i \cos i \omega_1 t + \sum_{i=1}^{\infty} n_i \sin i \omega_1 t. \quad (3.3)$$

Не накладаючи умови – відомий, чи ні, нам вхідний сигнал $x(t)$ – представимо його у вигляді ряду Фур'є на тому ж відрізку часу T :

$$x(t) = \frac{a_0}{2} + \sum_{i=1}^{\infty} a_i \cos i \omega_1 t + \sum_{i=1}^{\infty} b_i \sin i \omega_1 t. \quad (3.4)$$

У виразах (3.3), (3.4)

$$m_i, i = \overline{0, \infty}; \quad n_i, i = \overline{1, \infty}; \quad a_i, i = \overline{0, \infty}; \quad b_i, i = \overline{1, \infty}$$

– коефіцієнти Фур'є сигналів $x(t)$, $y(t)$, які знаходяться за відомими формулами:

$$\begin{Bmatrix} a_i \\ m_i \end{Bmatrix} = \frac{2}{T} \int_0^T \begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix} \cos i \omega_1 t \, dt, \quad i = 0, 1, 2, \dots, \quad (3.5)$$

$$\begin{Bmatrix} b_i \\ n_i \end{Bmatrix} = \frac{2}{T} \int_0^T \begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix} \sin i \omega_1 t \, dt, \quad i = 1, 2, \dots, \quad (3.6)$$

де $\omega_1 = \frac{2\pi}{T}$ – частота першої гармоніки.

Підставляючи ряди (3.3) та (3.4) в інтеграл згортки (3.1), після низки нескладних перетворень, отримаємо

$$\begin{aligned} & \frac{m_0}{2} + \sum_{i=1}^{\infty} m_i \cos i \omega_1 t + \sum_{i=1}^{\infty} n_i \sin i \omega_1 t = \\ & = \frac{a_0}{2} R(0) + \sum_{i=1}^{\infty} (a_i \cdot R(i \omega_1) + b_i \cdot Q(i \omega_1)) \cos i \omega_1 t + \\ & + \sum_{i=1}^{\infty} (b_i \cdot R(i \omega_1) - a_i \cdot Q(i \omega_1)) \sin i \omega_1 t, \end{aligned} \quad (3.7)$$

де $R(i \omega_1)$, $Q(i \omega_1)$ – значення, відповідно, дійсної $R(\omega)$ та уявної $Q(\omega)$ частотних характеристик вимірювальної системи на частотах $i \omega_1$, $i = 0, 1, 2, \dots$.

Оскільки вираз (3.7) – це тотожність, то справедливими є, по-перше, рівняння

$$a_0 R(0) = m_0 \quad (3.8)$$

для постійних складових a_0 , m_0 сигналів $x(t)$, $y(t)$, а, по-друге, система рівнянь

$$\begin{cases} a_i R(i \omega_1) + b_i Q(i \omega_1) = m_i, \\ b_i R(i \omega_1) - a_i Q(i \omega_1) = n_i, \end{cases} \quad (3.9)$$

де $i = 1, 2, \dots$ для всіх інших коефіцієнтів Фур'є a_i, b_i, m_i, n_i сигналів $x(t), y(t)$.

З теорії рядів Фур'є та властивостей уявної частотної характеристики $Q(\omega)$ відомо, що

$$\begin{cases} b_0 = 0, \\ Q(0) = 0, \end{cases} \quad (3.10)$$

а тому рівняння (3.8) теж можна отримати із системи (3.9) при $i = 0$. Тож у подальшому будемо розглядати лише систему рівнянь (3.9), поклавши в ній: $i = 0, 1, 2, \dots$.

Розв'яжемо цю систему відносно a_i, b_i . Отримаємо:

$$a_i = \frac{m_i R(i\omega_1) - n_i Q(i\omega_1)}{R^2(i\omega_1) + Q^2(i\omega_1)}, \quad i = 0, 1, 2, \dots, \quad (3.11)$$

$$b_i = \frac{n_i R(i\omega_1) + m_i Q(i\omega_1)}{R^2(i\omega_1) + Q^2(i\omega_1)}, \quad i = 0, 1, 2, \dots \quad (3.12)$$

Далі розв'язання задачі ідентифікації вхідного сигналу $x(t)$ вимірювальної системи можна здійснювати, використовуючи один із двох підходів.

Під час використання першого з підходів, який назвемо прямим, алгоритм розв'язання задачі містить такі етапи: спочатку за реалізацією вихідного сигналу $y(t)$, зафіксованою на відрізку часу T , за допомогою формул (3.5) і (3.6) визначаємо коефіцієнти Фур'є m_i , n_i цього сигналу для значень $i = \overline{0, N}$. Потім для цих же значень i за заданими частотними характеристиками $R(\omega)$, $Q(\omega)$ вимірювальної системи визначаємо їх значення на частотах $i\omega_1$, тобто $R(i\omega_1)$, $Q(i\omega_1)$. Підставляючи m_i , n_i , $R(i\omega_1)$, $Q(i\omega_1)$ у формули (3.11), (3.12), отримаємо для кожного значення i із множини $\overline{0, N}$ пару коефіцієнтів Фур'є a_i , b_i вхідного сигналу $x(t)$.

Підстановкою всіх цих коефіцієнтів в зрізаний ряд (3.4) і завершуємо розв'язання задачі ідентифікації (відновлення) вхідного сигналу $x(t)$ за вихідним сигналом $y(t)$ вимірювальної системи за допомогою прямого алгоритму.

Продемонструємо розв'язання цієї задачі на прикладі № 9, взятому із нашої роботи [18].

Для створення можливості дати оцінку отриманим результатам розв'язання поставленої задачі конкретизуємо ІВС та математичну модель її вихідного сигналу.

Отже будемо розглядати ІВС для вимірювання температури потоку води чи газу в трубопроводі з терморезисторним первинним перетворювачем, включеним по мостовій схемі, та з підсилювачем сигналу, який виникає в діагоналі мостового перетворювача.

Передаточну функцію такої ІВС можна записати у вигляді

$$W(p) = \frac{K}{T_s p + 1}, \quad (3.13)$$

де T_s – стала часу терморезисторного первинного перетворювача, а K – коефіцієнт підсилення підсилювача сигналу.

Якщо використати терморезистор з захисною діелектричною оболонкою, то його стала часу T_s в залежності від товщини та матеріалу оболонки знаходитиметься в діапазоні значень (1-2) одиниць за секунду. Прийmemo, що

$$T_s = 1 \quad (3.14)$$

Для того, щоб зручно було порівнювати вихідний сигнал $y(t)$ з вхідним сигналом $x(t)$ налаштуємо підсилювач так, щоб мати

$$K = 1. \quad (3.15)$$

Підставляючи умови (3.14) і (3.15) у вираз (3.13), отримаємо конкретне значення передаточної функції терморезисторної ІВС у вигляді

$$W(p) = \frac{1}{p+1}. \quad (3.16)$$

Припустимо, що після включення цієї ІВС в роботу на її виході ми на протязі 10 секунд зафіксували сигнал $y(t)$, математична модель якого має вигляд

$$y(t) = 5\sqrt{t}. \quad (3.17)$$

З використанням виразів (3.16) та (3.17) Python-програма відновлення вхідного сигналу $x(t)$ у **прикладі № 9** буде мати наступний вигляд:

```
In [1]: import sympy
In [2]: from sympy import*
In [3]: p=symbols('p')
In [4]: r,v= symbols('r v',real=True)
In [5]: p=r+v*I
In [6]: W=Function ('W')(p)
In [7]: W=1/(p+1)
In [8]: W_expr=W
In [9]: W1=Function('W1')(v)
In [10]: W1=W_expr.subs(p,r+v*I)
In [11]: W1
Out[11]: 1/(r + I*v + 1)
In [12]: W1_expr=W1
In [13]: W1=W1_expr.subs(r,0)
In [14]: W1
Out[14]: 1/(I*v + 1)
In [15]: R=Function('R')(v)
In [16]: Q=Function('Q')(v)
In [17]: R=re(W1);R
Out[17]: 1/(v**2 + 1)
In [18]: Q=im(W1);Q
Out[18]: -v/(v**2 + 1)
In [19]: k,T,v0=symbols('k T v0')
In [20]: T=10
In [21]: v0=(2*pi/T).evalf(3);v0
Out[21]: 0.628
In [22]: R_expr=R
In [23]: R_k_expr=R_expr.subs(v,k*\
v0).evalf(3);R_k_expr
Out[23]: 1/(0.395*k**2 + 1.0)
In [24]: Q_expr=Q
In [25]: Q_k_expr=Q_expr.subs(v,k*\
v0).evalf(3);Q_k_expr
Out[25]: -0.628*k/(0.395*k**2 + 1.0)

In [26]: R_0_expr=R_k_expr.subs(k,\
0);R_0_expr
Out[26]: 1.00
In [27]: Q_0_expr=Q_k_expr.subs(k,\
0);Q_0_expr
Out[27]: 0
In [28]: R_1_expr=R_k_expr.subs(k,\
1).evalf(3)
In [29]: Q_1_expr=Q_k_expr.subs(k,\
1).evalf(3)
In [30]: R_2_expr=R_k_expr.subs(k,\
2).evalf(3)
In [31]: Q_2_expr=Q_k_expr.subs(k,\
2).evalf(3)
In [32]: R_3_expr=R_k_expr.subs(k,\
3).evalf(3)
In [33]: Q_3_expr=Q_k_expr.subs(k,\
3).evalf(3)
In [34]: R_4_expr=R_k_expr.subs(k,\
4).evalf(3)
In [35]: Q_4_expr=Q_k_expr.subs(k,\
4).evalf(3)
In [36]: R_5_expr=R_k_expr.subs(k,\
5).evalf(3)
In [37]: Q_5_expr=Q_k_expr.subs(k,\
5).evalf(3)
In [38]: R_6_expr=R_k_expr.subs(k,\
6).evalf(3)
In [39]: Q_6_expr=Q_k_expr.subs(k,\
6).evalf(3)
In [40]: R_7_expr=R_k_expr.subs(k,\
7).evalf(3)
In [41]: Q_7_expr=Q_k_expr.subs(k,\
7).evalf(3)
```

```

In [42]: R_8_expr=R_k_expr.subs(k,\
      8).evalf(3)
In [43]: Q_8_expr=Q_k_expr.subs(k,\
      8).evalf(3)
In [44]: R_9_expr=R_k_expr.subs(k,\
      9).evalf(3)
In [45]: Q_9_expr=Q_k_expr.subs(k,\
      9).evalf(3)
In [46]: t=symbols('t')
In [47]: f=Function('f')(v,t)
In [48]: g=Function('g')(v,t)
In [49]: f=cos(v*t)
In [50]: f_expr=f
In [51]: g=sin(v*t)
In [52]: g_expr=g

In [53]: f_k_t_expr=f_expr.subs(v,k*\
      v0).evalf(3);f_k_t_expr
Out[53]: cos(0.628*k*t)
In [54]: g_k_t_expr=g_expr.subs(v,k*\
      v0).evalf(3);g_k_t_expr
Out[54]: sin(0.628*k*t)
In [55]: f_0_t_expr=f_k_t_expr.subs(k,\
      0).evalf(3);f_0_t_expr
Out[55]: 1.00
In [56]: g_0_t_expr=g_k_t_expr.subs(k,\
      0).evalf(3);g_0_t_expr
Out[56]: 0
In [57]: f_1_t_expr=f_k_t_expr.subs(k,\
      1).evalf(3);f_1_t_expr
Out[57]: cos(0.628*t)
In [58]: g_1_t_expr=g_k_t_expr.subs(k,\
      1).evalf(3);g_1_t_expr
Out[58]: sin(0.628*t)
In [59]: f_2_t_expr=f_k_t_expr.subs(k,\
      2).evalf(3);f_2_t_expr
Out[59]: cos(1.26*t)
In [60]: g_2_t_expr=g_k_t_expr.subs(k,\
      2).evalf(3);g_2_t_expr
Out[60]: sin(1.26*t)
In [61]: f_3_t_expr=f_k_t_expr.subs(k,\
      3).evalf(3);f_3_t_expr
Out[61]: cos(1.88*t)

In [62]: g_3_t_expr=g_k_t_expr.subs(k,\
      3).evalf(3);g_3_t_expr
Out[62]: sin(1.88*t)
In [63]: f_4_t_expr=f_k_t_expr.subs(k,\
      4).evalf(3);f_4_t_expr
Out[63]: cos(2.51*t)
In [64]: g_4_t_expr=g_k_t_expr.subs(k,\
      4).evalf(3);g_4_t_expr
Out[64]: sin(2.51*t)
In [65]: f_5_t_expr=f_k_t_expr.subs(k,\
      5).evalf(3);f_5_t_expr
Out[65]: cos(3.14*t)
In [66]: g_5_t_expr=g_k_t_expr.subs(k,\
      5).evalf(3);g_5_t_expr
Out[66]: sin(3.14*t)
In [67]: f_6_t_expr=f_k_t_expr.subs(k,\
      6).evalf(3);f_6_t_expr
Out[67]: cos(3.77*t)
In [68]: g_6_t_expr=g_k_t_expr.subs(k,\
      6).evalf(3);g_6_t_expr
Out[68]: sin(3.77*t)
In [69]: f_7_t_expr=f_k_t_expr.subs(k,\
      7).evalf(3);f_7_t_expr
Out[69]: cos(4.4*t)
In [70]: g_7_t_expr=g_k_t_expr.subs(k,\
      7).evalf(3);g_7_t_expr
Out[70]: sin(4.4*t)

In [71]: f_8_t_expr=f_k_t_expr.subs(k,\
      8).evalf(3);f_8_t_expr
Out[71]: cos(5.03*t)
In [72]: g_8_t_expr=g_k_t_expr.subs(k,\
      8).evalf(3);g_8_t_expr
Out[72]: sin(5.03*t)
In [73]: f_9_t_expr=f_k_t_expr.subs(k,\
      9).evalf(3);f_9_t_expr
Out[73]: cos(5.65*t)
In [74]: g_9_t_expr=g_k_t_expr.subs(k,\
      9).evalf(3);g_9_t_expr
Out[74]: sin(5.65*t)
In [75]: y=Function('y')(t)
In [76]: y=5*t**0.5
In [77]: a=symbols('a:10')
In [78]: b=symbols('b:10')
In [79]: m=symbols('m:10')
In [80]: n=symbols('n:10')

In [81]: m0=((2/T)*integrate(y*f_0_t_expr,(t,0,T))).evalf(3)
In [82]: m1=((2/T)*integrate(y*f_1_t_expr,(t,0,T))).evalf(3)
In [83]: n1=((2/T)*integrate(y*g_1_t_expr,(t,0,T))).evalf(3)
In [84]: m2=((2/T)*integrate(y*f_2_t_expr,(t,0,T))).evalf(3)
In [85]: n2=((2/T)*integrate(y*g_2_t_expr,(t,0,T))).evalf(3)
In [86]: m3=((2/T)*integrate(y*f_3_t_expr,(t,0,T))).evalf(3)

```

```

In [87]: n3=((2/T)*integrate(y*g_3_t_expr,(t,0,T))).evalf(3)
In [88]: m4=((2/T)*integrate(y*f_4_t_expr,(t,0,T))).evalf(3)
In [89]: n4=((2/T)*integrate(y*g_4_t_expr,(t,0,T))).evalf(3)
In [90]: m5=((2/T)*integrate(y*f_5_t_expr,(t,0,T))).evalf(3)
In [91]: n5=((2/T)*integrate(y*g_5_t_expr,(t,0,T))).evalf(3)
In [92]: m6=((2/T)*integrate(y*f_6_t_expr,(t,0,T))).evalf(3)
In [93]: n6=((2/T)*integrate(y*g_6_t_expr,(t,0,T))).evalf(3)
In [94]: m7=((2/T)*integrate(y*f_7_t_expr,(t,0,T))).evalf(3)
In [95]: n7=((2/T)*integrate(y*g_7_t_expr,(t,0,T))).evalf(3)
In [96]: m8=((2/T)*integrate(y*f_8_t_expr,(t,0,T))).evalf(3)
In [97]: n8=((2/T)*integrate(y*g_8_t_expr,(t,0,T))).evalf(3)
In [98]: m9=((2/T)*integrate(y*f_9_t_expr,(t,0,T))).evalf(3)
In [99]: n9=((2/T)*integrate(y*g_9_t_expr,(t,0,T))).evalf(3)
In[100]: a0=m0/R_0_expr
In [101]: a1=(m1*R_1_expr-n1*Q_1_expr)/(R_1_expr**2+Q_1_expr**2)
In [102]: b1=(n1*R_1_expr+m1*Q_1_expr)/(R_1_expr**2+Q_1_expr**2)
In [103]: a2=(m2*R_2_expr-n2*Q_2_expr)/(R_2_expr**2+Q_2_expr**2)
In [104]: b2=(n2*R_2_expr+m2*Q_2_expr)/(R_2_expr**2+Q_2_expr**2)
In [105]: a3=(m3*R_3_expr-n3*Q_3_expr)/(R_3_expr**2+Q_3_expr**2)
In [106]: b3=(n3*R_3_expr+m3*Q_3_expr)/(R_3_expr**2+Q_3_expr**2)
In [107]: a4=(m4*R_4_expr-n4*Q_4_expr)/(R_4_expr**2+Q_4_expr**2)
In [108]: b4=(n4*R_4_expr+m4*Q_4_expr)/(R_4_expr**2+Q_4_expr**2)
In [109]: a5=(m5*R_5_expr-n5*Q_5_expr)/(R_5_expr**2+Q_5_expr**2)
In [110]: b5=(n5*R_5_expr+m5*Q_5_expr)/(R_5_expr**2+Q_5_expr**2)
In [111]: a6=(m6*R_6_expr-n6*Q_6_expr)/(R_6_expr**2+Q_6_expr**2)
In [112]: b6=(n6*R_6_expr+m6*Q_6_expr)/(R_6_expr**2+Q_6_expr**2)
In [113]: a7=(m7*R_7_expr-n7*Q_7_expr)/(R_7_expr**2+Q_7_expr**2)
In [114]: b7=(n7*R_7_expr+m7*Q_7_expr)/(R_7_expr**2+Q_7_expr**2)
In [115]: a8=(m8*R_8_expr-n8*Q_8_expr)/(R_8_expr**2+Q_8_expr**2)
In [116]: b8=(n8*R_8_expr+m8*Q_8_expr)/(R_8_expr**2+Q_8_expr**2)
In [117]: a9=(m9*R_9_expr-n9*Q_9_expr)/(R_9_expr**2+Q_9_expr**2)
In [118]: b9=(n9*R_9_expr+m9*Q_9_expr)/(R_9_expr**2+Q_9_expr**2)
In [119]: x=Function('x')(t)
In [120]: x=a0/2+a1*f_1_t_expr+b1*g_1_t_expr+a2*f_2_t_expr+\
          b2*g_2_t_expr+ a3*f_3_t_expr+b3*g_3_t_expr+a4*f_4_t_expr+\
          b4*g_4_t_expr+ a5*f_5_t_expr+b5*g_5_t_expr+a6*f_6_t_expr+\
          b6*g_6_t_expr+ a7*f_7_t_expr+b7*g_7_t_expr+a8*f_8_t_expr+\
          b8*g_8_t_expr+ a9*f_9_t_expr+b9*g_9_t_expr;x
Out[120]:
-3.26*sin(0.628*t) - 1.64*sin(1.26*t) - 1.06*sin(1.88*t) - 0.767*sin(2.51*t) - 0.587*sin(3.14*t) -
0.463*sin(3.77*t) - 0.399*sin(4.4*t) - 0.322*sin(5.03*t) - 0.27*sin(5.65*t) - 3.26*cos(0.628*t) -
2.95*cos(1.26*t) - 2.91*cos(1.88*t) - 2.9*cos(2.51*t) - 2.91*cos(3.14*t) - 2.92*cos(3.77*t) -
2.92*cos(4.4*t) - 2.93*cos(5.03*t) - 2.94*cos(5.65*t) + 10.5
In [121]: from sympy.plotting import plot
In [122]: gr1=plot(y,(t,0,10),show=False,line_color='g')
In [123]: gr2=plot(x,(t,0,10),show=False,line_color='r')
In [124]: gr1.extend(gr2)
In [125]: gr1.show()

```

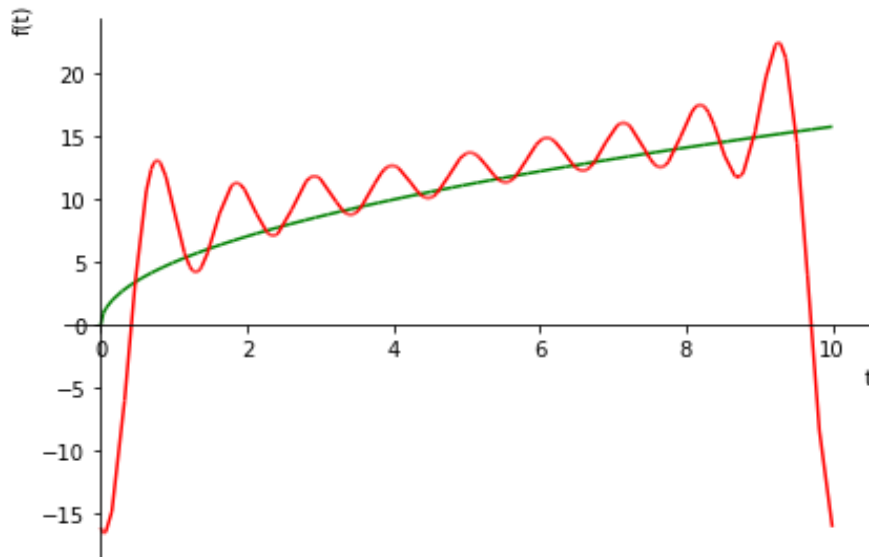


Рисунок 3.1 – Графіки сигналів на виході ІВС (плавна крива, що починається з нуля) з передаточною функцією, заданою виразом (3.16), та на її вході (коливальна крива, складена з дев'яти гармонійних складових)

```

In [126]: x_expr=x
In [127]: x_k_expr=x_expr.subs(t,k)
In [128]: x_0_expr=x_k_expr.subs(k,\
0);x_0_expr
Out[128]: -16.1
In [129]: x_1_expr=x_k_expr.subs(k,\
1);x_1_expr
Out[129]: 9.30
In [130]: x_2_expr=x_k_expr.subs(k,\
2);x_2_expr
Out[130]: 10.5
In [131]: x_3_expr=x_k_expr.subs(k,\
3);x_3_expr
Out[131]: 11.7
In [132]: x_4_expr=x_k_expr.subs(k,\
4);x_4_expr
Out[132]: 12.8
In [133]: x_5_expr=x_k_expr.subs(k,\
5);x_5_expr
Out[133]: 13.8
In [134]: x_6_expr=x_k_expr.subs(k,\
6);x_6_expr
Out[134]: 14.7
In [135]: x_7_expr=x_k_expr.subs(k,\
7);x_7_expr
Out[135]: 15.6
In [136]: x_8_expr=x_k_expr.subs(k,\
8);x_8_expr
Out[136]: 16.3
In [137]: x_9_expr=x_k_expr.subs(k,\
9);x_9_expr
Out[137]: 16.9

In [138]: print(x_0_expr,x_1_expr,x_2_expr,x_3_expr,x_4_expr,x_5_expr,
x_6_expr,x_7_expr,x_8_expr,x_9_expr)
-16.1 9.30 10.5 11.7 12.8 13.8 14.7 15.6 16.3 16.9

In [139]: import numpy as np
In [140]: import matplotlib
In [141]: import matplotlib.pyplot\
as plt
In [142]: import scipy

In [143]: import scipy.linalg as la
In [144]: from scipy.optimize\
import curve_fit
In [145]: def model(t,A,a):
return A*(a*t)**0.5

In [146]: t=np.array([0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0])
In [147]: z=np.array([-16.1,9.30,10.5,11.7,12.8,13.8,14.7,15.6,16.3,16.9])
In [148]: initial_guess=[5.0,1.0]
In [149]: params=curve_fit(model,t,z,p0=initial_guess)

```



```

In [150]: A=params[0][0]
In [151]: A=A.round(3);A
Out[151]: 6.09
In [152]: a=params[0][1]
In [153]: a=a.round(3);a
Out[153]: 1.007
In [154]: x1=model(t,A,a)
Out[154]:
array([ 0.        ,  6.11127783,  8.64265199, 10.5850437 , 12.22255566,
        13.66523265, 14.96951236, 16.16892133, 17.28530398, 18.33383349])
In [155]: def y(t):
            return 5*t**0.5

In [156]: yvec=np.vectorize(y)
In [157]: y1=yvec(t); y1
Out[157]:
array([ 0.        ,  5.        ,  7.07106781,  8.66025404, 10.        ,
        11.18033989, 12.24744871, 13.22875656, 14.14213562, 15.        ])
In [158]: plt.plot(t,y1,'-g',t,x1,'-r',linewidth=4)
Out[158]:
[<matplotlib.lines.Line2D at 0x1b4511f97c0>,
 <matplotlib.lines.Line2D at 0x1b4511f9880>]

```

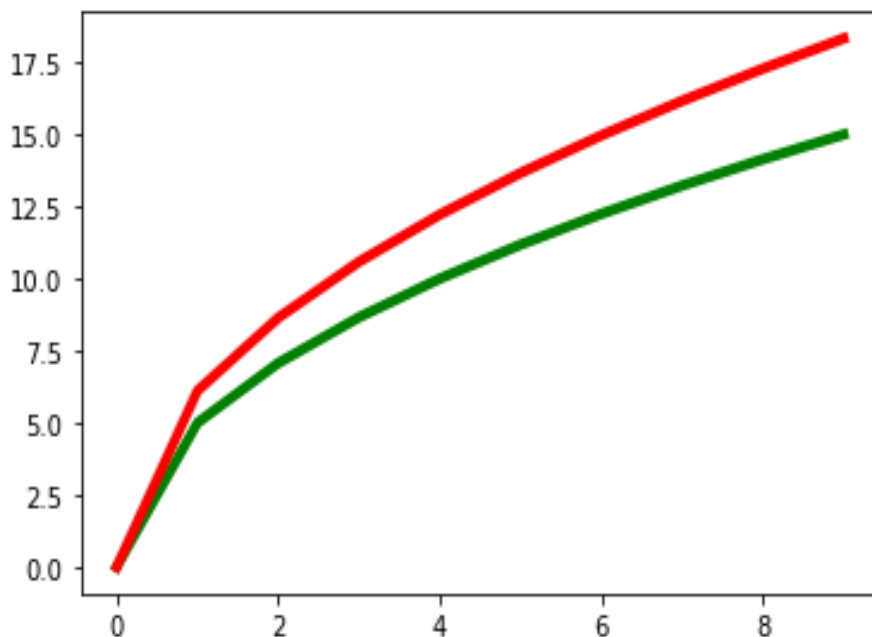


Рисунок 3.2 – Графіки сигналів на виході ІВС (нижня крива) та на її вході (верхня крива, складена з усіх гармонійних складових)

Відзначимо, що у приведеній вище Python-програмі, покладеній в основу ІТ відновлення вхідних сигналів ІВС за ФІМІ з використанням вихідних сигналів:

командою In [7] в програму вводиться передаточна функція $W(p)$;
командою In [17] обчислюється дійсна частотна характеристика $R(\omega)$, в якій для зручності грецька ω замінена на латинську ν ;

командою In [18] обчислюється уявна частотна характеристика $Q(\omega)$, в якій для зручності грецька ω замінена на латинську v ;

командою In [20] в програму вводиться відрізок часу T , протягом якого фіксується реалізація вихідного сигналу IBC (в тих же одиницях часу, що й стала часу T_s IBC);

командою In [21] обчислюється частота першої гармонійної складової ω_0 , яка в програмі позначена для зручності v_0 ;

командами In [22]- In [45] обчислюються значення дійсної та уявної частотних характеристик IBC на частотах kv_0 ($k = 0,1, \dots, 9$);

командами In [49]- In [74] створюються базові функції гармонійних складових;

командою In [76] в програму вводиться вихідний сигнал $y(t)$ IBC;

командами In [81]- In [99] обчислюються значення коефіцієнтів Фур'є вихідного сигналу $y(t)$ IBC;

командами In [100]- In [118] обчислюються значення коефіцієнтів Фур'є вхідного сигналу $x(t)$ IBC;

командою In [120] формується із 9 гармонійних складових відновлений вхідний сигнал $x(t)$ IBC;

командами In [121]- In [125] створюються графіки вихідного сигналу $y(t)$ та відновленого 9 гармонійними складовими вхідного сигналу $x(t)$ IBC, які приведені на рис. 3.1;

командами In [144]- In [153] з застосуванням МНК визначається математична модель відновленого вхідного сигналу $x(t)$ IBC, еквівалентна нескінченній кількості гармонійних складових;

командами In [154]- In [168] створюються графіки вихідного сигналу $y(t)$ та відновленого еквівалентною моделлю вхідного сигналу $x(t)$ IBC, які приведені на рис. 3.2.

Здійснимо аналіз отриманих результатів.

Першим, що ми хочемо пояснити в нашому аналізі отриманих результатів, це те, чому складено Python-програму реалізації ФІМІ в задачі відновлення вхідних сигналів $x(t)$ IBC з використанням командних рядків, а не у формі файлу, та чому блоки команд In [22]- In [45], In [49]- In [74], In [81]- In [99], In [100]- In [118] не реалізовані у вигляді **for**-циклів, що привело б до зменшення кількості використаних команд у цій програмі.

Пояснення наше буде таким: саме завдяки командним рядкам структура Python-програми є повністю адекватною структурі ФІМІ, і результати кожного обчислення за співвідношеннями цього методу легко проконтролювати на кожному кроці виконання програми, чого не можна досягти в разі реалізації програми файлом, особливо ж з застосуванням у ній **for**-циклів. А необхідність супроводжувати кожний цикл у файлі програми ще й командою `print( )` з пошуком потрібної інформації в загальній роздруківці створює додаткові незручності при файлової реалізації програми з застосуванням **for**-циклів. Але кожен, хто більше звик до роботи з файловою реалізацією комп'ютерних

програм і побажає використати нашу Python-програму реалізації ФІМІ в задачі відновлення вхідних сигналів $x(t)$ ІВС, складену з використанням командних рядків, легко зможе її самостійно трансформувати у форму файлу з **for**-циклами.

Другим, на що ми хочемо звернути увагу в нашому аналізі отриманих результатів, це те, що відновлений в процесі застосування ІТ на основі ФІМІ вхідний сигнал $x(t)$ ІВС у вигляді ряду Фур'є у формі (3.4) навіть з використанням лише 9 гармонійних складових суттєво відрізняється від вихідного сигналу $y(t)$, про що однозначно свідчать їх графіки, приведені на рис. 3.1. І чим більшим буде значення сталої часу первинного вимірювального перетворювача, тим більші динамічні спотворення проявлятимуться у вихідному сигналі ІВС. У цьому легко переконатись якщо командою In [7] в програму ввести не передаточну функція $W(p)$, задану виразом (3.16), а задану виразом

$$W(p) = \frac{1}{0,2p+1}, \quad (3.18)$$

в якому стала часу ϵ у 5 разів меншою.

Реалізуючи нашу ІТ з приведеною вище Python-програмою і використанням передаточної функції (3.18), ми отримаємо графік відновленого вхідного сигналу, приведений на рис. 3.3

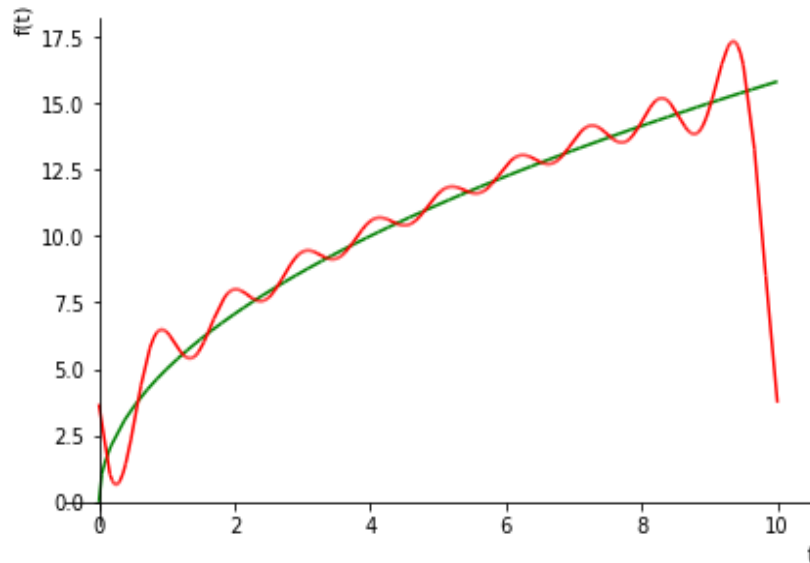


Рисунок 3.3 – Графіки сигналів на виході ІВС (плавна крива, що починається з нуля) з передаточною функцією, заданою виразом (3.18), та на її вході (коливальна крива, складена з дев'яти гармонійних складових)

Порівнюючи графіки відновленого вхідного сигналу, приведені на рисунках 3.1 та 3.3, бачимо, що графік відновленого вхідного сигналу ІВС з передаточною функцією (3.18) менше відхиляється по тренду від вихідного сигналу цієї ІВС у порівнянні з графіком відновленого вхідного сигналу з передаточною функцією (3.16).

Ну, а третім, на що ми хочемо звернути увагу в нашому аналізі отриманих результатів, це те, що відновлений в процесі застосування ІТ на основі ФІМІ вхідний сигнал $x(t)$ ІВС у вигляді ряду Фур'є у формі (3.4) з використанням лише 9 гармонійних складових за допомогою метода найменших квадратів (МНК) нами трансформовано у ряд Фур'є з нескінченною кількістю гармонійних складових. Це обумовлено тим, що додавання кожної наступної гармонійної складової «згладжує» сумарну криву вхідного сигналу, наближаючи її до тренду, для ідентифікації якого логічно застосувати той же вираз (3.17), яким задана модель вихідного сигналу, але з невідомими коефіцієнтами A, a , тобто, задавати тренд вхідного сигналу, який ми будемо трактувати, як його еквівалентну модель, у вигляді

$$x(t) = A\sqrt{at}. \quad (3.19)$$

А для визначення коефіцієнтів еквівалентної моделі (3.19) відновленого вхідного сигналу ми використали нелінійний варіант МНК, реалізувавши його в нашій Python-програмі командами `In [144]- In [153]`, що дало змогу ідентифікувати вхідний сигнал $x(t)$ ІВС у вигляді еквівалентної моделі

$$x(t) = 6,09\sqrt{1,007t}. \quad (3.20)$$

У цьому алгоритмі залишилось нез'ясованим одне важливе питання – а яку ж кількість гармонічних складових необхідно взяти в зрізаному ряді (3.4)?

Іншими словами, яку потужність N повинна мати множина коефіцієнтів Фур'є параметрично ідентифікованого сигналу $x(t)$?

Для отримання відповіді на це питання використаємо встановлену математиками закономірність, яка полягає у тому, що коефіцієнти Фур'є довільного сигналу спадають у порядку, обернено пропорційному своєму номеру. Це означає, що спектр Фур'є сигналу складають лише ті коефіцієнти, які задовольняють цю властивість.

Всі інші коефіцієнти Фур'є, що мають номери вищі від того, для якого порушується вищеозначена умова, складають спектр завади.

Таким чином, число N гармонічних складових у вхідному сигналі $x(t)$ необхідно залишити, виходячи з того, на якому коефіцієнті починає порушуватись умова обернено пропорційного спаду коефіцієнтів Фур'є за порядком величини відносно свого номера.

Розглянутий прямий алгоритм ідентифікації вхідного сигналу $x(t)$ вимірювальної системи не можна віднести до класу оптимальних, оскільки під час його побудови не був використаний ніякий критерій оптимальності. Крім того, запропонований спосіб вибору потужності N множини коефіцієнтів Фур'є сигналу $x(t)$ дає, у загальному випадку, «розмите» значення цього числа.

Більш ефективний розв'язок поставленої задачі можна отримати, використовуючи другий підхід, який базується на алгоритмі оптимального відновлення сигналу $x(t)$.

Під час побудови оптимального алгоритму задачу відновлення також будемо розв'язувати, виходячи з рівнянь (3.9), але скористаємось ними не для отримання розрахункових співвідношень (3.11), (3.12), а для побудови критеріїв оптимальності:

$$\Sigma^m = \sum_{i=0}^l (m_i - a_i R(i\omega_1) - b_i Q(i\omega_1))^2, \quad (3.21)$$

$$\Sigma^n = \sum_{i=0}^l (n_i - b_i R(i\omega_1) + a_i Q(i\omega_1))^2. \quad (3.22)$$

Із цих виразів видно, що коефіцієнти a_i , b_i входять в кожне із критеріальних співвідношень (3.21), (3.22), а тому одне з них можна використати для чисельного визначення значень коефіцієнтів a_i , b_i вхідного сигналу $x(t)$, а друге – для визначення їх оптимальної кількості N .

Покажемо, як це зробити.

Візьмемо частинні похідні за a_i , b_i , $i = \overline{0, l}$ від Σ^m і прирівняємо їх нулю. Отримаємо систему $2l + 1$ нормальних рівнянь Гауса

$$\begin{cases} \frac{\partial \Sigma^m}{\partial a_i} = 0, & i = \overline{0, l}; \\ \frac{\partial \Sigma^m}{\partial b_i} = 0, & i = \overline{1, l} \end{cases} \quad (3.23)$$

для визначення $2l + 1$ оцінок коефіцієнтів a_0 ; a_1 , b_1 ; ... ; a_l , b_l , які позначимо індексними зірочками, тобто

$$a_0^*; a_1^*, b_1^*; a_2^*, b_2^*; \dots; a_l^*, b_l^*. \quad (3.24)$$

Розв'язавши систему алгебраїчних рівнянь (3.23) одним із стандартних методів, отримаємо чисельні значення оцінок вищезначених коефіцієнтів.

Підставивши знайдені шляхом розв'язання системи (3.23) оцінки коефіцієнтів (3.24) у вираз (3.22), в який підставимо також знайдені раніше коефіцієнти Фур'є n_i , $i = \overline{0, l}$ вихідного сигналу $y(t)$, визначимо чисельне значення Σ^n , яке позначимо Σ_l^n .

Використовуючи ще один коефіцієнт Фур'є m_{l+1} вихідного сигналу $y(t)$ додамо до системи рівнянь (3.23) ще два рівняння

$$\frac{\partial \Sigma^m}{\partial a_{l+1}} = 0, \quad \frac{\partial \Sigma^m}{\partial b_{l+1}} = 0. \quad (3.25)$$

Розв'язавши систему рівнянь (3.23), (3.25), отримаємо нові оцінки a_i^{**} , $i = \overline{0, l}$; b_i^{**} , $i = \overline{1, l}$ коефіцієнтів a_i , b_i , а також оцінки a_{l+1}^{**} , b_{l+1}^{**} коефіцієнтів a_{l+1} , b_{l+1} , які раніше не визначались.

Підставивши знайдені тепер уже шляхом розв'язання системи рівнянь (3.23), (3.25) оцінки всіх отриманих коефіцієнтів Фур'є вхідного сигналу $x(t)$ та значення коефіцієнтів n_i , $i = \overline{0, l+1}$ вихідного сигналу $y(t)$ у вираз (3.22), в якому верхню границю суми замінено з l на $l+1$, отримаємо нове чисельне значення Σ^n , яке позначимо Σ_{l+1}^n .

Повторюючи наведену вище процедуру для $l+2$, $l+3$, ..., $l+q$, отримаємо відрізок числового ряду

$$\Sigma_l^n, \Sigma_{l+1}^n, \Sigma_{l+2}^n, \dots, \Sigma_{l+q}^n, \quad (3.26)$$

найменший член якого

$$\Sigma_{l+p}^n = \min \quad (3.27)$$

визначає, згідно з основною ідеєю методу найменших квадратів, оптимальну кількість

$$N_{onm} = l + p \quad (3.28)$$

членів ряду Фур'є у виразі (3.4), яким задається математична модель вхідного сигналу $x(t)$.

У цій процедурі залишається нез'ясованим тільки одне питання: як вибрати значення l , з якого повинен починатись відрізок ряду (3.26)?

Для відповіді на це питання скористаємось критерієм спаду коефіцієнтів

$$c_k = \sqrt{m_k^2 + n_k^2} \quad (3.29)$$

вихідного сигналу $y(t)$ за порядком, обернено пропорційним своєму номеру k , про який вже йшла мова під час розгляду прямого алгоритму розв'язання задачі параметричної ідентифікації вхідного сигналу $x(t)$.

Визначивши номер k коефіцієнта Фур'є c_k , для якого порушується вищевизначена умова, необхідно взяти значення l на 2–3 одиниці менше, тобто

$$l = k - 2(3). \quad (3.30)$$

Таке зміщення вліво початку процедури пошуку оптимальної кількості N_{onm} членів ряду Фур'є у моделі сигналу $x(t)$ забезпечує надійний і однозначний розв'язок задачі оптимального відновлення цього сигналу на множині коефіцієнтів Фур'є, які параметрично визначають даний сигнал.

3.3 Побудова алгоритму параметричної ідентифікації лінійної динамічної системи за допомогою ФІМІ

Якщо задача відновлення сигналів $x(t)$ та $y(t)$, які фіксуються, відповідно, на вході та на виході лінійної динамічної системи, розв'язана за допомогою ФІМІ, то найбільш ефективним методом синтезу параметричної математичної моделі цієї системи є також ФІМІ.

Ефективність цього методу обумовлена перш за все тим, що коефіцієнти Фур'є вихідних сигналів $x(t)$, $y(t)$ інформаційно-вимірювальних систем, встановлених на вході і виході динамічної системи, що ідентифікується, які відновлюються за допомогою ФІМІ, можуть бути безпосередньо використаними і під час розв'язання задачі ідентифікації цієї ж динамічної системи цим же методом, а тому і етап відновлення сигналів, і етап ідентифікації системи можуть бути реалізовані в одній і тій ж мікропроцесорній системі обробки вимірювальної інформації. Це суттєво зменшує кількість апаратних засобів, необхідних для реалізації обох задач, та час, який витрачається на розв'язання, особливо у випадках, коли для обчислення коефіцієнтів Фур'є використовуються методи швидкого перетворення Фур'є.

Всі перетворення (до отримання виразів (3.9), (3.10) включно), здійснені у попередньому підрозділі, залишаються справедливими і під час побудови алгоритму ідентифікації динамічної системи, для якої $x(t)$ є вхідним сигналом, а $y(t)$ – вихідним.

Із системи рівнянь (3.9), (3.10) випливає, що

$$R(i\omega_1) = \frac{a_i m_i + b_i n_i}{a_i^2 + b_i^2}, \quad i = 0, 1, 2, \dots, \quad (3.31)$$

$$Q(i\omega_1) = \frac{b_i m_i - a_i n_i}{a_i^2 + b_i^2}, \quad i = 0, 1, 2, \dots. \quad (3.32)$$

Отриманням формул (3.31), (3.32) закінчується перший етап побудови алгоритму параметричної ідентифікації динамічної системи за допомогою ФІМІ.

На другому етапі будемо шукати математичну модель системи, дійсна $R(\omega)$ та уявна $Q(\omega)$ частотні характеристики якої в окремих точках визначаються співвідношеннями (3.31), (3.32) у класі передаточних функцій виду

$$W_{ls}(p) = \frac{\sum_{v=0}^l \psi_v p^v}{\sum_{\mu=0}^s \chi_\mu p^\mu}, \quad (3.33)$$

де ψ_ν , χ_μ – параметри, числові значення яких потребують визначення у процесі ідентифікації. Нагадаємо, що для динамічних систем, які можна фізично реалізувати, завжди

$$l \leq s. \quad (3.34)$$

Критеріальне співвідношення

$$\Sigma_{ls}^R = \sum_{i=0}^N (R(i\omega_1) - R_{ls}(i\omega_1))^2 \quad (3.35)$$

будемо використовувати для пошуку оптимальних числових значень параметрів ψ_ν , χ_μ передаточних функцій $W_{ls}(p)$ вибраної структури, тобто, за заданих значень l і S , а критеріальне співвідношення

$$\Sigma_{ls}^Q = \sum_{i=0}^N (Q(i\omega_1) - Q_{ls}(i\omega_1))^2 \quad (3.36)$$

– для оптимізації процесу вибору структури цієї передаточної функції, тобто, для визначення оптимальних значень l і S .

У критеріальних співвідношеннях (3.35), (3.36) $R_{ls}(i\omega_1)$, $Q_{ls}(i\omega_1)$ – функціонально задані, виходячи з вибраної структури передаточної функції $W_{ls}(p)$, у точках $i\omega_1$ ($i = 0, 1, 2, \dots$) вирази для дійсної $R_{ls}(\omega)$ і уявної $Q_{ls}(\omega)$ частотних характеристик об'єкта, що ідентифікується, для отримання яких слід використати відомі з теорії автоматичного керування та першого розділу цього посібника співвідношення:

$$\begin{cases} R_{ls}(\omega) = \operatorname{Re} W_{ls}(p) \big|_{p=j\omega}, \\ Q_{ls}(\omega) = \operatorname{Im} W_{ls}(p) \big|_{p=j\omega}, \end{cases} \quad (3.37)$$

а $R(i\omega_1)$, $Q(i\omega_1)$ – розв'язки рівнянь (3.31), (3.32).

Використовуються співвідношення (3.31) – (3.37) у процесі параметричної ідентифікації динамічної системи у такій послідовності: задаючись у (3.33) різними конкретними значеннями l і S так, щоб виконувалась умова фізичної реалізованості (3.34), для кожної конкретної пари значень l і S за співвідношеннями (3.37) визначаємо функції $R_{ls}(\omega)$ та $Q_{ls}(\omega)$. У результаті цієї операції отримуємо множину пар частотних характеристик

$$\begin{aligned} &R_{01}(\omega), Q_{01}(\omega); R_{11}(\omega), Q_{11}(\omega); \\ &R_{02}(\omega), Q_{02}(\omega); R_{12}(\omega), Q_{12}(\omega); R_{22}(\omega), Q_{22}(\omega); \\ &R_{03}(\omega), Q_{03}(\omega); R_{13}(\omega), Q_{13}(\omega); \dots \end{aligned} \quad (3.38)$$

Підставивши по черзі функції $R_{01}(\omega)$, $R_{11}(\omega)$, $R_{02}(\omega)$, $R_{12}(\omega)$, ... із множини (3.38), задані у точках $i\omega_1$, разом із значеннями $R(i\omega_1)$, взятими з (3.31), у критеріальне співвідношення (3.35), і здійснивши відносно (3.35) стандартну процедуру методу найменших квадратів, визначимо оптимальні за критерієм мінімуму суми квадратів відхилень оцінки коефіцієнтів ψ_ν , χ_μ передаточних функцій виду (3.33) для кожної конкретної пари значень l і S із наперед заданого діапазону.

Для того, щоб оптимізувати структуру передаточної функції $W_{ls}(p)$ системи, що ідентифікується, підставимо найдені з використанням функції $R_{ls}(\omega)$ за допомогою методу найменших квадратів числові значення коефіцієнтів ψ_ν , χ_μ у відповідні вирази для $Q_{ls}(\omega)$ із множини пар (3.38).

Конкретизовані у такий спосіб функції $Q_{ls}(\omega)$, у свою чергу, підставимо у друге критеріальне співвідношення (3.36). Після підстановки у (3.36) також і значень $Q(i\omega_1)$, отриманих із (3.32), і прямих обчислень значень Σ_{ls}^Q побудуємо числовий ряд

$$\Sigma_{01}^Q, \Sigma_{11}^Q, \Sigma_{02}^Q, \Sigma_{12}^Q, \Sigma_{22}^Q, \dots \quad (3.39)$$

Виходячи із основної ідеї методу найменших квадратів, можна стверджувати, що та пара характеристик $R_{ls}(\omega)$, $Q_{ls}(\omega)$, яка задає найменший член

$$\Sigma_{ls}^Q = \min_{ls} \quad (3.40)$$

ряду (3.39), і визначає оптимальну структуру математичної моделі динамічної системи, що ідентифікується.

Під час розгляду цього алгоритму привертає до себе увагу той факт, що на відміну від оптимального алгоритму відновлення сигналу, побудованого у попередньому підрозділі, в ньому не виникає запитання про кількість гармонічних складових, які треба утримувати в зрізаних рядах Фур'є сигналів $x(t)$, $y(t)$, оскільки відповідь на це питання однозначна – чим більше, тим краще, але кількість їх обов'язково повинна бути більшою від числа коефіцієнтів передаточної функції $W_{ls}(p)$, які нам потрібно визначити у процесі ідентифікації системи.

Але у цій задачі потрібно мати відповідь на інше запитання, а саме: за яких умов задача ідентифікації лінійної динамічної системи за допомогою ФІМІ може бути віднесена до класу коректних і мати однозначний розв'язок?

Для відповіді на це запитання нагадаємо, що задача ідентифікації лінійної динамічної системи у частотній області з використанням інформації про вхідний $x(t)$ та вихідний $y(t)$ сигнали може бути розв'язана лише за умови, що спектр Фур'є вхідного сигналу є ширшим частотного спектра системи. У

цьому випадку кожна частота спектра системи може бути збуджена відповідною складовою спектра вхідного сигналу і у тій чи іншій мірі проявити себе у спектрі вихідного сигналу.

Якщо ж частотний спектр вхідного сигналу є вузьким, ніж спектр системи, то деякі частоти спектра системи не будуть збуджені і не проявлять себе у спектрі вихідного сигналу, а тому вони не зможуть бути ідентифікованими жодним із частотних методів.

Тож, як висновок із наведеного вище аналізу, впливає, що задача параметричної ідентифікації лінійної динамічної системи за допомогою ФІМІ розв'язується однозначно лише тоді, коли коефіцієнти Фур'є вихідного сигналу цієї системи спадають швидше коефіцієнтів Фур'є її вхідного сигналу.

3.4 Експурс у метод найменших квадратів

Під час реалізації алгоритмів оптимальної параметричної ідентифікації сигналів та систем, розглянутих у попередніх підрозділах цього розділу, використано метод найменших квадратів (МНК), ідея якого належить Гаусу.

І хоча виклад основ МНК можна знайти у посібниках із математичної статистики та обробки результатів експериментів, доцільно подати їх у цьому посібнику також. Оскільки, по-перше, його використання є невід'ємною частиною Фур'є-інтегрального методу ідентифікації, змістом якого заповнено саме цей розділ, а по-друге, використання МНК має певні обмеження, на які автори багатьох посібників не звертають увагу студентів, що у подальшому може стати джерелом помилок в їх дослідженнях, обробка результатів яких здійснюється з використанням МНК. В цьому посібнику ці обмеження не лише показано, але й висвітлено можливі наслідки їх ігнорування.

Після цього коротенького вступу перейдемо до викладу безпосередньо МНК.

Нехай відомо, що вихідний параметр процесу, який вивчається, позначимо його y , лінійно залежить від вхідного параметра x (суцільна пряма лінія на рис. 3.1).

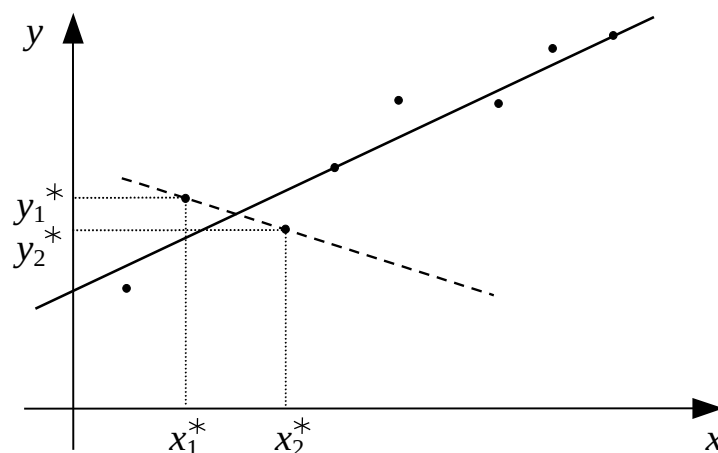


Рисунок 3.4 – Графічна інтерпретація причин, які обумовлюють необхідність використання МНК

Тобто припустимо, що статична характеристика цього процесу може бути подана у вигляді

$$y = ax + b, \quad (3.41)$$

де a і b – коефіцієнти, для визначення числових значень яких необхідно, як мінімум, задати два значення x_1, x_2 вхідній величині x і заміряти відповідні їм значення y_1, y_2 вихідної величини y , оскільки лише під час виконання цих умов для моделі (3.41) можна скласти систему двох алгебраїчних рівнянь із двома невідомими a і b

$$\begin{cases} y_1 = ax_1 + b, \\ y_2 = ax_2 + b. \end{cases} \quad (3.42)$$

Але результати будь-яких експериментальних вимірювань несуть у собі похибки, обумовлені класом точності вимірювальних засобів, дією різноманітних завад, неточністю зчитування показів приладів, округленням під час приведення даних до однакових умов обробки інформації – список умов виникнення похибок можна продовжити, але для обґрунтування МНК цього досить.

Тож через наявність цих похибок в експериментальних значеннях x_1, x_2, y_1, y_2 безпосередній розв’язок системи рівнянь (3.42) відносно a та b може нести в собі похибку в 10, 100, 1000 і більше відсотків.

Наприклад, якщо використати лише значення $x_1^*, y_1^*; x_2^*, y_2^*$ (рис. 3.4) для розв’язання системи рівнянь (3.42), то похибка буде вже не у відсотках, а у характері функціональної залежності (пунктирна лінія на рис. 3.4).

У свій час Гаус запропонував інший спосіб визначення коефіцієнтів a, b моделі (3.41). Він запропонував сформулювати суму квадратів різниць $\sum^N$ між теоретично заданими за допомогою рівняння (3.41) значеннями вихідної координати y при значеннях аргументу $x_i, i = \overline{1, N}$ та її експериментальними значеннями y_i :

$$\Sigma^N = \sum_{i=1}^N (y(x_i) - y_i)^2, \quad (3.43)$$

а потім знайти такі значення коефіцієнтів a, b рівняння (3.41), котрі мінімізують вираз (3.43).

Від цієї процедури і назва методу – метод найменших квадратів.

З курсу математичного аналізу відомо, що для знаходження мінімуму якоїсь функції необхідно взяти від неї похідну, прирівняти цю похідну до нуля і розв’язати отримане рівняння – його корінь задає значення аргументу, за якого функція досягає мінімуму, а само значення функції у цій точці, якщо вона опукла донизу, задає її мінімальне значення.

Згідно з цією ідеєю, підставимо у вираз (3.43) замість $y(x_i)$ його значення з (3.41) і візьмемо від отриманого виразу частинні похідні за b та a , які прирівняємо до нуля, тобто, синтезуємо вирази:

$$\Sigma^N = \sum_{i=1}^N (ax_i + b - y_i)^2, \quad (3.44)$$

$$\begin{cases} \frac{\partial \Sigma^N}{\partial b} = \sum_{i=1}^N 2 (ax_i + b - y_i)(1) = 0, \\ \frac{\partial \Sigma^N}{\partial a} = \sum_{i=1}^N 2 (ax_i + b - y_i)(x_i) = 0. \end{cases} \quad (3.45)$$

Із виразу (3.45) після низки нескладних перетворень отримаємо:

$$\begin{cases} b \cdot N + a \sum_{i=1}^N x_i = \sum_{i=1}^N y_i, \\ b \sum_{i=1}^N x_i + a \sum_{i=1}^N x_i^2 = \sum_{i=1}^N y_i x_i. \end{cases} \quad (3.46)$$

Розв'язавши систему рівнянь (3.46) відносно b і a , отримаємо такі їх значення, які мінімізують суму квадратів відхилень експериментально виміряних значень величин x_i , y_i від теоретично заданих згідно з вибраною функціональною залежністю.

Рівняння, що входять у систему (3.46), називають *нормальними рівняннями Гауса*. Коефіцієнтами у них є суми, які «згладжують» дію похибок вимірювань величин x , y і зменшують їх вплив на оцінки параметрів b , a . Завдяки цьому підвищується точність їх визначення.

А тепер припустимо, що поле точок (x_i, y_i) експериментально визначених величин x , y має такий вигляд, як це показано на рис. 3.5.

Із цього рисунка видно, що середньою лінією цього поля, яка віддзеркалює «в середньому» функціональну залежність y від x , є парабола

$$y = ax^2 + bx + c, \quad (3.47)$$

параметри якої a , b , c також доцільно визначати за допомогою МНК.

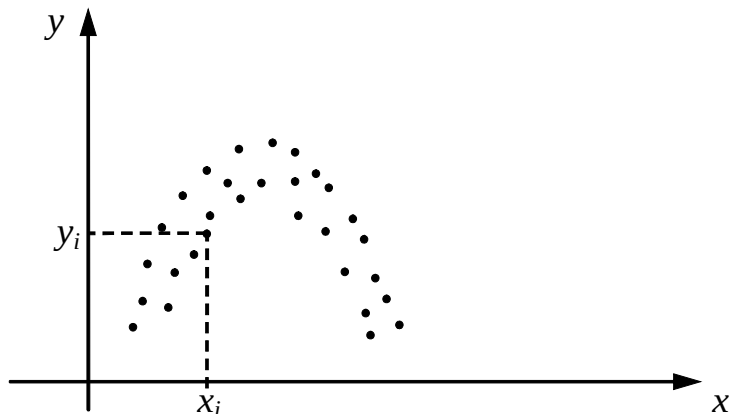


Рисунок 3.5 – Поле точок (x_i, y_i) експериментально визначених величин x , y

Для отримання нормальних рівнянь Гауса у цьому випадку підставимо (3.47) у (3.43), що дасть вираз

$$\Sigma^N = \sum_{i=1}^N \left(ax_i^2 + bx_i + c - y_i \right)^2. \quad (3.48)$$

Далі від цього виразу візьмемо частинні похідні за c , b , a та прирівняємо їх нулю, що дасть систему рівнянь:

$$\begin{cases} \frac{\partial \Sigma^N}{\partial c} = \sum_{i=1}^N 2 \left(ax_i^2 + bx_i + c - y_i \right) \cdot (1) = 0, \\ \frac{\partial \Sigma^N}{\partial b} = \sum_{i=1}^N 2 \left(ax_i^2 + bx_i + c - y_i \right) \cdot (x_i) = 0, \\ \frac{\partial \Sigma^N}{\partial a} = \sum_{i=1}^N 2 \left(ax_i^2 + bx_i + c - y_i \right) \cdot (x_i^2) = 0. \end{cases} \quad (3.49)$$

Після спрощень в системі (3.49) матимемо систему нормальних рівнянь Гауса (3.50), розв'язавши яку отримаємо оптимальні за критерієм мінімуму відхилень експерименту від теорії значення параметрів c , b , a математичної моделі (3.47) функціональної залежності величини y від величини x , заданих експериментальним полем точок (x_i, y_i) , яке зображено на рис. 3.5:

$$\begin{cases} cN + b \sum_{i=1}^N x_i + a \sum_{i=1}^N x_i^2 = \sum_{i=1}^N y_i, \\ c \sum_{i=1}^N x_i + b \sum_{i=1}^N x_i^2 + a \sum_{i=1}^N x_i^3 = \sum_{i=1}^N y_i x_i, \\ c \sum_{i=1}^N x_i^2 + b \sum_{i=1}^N x_i^3 + a \sum_{i=1}^N x_i^4 = \sum_{i=1}^N y_i x_i^2. \end{cases} \quad (3.50)$$

Приклад № 10. Нехай в результаті експерименту отримана послідовність десяти парних значень величин x , y , яка наведена у табл. 3.1.

Таблиця 3.1 – Експериментальні дані та проміжні дані їх обробки

i	1	2	3	4	5	6	7	8	9	10
x_i	0,2	0,4	0,6	0,8	1,0	1,2	1,4	1,6	1,8	2,0
y_i	1,16	1,30	1,20	1,20	0,9	0,9	0,4	0	-0,5	-1,0
x_i^2	0,04	0,16	0,36	0,64	1,0	1,44	1,96	2,56	3,24	4,0

Якщо нанести на координатну площину x , y ці точки, то можна побачити, що це є «збурена» завадами парабола. Але спочатку спробуємо використати для побудови моделі цієї залежності рівняння прямої (3.41).

Підставляючи значення x_i , y_i , $i = \overline{1, 10}$ із табл. 3.1 у рівняння (3.46), отримуємо:

$$\begin{cases} 10b + 11a = 5,56, \\ 11b + 15,4a = 2,072. \end{cases} \quad (3.51)$$

Розв'язком системи рівнянь (3.51) є оцінки

$$b^* = 1,904; \quad a^* = -1,225, \quad (3.52)$$

які задають модель залежності у вигляді прямої лінії

$$y^* = -1,225x + 1,904. \quad (3.53)$$

А тепер використаємо для побудови моделі залежності, заданої табл. 3.1, рівняння параболи (3.47).

Підставляючи значення $x_i, y_i, i = \overline{1, 10}$ із табл. 3.1 у рівняння (3.50), отримуємо:

$$\begin{cases} 10c + 11b + 15,4a = 5,560, \\ 11c + 15,4b + 24,2a = 2,072, \\ 15,4c + 24,2b + 40,533a = -1,186. \end{cases} \quad (3.54)$$

Розв'язком системи рівнянь (3.54) є оцінки

$$c^{**} = 1,017; \quad b^{**} = 0,991; \quad a^{**} = -1,008, \quad (3.55)$$

які задають модель вказаної вище залежності у вигляді параболи

$$y^{**} = -1,008x^2 + 0,991x + 1,017. \quad (3.56)$$

Для того, щоб визначити яка із двох моделей (3.53), (3.56) краще віддзеркалює функціональну залежність, що задана табл. 3.1, знайдемо числове значення критеріальних співвідношень (3.44), (3.48) з використанням (3.52) та (3.55).

Із (3.44) та (3.52) для поля точок, заданих табл. 3.1, знаходимо, що

$$\Sigma_*^N = 0,899, \quad (3.57)$$

а із (3.48) та (3.55), що

$$\Sigma_{**}^N = 0,041. \quad (3.58)$$

Отже, оскільки

$$\Sigma_{**}^N \ll \Sigma_*^N, \quad (3.59)$$

то, згідно з основною ідеєю МНК, модель (3.56) є набагато кращою, ніж модель (3.53), у чому легко переконатись, якщо нанести поле точок із табл. 3.1 на координатну площину (x, y) (рис. 3.6).

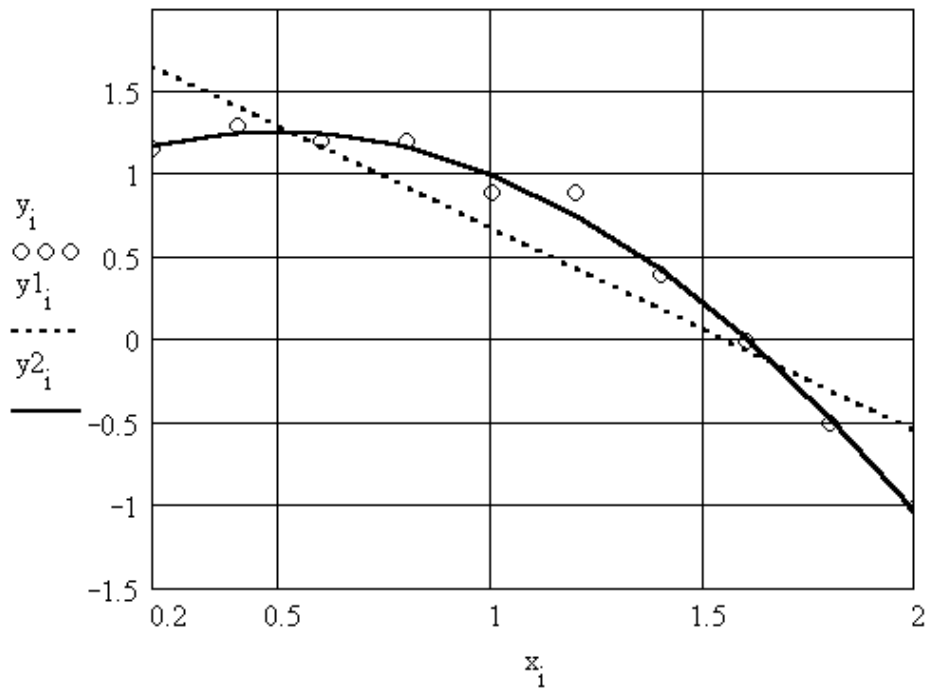


Рисунок 3.6 – Поля точок (x_i, y_i) , $(x_i, y1_i)$ і $(x_i, y2_i)$ експериментально визначених величин x , y та величин $y1$ і $y2$, обчислених за методом найменших квадратів і яким відповідають у тексті величини y^* та y^{**} (графік функції $y1(x)$ зображений у вигляді пунктирної лінії, а графік функції $y2(x)$ – у вигляді суцільної)

З усього викладеного у даному підрозділі можна зробити такі висновки:

- 1) за вибраної структури моделі функціональної залежності оптимальні значення її параметрів у межах заданого діапазону значень аргументу та його функції найбільш просто і ефективно визначаються за допомогою МНК;
- 2) вибір виду та структури моделі є прерогативою дослідника;
- 3) у класі заданих структур МНК дозволяє отримати не лише оптимальні значення коефіцієнтів, але й оптимальну структуру моделі.

Але заслуговує на увагу одне надзвичайно важливе зауваження, а точніше, попередження, яке полягає у тому, що, оскільки обчислювальні алгоритми МНК побудовані виключно на основі апроксимаційних процедур, то їх використання для розв'язання задач екстраполяції може призвести до неприпустимих помилок. Іншими словами, за допомогою МНК можна успішно розв'язувати задачі оптимізації відновлювання сигналів на вході вимірювальної системи за інформацією про її вихід та задачі оптимального синтезу динамічної характеристики (наприклад, передаточної функції) системи, за інформацією про її вхід та вихід, але ні в якому разі за допомогою МНК не можна синтезувати моделі прогнозу на значні відрізки часу.

Про те, яких значень можуть сягати помилки через ігнорування цього попередження, легко зробити висновок із рис. 3.7.

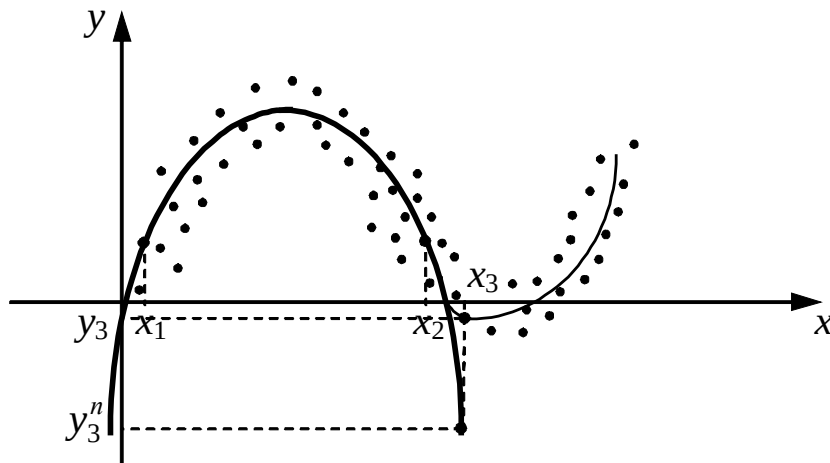


Рисунок 3.7 – Графічна ілюстрація недоцільності використання МНК для розв’язання задачі прогнозування на значні відрізки часу

З рис. 3.7 видно, що якщо маємо лише точки з діапазону значень координати X від x_1 до x_2 , то МНК як оптимальну модель залежності $y = f(x)$ дасть параболу. Під час використання цієї параболу для прогнозування значення y у точці x_3 отримаємо значення y_3^n , у той же час як реально координата y матиме значення y_3 , яке суттєво відрізняється від y_3^n .

На завершення цього підрозділу наведемо ще один приклад синтезу оптимальної моделі за МНК, але тепер уже з використання програм, складених мовою Python. Цей приклад вважатимемо **прикладом № 11**

Нехай задане поле точок $\{x_i, y_i\}, i = 0, 1, 2, 3, 4$, координати яких приведені у таблиці 3.2.

Таблиця 3.2.- Координати експериментально отриманих точок

I	0	1	2	3	4
x_i	0	1	2	3	4
y_i	0	0,5	1,5	1,0	0,5

Спочатку апроксимуємо за МНК це поле точок прямою – це буде приклад № 10 –

$$f_1(x) = a x + b. \quad (3.60)$$

Для цього нами буде використана наступна програма на мові Python:

```
In [1]: import numpy as np
In [2]: import scipy
In [3]: import scipy.linalg as la
In [4]: def f1(x):
        return a*x+b
In [5]: x=np.array([0,1,2,3,4])
In [6]: y=np.array([0.0,0.5,1.5,1.0,0.5])
In [7]: A=np.vstack([x,
np.ones(len(x))]).T
```



```

In [8]: a,b=la.lstsq(A,y)[0]
In [9]: a.round(3)
Out[9]: 0.15
In [10]: b.round(3)
Out[10]: 0.4
In [11]: import matplotlib
In [12]: import matplotlib.pyplot as plt
In [13]: plt.plot(x,y,'o',x,f1(x),'-r',\
                    markersize=8,linewidth=4)
In [14]: f1vec=np.vectorize(f1)
In [15]: f11=f1vec(x)
In [16]: f111=f11-y
In [17]: mf1y=la.norm(f111);mf1y
Out[17]:
1.036822067666386

```

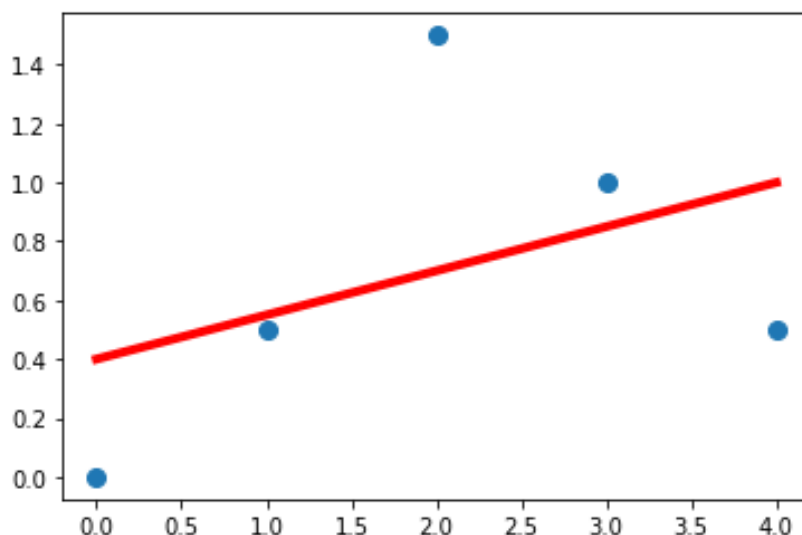


Рисунок 3.8 – Графік апроксимації поля точок прямою лінією

Підставляючи у вираз (3.60) значення коефіцієнтів, отримані командами In [9], In [10], отримаємо математичну модель у вигляді

$$f_1(x) = 0,15x + 0.4. \quad (3.61)$$

Ця модель апроксимує поле точок, заданих таблицею 3.2, з похибкою, визначеною командою In [17] приведеної вище програми.

А тепер апроксимуємо за МНК це ж саме поле точок, координати яких задані таблицею 3.2, трансцендентною функцією

$$f_2(x) = B \sin(bx) + C. \quad (3.62)$$

Для цього нами буде використана наступна програма на мові Python:

```

In [1]: import numpy as np
In [2]: import matplotlib
In [3]: import matplotlib.pyplot \
as plt
In [4]: import scipy
In [5]: import scipy.linalg as la
In [6]: from scipy.optimize import \
curve_fit
In [7]: def model(x,B,b,C):
        return B*np.sin(b*x)+C
In [8]: x=np.array([0.0,1.0,2.0,3.0,4.0])
In [9]: y=np.array([0.0,0.5,1.5,1.0,0.5])
In [10]: initial_guess=[1.0,1.0,0.0]
In [11]: params=curve_fit(model,x,y,\
p0=initial_guess)
In [12]: B=params[0][0]
In [13]: B=B.round(3);B
Out[13]: 1.354
In [14]: b=params[0][1]
In [15]: b=b.round(3);b
Out[15]: 0.677
In [16]: C=params[0][2]
In [17]: C=C.round(3);C
Out[17]: -0.09
In [18]: y1=model(x,B,b,C);y1

```

```
Out[18]: array([-0.09, 0.75822343, 1.23230499, 1.12313263, 0.47886104])
In [19]: plt.plot(x,y,'o',x,y1,'-r',\
                 markersize=8,linewidth=4)
```

```
In [20]: y11=y-y1
In [21]: myy1=la.norm(y11);myy1
Out[21]: 0.4025524242613081
```

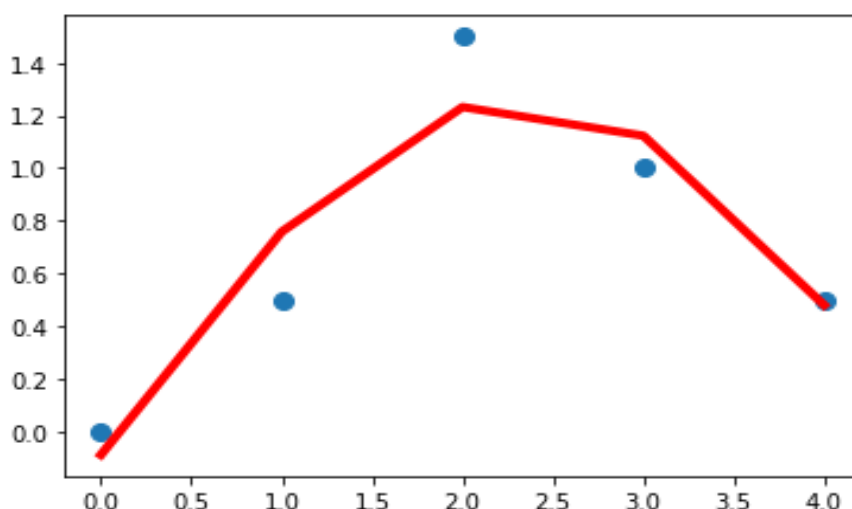


Рисунок 3.9 – Графік апроксимації поля точок синусоїдою

Підставляючи у вираз (3.62) значення коефіцієнтів, отримані командами In [12] - In [17], отримаємо математичну модель у вигляді

$$f_2(x) = 1,354 \sin(0,677x) - 0,09. \quad (3.63)$$

Ця модель апроксимує поле точок, заданих таблицею 3.2, з похибкою, визначеною командою In [21] приведеної вище програми.

Із графіків, приведених на рисунках 3.8 та 3.9, навіть візуально бачимо, що модель (3.63) краще апроксимує поле точок, заданих таблицею 3.2, у порівнянні з моделлю (3.61). А як свідчать результати обчислення похибок апроксимації, приведених в програмах, синусоїдальна модель здійснює апроксимацію заданого поля точок з похибкою, у два з половиною рази меншою похибки апроксимації лінійною моделлю.

3.5 Завдання для самоперевірки

Які умови повинні виконуватись для забезпечення можливості використання Фур'є-інтегрального методу ідентифікації?

Які задачі можна розв'язувати за допомогою ФІМІ?

У чому суть задачі відновлення сигналів?

Коли ФІМІ поза конкуренцією в задачах ідентифікації?

Як розкласти сигнал у ряд Фур'є?

Які етапи має прямий алгоритм відновлення сигналів?

Як визначити скільки гармонічних складових треба залишити в зрізаному ряді Фур'є відновлюваного сигналу?

Які критерії і яким чином використовуються під час синтезу алгоритму оптимального відновлення сигналів?

Як визначити номер гармонічної складової, з якої запускається алгоритм оптимального відновлення сигналів?

Який алгоритм – прямий чи оптимальний – є структурно більш простим і чому?

Яка процедура використовується у ФІМІ для визначення оптимальних значень параметрів моделі заданої структури?

Яка процедура використовується для визначення оптимальної структури моделі під час використання ФІМІ?

Чому алгоритми ідентифікації сигналів та системи за допомогою ФІМІ називають параметричними?

За яких умов задача ідентифікації динамічної системи за допомогою ФІМІ належить до класу коректних?

У чому полягає основна ідея методу найменших квадратів?

Як отримати систему нормальних рівнянь Гаусса для визначення оптимальних оцінок коефіцієнтів моделі заданої структури?

Як визначити оптимальну структуру моделі за допомогою МНК у вибраному класі структур?

Чому недоцільно використовувати МНК в задачах прогнозу?

Чим відрізняються програми апроксимації за МНК з використанням поліноміальних і трансцендентних функцій?

Частина 2

ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ СТОХАСТИЧНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ

У цій частині навчального посібника покажемо як побудувати математичні моделі як неперервних стохастичних лінійних динамічних систем з зосередженими параметрами, так і дискретних, як в часовій області, так і в частотній.

Як було зазначено у вступі до посібника, для того, щоб віднести динамічну систему до класу стохастичних, достатньо щоб рівень шумів у її структурі або в структурі сигналів, що надходять на її входи, був помітним у порівнянні з рівнем процесів у системі або рівнем регулярних складових сигналів на входах.

Розділ 4 СТОХАСТИЧНІ МОДЕЛІ НЕПЕРЕРВНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ НА ОСНОВІ СТАЦІОНАРНИХ ВИПАДКОВИХ ПРОЦЕСІВ

4.1 Основні характеристики стаціонарних випадкових процесів

Строга математична модель неперервного випадкового процесу припускає, що він протікає у часі від мінус нескінченності до плюс нескінченності, тобто $t \in (-\infty, \infty)$. А ту його частину $x^*(t)$, яку вдалося у якийсь спосіб зафіксувати, називають *реалізацією* випадкового процесу $x(t)$.

Для будь-якої реалізації $x^*(t)$ неперервного випадкового процесу $x(t)$ характерним є те, що вона містить у собі нескінченну кількість щільно розміщених поряд у часі значень $x(t)$ на будь-якому скінченному відрізку часу $[t_n, t_k]$, обмеженому моментами початку t_n та кінця t_k реєстрації, тобто для $x^*(t)$ справедливим є те, що $t \in [t_n, t_k]$.

Зрозуміло, що якщо випадковий процес є дискретним у часі $x(t_i)$, то його реалізація $x^*(t_i)$ є скінченною послідовністю випадкових чисел x_i , $i = \overline{1, N}$, зафіксованих на відрізку часу $[t_n, t_k]$, що можна віддзеркалити у такий спосіб:

$$x^*(t_i) = x_i; \quad i = \overline{1, N}; \quad t_i \in [t_n, t_k]. \quad (4.1)$$

У зв'язку з накладенням людьми на неперервний плин часу t циклічної системи його відліку (секунда, хвилина, доба, рік), що пов'язано із циклічністю обертання Землі навколо Сонця, зручно початок t кожної реалізації $x^*(t)$ відносити до початку відповідного циклу відліку часу. Це дає змогу, накладаючи реалізації на їх графіку одна на одну, вивчати їх у сукупності,

оскільки лише за такого підходу вдається встановити певні закономірності випадкових процесів і визначити цілий ряд цікавих їх числових і функціональних характеристик.

Як приклад на рис. 4.1 показані три добові реалізації споживання електричної потужності $P_{ел}$ містом, наближеним за розмірами та промисловими характеристиками до середнього для України.

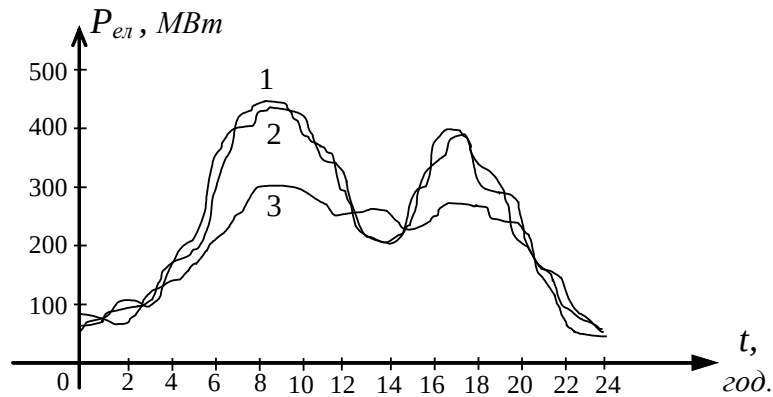


Рисунок 4.1 – Графік трьох добових реалізацій процесу споживання електричної потужності містом, середніх для України розмірів

Перша та друга реалізації характеризують добове споживання електричної потужності у середині тижня, а третя – у вихідний день, коли більшість підприємств не працюють, а побутові витрати стають більшими навіть у години обідньої перерви.

Для подальших викладок потрібні будуть деякі елементарні положення із теорії ймовірностей – нагадаємо їх.

Однією з основних характеристик випадкової величини X є її функція розподілу $F(x)$, яка задає ймовірність $P(X \leq x)$ отримання випадковою величиною X конкретного значення, не більшого від значення x , тобто

$$F(x) = P(X \leq x). \quad (4.2)$$

Графік функції розподілу $F(x)$ неперервної випадкової величини X має вигляд, наведений на рис. 4.2.

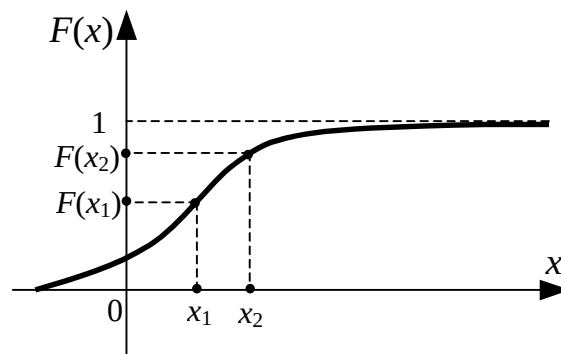


Рисунок 4.2 – Графік функції $F(x)$ розподілу неперервної випадкової величини X

Нагадаємо властивості функції $F(x)$ розподілу неперервної випадкової величини X :

- 1) якщо $x_2 \geq x_1$, то $F(x_2) \geq F(x_1)$;
- 2) якщо $x = -\infty$, то $F(-\infty) = 0$;
- 3) якщо $x = +\infty$, то $F(+\infty) = 1$;
- 4) якщо $x_1 \leq x < x_2$, то:

$$P(x_1 \leq x < x_2) = F(x_2) - F(x_1). \quad (4.3)$$

Перша із цих властивостей означає, що графік функції розподілу неперервної випадкової величини X є неперервною зростаючою зі зростанням аргументу x кривою.

Друга властивість означає, що випадкова величина X не може мати значень, менших від мінус нескінченності.

Третя властивість означає, що випадкова величина X не може мати значень, більших від плюс нескінченності.

А четверта властивість означає, що для знаходження ймовірності потрапляння випадкової величини X у проміжок значень $[x_1, x_2)$ достатньо взяти різницю значень її функції розподілу $F(x)$ на границях цього проміжку.

Всі ці властивості є справедливими і для дискретної випадкової величини, але слід пам'ятати, що графік її функції розподілу має східчасту форму (рис. 4.3).

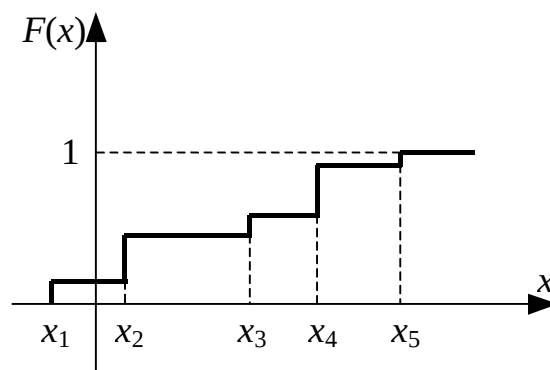


Рисунок 4.3 – Графік функції $F(x)$ розподілу дискретної випадкової величини X за умови, що величина X здатна набувати лише одне із п'яти значень x_1, x_2, x_3, x_4, x_5

Похідна $\frac{dF}{dx}$ від функції розподілу $F(x)$ неперервної випадкової величини X задає густину $f(x)$ ймовірностей значень цієї величини, тобто

$$f(x) = \frac{dF}{dx}. \quad (4.4)$$

Нагадаємо властивості густини $f(x)$ ймовірностей неперервної випадкової величини X :

- 1) за будь-яких значень x : $f(x) \geq 0$;
- 2) якщо $x = -\infty$, то $f(-\infty) = 0$;
- 3) якщо $x = +\infty$, то $f(+\infty) = 0$;
- 4) якщо $x_1 \leq X < x_2$, то

$$P(x_1 \leq X < x_2) = \int_{x_1}^{x_2} f(x) dx = F(x_2) - F(x_1), \quad (4.5)$$

причому:

$$\int_{-\infty}^{\infty} f(x) dx = 1. \quad (4.6)$$

Інтерпретація перших трьох властивостей густини ймовірностей $f(x)$ неперервної випадкової величини X така ж, як і та, що була дана для перших трьох властивостей її функції розподілу $F(x)$, а четверта властивість вказує на те, що ймовірність потрапляння випадкової величини X у проміжок значень $[x_1, x_2)$ чисельно дорівнює площі фігури під кривою $f(x)$, відсіченої ординатами при $x = x_1$ та $x = x_2$. П'ята властивість є наслідком попередніх відносно $f(x)$ та $F(x)$.

Здійснюючи графічне диференціювання кривої $F(x)$, зображеної на рис. 4.2, неважко переконатись у тому, що графік густини ймовірностей $f(x)$ має вигляд, наведений на рис. 4.4. В інших випадках графік функції $f(x)$ може бути дещо іншим – він може бути зсунутим вліво чи вправо, але незмінними є дві обставини. По-перше, для всіх значень x повинна виконуватись нерівність $f(x) \geq 0$, а по-друге, площа під кривою $f(x)$ завжди повинна дорівнювати одиниці.

Зазначимо, що для дискретної випадкової величини поняття густини ймовірностей не існує, оскільки похідна від її функції розподілу має розриви 2-го роду на початку кожної сходинки, що робить її принципово невизначеною.

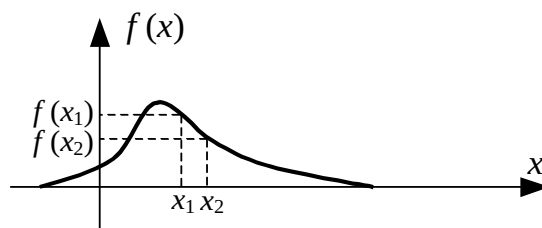


Рисунок 4.4 – Графік густини ймовірностей $f(x)$ неперервної випадкової величини X , отриманий графічним диференціюванням кривої $F(x)$, зображеної на рис. 4.2

Однак для аналізу дискретних випадкових процесів поняття густини і не потрібне.

Дуже важливими характеристиками як неперервної, так і дискретної випадкової величини X є її математичне очікування m_x та дисперсія D_x або позитивне значення кореня квадратного з неї, яке називають *середньоквадратичним відхиленням* σ_x . Перша з цих характеристик – m_x – задає теоретичне середнє значення випадкової величини X , а друга – D_x чи σ_x – характеризує розкид значень випадкової величини X відносно її середнього значення m_x .

Для неперервної випадкової величини X :

$$m_x = \int_{-\infty}^{\infty} x f(x) dx, \quad (4.7)$$

$$D_x = \int_{-\infty}^{\infty} (x - m_x)^2 f(x) dx, \quad (4.8)$$

а для дискретної:

$$m_x = \sum_{i=1}^N x_i p_i, \quad (4.9)$$

$$D_x = \sum_{i=1}^N (x_i - m_x)^2 p_i, \quad (4.10)$$

де p_i – імовірність того, що дискретна випадкова величина X набуде значення x_i :

$$p_i = P(X = x_i). \quad (4.11)$$

Звертаємо увагу на те, що у термінах теоретичної механіки формула (4.7) задає перший початковий момент, а формула (4.8) – другий центральний момент густини розподілу $f(x)$ неперервної випадкової величини X . А тому математичне очікування m_x випадкової величини X , незалежно від того, є вона неперервною чи дискретною, часто називають *першим початковим моментом* цієї випадкової величини, не пов'язуючи це з тим, визначена чи ні для неї густина ймовірностей. За аналогією D_x називають *другим центральним моментом* випадкової величини X .

Узагальнюючи, можна зазначити, що j -ий початковий момент $m_j(x)$ випадкової величини X можна визначити як математичне очікування j -го ступеня цієї величини, тобто

$$m_j(x) = \int_{-\infty}^{\infty} x^j f(x) dx \quad (4.12)$$

для неперервного випадку, або

$$m_j(x) = \sum_{i=1}^N x^j p_i \quad (4.13)$$

для дискретного.

За аналогією j -ий центральний момент випадкової величини X можна визначити як математичне очікування j -го ступеня відхилення цієї величини від її середнього значення, тобто

$$M_j(x) = \int_{-\infty}^{\infty} (x - m_x)^j f(x) dx \quad (4.14)$$

для неперервного випадку, або

$$M_j(x) = \sum_{i=1}^N (x - m_x)^j p_i \quad (4.15)$$

для дискретного.

Очевидно, що

$$\begin{cases} m_1(x) = m_x, \\ M_2(x) = D_x. \end{cases} \quad (4.16)$$

Очевидно й те, що

$$M_1(x) = 0. \quad (4.17)$$

Для кого тотожність (4.17) не очевидна, доведіть її справедливність самостійно, скориставшись формулами (4.14), (4.7) та (4.6).

Нагадавши вищевикладені елементи теорії ймовірностей відносно випадкових величин X , повернемося знову до випадкових процесів $X(t)$, які є теж випадковими величинами, але прив'язаними до плину часу.

Якщо розглядати час як параметр, який можна «зупинити», зафіксувавши на конкретному значенні t^* , то будь-яку сукупність реалізацій $x^*(t)$ випадкового процесу $X(t)$ можна «порізати» прямими, паралельними осі X , встановивши перпендикуляри із кожного значення t^* .

Випадкову величину $X(t^*)$, яка формується в кожному «перерізі» t^* сукупністю реалізацій $x^*(t)$ випадкового процесу $X(t)$, можна характеризувати густиною ймовірностей $f(x, t^*)$, а для всього процесу можна, «відпустивши» t^* , сформуувати густину імовірностей $f(x, t)$, в якій, нагадаємо, t є неперервним параметром із детерміновано визначеними нами, чи кимось іншим, значеннями.

Очевидно, що формули (4.7), (4.8) для неперервного випадкового процесу $X(t)$ перетворюються на:

$$m_x(t) = \int_{-\infty}^{\infty} x(t) f(x, t) dx, \quad (4.18)$$

$$D_x(t) = \int_{-\infty}^{\infty} (x(t) - m_x(t))^2 f(x, t) dx, \quad (4.19)$$

а математичне очікування $m_x(t)$ та дисперсія $D_x(t)$ у загальному випадку стають функціями часу t , як це видно на рис. 4.1, з якого легко бачити, що середні значення електричної потужності P_{el} , яка споживається містом у 2 години ночі і у 9 годин ранку, суттєво відрізняються; як суттєво відрізняється і розкид відхилення цієї потужності у ці ж години від її середнього значення.

Тепер уявимо собі, що дві реалізації $x^*(t)$ деякого випадкового процесу $X(t)$ мають вигляд, наведений на рис. 4.5.

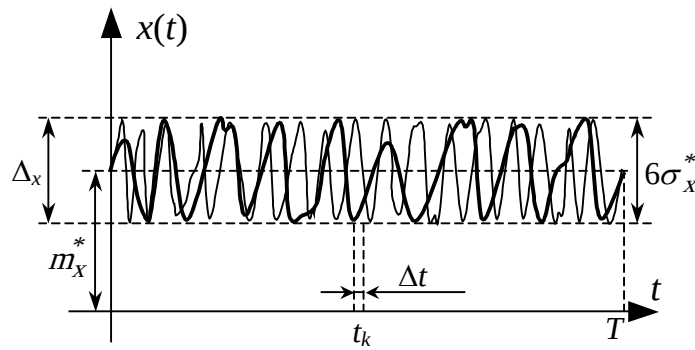


Рисунок 4.5 – Графік двох реалізацій деякого випадкового процесу $X(t)$

Навіть «неозброєним» оком видно, що і середнє значення цього процесу, і його дисперсія є не функціями часу, а константами, тобто

$$\begin{cases} m_x(t) = \text{const}_1 = m_x, \\ D_x(t) = \text{const}_2 = D_x. \end{cases} \quad (4.20)$$

Однак, згідно з формулами (4.18), (4.19), такий результат можна отримати лише в одному випадку, коли

$$f(x, t) = f(x). \quad (4.21)$$

А це, у свою чергу, справедливо лише для неперервного випадкового процесу $X(t)$, густина ймовірностей у кожному «перерізі» t сукупності реалізацій $x^*(t)$ якого описується одним і тим же законом $f(x)$, тобто не залежить від t .

Випадкові процеси, для яких виконується умова (4.21) і, як наслідок, умова (4.20), відносять до класу стаціонарних випадкових процесів.

Відзначимо, що математики розрізняють стаціонарність у широкому та у вузькому розумінні, пов'язуючи це з багатовимірними функціями розподілу

випадкових процесів, що їх відносять до моделей красивих, але практично не придатних для розв'язання конкретних задач. В інженерних задачах стаціонарність випадкових процесів і у широкому, і у вузькому розумінні збігається, а тому не будемо ускладнювати матеріал введенням багатовимірних функцій розподілу цих процесів.

У класі стаціонарних випадкових процесів $X(t)$ виділяють підклас ергодичних, для яких усереднення на множині значень x дає той же результат, що й усереднення в часі t .

Це означає, що математичне очікування m_x та дисперсію D_x ергодичного випадкового процесу $X(t)$ можна отримати не лише за допомогою формул (4.18), (4.19), але і за допомогою формул, набагато простіших за структурою обчислень:

$$m_x = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t) dt, \quad (4.22)$$

$$D_x = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} (x(t) - m_x)^2 dt. \quad (4.23)$$

Одразу ж зазначимо, що всі стаціонарні випадкові процеси, з якими під час розв'язання практичних професійних задач мають справу інженери, є ергодичними, а тому формули (4.22), (4.23) для обчислення очікуваного середнього значення процесу m_x та характеристики його відхилень від середнього значення - очікуваної дисперсії D_x - є основними у теоретичному плані.

Але у практиці розрахунків використовуються дещо інші формули:

$$m_x^* = \frac{1}{T} \int_0^T x^*(t) dt, \quad (4.24)$$

$$D_x^* = \frac{1}{T} \int_0^T (x^*(t) - m_x)^2 dt, \quad (4.25)$$

за якими знаходяться статистична оцінка m_x^* математичного очікування m_x ергодичного випадкового процесу $X(t)$ та статистична оцінка D_x^* дисперсії D_x цього процесу з використанням однієї, але достатньо інформативної реалізації $x^*(t)$ випадкового процесу $X(t)$, зафіксованої на відрізку часу T .

Очевидно, що оцінки m_x^* та D_x^* теж є випадковими величинами, залежними від довжини T реалізації $x^*(t)$, але, зрозуміло, що дисперсія цих оцінок є набагато меншою у порівнянні із дисперсією процесу $X(t)$ і з ростом T наближається до нуля.

Нагадаємо, що до статистичних оцінок числових характеристик випадкових величин математична статистика висуває три вимоги: по-перше, вони повинні бути незміщеними, по-друге, ефективними, а по-третє, переконливими.

Вимога незміщеності означає, що з ростом T рано чи пізно значення оцінки точно збіжиться зі значенням відповідної числової характеристики.

Вимога ефективності означає, що із множини різноманітних формул, які можуть бути запропоновані для отримання числового значення оцінки, необхідно вибрати ту, за допомогою якої оцінка визначається з найменшою власною дисперсією.

Вимога переконливості означає, що оцінка з ростом T неперервно наближається до числової характеристики, що дає право зупиняти обчислення варіантів оцінки, як тільки вони перестають відрізнятися один від одного у заданих межах.

Більш детально ознайомитись із вимогами до статистичних оцінок числових характеристик випадкових величин можна у будь-якому з посібників із теорії ймовірностей та математичної статистики. А у нашому посібнику згадаємо ще лише про те, що мало знайти числове значення статистичної оцінки m_x^* чи D_x^* . Треба обов'язково це значення доповнити визначенням довірчого інтервалу, в який із деякою наперед заданою довірчою імовірністю потрапляє отримане значення статистичної оцінки. Оскільки, ще раз підкреслимо, статистичні оцінки є теж випадковими величинами, то дуже важливо знати, за які межі вони не вийдуть, якщо нарощувати масиви інформації, що використовуються у розрахунках.

За тим, як розраховуються довірчі інтервали для оцінок числових характеристик випадкових величин, теж відсилаємо до навчальних посібників з математичної статистики.

Декілька корисних порад.

Порада перша. Якщо, зафіксувавши реалізацію неперервного випадкового процесу $x(t)$, ви бачите (див. рис. 4.5), що вона не виходить за межі деякої смуги, границі якої проходять паралельно осі часу, і досить часто наближається до обох границь, то можете не сумніватись у тому, що процес $X(t)$ є не лише стаціонарним, але й ергодичним.

Порада друга. Для грубої оцінки середнього значення m_x^* ергодичного процесу $X(t)$ за зафіксованою на відрізку часу T його реалізацією $x(t)$ досить «на око» провести паралельно осі t вісь «квазісиметрії» цієї реалізації (див. рис. 4.5), відстань від якої до осі t і дасть значення оцінки m_x^* з похибкою не більше 5%.

Порада третя. Для грубої оцінки середньоквадратичного відхилення $\sigma_x^* = \sqrt{D_x^*}$ процесу $X(t)$ від середнього значення m_x необхідно ширину смуги

Δ_x , за межі якої не виходить реалізація $x(t)$ за умови, що лінія m_x^* проходить центром смуги, розділити на шість, тобто

$$\sigma_x^* = \frac{\Delta_x}{6}. \quad (4.26)$$

Формула (4.26) є наслідком відомого у математичній статистиці «правила трьох сигм», згідно з яким всі значення нормально розподіленої величини X з імовірністю 0,997 знаходяться у смузі значень $(m_x - 3\sigma_x, m_x + 3\sigma_x)$.

Нагадаємо, що *нормально розподіленою* (гауссівською) називається така випадкова величина, густина ймовірностей якої задається законом:

$$f(x) = \frac{1}{\sigma_x \sqrt{2\pi}} \cdot e^{-\frac{1}{2} \left(\frac{x - m_x}{\sigma_x} \right)^2}. \quad (4.27)$$

Порада четверта. Якщо для подальших розрахунків потрібне якомога точніше значення оцінки m_x^* , то, вибравши по можливості якомога менший інтервал квантування часу Δ_t (див. рис. 4.5), сформууйте з реалізації $x(t)$ решітчасту функцію $x(k \cdot \Delta_t) = x_k$, середнє значення якої знаходиться за допомогою виразу, що є дискретним аналогом (4.24), а саме:

$$m_x^* = \frac{1}{N} \sum_{k=1}^N x_k, \quad (4.28)$$

де $N = \frac{T}{\Delta_t}$, або

$$m_x^* = \frac{1}{N} \sum_{k=0}^{N-1} x_k. \quad (4.29)$$

Оцінка m_x^* за допомогою формули (4.28) чи (4.29) є незміщеною, ефективною і переконливою.

Порада п'ята. Під час використання решітчастого аналога x_k ергодичного випадкового процесу $X(t)$ для обчислення оцінки дисперсії за дискретним аналогом формула (4.25) набуває вигляду:

$$D_x^* = \frac{1}{N} \sum_{k=1}^N (x_k - m_x)^2. \quad (4.30)$$

Оскільки в формулі (4.30) доводиться використовувати не математичне очікування m_x випадкового процесу $X(t)$, а його статистичну оцінку m_x^* , знайдену за формулою (4.27) чи (4.28), то оцінка дисперсії буде незміщеною, ефективною і переконливою лише тоді, коли вона знаходиться за формулою:

$$D_x^* = \frac{1}{N-1} \sum_{k=1}^N (x_k - m_x^*)^2. \quad (4.31)$$

Звичайно, якщо число N є значним, то формули (4.30) та (4.31), у разі використання в обох m_x^* , будуть давати практично один і той же результат, але вже при $N = 20$ результати різнитимуться на 5%, а з подальшим зменшенням N ця різниця зростатиме.

4.2 Кореляційні функції та спектральні густини стаціонарних випадкових процесів

Нехай маємо сукупність добових реалізацій такого випадкового процесу, як зміна температури зони обмотки статора електричного двигуна конвеєра, яка є функцією струму, що протікає у цій обмотці. Випадковість значень цього процесу залежить від рівномірності подачі та гранулометричного складу того сипучого чи кускового матеріалу, який транспортується цим конвеєром.

Якщо струм зростає, то і температура зони теж зростає. Якщо струм спадає, то з деяким запізненням спадає і температура зони. Очевидно, що від значення температури у зоні в даний момент часу суттєво залежить те її значення, якого вона набуде через одну хвилину. Через 10 хвилин ця залежність уже буде менш помітною. Ще меншим температурний вплив буде через 100 хвилин. А через 1000 хвилин навряд чи є сенс його взагалі оцінювати.

Оскільки, як було зазначено, температурний процес, що розглядається, є випадковим, то безпосередньо оцінити вплив одного його значення на інше неможливо.

Для оцінювання лінійного зв'язку між двома значеннями x_1 , x_2 випадкового процесу $X(t)$ у момент часу t_1 та t_2 математики ввели таку характеристику, як кореляційна функція $K_x(t_1, t_2)$, яку визначають як

$$K_x(t_1, t_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_1 - m_x(t_1))(x_2 - m_x(t_2)) \cdot f(x_1, x_2, t_1, t_2) dx_1 dx_2, \quad (4.32)$$

де $f(x_1, x_2, t_1, t_2)$ – двовимірна густина ймовірностей випадкового процесу $X(t)$, котра визначається у моменти часу t_1 , t_2 , відносно яких випадковий процес $X(t)$ можна розглядати як систему двох випадкових величин X_1 та X_2 , значеннями x_1 та x_2 яких є значення $x(t_1)$, $x(t_2)$ реалізацій випадкового процесу, зафіксовані у моменти часу t_1 , t_2 :

$$f(x_1, x_2, t_1, t_2) = \frac{\partial^2 F(x_1, x_2, t_1, t_2)}{\partial x_1 \partial x_2}. \quad (4.33)$$

$F(x_1, x_2, t_1, t_2)$ – двовимірна функція розподілу ймовірностей випадкового процесу $X(t)$, яка задає значення ймовірності того, що у момент t_1 виконується нерівність $X_1 \leq x_1$, а у момент t_2 виконується нерівність $X_2 \leq x_2$, тобто

$$F(x_1, x_2, t_1, t_2) = P(X(t_1) \leq x_1, X(t_2) \leq x_2). \quad (4.34)$$

Слід зазначити, що формула (4.32) грає не стільки практичну, скільки загальнотеоретичну роль, оскільки на практиці ніхто не проводить оцінювання функцій $f(x_1, x_2, t_1, t_2)$ та $F(x_1, x_2, t_1, t_2)$ за експериментальними даними.

Формула (4.32) навіть у теоретичному плані спрощується для стаціонарного випадкового процесу $X(t)$, для якого двовимірна густина $f(\bullet)$ та функція розподілу $F(\bullet)$ ймовірностей залежать не від конкретних значень t_1, t_2 моментів часу t , а лише від їх різниці $\tau = t_1 - t_2$, тобто

$$\begin{cases} f(x_1, x_2, t_1, t_2) = f(x_1, x_2, \tau), \\ F(x_1, x_2, t_1, t_2) = F(x_1, x_2, \tau). \end{cases} \quad (4.35)$$

З урахуванням виразів (4.35), формула для кореляційної функції стаціонарного випадкового процесу $X(t)$ набуває вигляду:

$$K_x(\tau) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_1 - m_x)(x_2 - m_x) \cdot f(x_1, x_2, \tau) dx_1 dx_2. \quad (4.36)$$

Але і формулу (4.36) у практичних розрахунках теж не використовують, оскільки оцінку двовимірної густини ймовірностей $f(x_1, x_2, \tau)$ за експериментальними даними визначати непросто. Набагато простішим є визначення кореляційної функції $K_x(\tau)$ стаціонарного випадкового процесу $X(t)$ з використанням властивості ергодичності, згідно з якою оцінка кореляційної функції $K_x^*(\tau)$ може бути визначена як

$$K_x^*(\tau) = \frac{1}{T - \tau} \int_0^{T-\tau} (x^*(t) - m_x) \cdot (x^*(t + \tau) - m_x) dt, \quad (4.37)$$

де T – відрізок часу, протягом якого фіксується реалізація $x^*(t)$ випадкового процесу $X(t)$.

Математиками доведено, що для ергодичного випадкового процесу

$$K_x(\tau) = \lim_{T \rightarrow \infty} K_x^*(\tau), \quad (4.38)$$

що дає право за оцінку $K_x^*(\tau)$ кореляційної функції $K_x(\tau)$ брати будь-яку функцію, отриману за допомогою формули (4.37).

Очевидно, що чим більшим є значення T , тим точніше оцінка $K_x^*(\tau)$ відображає кореляційну функцію $K_x(\tau)$.

Аналізуючи формулу (4.37), бачимо, що для отримання значення оцінки $K_x^*(\tau)$ кореляційної функції $K_x(\tau)$ ергодичного випадкового процесу $X(t)$ для значення τ_1 параметра τ при реалізації $x^*(t)$ цього процесу (рис. 4.6, а) необхідно, зсунувши цю реалізацію на відрізок часу $t = \tau_1$ (рис. 4.6, б), перемножити одна на одну на відрізку $[0, T - \tau_1]$ основну та «зсунуту» реалізації $x^*(t)$, знайти площу під кривою, яка є результатом цього перемноження, і поділити отриману площу на довжину відрізка інтегрування $T - \tau_1$.

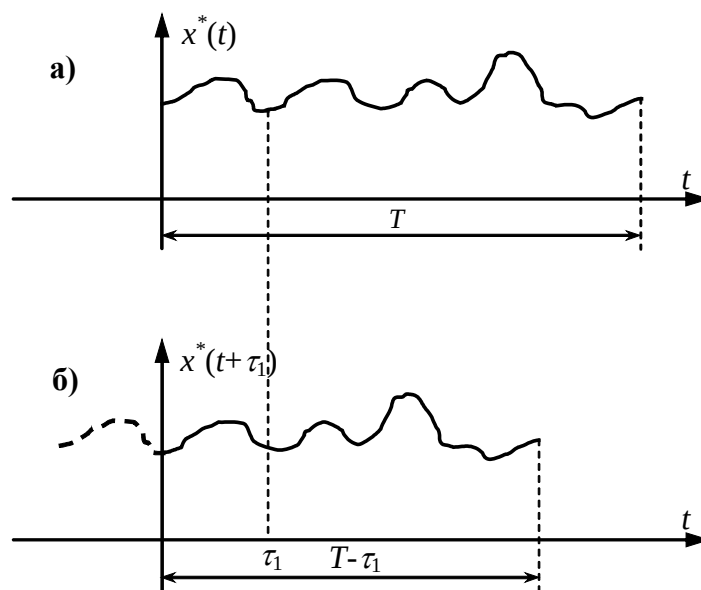


Рисунок 4.6 – Геометрична інтерпретація формування підінтегрального виразу у формулі (4.37)

Зрозуміло, що з ростом τ точність визначення значень $K_x^*(\tau)$ зменшується, оскільки зменшується інтервал «перекриття» основної та «зсунутої» на відрізок τ реалізації $x^*(t)$, що у разі використання чисельних методів приводить до зменшення кількості значень випадкового процесу $X(t)$, які беруть участь у формуванні підінтегрального виразу у формулі (4.37).

Оскільки результати інтегрування у формулі (4.37) не залежать від того, вправо чи вліво на τ «зсувається» реалізація $x^*(t)$, то справедливим є вираз

$$K_x(\tau) = K_x(-\tau), \quad (4.39)$$

який констатує той факт, що кореляційна функція випадкового процесу є симетричною відносно осі ординат.

Підставивши у формулу (4.37) $\tau = 0$, отримаємо:

$$K_x^*(0) = \frac{1}{T} \int_0^T (x(t) - m_x)^2 dt. \quad (4.40)$$

Порівнюючи вираз (4.40) з виразом (4.25), бачимо, що вони дають один і той же результат, тобто

$$K_x^*(0) = D_x^*. \quad (4.41)$$

Отже значення $K_x(0)$ кореляційної функції $K_x(\tau)$ центрованого ергодичного випадкового процесу $X(t)$ за нульового зсуву $\tau=0$ дорівнює дисперсії D_x .

Нагадаємо, що *центрованим* називається випадковий процес $X(t)$, від усіх значень $x(t)$ якого віднімається його середнє значення $m_x(t)$ (рис. 4.7).

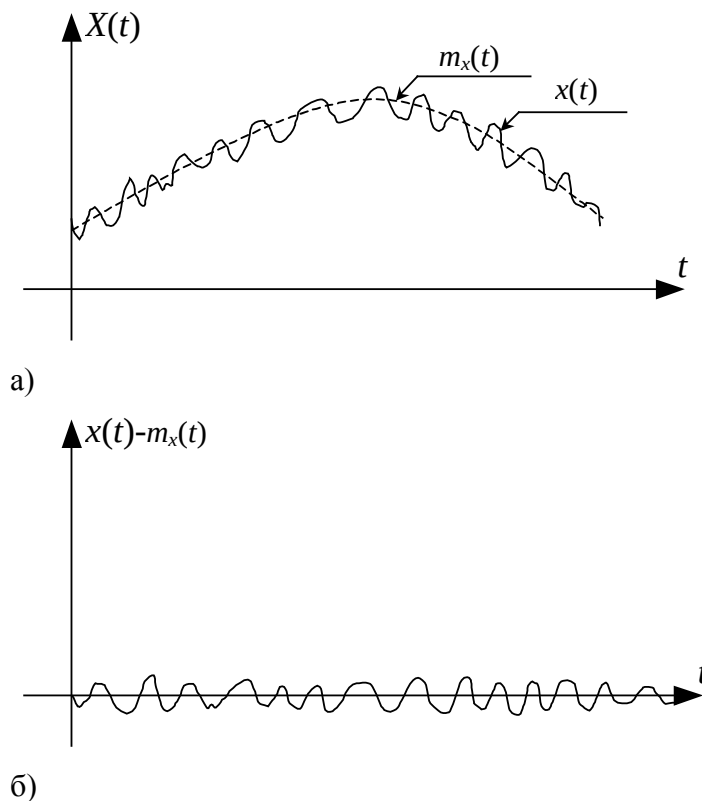


Рисунок 4.7 – Графічна інтерпретація центрування випадкового процесу

З рис. 4.7 видно, що навіть нестационарний випадковий процес (рис. 4.7, а) у разі його центрування може стати не лише стаціонарним (рис. 4.7, б), але й ергодичним.

Зазначимо, що, як правило, під час розв'язання практичних інженерних задач, оцінки кореляційних функцій визначаються саме для центрованих випадкових процесів, для яких у такому випадку майже завжди присутня ергодичність.

При виконанні практичного аналізу центрованих ергодичних випадкових процесів, які мають місце при розв'язанні інженерних задач, найбільше

розповсюдження отримали чотири моделі кореляційних функцій цих процесів, а саме:

$$K_x(\tau) = \sigma_x^2 e^{-\alpha|\tau|}, \quad (4.42)$$

$$K_x(\tau) = \sigma_x^2 e^{-\alpha|\tau|} (1 + \alpha|\tau|), \quad (4.43)$$

$$K_x(\tau) = \sigma_x^2 e^{-\alpha|\tau|} \cdot \cos \beta\tau, \quad (4.44)$$

$$K_x(\tau) = \sigma_x^2 e^{-\alpha|\tau|} \cdot \left(\cos \beta\tau + \frac{\alpha}{\beta} \sin \beta\tau \right). \quad (4.45)$$

Приклади графіків кореляційних функцій, заданих моделями (4.42) та (4.44), показані на рис. 4.8.

Зазначимо, що моделі (4.43) та (4.45) можна інтерпретувати як «розтягувальні» по осі τ для моделей (4.42) та (4.44).

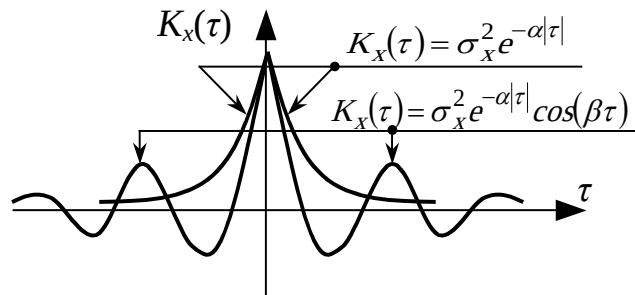


Рисунок 4.8 – Приклади графіків кореляційних функцій ергодичних випадкових процесів

А тепер розглянемо такий специфічний технологічний об'єкт, як кульовий млин, який використовується для тонкого помолу залізної руди перед подальшим збагаченням її у флотаційній машині.

Кульовим млином є циліндр, на третину заповнений металевими кулями, який має вхідний отвір для подачі подрібненої залізної руди та вихідний отвір для відбору тієї ж залізної руди, але перетертої на пил металевими кулями під час їх хаотичного переміщення разом із залізною рудою внаслідок обертання циліндра навколо повздовжньої осі за допомогою потужного електропривода.

Очевидно, що зміна в часі і гранулометричного складу $X(t)$ подрібненої залізної руди на вході млина, і струму якоря його приводного електродвигуна $Y(t)$ є випадковими процесами, які в усталеному режимі роботи млина можна вважати не лише стаціонарними, а й ергодичними.

Очевидним є і те, що лише середнє значення струму $Y(t)$ функціонально залежить від середнього значення гранулометричного складу $X(t)$, а кожне конкретне значення процесу $Y(t)$ функціонально зв'язати з конкретними значеннями процесу $X(t)$ не можна, оскільки на випадковість значень $Y(t)$

суттєво впливає не лише процес $X(t)$, але і внутрішній процес постійних змін самої структури кульового млина, викликаний як переміщенням металевих куль одна відносно одної та стінок циліндричного барабана, так і переміщенням шматочків руди між кулями та між собою.

Тож для характеристики усередненого лінійного зв'язку між значеннями випадкового процесу $X(t)$ у часовому перерізі t_1 та випадкового процесу $Y(t)$ у часовому перерізі t_2 (рис. 4.9) математики вводять таку характеристику, як взаємна кореляційна функція $K_{yx}(t_1, t_2)$, для якої справедливим є вираз

$$K_{yx}(t_1, t_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_1 - m_x(t_1))(y_2 - m_y(t_2)) \cdot f(x_1, y_2, t_1, t_2) dx_1 dy_2, \quad (4.46)$$

де

$$f(x_1, y_2, t_1, t_2) = \frac{\partial^2 F(x_1, y_2, t_1, t_2)}{\partial x_1 \partial y_2}, \quad (4.47)$$

$$F(x_1, y_2, t_1, t_2) = P(X(t_1) \leq x_1, Y(t_2) \leq y_2) \quad (4.48)$$

– відповідно, двовимірні взаємні густина $f(\bullet)$ та функція розподілу $F(\bullet)$ ймовірностей процесів $X(t)$ та $Y(t)$.

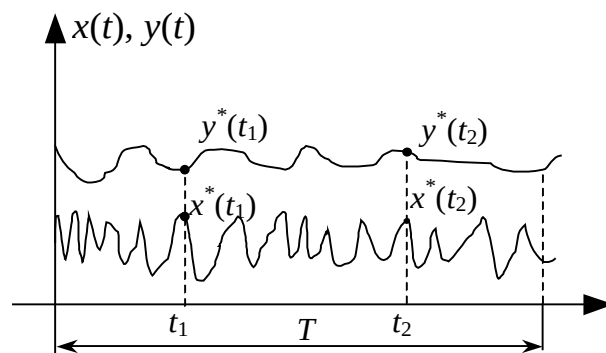


Рисунок 4.9 – Графіки реалізацій $x^*(t)$, $y^*(t)$ взаємопов'язаних випадкових процесів $X(t)$, $Y(t)$, зафіксованих на проміжку часу T

Для стаціонарних випадкових процесів $X(t)$ та $Y(t)$ за аналогією з виразами (4.34) – (4.36) маємо:

$$\begin{cases} f(x_1, y_2, t_1, t_2) = f(x_1, y_2, \tau), \\ F(x_1, y_2, t_1, t_2) = F(x_1, y_2, \tau), \end{cases} \quad (4.49)$$

$$K_{yx}(\tau) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_1 - m_x)(y_2 - m_y) \cdot f(x_1, y_2, \tau) dx_1 dy_2, \quad (4.50)$$

а для ергодичних випадкових процесів за аналогією з виразами (4.37), (4.38) маємо:

$$K_{yx}^*(\tau) = \frac{1}{T-\tau} \int_0^{T-\tau} (y^*(t) - m_y) \cdot (x^*(t+\tau) - m_x) dt, \quad (4.51)$$

$$K_{yx}(\tau) = \lim_{T \rightarrow \infty} K_{yx}^*(\tau). \quad (4.52)$$

Приклади графіків можливих взаємних кореляційних функцій центрованих ергодичних процесів $X(t)$, $Y(t)$ наведені на рис. 4.10.

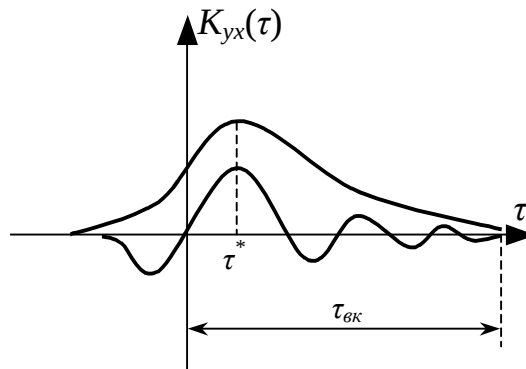


Рисунок 4.10 – Приклади графіків взаємних кореляційних функцій центрованих ергодичних процесів

Зсув максимуму взаємної кореляційної функції $K_{yx}(\tau)$ на відрізок τ^* вправо від осі ординат характеризує той відрізок часу, на який запізнюється поява на виході системи процесу $Y(t)$ відносно процесу $X(t)$, котрий його викликав, оскільки значення реакції системи найкраще корелюється саме з тим значенням вхідного процесу, яке його викликало. Тобто τ^* можна використовувати як одну з характеристик інерційних властивостей системи, що продукує процес $Y(t)$ у відповідь на вхідний для неї процес $X(t)$. Якщо застосувати до кореляційної функції $K_x(\tau)$ перетворення Фур'є, тобто взяти інтеграл

$$S_x(\omega) = \int_{-\infty}^{\infty} K_x(\tau) e^{-j\omega\tau} d\tau, \quad (4.53)$$

тоді отримаємо іншу характеристику стаціонарного випадкового процесу, яку називають його *спектральною густиною* $S_x(\omega)$, оскільки вона характеризує щільність спектра частот гармонічних складових з випадковими значеннями амплітуди і фази, сукупністю яких можна задати даний процес.

Для прикладу отримаємо спектральну густину стаціонарного випадкового процесу $X(t)$, який має кореляційну функцію (4.42).

Підставивши вираз (4.42) у (4.53), отримаємо

$$\begin{aligned}
S_x(\omega) &= \int_{-\infty}^{\infty} \sigma_x^2 e^{-\alpha|\tau|} e^{-j\omega\tau} d\tau = \int_{-\infty}^0 \sigma_x^2 e^{\alpha\tau} e^{-j\omega\tau} d\tau + \int_0^{\infty} \sigma_x^2 e^{-\alpha\tau} e^{-j\omega\tau} d\tau = \\
&= \sigma_x^2 \int_{-\infty}^0 e^{\tau(\alpha-j\omega)} d\tau + \sigma_x^2 \int_0^{\infty} e^{-\tau(\alpha+j\omega)} d\tau = \frac{\sigma_x^2}{\alpha-j\omega} e^{\tau(\alpha-j\omega)} \Big|_{-\infty}^0 + \\
&+ \frac{\sigma_x^2}{-(\alpha+j\omega)} e^{-\tau(\alpha+j\omega)} \Big|_0^{\infty} = \frac{\sigma_x^2}{\alpha-j\omega} + \frac{\sigma_x^2}{\alpha+j\omega} = \frac{2\sigma_x^2\alpha}{\alpha^2 + \omega^2}. \quad (4.54)
\end{aligned}$$

Аналізуючи вирази (4.42) та (4.54) разом, бачимо, що чим більшим є значення α , тим швидше «згасає» кореляційна функція $K_x(\tau)$ і тим повільніше «згасає» спектральна густина $S_x(\omega)$, тобто чим менше значень випадкового процесу $X(t)$ корелюються між собою, тим більше необхідно гармонічних складових для адекватного представлення цього процесу у частотній області. На рис. 4.11 показані два графіки кореляційних функцій $K_x(\tau)$ при $\sigma_x^2=1$ та $\alpha=1$ і $\alpha=2$ (рис. 4.11, а), а також два графіки спектральних густин $S_x(\omega)$ для цих же значень σ_x^2 і τ (рис. 4.11, б).

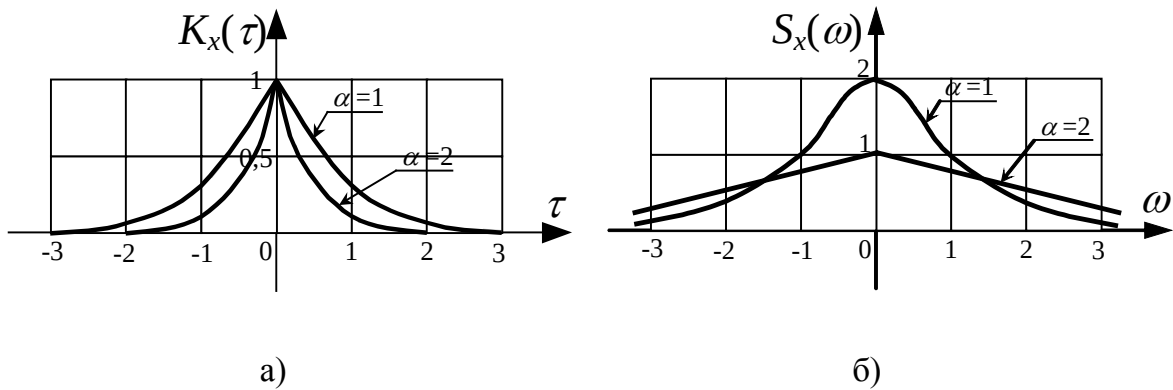


Рисунок 4.11 – Приклади графіків кореляційних функцій $K_x(\tau)$ стаціонарних випадкових процесів та їх спектральних густин $S_x(\omega)$

Очевидно, що, перетворюючи за Фур'є взаємну кореляційну функцію $K_{yx}(\tau)$ стаціонарних випадкових процесів $X(t)$ та $Y(t)$, отримаємо взаємну спектральну густина $S_{yx}(j\omega)$ цих процесів, тобто

$$S_{yx}(j\omega) = \int_{-\infty}^{\infty} K_{yx}(\tau) e^{-j\omega\tau} d\tau, \quad (4.55)$$

яка, на жаль, вже буде комплексною функцією, оскільки підінтегральна функція у функціоналі (4.55) $K_{yx}(\tau)$ не є симетричною відносно осі ординат.

Особливу роль під час аналізу випадкових процесів має стаціонарний процес $X(t)$ з нульовим середнім $m_x=0$, жодна із гармонічних складових

якого не корелюється ні з якою іншою складовою, крім самої себе. Кореляційну функцію такого процесу можна задати у вигляді

$$K_x(\tau) = \sigma_x^2 \delta(\tau), \quad (4.56)$$

де $\delta(t)$ – дельта-функція Дірака, яка, нагадаємо для зручності, має такі властивості:

$$\delta(t) = \begin{cases} \infty, & \text{якщо } t=0, \\ 0, & \text{якщо } t \neq 0, \end{cases} \quad (4.57)$$

$$\int_{-\infty}^{\infty} \delta(t) dt = 1. \quad (4.58)$$

Якщо підставити вираз (4.56) у (4.53), то отримаємо

$$S_x(\omega) = \int_{-\infty}^{\infty} \sigma_x^2 \delta(\tau) e^{-j\omega\tau} d\tau = \sigma_x^2 e^{-j\omega \cdot 0} \int_{-\infty}^{\infty} \delta(\tau) d\tau = \sigma_x^2. \quad (4.59)$$

Із виразу (4.59) випливає, що спектр цього процесу рівномірний від $-\infty$ до $+\infty$, тобто у сигналі присутні всі частоти з амплітудами випадковими, але такими, які забезпечують дисперсію процесу в усіх часових перерізах на рівні σ_x^2 .

Процес із визначеними у такий спосіб властивостями називають *білим шумом*.

Пропускаючи білий шум через фільтр із заданими параметрами, що формують глибину кореляції складових, можемо сформувати випадковий процес з будь-якими наперед заданими властивостями, тобто із білого шуму за допомогою частотного фільтра можемо виділяти будь-який «колір».

Як це зробити практично, буде показано у наступному розділі.

4.3 Ідентифікація неперервних стохастичних ЛДС ЗП за допомогою рівняння Вінера-Хопфа

Нехай в лінійній неперервній системі з невідомими, але зосередженими параметрами має місце стохастичний процес, обумовлений дією на її вході ергодичного стаціонарного випадкового сигналу $X(t)$ (рис. 4.12).



Рисунок 4.12 – Умовне представлення неперервної стохастичної ЛДС ЗП у вигляді «чорного ящика»

Зрозуміло, що вихідний сигнал $Y(t)$ системи теж буде ергодичним стаціонарним процесом.

Припустимо, що вдалося зафіксувати центровані реалізації $x^*(t)$ та $y^*(t)$ випадкових процесів $X(t)$, $Y(t)$ на одному і тому ж проміжку часу T (див. рис. 4.9). Незалежно від характеру сигналів $x(t)$, $y(t)$ для центрованих реалізацій $x^*(t)$, $y^*(t)$ є справедливою згортка

$$y^*(t) = \int_0^{\infty} x^*(t-\theta)g(\theta)d\theta, \quad (4.60)$$

де $g(t)$ – вагова (імпульсна перехідна) функція неперервної ЛДС ЗП, для якої $x(t)$ є вхідним, а $y(t)$ – вихідним сигналами.

Якщо рівняння (4.60) є справедливим для довільного моменту часу t , то воно залишається справедливим і для моменту часу $t + \tau$, тобто

$$y^*(t + \tau) = \int_0^{\infty} x^*(t + \tau - \theta)g(\theta)d\theta. \quad (4.61)$$

Помножимо обидві частини рівняння (4.61) на $x^*(t)$ і проінтегруємо їх у межах від нуля до $T - \tau$, розділивши результат інтегрування на $T - \tau$. Внаслідок цих дій отримаємо:

$$\begin{aligned} & \frac{1}{T - \tau} \int_0^{T - \tau} y^*(t + \tau) \cdot x^*(t) dt = \\ & = \frac{1}{T - \tau} \int_0^{T - \tau} \left(\int_0^{\infty} x^*(t + \tau - \theta)g(\theta)d\theta \right) \cdot x^*(t) dt. \end{aligned} \quad (4.62)$$

Порівнюючи ліву частину рівняння (4.62) з виразом (4.51), бачимо, що це є оцінка $K_{yx}^*(\tau)$ взаємної кореляційної функції ергодичних випадкових процесів $X(t)$, $Y(t)$.

Як відомо з курсу математики, порядок інтегрування у подвійних інтегралах можна міняти місцями, оскільки ця операція є лінійною.

З урахуванням цих двох зауважень перепишемо рівняння (4.62) таким чином:

$$K_{yx}^*(\tau) = \int_0^{\infty} \left[\frac{1}{T - \tau} \int_0^{T - \tau} x^*(t + \tau - \theta) \cdot x^*(t) dt \right] g(\theta) d\theta. \quad (4.63)$$

Порівнюючи вираз, що стоїть у квадратних дужках у правій частині рівняння (4.63), з рівнянням (4.37), бачимо, що це є оцінка $K_x^*(\tau - \theta)$ кореляційної функції ергодичного випадкового процесу $X(t)$. З урахуванням цього зауваження перепишемо рівняння (4.63) таким чином:

$$K_{yx}^*(\tau) = \int_0^{\infty} K_x^*(\tau - \theta) g(\theta) d\theta. \quad (4.64)$$

У правій частині рівняння (4.64), як і у правій частині рівняння (4.60), теж маємо згортку, але, на відміну від стохастичної згортки в (4.60), в (4.64) маємо вже детерміновану згортку, оскільки у теоретичному плані кореляційна функція випадкового процесу ϵ , на відміну від нього самого, функцією детермінованою.

Рівняння (4.64), яке у різний час незалежно один від одного і різними способами вивели відомі математики Вінер і Хопф, так і називають *рівнянням Вінера-Хопфа*.

Оскільки Вінер прийшов до цього рівняння, розв'язуючи задачу оптимальної фільтрації «зашумленого» сигналу $x(t)$ фільтром із ваговою характеристикою $g(t)$ шляхом мінімізації квадратичного критерію якості J , який має вигляд

$$J = \frac{1}{T} \int_0^T \left(y^*(t) - \int_0^{\infty} x^*(t - \theta) g(\theta) d\theta \right)^2 dt, \quad (4.65)$$

то можна стверджувати, що функція $g(t)$, якщо вона знайдена шляхом розв'язання рівняння Вінера-Хопфа (4.64), є не просто одним із розв'язків цього рівняння, а розв'язком оптимальним за квадратичним критерієм (4.65).

У першій частині цього посібника показано, що вагова характеристика $g(t)$ неперервної лінійної системи з зосередженими параметрами є однією із форм подання математичної моделі цієї системи, а тому, визначаючи цю характеристику шляхом розв'язання рівняння Вінера-Хопфа (4.64), ми розв'язуємо задачу ідентифікації неперервної лінійної динамічної системи з зосередженими параметрами, в якій протікає стохастичний процес.

Способів розв'язання рівняння (4.64), яке відносять до класу некоректних, існує дуже багато.

Найпростішим є спосіб розв'язання системи алгебраїчних рівнянь відносно числових значень $g(\theta_0), g(\theta_1), g(\theta_2), \dots, g(\theta_{n-1})$, які формуються з інтегрального рівняння (4.64) переходом від неперервного інтегрування у нескінченних границях до знаходження скінченних сум відносно фіксованих зсувів $\tau_0, \tau_1, \dots, \tau_{n-1}$, тобто

$$K_{yx}^*(\tau_i) = \sum_{j=0}^{n-1} K_x^*(\tau_i - \theta_j) g(\theta_j) \Delta\theta_j, \quad i = \overline{0, n-1}. \quad (4.66)$$

Варто запам'ятати, що для збіжності розв'язку систем рівнянь (4.66) необхідно квантувати час θ і зсув τ рівними відрізками, тобто

$$\begin{aligned}\Delta \tau_i &= \tau_{i+1} - \tau_i = \Delta \theta_j = \theta_{j+1} - \theta_j, \\ i &= \overline{0, n-1}; \quad j = \overline{0, n-1}.\end{aligned}\quad (4.67)$$

Варто пам'ятати також і про те, що згідно з виразом (4.39)

$$K_x^*(\tau_i - \theta_j) = K_x^*(\theta_j - \tau_i). \quad (4.68)$$

Для того, щоб розв'язати рівняння Вінера-Хопфа (4.64) у частотній області, помножимо його ліву і праву частини на $e^{-j\omega\tau}$ і проінтегруємо у нескінченних границях. Виконавши це, маємо

$$\int_{-\infty}^{\infty} K_{yx}^*(\tau) e^{-j\omega\tau} d\tau = \int_{-\infty}^{\infty} \left(\int_0^{\infty} K_x^*(\tau - \theta) g(\theta) d\theta \right) e^{-j\omega\tau} d\tau. \quad (4.69)$$

Порівнюючи ліву частину рівняння (4.69) з виразом (4.55), бачимо, що це є оцінка взаємної спектральної густини $S_{yx}^*(j\omega)$ випадкових процесів $X(t)$ та $Y(t)$.

У правій частині рівняння (4.69) здійснимо заміну змінних. Покладемо, що

$$\tau - \theta = \lambda. \quad (4.70)$$

Тоді матимемо:

$$\begin{cases} \tau = \lambda + \theta, \\ d\tau = d\lambda. \end{cases} \quad (4.71)$$

З урахуванням цієї заміни перепишемо рівняння (4.69) таким чином:

$$S_{yx}^*(j\omega) = \int_0^{\infty} g(\theta) e^{-j\omega\theta} d\theta \int_{-\infty}^{\infty} K_x(\lambda) e^{-j\omega\lambda} d\lambda. \quad (4.72)$$

Оскільки

$$\int_0^{\infty} g(\theta) e^{-j\omega\theta} d\theta = W(j\omega) \quad (4.73)$$

– це амплітудно-фазова частотна характеристика (АФЧХ) системи, визначення якої здійснено у першій частині посібника, то, з урахуванням виразу (4.53), рівняння (4.72) можна переписати у вигляді

$$S_{yx}^*(j\omega) = W(j\omega) S_x^*(\omega). \quad (4.74)$$

Легко бачити, що розв'язком рівняння Вінера-Хопфа (4.64) у частотній області є вираз

$$W(j\omega) = \frac{S_{yx}^*(j\omega)}{S_x^*(\omega)}, \quad (4.75)$$

яким визначається АФЧХ $W(j\omega)$ системи через оцінки спектральних густин $S_x^*(\omega)$, $S_{yx}^*(j\omega)$ випадкових процесів $X(t)$, $Y(t)$, що мають місце на вході та виході цієї системи.

Варто зазначити, що простий за зовнішніми ознаками розв'язок рівняння Вінера-Хопфа у вигляді (4.75) досить серйозні труднощі свого отримання переносить на знаходження аналітичного виразу для оцінки взаємної спектральної густини $S_{yx}^*(j\omega)$ по несиметричній кривій $K_{yx}^*(\tau)$, для якої дуже непросто здійснити зручну для інтегрування при перетворенні за Фур'є аналітичну апроксимацію.

Як буде показано у наступному підрозділі, найзручнішим способом розв'язання рівняння Вінера-Хопфа є Фур'є-інтегральний метод ідентифікації (ФІМІ), основи якого для детермінованих систем викладені у першій частині посібника.

4.4 Фур'є-інтегральний метод ідентифікації стохастичних неперервних ЛДС ЗП

Вихідних передумов для можливості застосування Фур'є-інтегрального методу ідентифікації (ФІМІ) для синтезу математичної моделі стохастичної неперервної ЛДС ЗП є дві, а саме:

1) стохастичний процес у системі повинен належати до класу ергодичних, що дає право на використання рівняння Вінера-Хопфа (4.64);

2) кореляційна функція $K_x(\tau)$ вхідного сигналу $x(t)$ та взаємна кореляційна функція $K_{yx}(\tau)$ сигналів $x(t)$ та $y(t)$, де $y(t)$ – реакція системи на вхідний сигнал $x(t)$, повинні забезпечувати можливість подання їх рядами Фур'є.

4.4.1 Отримання основних розрахункових співвідношень та побудова алгоритму ідентифікації

Нехай τ_{ex} – час взаємної кореляції вхідного $x(t)$ і вихідного $y(t)$ стохастичних сигналів системи (див. рис. 4.10, 4.12). І нехай виконуються передумови, визначені вище.

Тоді на відрізку $[-\tau_{ex}, \tau_{ex}]$ обидві кореляційні функції $K_x(\tau)$ і $K_{yx}(\tau)$ можна розкласти у ряди Фур'є у вигляді:

$$K_x(\tau) = \frac{a_0}{2} + \sum_{i=1}^{\infty} a_i \cos i\omega_1 \tau, \quad (4.76)$$

$$K_{yx}(\tau) = \frac{m_0}{2} + \sum_{i=1}^{\infty} m_i \cos i\omega_1 \tau + \sum_{i=1}^{\infty} n_i \sin i\omega_1 \tau, \quad (4.77)$$

де

$$\omega_1 = \frac{\pi}{\tau_{ex}}; \quad i=0, 1, 2, \dots;$$

$$\left\{ \begin{matrix} a_i \\ m_i \end{matrix} \right\} = \frac{1}{\tau_{ex}} \int_{-\tau_{ex}}^{\tau_{ex}} \left\{ \begin{matrix} K_x(\tau) \\ K_{yx}(\tau) \end{matrix} \right\} \cos i\omega_1 \tau \, d\tau; \quad (4.78)$$

$$n_i = \frac{1}{\tau_{ex}} \int_{-\tau_{ex}}^{\tau_{ex}} K_{yx}(\tau) \sin i\omega_1 \tau \, d\tau. \quad (4.79)$$

Звертаємо увагу на те, що функція $K_x(\tau)$ є парною (див. рис. 4.8), тому її розкладання у ряд Фур'є не містить синусоїд.

Підставляючи ряди (4.76), (4.77) у рівняння Вінера-Хопфа (4.64), після тривіальних перетворень отримаємо:

$$\begin{aligned} & \frac{m_0}{2} + \sum_{i=1}^{\infty} m_i \cos i\omega_1 \tau + \sum_{i=1}^{\infty} n_i \sin i\omega_1 \tau = \\ & = \frac{a_0}{2} R(0) + \sum_{i=1}^{\infty} a_i R(i\omega_1) \cos i\omega_1 \tau - \sum_{i=1}^{\infty} a_i Q(i\omega_1) \sin i\omega_1 \tau, \end{aligned} \quad (4.80)$$

де $R(i\omega_1)$ та $Q(i\omega_1)$ – значення, відповідно, дійсної та уявної частотних характеристик ЛДС ЗП, взятих при значеннях частоти $i\omega_1$:

$$\begin{cases} R(i\omega_1) = R(\omega) \big|_{\omega=i\omega_1}, \\ Q(i\omega_1) = Q(\omega) \big|_{\omega=i\omega_1}. \end{cases} \quad (4.81)$$

З тотожності (4.80), як наслідок, маємо:

$$\begin{cases} m_i = a_i R(i\omega_1); \quad i = 0, 1, 2, \dots; \\ n_i = -a_i Q(i\omega_1); \quad i = 1, 2, \dots, \end{cases} \quad (4.82)$$

$$\begin{cases} R(i\omega_1) = \frac{m_i}{a_i}; \quad i = 0, 1, 2, \dots; \\ Q(i\omega_1) = -\frac{n_i}{a_i}; \quad i = 1, 2, \dots. \end{cases} \quad (4.83)$$

Отриманням формул (4.83) закінчується перший етап побудови алгоритму параметричної ідентифікації стохастичних неперервних ЛДС ЗП за допомогою ФІМІ.

Другий етап побудови алгоритму у цій задачі є ідентичним другому етапу розв'язання задачі, описаної у підрозділі 3.3 першої частини посібника. Відмінність полягає лише у тому, що $R(i\omega_1)$, $Q(i\omega_1)$ у нашому випадку визначаються за формулами (4.83) цього підрозділу, а не за формулами (3.23), (3.24) першої частини посібника – всі інші формули і дії збігаються, тому немає потреби їх знову наводити.

4.4.2 Обґрунтування регуляризовальних властивостей ФІМІ

Рівняння Вінера-Хопфа (4.64) належить до класу некоректних, тобто таких, в яких малі зміни значень коефіцієнтів можуть призводити до суттєвих змін у значеннях розв'язків.

Для отримання розв'язків таких рівнянь із прийнятною для практичних задач точністю А. Н. Тихонов, а також А. М. Дмитрієв і В. В. Солодовніков запропонували відповідно регуляризовальний та квазірегуляризовальний алгоритми.

Регуляризація за Тихоновим є швидше математичним мистецтвом, аніж чітким алгоритмом. Отже, використовується вона на практиці в інженерних задачах рідко, хоча дає найбільш точні результати.

Квазірегуляризовальний алгоритм Дмитрієва-Солодовнікова, оснований на ортогональному методі моментів, застосовується частіше, оскільки, маючи досить просту обчислювальну схему, дає прийнятні для цілей практики результати.

Регуляризовальна властивість ортогонального методу моментів полягає у процедурі обчислення моментів непарного порядку. Під час їх визначення за допомогою інтегральних формул одночасно здійснюється і фільтрація даних від завад із нульовим середнім. І якби під час розв'язання задач ідентифікації використовувались лише моменти непарного порядку, то ефективність алгоритму, оснований на ортогональному методі моментів, була б не гіршою регуляризованого алгоритму Тихонова за суттєво простішої структури і менших затратах часу на розв'язання задач. Зниження ж ефективності цього алгоритму пояснюється тим, що разом із моментами непарного порядку він використовує також і моменти парного порядку, для яких характерним є те, що навіть у разі незалежної адитивної завади вони є сумою відповідних моментів сигналу і завади. Тож, тригонометричні ряди Фур'є, під час визначення коефіцієнтів яких завжди здійснюється фільтрація адитивної завади з нульовим середнім, задають більший ступінь регуляризації, ніж будь-яка інша ортогональна система функцій.

Регуляризовальна властивість ФІМІ, який використовує ряди Фур'є, підсилюється додатково ще й тим, що за допомогою критеріальних співвідношень (3.27), (3.28) (дивись підрозділ 3.3 першої частини посібника) під час розв'язання задач ідентифікації паралельно здійснюється як параметрична оптимізація математичної моделі, так і оптимізація її структури.

Тож можна стверджувати, що для ФІМІ характерна властивість саморегуляризації, завдяки якій цей метод дозволяє отримувати робастні моделі, тобто такі, які дають цілком прийнятні для цілей практики результати за умов використання суттєво «зашумлених» вхідних даних.

4.5 Еквівалентні моделі функцій та густин розподілу випадкових величин

При системному аналізі усталених режимів стохастичних динамічних систем часто виникає необхідність у синтезі законів розподілу ймовірностей

попадання тих чи інших стохастичних параметрів, що характеризують ці усталені режими, в задані діапазони значень - і при цьому для багатьох з них наявних експериментальних даних недостатньо для отримання повноцінної статистики. Тож набуває актуальності задача створення методу визначення цих законів розподілу в якомусь еквівалентному варіанті, що ґрунтується на малих вибірках, щоб з його використанням можна було обчислювати потрібні для аналізу характеристики стохастичних об'єктів дослідження з похибками, які не виходили б за припустимі межі.

В класичній математичній статистиці синтез законів розподілу ймовірностей випадкових величин починається з побудови гістограм та їх «вирівнювання» за процедурою Пірсона однією із запропонованих різними математиками в різні роки теоретичною функцією розподілу, але ми запропонуємо інший підхід, виходячи з наступного.

У будь-якому підручнику чи навчальному посібнику з теорії ймовірностей, виданому як в Україні, так і за її межами, ми знайдемо інформацію про те, що одними із базових понять цієї науки є поняття законів розподілу випадкових величин X в їх інтегральному $F(x)$ чи диференціальному $f(x)$ вираженні, що зв'язані між собою і характеризуються уже уведеними нами співвідношеннями (4.2)-(4.6).

А синтез та ідентифікація математичних моделей інтегрального $F(x)$ та диференціального $f(x)$ законів розподілу випадкової величини X , заданої обмеженою множиною її відомих значень, є змістом іншої навчальної дисципліни, яка називається математичною статистикою і з навчальних посібників з якої, виданих як в Україні, так і за кордоном, легко пересвідчитись, що процедура синтезу диференціального $f(x)$ закону розподілу випадкової величини X починається з побудови гістограми, що являє собою сходинову лінію над віссю x , висота кожної сходинок якої дорівнює відношенню кількості значень випадкової величини, що попали в межі цієї сходинок, до загальної кількості значень випадкової величини, заданої обмеженою множиною усіх її відомих значень.

А другим етапом цієї класичної процедури синтезу, тобто, етапом ідентифікації диференціального $f(x)$ закону розподілу випадкової величини X , є алгоритм «вирівнювання» сходинок гістограми гіпотетичними неперервними кривими розподілу, запропонованими відомими математиками різних епох, починаючи з Гаусса, з використанням χ^2 –розподілу Пірсона.

І ось саме цей другий етап процедури синтезу диференціального закону розподілу $f(x)$ випадкової величини X - етап «вирівнювання» сходинок гістограми гіпотетичними неперервними кривими розподілу є найбільш проблематичним, оскільки при малих потужностях відомих значень множин випадкових величин, з використанням яких здійснюється ідентифікація, важко домогтись прийняттого значення довірчої ймовірності для «вирівнюючої» кривої, а при великих потужностях цих множин не вдається ідентифікувати гістограму однією із відомих теоретичних кривих, запропонованих різними математиками. Спроби ж об'єднання кількох кривих в один ансамбль для

ідентифікації гістограми складного профілю не відповідає принципу коректного розв'язання цієї задачі.

Тож ми пропонуємо долати невизначеність цієї проблемної ситуації шляхом синтезу еквівалентних моделей законів розподілу випадкових величин, придатних для обчислення ймовірностей попадання цих величин в заданий діапазон значень, для побудови яких не потрібна процедура «вирівнювання» гістограм відомими функціями розподілу з використанням χ^2 –розподілу Пірсона.

Почнемо процедуру створення методу синтезу еквівалентних моделей з побудови гістограми, яка, як відомо, є статистичною оцінкою функції, що апроксимує густину (щільність) розподілу $f(x)$ випадкової величини X і яка є фігурою, обмеженою зверху сходявкою лінією, а знизу віссю абсцис, з площею, що дорівнює одиниці.

Для подальших викладок, скориставшись Python-програмою, приведеною нижче, побудуємо гіпотетичну гістограму типового характеру випадкової величини X , не конкретизуючи множину значень цієї випадкової величини, але з дотриманням для її гістограми умови, заданої виразом (4.6) – це буде приклад № 12

```
In [1]: import numpy as np
In [2]: import matplotlib
In [3]: import matplotlib.pyplot as plt
In [7]: plt.bar(locs,data,\
width=0.99,color='blue')
In [4]: locs=np.arange(5)
In [5]: L=[0.1,0.15,0.35,0.25,0.15]
In [6]: data=np.array(L)
```

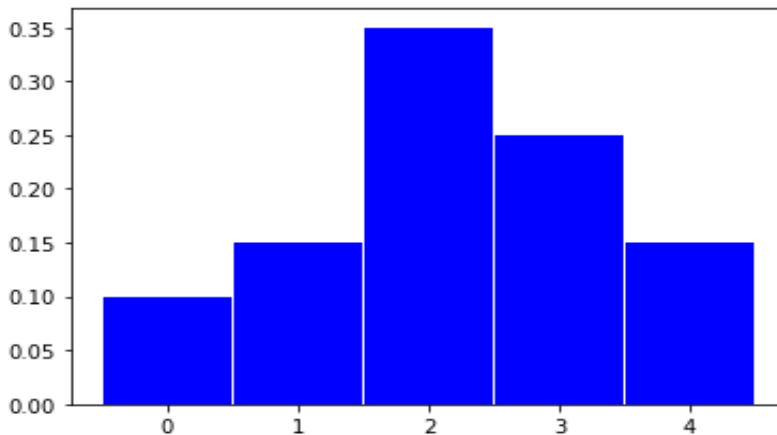


Рисунок 4.13 – Гіпотетична гістограма випадкової величини X , номери одиничної ширини розрядів якої проставлено в їх середині

Аналізуючи гістограму, згенеровану Python-програмою і приведену на рис. 4.13, бачимо, що навіть у цьому структурно і контурно простому її варіанті важко передбачити, в який бік вона деформується при збільшенні кількості використаних для її побудови значень випадкової величини – чи у бік «підтягування» до більш високого порогу перших двох розрядів з лівого боку від максимуму, що демонструватиме її наближення в граничному варіанті до нормального закону розподілу Гауса, що описується осьовим зрізом «дзвіницевої» поверхні, чи у бік «опущення» до ще менших значень порогів перших двох розрядів з лівого боку від максимуму та «підвищення» до ще

більших значень порогів останніх двох розрядів з правого боку від максимуму, що демонструватиме її наближення до β –розподілу. Тож, намагаючись «вирівняти» цю гістограму одним із цих двох розподілів, ми можемо отримати довірчу ймовірність апроксимації на рівні, меншому того, якому можна довіряти. І, якщо немає можливості додати до множини значень випадкової величини, згідно з якою будувалась гістограма, додаткових її значень, то з позицій класичної теорії ймовірностей та класичної математичної статистики задача переходить в категорію таких, що не мають коректного розв’язку.

Але в нашу комп’ютерну епоху, коли не виникає ускладнень у здійсненні будь-яких цифрових об’ємів обчислень, визначення будь-яких характеристик випадкових величин можна звести до комп’ютерних обчислень з використанням відповідних чисельно заданих їх законів розподілу. Тож реалізація етапу «вирівнювання» гістограм класичними функціями, якими в класичній математичній статистиці пропонується апроксимувати закони розподілу випадкових величин, стає не потрібною. І замість процедури «вирівнювання» гістограм доцільно переходити до процедури чисельного їх наближення до чисельно заданих функцій, які ми назвали еквівалентними моделями законів розподілу. Далі ми покажемо, як здійснити це наближення.

Отже почнемо з першого еквівалентного наближення функції $F(x)$, яке отримаємо кумулятивним сумуванням площ усіх розрядів гістограми, приведеної на рис. 4.13. Для цього використаємо як приклад № 13 Python-програму, яка має вигляд:

```
In [1]: import numpy as np
In [2]: import matplotlib
In [3]: import matplotlib.pyplot
        as plt
In [4]: locs=np.arange(5)

In [5]: L=[0.1,0.15,0.35,0.25,0.15]
In [6]: data=np.array(L)
In [7]: data1=np.cumsum(data)
In [8]: plt.bar(locs,data1,width=0.99,\
color='blue')
```

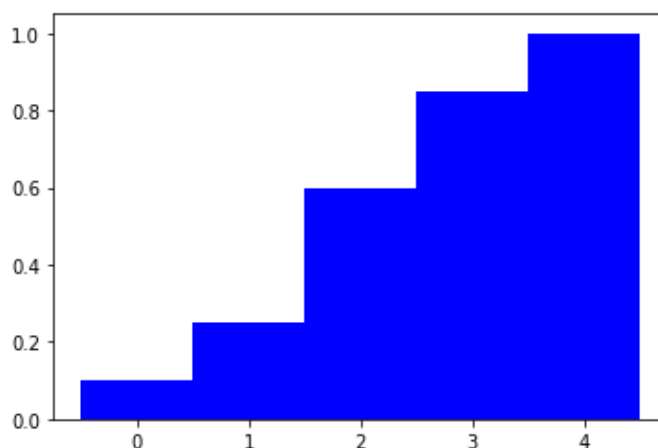


Рисунок 4.14 – Перше графічне еквівалентне наближення функції розподілу $F(x)$ випадкової величини X , отримане з гістограми рис. 4.13 у вигляді сходинок з числом сходинок, що дорівнює числу розрядів гістограми

А далі, використавши наступну Python-програму як приклад № 14, розіб'ємо кожний розряд гістограми, приведеної на рис. 4.13, на два підрозряди і за допомогою кумулятивного сумування площ усіх підрозрядів отримаємо друге еквівалентне наближення функції $F(x)$, зображене на рисунку 4.15.

In [1]: import numpy as np	Out[7]:
In [2]: import matplotlib	array([0.1, 0.1, 0.15, 0.15, 0.35, 0.35,\
In [3]: import matplotlib.pyplot as plt	0.25, 0.25, 0.15, 0.15])
In [4]: locs1=np.arange(10)	In [8]: data2=np.cumsum(data1/2)
In [5]: L=[0.1,0.15,0.35,0.25,0.15]	In [9]: plt.bar(locs1,data2,width=0.99,\
In [6]: data=np.array(L)	color='blue')
In [7]: data1=np.repeat(data,2);data1	

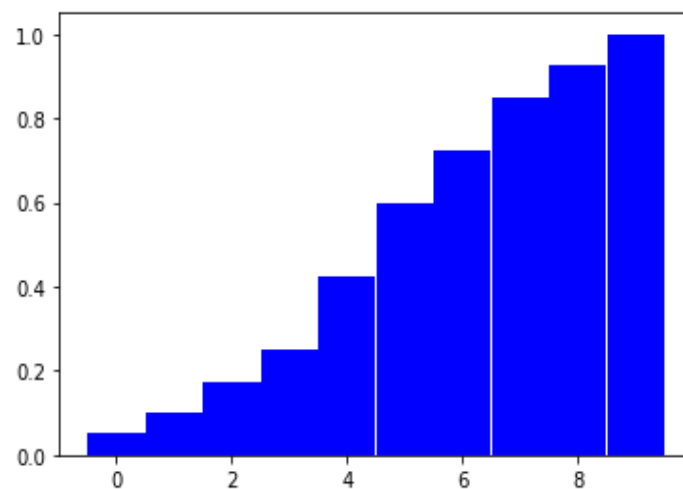


Рисунок 4.15 – Друге графічне еквівалентне наближення функції розподілу $F(x)$ випадкової величини X , отримане з гістограми рисунка 4.13 у вигляді сходинкової кривої з числом сходинок, що дорівнює подвоєному числу розрядів гістограми

Для однозначної інтерпретації процедури еквівалентування, використавши наведену далі Python-програму як приклад №15, отримаємо третє еквівалентне наближення функції $F(x)$, зображене на рисунку 4.16.

In [1]: import numpy as np	In [7]: data3=np.repeat(data,3)
In [2]: import matplotlib	In [8]: data4=np.cumsum(data3/3)
In [3]: import matplotlib.pyplot as plt	In [8]: plt.bar(locs2,data4,width=0.99,\
In [4]: locs2=np.arange(15)	color='blue')
In [5]: L=[0.1,0.15,0.35,0.25,0.15]	Кінець програми 13
In [6]: data=np.array(L)	

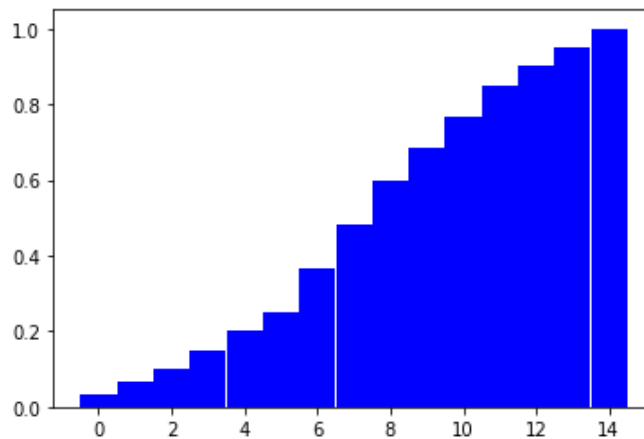


Рисунок 4.16 – Третє графічне еквівалентне наближення функції розподілу $F(x)$, отримане з гістограми рисунка 4.13

Із рис. 4.16 та формули (4.3) легко бачити, що ймовірність попадання масштабованої випадкової величини X^* в діапазон значень $[2,6]$ дорівнює 0,3, тобто у відповідності з формулою (4.3) маємо

$$P(2 < X^* \leq 6) = F(6) - F(2) = 0,4 - 0,1 = 0,3. \quad (4.84)$$

Якщо ж нам в подальших математичних перетвореннях законів розподілу випадкової величини необхідно мати аналітичну модель її диференціального закону розподілу $f(x)$, то необхідно спочатку здійснити інтерполяційну ідентифікацію сходинок кривої функції розподілу $F(x)$, яка відноситься до класу нульвимірних, бо в точках початку сходінок є недиференційованою, а потім до результату інтерполяції застосувати вираз (4.4), яким ми визначимо математичну модель неперервного диференціального закону розподілу $f(x)$ теж в еквівалентному варіанті – в цьому і полягає суть трансформації дискретних інтегральних законів розподілу випадкових величин, синтезованих на множині міри «нуль», в неперервні диференціальні закони їх густин (щільностей).

Алгоритм розв’язання задачі інтерполяції з використанням усіх варіантів інтерполяційних поліномів до сплайнів включно дуже детально розписаний і проілюстрований прикладами в роботах професора Романа Кветного і його наукової школи, тож і ми ними скористаємось.

Отже суть запропонованого методу трансформації дискретних інтегральних законів розподілу випадкових величин, синтезованих на множині міри «нуль», в неперервні диференціальні закони їх густин (щільностей), полягає на першому етапі в подрібненні базової гістограми, зображеної на рис. 4.13, з наступним кумулятивним сумуванням за приведеними вище Python-програмами для отримання сходинок еквівалентної функції розподілу, зображеної на рис. 4.16 після трикратного подрібнення з наступним сумуванням.

А на другому етапі реалізації цього методу необхідно здійснити інтерполяцію останньої версії статистичної оцінки кумулятивного варіанту

еквівалентної сходиноквої функції розподілу випадкової величини, представленої на рис. 4.16, за допомогою кубічних сплайнів.

А на третьому етапі запропонованого нами методу трансформації ми пропонуємо прямим диференціюванням сплайн-інтерпольованої функції розподілу випадкової величини синтезувати еквівалентну математичну модель густини (щільності) розподілу даних випадкового характеру.

Оскільки перший етап запропонованого методу трансформації еквівалентних математичних моделей, що характеризують випадкові величини до Python-програм їх реалізації включно, уже розкритий вище, то далі ми зосередимо увагу лише на розкритті другого та третього етапів цього методу.

Почнемо розв'язання поставленої задачі з синтезу сплайн-моделі еквівалентної функції розподілу випадкової величини, гістограма якої представлена на рис. 4.16, до граничних точок розрядів якої ми будемо «прив'язувати» відповідні кубічні сплайни, пам'ятаючи при цьому, що після трикратного подрібнення базової гістограми, представленої на рис. 4.13, і переходу до її еквіваленту, представленому на рис. 4.16, ординаті з абсцисою 1 на цьому рисунку буде відповідати ордината з абсцисою 3, ординаті з абсцисою 2 буде відповідати ордината з абсцисою 6, ординаті з абсцисою 3 буде відповідати ордината з абсцисою 9, ординаті з абсцисою 4 буде відповідати ордината з абсцисою 12, а ординаті з абсцисою 5 буде відповідати ордината з абсцисою 15.

Усі ці абсциси і ординати, що визначні по рис. 4.16, зведені в таблицю 4.1

Таблиця 4.1. Поле точок з гістограми, приведеної на рис. 4.16

I	0	1	2	3	4	5
x_i	x_0	x_1	x_2	x_3	x_4	x_5
y_0	0,03					
y_1		0,07				
y_2			0,15			
y_3				0,40		
y_4					0,70	
y_5						1,0

В процесі сплайн-інтерполяції необхідно розв'язати систему із 20 рівнянь з 20 невідомими, складену з дотриманням умов рівності в дотичних точках базових для сусідніх відрізків сплайнів, умов рівності їх перших та других похідних у цих точках, а також умов рівності нулю других похідних на кінцях поля базової гістограми, для розв'язання якої нами теж розроблена програма мовою Python.

Запишемо цю систему рівнянь відносно сплайнів

$$s_j = \sum_{i=0}^3 c_{ij} x_j^i, \quad j = 1, 2, \dots, 5: \quad (4.85)$$

та похідних від них

$$f_j = \sum_{i=1}^3 i c_{ij} x_j^{i-1}, \quad j = 1, 2, \dots, 5 \quad (4.86)$$

з врахуванням вище викладених умов.

В результаті отримаємо:

$$\left\{ \begin{array}{l} c_{01} = 0.03, \\ c_{01} + c_{11} + c_{21} + c_{31} = 0.07, \\ c_{02} + c_{12} + c_{22} + c_{32} = 0.07, \\ c_{02} + 2c_{12} + 4c_{22} + 8c_{32} = 0.15, \\ c_{03} + 2c_{13} + 4c_{23} + 8c_{33} = 0.15, \\ c_{03} + 3c_{13} + 9c_{23} + 27c_{33} = 0.40 \\ c_{04} + 3c_{14} + 9c_{24} + 27c_{34} = 0.40, \\ c_{04} + 4c_{14} + 16c_{24} + 64c_{34} = 0.70, \\ c_{05} + 4c_{15} + 16c_{25} + 64c_{35} = 0.70, \\ c_{05} + 5c_{15} + 25c_{25} + 125c_{35} = 1.0 \\ c_{11} + 2c_{21} + 3c_{31} = c_{12} + 2c_{22} + 3c_{32}, \\ c_{12} + 4c_{22} + 12c_{32} = c_{13} + 4c_{23} + 12c_{33}, \\ c_{13} + 6c_{23} + 27c_{33} = c_{14} + 6c_{24} + 27c_{34}, \\ c_{14} + 8c_{24} + 48c_{34} = c_{15} + 8c_{25} + 48c_{35}, \\ 2c_{21} + 6c_{31} = 2c_{22} + 6c_{32}, \\ 2c_{22} + 12c_{32} = 2c_{23} + 12c_{33}, \\ 2c_{23} + 18c_{33} = 2c_{24} + 18c_{34}, \\ 2c_{24} + 24c_{34} = 2c_{25} + 24c_{35}, \\ 2c_{25} + 30c_{35} = 0, \\ 2c_{21} = 0 \end{array} \right. \quad (4.87)$$

Для розв'язання системи алгебраїчних рівнянь (4.87) використаємо як приклад №16 Python-програму, записану нижче.

```

In [1]: import sympy
In [2]: from sympy import*
In [3]: c01,c11,c21,c31=symbols('c01
c11 c21 c31')
In [4]: c02,c12,c22,c32=symbols('c02
c12 c22 c32')
In [5]: c03,c13,c23,c33=symbols('c03
c13 c23 c33')
In [6]: c04,c14,c24,c34=symbols('c04
c14 c24 c34')
In [7]: c05,c15,c25,c35=symbols('c05
c15 c25 c35')
In [8]: x=symbols('x')
In [9]: s1=Function('s1')(x)
In [10]: s2=Function('s2')(x)
In [11]: s3=Function('s3')(x)
In [12]: s4=Function('s4')(x)
In [13]: s5=Function('s5')(x)
In [14]: s1=c01+c11*x+c21*x**2+\
c31*x**3
In [15]: s2=c02+c12*x+c22*x**2+\
c32*x**3
In [16]: s3=c03+c13*x+c23*x**2+\
c33*x**3
In [17]: s4=c04+c14*x+c24*x**2+\
c34*x**3
In [18]: s5=c05+c15*x+c25*x**2+\
c35*x**3
In [19]: eq1=c01-0.03
In [20]: eq2=c01+c11+c21+c31-0.07
In [21]: eq3=c02+c12+c22+c32-0.07
In [22]: eq4=c02+2*c12+4*c22+\
8*c32-0.15
In [23]: eq5=c03+2*c13+4*c23+\
8*c33-0.15
In [24]: eq6=c03+3*c13+9*c23+\
27*c33-0.40
In [25]: eq7=c04+3*c14+9*c24+\
27*c34-0.40
In [26]: eq8=c04+4*c14+16*c24+\

```

```

64*c34-0.70
ln [27]: eq9=c05+4*c15+16*c25+\
64*c35-0.70
ln [28]: eq10=c05+5*c15+25*c25+\
125*c35-1.0
ln [29]: eq11=c11+2*c21+3*c31-c12-\
2*c22-3*c32
ln [30]: eq12=c12+4*c22+12*c32-c13\
-4*c23-12*c33
ln [31]: eq13=c13+6*c23+27*c33-c14\
-6*c24-27*c34
ln [32]: eq14=c14+8*c24+48*c34-c15\
ln [39]: solve([eq1,eq2,eq3,eq4,eq5,eq6,eq7,eq8,eq9,eq10,eq11,eq12,eq13,
eq14,eq15,eq16,eq17,eq18,eq19,eq20],c01,c11,c21,c31,c02,c12,c22,
c32,c03,c13,c23,c33,c04,c14,c24,c34,c05,c15,c25,c35)
Out[39]:
{c01: 0.030000000000000000, c11: 0.0405263157894737,
c21: 0.0, c31: -0.000526315789473684,
c02: -0.0131578947368421, c12: 0.17000000000000000,
c22: -0.129473684210526, c32: 0.0426315789473684,
c03: 0.647894736842105, c13: -0.821578947368421,
c23: 0.366315789473684, c33: -0.04000000000000000,
c04: -0.361052631578947, c14: 0.187368421052632,
c24: 0.030000000000000000, c34: -0.00263157894736842,
c05: -0.563157894736842, c15: 0.338947368421053,
c25: -0.00789473684210526, c35: 0.000526315789473684}
ln [40]: d1={c01: 0.030000000000000000, c11: 0.0405263157894737,
c21: 0.0,c31: -0.000526315789473684,
c02: -0.0131578947368421, c12: 0.17000000000000000,
c22: -0.129473684210526, c32: 0.0426315789473684,
c03: 0.647894736842105, c13: -0.821578947368421,
c23: 0.366315789473684, c33: -0.04000000000000000,
c04: -0.361052631578947, c14: 0.187368421052632,
c24: 0.030000000000000000, c34: -0.00263157894736842,
c05: -0.563157894736842, c15: 0.338947368421053,
c25: -0.00789473684210526, c35: 0.000526315789473684}
ln [41]: s11=s1.subs([(c01,d1[c01]),(c11,d1[c11]),(c21,d1[c21]),
(c31,d1[c31])]).evalf(3);s11
Out[41]:
-0.000526*x**3 + 0.0405*x + 0.03
ln [42]: s22=s2.subs([(c02,d1[c02]),(c12,d1[c12]),(c22,d1[c22]),
(c32,d1[c32])]).evalf(3);s22
Out[42]:
0.0426*x**3 - 0.129*x**2 + 0.17*x - 0.0132
ln [43]: s33=s3.subs([(c03,d1[c03]),(c13,d1[c13]),(c23,d1[c23]),
(c33,d1[c33])]).evalf(3);s33
Out[43]:
-0.04*x**3 + 0.366*x**2 - 0.822*x + 0.648

```

```

In [44]: s44=s4.subs([(c04,d1[c04]),(c14,d1[c14]),(c24,d1[c24]),
(c34,d1[c34])]).evalf(3);s44
Out[44]:
-0.00263*x**3 + 0.03*x**2 + 0.187*x - 0.361
In [45]: s55=s5.subs([(c05,d1[c05]),
(c15,d1[c15]),(c25,d1[c25]),
(c35,d1[c35])]).evalf(3);s55
Out[45]:
0.000526*x**3 - 0.00789*x**2 + 0.339*x -
0.563
In [46]: ds1=s11.diff();ds1
Out[46]:
0.0405 - 0.00158*x**2
In [47]: ds2=s22.diff();ds2
Out[47]:
0.128*x**2 - 0.259*x + 0.17
In [48]: ds3=s33.diff();ds3
Out[48]:
-0.12*x**2 + 0.733*x - 0.822
In [49]: ds4=s44.diff();ds4
Out[49]:
-0.00789*x**2 + 0.06*x + 0.187
In [50]: ds5=s55.diff();ds5
In [67]: Fr1(x)
Out[67]:
array([0.03      , 0.03202493, 0.03404947, 0.03607322, 0.03809579,
        0.04011678, 0.0421358 , 0.04415245, 0.04616634, 0.04817707,
        0.05018425, 0.05218749, 0.05418638, 0.05618055, 0.05816958,
        0.06015309, 0.06213069, 0.06410197, 0.06606655, 0.06802402,
        0.069974  ])
In [68]: Fr2=lambdify(y,expr2,"numpy")
In [69]: y=np.linspace(1,2,21)
In [70]: Fr2(y)
Out[70]:
array([0.0704      , 0.07239233, 0.0744106 , 0.07648678, 0.0786528 ,
        0.08094063, 0.0833822 , 0.08600947, 0.0888544 , 0.09194892,
        0.095325  , 0.09901457, 0.1030496 , 0.10746203, 0.1122838 ,
        0.11754688, 0.1232832 , 0.12952473, 0.1363034 , 0.14365118,
        0.1516  ])
In [71]: Fr3=lambdify(z,expr3,"numpy")
In [72]: z=np.linspace(2,3,21)
In [73]: Fr3(z)
Out[73]:
array([0.148 , 0.15641, 0.16542, 0.175 , 0.18512, 0.19575, 0.20686,
        0.21842, 0.2304 , 0.24277, 0.2555 , 0.26856, 0.28192, 0.29555,
        0.30942, 0.3235 , 0.33776, 0.35217, 0.3667 , 0.38132, 0.396  ])
In [74]: Fr4=lambdify(u,expr4,"numpy")
In [75]: u=np.linspace(3,4,21)

```

```

In [76]: Fr4(u)
Out[76]:
array([0.39899 , 0.413805 , 0.42864967, 0.44352205, 0.45842016,
       0.47334203, 0.48828569, 0.50324916, 0.51823048, 0.53322767,
       0.54823875, 0.56326176, 0.57829472, 0.59333566, 0.60838261,
       0.62343359, 0.63848664, 0.65353978, 0.66859103, 0.68363843,
       0.69868  ])
In [77]: Fr5=lambdify(v,expr5,"numpy")
In [78]: v=np.linspace(4,5,21)
In [79]: Fr5(v)
Out[79]:
array([0.700424 , 0.71547652, 0.73052155, 0.74555947, 0.76059069,
       0.77561559, 0.79063458, 0.80564805, 0.82065638, 0.83565999,
       0.85065925, 0.86565457, 0.88064634, 0.89563495, 0.9106208 ,
       0.92560428, 0.94058579, 0.95556572, 0.97054447, 0.98552243,
       1.0005  ])
In [80]: plt.plot(x,Fr1(x),'-r',y,Fr2(y),'-g',z,Fr3(z),'-c',u,Fr4(u),'-b',\
v,Fr5(v),'-r',linewidth=3)

```

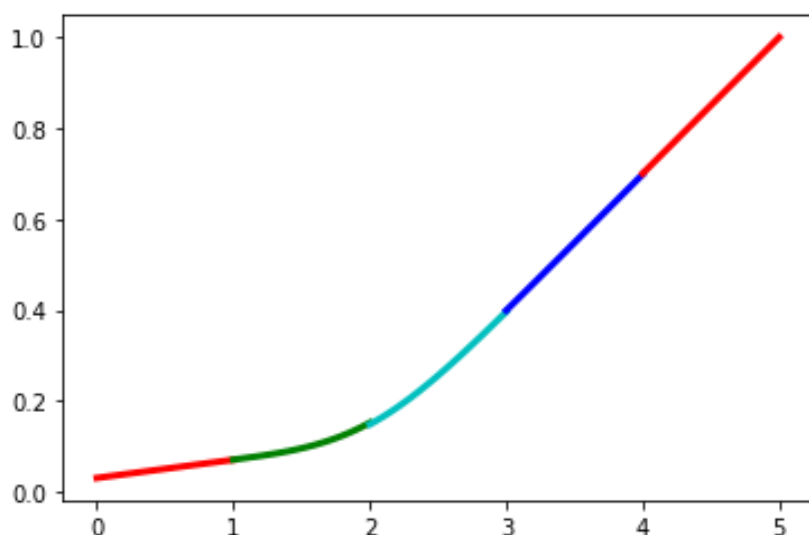


Рисунок 4.17 – Графік еквівалентної функції розподілу, синтезований Python-програмою на основі кубічних сплайнів

(Продовження програми):

```

In [81]: x=symbols('x')
In [82]: f1=lambdify(x,expr11,"numpy")
In [83]: x=np.linspace(0,1,21)
In [84]: f1(x)
Out[84]:
array([0.0405 , 0.04049605, 0.0404842 , 0.04046445, 0.0404368 ,
       0.04040125, 0.0403578 , 0.04030645, 0.0402472 , 0.04018005,
       0.040105 , 0.04002205, 0.0399312 , 0.03983245, 0.0397258 ,
       0.03961125, 0.0394888 , 0.03935845, 0.0392202 , 0.03907405,
       0.03892  ])
In [85]: y=symbols('y')

```

```

In [86]: f2=lambdify(y,expr22,"numpy")
In [87]: y=np.linspace(1,2,21)
In [88]: f2(y)
Out[88]:
array([0.039 , 0.03917, 0.03998, 0.04143, 0.04352, 0.04625, 0.04962,
       0.05363, 0.05828, 0.06357, 0.0695 , 0.07607, 0.08328, 0.09113,
       0.09962, 0.10875, 0.11852, 0.12893, 0.13998, 0.15167, 0.164 ])
In [89]: z=symbols('z')
In [90]: f3=lambdify(z,expr33,"numpy")
In [91]: z=np.linspace(2,3,21)
In [92]: f3(z)
Out[92]:
array([0.164 , 0.17635, 0.1881 , 0.19925, 0.2098 , 0.21975, 0.2291 ,
       0.23785, 0.246 , 0.25355, 0.2605 , 0.26685, 0.2726 , 0.27775,
       0.2823 , 0.28625, 0.2896 , 0.29235, 0.2945 , 0.29605, 0.297 ])
In [93]: u=symbols('u')
In [94]: f4=lambdify(u,expr44,"numpy")
In [95]: u=np.linspace(3,4,21)
In [96]: f4(u)
Out[96]:
array([0.29599 , 0.29660327, 0.2971771 , 0.29771148, 0.2982064 ,
       0.29866188, 0.2990779 , 0.29945447, 0.2997916 , 0.30008927,
       0.3003475 , 0.30056627, 0.3007456 , 0.30088548, 0.3009859 ,
       0.30104687, 0.3010684 , 0.30105047, 0.3009931 , 0.30089627,
       0.30076 ])
In [97]: v=symbols('v')
In [98]: f5=lambdify(v,expr55,"numpy")
In [99]: v=np.linspace(4,5,21)
In [100]: f5(v)
Out[100]:
array([0.30108 , 0.30092595, 0.3007798 , 0.30064155, 0.3005112 ,
       0.30038875, 0.3002742 , 0.30016755, 0.3000688 , 0.29997795,
       0.299895 , 0.29981995, 0.2997528 , 0.29969355, 0.2996422 ,
       0.29959875, 0.2995632 , 0.29953555, 0.2995158 , 0.29950395,
       0.2995 ])
In [101]: fig=plt.figure(facecolor='white')
<Figure size 432x288 with 0 Axes>
In [102]: plt.plot(x,f1(x),'-r',y,f2(y),'-g',z,f3(z),'-c',u,f4(u),'-b',\
v,f5(v),'-r',linewidth=3)
Out[102]:
[<matplotlib.lines.Line2D at 0x1f8ac834310>,
 <matplotlib.lines.Line2D at 0x1f8ac8343d0>,
 <matplotlib.lines.Line2D at 0x1f8ac8344c0>,
 <matplotlib.lines.Line2D at 0x1f8ac8341f0>,
 <matplotlib.lines.Line2D at 0x1f8ac834610>]

```

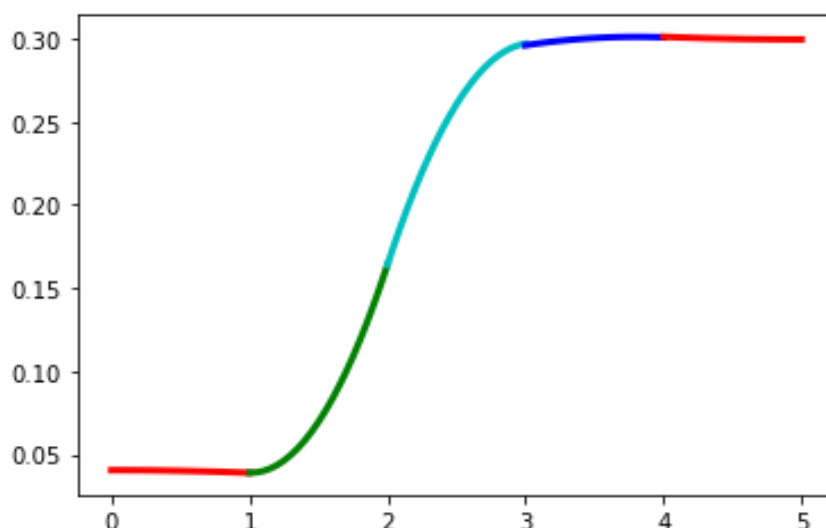


Рисунок 4.18 – Графік еквівалентної густини (щільності) розподілу, синтезований Python-програмою на основі кубічних сплайнів

Об'єднуючи результати, отримані після виконання команд In [41]-In [45], отримаємо еквівалентну сплайн-модель функції розподілу у вигляді

$$s = \begin{cases} -0.000526x^3 + 0.405x + 0.03, & x \in [0,1], \\ 0.0426x^3 - 0.129x^2 + 0.17x - 0.0132, & x \in [1,2], \\ -0.04x^3 + 0.366x^2 - 0.822x + 0.648, & x \in [2,3], \\ -0.00263x^3 + 0.03x^2 + 0.187x - 0.361, & x \in [3,4], \\ 0.000526x^3 - 0.00789x^2 + 0.339x - 0.563, & x \in [4,5]. \end{cases} \quad (4.88)$$

А об'єднуючи результати, отримані після виконання команд In [46]-In [50], отримаємо еквівалентну сплайн-модель густини (щільності) розподілу у вигляді

$$f = \begin{cases} -0.00158x^2 + 0.0405, & x \in [0,1], \\ 0.128x^2 - 0.259x + 0.17, & x \in [1,2], \\ -0.12x^2 + 0.733x - 0.822, & x \in [2,3], \\ -0.00789x^2 + 0.06x + 0.187, & x \in [3,4], \\ 0.00158x^2 - 0.0158x + 0.339, & x \in [4,5]. \end{cases} \quad (4.89)$$

Графіки еквівалентних сплайн-моделей (4.88),(4.89), що приведені на рисунках 4.17 та 4.18, підтверджують і правильність розв'язання поставленої задачі, і високу точність отриманих результатів, адже, якби точність була недостатньою, то кінці ліній об'єданого графіка на границях відрізків не співпадали б.

Підводячи підсумок викладеному вище у цьому підрозділі, можна стверджувати, що нами запропоновано метод синтезу еквівалентних моделей функцій розподілу та густин (щільностей) розподілу даних випадкового характеру, який не вимагає виконання процедури «вирівнювання» гістограм з використанням χ^2 –розподілу Пірсона в класичному варіанті синтезу цих моделей.

Запропонований метод синтезу базується на використанні кубічних сплайнів і їх лекальному припасуванні після трансформації гістограми у сходящову функцію розподілу і її подрібнення.

Графік густини (щільності) розподілу, представлений на рис. 4.18, яка набула властивості класичного β –розподілу в його граничному варіанті, підтверджує високу ефективність запропонованого способу синтезу, оскільки вихідну гістограму до такого розподілу класичним «вирівнюванням» за Пірсоном з високою довірчою ймовірністю привести не вдалося б.

А завершимо ми цю частину нашого викладу двома зауваженнями, по першому із яких варто акцентувати увагу на тому, що запропонований нами метод синтезу та ідентифікації еквівалентних моделей законів розподілу випадкових величин окрім своєї простоти в реалізації сприяє використанню в подальшому саме тих характеристик цієї випадкової величини, які проявились в тому обмеженому масиві її значень, з використанням якого ми побудували гістограму, і не нав'язує нам гіпотетичного віднесення цієї випадкової величини шляхом «вирівнювання» гістограм відомими теоретичними розподілами до класу тих, що визначені на нескінченних множинах.

А другим зауваженням ми хочемо привернути увагу читачів до нашого твердження, що цим методом ми не лише синтезуємо еквівалентні функції розподілу випадкових величин та густини (щільності) цього розподілу, але і прокладаємо шлях трансформації функції, визначений на множині міри «нуль», до множини неперервних функцій, що в математичному плані теж є непересічним результатом.

4.6 Завдання для самоперевірки

Які системи називають стохастичними та чому?

Дайте означення випадкового процесу та його реалізації.

Що і як задає функція розподілу ймовірностей випадкової величини? Які її властивості ви знаєте?

Що і як задає густина ймовірностей випадкової величини? Які її властивості ви знаєте?

Чим відрізняються функції розподілу та густини ймовірностей безперервних та дискретних випадкових величин?

Що таке математичне очікування випадкової величини? Як знайти його статистичну оцінку?

Що характеризує дисперсія випадкової величини? Як знайти її статистичну оцінку? Як пов'язані між собою дисперсія та середнє квадратичне відхилення випадкової величини?

Як визначаються математичне очікування та дисперсія випадкового процесу?

Який випадковий процес можна вважати стаціонарним?

У чому сутність ергодичності стаціонарного випадкового процесу?

Як знайти оцінки математичного очікування та дисперсії ергодичного випадкового процесу?

Яким вимогам повинні відповідати математичні моделі, які використовуються для визначення числових характеристик випадкових величин? Наведіть формули для визначення оцінки математичного очікування та дисперсії стаціонарного випадкового процесу, які задовольняють ці вимоги.

Що характеризує кореляційна функція випадкового процесу? Як вона визначається у загальному вигляді?

Як визначити кореляційну функцію стаціонарного випадкового процесу?

Як визначити оцінку кореляційної функції ергодичного випадкового процесу?

Які ви знаєте найбільш поширені в інженерному аналізі моделі кореляційних функцій ергодичних випадкових процесів?

Який зв'язок існує між кореляційною функцією та дисперсією випадкового процесу?

Що собою являє операція «центрування» випадкового процесу? Чому дорівнює математичне очікування центрованого процесу?

Як визначається взаємна кореляційна функція двох випадкових процесів?

Як визначити її статистичну оцінку, якщо процеси ергодичні?

Чому максимум взаємної кореляційної функції двох випадкових процесів має зсув вправо і що можна оцінити, знаючи числове значення цього зсуву?

Що характеризує спектральна густина випадкового процесу, і як вона визначається?

Що характеризує взаємна спектральна густина двох випадкових процесів, і як вона визначається?

Що собою являє «білий» шум, і які його характеристики ви знаєте?

Який вигляд має рівняння Вінера-Хопфа? Який спосіб його отримання ви знаєте?

Як за допомогою рівняння Вінера-Хопфа здійснити ідентифікацію стохастичної системи у часовому просторі?

Що собою являє дельта-функція і які її характеристики ви знаєте?

Як за допомогою рівняння Вінера-Хопфа здійснити ідентифікацію стохастичної системи у частотній області? Що ускладнює алгоритм ідентифікації у цій області?

У чому полягає основна ідея ФІМІ?

Чому ФІМІ можна застосовувати лише у стохастичних системах, яким притаманна властивість ергодичності?

Як розкласти кореляційну функцію у ряд Фур'є?

Чому кореляційна функція стохастичного процесу розкладається у ряд Фур'є, котрий містить лише косинуси?

На якому проміжку часу розкладається у ряд Фур'є взаємна кореляційна функція у разі застосування ФІМІ?

Як отримати основні розрахункові співвідношення ФІМІ стохастичних систем?

В якому класі моделей динамічних систем необхідно шукати розв'язок за допомогою ФІМІ?

Як використати квадратичний критерій у ФІМІ для пошуку оптимальних значень коефіцієнтів моделі?

Як у ФІМІ оптимізується структура моделі?

Для чого потрібна регуляризація алгоритму ідентифікації?

Чому ортогональний метод моментів дає оцінки моделі, які є близькими до оптимальних?

У чому полягає регуляризувальна сутність рядів Фур'є?

Чому ФІМІ належить до методів ідентифікації, для яких властива саморегуляризація алгоритму?

У чому сутність робастних моделей?

На основі якої ідеї реалізуються еквівалентні функції розподілу випадкової величини та густини цього розподілу?

Який механізм застосовано для отримання еквівалентних функцій розподілу випадкової величини та густини цього розподілу?

Як скласти систему рівнянь для апроксимації еквівалентних функцій розподілу випадкової величини з використанням кубічних сплайнів?

Як отримати еквівалентну функцію густини розподілу випадкової величини на множині кубічних сплайнів?

Розділ 5 СТОХАСТИЧНІ МОДЕЛІ ДИСКРЕТНИХ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ НА ОСНОВІ ЧАСОВИХ РЯДІВ

5.1 Часові ряди та їх основні характеристики

У першій частині цього посібника та у попередніх розділах другої частини викладені методи побудови математичних моделей ЛДС ЗП, придатних для аналізу процесів у цих системах у даний момент часу та синтезу систем автоматичного керування ними.

Але не менш важливою є задача прогнозування розвитку процесу у системі в майбутньому, для розв'язання якої ті математичні моделі, що вже розглянуті, непридатні.

На сьогоднішній день найбільш ефективними математичними моделями, за допомогою яких можна прогнозувати розвиток процесів, є ті, під час побудови яких використовуються часові ряди.

Нагадаємо, що часовий ряд – це сукупність значень випадкового процесу, взятих через рівні проміжки часу t . Позначимо цю сукупність символом Z_t .

Фактично Z_t – це є дискретний у часі випадковий процес.

Задача прогнозу полягає в тому, щоб за значеннями процесу у момент t спрогнозувати його значення в момент $t + l$, де l – час упередження. Щоб відрізнити прогнозне значення процесу від дійсного, дійсне значення часового ряду у момент $t + l$ позначають символом Z_{t+l} , а прогнозне значення – символом $z_t(l)$.

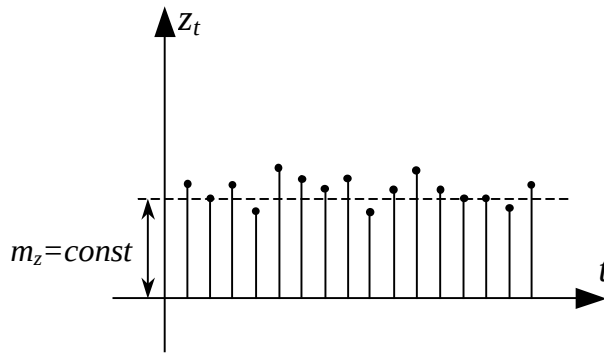
Зрозуміло, що точно спрогнозувати значення випадкового процесу, яким є часовий ряд, неможливо в принципі, а тому прогноз здійснюють, домагаючись мінімуму якогось критерію адекватності прогнозової моделі.

Якщо значення t є малим (1, 2 кроки), то одним із таких критеріїв може бути дисперсія відхилення $z_t(l)$ від Z_{t+l} , яка повинна бути для оптимальної моделі прогнозу мінімальною, тобто

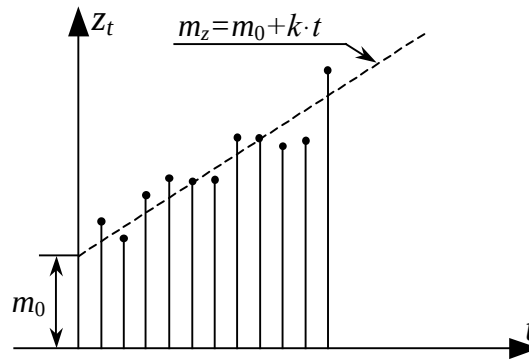
$$E \left\{ \left(z_{t+l} - z_t(l) \right)^2 \right\} \rightarrow \min, \quad (5.1)$$

де E – символ операції математичного очікування. Але, звичайно ж, як продемонстровано нами у першому виданні цього навчального посібника, можуть бути запропоновані й інші критерії оптимальності моделей, однак у нинішньому другому виданні нашого посібника ми їм приділяти уваги не будемо, розраховуючи на те, що в разі, якщо когось із читачів вони зацікавлять, то вони їх віднайдуть у першому виданні.

Як і будь-який інший випадковий процес, часовий ряд Z_t може бути стаціонарним (рис. 5.1,а) або нестаціонарним (рис. 5.1,б).



а)



б)

Рисунок 5.1 – Графіки реалізації стаціонарного (а) та нестаціонарного (б) часових рядів

Для стаціонарного часового ряду характерною є рівновага його значень Z_t біля середнього значення m_z , яке є константою.

Для нестаціонарного часового ряду ковізне середнє значення $m_z(t)$ процесу є функцією часу t , як показано на рис. 5.1,б.

Введемо ряд корисних операторів, які будуть потрібними у подальшому.

1. Оператор B зсуву назад на одну одиницю часу. Цей оператор формує рівняння

$$z_{t-1} = Bz_t. \quad (5.2)$$

Зрозуміло, що згідно з (5.2) справедливим є і рівняння

$$z_{t-2} = Bz_{t-1}. \quad (5.3)$$

Підставляючи значення z_{t-1} із виразу (5.2) в (5.3), отримаємо:

$$z_{t-2} = B(Bz_t) = B^2 z_t. \quad (5.4)$$

Узагальнюючи вираз (5.4), маємо

$$z_{t-m} = B^m z_t. \quad (5.5)$$

2. Оператор F зсуву вперед на одну одиницю часу. Цей оператор формує рівняння

$$z_{t+1} = Fz_t. \quad (5.6)$$

Зрозуміло, що згідно з виразом (5.6) справедливим є і рівняння

$$z_{t+2} = Fz_{t+1}. \quad (5.7)$$

Підставляючи значення z_{t+1} із виразу (5.6) в (5.7), отримаємо:

$$z_{t+2} = F(Fz_t) = F^2 z_t. \quad (5.8)$$

Узагальнюючи вираз (5.8), маємо:

$$z_{t+m} = F^m z_t. \quad (5.9)$$

3. Різницевий оператор ∇ із зсувом назад на одну одиницю часу. Цей оператор формує рівняння

$$\nabla z_t = z_t - z_{t-1}. \quad (5.10)$$

Підставляючи значення z_{t-1} із виразу (5.2) в (5.10), отримаємо:

$$\nabla z_t = z_t - z_{t-1} = z_t - Bz_t = (1 - B)z_t. \quad (5.11)$$

Із виразу (5.11) випливає, що

$$\nabla = 1 - B. \quad (5.12)$$

4. Різницевий оператор Δ із зсувом вперед на одну одиницю часу. Цей оператор формує рівняння

$$\Delta z_t = z_{t+1} - z_t. \quad (5.13)$$

Підставляючи значення z_{t+1} із виразу (5.6) в (5.13), отримаємо:

$$\Delta z_t = Fz_t - z_t = (F - 1)z_t. \quad (5.14)$$

Із виразу (5.14) випливає, що

$$\Delta = F - 1. \quad (5.15)$$

5. Оператор суми S . Цей оператор формує рівняння

$$Sz_t = z_t + z_{t-1} + z_{t-2} + \dots = \sum_{j=0}^{\infty} z_{t-j}. \quad (5.16)$$

Підставляючи значення z_{t-m} із виразу (5.5) в (5.16), отримаємо:

$$Sz_t = \left(1 + B + B^2 + \dots\right) z_t = \frac{1}{1 - B} z_t = (1 - B)^{-1} z_t. \quad (5.17)$$

Примітка. Під час виведення співвідношення (5.17) було використано формулу суми членів нескінченно спадної геометричної прогресії зі знаменником B , який за умови розгляду його як числа та умови збіжності ряду (5.16) повинен бути меншим одиниці.

Із співвідношень (5.12) та (5.17) випливає, що

$$S z_t = \nabla^{-1} z_t, \quad (5.18)$$

або

$$S = \nabla^{-1}. \quad (5.19)$$

Отже оператор суми є оберненим різницевому оператору із зсувом назад.

5.2 Синтез моделі стаціонарного часового ряду на основі лінійного фільтра

Послідовність некорельованих і розподілених нормально випадкових імпульсів a_t з нульовим середнім значенням і дисперсією

$$\sigma_a^2 = \text{const} \quad (5.20)$$

називають дискретним *білим шумом*.

Спробуємо використати імпульси білого шуму a_t для побудови моделі часового ряду z_t у такий спосіб

$$z_t = \mu + a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \dots, \quad (5.21)$$

де μ – рівень відліку (середнє значення) часового ряду z_t , а ψ_k , $k=1, 2, \dots$ – коефіцієнти ваги імпульсів білого шуму, з якими вони входять до суми (5.21).

Здійснимо операцію центрування часового ряду z_t , віднявши від z_t середнє значення μ . Для центрованого часового ряду

$$\tilde{z}_t = z_t - \mu. \quad (5.22)$$

Із (5.21) з врахуванням (5.22) отримаємо:

$$\tilde{z}_t = a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \dots. \quad (5.23)$$

Використовуючи для імпульсів a_{t-m} співвідношення (5.5), із (5.23) маємо:

$$\tilde{z}_t = a_t + \psi_1 B a_t + \psi_2 B^2 a_t + \dots = (1 + \psi_1 B + \psi_2 B^2 + \dots) \cdot a_t. \quad (5.24)$$

Позначимо

$$\Psi(B) = 1 + \psi_1 B + \psi_2 B^2 + \dots. \quad (5.25)$$

З урахуванням (5.25) співвідношення (5.24) можна записати і так:

$$\tilde{z}_t = \Psi(B) a_t. \quad (5.26)$$

Функцію $\Psi(B)$ називають передаточною функцією фільтра, який перетворює послідовність імпульсів білого шуму a_t у часовий ряд з заданими властивостями (рис. 5.2).

Коефіцієнти фільтра ψ_k , $k = 1, 2, \dots$ підбираються у процедурі мінімізації критерію (5.1) при $l = 0$.

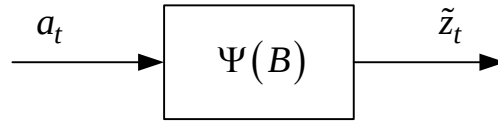


Рисунок 5.2 – Структурна схема лінійного фільтра

5.3 Модель стаціонарного часового ряду на основі авторегресії

З філософської точки зору модель регресії – це модель «з оглядкою назад, у бік, звідки прийшов»; тобто це модель, яка встановлює значення якоїсь координати процесу у даний момент часу за її незалежними складовими, визначеними у попередні моменти. Кількість врахованих складових визначає порядок регресії.

Виходячи з даного трактування, модель авторегресії – це модель, яка встановлює значення якоїсь координати процесу у даний момент часу на основі своїх попередніх значень. Кількість врахованих попередніх значень визначає порядок авторегресії.

Для центрованого часового ряду $\tilde{z}_t$ модель авторегресії порядку P (скорочено: $AR(p)$) можна записати у вигляді:

$$\tilde{z}_t = \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-p} + a_t, \quad (5.27)$$

де a_t – імпульс білого шуму, означення якому дано вище.

З урахуванням співвідношення (5.5) вираз (5.27) можна переписати і так:

$$\tilde{z}_t - \phi_1 B \tilde{z}_t - \phi_2 B^2 \tilde{z}_t - \dots - \phi_p B^p \tilde{z}_t = a_t,$$

або

$$\Phi(B) \tilde{z}_t = a_t, \quad (5.28)$$

де $\Phi(B)$ – оператор авторегресії порядку P :

$$\Phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p. \quad (5.29)$$

А далі скористаємось тотожністю

$$\Phi(B) \Phi^{-1}(B) = \Phi^{-1}(B) \Phi(B) = I, \quad (5.30)$$

в якій I – одиничний оператор, а оператор $\Phi^{-1}(B)$ є оберненим до оператора $\Phi(B)$.

Помноживши на $\Phi^{-1}(B)$ зліва рівняння (5.28), отримаємо:

$$\Phi^{-1}(B)\Phi(B)\tilde{z}_t = \Phi^{-1}(B)a_t, \quad (5.31)$$

або (з урахуванням тотожності (5.30)):

$$I\tilde{z}_t = \Phi^{-1}(B)a_t. \quad (5.32)$$

Оскільки перемноження на одиничний оператор результату не змінює, то рівняння (5.32) можна записати і так:

$$\tilde{z}_t = \Phi^{-1}(B)a_t. \quad (5.33)$$

Порівнюючи вираз (5.33) з (5.26), можна стверджувати, що

$$\Psi(B) = \Phi^{-1}(B). \quad (5.34)$$

Отже, визначивши за реалізацією досліджуваного часового ряду Z_t оператор авторегресії $\Phi(B)$, що зробити, як буде показано далі, нескладно, та синтезувавши оператор $\Phi^{-1}(B)$, обернений до $\Phi(B)$, що теж досить просто, ми одночасно визначаємо і передаточну функцію $\Psi(B)$ лінійного фільтра, який з білого шуму a_t формує потрібний часовий ряд Z_t .

Звертаємо увагу на те, що у цьому випадку $\Psi(B)$ знаходиться без мінімізації критерію (5.1) при $l = 0$, про що йшла мова в кінці попереднього підрозділу.

Під час розв'язання задачі ідентифікації моделі часового ряду Z_t на основі авторегресії порядку p необхідно визначити $p + 2$ невідомих, якими є коефіцієнти $\phi_1, \phi_2, \dots, \phi_p$ оператора $\Phi(B)$, середнє значення μ процесу Z_t та дисперсія σ_a^2 білого шуму a_t .

Про те, як розв'язати цю задачу, мова буде йти у підрозділах 5.8, 5.9.

5.4 Модель стаціонарного часового ряду на основі ковзного середнього

В моделі лінійного фільтра (5.23) значення часового ряду Z_t визначаються через зважену суму поточного та попередніх імпульсів білого шуму a_t .

Характерною особливістю передаточної функції $\Psi(B)$ (5.25) фільтра є те, що теоретично вона має нескінченну кількість членів, що створює певні незручності у разі використання цієї моделі.

Тому цікавою виявилась пропозиція будувати модель часового ряду Z_t на основі скінченної множини потужністю q зважених імпульсів білого шуму a_t у вигляді

$$\tilde{z}_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}. \quad (5.35)$$

Оскільки у співвідношенні (5.35) використовується q попередніх значень білого шуму a_{t-i} , $i = \overline{1, q}$, які зважено віднімаються від поточного імпульсу a_t , то це співвідношення фактично задає «ковзне середнє» яке «зміщується» вздовж послідовності a_t з ростом t , утримуючи в собі при «зміщенні» одну і ту ж кількість членів.

Застосовуючи до імпульсів a_{t-m} ідеологію співвідношення (5.5), із виразу (5.35) отримуємо:

$$\tilde{z}_t = a_t - \theta_1 B a_t - \theta_2 B^2 a_t - \dots - \theta_q B^q a_t, \quad (5.36)$$

або

$$\tilde{z}_t = \Theta(B) a_t, \quad (5.37)$$

де $\Theta(B)$ – оператор ковзного середнього

$$\Theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \quad (5.38)$$

порядку q .

Співвідношення (5.37), (5.38) задають модель стаціонарного часового ряду Z_t у вигляді моделі ковзного середнього порядку q (скорочено: модель КС(q)).

При розв'язанні задачі ідентифікації моделі часового ряду Z_t на основі ковзного середнього порядку q необхідно визначити $q + 2$ невідомих, якими є коефіцієнти $\theta_1, \theta_2, \dots, \theta_q$ оператора $\Theta(B)$, середнє значення μ процесу Z_t та дисперсія σ_a^2 білого шуму a_t .

5.5 Комбінована модель стаціонарного часового ряду на основі авторегресії – ковзного середнього

В моделі часового ряду Z_t на основі авторегресії порядку p у формуванні поточного значення ряду бере участь лише один поточний імпульс білого шуму a_t . Природно припустити, що якщо від цього імпульсу a_t відняти зважену суму q попередніх значень білого шуму, то отримаємо модель, яка буде враховувати більш «тонкі» моменти випадкового процесу і більш адекватно віддзеркалюватиме його властивості, оскільки крім авторегресії ця модель буде враховувати ще й ковзне середнє процесу.

Така модель часового ряду Z_t носить назву модель авторегресії – ковзного середнього порядку (p, q) (скорочено: модель АРКС (p, q)) і має вигляд:

$$\begin{aligned} \tilde{z}_t = & \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-p} + \\ & + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}. \end{aligned} \quad (5.39)$$

Перенісши всі члени з $\tilde{z}_{t-i}$, $i = \overline{1, p}$ у ліву частину рівняння (5.39) та виконавши вже відомі перетворення, отримаємо рівняння

$$\Phi(B) \tilde{z}_t = \Theta(B) a_t, \quad (5.40)$$

в якому оператори $\Phi(B)$ та $\Theta(B)$ визначаються формулами (5.29), (5.38).

Рівняння (5.40) і є основною формою моделі часового ряду Z_t на основі АРКС (p, q) .

Для ідентифікації цієї моделі необхідно визначити $p + q + 2$ невідомих, якими є коефіцієнти ϕ_i , $i = \overline{1, p}$ оператора $\Phi(B)$, коефіцієнти θ_j , $j = \overline{1, q}$ оператора $\Theta(B)$, середнє значення μ процесу Z_t та дисперсія σ_a^2 білого шуму a_t .

5.6 Модель нестационарного часового ряду на основі авторегресії – проінтегрованого ковзного середнього

Всі моделі часових рядів, що побудовані у попередніх підрозділах, базувались на умові стаціонарності цих рядів. Але у повсякденному житті постійно стикаємося і з нестационарними випадковими процесами. Наприклад, процес пуску потужного електропривода, який працює на навантаження випадкового характеру (екскавация, конвеєрна доставка кускової руди з шахти, перемелювання рудних матеріалів у дробарках та млинах тощо), має досить суттєві проміжки нестационарності.

Покажемо, що такі нестационарні випадкові процеси, які при їх дискретизації перетворюються на часові ряди, можна досить адекватно описувати за допомогою моделі авторегресії – проінтегрованого ковзного середнього.

Припустимо, що у моделі АРКС (p, q) , поданій виразом (5.40), оператор $\Phi(B)$ має d кратних коренів, що дорівнюють одиниці. У цьому випадку, згідно з теоремою Вієта, оператор $\Phi(B)$ можна записати у вигляді

$$\Phi(B) = (1 - B)^d \cdot (1 - \phi_1^* B - \phi_2^* B^2 - \dots - \phi_l^* B^l), \quad (5.41)$$

де

$$d + l = p. \quad (5.42)$$

Позначимо

$$\Phi^*(B) = 1 - \phi_1^* B - \phi_2^* B^2 - \dots - \phi_l^* B^l. \quad (5.43)$$

З урахуванням (5.41) та (5.43) рівняння (5.40) можна переписати так:

$$\Phi^*(B) \cdot (1-B)^d \tilde{z}_t = \Theta(B) a_t. \quad (5.44)$$

Оскільки згідно зі співвідношенням (5.12)

$$(1-B)^d = \nabla^d, \quad (5.45)$$

тобто $(1-B)^d$ є різницеvim оператором із зсувом назад порядку d , то рівняння

$$(1-B)^d z_t = \nabla^d z_t \quad (5.46)$$

задає нову змінну w_t , яка пов'язана з z_t співвідношенням

$$w_t = \nabla^d z_t. \quad (5.47)$$

Підставляючи (5.47) у рівняння (5.44), отримуємо:

$$\Phi^*(B) w_t = \Theta(B) a_t. \quad (5.48)$$

Очевидно, що рівняння (5.48) задає модель АРКС(l, q) відносно w_t , яке можна переписати і так:

$$\begin{aligned} w_t = & \phi_1^* w_{t-1} + \phi_2^* w_{t-2} + \dots + \phi_l^* w_{t-l} + \\ & + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}. \end{aligned} \quad (5.49)$$

Рівняння (5.47), (5.48) задають модель нестационарного часового ряду z_t у вигляді авторегресії – проінтегрованого ковзного середнього порядку (l, q, d). Скорочено: модель АРПКС(l, q, d).

Звертаємо увагу на те, що вже перша різниця ∇z_t значень будь-якого нестационарного часового ряду z_t має менший ступінь нестационарності, ніж сам часовий ряд z_t . Ще менший ступінь нестационарності матиме друга різниця $\nabla^2 z_t$, яка є різницею перших різниць $\nabla(\nabla z_t)$ цього часового ряду z_t .

Підвищуючи порядок d різниці $\nabla^d z_t$, рано чи пізно дійдемо до такого її значення w_t , яке вже являтиме собою стаціонарний часовий ряд відносно w_t . На рис. 5.3 наведена графічна інтерпретація цього факту.

Зрозуміло, що з моделі АРПКС(l, q, d) (5.47), (5.48) при $d = 0$ отримаємо модель АРКС(p, q).

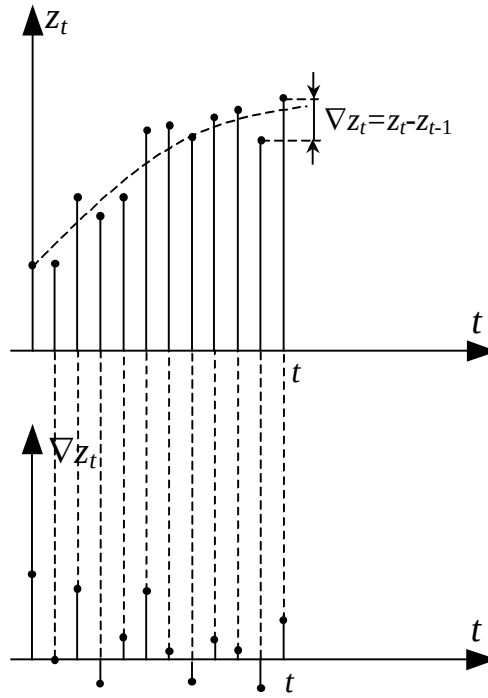


Рисунок 5.3 – Графічна інтерпретація перетворення нестационарного часового ряду Z_t у стаціонарний часовий ряд для його різниці ∇Z_t

Дамо пояснення чому у назві моделі АРПКС(l, q, d) має місце слово «проінтегроване» стосовно ковзного середнього.

Нагадаємо, що оберненим оператором для ∇ згідно з виразом (5.19) є оператор суми S , який є синонімом оператора інтегрування. Тому, отримавши w_t із рівняння (5.48), для переходу до часового ряду Z_t необхідно координату w_t проінтегрувати d разів, оскільки, домножуючи рівняння (5.47) зліва на ∇^{-d} , маємо:

$$\nabla^{-d} w_t = \nabla^{-d} \nabla^d z_t, \quad (5.50)$$

або

$$\nabla^{-d} w_t = I \cdot z_t, \quad (5.51)$$

звідки, з урахуванням виразу (5.19), матимемо:

$$z_t = \nabla^{-d} w_t = S^d w_t. \quad (5.52)$$

Зрозуміло, що найскладнішим завданням під час використання моделі АРПКС(l, q, d) є визначення чисельного значення параметра інтегрування d , або, в інших термінах, визначення кількості різниць, які треба послідовно взяти від нестационарного часового ряду Z_t , щоб перетворити його у стаціонарний ряд відносно якоїсь різниці цього ряду.

5.7 Автоковаріація та автокореляція часового ряду

Автоковаріацією γ_k часового ряду Z_t з затримкою k називають вираз

$$\gamma_k = \text{COV}\{Z_t, Z_{t+k}\} = E\{(Z_t - \mu) \cdot (Z_{t+k} - \mu)\}, \quad (5.53)$$

в якому E – символ обчислення математичного очікування від виразу, що стоїть у фігурних дужках.

Зрозуміло, що

$$\gamma_0 = E\{(Z_t - \mu)^2\} = \sigma_z^2 \quad (5.54)$$

– дисперсія часового ряду Z_t .

Для отримання статистичної оцінки γ_k^* автоковаріації γ_k використовують вираз

$$\gamma_k^* = \frac{1}{N-k} \sum_{t=1}^{N-k} (Z_t - \mu) \cdot (Z_{t+k} - \mu). \quad (5.55)$$

Автоковаріація γ_k характеризує ступінь лінійного зв'язку між значеннями часового ряду Z_t та Z_{t+k} .

Зрозуміло, що:

$$\begin{cases} |\gamma_k| \leq \gamma_0, \\ \gamma_k = \gamma_{-k}. \end{cases} \quad (5.56)$$

Автокореляцією ρ_k часового ряду Z_t із затримкою k називають вираз

$$\rho_k = \frac{\gamma_k}{\gamma_0} = \frac{E\{(Z_t - \mu) \cdot (Z_{t+k} - \mu)\}}{E\{(Z_t - \mu)^2\}}. \quad (5.57)$$

Для довільної автокореляції ρ_k справедливими є такі співвідношення:

$$\begin{cases} |\rho_k| \leq \rho_0, \\ \rho_0 = 1, \\ \rho_k = \rho_{-k}. \end{cases} \quad (5.58)$$

Вся можлива сукупність γ_k є автоковаріаційною функцією часового ряду Z_t . Вона належить до класу решітчастих функцій.

Аналогічно сукупність всіх значень ρ_k задає автокореляційну функцію часового ряду Z_t .

Приклад графіка автокореляційної функції ρ_k наведено на рис. 5.4.

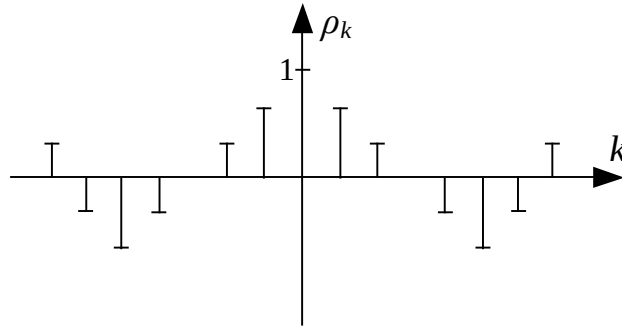


Рисунок 5.4 – Один із можливих графіків автокореляційної функції ρ_k , $k = -N, N$

5.8 Рівняння Юла-Уокера

Рівняння Юла-Уокера дають відповідь на запитання: «А як же визначити коефіцієнти авторегресії у моделі часового ряду у формі $AR(p)$?»

Для їх синтезу спочатку помножимо вираз (5.27) на $\tilde{z}_{t-k}$. У результаті цього отримаємо:

$$\tilde{z}_{t-k}\tilde{z}_t = \phi_1\tilde{z}_{t-k}\tilde{z}_{t-1} + \phi_2\tilde{z}_{t-k}\tilde{z}_{t-2} + \dots + \phi_p\tilde{z}_{t-k}\tilde{z}_{t-p} + \tilde{z}_{t-k}a_t. \quad (5.59)$$

Зробимо у рівнянні (5.59) заміну дискретної змінної, поклавши

$$t - k = \lambda. \quad (5.60)$$

Отримаємо:

$$\tilde{z}_\lambda\tilde{z}_{\lambda+k} = \phi_1\tilde{z}_\lambda\tilde{z}_{\lambda+k-1} + \phi_2\tilde{z}_\lambda\tilde{z}_{\lambda+k-2} + \dots + \phi_p\tilde{z}_\lambda\tilde{z}_{\lambda+k-p} + \tilde{z}_\lambda a_{\lambda+k}. \quad (5.61)$$

Знайдемо математичне очікування E від обох частин рівняння (5.61). Отримаємо:

$$\begin{aligned} E\{\tilde{z}_\lambda\tilde{z}_{\lambda+k}\} &= \phi_1 E\{\tilde{z}_\lambda\tilde{z}_{\lambda+k-1}\} + \phi_2 E\{\tilde{z}_\lambda\tilde{z}_{\lambda+k-2}\} + \dots + \\ &+ \phi_p E\{\tilde{z}_\lambda\tilde{z}_{\lambda+k-p}\} + E\{\tilde{z}_\lambda a_{\lambda+k}\}. \end{aligned} \quad (5.62)$$

Використовуючи вираз (5.53), рівнянню (5.62) надаємо вигляд:

$$\gamma_k = \phi_1\gamma_{k-1} + \phi_2\gamma_{k-2} + \dots + \phi_p\gamma_{k-p} \quad (5.63)$$

для всіх k від 1 до P .

Але при $k = 0$ з урахуванням властивості (5.56) із виразу (5.62) отримаємо ще одне рівняння:

$$\gamma_0 = \phi_1\gamma_1 + \phi_2\gamma_2 + \dots + \phi_p\gamma_p + \sigma_a^2. \quad (5.64)$$

Відсутність результатів обчислення математичного очікування $E\{\tilde{z}_\lambda \cdot a_{\lambda+k}\}$ у рівняннях (5.63) та їх присутність у рівнянні (5.64) у вигляді дисперсії білого шуму σ_a^2 пояснюється тим, що згідно з властивостями білого

шуму кожний його імпульс корелюється (взаємо пов'язується) лише сам із собою і зовсім не корелюється ні з яким іншим, навіть розміщеним у часі поряд. Тож

$$E\{\tilde{z}_\lambda \cdot a_{\lambda+k}\} = \begin{cases} 0, & npu \quad k \neq 0 \\ \sigma_a^2, & npu \quad k = 0 \end{cases}. \quad (5.65)$$

З рівняння (5.64), з урахуванням символіки (5.54), маємо:

$$\sigma_a^2 = \sigma_z^2 - \phi_1 \gamma_1 - \phi_2 \gamma_2 - \dots - \phi_p \gamma_p. \quad (5.66)$$

Тож, якщо для реалізації стаціонарного часового ряду z_t довжиною N вже обчислені оціночні середнє значення μ і дисперсія σ_z^2 за відомими формулами

$$\mu = \frac{1}{N} \sum_{t=1}^N z_t, \quad (5.67)$$

$$\sigma_z^2 = \frac{1}{N-1} \sum_{t=1}^N (z_t - \mu)^2, \quad (5.68)$$

уже знайдені коефіцієнти $\phi_1, \phi_2, \dots, \phi_p$ і обчислені за формулою (5.55) автоковаріації $\gamma_1, \gamma_2, \dots, \gamma_p$, визначити дисперсію білого шуму σ_a^2 за формулою (5.66) зовсім нескладно.

Перш ніж викласти алгоритм отримання числових значень коефіцієнтів $\phi_1, \phi_2, \dots, \phi_p$, розділимо всі рівняння системи (5.63) на γ_0 . У результаті цього отримаємо систему рівнянь вже не відносно автоковаріацій $\gamma_k, k = \overline{1, p}$, а відносно автокореляцій $\rho_k, k = \overline{1, p}$, тобто систему рівнянь:

$$\left\{ \begin{array}{l} \phi_1\rho_0 + \phi_2\rho_1 + \phi_3\rho_2 + \dots + \phi_p\rho_{p-1} = \rho_1, \\ \phi_1\rho_1 + \phi_2\rho_0 + \phi_3\rho_1 + \dots + \phi_p\rho_{p-2} = \rho_2, \\ \phi_1\rho_2 + \phi_2\rho_1 + \phi_3\rho_0 + \dots + \phi_p\rho_{p-3} = \rho_3, \\ . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \\ \phi_1\rho_{p-1} + \phi_2\rho_{p-2} + \phi_3\rho_{p-3} + \dots + \phi_p\rho_0 = \rho_p, \end{array} \right. \tag{5.69}$$

або у матричній формі -

$$M \cdot \Phi = \rho, \quad (5.70)$$

де матриці $M(p \times p)$, $\Phi(p \times 1)$, $\rho(p \times 1)$ мають вигляд:

$$M = \begin{bmatrix} 1 & \rho_1 & \rho_2 & \cdots & \rho_{p-1} \\ \rho_1 & 1 & \rho_1 & \cdots & \rho_{p-2} \\ \rho_2 & \rho_1 & 1 & \cdots & \rho_{p-3} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \rho_{p-1} & \rho_{p-2} & \rho_{p-3} & \cdots & 1 \end{bmatrix},$$

$$\Phi = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_p \end{bmatrix}, \quad \rho = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \vdots \\ \rho_p \end{bmatrix}.$$

У рівняннях (5.70) невідомими є значення коефіцієнтів $\phi_1, \phi_2, \dots, \phi_p$.

Для розв'язання системи рівнянь (5.70) спочатку визначаємо матрицю M^{-1} , яка є оберненою матриці M . Потім множимо матричне рівняння (5.70) зліва на M^{-1} . У результаті цього отримуємо:

$$M^{-1}M \cdot \Phi = M^{-1}\rho, \quad (5.71)$$

або

$$I \cdot \Phi = M^{-1}\rho, \quad (5.72)$$

і остаточно -

$$\Phi = M^{-1}\rho. \quad (5.73)$$

Рівняння (5.69), (5.70) носять назву *рівнянь Юла-Уокера*. Їх розв'язок у вигляді (5.73) дозволяє за попередньо обрахованими автокореляціями $\rho_k, k = \overline{1, p}$ часового ряду z_t визначити вектор $\Phi(p \times 1)$ коефіцієнтів $\phi_1, \phi_2, \dots, \phi_p$ оператора авторегресії у моделі $AR(p)$.

5.9 Узагальнений метод ідентифікації моделі авторегресії-ковзного середнього

В моделі часового ряду z_t на основі авторегресії порядку p , тобто в моделі $AR(p)$, у формуванні поточного значення ряду бере участь лише один поточний імпульс білого шуму a_t . Природно припустити, що якщо від цього імпульсу a_t відняти зважену суму q попередніх значень білого шуму, то отримаємо модель, яка буде враховувати більш «тонкі» моменти випадкового процесу і більш адекватно віддзеркалюватиме його властивості, оскільки крім авторегресії ця модель буде враховувати ще й ковзне середнє процесу.

Така модель часового ряду z_t носить назву моделі авторегресії – ковзного середнього порядку (p, q) (скорочено: модель АРКС (p, q)) і має вигляд (5.39), який для зручності використання повторимо під номером, поточним для цього підрозділу -

$$\begin{aligned} \tilde{z}_t = & \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \dots + \phi_p \tilde{z}_{t-p} + \\ & + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}. \end{aligned} \quad (5.74)$$

Перенісши всі члени з $\tilde{z}_{t-i}$, $i = \overline{1, p}$ у ліву частину рівняння (5.74) та виконавши вже відомі перетворення, отримаємо рівняння (5.40), яке для зручності використання теж повторимо під номером, поточним для цього підрозділу -

$$\Phi(B) \tilde{z}_t = \Theta(B) a_t, \quad (5.75)$$

в якому оператори $\Phi(B)$ та $\Theta(B)$ визначаються виразами (5.29), (5.38), які теж для зручності повторимо під поточними номерами цього підрозділу:

$$\Phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \quad (5.76)$$

$$\Theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \quad (5.77)$$

Рівняння (5.74)-(5.77) і задають модель стаціонарного центрованого часового ряду z_t на основі авторегресії-ковзного середнього, тобто модель у формі АРКС (p, q) .

Для ідентифікації цієї моделі потрібно визначити $p + q + 2$ невідомих, якими є коефіцієнти ϕ_i , $i = \overline{1, p}$ оператора $\Phi(B)$, коефіцієнти θ_j , $j = \overline{1, q}$ оператора $\Theta(B)$, середнє значення μ процесу z_t та дисперсія σ_a^2 білого шуму a_t .

А далі сконцентруємо увагу на наших роботах, приведених в кінці списку використаної літератури, в яких класична методологія синтезу та ідентифікації математичних моделей дискретних лінійних стохастичних динамічних об'єктів, що викладена в роботах професорів Box George E. P., Jenkins Gwilym M., доповнена нами ще одним методом, в якому класичний метод Юла-Уокера використовується лише як один із проміжних етапів реалізації методу, запропонованого нами, який ми назвали узагальненим методом ідентифікації.

Нагадаємо, що, як показано вище в підрозділі 5.8, ідентифікувати модель дискретного лінійного стохастичного динамічного об'єкта у вигляді авторегресії порядку p , тобто, модель АР (p) , яка має вигляд (5.27), де z_t – центроване значення часового ряду в момент часу t , a_t – імпульс білого шуму з дисперсією σ_a^2 , згенерований в той же момент часу t , можна, застосовуючи відомий метод Юла-Уокера, приведений вище у підрозділі 5.8, згідно з яким

числові значення коефіцієнтів регресії ϕ_i , $i = 1, 2, \dots, p$ знаходяться з матричного рівняння (5.73), в якому матриця M^{-1} є оберненою до матриці M , а матриці ϕ, M, ρ мають вигляд :

[illegible]

де $\rho_i, i = 1, 2, \dots, p$ – це автокореляції центрованого часового ряду z_t , які розраховуються за виразом (5.57), в якому $\gamma_i, i = 0, 1, 2, \dots, p$ – автоковаріації, що розраховуються для центрованого часового ряду z_t за виразом (5.55) та використовуються, окрім розв’язання матричного рівняння (5.70), і для обчислення дисперсії σ_a^2 білого шуму a_t за виразом (5.66), в який також підставляються результати (5.73) розв’язання матричного рівняння (5.70).

Ще раз підкреслюємо, що усі вирази, за допомогою яких розв'язуються рівняння Юла-Уокера, задають лише вихідні передумови для запропонованого нами узагальненого методу ідентифікації моделі авторегресії-ковзного середнього $АРКС(p,q)$ з довільними значеннями порядків p, q , структура якої для центрованого часового ряду z_t має вигляд (5.74).

Одразу ж відзначимо, що при виведенні розрахункових співвідношень запропонованого нами методу ідентифікації моделі авторегресії-ковзного середнього $APKC(p,q)$, як і в алгоритмі Юла-Уокера, теж використана умова відсутності кореляції між сусідніми імпульсами білого шуму, що має вигляд (5.65).

На першому етапі синтезу узагальненого методу ідентифікації моделі АРКС(p, q) та його алгоритму ми конкретизували значення параметрів на такому рівні, на якому уже можна буде робити узагальнення, але які ще не приводять до занадто громіздких математичних виразів. Тобто, якби ми задали $p = 1$, $q = 1$, або $p = 2$, $q = 2$, то отриманої інформації для узагальнення було б недостатньо, а тому ми задали $p = 3$ і $q = 3$. Для цього значення параметрів вираз (5.74) у вигляді моделі АРКС(3,3) набуває вигляду, в якому ми використали для зручності програмування при створенні Python-програми латинські позначення вагових коефіцієнтів, тобто набуває вигляду

$$z_t = g_1 z_{t-1} + g_2 z_{t-2} + g_3 z_{t-3} + a_t - p_1 a_{t-1} - p_2 a_{t-2} - p_3 a_{t-3}. \quad (5.79)$$

Помножуючи обидві частини рівняння (5.79) по черзі на $z_t, z_{t-1}, z_{t-2}, z_{t-3}, z_{t-4}, z_{t-5}, z_{t-6}$, усереднюючи отримані добутки за виразом для автоковаріації (5.53) і враховуючи умови (5.65), отримаємо систему рівнянь, яка має вигляд:

$$\begin{cases} \{q_0 = g_1 q_1 + g_2 q_2 + g_3 q_3 + \sigma_a^2 - p_1(g_1 - p_1)\sigma_a^2 - p_2(g_2 - p_2)\sigma_a^2 - p_3(g_3 - p_3)\sigma_a^2\}, \\ \{q_1 = g_1 q_0 + g_2 q_1 + g_3 q_2 - p_1 \sigma_a^2 - p_2(g_1 - p_1)\sigma_a^2 - p_3(g_2 - p_2)\sigma_a^2\} \\ \{q_2 = g_1 q_1 + g_2 q_0 + g_3 q_1 - p_2 \sigma_a^2 - p_3(g_1 - p_1)\sigma_a^2\}, \\ \{q_3 = g_1 q_2 + g_2 q_1 + g_3 q_0 - p_3 \sigma_a^2\}, \end{cases} \quad (5.80)$$

$$\begin{cases} q_4 = g_1 q_3 + g_2 q_2 + g_3 q_1, \\ q_5 = g_1 q_4 + g_2 q_3 + g_3 q_2, \\ q_6 = g_1 q_5 + g_2 q_4 + g_3 q_3, \end{cases} \quad (5.81)$$

Аналізуючи отриману систему рівнянь (5.80), (5.81), бачимо, що з підсистеми (5.81) цієї системи рівнянь, застосовуючи до даної підсистеми методику Юла-Уокера, можна знайти усі три параметри авторегресії, використовуючи у цьому випадку матриці методу Юла-Уокера у вигляді:

$$g = \begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix}, \quad M = \begin{bmatrix} q_3 \cdot q_2 \cdot q_1 \\ q_4 \cdot q_3 \cdot q_2 \\ q_5 \cdot q_4 \cdot q_3 \end{bmatrix}, \quad q = \begin{bmatrix} q_4 \\ q_5 \\ q_6 \end{bmatrix} \quad (5.82)$$

та матричний розв'язок (5.73) у вигляді

$$g = M^{-1}q, \quad (5.83)$$

у якому M^{-1} матриця, обернена до матриці M , заданої виразом (5.82).

А далі для кожного рівняння підсистеми (5.80) сконструюємо числові значення алгебраїчної суми складових, які позначимо як А, В, С, Д, використовуючи вирази:

$$\{q_0 - g_1 q_1 - g_2 q_2 - g_3 q_3 = A\}, \quad (5.84)$$

$$\{q_1 - g_1 q_0 - g_2 q_1 - g_3 q_2 = B\}, \quad (5.85)$$

$$\{q_2 - g_1 q_1 - g_2 q_0 - g_3 q_1 = C\}, \quad (5.86)$$

$$\{q_3 - g_1 q_2 - g_2 q_1 - g_3 q_0 = D\}, \quad (5.87)$$

за допомогою яких та за допомогою співвідношень:

$$\sigma_a^2 = -\frac{D}{p_3}, \quad (5.88)$$

$$p_3 = \frac{Dp_2}{C - D(g_1 - p_1)} \quad (5.89)$$

цю підсистему (5.80) трансформуємо до вигляду:

$$\begin{cases} f_1(p_1, p_2) = 0, \\ f_2(p_1, p_2) = 0, \end{cases} \quad (5.90)$$

де

$$f_1(p_1, p_2) = Ap_2(C - D(g_1 - p_1)) - (C - D(g_1 - p_1))^2(-1 + p_1(g_1 - p_1) + p_2(g_2 - p_2)) - Dp_2(g_3(C - D(g_1 - p_1)) - Dp_2),$$

$$f_2(p_1, p_2) = Bp_2 - (C - D(g_1 - p_1))(p_1 + p_2(g_1 - p_1)) - Dp_2(g_2 - p_2).$$

Розв'язуючи систему рівнянь (5.90), знаходимо параметри p_1, p_2 , а зі співвідношень (5.89) та (5.88) знаходимо параметри p_3, σ_a^2 .

Але, якщо в процесі визначення параметрів p_1, p_2, p_3 виявиться, що $p_3 = 0$, дисперсію білого шуму потрібно визначати з першого рівняння системи (5.80) з урахуванням виразу (5.84), тобто використовуючи вираз

$$\sigma_a^2 = \frac{A}{1 - p_1(g_1 - p_1) - p_2(g_2 - p_2) - p_3(g_3 - p_3)}. \quad (5.91)$$

А завершуючи викладення синтезованого нами узагальненого методу ідентифікації, акцентуємо увагу на тому, що, на відміну від алгоритму ідентифікації моделі АРКС(p,q) за методикою, викладеною в роботах професорів Box George E. P., Jenkins Gwilym M, де і параметри авторегресії, і параметри ковзного середнього пропонується визначати, використовуючи одні і ті ж значення автоковаріацій та застосовуючи процедуру мінімізації суми квадратів відхилень для пошуку оцінок параметрів ковзного середнього при визначених попередньо на тих же значеннях автоковаріацій за методикою Юла – Уокера параметрах авторегресії, у запропонованому нами узагальненому методі, по-перше, параметри авторегресії розраховуються з використанням одного набору автоковаріацій, а параметри ковзного середнього розраховуються з використанням іншого набору автоковаріацій, що відповідає кібернетичному принципу використання «свіжих точок» при розширенні набору параметрів, оцінки яких знаходяться, а по-друге, для визначення параметрів ковзного середнього застосовується пряма процедура, яка не вимагає поновлення процедури мінімізації суми квадратів відхилень при переході до інших значень порядків авторегресії та ковзного середнього. Ще однією перевагою узагальненого методу є те, що в разі, якщо часовий ряд є таким, що для врахування його стохастичного характеру в моделі АРКС(p,q) достатньо використати лише один імпульс білого шуму, то в результаті реалізації запропонованого алгоритму інші вагові коефіцієнти при попередніх значеннях імпульсів виявляться або рівними нулю, або близькими до нуля.

5.10 Приклади ідентифікації дискретної ЛДС ЗП на основі моделей часових рядів

Розглянемо процес $z(t)$ формування струму якоря електродвигуна системи електропривода конвеєра, на який з бункера надходить вугілля кусками різного об'єму та форми для транспортування його до дробарки.

У зв'язку з тим, що кількість вугілля, яке надходить з бункера на конвеєр у кожний момент часу, є величиною випадковою, то і момент навантаження на валу електродвигуна теж буде величиною випадковою, відповідною масі вугілля, яке в даний момент часу знаходиться на конвеєрі. У свою чергу, такий же випадковий характер буде мати і обертальний момент електродвигуна, який

врівноважує момент навантаження, і струм якоря електродвигуна, який створює обертальний момент.

Якщо вимірювати струм якоря електродвигуна лише через рівні відрізки часу t , то матимемо реалізацію часового ряду z_t , наведену у табл. 5.1.

Таблиця 5.1 – Реалізація часового ряду z_t струму якоря електродвигуна системи електропривода конвеєра

t	1	2	3	4	5	6	7	8	9	10
z_t	20	50	40	20	30	50	10	40	50	20

Побудуємо математичні моделі цього часового ряду z_t на основі авторегресії 1-го, 2-го та 3-го порядків, що мають вигляд:

$$\tilde{z}_t^{(1)} = \phi_1 \tilde{z}_{t-1} + a_t, \quad (5.92)$$

$$\tilde{z}_t^{(2)} = \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + a_t, \quad (5.93)$$

$$\tilde{z}_t^{(3)} = \phi_1 \tilde{z}_{t-1} + \phi_2 \tilde{z}_{t-2} + \phi_3 \tilde{z}_{t-3} + a_t. \quad (5.94)$$

Почнемо розв'язання задачі ідентифікації часового ряду z_t зі знаходження оцінок його середнього значення μ , дисперсії σ_a^2 та автоковаріацій $\gamma_1, \gamma_2, \gamma_3$.

$$\mu = \frac{1}{10} \sum_{t=1}^{10} z_t = \frac{1}{10} (20 + 50 + 40 + 20 + 30 + 50 + 10 + 40 + 50 + 20) = 33. \quad (5.95)$$

$$\sigma_z^2 = \frac{1}{10} \sum_{t=1}^{10} (z_t - \mu)^2 = \frac{1}{10} \left[(-13)^2 + (17)^2 + (7)^2 + (-13)^2 + (-3)^2 + (17)^2 + (-23)^2 + (7)^2 + (17)^2 + (-13)^2 \right] = 201. \quad (5.96)$$

$$\begin{aligned} \gamma_1 = \frac{1}{9} \sum_{t=1}^9 (z_t - \mu)(z_{t+1} - \mu) = \frac{1}{9} & \left[(-13) \cdot 17 + 17 \cdot 7 + 7 \cdot (-13) + (-13) \cdot (-3) + (-3) \cdot 17 + 17 \cdot (-23) + \right. \\ & \left. + (-23) \cdot 7 + 7 \cdot 17 + 17 \cdot (-13) \right] = -95,4. \end{aligned} \quad (5.97)$$

$$\begin{aligned} \gamma_2 = \frac{1}{8} \sum_{t=1}^8 (z_t - \mu)(z_{t+2} - \mu) = \frac{1}{8} & \left[(-13) \cdot 7 + 17 \cdot (-13) + 7 \cdot (-3) + (-13) \cdot 17 + (-3) \cdot (-23) + 17 \cdot 7 + \right. \\ & \left. + (-23) \cdot 17 + 7 \cdot (-13) \right] = -106. \end{aligned} \quad (5.98)$$

$$\gamma_3 = \frac{1}{7} \sum_{t=1}^7 (z_t - \mu)(z_{t+3} - \mu) = \frac{1}{7} [(-13) \cdot (-13) + 17 \cdot (-3) + 7 \cdot 17 + (-13) \cdot (-23) + (-3) \cdot 7 + 17 \cdot 17 + (-23) \cdot (-13)] = -157,6. \quad (5.99)$$

Підраховуємо автокореляції ρ_1, ρ_2, ρ_3 .

$$\rho_1 = \frac{\gamma_1}{\gamma_0} = \frac{\gamma_1}{\sigma_z^2} = -\frac{95,4}{201} = -0,475. \quad (5.100)$$

$$\rho_2 = \frac{\gamma_2}{\gamma_0} = \frac{\gamma_2}{\sigma_z^2} = -\frac{106}{201} = -0,527. \quad (5.101)$$

$$\rho_3 = \frac{\gamma_3}{\gamma_0} = \frac{\gamma_3}{\sigma_z^2} = -\frac{157,6}{201} = -0,784. \quad (5.102)$$

Запишемо рівняння Юла-Уокера для моделі $AP(1)$ (5.92):

$$\rho_1 = \phi_1 \rho_0 = \phi_1. \quad (5.103)$$

Підставляючи значення ρ_1 із виразу (5.100) в рівняння (5.103), отримаємо:

$$\phi_1 = -0,475. \quad (5.104)$$

Тож модель $AP(1)$ для часового ряду z_t , заданого табл. 5.1, буде мати вигляд:

$$\tilde{z}_t^{(1)} = -0,475 \tilde{z}_{t-1} + a_t, \quad (5.105)$$

де a_t – імпульс білого шуму з дисперсією (згідно з виразом (5.66)):

$$\sigma_a^2 = \sigma_z^2 - \phi_1 \gamma_1 = 201 - (-0,475) \cdot (-95,4) = 155,69. \quad (5.106)$$

Отже, якщо потрібно за допомогою моделі (5.105) спрогнозувати одинадцяте значення часового ряду z_t , заданого табл. 5.1, то ми повинні його десяте значення перемножити на $(-0,475)$ і до цього результату додати значення імпульсу білого шуму a_t , згенерованого комп'ютером за стандартною програмою, в яку треба лише задати діапазон генерації, який визначається, виходячи зі значення дисперсії цього шуму, визначеної за формулою (5.106), тобто, виходячи з того, що $\sigma_a^2 = 155,69$.

Тепер запишемо рівняння Юла-Уокера для моделі $AP(2)$ (5.93):

$$\begin{cases} \phi_1 \rho_0 + \phi_2 \rho_1 = \rho_1, \\ \phi_1 \rho_1 + \phi_2 \rho_0 = \rho_2, \end{cases} \quad (5.107)$$

або

$$\begin{cases} \phi_1 - 0,475\phi_2 = -0,475, \\ -0,475\phi_1 + \phi_2 = -0,527. \end{cases} \quad (5.108)$$

Розв'язуючи цю систему двох рівнянь з двома невідомими ϕ_1, ϕ_2 , отримаємо:

$$\begin{cases} \phi_1 = -1,050, \\ \phi_2 = -0,972. \end{cases} \quad (5.109)$$

Тож модель $AP(2)$ для часового ряду Z_t , заданого табл. 5.1, буде мати вигляд:

$$\tilde{z}_t^{(2)} = -1,050\tilde{z}_{t-1} - 0,972\tilde{z}_{t-2} + a_t, \quad (5.110)$$

де a_t – імпульс білого шуму з дисперсією (згідно з виразом (5.66)):

$$\begin{aligned} \sigma_a^2 = \sigma_z^2 - \phi_1\gamma_1 - \phi_2\gamma_2 = 201 - (-1,05) \cdot (-95,4) - \\ - (-0,972) \cdot (-106) = -2,2. \end{aligned} \quad (5.111)$$

Але дисперсія σ_a^2 ніколи не може мати від'ємний знак, отже, отримання результату (5.111) свідчить про неадекватність моделі $AP(2)$ для часового ряду Z_t , заданого табл. 5.1.

Тож зупиняємося на моделі (5.105).

Але, якби σ_a^2 було б більшим за нуль, тоді і друга модель теж мала б право на існування. І для того, щоб здійснити вибір, необхідно було б знайти суму квадратів відхилень, прогнозованих за допомогою моделі значень часового ряду від отриманих експериментально. Та модель, для якої ця сума квадратів відхилень виявиться найменшою, і буде оптимальною для часового ряду, який моделюється.

А завершимо ми цей підрозділ трьома програмами, складеними мовою Python, згідно з першою з яких моделюється стаціонарний часовий ряд на основі $AP(3)$ – це буде приклад № 17, згідно з другою – нестаціонарний часовий ряд на основі $ARPKC(3,0,2)$ – це буде приклад № 18, а згідно з третьою стаціонарний часовий ряд на основі $ARKC(3,3)$ – це буде приклад № 19.

Програма мовою Python для розв'язання задачі, пов'язаної з ідентифікацією та моделюванням стаціонарного часового ряду $y_t = \{5, 8, 3, 4, 6, 3, 2, 7, 5, 4, 3, 6, 4, 5, 3, 8, 6, 4, 3, 5, 4, 2, 7, 4, 5, 3, 6, 3, 4, 5\}$, заданого в N точках $t \in [0, N - 1]$ при $N=30$, з використанням моделі $AR(3)$ зі структурою:

$$y_t = b + m_t,$$

$$b = \frac{1}{N} \sum_{i=1}^N y_i,$$

$$m_t = g_1 m_{t-1} + g_2 m_{t-2} + g_3 m_{t-3} + a_t$$

та алгоритму Юла-Уокера визначення її параметрів

In [1]: import numpy as np

2., 7., 5., 4., 3., 6., 4., 5., 3., \

In [2]: L=[5., 8., 3., 4., 6., 3., \

8., 6., 4., 3., 5., 4., 2., \


```

7.,4.,5.,3.,6.,3.,4.,5.]
In [3]: N=30
In [4]: def fun (x):
        return np.sum(x)
In [5]: fun(L)
Out[5]: 137.0
In [6]: b= _/N;b
Out[6]:
4.5666666666666666
In [7]: b=b.round(3);b
Out[7]: 4.567
In [8]: L1=L-b
In [9]: def fun (x):
        return np.dot(x,x)
In [10]: fun(L1)
Out[10]:
77.36667
In [11]: q0= _/N; q0
Out[11]:
2.5788889999999998
In [12]: q0=q0.round(3);\
        q0
Out[12]:
2.579
In [13]: L2=L1[:-1]
In [14]: L5=L1[1:]
In [15]: def fun (x,y):
        return np.dot(x,y)
In [16]: fun(L2, L5)
Out[16]:
-22.820818999999993
In [17]: q1= _/(N-1); q1
Out[17]:
-0.786924793103448
In [18]: q1=q1.round(3);q1
Out[18]:
-0.787
In [19]: L3=L2[:-1]
In [20]: L6=L5[1:]
In [21]: fun(L3, L6)
Out[21]:
-14.874308000000001
In [22]: q2= _/(N-2);q2
Out[22]:
-0.5312252857142857
In [23]: q2=q2.round(3);q2
Out[23]:
-0.531

```

```

In [24]: L4=L3[:-1]
In [25]: L7=L6[1:]
In [26]: fun(L4,L7)
Out[26]:
2.67020300000000008
In [27]: q3= _/(N-3); q3
Out[27]:
0.09889640740740743
In [28]: q3=q3.round(3);q3
Out[28]:
0.099
In [29]: r0=q0/q0;r0
Out[29]:
1.0
In [30]: r1=q1/q0;r1
Out[30]:
-0.30515703761147733
In [31]: r1=r1.round(3);r1
Out[31]:
-0.305
In [32]: r2=q2/q0;r2
Out[32]:
-0.2058937572702598
In [33]: r2=r2.round(3);r2
Out[33]:
-0.206
In [34]: r3=q3/q0;r3
Out[34]:
0.038386971694455214
In [35]: r3=r3.round(3);r3
Out[35]:
0.038
In [36]: L9=[r0,r1,r2,\
        r3];L9
Out[36]:
[1.0, -0.305, -0.206, 0.038]
In [37]: import sympy
In [38]: from sympy\
        import*
In [39]: r,r0,r1,r2,r3 =\
        symbols('r r0 r1
        r2 r3')
In [40]: M = symbols('M')
In [41]: M = Matrix ([[r0,r1,
        r2],[r1,r0,r1],
        [r2,r1,r0]]);M
Out[41]:
Matrix([

```

```

[r0, r1, r2],
[r1, r0, r1],
[r2, r1, r0]])
In [42]: g,g1,g2,g3=\
symbols('g g1
g2 g3')
In [43]: g = Matrix(
[g1,g2,g3]);g
Out[43]:
Matrix([
[g1],
[g2],
[g3]])
In [44]: M =M.subs(
[(r0,1),(r1,-0.305),
(r2,-0.206)];M
Out[44]:
Matrix([
[ 1, -0.305, -0.206],
[-0.305, 1, -0.305],
[-0.206, -0.305, 1]])
In [45]: r=Matrix([r1,r2,r3]);\
r
Out[45]:
Matrix([
[r1],
[r2],
[r3]])
In [46]:r=r.subs([(r1,-0.305),
(r2,-0.206),
(r3,0.038)]); r
Out[46]:
Matrix([
[-0.305],
[-0.206],
[ 0.038]])
In [47]: B=simplify(M.inv())
In [48]: g=B*r
In [49]: g=g.evalf(3); g
Out[49]:
Matrix([
[-0.465],
[-0.403],
[-0.181]])
In [50]: g1=g[0,0]; g1
Out[50]:
-0.465
In [51]: g2=g[1,0]; g2

```

```

Out[51]:
-0.403
In [52]: g3=g[2,0]; g3
Out[52]:
-0.181
In [53]: a = symbols ('a')
In [54]: ska = symbols \
('ska')
In [55]: ska = q0-g1*q1-\
g2*q2-g3*q3; ska
Out[55]:
2.01681854248047
In [56]: skv = symbols ('skv')
In [57]: skv = ska**(0.5); skv
Out[57]:
1.42014736646605
In [58]: skv = skv.evalf(3);\
skv
Out[58]:
1.42
In [59]: a11,a22 = \
symbols ('a11 a22')
In [60]: a11 = -3*skv; a11
Out[60]:
-4.26
In [61]: a22 = 3*skv; a22
Out[61]:
4.26
In [62]: import random as rnd
In [63]: m = symbols('m:10')
In [64]: l = symbols ('l:10')
In [65]: m=list(m)
In [66]: l=list(l)
In [67]: d = symbols ('d:10')
In [68]: d = list (d)
In [69]: d[0] = rnd.\
uniform (-4.26, 4.26);\
d[0]
Out[69]:
3.7878324717658174
In [70]: m[0]=g1*L1[29]+\
g2*L1[28]+g3*L1[27]\
+d[0]; m[0]
Out[70]:
4.09812879744941
In [71]: L1 = np.append(L1,
[m[0]])
In [72]: l[0]=b+m[0]; l[0]

```

```

Out[72]:
8.66512879744941
In [73]: L = np.append(L, [[0]])
In [74]: d[1] = rnd.\
          uniform (-4.26,
4.26); d[1]
Out[74]:
-2.587298897242724
In [75]: m[1]= g1*L1[30]+\
g2*L1[29]+g3*L1[28]+\
d[1]; m[1]
Out[75]:
-2.08074576204602
In [76]: L1 = np.append(L1,
[m[1]])
In [77]: l[1]=b+m[1]; l[1]
Out[77]:
2.48625423795398
In [78]: L=np.append(L,[l[1]])
In [79]: d[2] = rnd.uniform \
(-4.26,4.26); d[2]
Out[79]:
0.8071389548290568
In [80]: m[2]= g1*L1[31]+\
g2*L1[30]+g3*L1[29]\
+d[2]; m[2]
Out[80]:
0.0451337059883733
In [81]: L1 = np.append\
(L1,[m[2]])
In [82]: l[2]=b+m[2]; l[2]
Out[82]:
4.61213370598837
In [83]: L=np.append(L,[l[2]])
In [84]: d[3] = rnd.uniform\
(-4.26,4.26); d[3]
Out[84]:
3.5146713080878706
In [85]: m[3]= g1*L1[32]+\
g2*L1[31]+g3*L1[30]\
+d[3]; m[3]
Out[85]:
3.59161472387185
In [86]: L1 = np.\
append (L1,[m[3]])
In [87]: l[3]=b+m[3]; l[3]
Out[87]:
8.15861472387185

```

```

In [88]: L=np.append\
(L,[l[3]])
In [89]: d[4] = rnd.\
uniform (-4.26,4.26);
d[4]
Out[89]:
3.332153915638866
In [90]: m[4]= g1*L1[33]+\
g2*L1[32]+g3*L1[31]\
+d[4]; m[4]
Out[90]:
0.178801469969288
In [91]: L1 = np.append\
(L1,[m[4]])
In [92]: l[4]=b+m[4]; l[4]
Out[92]:
4.74580146996929
In [93]: L=np.append\
(L,[l[4]])
In [94]: d[5] = rnd.uniform\
(-4.26,4.26); d[5]
Out[94]:
2.867785147080328
In [95]: m[5]= g1*\
L1[34]+g2*L1[33]+\
g3*L1[32]+ d[5];\
m[5]
Out[95]:
1.32898394299138
In [96]: L1 = np.append\ (L1,\
[m[5]])
In [97]: l[5]=b+m[5]; l[5]
Out[97]:
5.89598394299138
In [98]: L=np.append(L,\
[l[5]])
In [99]: d[6] = rnd.uniform\
(-4.26,4.26); d[6]
Out[99]:
1.69270588124252
In [100]: m[6]= g1*\
L1[35]+g2*L1[34]+\
g3*L1[33]+ \
d[6]; m[6]
Out[100]:
-0.471995713797095
In [101]: L1 = np.append\
(L1,[m[6]])

```

```

In [102]: l[6]=b+m[6]; l[6]
Out[102]:
4.09500428620290
In [103]: L=np.append\
          (L,l[6])
In [104]: d[7] = rnd.uniform\
          (-4.26,4.26); d[7]
Out[104]:
0.10021632009046666
In [105]: m[7]= g1*L1[36]+\
          g2*L1[35]+g3*L1[34]\
          +d[7]; m[7]
Out[105]:
-0.248149939266434
In [106]: L1 = np.append\
          (L1,m[7])
In [107]: l[7]=b+m[7]; l[7]
Out[107]:
4.31885006073357
In[108]:L=np.append(L,l[7])
In [109]: d[8] = rnd.uniform\
          (-4.26,4.26); d[8]
Out[109]:
-2.232660689069062
In [110]: m[8]= g1*L1[37]+\
          g2*L1[36]+g3*L1[35]\
          +d[8]; m[8]
Out[110]:
-2.16719334714070

```

```

In [111]: L1 = np.append\
          (L1,m[8])
In [112]: l[8]=b+m[8]; l[8]
Out[112]:
2.39980665285930
In [113]: L=np.append\
          (L,l[8])
In [114]: d[9] = rnd.uniform\
          (-4.26,4.26); d[9]
Out[114]:
1.544009122634895
In [115]: m[9]= g1*L1[38]+\
          g2*L1[37]+g3*L1[36] \
          +d[9]; m[9]
Out[115]:
2.73738643318719
In [116]: L1 = np.append\
          (L1,m[9])
In [117]: l[9]=b+m[9]; l[9]
Out[117]:
7.30438643318719
In[118]:L=np.append(L,l[9])
In [119]: import matplotlib
In [120]: import matplotlib.\
          pyplot as plt
In [121]: plt.plot(L1)
Кінець програми.

```

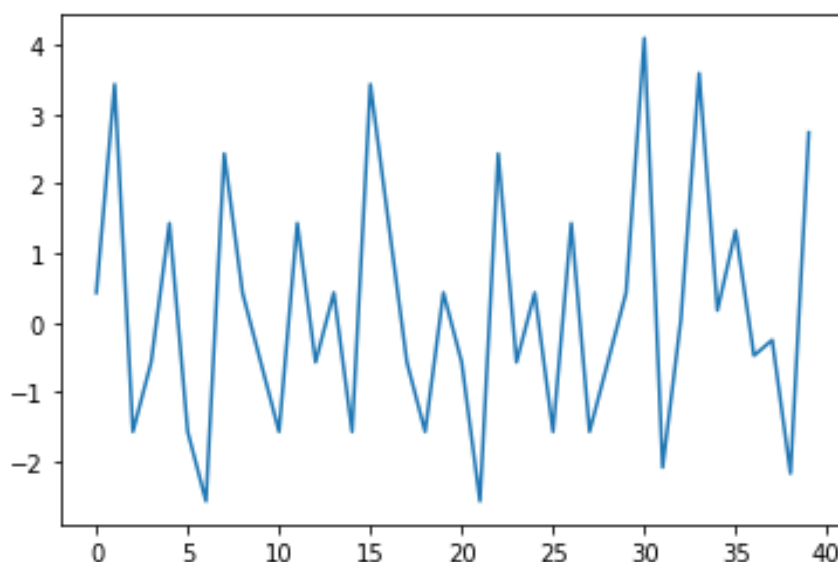


Рисунок 5.5 – Графік часового ряду m_t , який в діапазоні $t \in [0,30]$ заповнений експериментально-отриманими значеннями, а за межами цього діапазону заповнений прогностичними значеннями, отриманими з використанням авторегресійної моделі, ідентифікованої за експериментально-отриманими значеннями даною програмою.

Програма мовою Python для розв'язання задачі, пов'язаної з ідентифікацією та моделюванням нестационарного часового ряду $y_t = \{5., 8., 3., 4., 6., 8., 10., 7., 6., 9., 12., 8., 11., 15., 12., 10., 7., 8., 12., 15., 18., 20., 17., 14., 15., 17., 16., 19., 22., 25.\}$, заданого в N точках $t \in [0, N - 1]$ при N=30, з використанням моделі АРПКС(3,0,2):

$$v_t = y_t - y_{t-1}, w_t = v_t - v_{t-1},$$

$$w_t = g_1 w_{t-1} + g_2 w_{t-2} + g_3 w_{t-3} + a_t,$$

$$y_k = y_{-1} + \sum_{t=0}^k v_t, \quad v_k = v_{-1} + \sum_{t=0}^k w_t$$

та алгоритму Юла-Уокера визначення її параметрів

```
In [1]: import numpy as np
In [2]: L=[5.,8.,3.,4.,6.,8.,10.,
          7.,6.,9.,12.,8.,11.,15.,
          12.,10.,7.,8.,12.,15.,18.,
          20.,17.,14.,15.,17.,16.,
          19.,22.,25.]
In [3]: N=30
In [4]: L1 = np.diff(L)
In [5]: def fun (x):
          return np.sum(x)
In [6]: fun(L1)
Out[6]: 20.0
In [7]: _/(N-1)
Out[7]: 0.6896551724137931
In [8]: L11=np.diff(L1)
In [9]: fun(L11)
Out[9]: 0.0
In [10]: def fun (x):
          return np.dot(x,x)
In [11]: fun(L11)
Out[11]:
390.0
In [12]: q0=_/(N-2); q0
Out[12]:
13.928571428571429
In [13]: q0=q0.round(3);q0
Out[13]:
13.929
```

```

In [14]: L2=L11[: -1]
In [15]: L5=L11[1:]
In [16]: def fun (x,y):
            return np.dot(x,y)
In [17]: fun(L2, L5)
Out[17]:
-102.0
In [18]: q1=_/(N-3); q1
Out[18]:
-3.7777777777777777
In [19]: q1=q1.round(3);q1
Out[19]:
-3.778
In [20]: L3=L2[: -1]
In [21]: L6=L5[1:]
In [22]: fun(L3, L6)
Out[22]:
-134.0
In [23]: q2=_/(N-4);q2
Out[23]:
-5.153846153846154
In [24]: q2=q2.round(3);q2
Out[24]:
-5.154
In [25]: L4=L3[: -1]
In [26]: L7=L6[1:]
In [27]: fun(L4,L7)
Out[27]:
49.0
In [28]: q3=_/(N-5); q3
Out[28]:
1.96
In [29]: r0=q0/q0;r0
Out[29]:
1.0
In [30]: r1=q1/q0;r1
Out[30]:
-0.29059829059829057
In [31]: r1=r1.round(3);r1
Out[31]:
-0.291
In [32]: r2=q2/q0;r2
Out[32]:
-0.3964615384615385
In [33]: r2=r2.round(3);r2
Out[33]:
-0.396
In [34]: r3=q3/q0;r3
Out[34]:

```

```

0.15076923076923077
In [35]: r3=r3.round(3);r3
Out[35]:
0.151
In [36]: L9=[r0,r1,r2,r3];L9
Out[36]:
[1.0, -0.291, -0.396, 0.151]
In [37]: import sympy
In [38]: from sympy import*
In [39]: r,r0,r1,r2,r3 = \
            symbols('r r0 r1 r2 r3')
In [40]: M = symbols('M')
In [41]: M = Matrix([[r0,r1,r2],
                    [r1,r0,r1],[r2,r1,r0]]);M
Out[41]:
Matrix([
[r0, r1, r2],
[r1, r0, r1],
[r2, r1, r0]])
In [42]: g,g1,g2,g3=\
            symbols('g g1 g2 g3')
In [43]: g = Matrix([g1,g2,g3]);g
Out[43]:
Matrix([
[g1],
[g2],
[g3]])
In [44]: M =M.subs([(r0,1),(r1,
                    -0.291),(r2,-0.396)]);M
Out[44]:
Matrix([
[  1, -0.291, -0.396 ],
[-0.291,   1, -0.291],
[-0.396, -0.291,   1 ]])
In [45]: r=Matrix([r1,r2,r3]); r
Out[45]:
Matrix([
[r1],
[r2],
[r3]])
In [46]: r=r.subs([(r1,-0.291),
                    (r2,-0.396),(r3,0.151)]); r
Out[46]:
Matrix([
[-0.291],
[-0.396],
[ 0.151]])
In [47]: B=simplify(M.inv())

```

```

In [48]: g=B*r
In [49]: g=g.evalf(3); g
Out[49]:
Matrix([
[-0.584 ],
[-0.644],
[-0.268]])
In [50]: g1=g[0,0]; g1
Out[50]:
-0.584
In [51]: g2=g[1,0]; g2
Out[51]:
-0.644
In [52]: g3=g[2,0]; g3
Out[52]:
-0.268
In [53]: a = symbols('a')
In [54]: ska = symbols('ska')
In [55]: ska = q0-g1*q1-g2*q2-\
          g3*q3; ska
Out[55]:
7.99764599609375
In [56]: skv = symbols('skv')
In [57]: skv = ska**(0.5)
In [58]: skv = skv.evalf(3); skv
Out[58]:
2.83
In [59]: a11,a22 = \
          symbols('a11 a22')
In [60]: a11 = -2*skv; a11
Out[60]:
-5.66
In [61]: a22 = 2*skv; a22
Out[61]:
5.66
In [62]: import random as rnd
In [63]: w = symbols('w:10')
In [64]: v = symbols('v:10')
In [65]: w=list(w)
In [66]: v=list(v)
In [67]: d = symbols('d:10')
In [68]: d = list(d)
In [69]: y = symbols('y:10')
In [70]: y=list(y)
In [71]: L21=[]
In [72]: L22=[]
In [73]: L23=[]
In [74]: d[0] = rnd.uniform \

```

```

          (-5.66,5.66); d[0]
Out[74]:
-3.949689744213309
In [75]: w[0]=g1*L11[27]+g2*\
          L11[26]+g3*L11[25]+\
          d[0]; w[0]
Out[75]:
-5.02122294733831
In [76]: L23.append(w[0]);L23
Out[76]:
[-5.02122294733831]
In [77]: v[0]=L1[28]+w[0];v[0]
Out[77]:
-2.02122294733831
In [78]: L22.append(v[0]);L22
Out[78]:
[-2.02122294733831]
In [79]: y[0]=L[29]+v[0];y[0]
Out[79]:
22.9787770526617
In [80]: L21.append(y[0]); L21
Out[80]:
[22.9787770526617]
In [81]: d[1] = rnd.uniform \
          (-5.66,5.66); d[1]
Out[81]:
-0.35599763798745787
In [82]: w[1]=g1*L23[0]+\
          g2*L11[27]+g3*L11[26]\
          + d[1]; w[1]
Out[82]:
2.57876987566682
In [83]: L23.append(w[1]);L23
Out[83]:
[-5.02122294733831,
2.57876987566682]
In [84]: v[1]=L22[0]+w[1];v[1]
Out[84]:
0.557546928328509
In [85]: L22.append(v[1]);L22
Out[85]:
[-2.02122294733831,
0.557546928328509]
In [86]: y[1]=L21[0]+v[1];y[1]
Out[86]:
23.5363239809902
In [87]: L21.append(y[1]);L21
Out[87]:

```

```

[22.9787770526617,
23.5363239809902]
In [88]: d[2] = rnd.uniform \
        (-5.66,5.66); d[2]
Out[88]:
5.4505096830373745
In [89]: w[2]=g1*L23[1]+\
        g2*L23[0]+g3*L11[27]+\
        d[2]; w[2]
Out[89]:
7.17717253770830
In [90]: L23.append(w[2]);L23
Out[90]:
[-5.02122294733831,
2.57876987566682,
7.17717253770830]
In [91]: v[2]=L22[1]+w[2];v[2]
Out[91]:
7.73471946603681
In [92]: L22.append(v[2]);L22
Out[92]:
[-2.02122294733831,
0.557546928328509,
7.73471946603681]
In [93]: y[2]=L21[1]+v[2];y[2]
Out[93]:
31.2710434470270
In [94]: L21.append(y[2]);L21
Out[94]:
[22.9787770526617,
23.5363239809902,
31.2710434470270]
In [95]: d[3] = rnd.uniform\
        (-5.66,5.66); d[3]
Out[95]:
1.5127310202924091
In [96]: w[3]=g1*L23[2]+\
        g2*L23[1]+g3*L23[0]+\

```

```

        d[3]; w[3]
Out[96]:
-2.99786690654250
In [97]: L23.append(w[3])
In [98]: v[3]=L22[2]+w[3];v[3]
Out[98]:
4.73685255949431
In [99]: L22.append(v[3])
In [100]: y[3]=L21[2]+v[3];y[3]
Out[100]:
36.0078960065213
In [101]: L21.append(y[3])
In [102]: d[4] = rnd.uniform \
        (-5.66,5.66); d[4]
Out[102]:
-2.938526432470273
In [103]: w[4]=g1*L23[3]+\
        g2*L23[2]+g3*L23[1]+\
        d[4]; w[4]
Out[103]:
-6.49957209318491
In [104]: L23.append(w[4])
In [105]: v[4]=L22[3]+w[4];v[4]
Out[105]:
-1.76271953369060
In [106]: L22.append(v[4])
In [107]: y[4]=L21[3]+v[4];y[4]
Out[107]:
34.2451764728307
In [108]: L21.append(y[4])
In [109]: L555 = L+L21
In [110]: import matplotlib
In [111]: import\
        matplotlib.pyplot\
        as plt
In [112]: plt.plot(L555)
Кінець програми

```

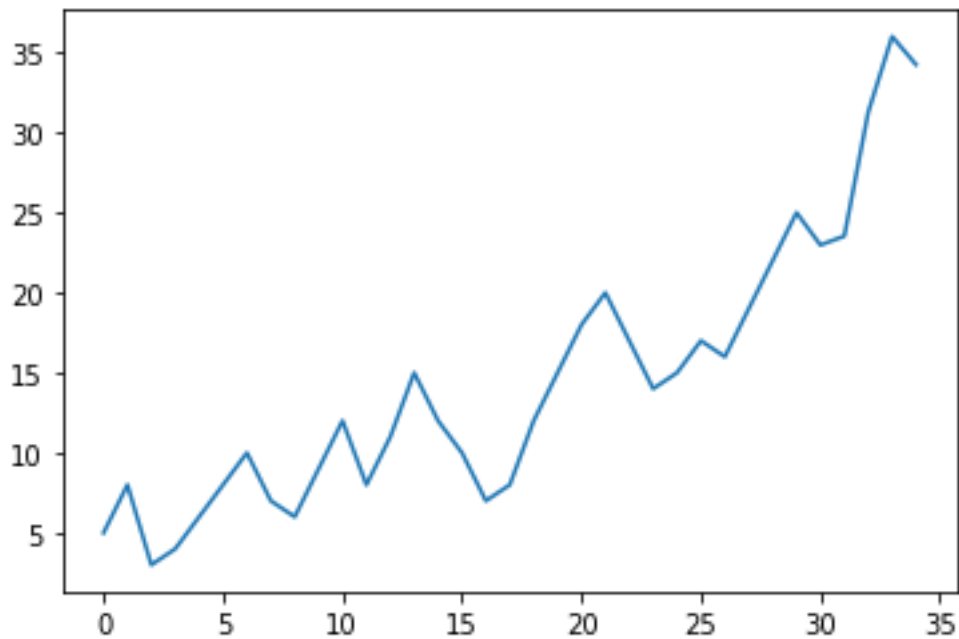



Рисунок 5.6 – Графік часового ряду y_t , який в діапазоні $t \in [0,30]$ заповнений експериментальними значеннями, а за його межами заповнений прогнозними значеннями, отриманими з використанням програми, приведеної вище.

Програма мовою Python для розв'язання задачі, пов'язаної з авторегресійним моделюванням стаціонарного часового ряду $y_t = \{5., 8., 3., 4., 6., 3., 2., 7., 5., 4., 3., 6., 4., 5., 3., 8., 6., 4., 3., 5., 4., 2., 7., 4., 5., 3., 6., 3., 4., 5.\}$, заданого в N точках $t \in [0, N-1]$ при $N = 30$, з використанням моделі АРКС(3,3) зі структурою:

$$y_t = b + m_t,$$

$$b = \frac{1}{N} \sum_{i=1}^N y_i,$$

$$m_t = g_1 m_{t-1} + g_2 m_{t-2} + g_3 m_{t-3} + a_t - p_1 a_{t-1} - p_2 a_{t-2} - p_3 a_{t-3}$$

та узагальненого методу її ідентифікації

```
In [1]: import numpy as np
In [2]: L=[5.,8.,3.,4.,6.,3.,2.,7.,5.,
4.,3.,6.,4.,5.,3.,8.,6.,4.,3.,
5.,4.,2.,7.,4.,5.,3.,6.,3.,4.,5.]
In [3]: N=30
[4]: def fun (x):
    return np.sum(x)
```

```
In [5]: fun(L)
Out[5]: 137.0
In [6]: b=_/N;b
```

```
Out[6]: 4.566666666666666
In [7]: b=b.round(3);b
Out[7]: 4.567
In [8]: L1=L-b
In [9]: def fun (x):
    return np.dot(x,x)
```

```
In [10]: fun(L1)
Out[10]: 77.36667
In [11]: q0=_/N; q0
Out[11]:
```

```

2.5788889999999998
In [12]: q0=q0.round(3); q0
Out[12]: 2.579
In [13]: L2=L1[:-1]
In [14]: L5=L1[1:]
In [15]: def fun (x,y):
            return np.dot(x,y)

In [16]: fun(L2,L5)
Out[16]:
-22.820818999999993
In [17]: q1=_/(N-1); q1
Out[17]:
-0.786924793103448
In [18]: q1=q1.round(3);q1
Out[18]: -0.787
In [19]: L3=L2[:-1]
In [20]: L6=L5[1:]
In [21]: fun(L3,L6)
Out[21]:
-14.874308000000001
In [22]: q2=_/(N-2);q2
Out[22]:
-0.5312252857142857
In [23]: q2=q2.round(3);q2
Out[23]: -0.531
In [24]: L4=L3[:-1]
In [25]: L7=L6[1:]
In [26]: fun(L4,L7)
Out[26]:
2.6702030000000008
In [27]: q3=_/(N-3); q3
Out[27]:
0.09889640740740743
In [28]: q3=q3.round(3); q3
Out[28]: 0.099
In [29]: L41=L4[:-1]
In [30]: L71=L7[1:]
In [31]: fun(L41,L71)
Out[31]:
20.482714
In [32]: q4=_/(N-4); q4
Out[32]:
0.7877966923076923
In [33]: q4=q4.round(3); q4
Out[33]: 0.788
In [34]: L31=L41[:-1]
In [35]: L61=L71[1:]

```

```

In [36]: fun(L31,L61)
Out[36]:
-32.271775
In [37]: q5=_/(N-5); q5
Out[37]:
-1.2908709999999999
In [38]: q5=q5.round(3); q5
Out[38]: -1.291
In [39]: L21=L31[:-1]
In [40]: L51=L61[1:]
In [41]: fun(L21,L51)
Out[41]:
-3.5932640000000005
In [42]: q6=_/(N-6); q6
Out[42]:
-0.14971933333333354
In [43]: q6=q6.round(3); q6
Out[43]: -0.15
In [44]: L9=[q0,q1,q2,q3,q4,q5,
q6];L9
Out[44]:
[2.579, -0.787, -0.531, 0.099,
0.788, -1.291, -0.15]
In [45]: import sympy
In [46]: from sympy import*
In [47]: q,q0,q1,q2,q3,q4,q5,\
q6 = symbols('q q0 q1
q2 q3 q4 q5 q6')
In [48]: M = symbols('M')
In [49]: M = Matrix([[q3,q2,q1],
[q4,q3,q2],[q5,q4,q3]]);M
Out[49]:
Matrix([
[q3, q2, q1],
[q4, q3, q2],
[q5, q4, q3]])
In [50]: g,g1,g2,g3=symbols('g g1
g2 g3')
In [51]: g = Matrix([g1,g2,g3]);g
Out[51]:
Matrix([
[g1],
[g2],
[g3]])
In [52]: M =M.subs([(q1,L9[1]),
(q2,L9[2]),(q3,L9[3]),
(q4,L9[4]),(q5,L9[5])]);M
Out[52]:

```

```

Matrix([
[ 0.099, -0.531, -0.787],
[ 0.788, 0.099, -0.531],
[-1.291, 0.788, 0.099]])
In [53]: q=Matrix([q4,q5,q6]); q
Out[53]:
Matrix([
[q4],
[q5],
[q6]])
In [54]: q=q.subs([(q4,L9[4]),
(q5,L9[5]),
(q6,L9[6])]); q
Out[54]:
Matrix([
[ 0.788],
[-1.291],
[ -0.15]])
In [55]: M1=simplify(M.inv())
In [56]: g=M1*q; g
Out[56]:
[-1.16879067955301],
[-2.14256824489834],
[0.297323330070228]])
In [57]: g=g.evalf(3); g
Out[57]:
Matrix([
[-1.17],
[-2.14],
[0.297]])
In [58]: g1=g[0,0]; g1
Out[58]: -1.17
In [59]: g2=g[1,0]; g2
Out[59]: -2.14
In [60]: g3=g[2,0]; g3
Out[60]: 0.297
In [61]: A,B,C,D = symbols('A B
C D')
In [62]: p,p0,p1,p2,p3 = symbols('p
p0 p1 p2 p3')
In [63]: A=q0-g1*q1-g2*q2-g3*q3;A
Out[63]:
q0 + 1.17*q1 + 2.14*q2 - 0.297*q3
In [64]: A = A.subs([(q0,L9[0]),(q1,
L9[1]),(q2,L9[2]),(q3,
L9[3])]);A
Out[64]:
0.492090270996094

```

```

In [65]: A=A.evalf(3);A
Out[65]: 0.492
In [66]: B =q1-g1*q0-g2*q1-\
g3*q2;B
Out[66]:
1.17*q0 + 3.14*q1 - 0.297*q2
In [67]: B = B.subs([(q0,L9[0]),(q1,
L9[1]),(q2,L9[2])]);B
Out[67]:
0.698738830566406
In [68]: B=B.evalf(3);B
Out[68]: 0.699
In [69]: C = q2-g1*q1-g2*q0-\
g3*q1;C
Out[69]:
2.14*q0 + 0.871*q1 + q2
In [70]: C = C.subs([(q0,L9[0]),
(q1,L9[1]),(q2,L9[2])]);C
Out[70]:
4.30896606445313
In [71]: C=C.evalf(3);C
Out[71]: 4.31
In [72]: D = q3-g1*q2-g2*q1-\
g3*q0;D
Out[72]:
-0.297*q0 + 2.14*q1 + 1.17*q2 + q3
In [73]: D = D.subs([(q0,L9[0]),
(q1,L9[1]),(q2,L9[2]),
(q3,L9[3])]);D
Out[73]:
-2.97453179931641
In [74]: D=D.evalf(3);D
Out[74]: -2.97
In [75]: p,p1,p2,p3 = symbols('p p1
p2 p3')
In [76]: f1 = Function('f1')(p1,p2)
In [77]: f2 = Function('f2')(p1,p2)
In [78]: f1=-A*p2/(C-D*(g1-p1))
-1+p1*(g1-p1)+p2*(g2-
p2)+(D*p2/(C-D*(g1-
p1)))*(g3-D*p2/(C-
D*(g1-p1)));f1
Out[78]:
p1*(-p1 - 1.17) + p2*(-p2 - 2.14) +
0.492*p2/(2.97*p1 - 0.832) - 2.97*
p2*(2.97*p2/(0.832 - 2.97*p1) +
0.297)/(0.832 - 2.97*p1) - 1
In [79]: f2=expand(f1);f2

```

```

Out[79]:
(0.492*p2+(1+p1*(1.17+p1))+p2*
(2.14+p2))*(4.31-2.97*(1.17+p1))*
(4.31-2.97*(1.17+p1))+2.97*p2*
(0.297*(4.31-2.97*(1.17+p1))+
2.97*p2)
In [80]: f11=Function('f11')(p1,p2)
In [81]: f22=Function('f22')(p1,p2)
In [82]: f11=-B*p2/(C-D*(g1-p1))+
p1+p2*(g1-p1)+D*p2*(g2-
p2)/(C-D*(g1-p1));f11
Out[82]:
p1 + p2*(-p1 - 1.17) + 0.699*p2/
(2.97*p1 - 0.832) - 2.97*p2*(-p2 -
2.14)/(0.832 - 2.97*p1)
In [83]: f22=expand(f11);f22
Out[83]:
0.699*p2-(p1-p2*(1.17+p1))*(4.31-
2.97*(1.17+p1))-2.97*p2*(2.14+p2)
In [84]: L15= solve([(0.492*p2+(1+
p1*(1.17+p1)+ p2*(2.14+p2))*
(4.31-2.97*(1.17+p1))*(4.31-
2.97*(1.17+p1))+2.97*p2*
(0.297*(4.31-2.97*(1.17+p1))+
2.97*p2),0.699*p2-(p1-p2*
(1.17+p1))*(4.31-2.97*(1.17+
p1))-2.97*p2*(2.14+p2)],
p1,p2);L15
Out[84]:
[(0.281178451178451, 0.0)]
In [85]: p1=L15[0][0];p1
Out[85]:
0.280193236714976
In [86]: p1=p1.evalf(3);p1
Out[86]: 0.280
In [87]: p2=L15[0][1];p2
Out[87]: 0.0
In [88]: p3=D*p2/(C-D*(g1-p1));p3
Out[88]: 0.0
In [89]: ska,skv = symbols('ska skv')
In [90]: ska=A/(1-p1*(g1-p1));ska
Out[90]: 0.350
In [91]: skv=ska**(0.5);skv
Out[91]:
0.591587344301330
In [92]: skv=skv.evalf(3);skv
Out[92]: 0.592
In [93]: a11,a22 = symbols('a11 a22')

```

```

In [94]: a11=-2*skv;a11
Out[94]: -1.18
In [95]: a22=2*skv;a22
Out[95]: 1.18
In [96]: p=[p0,p1];p
Out[96]:
[1.0, 0.280]
In [97]: import random as rnd
In [98]: m = symbols('m:5');m
Out[98]:
(m0, m1, m2, m3, m4)
In [99]: l = symbols('l:5');l
Out[99]:
(l0, l1, l2, l3, l4)
In [100]: m=list(m);m
Out[100]:
[m0, m1, m2, m3, m4]
In [101]: l=list(l);l
Out[101]:
[l0, l1, l2, l3, l4]
In [102]: d = symbols('d:5'); d
Out[102]:
(d0, d1, d2, d3, d4)
In [103]: d = list (d); d
Out[103]:
[d0,d1,d2,d3,d4]
In [104]: d0 = rnd.uniform (-1.18,
1.18); d0
Out[104]:
-0.9786481619579281
In [105]: m0= g1*L1[29]+g2*\
L1[28]+g3*L1[27]+ d0; m0
Out[105]:
-0.735726592133709
In [106]: L1 = np.append(L1,[m0])
In [107]: l0=b+m0; l0
Out[107]:
3.83127340786629
In [108]: L = np.append(L,[l0])
In [109]: d1 = rnd.uniform (-1.18,
1.18); d1
Out[109]:
-0.9602434653342734
In [110]: m1= g1*L1[30]+g2*\
L1[29]+g3*L1[28]+ d1-\
p1*d0; m1
Out[110]:
-0.922476284076637

```

```

In [111]: L1 = np.append(L1,[m1])
In [112]: l1=b+m1; l1
Out[112]:
3.64452371592336
In [113]: L=np.append(L,[l1])
In [114]: d2 = rnd.uniform (-1.18,
1.18); d2
Out[114]:
0.1282150195429812
In [115]: m2= g1*L1[31]+g2*\
L1[30]+g3*L1[29]+ d2-\
p1*d1; m2
Out[115]:
3.18046983687152
In [116]: L1 = np.append(L1,[m2])
In [117]: l2=b+m2; l2
Out[117]:
7.74746983687152
In [118]: L=np.append(L,[l2])
In [119]: d3 = rnd.uniform(-1.18,
1.18); d3
Out[119]:
-1.133835327586167
In [120]: m3= g1*L1[32]+g2*\
L1[31]+g3*L1[30]+ d3-\
p1*d2; m3

```

```

Out[120]:
-3.12903725295206
In [121]: L1 = np.append(L1,[m3])
In [122]: l3=b+m3; l3
Out[122]:
1.43796274704794
In [123]: L=np.append(L,[l3])
In [124]: d4 = rnd.uniform(-1.18,
1.18); d4
Out[124]:
0.13261500744833254
In [125]: m4= g1*L1[32]+g2*\
L1[31]+g3*L1[30]+ d4-\
p1*d3; m4
Out[125]:
-1.50894475826974
In [126]: L1 = np.append(L1,[m4])
In [127]: l4=b+m4; l4
Out[128]:
3.05805524173026
In [129]: L=np.append(L,[l4])
In [130]: import matplotlib
In [131]: import matplotlib.pyplot \
as plt
In [132]: plt.plot(L1)

```

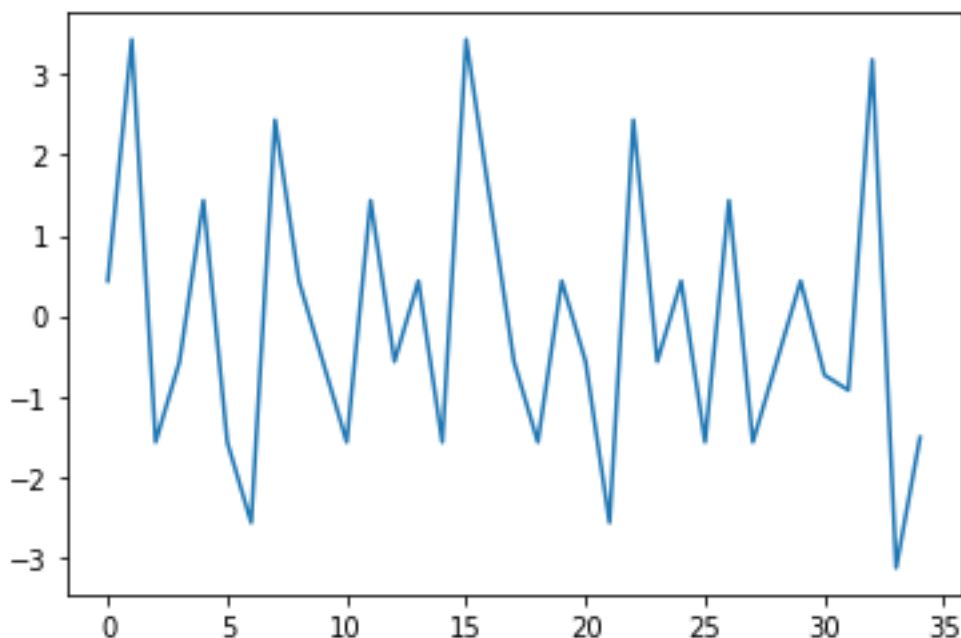


Рисунок 5.7 – Графік часового ряду m_t , який для $t \in [0,30]$ заповнений експериментальними значеннями, а за межами цього діапазону прогнозними

5.11 Завдання для самоперевірки

Що іменують «часовим рядом»? Наведіть приклади стаціонарного та нестаціонарного часового ряду.

Дайте означення операторів зсуву назад та вперед, різницевого оператору та оператору суми.

Що собою являє модель часового ряду у формі лінійного фільтра?

Що іменують «білим шумом»? Які його основні властивості?

Що розуміють під регресією та авторегресією?

Синтезуйте модель часового ряду у формі авторегресії.

Як пов'язані між собою передаточна функція лінійного фільтра та оператор авторегресії для часового ряду?

Що собою являє ковзне середнє часового ряду і модель у формі ковзного середнього?

Побудуйте модель часового ряду у формі авторегресії – ковзного середнього.

Яким чином можна нестаціонарний часовий ряд трансформувати у стаціонарний?

Побудуйте модель нестаціонарного часового ряду у формі авторегресії – проінтегрованого ковзного середнього.

Дайте означення автоковаріації і автокореляції часового ряду. Як знайти їх числові оцінки? Які основні їх властивості ви знаєте?

Для чого потрібні і як виводяться рівняння Юла-Уокера?

Як розв'язати рівняння Юла-Уокера?

Що собою являє узагальнений метод ідентифікації моделі $AR(p,q)$?

Частина 3

ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З РОЗПОДІЛЕНИМИ ПАРАМЕТРАМИ ТА НЕЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ

Розділ 6 МАТЕМАТИЧНІ МОДЕЛІ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З РОЗПОДІЛЕНИМИ ПАРАМЕТРАМИ

6.1 Загальна характеристика лінійних динамічних систем з розподіленими параметрами

В усіх попередніх розділах здійснювалась побудова математичних моделей динамічних систем за умови, що їх параметри є зосередженими в просторі.

Але існує досить багато фізичних процесів, що протікають в динамічних об'єктах, параметри яких не можна вважати зосередженими в одній точці простору.

Більше того, деякі об'єкти при моделюванні за одними ознаками мають зосереджені параметри, а за іншими переходять в клас об'єктів з розподіленими параметрами.

Нагадаємо, що будь-яку динамічну систему відносять до класу систем з розподіленими параметрами, якщо в її структурі є хоча б один елемент, реакція на вхідний сигнал у якому суттєво запізнюється у порівнянні з реакціями усіх інших елементів цієї системи.

У класі електронних систем – це системи, в структурі яких використовуються лінії затримки сигналу. У класі електричних систем – це системи з довгими лініями електропередачі. У класі електромеханічних систем – це системи з електроприводами, до структури яких входять троси, довгі штанги, конвеєри і трубопроводи. У класі систем автоматичного керування технологічними процесами – це системи з хімічними реакціями, з плавильними печами, з термоустановками для випалювання, сушки чи випарювання та з іншими об'єктами, процеси в яких через особливості динаміки чи протяжність у просторі протікають суттєво повільніше, ніж в контурах управління ними. У класі автоматизованих систем управління – це системи з комп'ютерною обробкою інформації за алгоритмами, що характеризуються суттєво більшими затратами часу на їх реалізацію у порівнянні з часом обробки інформації і прийняття рішень в інших контурах автоматизованої системи.

Які ж особливості є характерними саме для систем з розподіленими параметрами?

Спробуємо дати відповідь на це запитання в загальних рисах на конкретних прикладах.

В підйомних кранах вантажі піднімаються і опускаються з використанням тросів. Оскільки трос сплітається з великої кількості тонких сталевих дротів,

то він має властивість витягуватись чи скорочуватись в залежності від ваги вантажу, прикріпленого до одного з його кінців.

При підніманні вантажу трос спочатку витягується в «струну», але потім, навіть при постійній швидкості обертання валу електродвигуна, який намотує трос на катушку, завдяки взаємодії сил, обумовлених вагою вантажу і інерцією, в тросі виникають небажані повздовжні коливання, котрі розгойдують вантаж у вертикальній площині, заважаючи точно встановити його на нове місце і завдаючи цьому місцю та вантажу якогось рівня деформації.

Як ще один приклад електромеханічних систем з тросами розглянемо систему електропривода конуса, яким закривається кругле завантажувальне вікно доменної печі на її вершині. Через це вікно у доменну піч до початку плавлення завантажуються шихта. При завантаженні шихти металевий конус, призначенням якого є відкривання і закривання завантажувального вікна і який є підвішеним на тросі, знаходиться в своєму нижньому положенні, відкриваючи широку щілину між конусом і місцем його посадки у завантажувальному вікні та створюючи можливість шихті через цю щілину висипатись у доменну піч.

Після завантаження шихти за допомогою троса, інший кінець якого прикріплено до катушки, з'єднаної з валом електродвигуна, металевий конус підтягується, закриваючи завантажувальне вікно.

Висота деяких доменних печей сягає 70 метрів. Відстань від вертикальної осі доменної печі до місця встановлення електропривода конуса біля її підніжжя має аналогічний порядок. Тож довжина троса, яким передається рух від валу електродвигуна до конуса, може сягати 150 метрів.

За рахунок повздовжніх коливань троса при підтягуванні конуса він завдає таку кількість ударів по своєму ложу у завантажувальному вікні доменної печі і такої сили, що кожна доменна піч не менше одного разу на рік вимагає ремонтних робіт для відновлення щільного прилягання в системі «конус – вікно».

За допомогою електроприводів приводять у рух насоси перекачувальних станцій нафтопроводів і компресори перекачувальних станцій газопроводів, котрих в Україні багато тисяч кілометрів.

Основним параметром, яким регулюється продуктивність нафто- і газопроводів, є тиск в рідині чи газі, котрий створюється дією насосів та компресорів і який передається вздовж трубопроводу зі швидкістю звуку, яка залежить від складу рідини чи газу.

Через невисоку швидкість розповсюдження звукових хвиль, запізнення в їх появі на виході відрізка трубопроводу від однієї перекачувальної станції до іншої можуть сягати суттєвих значень, без врахування яких ефективної системи стабілізації тиску в трубопроводі побудувати не можна, оскільки сигнал з виходу об'єкта регулювання надходить по каналу зворотного зв'язку на регулятор, встановлений на вході об'єкта, зі значним запізненням і може слугувати причиною появи коливань тиску зі значною амплітудою.

Для ряду конвеєрних транспортних систем, які є складовими в більш складних системах, наприклад, в системах випалювання залізородних обкотишів, чи системах сушіння сипких матеріалів, важливим є дотримання однакової товщини шару матеріалу на конвеєрній стрічці. При нерівномірній подачі матеріалу на стрічку на вході конвеєра стабілізувати товщину шару матеріалу на цій стрічці можна лише змінюючи швидкість обертання приводних електродвигунів. І додаткові складнощі у побудові такої системи стабілізації створюються саме тим, що має місце запізнення появи вихідного сигналу об'єкта регулювання. І, звичайно ж, чим довшим є конвеєр, тим більшим буде запізнення появи сигналу на його виході.

В теорії автоматичного керування розроблено багато способів управління об'єктами з розподіленими параметрами, але кожен з них базується на тому, що відомою є математична модель об'єкта.

Тож, в наступних підрозділах цього розділу покажемо, як отримати математичну модель того чи іншого процесу, який протікає в об'єктах з розподіленими параметрами.

6.2 Рівняння математичної фізики як моделі елементів систем з розподіленими параметрами

Почнемо з побудови математичної моделі довгої лінії електропередачі ЛЕП (рис. 7.1), за допомогою якої електроенергія від генератора Г електростанції з електрорушійною силою E через підвищувальний трансформатор ПТ з приведеним повним опором Z_{Π} та знижувальний трансформатор Т з приведеним повним опором Z_T передається до навантаження Н з повним опором Z_H .

Об'єднаємо опір Z_T знижувального трансформатора Т з опором Z_H навантаження в сумарний опір Z_K , підключений до кінця лінії електропередачі.

Тоді функціональну схему (рис. 7.1) можна замінити принциповою схемою, наведеною на рис. 7.2, на якій довга лінія електропередачі ЛЕП подається як послідовний ланцюг Г-подібних чотириполіусників з параметрами R_j, L_j, G_j, C_j (на рис. 7.2 показано лише один з них: j -ий), де R_j – питомий активний опір j -ої ділянки фази ЛЕП, а L_j, G_j, C_j – відповідно, питома індуктивність, питома активна провідність і питома ємність цієї ділянки.

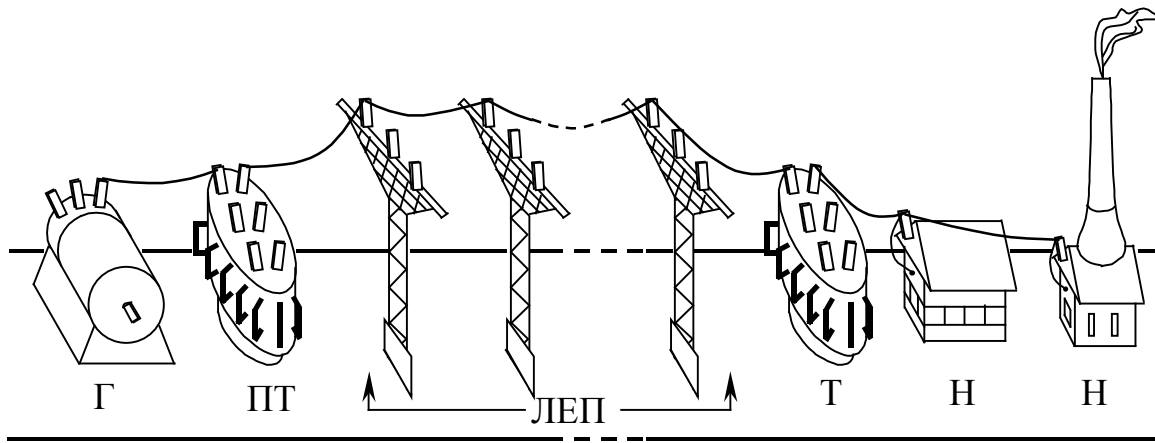


Рисунок 6.1 – Функціональна схема однієї фази електричної системи, до складу якої входить генератор Г електростанції, підвищувальний трансформатор ПТ, довга лінія електропередачі ЛЕП, знижувальний трансформатор Т та навантаження Н

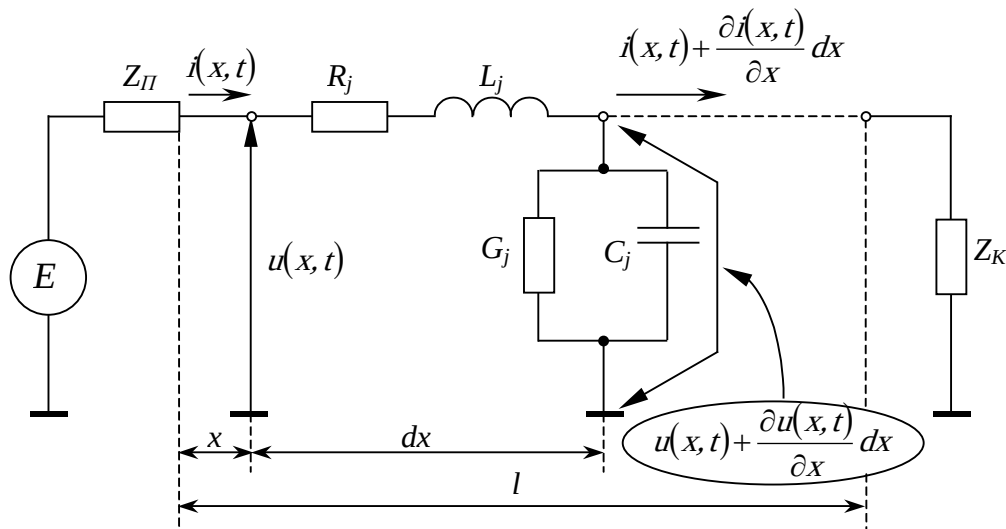


Рисунок 6.2 – Принципова схема однієї фази електричної системи

Позначимо $u(x, t)$, $i(x, t)$, відповідно, напругу і струм на вході j -го елемента схеми фази ЛЕП. Тоді на виході j -го елемента довжиною dx матимемо напругу і струм, що дорівнюють:

$$\begin{cases} u(x, t) + \Delta u = u(x, t) + \frac{\partial u(x, t)}{\partial x} dx, \\ i(x, t) + \Delta i = i(x, t) + \frac{\partial i(x, t)}{\partial x} dx. \end{cases} \quad (6.1)$$

Запишемо перший та другий закони Кірхгофа для j -го елемента схеми фази ЛЕП, опускаючи у першому законі диференціали 2-го порядку, які є нескінченно малими величинами 2-го порядку малості. Матимемо:

$$u(x, t) - \left[u(x, t) + \frac{\partial u(x, t)}{\partial x} dx \right] = (R_j dx) \cdot i(x, t) + (L_j dx) \cdot \frac{\partial i(x, t)}{\partial t}, \quad (6.2)$$

$$i(x, t) - \left[i(x, t) + \frac{\partial i(x, t)}{\partial x} dx \right] = (G_j dx) \cdot u(x, t) + (C_j dx) \cdot \frac{\partial u(x, t)}{\partial t}. \quad (6.3)$$

Розкриваючи квадратні дужки в лівій частині рівнянь (6.2), (6.3) і ділячи ці рівняння на dx , отримаємо:

$$-\frac{\partial u(x, t)}{\partial x} = R_j \cdot i(x, t) + L_j \frac{\partial i(x, t)}{\partial t}, \quad (6.4)$$

$$-\frac{\partial i(x, t)}{\partial x} = G_j \cdot u(x, t) + C_j \frac{\partial u(x, t)}{\partial t}. \quad (6.5)$$

Приймаючи умову, що параметри R_j , L_j , G_j , C_j є величинами сталими і однаковими для всіх ділянок схеми фази ЛЕП, рівняння (6.4), (6.5) можна записати так:

$$\begin{cases} -\frac{\partial u(x, t)}{\partial x} = R \cdot i(x, t) + L \frac{\partial i(x, t)}{\partial t}, \\ -\frac{\partial i(x, t)}{\partial x} = G \cdot u(x, t) + C \frac{\partial u(x, t)}{\partial t}. \end{cases} \quad (6.6)$$

Диференціальні рівняння (6.6) в частинних похідних і задають математичну модель довгої лінії електропередачі. Для отримання їх однозначного розв'язку необхідно задати початкові $i(x, 0)$, $u(x, 0)$ та граничні $i(0, 0)$, $u(0, 0)$ або $i(l, t_k)$, $u(l, t_k)$ умови.

Замість системи двох диференціальних рівнянь в частинних похідних 1-го порядку (6.6) відносно функцій $u(x, t)$, $i(x, t)$ як модель довгої ЛЕП можна використовувати і одне диференціальне рівняння в частинних похідних 2-го порядку відносно лише однієї функції $u(x, t)$:

$$\frac{\partial^2 u}{\partial x^2} - LC \frac{\partial^2 u}{\partial t^2} - (RC + GL) \frac{\partial u}{\partial t} - GRu = 0, \quad (6.7)$$

або лише однієї функції $i(x, t)$:

$$\frac{\partial^2 i}{\partial x^2} - LC \frac{\partial^2 i}{\partial t^2} - (RC + GL) \frac{\partial i}{\partial t} - GRi = 0, \quad (6.8)$$

які легко отримати із рівняння (6.6) шляхом додаткового диференціювання і підстановки одного рівняння в інше.

Покажемо це на прикладі отримання рівняння (6.8).

Перший крок: продиференціюємо за x друге рівняння системи (6.6), в результаті чого отримаємо:

$$-\frac{\partial^2 i(x, t)}{\partial x^2} = G \frac{\partial u(x, t)}{\partial x} + C \frac{\partial^2 u(x, t)}{\partial t \partial x}. \quad (6.9)$$

Другий крок: підставимо замість першого члена в правій частині рівняння (6.9) його значення із першого рівняння системи (6.6), в результаті чого отримаємо:

$$-\frac{\partial^2 i(x, t)}{\partial x^2} = G \left[-R \cdot i(x, t) - L \frac{\partial i(x, t)}{\partial t} \right] + C \frac{\partial^2 u(x, t)}{\partial t \partial x}. \quad (6.10)$$

Третій крок: продиференціюємо перше рівняння системи (6.6) за t , в результаті чого отримаємо:

$$-\frac{\partial^2 u(x, t)}{\partial x \partial t} = R \frac{\partial i(x, t)}{\partial t} + L \frac{\partial^2 i(x, t)}{\partial t^2}. \quad (6.11)$$

Четвертий крок: пам'ятаючи про те, що при незалежних змінних черговість диференціювання функції від них для отримання змішаної частинної похідної 2-го порядку ролі не грає, підставимо замість останнього члена в правій частині рівняння (6.10) його значення з рівняння (6.11), в результаті чого отримаємо:

$$-\frac{\partial^2 i(x, t)}{\partial x^2} = -GR \cdot i(x, t) - GL \frac{\partial i(x, t)}{\partial t} + C \left[-R \frac{\partial i(x, t)}{\partial t} - L \frac{\partial^2 i(x, t)}{\partial t^2} \right]. \quad (6.12)$$

П'ятий крок: переносимо усі члени рівняння (6.12) в одну сторону і зводимо подібні члени, в результаті чого отримаємо рівняння (6.8).

Аналогічним способом шляхом вилучення функції $i(x, t)$ та її похідних із системи (6.6) виводиться і рівняння (6.7).

Легко бачити, що обидва вони мають однакову структуру.

В теорії диференціальних рівнянь в частинних похідних рівняння (6.7), (6.8) відносять до рівнянь математичної фізики гіперболічного типу. З цих рівнянь шляхом прирівнювання до нуля окремих параметрів можна отримати математичні моделі трубопроводних ліній і тросів.

Очевидно, що в трубопроводах витік газу чи рідини через стінку відсутній, тому для них справедливою буде умова

$$G = 0. \quad (6.13)$$

З фізики відомо, що для гідравлічних і пневматичних систем аналогом електричної напруги $u(x, t)$ є тиск $p(x, t)$, а аналогом електричного струму $i(x, t)$ є об'ємна витрата $Q(x, t)$, тобто для трубопроводів є справедливим:

$$\begin{cases} u(x, t) = p(x, t), \\ i(x, t) = Q(x, t). \end{cases} \quad (6.14)$$

З врахуванням умов (6.13), (6.14) з рівнянь (6.7), (6.8) знайдемо, що математичні моделі гідравлічних або пневматичних трубопроводів мають вигляд:

$$\frac{\partial^2 p(x, t)}{\partial x^2} - LC \frac{\partial^2 p(x, t)}{\partial t^2} - RC \frac{\partial p(x, t)}{\partial t} = 0, \quad (6.15)$$

$$\frac{\partial^2 Q(x, t)}{\partial x^2} - LC \frac{\partial^2 Q(x, t)}{\partial t^2} - RC \frac{\partial Q(x, t)}{\partial t} = 0. \quad (6.16)$$

В рівняннях (6.15), (6.16):

$$R = \frac{8\pi\mu_D}{f^2}, \quad (6.17)$$

$$L = \frac{\rho}{f}, \quad (6.18)$$

$$C = \frac{f}{\rho a^2}, \quad (6.19)$$

де f – площа поперечного перерізу труби, μ_D – динамічний коефіцієнт в'язкості газу чи рідини, ρ – густина газу чи рідини, a – швидкість розповсюдження звуку в газі чи рідині.

Рівняння (6.15), (6.16) залишаються в класі гіперболічних рівнянь математичної фізики.

Якщо цікавить передача тепла вздовж довгого ізольованого стержня, то, враховуючи умови:

$$u(x, t) = \theta(x, t), \quad (6.20)$$

$$\begin{cases} G = 0, \\ L = 0, \end{cases} \quad (6.21)$$

для опису процесу зміни температури $\theta(x, t)$ в часі і вздовж стержня, з рівняння (6.7) отримаємо модель:

$$\frac{\partial^2 \theta(x, t)}{\partial x^2} - RC \frac{\partial \theta(x, t)}{\partial t} = 0, \quad (6.22)$$

яку називають *рівнянням теплопровідності* і відносять до класу параболічних рівнянь математичної фізики.

В рівнянні (6.22)

$$RC = \frac{\rho c}{\lambda}, \quad (6.23)$$

де ρ – густина матеріалу стержня, c – питома теплоємність цього матеріалу, а λ – коефіцієнт його теплопровідності.

Аналогічну модель за умови (6.21) та умови

$$i(x, t) = q(x, t) \quad (6.24)$$

із рівняння (6.8) можна отримати для опису процесу розповсюдження теплового потоку $q(x, t)$ в часі та вздовж стержня, а саме:

$$\frac{\partial^2 q(x, t)}{\partial x^2} - RC \frac{\partial q(x, t)}{\partial t} = 0, \quad (6.25)$$

для якої є також справедливою умова (6.23).

Слід зазначити, що рівняннями параболічного типу описуються також процеси передачі низькочастотних сигналів в телеграфних і телефонних кабелях, для яких в достатньому для практичних розрахунків наближенні виконується умова (6.21). Тож процес розповсюдження напруги $u(x, t)$ в такому кабелі досить точно описується рівнянням

$$\frac{\partial^2 u(x, t)}{\partial x^2} - RC \frac{\partial u(x, t)}{\partial t} = 0, \quad (6.26)$$

а процес розповсюдження струму – рівнянням

$$\frac{\partial^2 i(x, t)}{\partial x^2} - RC \frac{\partial i(x, t)}{\partial t} = 0. \quad (6.27)$$

З теорії подібності відомо, що аналогом електричної напруги $u(x, t)$ для механічних систем є швидкість руху тіла $v(x, t)$, аналогом електричного струму $i(x, t)$ є сила $F(x, t)$, яка діє на тіло, а аналогами індуктивності L , активного опору R та ємності C , відповідно, є податливість n , механічна провідність тертя h та маса m .

Виходячи з указаних аналогій, а також з того, що для тросів, як і для трубопроводів, справедливою є умова (6.13), із рівнянь (6.7), (6.8) можна записати математичні моделі для троса у вигляді:

$$\frac{\partial^2 v(x, t)}{\partial x^2} - n \cdot m \frac{\partial^2 v(x, t)}{\partial t^2} - h \cdot m \frac{\partial v(x, t)}{\partial t} = 0, \quad (6.28)$$

$$\frac{\partial^2 F(x, t)}{\partial x^2} - n \cdot m \frac{\partial^2 F(x, t)}{\partial t^2} - h \cdot m \frac{\partial F(x, t)}{\partial t} = 0. \quad (6.29)$$

Ці моделі суттєво спрощуються, не вносячи суттєвих похибок, якщо не враховувати механічну провідність тертя в тросі, тобто за умови, що

$$h = 0. \quad (6.30)$$

У цьому випадку з рівнянь (6.28), (6.29) матимемо:

$$\frac{\partial^2 v(x, t)}{\partial x^2} - n \cdot m \frac{\partial^2 v(x, t)}{\partial t^2} = 0, \quad (6.31)$$

$$\frac{\partial^2 F(x, t)}{\partial x^2} - n \cdot m \frac{\partial^2 F(x, t)}{\partial t^2} = 0. \quad (6.32)$$

Рівняння (6.31), (6.32) в математичній фізиці носять назву рівнянь коливання струни.

Завершимо розгляд математичних моделей об'єктів з розподіленими параметрами моделлю широко розповсюдженого теплообмінника типу «труба в трубі» (рис. 6.3).

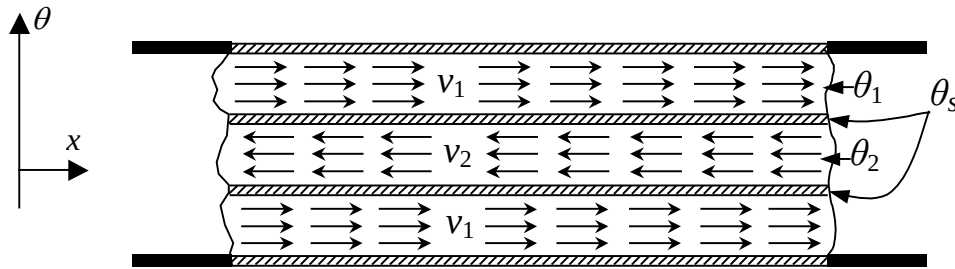


Рисунок 6.3 – Повздовжній розріз секції теплообмінника типу «труба в трубі»

Цю модель запишемо лише для процесу зміни в часі та вздовж труби температури в рідких середовищах θ_1 , θ_2 та у стінці θ_s , що їх розділяє.

При розробці цієї моделі використано дуже розповсюджений і практично підтверджений принцип, що швидкість зміни температури θ рідини, що переміщується зі швидкістю v в часі та вздовж труби, яка характеризується відповідними похідними, пропорційна різниці температур цієї рідини та стінки, від якої надходить потік тепла, тобто

$$\begin{cases} \frac{\partial \theta_1}{\partial t} + \lambda_1 \frac{\partial \theta_1}{\partial x} = \chi_{s1}(\theta_s - \theta_1), \\ \frac{\partial \theta_2}{\partial t} + \lambda_2 \frac{\partial \theta_2}{\partial x} = \chi_{s2}(\theta_s - \theta_2), \\ \frac{\partial \theta_s}{\partial t} = \chi_{1s}(\theta_1 - \theta_s) + \chi_{2s}(\theta_2 - \theta_s), \end{cases} \quad (6.33)$$

де θ_1 – температура рідини в зовнішній трубі, θ_2 – температура рідини у внутрішній трубі, θ_s – температура тонкої стінки внутрішньої труби, яка розділяє рідини, χ_{s1} , χ_{s2} , χ_{1s} , χ_{2s} – коефіцієнти теплопровідності між стінкою внутрішньої труби і рідинами, λ_1 , λ_2 – коефіцієнти, які характеризують режими роботи теплообмінника і визначаються через швидкість рідин, тобто:

- в режимі току (коли обидві рідини рухаються по трубах в одному напрямку):

$$\lambda_1 = v_1, \quad \lambda_2 = v_2, \quad (6.34)$$

- в режимі проти току (коли обидві рідини рухаються по трубах назустріч одна одній):

$$\lambda_1 = v_1, \quad \lambda_2 = -v_2. \quad (6.35)$$

Моделлю (6.33) і завершимо розгляд моделей елементів систем з розподіленими параметрами, що базуються на рівняннях математичної фізики.

Очевидно, що у випадках, коли необхідно проаналізувати лише процеси, які протікають в конкретних об'єктах з розподіленими параметрами, достатньо задати початкові та граничні умови і розв'язати відповідні рівняння математичної фізики, що описують процеси в цих об'єктах. Якщо ж об'єкт з розподіленими параметрами є складовою ланкою системи, то аналіз процесу в ньому буде не таким простим.

6.3 Аналіз математичних моделей систем з розподіленими параметрами

В найбільш загальному вигляді математична модель об'єкта з розподіленими параметрами на основі диференціальних рівнянь в частинних похідних отримана у вигляді (6.6).

Як показано в попередньому підрозділі, із системи рівнянь (6.6) можна отримати всі інші важливі для практики моделі об'єктів з розподіленими параметрами як в класі диференціальних рівнянь математичної фізики гіперболічного типу, так і параболічного.

Тож немає потреби аналізувати всі отримані в попередньому підрозділі моделі. Достатньо провести аналіз найбільш загальної моделі у вигляді (6.6) і показати як розповсюдити отримані результати на всі інші випадки.

Вважаючи початкові умови нульовими, перетворимо за Лапасом рівняння системи (6.6), вважаючи незалежною змінною час t , а x розглядаючи як параметр. В результаті перетворення отримаємо:

$$-\frac{dU(x, p)}{dx} = (R + pL) \cdot I(x, p), \quad (6.36)$$

$$-\frac{dI(x, p)}{dx} = (G + pC) \cdot U(x, p). \quad (6.37)$$

Звертаємо увагу, що після перетворення за Лапласом, тобто взяття інтегралів Лапласа за змінною t , функції $U(x, p)$, $I(x, p)$ стають функціями лише однієї незалежної змінної X з параметром P , тож в рівняннях (6.36), (6.37) маємо право розглядати не частинні похідні від функцій, як було в рівняннях (6.6), а звичайні.

Продиференціювавши за X рівняння (6.36), отримаємо:

$$-\frac{d^2 U(x, p)}{dx^2} = (R + pL) \frac{dI(x, p)}{dx}. \quad (6.38)$$

Підставивши в рівняння (6.38) значення похідної струму з рівняння (6.37) і перенісши всі члени в одну сторону, матимемо:

$$\frac{d^2 U(x, p)}{dx^2} - (R + pL) \cdot (G + pC) \cdot U(x, p) = 0. \quad (6.39)$$

Рівняння (6.39) є диференціальним рівнянням 2-го порядку відносно $U(x, p)$ за змінною X , характеристичне рівняння якого має вигляд:

$$\gamma^2 - (R + pL) \cdot (G + pC) = 0. \quad (6.40)$$

Коренями рівняння (6.40), очевидно, є

$$\gamma_{1,2} = \pm \sqrt{(R + pL) \cdot (G + pC)}. \quad (6.41)$$

Зауважимо, що корінь γ_1 , як правило, називають *коефіцієнтом розповсюдження хвилі*, або *хвильовою сталою*, і позначають як

$$\gamma(p) = \sqrt{(R + pL) \cdot (G + pC)}. \quad (6.42)$$

Із виразів (6.41), (6.42) витікає, що

$$\gamma_1(p) = \gamma(p), \quad \gamma_2(p) = -\gamma(p). \quad (6.43)$$

З теорії диференціальних рівнянь відомо, що загальний розв'язок диференціального рівняння 2-го порядку (6.39) можна записати у вигляді:

$$U(x, p) = a_1(p) \cdot e^{\gamma_1(p)x} + a_2(p) \cdot e^{\gamma_2(p)x}, \quad (6.44)$$

або, враховуючи вирази (6.43), у вигляді:

$$U(x, p) = a_1(p) \cdot e^{\gamma(p)x} + a_2(p) \cdot e^{-\gamma(p)x}. \quad (6.45)$$

Для визначення $a_1(p)$, $a_2(p)$ використаємо граничні умови, які, як видно з рис. 6.2, мають вигляд:

$$U(0, p) = E(p) - I(0, p) \cdot Z_{II}(p) \quad \text{для } x = 0, \quad (6.46)$$

$$U(l, p) = I(l, p) \cdot Z_K(p) \quad \text{для } x = l. \quad (6.47)$$

Підставляючи $x=0$ в рівняння (6.44), отримаємо:

$$U(0, p) = a_1(p) + a_2(p), \quad (6.48)$$

або, з врахуванням умови (6.46):

$$a_1(p) + a_2(p) = E(p) - I(0, p) \cdot Z_{II}(p). \quad (6.49)$$

Продиференціювавши вираз (6.45) за x , отримаємо:

$$\frac{dU(x, p)}{dx} = \gamma(p) \cdot a_1(p) \cdot e^{\gamma(p)x} - \gamma(p) \cdot a_2(p) \cdot e^{-\gamma(p)x}. \quad (6.50)$$

Зіставлення рівнянь (6.36) та (6.50) дає нам право записати, що

$$-(R + pL) \cdot I(x, p) = \gamma(p) \cdot a_1(p) \cdot e^{\gamma(p)x} - \gamma(p) \cdot a_2(p) \cdot e^{-\gamma(p)x}, \quad (6.51)$$

або

$$I(x, p) = \frac{\gamma(p) \cdot a_2(p)}{R + pL} e^{-\gamma(p)x} - \frac{\gamma(p) \cdot a_1(p)}{R + pL} e^{\gamma(p)x}. \quad (6.52)$$

При $x=0$ з рівняння (6.52) матимемо:

$$I(0, p) = \frac{\gamma(p) \cdot a_2(p)}{R + pL} - \frac{\gamma(p) \cdot a_1(p)}{R + pL}, \quad (6.53)$$

При $x=l$ із рівняння (6.52) матимемо:

$$I(l, p) = \frac{\gamma(p) \cdot a_2(p)}{R + pL} e^{-\gamma(p)l} - \frac{\gamma(p) \cdot a_1(p)}{R + pL} e^{\gamma(p)l}, \quad (6.54)$$

а з рівняння (6.45) матимемо:

$$U(l, p) = a_1(p) \cdot e^{\gamma(p)l} + a_2(p) \cdot e^{-\gamma(p)l}. \quad (6.55)$$

Підставляючи вираз (6.53) в рівняння (6.49), а вирази (6.54) і (6.55) в умову (6.47), отримаємо два рівняння з двома невідомими $a_1(p)$, $a_2(p)$, а саме:

$$\begin{cases} a_1(p) + a_2(p) = E(p) - \frac{\gamma(p)a_2(p)Z_{II}(p)}{R + pL} + \frac{\gamma(p)a_1(p)Z_{II}(p)}{R + pL}, \\ a_1(p)e^{\gamma(p)l} + a_2(p)e^{-\gamma(p)l} = \frac{\gamma(p)a_2(p)Z_K(p)}{R + pL}e^{-\gamma(p)l} - \\ - \frac{\gamma(p)a_1(p)Z_K(p)}{R + pL}e^{\gamma(p)l}, \end{cases} \quad (6.56)$$

або

$$\begin{cases} a_1(p) \left(1 - \frac{\gamma(p) Z_{II}(p)}{R + pL} \right) + a_2(p) \left(1 + \frac{\gamma(p) Z_{II}(p)}{R + pL} \right) = E(p), \\ a_1(p) e^{\gamma(p)l} \left(1 + \frac{\gamma(p) Z_K(p)}{R + pL} \right) + a_2(p) e^{-\gamma(p)l} \left(1 - \frac{\gamma(p) Z_K(p)}{R + pL} \right) = 0. \end{cases} \quad (6.57)$$

Розв'язуючи систему рівнянь (6.57), знайдемо, що:

$$\begin{cases} a_1(p) = - \frac{(R + pL - \gamma(p) Z_K(p)) \cdot (R + pL) \cdot e^{-\gamma(p)l} \cdot E(p)}{(R + pL + \gamma(p) Z_{II}(p)) \cdot (R + pL + \gamma(p) Z_K(p)) e^{\gamma(p)l} - \dots} \\ \dots \frac{- (R + pL - \gamma(p) Z_K(p)) \cdot (R + pL - \gamma(p) Z_{II}(p)) e^{-\gamma(p)l}}{(R + pL + \gamma(p) Z_K(p)) \cdot (R + pL + \gamma(p) Z_{II}(p)) e^{\gamma(p)l} - \dots} \\ a_2(p) = \frac{(R + pL + \gamma(p) Z_K(p)) \cdot (R + pL) \cdot e^{\gamma(p)l} \cdot E(p)}{(R + pL + \gamma(p) Z_{II}(p)) \cdot (R + pL + \gamma(p) Z_K(p)) e^{\gamma(p)l} - \dots} \\ \dots \frac{- (R + pL - \gamma(p) Z_K(p)) \cdot (R + pL - \gamma(p) Z_{II}(p)) e^{-\gamma(p)l}}{(R + pL + \gamma(p) Z_K(p)) \cdot (R + pL + \gamma(p) Z_{II}(p)) e^{\gamma(p)l} - \dots} \end{cases} \quad (6.58)$$

Нехай

$$\begin{aligned} A(R, L, \gamma, Z_{II}, Z_K, p) &= (R + pL + \gamma(p) Z_{II}(p)) \times \\ &\times (R + pL + \gamma(p) Z_K(p)) \cdot e^{\gamma(p)l} - (R + pL - \gamma(p) Z_K(p)) \times \\ &\times (R + pL - \gamma(p) Z_{II}(p)) \cdot e^{-\gamma(p)l}. \end{aligned} \quad (6.59)$$

Підставляючи значення $a_1(p)$, $a_2(p)$ із виразів (6.58) у рівняння (6.45) з врахуванням виразу (6.59), отримаємо:

$$\begin{aligned} U(x, p) &= \left[(R + pL + \gamma(p) Z_K(p)) \cdot e^{-\gamma(p)(x-l)} - \right. \\ &\left. - (R + pL - \gamma(p) Z_K(p)) \cdot e^{\gamma(p)(x-l)} \right] \cdot \frac{E(p) \cdot (R + pL)}{A(R, L, \gamma, Z_{II}, Z_K, p)}. \end{aligned} \quad (6.60)$$

Оскільки передаточна функція об'єкта $W(x, p)$ – це відношення перетворених за Лапасом вихідного сигналу $U(x, p)$ до вхідного $E(p)$, то з рівняння (6.60) матимемо:

$$W(x, p) = \frac{R + pL}{A(R, L, \gamma, Z_{II}, Z_K, p)} \times \\ \times \left[(R + pL + \gamma(p)Z_K(p)) \cdot e^{-\gamma(p)(x-l)} - \right. \\ \left. - (R + pL - \gamma(p)Z_K(p)) \cdot e^{\gamma(p)(x-l)} \right], \quad (6.61)$$

а з врахуванням виразу (6.61) саме рівняння (6.60) можна записати і так:

$$U(x, p) = W(x, p) \cdot E(p). \quad (6.62)$$

Якщо довга лінія є об'єктом управління, а тому цікавими є процеси, що в ній протікають, як у часі, так і в просторі, тобто вздовж обох координат t , X , то в структурній схемі такої системи цю лінію слід враховувати передаточною функцією (6.61).

Якщо ж довга лінія є елементом зв'язку між регулятором та об'єктом управління, то нас не цікавитимуть процеси, які протікають у самій лінії вздовж координати X , а цікавитимуть лише передавальними властивостями лінії, і тому у структурній схемі її слід враховувати передаточною функцією $W(l, p)$, яку отримаємо із виразу (6.61) підстановкою $x = l$, тобто функцією

$$W(l, p) = \frac{2(R + pL)\gamma(p)Z_K(p)}{A(R, L, \gamma, Z_{II}, Z_K, p)}. \quad (6.63)$$

Для трубопроводів структура передаточних функцій (6.61), (6.63) зберігається, але в зв'язку з виконанням для них умови (6.13) змінюється вираз, за допомогою якого визначається параметр $\gamma(p)$, тобто замість виразу (6.42) будемо використовувати вираз

$$\gamma^*(p) = \sqrt{pC(R + pL)}. \quad (6.64)$$

А для процесу передачі тепла в довгому ізольованому середовищі у зв'язку з виконанням умови (6.21) змінюється і параметр $\gamma(p)$, тобто замість виразу (6.42) слід використовувати вираз

$$*\gamma(p) = \sqrt{pRC}, \quad (6.65)$$

і структура передаточних функцій (6.61), (6.63), тобто замість них слід використовувати передаточні функції у вигляді:

$$*W(x, p) = \frac{R}{*A(R, *\gamma, Z_{II}, Z_K, p)} \times \\ \times \left[(R + *\gamma(p)Z_K(p)) \cdot e^{-*\gamma(p)(x-l)} - (R - *\gamma(p)Z_K(p)) \cdot e^{*\gamma(p)(x-l)} \right], \quad (6.66)$$

$${}^*W(l, p) = \frac{2R \cdot {}^*\gamma(p) \cdot Z_K(p)}{{}^*A(R, {}^*\gamma, Z_H, Z_K, p)}, \quad (6.67)$$

де

$$\begin{aligned} {}^*A(R, {}^*\gamma, Z_H, Z_K, p) = & (R + {}^*\gamma(p)Z_H(p))(R + {}^*\gamma(p)Z_K(p))e^{*\gamma(p)l} - \\ & - (R - {}^*\gamma(p)Z_K(p))(R - {}^*\gamma(p)Z_H(p))e^{-*\gamma(p)l}. \end{aligned} \quad (6.68)$$

Нагадаємо, що рівнянням математичної фізики параболічного типу описується не лише процес передачі тепла, а й процес передачі низькочастотних сигналів в телефонних і телеграфних лініях. А тому для передаточних функцій телефонних і телеграфних довгих ліній теж будуть справедливими співвідношення (6.65) – (6.68).

Ще більш простими є вирази для $\gamma(p)$, $W(x, p)$, $W(l, p)$, за допомогою яких в системах управління описуються троси, оскільки для них виконується не лише умова (6.13), але й умова (6.30). Для тросів замість виразів (6.42), (6.61) та (6.63) матимемо:

$${}^*\gamma^*(p) = p\sqrt{LC} = p\sqrt{n \cdot m}, \quad (6.69)$$

$$\begin{aligned} {}^*W^*(x, p) = & \frac{2p \cdot n \cdot {}^*\gamma^*(p) \cdot Z_K(p)}{{}^*A^*(n, {}^*\gamma^*, Z_H, Z_K, p)} \times \\ & \times \left[(p \cdot n + {}^*\gamma^*(p)Z_K(p)) \cdot e^{-*\gamma^*(p)(x-l)} - \right. \\ & \left. - (p \cdot n - {}^*\gamma^*(p)Z_K(p)) \cdot e^{*\gamma^*(p)(x-l)} \right], \end{aligned} \quad (6.70)$$

$${}^*W^*(l, p) = \frac{2p \cdot n \cdot {}^*\gamma^*(p) \cdot Z_K(p)}{{}^*A^*(n, {}^*\gamma^*, Z_H, Z_K, p)}, \quad (6.71)$$

$$\begin{aligned} {}^*A^*(n, {}^*\gamma^*, Z_H, Z_K, p) = & (p \cdot n + {}^*\gamma^*(p)Z_H(p)) \times \\ & \times (p \cdot n + {}^*\gamma^*(p)Z_K(p)) \cdot e^{*\gamma^*(p)l} - (p \cdot n - {}^*\gamma^*(p)Z_K(p)) \times \\ & \times (p \cdot n - {}^*\gamma^*(p)Z_H(p)) \cdot e^{-*\gamma^*(p)l}. \end{aligned} \quad (6.72)$$

Тепер розглянемо більш детально модель теплообмінника «труба в трубі», що має вигляд системи рівнянь (6.33).

Перетворюючи за Лапласом третє рівняння системи (6.33), отримаємо:

$$\begin{aligned} \theta_s(x, p) = & \frac{\chi_{1s}}{p + \chi_{1s} + \chi_{2s}} \theta_1(x, p) + \frac{\chi_{2s}}{p + \chi_{1s} + \chi_{2s}} \theta_2(x, p) + \\ & + \frac{\theta_s(x, 0)}{p + \chi_{1s} + \chi_{2s}}, \end{aligned} \quad (6.73)$$

де

$$\theta_s(x, 0) = \theta_s(x, t) \Big|_{t=0} \quad (6.74)$$

– початкова умова вздовж стінки, що розділяє рідини з температурами θ_1 і θ_2 .

Аналогічно, перетворюючи за Лапласом перше і друге рівняння системи (6.33), матимемо:

$$\begin{cases} (p + \chi_{s1})\theta_1(x, p) + \lambda_1 \frac{d\theta_1(x, p)}{dx} = \chi_{s1}\theta_s(x, p) + \theta_1(x, 0), \\ (p + \chi_{s2})\theta_2(x, p) + \lambda_2 \frac{d\theta_2(x, p)}{dx} = \chi_{s2}\theta_s(x, p) + \theta_2(x, 0), \end{cases} \quad (6.75)$$

де

$$\begin{cases} \theta_1(x, 0) = \theta_1(x, t) \Big|_{t=0}, \\ \theta_2(x, 0) = \theta_2(x, t) \Big|_{t=0} \end{cases} \quad (6.76)$$

– початкові умови для температури в обох рідинах вздовж труби.

Підставляючи в рівняння (6.75) значення $\theta_s(x, p)$ із рівняння (6.73), отримаємо:

$$\begin{aligned} \lambda_1 \frac{d\theta_1(x, p)}{dx} + \left(p + \chi_{s1} - \frac{\chi_{1s}\chi_{s1}}{p + \chi_{1s} + \chi_{2s}} \right) \theta_1(x, p) = \\ = \frac{\chi_{2s}\chi_{s1}}{p + \chi_{1s} + \chi_{2s}} \theta_2(x, p) + \theta_1(x, 0) + \frac{\chi_{s1}\theta_s(x, 0)}{p + \chi_{1s} + \chi_{2s}}, \end{aligned} \quad (6.77)$$

$$\begin{aligned} \lambda_2 \frac{d\theta_2(x, p)}{dx} + \left(p + \chi_{s2} - \frac{\chi_{s2}\chi_{2s}}{p + \chi_{1s} + \chi_{2s}} \right) \theta_2(x, p) = \\ = \frac{\chi_{1s}\chi_{s2}}{p + \chi_{1s} + \chi_{2s}} \theta_1(x, p) + \theta_2(x, 0) + \frac{\chi_{s2}\theta_s(x, 0)}{p + \chi_{1s} + \chi_{2s}}. \end{aligned} \quad (6.78)$$

Перетворюючи рівняння (6.77), (6.78) за Лапласом за просторовою координатою X , матимемо:

$$\begin{aligned} \left(\lambda_1 q + p + \chi_{s1} - \frac{\chi_{s1}\chi_{1s}}{p + \chi_{1s} + \chi_{2s}} \right) \theta_1(q, p) = \\ = \frac{\chi_{2s}\chi_{s1}}{p + \chi_{1s} + \chi_{2s}} \theta_2(q, p) + \theta_1(q, 0) + \frac{\chi_{s1}\theta_s(q, 0)}{p + \chi_{1s} + \chi_{2s}} + \lambda_1 \theta_1(0, p), \end{aligned} \quad (6.79)$$

$$\begin{aligned} & \left(\lambda_2 q + p + \chi_{s2} - \frac{\chi_{s2}\chi_{2s}}{p + \chi_{1s} + \chi_{2s}} \right) \theta_2(q, p) = \\ & = \frac{\chi_{1s}\chi_{s2}}{p + \chi_{1s} + \chi_{2s}} \theta_1(q, p) + \theta_2(q, 0) + \frac{\chi_{s2}\theta_s(q, 0)}{p + \chi_{1s} + \chi_{2s}} + \lambda_2 \theta_2(0, p). \end{aligned} \quad (6.80)$$

Позначимо:

$$\begin{cases} B_1(q, p) = \lambda_1 q + p + \chi_{s1} - \frac{\chi_{s1}\chi_{1s}}{p + \chi_{1s} + \chi_{2s}}, \\ B_2(q, p) = \lambda_2 q + p + \chi_{s2} - \frac{\chi_{s2}\chi_{2s}}{p + \chi_{1s} + \chi_{2s}}. \end{cases} \quad (6.81)$$

З врахуванням виразів (6.81) рівняння (6.79), (6.80) можна переписати так:

$$\begin{aligned} \theta_1(q, p) &= \frac{\chi_{2s}\chi_{s1}}{B_1(q, p)(p + \chi_{1s} + \chi_{2s})} \theta_2(q, p) + \\ &+ \frac{\chi_{s1}\theta_s(q, 0)}{B_1(q, p)(p + \chi_{1s} + \chi_{2s})} + \frac{\theta_1(q, 0) + \lambda_1 \theta_1(0, p)}{B_1(q, p)}, \end{aligned} \quad (6.82)$$

$$\begin{aligned} \theta_2(q, p) &= \frac{\chi_{1s}\chi_{s2}}{B_2(q, p)(p + \chi_{1s} + \chi_{2s})} \theta_1(q, p) + \\ &+ \frac{\chi_{s2}\theta_s(q, 0)}{B_2(q, p)(p + \chi_{1s} + \chi_{2s})} + \frac{\theta_2(q, 0) + \lambda_2 \theta_2(0, p)}{B_2(q, p)}. \end{aligned} \quad (6.83)$$

Підставляючи праву частину рівняння (6.82) в рівняння (6.83), отримаємо рівняння відносно $\theta_2(q, p)$ та початкових $\theta_1(q, 0)$, $\theta_s(q, 0)$, $\theta_2(q, 0)$ і граничних $\theta_1(0, p)$, $\theta_2(0, p)$ умов, розв'язуючи яке отримаємо двовимірне зображення за Лапласом моделі температурного процесу $\theta_2(q, p)$, застосовуючи до якого двовимірне обернене перетворення за Лапласом, знайдемо $\theta_2(x, t)$.

За аналогією, підставляючи праву частину рівняння (6.83) у рівняння (6.82) і застосовуючи наведений вище алгоритм, знайдемо $\theta_1(x, t)$.

Слід зазначити, що якщо теплообмінник є елементом якоїсь замкнutoї системи керування, то в структуру цієї системи він увійде структурною схемою, наведеною на рис. 6.4.

Оскільки передаточні функції окремих ланок на цій схемі легко визначаються її зіставленням з рівняннями (6.82), (6.83), то окремо виписувати їх не будемо.

Але, як і в кінці попереднього підрозділу, ще раз наголошуємо на тому, що структурною схемою, наведеною на рис. 6.4, з відповідно визначеними двовимірними передаточними функціями, теплообмінник «труба в трубі» слід задавати лише у випадках, коли він є складовою ланкою більш складної

системи, наприклад, системи автоматичного керування, і процеси в ньому цікавитимуть нас у взаємозв'язку з процесами в інших складових ланках цієї системи.

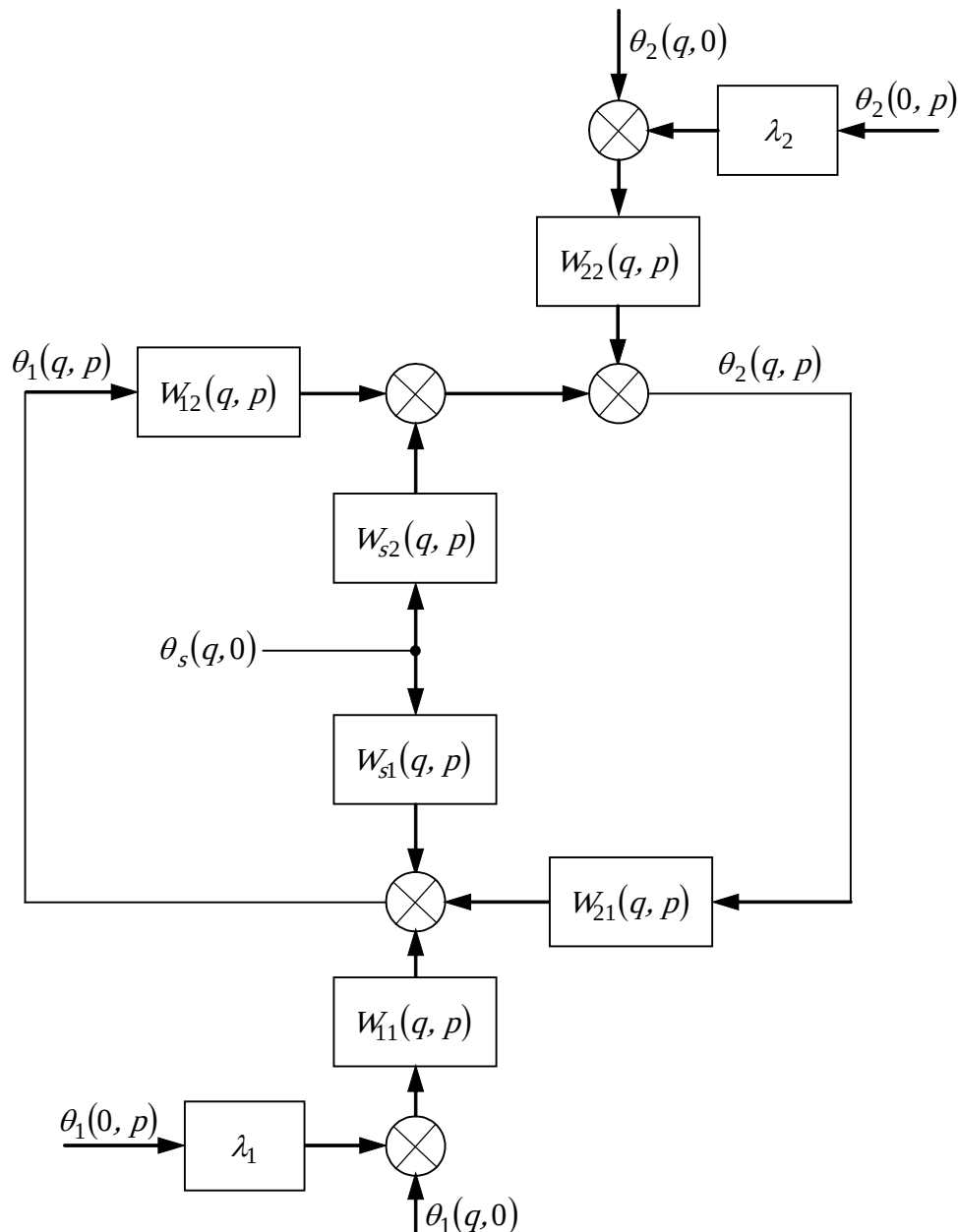


Рисунок 6.4 – Структурна схема моделі теплообмінника «труба в трубі»

Якщо ж нас цікавлять лише процеси, що протікають в теплообміннику, який розглядається сам по собі, то простіше такий аналіз здійснювати шляхом безпосереднього розв'язання рівнянь (6.33) чисельними методами за допомогою комп'ютерів, оснащених пакетами Python чи Matlab.

6.4 Математичні моделі систем з чистим запізненням

В класі динамічних систем з розподіленими параметрами особливе місце займають системи з чистим запізненням.

Це, наприклад, технологічні об'єкти з переміщенням матеріалів за допомогою конвеєрів, пристрої з лініями затримки сигналу чи хімічні реактори з конвеєрним або лопатевим переміщенням реагентів.

Їх особливе місце пояснюється тим, що після визначення чистого запізнення проходження сигналу в таких об'єктах процеси в них можна описувати за допомогою звичайних диференціальних рівнянь, а не рівнянь в частинних похідних, що значно простіше.

Почнемо розгляд моделей об'єктів з чистим запізненням з найпростішого з них – лінії затримки (ЛЗ) сигналу (рис. 6.5).

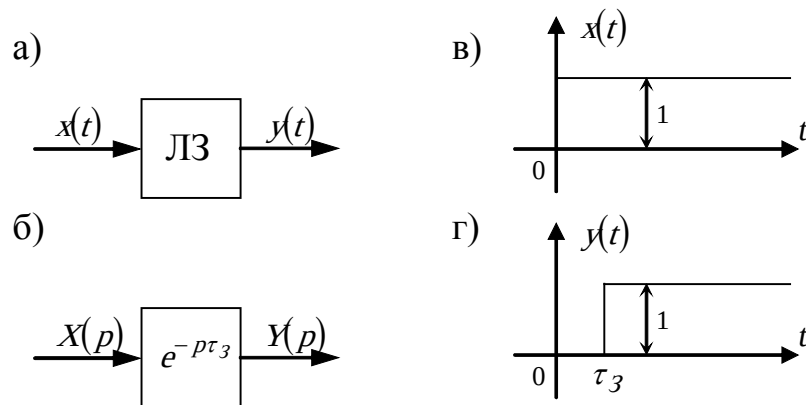


Рисунок 6.5 – Функціональна і структурна схеми лінії затримки та графіки її вхідного $x(t)$ і вихідного $y(t)$ сигналів

Як видно з рис. 6.5, в та рис. 6.5, г, ЛЗ лише затримує проходження вхідного сигналу, не змінюючи його форми.

Знайдемо передаточну функцію ЛЗ.

За визначенням вона дорівнює відношенню перетворених за Лапласом вихідного та вхідного сигналів. Тож

$$\begin{aligned}
 W_{ЛЗ}(p) &= \frac{Y(p)}{X(p)} = \frac{\int_0^{\infty} y(t) e^{-pt} dt}{\int_0^{\infty} x(t) e^{-pt} dt} = \\
 &= \frac{\int_0^{\infty} 1(t - \tau_z) e^{-pt} dt}{\int_0^{\infty} 1(t) e^{-pt} dt} = \frac{\frac{1}{p} e^{-p\tau_z}}{\frac{1}{p}} = e^{-p\tau_z}.
 \end{aligned} \tag{6.84}$$

Якщо згадати, що амплітудно-фазова частотна характеристика (АФЧХ) об'єкта – це

$$W(j\omega) = W(p)|_{p=j\omega} = P(\omega) + jQ(\omega) = A(\omega)e^{j\varphi(\omega)}, \quad (6.85)$$

де $P(\omega)$, $Q(\omega)$, $A(\omega)$, $\varphi(\omega)$ – відповідно, дійсна, уявна, амплітудна та фазова частотні характеристики, то із виразів (6.84), (6.85) випливає, що:

$$\begin{cases} W_{ЛЗ}(j\omega) = e^{-j\omega\tau_3}; \\ P_{ЛЗ}(\omega) = \cos(\omega\tau_3), \quad Q_{ЛЗ}(\omega) = -\sin(\omega\tau_3); \\ A_{ЛЗ}(\omega) = 1, \quad \varphi_{ЛЗ}(\omega) = -\omega\tau_3. \end{cases} \quad (6.86)$$

На рис. 6.6 наведені усі частотні характеристики ЛЗ, що визначаються співвідношеннями (6.86), які є її математичними моделями в частотній області.

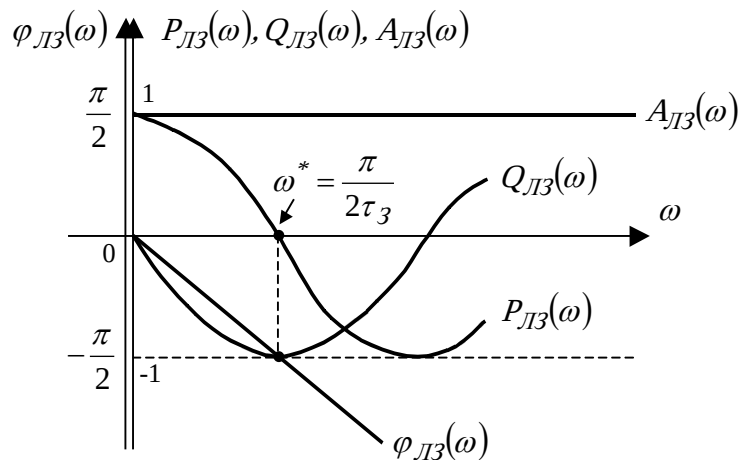


Рисунок 6.6 – Частотні характеристики лінії затримки сигналу

Очевидно, що всі моделі, визначені для лінії затримки сигналу, будуть справедливими і для конвеєрних ліній з тією лише різницею, що характеристики $W(p)$, $W(j\omega)$, $P(\omega)$, $Q(\omega)$ і $A(\omega)$ будуть мати множником коефіцієнт передачі k_K , який буде меншим одиниці в разі, якщо частина матеріалу, що насипається на конвеєр на його вході, просипаючись, не доходить до його кінця, тобто для конвеєра будуть справедливими вирази:

$$\begin{cases} W_K(p) = k_K e^{-p\tau_3}, \quad W_K(j\omega) = k_K e^{-j\omega\tau_3}, \\ P_K(\omega) = k_K \cos(\omega\tau_3), \quad Q_K(\omega) = -k_K \sin(\omega\tau_3), \\ A_K(\omega) = k_K, \quad \varphi_K(\omega) = -\omega\tau_3. \end{cases} \quad (6.87)$$

Перейдемо тепер до розгляду хімічних реакторів.

На рис. 6.7 наведені можливі варіанти перехідних характеристик хімічних реакторів $h_1(t)$, $h_{2a}(t)$, $h_{2k}(t)$, які є їх реакціями на одиничний вхідний сигнал $1(t)$ і які наближено можна описати так:

$$h_1(t, \tau_3) = y_0 \left(1 - e^{-\alpha_1(t-\tau_3)} \right) \cdot 1(t - \tau_3), \quad (6.88)$$

$$h_{2a}(t, \tau_3) = y_0 \left(1 - a_{1a} e^{-\alpha_{1a}(t-\tau_3)} + a_{2a} e^{-\alpha_{2a}(t-\tau_3)} \right) \cdot 1(t - \tau_3), \quad (6.89)$$

$$h_{2k}(t, \tau_3) = y_0 \left[1 - e^{-\alpha_{2k}(t-\tau_3)} \left(\cos(\beta(t-\tau_3)) - \frac{\alpha_{2k}}{\beta} \sin(\beta(t-\tau_3)) \right) \right] \times \\ \times 1(t - \tau_3), \quad (6.90)$$

В моделях (6.88), (6.89), (6.90) t – незалежна змінна (час), τ_3 – числовий параметр, який характеризує запізнення реакції об'єкта на вхідний сигнал, y_0 , a_{1a} , a_{2a} , α_1 , α_{1a} , α_{2a} , α_{2k} , β – числові коефіцієнти.

Зазначимо, що одиничні функції $1(t - \tau_3)$, що дописані як множники в вирази (6.88) – (6.90), необхідні в структурі цих виразів для того, щоб формально «обнуляти» функції $h_1(t, \tau_3)$, $h_{2a}(t, \tau_3)$, $h_{2k}(t, \tau_3)$ при значеннях аргументу t , менших τ_3 .

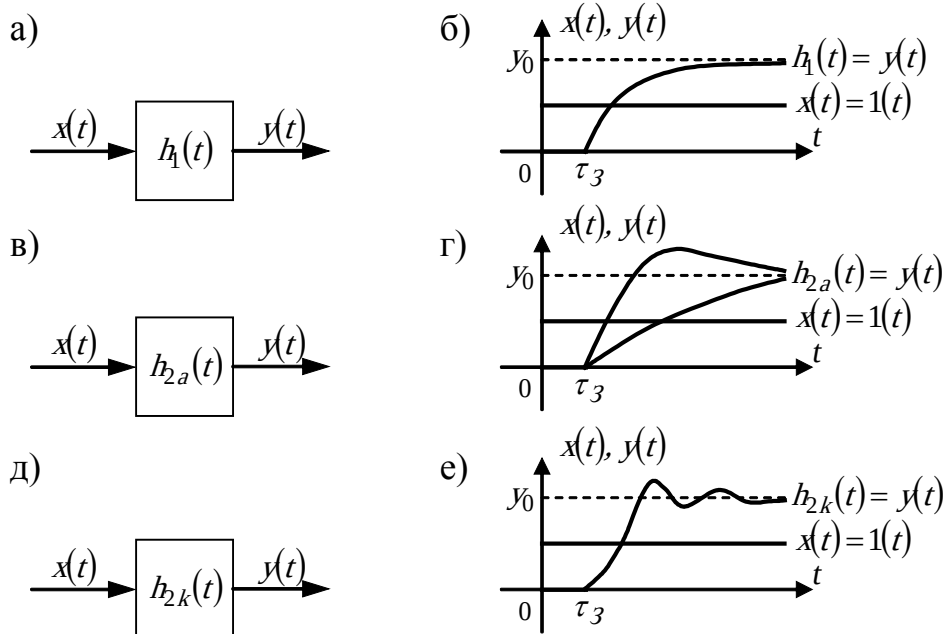


Рисунок 6.7 – Можливі варіанти перехідних характеристик хімічних реакторів

Формально, функції (6.88) – (6.90) є розв'язками диференціальних рівнянь, відповідно:

$$T_1 \frac{dy}{dt} + y = k \cdot x(t - \tau_3) \cdot 1(t - \tau_3), \quad (6.91)$$

при

$$\alpha_1 = \frac{1}{T_1}, \quad y_0 = k; \quad (6.92)$$

$$T_1 T_2 \frac{d^2 y}{dt^2} + (T_1 + T_2) \frac{dy}{dt} + y = k \cdot x(t - \tau_3) \cdot 1(t - \tau_3), \quad (6.93)$$

при

$$\begin{cases} a_{1a} = \frac{T_1}{T_1 - T_2}, & a_{2a} = \frac{T_2}{T_1 - T_2}, \\ \alpha_{1a} = \frac{1}{T_1}, & \alpha_{2a} = \frac{1}{T_2}, \quad y_0 = k; \end{cases} \quad (6.94)$$

$$T^2 \frac{d^2 y}{dt^2} + 2\xi T \frac{dy}{dt} + y = k \cdot x(t - \tau_3) \cdot 1(t - \tau_3) \quad (6.95)$$

при

$$\begin{cases} \beta = \frac{\sqrt{1 - \xi^2}}{T}, \\ \alpha_{2k} = \frac{\xi}{T}, \quad y_0 = k. \end{cases} \quad (6.96)$$

Подаючи на вхід хімічного реактора вхідний сигнал $x(t)$ у вигляді одиничної сходинок $1(t)$ і фіксуючи за допомогою самописця реакцію $y(t)$ цього хімічного реактора, отримаємо графік його перехідної характеристики $h(t - \tau_3)$, з якого одразу ж визначаються параметри τ_3 та y_0 .

Підставляючи ці параметри τ_3 та y_0 в рівняння (6.88), якщо графік отриманої експериментально перехідної характеристики $h_1(t, \tau_3)$ має вигляд, наведений на рис. 6.7, б, отримаємо рівняння з одним невідомим α_1 , для розв'язання якого відносно α_1 достатньо на графіку характеристики $h_1(t, \tau_3)$ взяти лише одну точку і підставити її координати $t^*, h_1(t^*, \tau_3)$ в рівняння (6.88).

Підставивши знайдені значення α_1 і y_0 в співвідношення (6.92), визначимо коефіцієнти моделі (6.91), чим завершимо її ідентифікацію.

Якщо графік експериментально отриманої перехідної характеристики $h_{2a}(t, \tau_3)$ має вигляд, наведений на рис. 6.7, г, то для отримання чотирьох параметрів a_{1a} , a_{2a} , α_{1a} , α_{2a} , які однозначно визначаються згідно зі співвідношеннями (6.94) лише через два параметри T_1 , T_2 , необхідно спочатку підставити співвідношення (6.94) в рівняння (6.89), а потім взяти дві точки на графіку характеристики $h_{2a}(t, \tau_3)$ і, підставивши їх координати $t^*, h_{2a}(t^*, \tau_3)$

та $t^{**}, h_{2a}(t^{**}, \tau_3)$ в рівняння (6.89), побудувати систему двох рівнянь з двома невідомими T_1, T_2 . Розв'язуючи цю систему двох рівнянь, знайдемо числові значення T_1, T_2 , підставляючи які в модель (6.93), здійснимо її ідентифікацію.

Якщо ж графік експериментально отриманої перехідної характеристики $h_{2k}(t, \tau_3)$ має вигляд, наведений на рис. 6.7, е, то для отримання двох невідомих параметрів α_{2k}, β необхідно взяти дві точки на графіку характеристики $h_{2k}(t, \tau_3)$ і, підставивши їх координати $t^*, h_{2k}(t^*, \tau_3)$ та $t^{**}, h_{2k}(t^{**}, \tau_3)$ в рівняння (6.90), побудувати систему двох рівнянь з двома невідомими α_{2k}, β , розв'язавши яку, знайти числові значення цих невідомих. Підставивши ці числові значення в рівняння (6.96) і розв'язавши їх, знайдемо числові значення параметрів T, ξ , підставивши які разом з коефіцієнтом k в модель (6.95), здійснимо її ідентифікацію.

Два важливих зауваження:

1. В книгах, присвячених аналізу систем з розподіленими параметрами, обов'язково фігурують графічні або графоаналітичні методи визначення коефіцієнтів моделей (6.91), (6.93), (6.95). Ці методи ми вирішили не розглядати, оскільки в еру комп'ютерів та пакетів прикладних програм типу Python чи Matlab, за допомогою яких рівняння 2-го порядку з будь-якими коефіцієнтами розв'язуються дуже просто і швидко, на наш погляд, недоцільно «засмічувати» матеріал викладенням методів, розроблених в іншу епоху і розрахованих на ручні розрахунки.

2. В околі точки стаціонарного режиму автоматичної системи в разі використання елемента з розподіленими параметрами типу хімічного реактора як об'єкт регулювання, в силу недопущення регулятором значних відхилень режиму у цій системі від стаціонарного, структуру такого об'єкта регулювання з достатнім для практичних задач ступенем адекватності можна подати у вигляді послідовного з'єднання ланки чистого запізнення і структурної ланки з передаточною функцією $W_i(p)$, $i = 1, 2, 3$ не вище другого порядку (рис. 6.8), де

$$W_1(p) = \frac{k}{T_1 p + 1} \quad (6.97)$$

– у випадку, якщо перехідна характеристика $h(t, \tau_3)$ має вигляд, наведений на рис. 6.7, б;

$$W_2(p) = \frac{k}{T_1 T_2 p^2 + (T_1 + T_2) p + 1} \quad (6.98)$$

– у випадку, якщо перехідна характеристика $h(t, \tau_3)$ має вигляд, наведений на рис. 6.7, г, і

$$W_3(p) = \frac{k}{T^2 p^2 + 2\xi T p + 1} \quad (6.99)$$

– у випадку, якщо перехідна характеристика $h(t, \tau_3)$ має вигляд, наведений на рис. 6.7, е.

В наведеному вище легко переконатись, якщо перетворити за Лапласом рівняння (6.91), (6.93), (6.95) і знайти відношення перетвореного за Лапласом вихідного сигналу $Y(p)$ до вхідного $X(p)$.

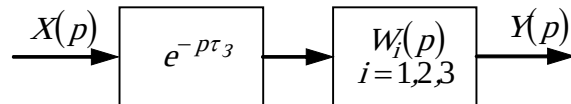


Рисунок 6.8 – Структурна схема об'єкта з розподіленими параметрами типу хімічного реактора як складової ланки системи автоматичного регулювання процесів в ньому

Завершимо цей підрозділ навчального посібника тим, як можна знайти математичну модель об'єкта з розподіленими параметрами у вигляді його імпульсної перехідної характеристики $g(t)$ з врахуванням запізнення τ_3 в разі, якщо і вхідний $x(t)$, і вихідний $y(t)$ сигнали цього об'єкта є стаціонарними випадковими процесами. У цьому випадку структурна схема об'єкта матиме вигляд, зображений на рис. 6.9, на якому D – це аналог комплексної змінної p в часовій області.

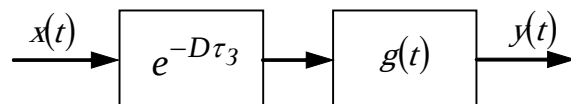


Рисунок 6.9 – Структурна схема об'єкта з розподіленими параметрами у вигляді послідовного з'єднання ланки чистого запізнення і ланки з імпульсною перехідною характеристикою $g(t)$

Як показано в другій частині цього навчального посібника, стаціонарні випадкові процеси $x(t)$, $y(t)$ у цьому випадку входять до рівняння Вінера-Хопфа кореляційною $R_x(\tau)$ та взаємною кореляційною $R_{yx}(\tau)$ функціями.

Користуючись тим же підходом, який використано у другій частині навчального посібника, легко показати, що рівняння Вінера-Хопфа у випадку об'єкта з розподіленими параметрами матиме вигляд:

$$R_{yx}(\tau, \tau_3) = \int_0^{\infty} R_x(\tau - \theta - \tau_3) g(\theta) d\theta, \quad (6.100)$$

в якому τ , θ є змінними, а τ_3 є параметром, який легко знаходиться з графіків $R_x(\tau)$, $R_{yx}(\tau)$ (рис. 6.12).

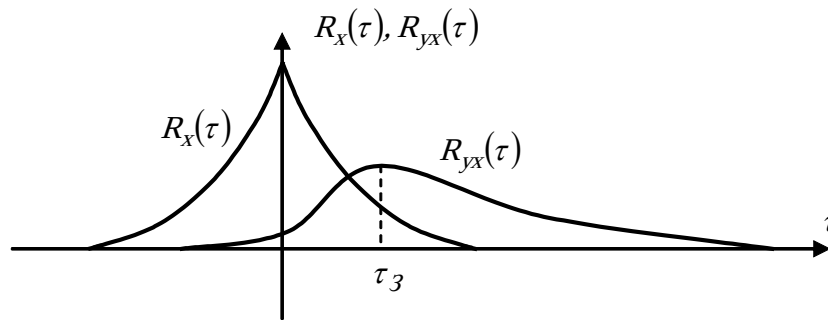


Рисунок 6.10 – Можливий варіант графіків функцій $R_x(\tau)$, $R_{yx}(\tau)$

Підставляючи значення τ_3 в рівняння (6.100) і розв'язуючи це рівняння одним із методів, наведених в другій частині навчального посібника, знайдемо функцію $g(t)$ для подання об'єкта з розподіленими параметрами у вигляді структури, зображеної на рис. 6.9.

6.5 Завдання для самоперевірки

В чому суть «розподіленості параметрів» об'єктів? Наведіть приклади об'єктів з розподіленими параметрами.

Що таке принципова схема однієї фази електричної системи? Зобразіть її.

Виведіть систему диференціальних рівнянь 1-го порядку в частинних похідних, які описують процеси в довгій електричній лінії.

Як отримати математичну модель довгої електричної лінії у вигляді диференціального рівняння в частинних похідних 2-го порядку гіперболічного типу?

За яких умов із загального рівняння гіперболічного типу можна отримати рівняння, які описують:

- процеси передачі тиску і об'ємних витрат в гідравлічних і пневматичних трубопроводах;
- процеси передачі температури в довгому ізолюваному середовищі;
- коливальні процеси в тросах?

Як виглядає модель у часовому просторі теплообмінника типу «труба в трубі»?

Отримайте передаточну функцію довгої лінії.

Як із загальної передаточної функції довгої лінії отримати передаточні функції трубопроводу, теплопроводу і троса?

Як виглядає структурна схема теплообмінника «труба в трубі», і якими є передаточні функції її структурних ланок?

Як виглядають передаточні функції і частотні характеристики ліній затримок і конвеєрів?

Як виглядають експериментальні перехідні характеристики хімічних реакторів?

За яким принципом підбираються математичні моделі хімічних реакторів?

Як визначити параметри математичних моделей хімічних реакторів?

Який вигляд має структурна схема хімічного реактора як складової частини системи автоматичного регулювання?

Запишіть множину можливих передаточних функцій хімічного реактора, що працює в складі системи автоматичного регулювання.

Як виглядає рівняння Вінера-Хопфа для об'єкта з розподіленими параметрами?

Як визначити чисте запізнення τ_z та імпульсну перехідну характеристику $g(t)$ об'єкта з розподіленими параметрами, на вхід якого надходить стаціонарний випадковий сигнал?

Розділ 7 МАТЕМАТИЧНІ МОДЕЛІ НЕЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З АНАЛІТИЧНИМИ НЕЛІНІЙНОСТЯМИ

Аналітичними нелінійностями будемо називати такі статичні характеристики нелінійних динамічних систем, тобто характеристики «вхід-вихід» після завершення перехідних процесів переходу від одного рівня входу до іншого, які не мають розривів і можуть бути апроксимовані неперервними функціями з неперервною першою похідною.

7.1 Математичні моделі нелінійних динамічних систем, статичні характеристики яких допускають лінеаризацію

Класичним і надзвичайно розповсюдженим прикладом нелінійних динамічних систем, статичні характеристики яких допускають лінеаризацію, є системи з електромеханічним перетворенням енергії, обов'язковою складовою яких є електромагнітні підсистеми. Цими складовими можуть бути електродвигуни, електричні генератори, електромагнітні муфти, магнітні підсилювачі, тягові соленоїди та інші елементи, характерною особливістю яких є наявність котушки індуктивності, намотаної на сталеве осердя.

Спільним для всіх цих елементів електромеханічної системи є те, що їх характеристики намагнічування, які є залежністю магнітного потоку Φ в осерді від струму I в обмотці, мають вигляд, наведений на рис. 7.1.

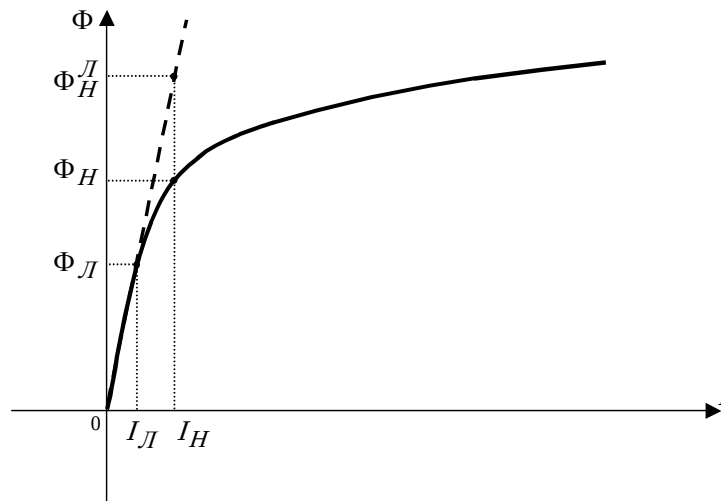


Рисунок 7.1 – Графік характеристики намагнічування $\Phi = f(I)$ електромагнітної системи

Із цього рисунка видно, що лінійною характеристику намагнічування $\Phi = f(I)$ можна вважати лише при значеннях струму I в обмотці, які не перевищують I_L . А при $I > I_L$ магнітна система починає насичуватись, і магнітний потік Φ в сталевому осерді хоча й збільшується зі зростанням струму I , але не пропорційно значенням струму.

Тож, якщо елемент електромеханічної системи з характеристикою намагнічування $\Phi = f(I)$, наведеною на рис. 7.1, працює при значеннях струму в обмотці $I < I_L$, то система з таким елементом буде належати до класу лінійних, і для математичного моделювання процесів в ній можна використовувати весь той математичний апарат, який наведено в попередніх двох частинах даного навчального посібника.

Але, якщо цей же елемент працює при значеннях $I > I_L$, наприклад, при $I = I_H$, то лінійною його характеристику намагнічування вже вважати не можна, оскільки, якщо її вважати лінійною, це означатиме, що при $I = I_H$ магнітний потік Φ в осерді повинен бути рівним Φ_H^L , в той час як насправді він дорівнює Φ_H (див. рис. 7.1).

Оскільки аналіз процесів в лінійних системах є найбільш простим і узагальненим, то плавні аналітичні нелінійності типу «насичення» намагаються «покрити» відрізками прямих так, щоб цих відрізків було найменше при допустимих значеннях похибок заміни реальної нелінійної кривої ламаною, складеною з відрізків прямих. Процес заміни плавних нелінійностей ламаними лініями, що складаються з відрізків прямих, називають їх лінеаризацією.

Приклад такого «покриття», тобто лінеаризації, наведено на рис. 7.2, на якому крива $0-a-b-c-d$ замінюється ламаною $0-a^*-b-c^*-d$, тобто вона «покривається» трьома відрізками $0-a^*$, a^*-c^* , c^*-d прямих.

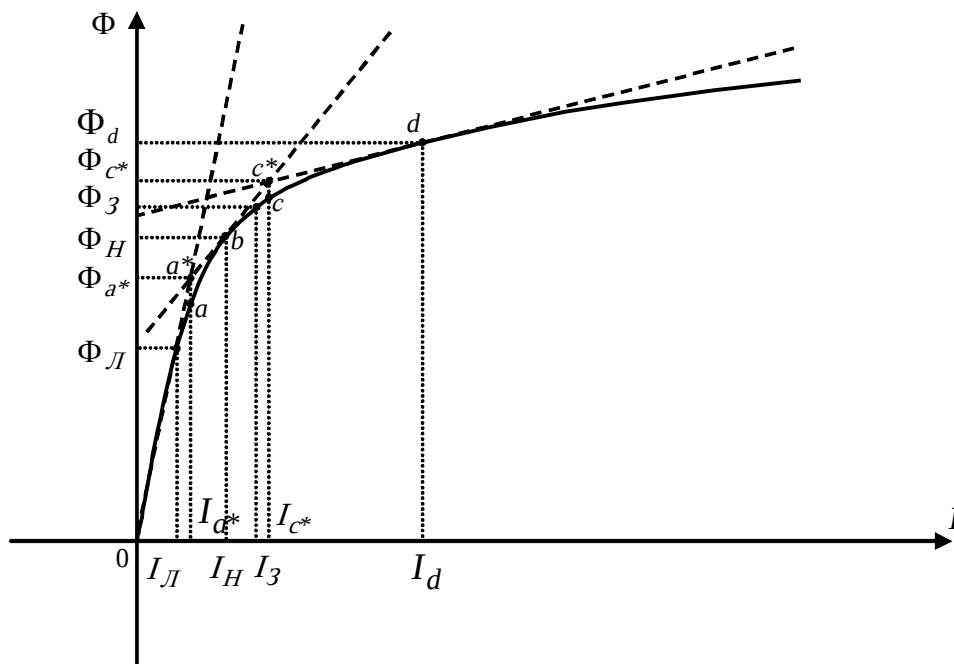


Рисунок 7.2 – Варіант «покриття» кривої намагнічування $\Phi = f(I)$ відрізками прямих

У цьому випадку під час аналізу процесів, що протікають в даному елементі електромеханічної системи, при значеннях струму $I < I_L$ як характеристику намагнічування $\Phi = f(I)$ будемо використовувати відрізок прямої $0 - a^*$, при значеннях струму $I_L \leq I < I_3$ – відрізок прямої $a^* - c^*$, а при значеннях струму $I_3 \leq I$ – відрізок прямої $c^* - d$.

Очевидно, що такою заміною кривої намагнічування ламаною лінією будемо мати певну похибку моделювання, найбільші значення якої досягатимуться в точках (I_{a^*}, Φ_{a^*}) , (I_{c^*}, Φ_{c^*}) .

Але якщо в усталеному режимі роботи елемент працює в точці (I_H, Φ_H) , то з рис. 7.2 легко побачити, що у цій точці похибка моделювання дорівнюватиме нулю.

Тож виходить, що кускова лінеаризація плавної нелінійної характеристики елемента електромеханічної системи, допускаючи певну похибку при моделюванні перехідних процесів, дозволяє звести цю похибку до нуля при математичному моделюванні усталеного режиму.

Якщо елемент електромеханічної системи працює здебільшого не в номінальному (I_H, Φ_H) , а в якомусь іншому режимі, наприклад, (I_3, Φ_3) , то підхід залишається тим же, однак в цьому випадку точку «b» ламаної суміщаємо з точкою (I_3, Φ_3) кривої.

Три важливих зауваження:

1. Під час математичного моделювання нелінійних динамічних систем з елементами, що мають аналітичні нелінійні характеристики типу наведених на рис. 7.1 та рис. 7.2, необхідно не забувати «зшивати» рішення сусідніх математичних моделей в моменти часу, коли процес, що моделюється, входить в точки стику ламаних, якими лінеаризується нелінійна характеристика.

2. Якщо моделюється нелінійна динамічна система з регулятором, що не допускає суттєвих відхилень процесу від заданого режиму роботи, то при моделюванні усталеного режиму такої системи та близьких до нього перехідних режимів завжди достатньо враховувати лише один лінійний відрізок ламаної, що «покриває» нелінійність в області, для якої точка заданого режиму є внутрішньою. При цьому чим більшим є коефіцієнт підсилення регулятором вхідного сигналу, тим меншою буде похибка моделювання, оскільки з ростом коефіцієнта підсилення регулятора звужується область відхилення робочої точки на нелінійній характеристиці елемента системи від точки заданого режиму.

3. Кусково-лінійна апроксимація нелінійної характеристики елемента системи в загальному вигляді $y = f(x)$ непридатна для використання в задачах оптимізації цієї системи аналітичними методами, оскільки в точках

стику ламаної перша похідна $\frac{dy}{dx}$ має розрив 1-го роду, а другої похідної $\frac{d^2y}{dx^2}$ не існує взагалі, в той час як аналітичні методи оптимізації вимагають існування як 1-ої, так і 2-ої похідних нелінійної характеристики перетворення в усіх точках області оптимізації.

7.2 Метод ідентифікації нелінійних динамічних систем з екстремальними статичними характеристиками

Загальна характеристика методу. Запропоновано метод ідентифікації нелінійних динамічних систем з екстремальними характеристиками, для реалізації алгоритму якого на систему потрібно подавати спочатку сходячковий, а потім синусоїдальний вхідний впливи і реєструвати реакції системи на ці впливи. Розрахункові співвідношення запропонованого методу отримані згідно з ідеологією Фур'є-інтегрального методу ідентифікації і передбачають розкладення вхідного впливу і реакції системи на нього у відрізки рядів Фур'є.

7.2.1 Вихідні умови та постановка задачі

Досить широкому класу динамічних систем в хімічній технології та теплоенергетиці з вхідним впливом $x(t)$ та реакцією на нього $y(t)$, динаміка яких характеризується імпульсною перехідною характеристикою $g(t)$, притаманна статична характеристика $y = f(x)$, яка має екстремальний характер (рис. 7.3).

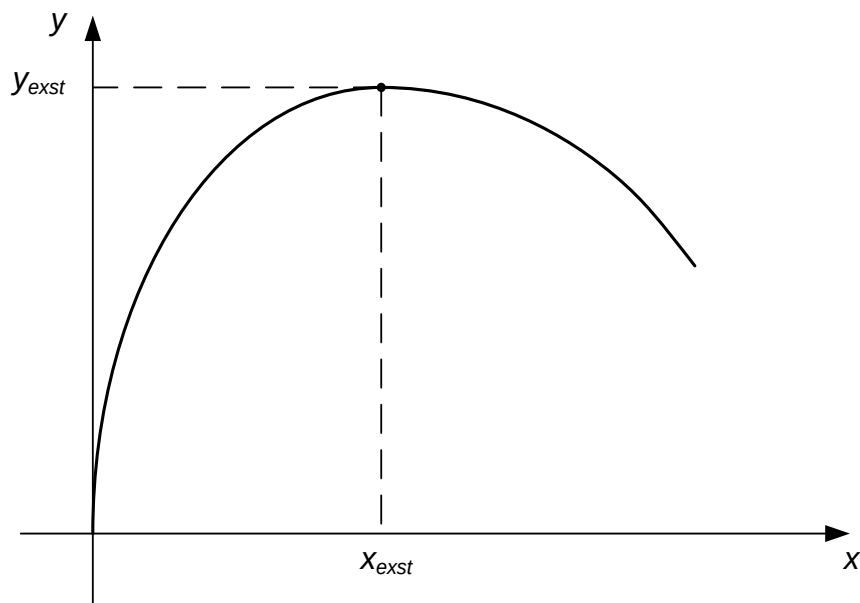


Рисунок 7.3 – Екстремальна статична характеристика нелінійної динамічної системи

Тож, щоб повністю ідентифікувати таку нелінійну систему, необхідно визначити обидві ці характеристики і $g(t)$, і $y = f(x)$.

Саме така задача і ставиться.

Для розв'язання цієї задачі висунемо дві вихідні передумови, згідно з першою з яких пропонується ідентифікувати статичну нелінійну характеристику $y = f(x)$ системи степеневим поліномом третього порядку, тобто

$$y = c_1 x + c_2 x^2 + c_3 x^3. \quad (7.1)$$

Звичайно, для апроксимації екстремальної статичної характеристики, наведеної на рисунку 7.3, можна використати і степеневий поліном другого порядку, оскільки у нього точка екстремуму буде мати такі ж самі координати (x_{ext}, y_{ext}) , але використаємо степеневий поліном третього порядку, тому що він точніше передає характер наростання координати y при початкових значеннях координати x .

А як другу вихідну передумову використаємо відомий підхід, оснований на тому, що структура динамічної системи розглядається у вигляді послідовного з'єднання її інерційної лінійної частини з імпульсною перехідною характеристикою $g(t)$ і проміжним вихідним сигналом $x^*(t)$ та нелінійної безінерційної ланки з характеристикою $y = f(x^*)$ (рис. 7.4).

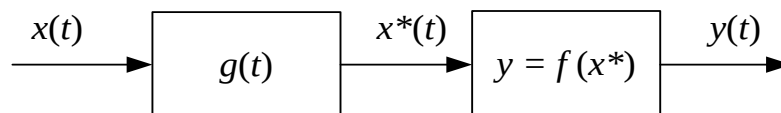


Рисунок 7.4 – Структура нелінійної динамічної системи з виділенням інерційної лінійної частини та безінерційної нелінійної частини

Очевидно, що після введення другої вихідної умови в рівнянні (7.1) потрібно замість X розглядати x^* .

7.2.2 Ідентифікація нелінійної статичної характеристики

Як відомо, вихідний сигнал $x^*(t)$ лінійної частини нелінійної динамічної системи можна знайти за допомогою інтеграла згортки

$$x^*(t) = \int_{-\infty}^{\infty} x(t-\tau)g(\tau)d\tau, \quad (7.2)$$

який за умови фізичної реалізованості цієї системи –

$$g(t) = \begin{cases} g(t), & t \geq 0, \\ 0, & t < 0, \end{cases} \quad (7.3)$$

набуває вигляду

$$x^*(t) = \int_0^{\infty} x(t-\tau)g(\tau)d\tau. \quad (7.4)$$

Згідно з теорією Фур'є-інтегрального методу ідентифікації динамічних систем розкладемо вхідний сигнал $x(t)$ у відрізок ряду Фур'є на вибраному проміжку часу T . В результаті цього матимемо

$$x(t) = \sum_{k=-n}^n a_k e^{jk\omega_1 t}, \quad (7.5)$$

де $\omega_1 = \frac{2\pi}{T}$, а

$$\begin{aligned} a_k &= \frac{1}{T} \int_0^T x(t) e^{-jk\omega_1 t} dt = \\ &= \frac{1}{T} \int_0^T x(t) \cos(k\omega_1 t) dt - j \frac{1}{T} \int_0^T x(t) \sin(k\omega_1 t) dt, \\ k &= -n, -(n-1), \dots, -1, 0, 1, 2, \dots, n. \end{aligned} \quad (7.6)$$

Одразу ж відзначимо, що при формуванні якогось сигналу фізичної системи з обмеженим запасом енергії, він завжди може бути апроксимований з заданою точністю відрізком ряду Фур'є.

Підставляючи значення $x(t)$ з виразу (7.5) у вираз (7.4), отримаємо

$$x^*(t) = \int_0^{\infty} \sum_{k=-n}^n a_k e^{jk\omega_1(t-\tau)} g(\tau) d\tau. \quad (7.7)$$

або

$$x^*(t) = \sum_{k=-n}^n a_k e^{jk\omega_1 t} \int_0^{\infty} g(\tau) e^{-jk\omega_1 \tau} d\tau. \quad (7.8)$$

Якщо згадати, що передаточна функція лінійної частини нелінійної динамічної системи – це

$$W(p) = \int_0^{\infty} g(\tau) e^{-p\tau} d\tau, \quad (7.9)$$

а її амплітудно-фазова частотна характеристика (АФЧХ) – це

$$W(j\omega) = W(p) \Big|_{p=j\omega}, \quad (7.10)$$

то рівняння (7.8) нескладно привести до вигляду

$$x^*(t) = \sum_{k=-n}^n a_k W(jk\omega_1) e^{jk\omega_1 t}. \quad (7.11)$$

Підставляючи значення $x^*(t)$ з виразу (7.11) у вираз (7.1) матимемо

$$y(t) = c_1 \sum_{k=-n}^n a_k W(jk\omega_1) e^{jk\omega_1 t} + \\ + c_2 \left(\sum_{k=-n}^n a_k W(jk\omega_1) e^{jk\omega_1 t} \right)^2 + c_3 \left(\sum_{k=-n}^n a_k W(jk\omega_1) e^{jk\omega_1 t} \right)^3. \quad (7.12)$$

В разі якщо вхідний сигнал $x(t)$ являє собою синусоїду з частотою ω_1 , тобто

$$x(t) = A \sin \omega_1 t = A \left(\frac{e^{j\omega_1 t} - e^{-j\omega_1 t}}{2j} \right) = \frac{A}{2j} e^{j\omega_1 t} + \frac{A}{-2j} e^{-j\omega_1 t}, \quad (7.13)$$

рівняння (7.12) перетвориться на рівняння

$$y(t) = c_1 \left(a_{-1} W(-j\omega_1) e^{-j\omega_1 t} + a_1 W(j\omega_1) e^{j\omega_1 t} \right) + \\ + c_2 \left(a_{-1} W(-j\omega_1) e^{-j\omega_1 t} + a_1 W(j\omega_1) e^{j\omega_1 t} \right)^2 + \\ + c_3 \left(a_{-1} W(-j\omega_1) e^{-j\omega_1 t} + a_1 W(j\omega_1) e^{j\omega_1 t} \right)^3, \quad (7.14)$$

у якому

$$a_1 = \frac{A}{2j}, \quad a_{-1} = -\frac{A}{2j}. \quad (7.15)$$

Підносячи до степеню в рівнянні (7.14) і групуючи члени з однаковими гармонічними складовими, отримаємо

$$y(t) = 2c_2 a_{-1} a_1 W(j\omega_1) W(-j\omega_1) + \\ + \left(c_1 a_{-1} + 3c_3 a_{-1}^2 a_1 W(j\omega_1) W(-j\omega_1) \right) W(-j\omega_1) e^{-j\omega_1 t} + \\ + \left(c_1 a_1 + 3c_3 a_{-1} a_1^2 W(j\omega_1) W(-j\omega_1) \right) W(j\omega_1) e^{j\omega_1 t} + \\ + c_2 a_{-1}^2 W^2(-j\omega_1) e^{-j2\omega_1 t} + c_2 a_1^2 W^2(j\omega_1) e^{j2\omega_1 t} + \\ + c_3 a_{-1}^3 W^3(-j\omega_1) e^{-j3\omega_1 t} + c_3 a_1^3 W^3(j\omega_1) e^{j3\omega_1 t}. \quad (7.16)$$

А далі розкладемо у відрізок ряду Фур'є на тому ж проміжку часу T вихідний сигнал $y(t)$, який є реакцією динамічної системи на вхідну синусоїду, тобто подамо його у вигляді

$$y(t) = \sum_{k=-m}^m b_k e^{jk\omega_1 t}, \quad (7.17)$$

де $\omega_1 = \frac{2\pi}{T}$, а

$$\begin{aligned} b_k &= \frac{1}{T} \int_0^T y(t) e^{-jk\omega_1 t} dt = \\ &= \frac{1}{T} \int_0^T y(t) \cos(k\omega_1 t) dt - j \frac{1}{T} \int_0^T y(t) \sin(k\omega_1 t) dt, \\ k &= -3, -2, -1, 0, 1, 2, 3. \end{aligned} \quad (7.18)$$

Оскільки у правій частині рівняння (7.16) маємо лише сталу складову і гармоніки з частотами $-3\omega_1, -2\omega_1, -\omega_1, \omega_1, 2\omega_1, 3\omega_1$, то зрізаний ряд (7.17) для реакції $y(t)$ системи на синусоїду частоти ω_1 теж буде мати лише ці складові – саме тому значення k при визначенні коефіцієнтів Фур'є b_k задані лише в межах від -3 до 3 .

Підставляючи значення $y(t)$ з виразу (7.17) при $m=3$ у рівняння (7.16), отримаємо тотожність, яка виконуватиметься лише тоді, коли коефіцієнти Фур'є при однакових гармонійних складових у правій та лівій частинах цієї тотожності будуть рівними.

Завдяки цьому отримаємо таку систему рівнянь:

$$\begin{cases} b_0 = 2c_2 a_{-1} a_1 W(j\omega_1) W(-j\omega_1), \\ b_{-1} = (c_1 a_{-1} + 3c_3 a_{-1}^2 a_1 W(j\omega_1) W(-j\omega_1)) W(-j\omega_1), \\ b_1 = (c_1 a_1 + 3c_3 a_{-1} a_1^2 W(j\omega_1) W(-j\omega_1)) W(j\omega_1), \\ b_{-2} = c_2 a_{-1}^2 W^2(-j\omega_1), \\ b_2 = c_2 a_1^2 W^2(j\omega_1), \\ b_{-3} = c_3 a_{-1}^3 W^3(-j\omega_1), \\ b_3 = c_3 a_1^3 W^3(j\omega_1). \end{cases} \quad (7.19)$$

З шостого та сьомого рівнянь цієї системи знайдемо, що

$$c_3 = \frac{b_{-3} + b_3}{a_{-1}^3 W^3(-j\omega_1) + a_1^3 W^3(j\omega_1)}, \quad (7.20)$$

з четвертого та п'ятого рівнянь отримаємо

$$c_2 = \frac{b_{-2} + b_2}{a_{-1}^2 W^2(-j\omega_1) + a_1^2 W^2(j\omega_1)}, \quad (7.21)$$

а з другого та третього –

$$c_1 = \frac{b_{-1} - f_{-1}(a_{-1}, a_1, c_3, W) + b_1 - f_1(a_{-1}, a_1, c_3, W)}{a_{-1} W(-j\omega_1) + a_1 W(j\omega_1)}, \quad (7.22)$$

де

$$\begin{cases} f_{-1}(a_{-1}, a_1, c_3, W) = (3c_3 a_{-1}^2 a_1 W(j\omega_1) W(-j\omega_1)) W(-j\omega_1), \\ f_1(a_{-1}, a_1, c_3, W) = (3c_3 a_{-1} a_1^2 W(j\omega_1) W(-j\omega_1)) W(j\omega_1). \end{cases} \quad (7.23)$$

Як бачимо, перше рівняння системи (7.19) є надлишковим, тож його можна використати як критерій правильності розв'язання задачі.

Ще одним критерієм правильності розв'язання задачі може служити поява суттєво відмінних від нуля значень коефіцієнтів Фур'є з номерами $k > 3$ та $-k < -3$ у вихідному сигналі $y(t)$ при його розкладенні в ряд (7.17) за умови, що на вхід системи надходить лише синусоїда однієї частоти ω_1 .

У цьому випадку реальна статична характеристика $y(x)$ об'єкта повинна апроксимуватись степеневим поліномом з порядком вище 3-го. При наявності гармонійних складових з частотою $4\omega_1$ та $-4\omega_1$ в сигналі $y(t)$ для апроксимації характеристики $y = f(x)$ поліном потрібно брати 4-го порядку, а при наявності гармонійних складових з частотою $5\omega_1$ та $-5\omega_1$ цей поліном повинен мати 5-ий порядок, і далі за зростанням.

Слід зазначити, що підвищення порядку полінома для апроксимації характеристики $y = f(x)$ не надто ускладнює отримання виразів для розрахунку коефіцієнтів цього полінома за умови використання на вході об'єкта синусоїди лише однієї частоти, оскільки для отримання співвідношень виду (7.19) і в цьому випадку в виразі виду (7.14) степені розкриваються за формулою бінома Ньютона.

Із співвідношень (7.20), (7.21), (7.22) випливає, що ідентифікувати екстремальну статичну характеристику $y = f(x)$ нелінійної динамічної системи класу, що розглядається, можна лише за умови, що є відомими значення АФЧХ $W(j\omega)$ лінійної інерційної частини цієї системи на частотах

ω_1 та $-\omega_1$, тобто відомими є $W(j\omega_1)$ та $W(-j\omega_1)$. Тож далі піде мова про те, як знайти ці значення.

7.2.3 Ідентифікація лінійної інерційної частини нелінійної динамічної системи

З фізики будь-якої динамічної системи випливає, що допоки її масовий чи енергетичний об'єм не наповниться масою чи енергією, які надходять разом із вхідним впливом, принаймні, наполовину, доти процеси в цій системі нарастають лінійно за винятком близької до нуля зони нечутливості або люфту.

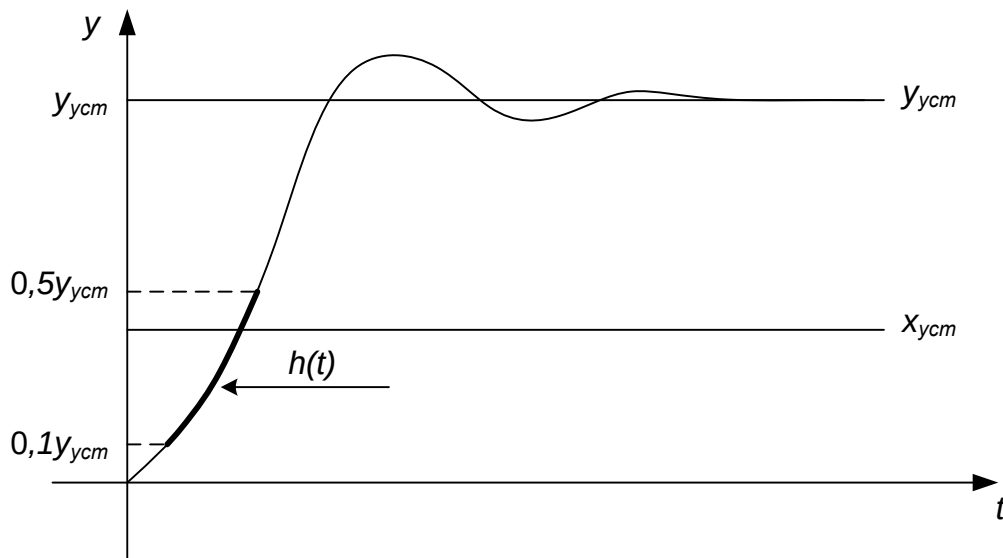


Рисунок 7.5 – Графік переходного процесу в динамічній системі після подачі на її вхід сходинок впливу на рівні його усталеного значення

Тож, якщо сформулювати вхідний вплив на нелінійну систему так, щоб його рівень відповідав усталеному рівню реакції цієї системи на прикладений вхідний вплив (рис. 7.5), то можна стверджувати, що в діапазоні значень вихідної координати $y(t)$ від $0,1y_{уст}$ до $0,5y_{уст}$ система вестиме себе як лінійна.

І якщо вхідний вплив подається стрибком від 0 до $x_{уст}$, то у вказаному діапазоні значень $y(t)$ вихідна координата буде збігатися з переходною характеристикою $h(t)$ лінійної частини цієї системи, яка для системи з одним об'ємом концентрації енергії чи маси має вигляд:

$$h(t) = K \left(1 - e^{-\frac{t}{T_1}} \right), \quad (7.24)$$

для об'єкта з двома об'ємами концентрації енергії чи маси –

$$h(t) = K \left(1 - \lambda_1 e^{-\frac{t}{T_1}} - \lambda_2 e^{-\frac{t}{T_2}} \right), \quad \lambda_1 + \lambda_2 = 1, \quad (7.25)$$

або

$$h(t) = K \left(1 - e^{-\frac{t}{T_1}} \cos \beta t \right), \quad (7.26)$$

а для об'єкта з трьома об'ємами концентрації енергії чи маси –

$$h(t) = K \left(1 - \lambda_1 e^{-\frac{t}{T_1}} - \lambda_2 e^{-\frac{t}{T_2}} - \lambda_3 e^{-\frac{t}{T_3}} \right), \quad \lambda_1 + \lambda_2 + \lambda_3 = 1, \quad (7.27)$$

або

$$h(t) = K \left(1 - \lambda_1 e^{-\frac{t}{T_1}} - \lambda_2 e^{-\frac{t}{T_2}} \cos \beta t \right), \quad \lambda_1 + \lambda_2 = 1. \quad (7.28)$$

Для більшої кількості об'ємів концентрації енергії чи маси ряд апроксимуючих функцій для $h(t)$ можна продовжити, користуючись тим самим принципом, але в цьому немає необхідності, оскільки в задачах синтезу систем екстремального регулювання порядок передаточної функції лінійної частини нелінійної динамічної системи немає сенсу брати вищим третього, адже екстремальний регулятор робочу точку від точки екстремуму далеко не відпускає.

Тож для апроксимації $h(t)$ у вигляді (7.24) потрібно з кривої $y(t)$ (див. рис. 7.5) скористатись двома точками в діапазоні $0,1y_{уст} \leq y(t) \leq 0,5y_{уст}$ для визначення параметрів K , T_1 . Очевидно, що їх можна знайти з системи рівнянь:

$$h(t_1) = K \left(1 - e^{-\frac{t_1}{T_1}} \right), \quad (7.29)$$

$$h(t_2) = K \left(1 - e^{-\frac{t_2}{T_1}} \right). \quad (7.30)$$

Для апроксимації $h(t)$ у вигляді (7.26) для визначення параметрів K , T_1 та β потрібно скористатись уже трьома точками з цього ж діапазону $[0,1y_{уст}; 0,5y_{уст}]$. Для них система рівнянь матиме вигляд:

$$\begin{cases} h(t_1) = K \left(1 - e^{-\frac{t_1}{T_1}} \cos \beta t_1 \right), \\ h(t_2) = K \left(1 - e^{-\frac{t_2}{T_1}} \cos \beta t_2 \right), \\ h(t_3) = K \left(1 - e^{-\frac{t_3}{T_1}} \cos \beta t_3 \right). \end{cases} \quad (7.31)$$

У такий же спосіб можна розрахувати параметри всіх інших апроксимацій характеристики $h(t)$.

Після цього необхідно, скориставшись кількома (наприклад, M) точками діапазону $[0, 1y_{уст}; 0, 5y_{уст}]$ з індексом l , розрахувати для всіх знайдених апроксимацій $h_i(t)$, $i = \overline{1, r}$ середній квадрат Δ_i похибки апроксимації за виразом

$$\Delta_i = \sum_{l=1}^M \left(y_l^* - h(t_l) \right)^2, \quad i = \overline{1, r}, \quad (7.32)$$

де y_l^* – значення перехідної характеристики, взяті з експериментального графіка (див. рис. 7.5), а $h(t_l)$ – значення цієї характеристики, вираховані за апроксимуючим виразом.

Те з чисел $\Delta_1, \Delta_2, \dots, \Delta_r$, яке буде найменшим, і визначатиме найкращу апроксимацію перехідної характеристики $h(t)$ лінійної частини динамічної системи за критерієм мінімуму суми квадратів похибок на «свіжих» точках.

Знаючи перехідну характеристику $h(t)$ лінійної частини об'єкта, легко знайти її передаточну функцію $W(p)$, адже

$$g(t) = \frac{dh(t)}{dt}, \quad (7.33)$$

а для отримання $W(p)$ за відомою імпульсною перехідною характеристикою $g(t)$ потрібно скористатись виразом (7.9).

В свою чергу значення $W(j\omega_1)$ та $W(-j\omega_1)$, які потрібні нам для ідентифікації екстремальної статичної характеристики $y = f(x)$, отримаємо прямою підстановкою у вираз для передаточної функції $W(p)$, отриманий за перетворенням (7.9), замість оператора P значень $j\omega_1$ та $-j\omega_1$.

Звичайно, якщо після кількох однотипних експериментів з подачею на вхід об'єкта однакового рівня вхідного впливу сходинкового характеру отримаємо експериментальні криві $y(t)$, які не збігатимуться у діапазоні $[0,1y_{уст}; 0,5y_{уст}]$, то параметри апроксимацій $h(t)$ у вигляді (7.24) – (7.28) слід обчислювати, користуючись стандартною процедурою методу найменших квадратів.

7.3 Завдання для самоперевірки

Які нелінійності відносять до аналітичних?

В чому полягає суть лінеаризації нелінійних характеристик динамічних систем?

Про що не слід забувати при лінеаризації нелінійностей?

Для якого класу нелінійних динамічних систем можна застосувати Фур'є-інтегральний метод ідентифікації?

Який вигляд мають зрізані ряди Фур'є при апроксимації ними вхідного і вихідного сигналів нелінійної динамічної системи? Як знайти коефіцієнти Фур'є цих сигналів?

Розкрийте суть алгоритму ідентифікації статичної характеристики нелінійної динамічної системи Фур'є-інтегральним методом?

Яким чином можна перевірити адекватність статичної характеристики нелінійної динамічної системи степені полінома, вибраного математичною моделлю?

Який експеримент потрібно провести на динамічній системі, щоб отримати інформацію, потрібну для ідентифікації її інерційної лінійної частини?

Розкрийте суть алгоритму ідентифікації лінійної частини нелінійної динамічної системи.

За допомогою якого критерію можна перевірити якість ідентифікації лінійної інерційної частини нелінійної динамічної системи?

Розділ 8 МАТЕМАТИЧНІ МОДЕЛІ НЕЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ З РЕЛЕЙНИМИ ХАРАКТЕРИСТИКАМИ

8.1 Математичні моделі релейних елементів з симетричними характеристиками

Практично жодна складна електрична, електромеханічна або радіотехнічна динамічна система не обходиться без релейних елементів, характеристики «вхід – вихід» яких мають вигляд, наведений на рис. 8.1.

Надалі релейні елементи з такими характеристиками будемо називати *симетричними*.

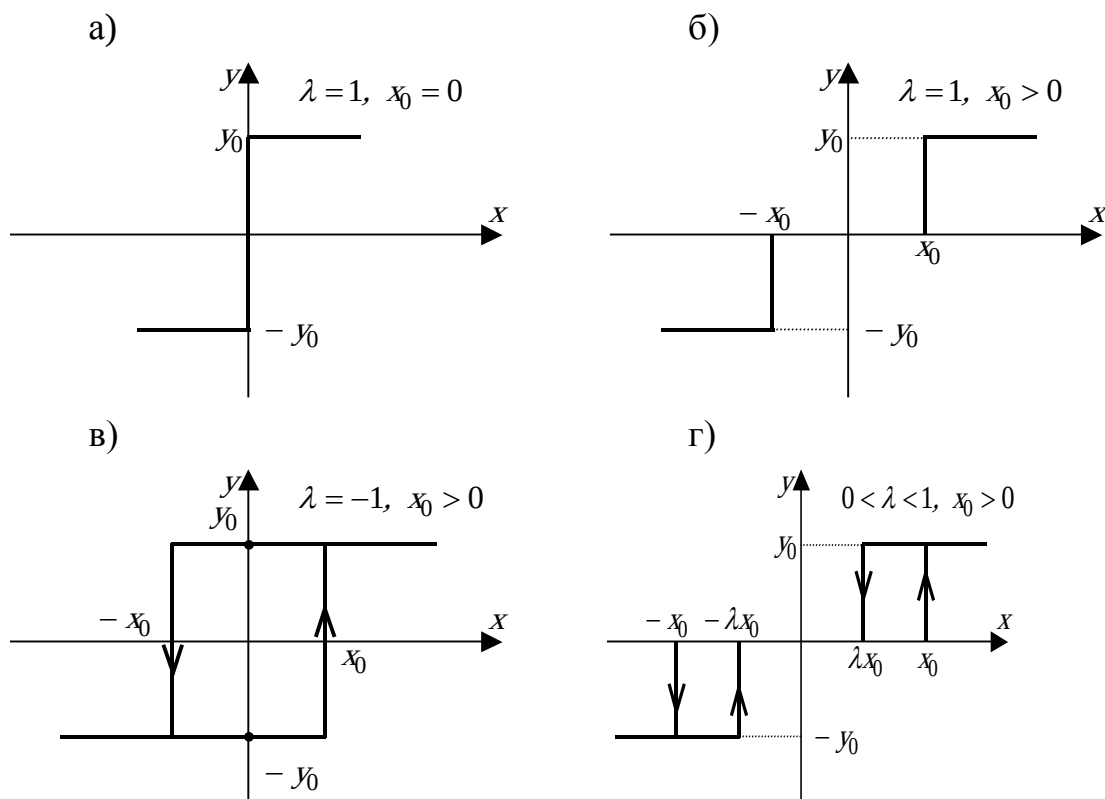


Рисунок 8.1 – Характеристики «вхід – вихід» симетричних релейних елементів: а) без зони нечутливості та гістерезису;

б) з зоною нечутливості, але без гістерезису;

в) без зони нечутливості, але з гістерезисом;

г) з зоною нечутливості та гістерезисом

Фізично найпростіше реалізувати симетричний релейний елемент з характеристикою, наведеною на рис. 8.1, б. Як показано на рис. 8.2, він являє собою котушку 1 з рухомим осердям 2, з'єднаним з рухомими контактами 3 чотирьох пар контактів, нерухомі контакти 4 яких з'єднані з джерелом, наприклад, постійного струму 5.

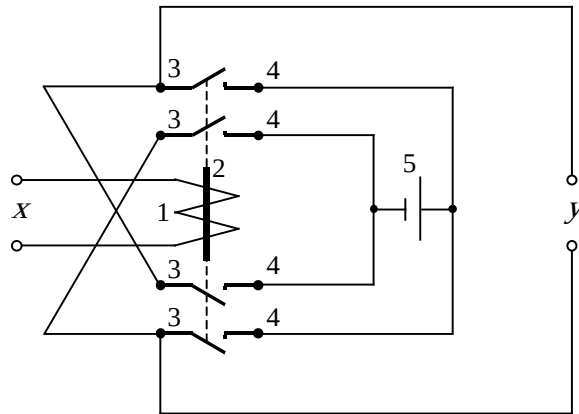


Рисунок 8.2 – Принципова схема релейного елемента з характеристикою, наведеною на рис. 8.1, б

При подачі на вхід цього елемента сигналу x у вигляді напруги, прикладеної до котушки 1, вихідний сигнал U у вигляді напруги джерела постійного струму з'явиться лише тоді, коли вхідна напруга стане більшою x_0 і створить такий струм, магніторушійна сила від якого стане достатньою для втягування осердя 2 і замикання верхньої пари контактів 3-4 при одній полярності вхідного сигналу x , або нижньої – при протилежній полярності.

Якщо магнітна характеристика осердя релейного елемента має практично помітну петлю гістерезису, то цей релейний елемент матиме характеристику «вхід – вихід», наведену на рис. 8.1, г.

Що стосується характеристик, наведених на рис. 8.1, а та рис. 8.1, в, то для їх реалізації потрібні вже спеціальні схеми виконання і включення котушки і осердя.

Схемними і конструктивними способами можна отримати й інший характер релейних характеристик, в тому числі і несиметричний уздовж однієї або обох осей координат.

В загальному вигляді математичну модель релейної характеристики без гістерезису можна записати як

$$y = \Phi(x), \quad (8.1)$$

а з гістерезисом – як

$$y(t) = \Phi^* \left(x(t) \Big|_0^t, \sigma \right). \quad (8.2)$$

Покажемо, який вигляд мають оператори Φ , Φ^* для характеристик, наведених на рис. 8.1.

Якщо ввести знакову функцію

$$\text{sign}(x) = \begin{cases} +1, & \forall x > 0, \\ 0, & x = 0, \\ -1, & \forall x < 0, \end{cases} \quad (8.3)$$

де символ $\forall$ читається як «для будь-якого», то оператор Φ моделі релейної характеристики без гістерезису і без зони нечутливості (див. рис. 8.1, а) матиме вигляд:

$$\Phi_1(x) = y_0 \cdot \text{sign}(x), \quad (8.4)$$

а оператор Φ моделі релейної характеристики без гістерезису, але з зоною нечутливості:

$$\Phi_2(x) = \frac{y_0}{2} [\text{sign}(x - x_0) + \text{sign}(x + x_0)]. \quad (8.5)$$

Для релейного елемента з гістерезисом (рис. 8.1, в та рис. 8.1, г) характерним є те, що його характеристика має зони неоднозначності, перебування в яких залежить не лише від поточних значень x , а й від значень x в попередні моменти часу. Крім того, цей оператор залежить від параметра σ , який для релейної характеристики без зони нечутливості має значення

$$\sigma_t = \{y_0, -y_0\}, \quad (8.6)$$

а для релейної характеристики з зоною нечутливості –

$$\sigma_t = \{y_0, 0, -y_0\}. \quad (8.7)$$

З урахуванням викладених зауважень оператор Φ^* моделі релейної характеристики з гістерезисом (рис. 8.1) має вигляд:

$$\Phi^* \left(x(t), \beta_{t_+}^{(i)}(\sigma_{t_-}) \right) = y_0 \cdot \text{sign} \left(\beta_{t_+}(\sigma_{t_-}) \right), \quad i = 1, 2, \quad (8.8)$$

де для характеристики без зони нечутливості (рис. 8.1, в)

$$\beta_{t_+}^{(1)}(\sigma_{t_-}) = \begin{cases} (x(t_+) > x_0) \vee (x_0 \geq x(t_+) > -x_0) \wedge (\sigma_{t_-} = y_0) = +1, \\ (x(t_+) < -x_0) \vee (-x_0 \leq x(t_+) < x_0) \wedge (\sigma_{t_-} = -y_0) = -1, \end{cases} \quad (8.9)$$

та

$$\beta_{t_+}^{(2)}(\sigma_{t_-}) = \begin{cases} (x(t_+) > x_0) \vee (x_0 \geq x(t_+) > \lambda x_0) \wedge (\sigma_{t_-} = y_0) = +1, \\ (\lambda x_0 \leq x(t_+) < x_0) \vee (-\lambda x_0 < x(t_+) < \lambda x_0) \wedge (\sigma_{t_-} = 0) = 0, \\ (-x_0 \leq x(t_+) < \lambda x_0) \vee (x(t_+) < -x_0) \wedge (\sigma_{t_-} = -y_0) = -1 \end{cases} \quad (8.10)$$

– для характеристики з зоною нечутливості (рис. 8.1, г).

Символами t_+ , t_- позначено значення моменту часу t за мить після його досягнення (t_+) і за мить до нього (t_-).

В подальшому, пам'ятаючи про те, що значення β_t розглядається при $t=t_+$, а значення σ_t розглядається при $t=t_-$, в загальному вигляді оператор моделі релейної характеристики з гістерезисом будемо записувати у вигляді

$$\Phi^*\left(x(t_+), \beta_{t_+}^{(i)}(\sigma_{t_-})\right) = \Phi^*(x, \beta(\sigma)). \quad (8.11)$$

Додаючи в рівняння (8.11) параметр $\pm x_0^*$ – зміщення вправо або вліво по осі x , або додаючи в рівняння (8.11) параметр $\pm y_0^*$ – зміщення вниз або вгору по осі y , можна сформулювати модель несиметричної релейної характеристики як з гістерезисом, так і без нього; як з зоною нечутливості, так і без неї.

Ми в цьому навчальному посібнику будемо розглядати моделі лише симетричних релейних характеристик.

Звертаємо увагу на те, що якщо вхідний сигнал x релейного елемента може змінюватись неперервно в заданому діапазоні значень, то вихідний сигнал y може мати лише два ($y_0, -y_0$) або три ($y_0, 0, -y_0$) значення, які з'являються на виході релейного елемента стрибком при досягненні його вхідним сигналом x порогового значення 0 , x_0 чи $-x_0$.

Ще однією характерною особливістю релейного елемента, яка впливає з попередньої, є те, що при подачі знакозмінного сигналу x на його вхід на виході цього елемента матимемо послідовність імпульсів однакової висоти, але різного знаку, різної ширини і з різними проміжками між ними.

Відзначимо, що якщо вхідний сигнал x не є випадковим процесом, то цілком визначеною буде і вихідна послідовність імпульсів релейного елемента, для якої завжди можна записати математичну модель.

8.2 Математичні моделі нелінійних динамічних систем з релейними елементами

Структуру будь-якої замкненої нелінійної динамічної системи з одним релейним елементом і з негативним зворотним зв'язком завжди можна привести до вигляду, наведеного на рис. 8.3, де РЕ – релейний елемент, а ЛЧ – сукупність усіх елементів з лінійними характеристиками «вхід – вихід».

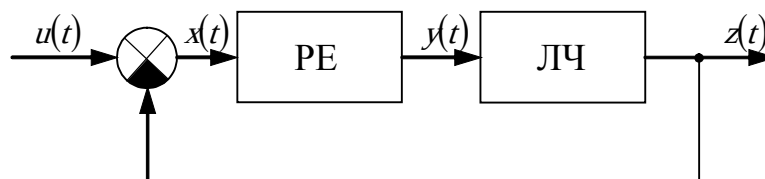


Рисунок 8.3 – Структурна схема замкненої нелінійної динамічної системи з одним релейним елементом і з негативним зворотним зв'язком

Якщо перетворити за Лапасом сигнали $u(t)$, $x(t)$, $y(t)$, $z(t)$ та алгебраїчні і диференціальні чи інтегральні рівняння, що описують лінійну частину ЛЧ, тобто перевести процес створення математичної моделі на комплексну площину, то структурну схему, наведену на рис. 8.3, можна привести до вигляду, наведеного на рис. 8.4, де $W(p)$ – передаточна функція ЛЧ.

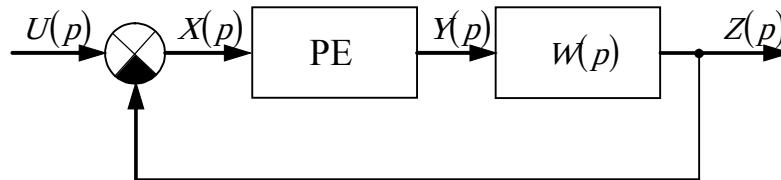


Рисунок 8.4 – Структурна схема замкненої нелінійної динамічної системи з одним релейним елементом і з негативним зворотним зв'язком на комплексній площині

Для схеми на рис. 8.4 цілком очевидними є такі співвідношення:

$$Z(p) = W(p) \cdot Y(p), \quad (8.12)$$

$$X(p) = U(p) - Z(p). \quad (8.13)$$

Що ж до зв'язку між сигналами x , y та характеристикою PE на комплексній площині, то формально його можна записати як

$$Y(p) = L\{\Phi(x(t))\} \quad (8.14)$$

у випадку математичної моделі PE у вигляді (8.1), де $L\{\bullet\}$ – оператор Лапласа;

$$Y(p) = L\{\Phi^*(x(t), \sigma)\} \quad (8.15)$$

– у випадку математичної моделі PE у вигляді (8.2);

$$Y(p) = L\{\Phi^*(x(t), \beta(\sigma))\} \quad (8.16)$$

– у випадку математичної моделі PE у вигляді (8.11).

Зв'язуючи між собою співвідношення (8.12), (8.13) і (8.14) або (8.15) чи (8.16), матимемо для сигналу $X(p)$ на вході PE:

$$X(p) = U(p) - W(p) \cdot L\{\Phi(x(t))\}, \quad (8.17)$$

або

$$X(p) = U(p) - W(p) \cdot L\{\Phi^*(x(t), \sigma)\}, \quad (8.18)$$

або

$$X(p) = U(p) - W(p) \cdot L\{\Phi^*(x(t), \beta(\sigma))\}. \quad (8.19)$$

Якщо ж викликає цікавість вихідний сигнал $Z(p)$ лінійної частини нелінійної системи, то, пам'ятаючи, що

$$x(t) = u(t) - z(t), \quad (8.20)$$

із співвідношень (8.12) – (8.16) та (8.20) матимемо:

$$Z(p) = W(p) \cdot L\{\Phi[u(t) - z(t)]\}, \quad (8.21)$$

або

$$z(p) = W(p) \cdot L\{\Phi^*[u(t) - z(t), \sigma]\}, \quad (8.22)$$

або

$$z(p) = W(p) \cdot L\{\Phi^*[u(t) - z(t), \beta(\sigma)]\}. \quad (8.23)$$

Розглянемо процес отримання вищенаведених співвідношень на двох прикладах.

Приклад 1. Нехай нелінійною динамічною системою, процеси в якій хочемо описати, є система регулювання електрорушійної сили $e(t)$ генератора за допомогою вібраційного регулятора на основі релейного елемента.

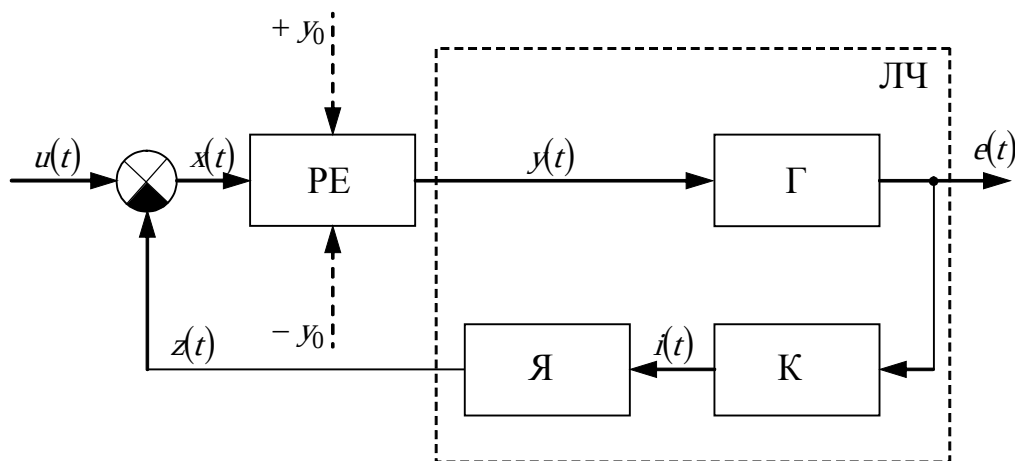


Рисунок 8.5 – Функціональна схема системи регулювання ЕРС генератора за допомогою вібраційного регулятора на основі релейного елемента

Функціональна схема такої системи матиме вигляд, наведений на рис. 8.5, на якій:

- Г – генератор,
- К – котушка електромагніту реле,
- Я – якор електромагніту реле,
- РЕ – контакти реле,

ЛЧ – лінійна частина системи.

Як бачите, котушку і якір реле ми віднесли до лінійної частини системи, а контакти – до релейного елемента, що формує релейну характеристику, наприклад, виду (8.1), графік якої має вигляд, показаний на рис. 8.1, б.

Фізично сигнал $u(t)$ являє собою переміщення рухомої контактної групи реле під дією пружини, натяг якої можна змінювати, а сигнал $z(t)$ – це переміщення тієї ж рухомої контактної групи реле під дією якоря реле, котрий, втягуючись у котушку, переборює дію пружини.

Якщо $z(t) = 0$, то $x(t) = u(t) > x_0$ і одна група контактів реле, яку називають *нормально замкнутою*, замикається, подаючи напругу $+U_0$ на обмотку збудження генератора, викликаючи цим появу $e(t)$.

Коли $x(t) = u(t) - z(t)$ стає меншим x_0 , нормально замкнута контактна група реле розмикається і обнуляє напругу на обмотці збудження генератора, викликаючи зменшення $e(t)$. Зменшення $e(t)$ приводить до зменшення $z(t)$ і настає момент, коли $x(t) = u(t) - z(t)$ стає більшим x_0 , що приводить до замикання нормально замкнутої групи контактів реле.

Оскільки інерційність як механічної, так і електромагнітної частин реле незначна, то частота перемикань контактів реле є високою і різниця $u(t) - z(t)$ стабілізується біля x_0 , а напруга збудження генератора $u(t)$ стабілізується біля $+U_0$.

Перейдемо до математичної моделі описаного вище процесу.

Зв'язок між ЕРС $e(t)$ генератора, яка подається на вхід котушки електромагніту реле, та струмом в котушці, як відомо з теоретичних основ електротехніки, задається диференціальним рівнянням 1-го порядку, що має вигляд

$$T_K \frac{di}{dt} + i = k_K e, \quad (8.24)$$

де $T_K = \frac{L_K}{r_K}$ – електромагнітна стала часу електромагніту, L_K – індуктивність котушки, r_K – активний опір цієї котушки, $k_K = \frac{1}{r_K}$ – коефіцієнт передачі котушки від вхідної напруги до струму в ній.

Зв'язок між струмом $i(t)$ в котушці електромагніта і лінійним переміщенням $z(t)$ якоря електромагніта, як відомо з фізики, задається диференціальним рівнянням 2-го порядку, що має вигляд

$$\ddot{z} + 2\xi\omega_0\dot{z} + \omega_0^2 \cdot z = k_M \cdot i, \quad (8.25)$$

де ξ – коефіцієнт затухання коливань якоря електромагніту, ω_0 – частота власних коливань цього якоря, а k_M – коефіцієнт передачі від струму $i(t)$ в котушці електромагніту до лінійного переміщення $z(t)$ його якоря.

Зв'язок між напругою $y(t)$, що подається на вхід обмотки збудження генератора, та ЕРС $e(t)$ його якоря, як відомо, задається диференціальним рівнянням 1-го порядку, що має вигляд

$$T_\Gamma \frac{de}{dt} + e = k_\Gamma y, \quad (8.26)$$

де $T_\Gamma = \frac{L_{O3\Gamma}}{r_{O3\Gamma}}$ – електромагнітна стала часу обмотки збудження генератора, $L_{O3\Gamma}$ – індуктивність обмотки збудження, $r_{O3\Gamma}$ – активний опір цієї обмотки, $k_\Gamma = \frac{k_\Gamma^*}{r_{O3\Gamma}}$ – коефіцієнт передачі від напруги, що подається на вхід обмотки збудження генератора, а k_Γ^* – коефіцієнт передачі від струму в обмотці збудження до ЕРС якоря генератора, який визначається з характеристики холостого ходу генератора.

Для замикання системи рівнянь, що описують систему вібраційного регулювання ЕРС генератора, функціональна схема якої наведена на рис. 8.5, залишилось лише зв'язати вхідний $x(t)$ та вихідний $y(t)$ сигнали релейного елемента РЕ рівнянням виду

$$y = \Phi^*(x(t), \sigma), \quad (8.27)$$

конкретний вигляд якого залежить від наявності чи відсутності зони нечутливості та гістерезису.

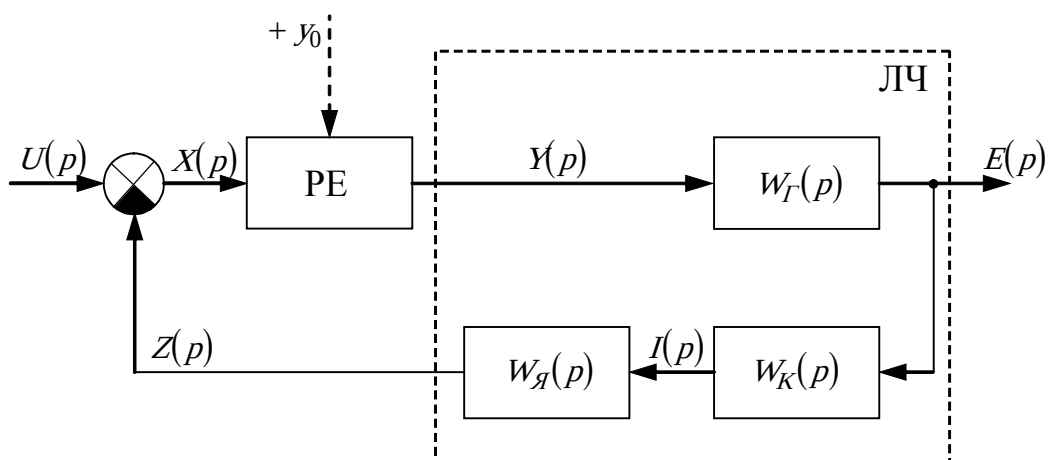


Рисунок 8.6 – Структурна схема нелінійної динамічної системи вібраційного регулювання ЕРС генератора

Перетворюючи рівняння (8.24), (8.25) та (8.26) за Лапласом і переходячи до передаточних функцій, отримаємо на комплексній площині структурну схему системи вібраційного регулювання ЕРС генератора у вигляді, показаному на рис. 8.6, де

$$W_{\Gamma}(p) = \frac{k_{\Gamma}}{T_{\Gamma}p + 1} \quad (8.28)$$

– передаточна функція генератора,

$$W_K = \frac{k_K}{T_K p + 1} \quad (8.29)$$

– передаточна функція котушки електромагніту реле,

$$W_{\text{Я}}(p) = \frac{k_M}{p^2 + 2\xi\omega_0 p + \omega_0^2} \quad (8.30)$$

– передаточна функція якоря електромагніту реле.

Структурна схема, що наведена на рис. 8.6, легко приводиться до загального вигляду, показаного на рис. 8.7, де

$$W_{\text{ЛЧ}}(p) = W_{\Gamma}(p) \cdot W_K(p) \cdot W_{\text{Я}}(p) \quad (8.31)$$

– передаточна функція лінійної частини системи, яка після підстановки у (8.31) виразів (8.28), (8.29), (8.30) набуває вигляду

$$W_{\text{ЛЧ}}(p) = \frac{k_{\Gamma} \cdot k_K \cdot k_M}{(T_{\Gamma}p + 1) \cdot (T_K p + 1) \cdot (p^2 + 2\xi\omega_0 p + \omega_0^2)}. \quad (8.32)$$

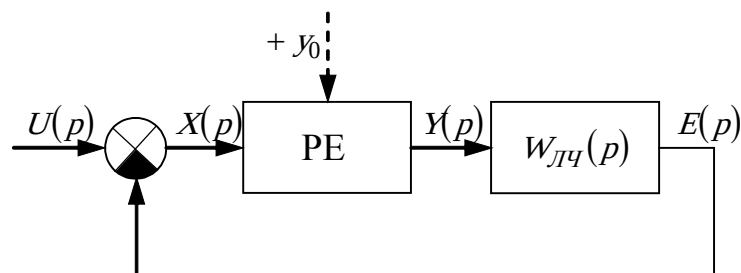


Рисунок 8.7 – Узагальнена структурна схема системи вібраційного регулювання ЕРС генератора

З врахуванням викладеного вище для системи вібраційного регулювання ЕРС генератора матимемо такі математичні моделі:

- для сигналу $x(p)$ на вході РЕ -

$$X(p) = U(p) - \frac{k_{\Gamma} \cdot k_K \cdot k_M}{(T_{\Gamma}p + 1) \cdot (T_K p + 1) \cdot (p^2 + 2\xi\omega_0 p + \omega_0^2)} L\left\{\Phi^*[x(t), \sigma]\right\}, \quad (8.33)$$

- для вихідного сигналу $Z(p)$ лінійної частини системи -

$$Z(p) = \frac{k_I \cdot k_K \cdot k_M}{(T_I p + 1) \cdot (T_K p + 1) \cdot (p^2 + 2\xi\omega_0 p + \omega_0^2)} L\{\Phi^*[(u(t) - z(t)), \sigma]\}. \quad (8.34)$$

Приклад 2. Нехай нелінійною динамічною системою, процеси в якій ми хочемо описати, є система стабілізації кутової швидкості $\omega(t)$ електродвигуна ЕД при змінах у часі моменту технологічного навантаження $M_{TH}(t)$ електропривода, функціональна схема якої має вигляд, наведений на рис. 8.8, на якій $M_D(t)$ – обертальний момент якоря електродвигуна.

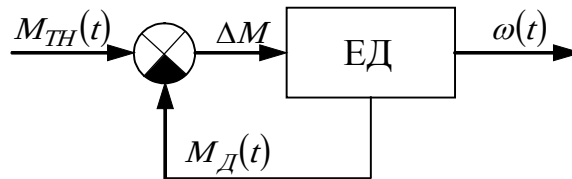


Рисунок 8.8 – Функціональна схема системи стабілізації кутової швидкості електродвигуна при змінах у часі моменту технологічного навантаження електропривода

У цій системі регулювання немає релейного елемента в явному вигляді, але він неявно формується у зв'язку з наявністю так званого «сухого тертя», за рахунок якого створюється момент $\Phi(\omega)$, котрий не дозволяє почати рух в системі до тих пір, поки M_D не стане більшим за $\Phi(\omega)$, і завжди має знак, протилежний знаку M_D . Тобто, у цьому випадку, за рахунок ефекту «сухого тертя», маємо замість функціональної схеми, наведеної на рис. 8.8, функціональну схему, наведену на рис. 8.9, на якій момент $\Phi^*(\omega, \sigma)$, що виникає за рахунок «сухого тертя», описується співвідношенням, що має вигляд

$$\Phi^*(\omega, \sigma) = \phi_0 \cdot \text{sign}(\omega). \quad (8.35)$$

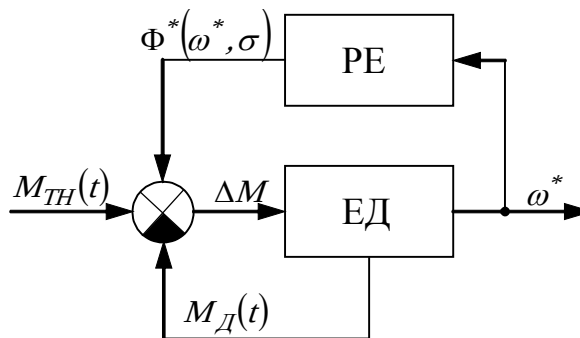


Рисунок 8.9 – Функціональна схема системи стабілізації кутової швидкості електродвигуна при змінах у часі моменту технологічного навантаження та при врахуванні «сухого тертя»

Якби не було «сухого тертя» і в'язкого демпфування, то модель руху електричного двигуна, як відомо, мала б вигляд

$$J \frac{d\omega}{dt} = M_D - M_{TH}, \quad (8.36)$$

де ω – кутова швидкість обертання валу ротора електродвигуна, а J – приведений до цього валу момент інерції махових мас, з ним з'єднаних.

Але при наявності гальмівних складових, обумовлених в'язким демпфуванням та «сухим тертям», для моделювання руху електродвигуна замість лінійного рівняння (8.36) необхідно використовувати більш складне нелінійне рівняння

$$J \cdot \dot{\omega}_{CT} + k_{BD} \cdot \omega_{CT} + \Phi(\omega_{CT}, \sigma) = M_D - M_{TH}, \quad (8.37)$$

в якому k_{BD} – коефіцієнт в'язкого демпфування, а нижній індекс «СТ» біля кутової швидкості ω означає, що розглядається її значення, яке враховує вплив «сухого тертя».

Рівняння (8.37) легко приводиться до виду

$$T_M \cdot \dot{\omega}_{CT} + \omega_{CT} = k_{D\phi} [M_D - M_{TH} - \Phi(\omega_{CT}, \sigma)], \quad (8.38)$$

де $T_M = \frac{J}{k_{BD}}$ – електромеханічна стала часу електродвигуна, а $k_{D\phi} = \frac{1}{k_{BD}}$ – коефіцієнт передачі електродвигуна від моменту до кутової швидкості з врахуванням в'язкого демпфування.

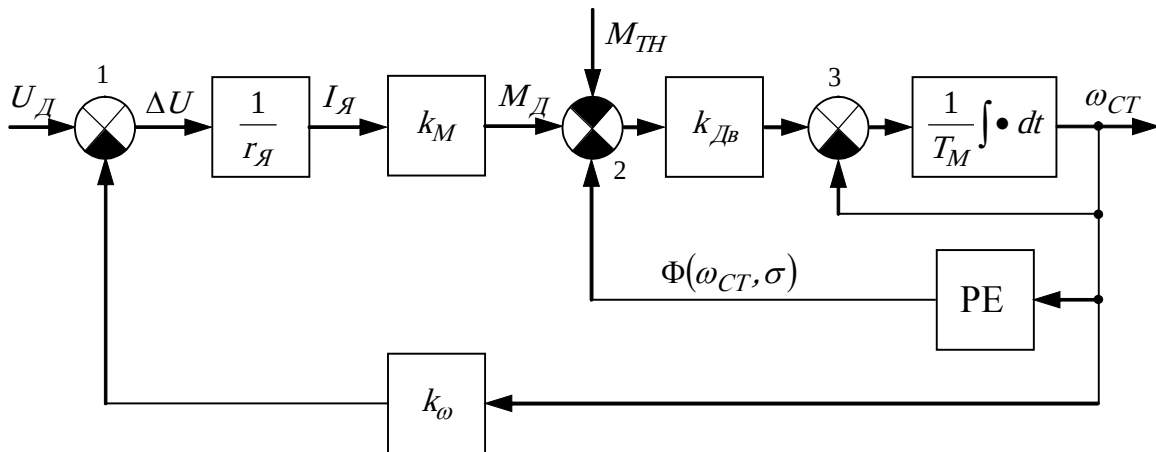


Рисунок 8.10 – Структурна схема в часовій області електричного двигуна, навантаженого моментом M_{TH} , модель якого враховує в'язке демпфування і «сухе тертя»

Легко бачити, що графічний образ рівняння (8.38) має вигляд, наведений на рис. 8.10, в якому враховано те, що

$$M_D = k_M \cdot I_J, \quad (8.39)$$

$$I_{\text{я}} = \frac{u_{\text{д}} - e_{\text{д}}}{r_{\text{я}}}, \quad (8.40)$$

$$e_{\text{д}} = k_{\omega} \cdot \omega, \quad (8.41)$$

де $I_{\text{я}}$ – струм якоря, $e_{\text{д}}$ – ЕРС його обертання, $u_{\text{д}}$ – напруга, що прикладена до його обмотки, $r_{\text{я}}$ – активний опір цієї обмотки, а k_M і k_{ω} – коефіцієнти, значення яких можна знайти у будь-якому підручнику з електропривода.

Суміщаючи на схемі рис. 8.10 1-ий суматор з 2-им і позбавляючись від 3-го суматора та переходячи на комплексну площину, матимемо схему, що зображена на рис. 8.11.

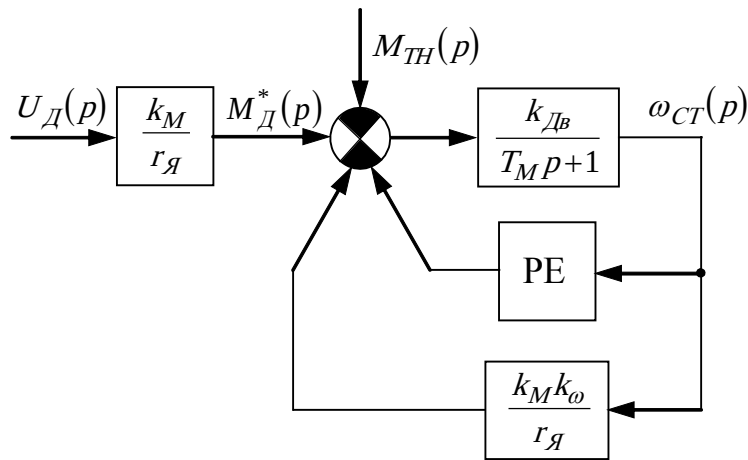


Рисунок 8.11 – Структурна схема на комплексній площині електричного двигуна, навантаженого моментом M_{TH} , модель якого враховує зв'язок демпфування і «сухе тертя»

У свою чергу, пам'ятаючи, що на схемі (рис. 8.11) ланка з передаточною функцією

$$W_{33}(p) = \frac{k_M \cdot k_{\omega}}{r_{\text{я}}} \quad (8.42)$$

задає негативний зворотний зв'язок для ланки з передаточною функцією

$$W_{Дв}(p) = \frac{k_{Дв}}{T_M p + 1}, \quad (8.43)$$

цю схему можна легко привести до схеми, зображеної на рис. 8.12, на якій

$$W_1(p) = \frac{k_M}{r_{\text{я}}}, \quad (8.44)$$

а

$$W_2(p) = \frac{\frac{k_{Дв}}{T_M p + 1}}{1 + \frac{k_M k_\omega}{r_J} \cdot \frac{k_{Дв}}{T_M p + 1}} = \frac{k_2}{T_2 p + 1}, \quad (8.45)$$

де

$$T_2 = T_M \cdot \frac{r_J}{r_J + k_M k_\omega k_{Дв}}, \quad (8.46)$$

$$k_2 = \frac{r_J}{r_J + k_M k_\omega k_{Дв}} \cdot k_{Дв}. \quad (8.47)$$

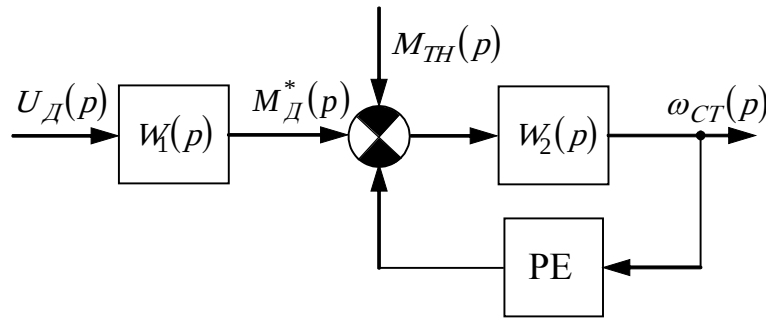


Рисунок 8.12 – Узагальнена структурна схема на комплексній площині електричного двигуна, навантаженого моментом M_{TH} , модель якого враховує в'язке демпфування і «сухе тертя»

А далі вчинимо так. Спочатку запишемо чому дорівнюватиме різниця $M_D^* - M_{TH}$.

Зі схеми (рис. 8.12) бачимо, що

$$\Delta M(p) = M_D^*(p) - M_{TH}(p) = W_1(p) \cdot U_D(p) - M_{TH}(p). \quad (8.48)$$

Очевидно, що аналогом рівняння (8.48) є рівняння

$$\Delta M(p) = L\{\Delta M(t)\}, \quad (8.49)$$

в якому

$$\Delta M(t) = L^{-1}\{W_1(p) \cdot U_D(p) - M_{TH}(p)\} \quad (8.50)$$

є оберненим перетворенням Лапласа виразу, який стоїть в правій частині рівняння (8.48).

Враховуючи співвідношення (8.49) і (8.50), маємо право структурну схему рис. 8.12 подати у вигляді, зображеному на рис. 8.13.

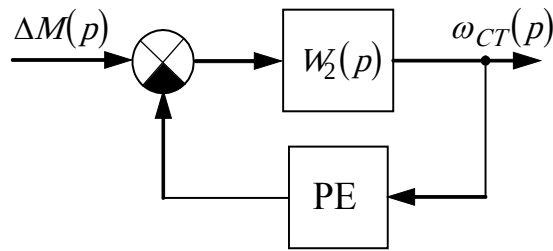


Рисунок 8.13 – Еквівалентна структурна схема на комплексній площині електричного двигуна, навантаженого моментом M_{TH} , модель якого враховує в'язке демпфування і «сухе тертя»

Із схеми рис. 8.13 видно, що ми вже звели початкову структуру нелінійної динамічної системи, графічно заданої на рис. 8.10, до структури, що складається лише з лінійної частини і релейного елемента.

Але для аналізу релейних систем зручніше, щоб суматор був на вході релейного елемента. Тож здійснимо ще одне структурне перетворення схеми, показаної на рис. 8.13, – перенесемо суматор з входу ланки з передаточною функцією $W_2(p)$ на її вихід. В результаті цього переносу отримаємо схему, зображену на рис. 8.14.

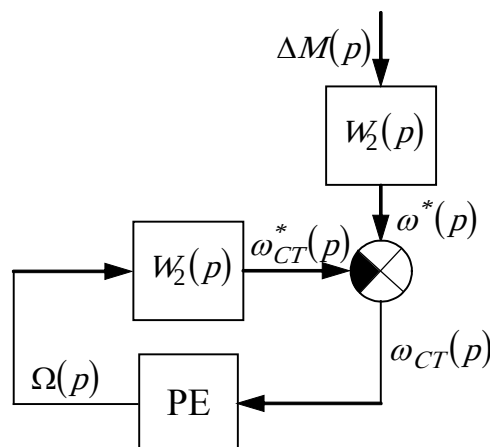


Рисунок 8.14 – Структурна схема, яка є еквівалентною схемі, показаній на рис. 8.13, але з суматором, перенесеним на вхід релейного елемента

На рис. 8.14 ми маємо, як і в схемі на рис. 8.3, класичне для аналізу з'єднання релейного елемента та лінійної частини.

Для цієї схеми є справедливим:

$$\left\{ \begin{array}{l} \omega_{CT}(p) = \omega^*(p) - \omega_{CT}^*(p), \\ \omega^*(p) = W_2(p) \cdot \Delta M(p) = W_2(p) \cdot (W_1(p) \cdot U_D(p) - M_{TH}(p)), \\ \omega^*(t) = L^{-1}\{\omega^*(p)\}, \\ \omega_{CT}^*(p) = W_2(p) \cdot \Omega(p), \\ \omega_{CT}^*(t) = L^{-1}\{\omega_{CT}^*(p)\}, \\ \Omega(p) = L\{\Phi[\omega^*(t) - \omega_{CT}^*(t)]\}. \end{array} \right. \quad (8.51)$$

Система рівнянь (8.51) і є математичною моделлю електричного двигуна, навантаженого моментом M_{TH} , моментом від в'язкого демпфування $k_{ВД} \cdot \omega$ та моментом від «сухого тертя» $\Phi(\omega, \sigma)$, з приведеним до входу релейного елемента суматором.

Як бачимо, синтез цієї моделі зводиться до визначення передаточної функції лінійної частини нелінійної динамічної системи і перерахунку її зовнішніх сигналів до входу релейного елемента.

8.3 Аналіз процесів в нелінійних динамічних системах з релейними елементами

Нехай релейний елемент РЕ (рис. 8.15, а) має характеристику, що зображена на рис. 8.1, а, яка для зручності повторена на рис. 8.15, б.

На рис. 8.15 ми бачимо, що яким би не був вхідний сигнал $x(t)$ (рис. 8.15, в) релейного елемента, його вихідний сигнал $y(t)$ завжди являтиме собою послідовність імпульсів однієї і тієї ж висоти y_0 , знак кожного із яких визначатиметься знаком функції $x(t)$ між двома сусідніми значеннями t_k і t_{k+1} аргументу t , в яких ця функція перетинає вісь абсцис, а протяжність

$$\Delta t_k = t_{k+1} - t_k, \quad k = 0, 1, \dots \quad (8.52)$$

дорівнюватиме відрізку часу між цими сусідніми його значеннями t_k і t_{k+1} .

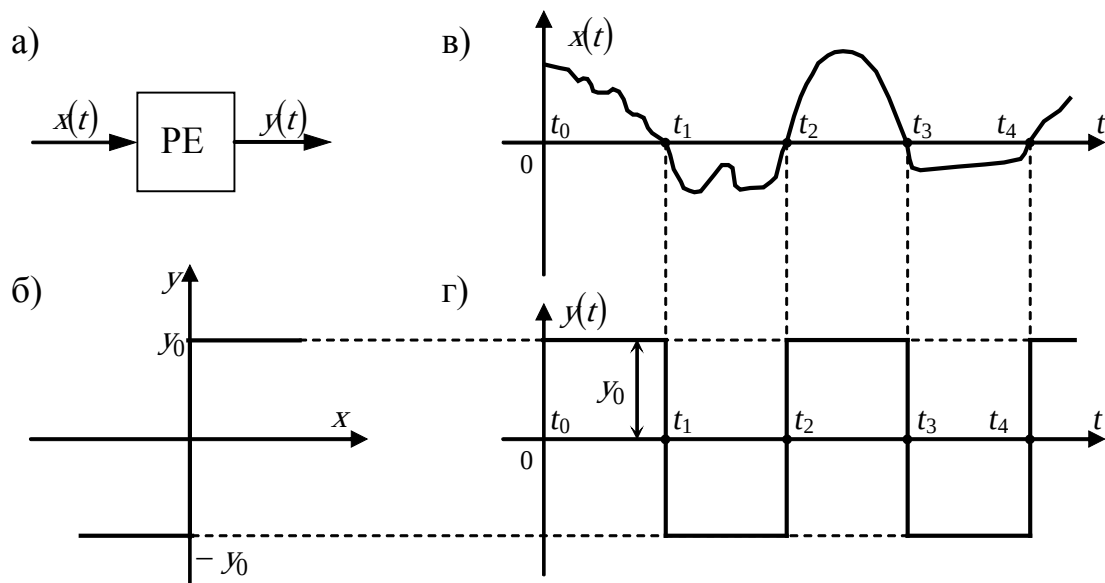


Рисунок 8.15 – Графічна інтерпретація формування вихідного сигналу релейного елемента без зони нечутливості і гістерезису

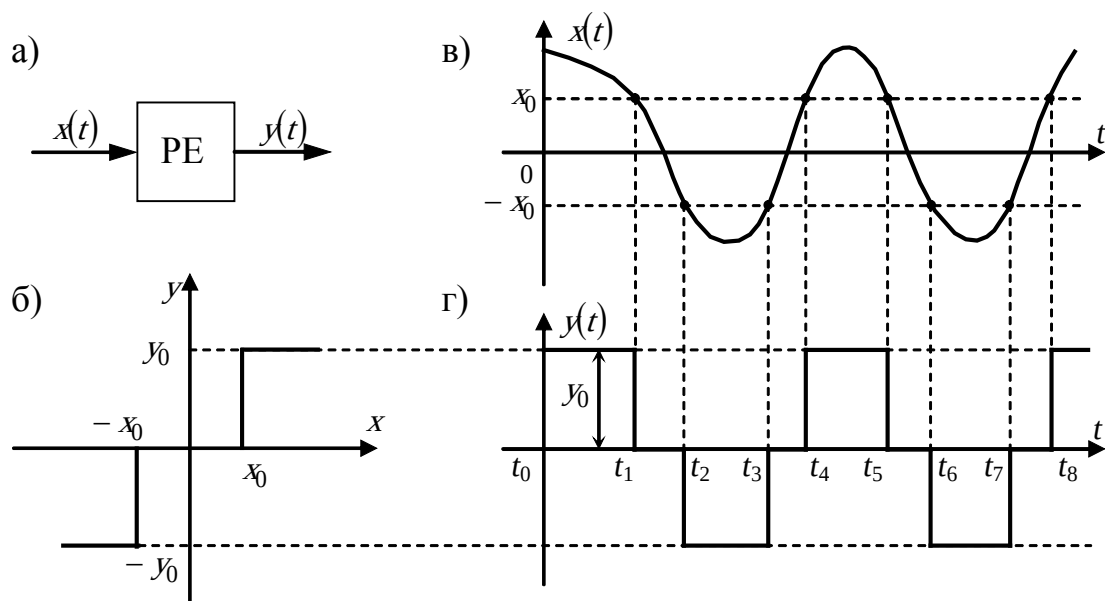


Рисунок 8.16 – Графічна інтерпретація формування вихідного сигналу релейного елемента з зоною нечутливості, але без гістерезису

Якщо ж релейний елемент РЕ (рис. 8.16, а) має характеристику, що зображена на рис. 8.1, б, яка для зручності повторена на рис. 8.16, б, то, очевидно, що його реакцією на вхідний сигнал $x(t)$ (рис. 8.16, в) буде вихідний сигнал $y(t)$, зображений на рис. 8.16, г.

Ми знову маємо знакозмінну послідовність імпульсів однакової висоти y_0 , але протяжність кожного з них буде меншою від значення, що задається

формулою (8.52). А між кожною парою сусідніх імпульсів з'являється проміжок часу, в який вихідний сигнал $y(t)$ релейного елемента дорівнює нулю.

Тепер розглянемо випадок, коли релейний елемент РЕ (рис. 8.17, а) має характеристику, що зображена на рис. 8.1, в, яка для зручності повторена на рис. 8.17, б.

Очевидно, що у цьому випадку реакцією релейного елемента РЕ на вхідний сигнал $x(t)$ (рис. 8.17, в) буде вихідний сигнал $y(t)$, зображений на рис. 8.17, г.

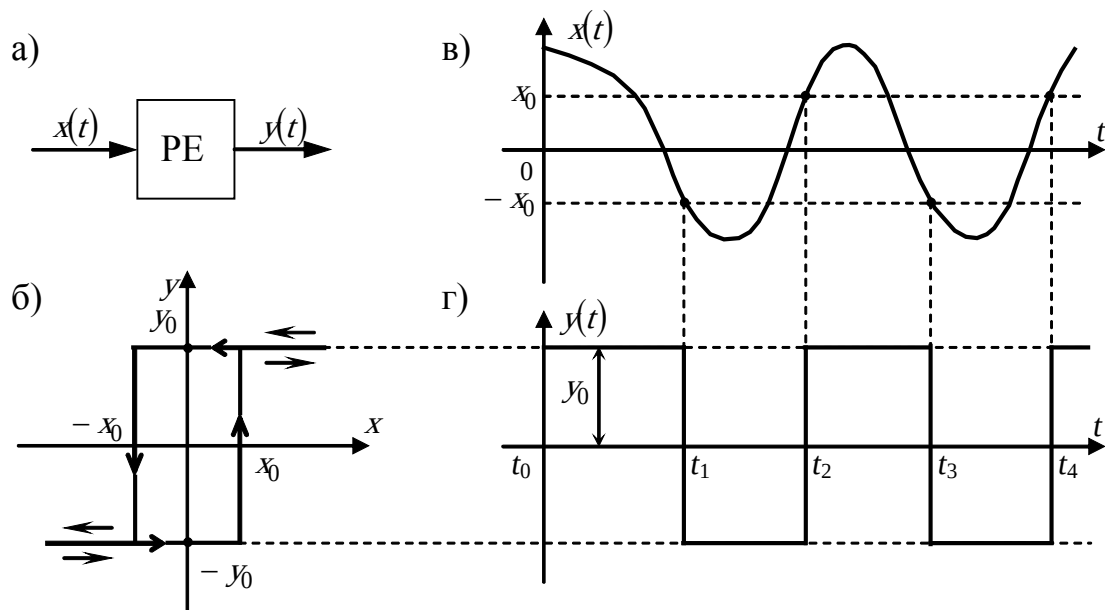


Рисунок 8.17 – Графічна інтерпретація формування вихідного сигналу релейного елемента з гістерезисом, але без зони нечутливості

На відміну від попереднього, у цьому випадку не буде нульових значень вихідного сигналу $y(t)$, але на відміну від випадку, який розглянули найпершим, кожний наступний імпульс буде з'являтися не в момент переходу сигналу $x(t)$ через нуль, а з запізненням, значення якого залежатиме як від форми сигналу $x(t)$, так і від ширини $2x_0$ гістерезисної петлі.

Останнім розглянемо випадок, коли релейний елемент має характеристику, що зображена на рис. 8.1, г, яка для зручності повторена на рис. 8.18, б.

Очевидно, що і в цьому випадку вихідний сигнал $y(t)$ релейного елемента матиме вигляд знакозмінної послідовності імпульсів однакової висоти, але, як і у випадку, розглянутому другим, тут матимуть місце проміжки часу з нульовим значенням вихідного сигналу $y(t)$ між сусідніми імпульсами, і, одночасно, як і у випадку, розглянутому третім, тут матиме місце запізнення появи кожного наступного імпульсу.

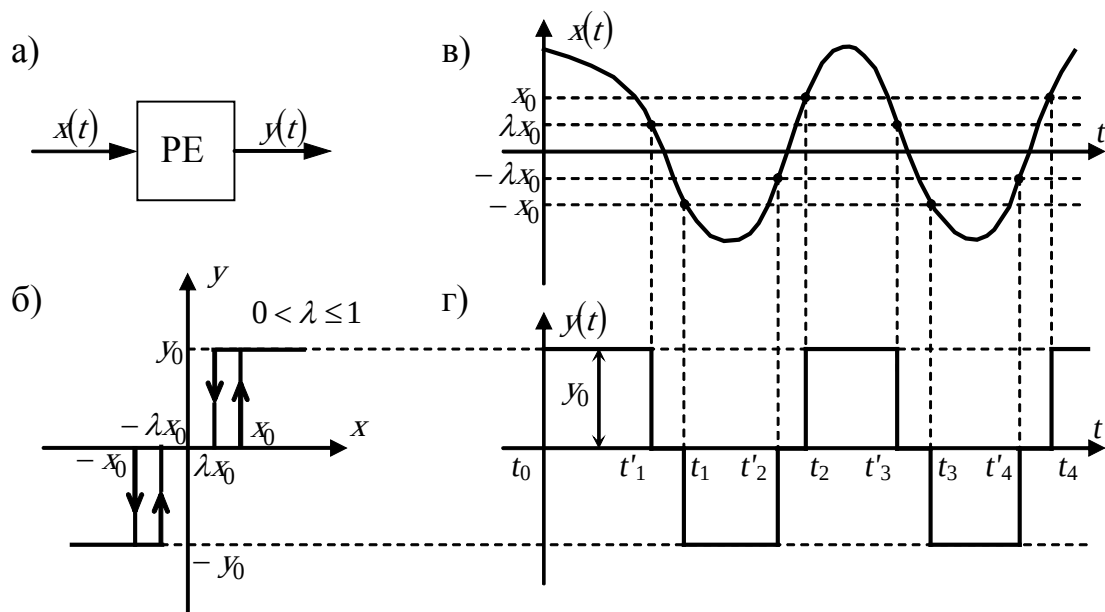


Рисунок 8.18 – Графічна інтерпретація формування вихідного сигналу релейного елемента з зоною нечутливості і гістерезисом

Усі розглянуті випадки об'єднує одне – релейний елемент створює на вході лінійної частини нелінійної динамічної системи знакозмінну послідовність імпульсів однакової висоти. А це, в свою чергу, означає, що вихідний сигнал лінійної частини нелінійної системи, який одночасно є і складовою вхідного сигналу релейного елемента (рис. 8.3), можна знайти як суму реакцій лінійної частини нелінійної динамічної системи на кожний із імпульсів вхідної імпульсної знакозмінної послідовності.

Тож, як бачимо, задача аналізу суттєво нелінійної динамічної системи з релейним елементом зводиться до:

1) пошуку моментів t_k , $k = 0, 1, \dots$ переключення релейного елемента під дією вхідного сигналу $x(t)$;

2) перетворення за Лапласом суми імпульсів знакозмінної послідовності, яка з виходу РЕ надходить на вхід лінійної частини системи;

3) оберненого перетворення за Лапласом вихідного сигналу лінійної частини системи, заданого в залежності від форми характеристики релейного елемента одним із співвідношень (8.21), (8.22) або (8.23).

Для конкретизації цього поки що загального алгоритму розв'язання задачі аналізу суттєво нелінійних динамічних систем з релейним елементом припустимо спочатку, що нам уже відомі моменти t_k , $k = 0, 1, \dots$ переключення релейного елемента, що має характеристику $y = \Phi(x)$, показану рис. 8.15, б.

Тоді модель імпульсу висотою $(-1)^k \cdot Y_0$ і протяжністю Δt_k (8.52) у часовій області матиме вигляд:

$$y\left(t^{(k)}\right)=(-1)^k \cdot y_0 \cdot\left(1\left(t^{(k)}-t_k\right)-1\left(t^{(k)}-t_{k+1}\right)\right),$$

$$t_k < t^{(k)} \leq t_{k+1}, \quad (8.53)$$

де $1(t)$ – це одинична східчаста функція, для якої справедливо

$$1(t)=\begin{cases} 1, & t \geq 0, \\ 0, & t < 0. \end{cases} \quad (8.54)$$

Перетворюючи за Лапласом вираз (8.53), отримаємо

$$L\left\{y\left(t^{(k)}\right)\right\}=\int_0^{\infty}\left[(-1)^k \cdot y_0 \cdot\left(1\left(t^{(k)}-t_k\right)-1\left(t^{(k)}-t_{k+1}\right)\right)\right] \cdot e^{-p t^{(k)}} d t^{(k)}=$$

$$=(-1)^k \cdot y_0 \cdot\left[\int_0^{\infty} 1\left(t^{(k)}-t_k\right) e^{-p t^{(k)}} d t^{(k)}-\int_0^{\infty} 1\left(t^{(k)}-t_{k+1}\right) e^{-p t^{(k)}} d t^{(k)}\right],$$

$$t_k < t^{(k)} \leq t_{k+1}. \quad (8.55)$$

Роблячи заміну змінних у першому інтегралі $\theta=t^{(k)}-t_k$, а у другому інтегралі $\tau=t^{(k)}-t_{k+1}$, із (8.55) отримаємо

$$L\left\{y\left(t^{(k)}\right)\right\}=(-1)^k \cdot y_0 \cdot\left[\int_{-t_k}^{\infty} 1(\theta) \cdot e^{-p(\theta+t_k)} d \theta-\int_{-t_{k+1}}^{\infty} 1(\tau) \cdot e^{-p(\tau+t_{k+1})} d \tau\right]=$$

$$=(-1)^k \cdot y_0 \cdot\left[e^{-p t_k} \cdot \int_0^{\infty} 1(\theta) \cdot e^{-p \theta} d \theta-e^{-p t_{k+1}} \cdot \int_0^{\infty} 1(\tau) \cdot e^{-p \tau} d \tau\right]=$$

$$=\frac{(-1)^k \cdot y_0}{p}\left[e^{-p t_k}-e^{-p t_{k+1}}\right],$$

$$t_k < t^{(k)} \leq t_{k+1}. \quad (8.56)$$

Узагальнюючи результат (8.56) на суму із n імпульсів, що слідують один за одним щільно, тобто без проміжків, матимемо:

$$L\left\{\sum_{k=0}^{n-1} y\left(t^{(k)}\right)\right\}=\frac{y_0}{p} \sum_{k=0}^{n-1}(-1)^k \cdot\left[e^{-p t_k}-e^{-p t_{k+1}}\right],$$

$$t_k < t^{(k)} \leq t_{k+1}. \quad (8.57)$$

Співвідношення, очевидно, буде справедливим для зображення за Лапласом вихідного сигналу релейного елемента з характеристикою, показаною на рис. 8.15, б. Тобто справедливою є тотожність

$$L\{\Phi(x(t))\} = \frac{y_0}{p} \sum_{k=0}^{n-1} (-1)^k \cdot \left[e^{-pt_k} - e^{-pt_{k+1}} \right]. \quad (8.58)$$

Пам'ятаючи, що

$$x(t) = u(t) - z(t), \quad (8.59)$$

що впливає із структурної схеми рис. 8.3, із співвідношень (8.21) і (8.58) матимемо:

$$Z(p) = W(p) \cdot \frac{y_0}{p} \sum_{k=0}^{n-1} (-1)^k \cdot \left[e^{-pt_k} - e^{-pt_{k+1}} \right], \quad (8.60)$$

або

$$Z(p) = y_0 \sum_{k=0}^{n-1} (-1)^k \cdot \left[\frac{W(p)}{p} e^{-pt_k} - \frac{W(p)}{p} e^{-pt_{k+1}} \right]. \quad (8.61)$$

У першій частині цього навчального посібника було показано, що зображення за Лапласом перехідної характеристики $h(t)$ системи має вигляд:

$$L\{h(t)\} = \int_0^{\infty} h(t) \cdot e^{-pt} dt = \frac{W(p)}{p}. \quad (8.62)$$

Знайдемо, чому дорівнюватиме зображення за Лапласом зміщеної на час t_k перехідної характеристики цієї ж системи.

З означення –

$$L\{h(t - t_k)\} = \int_0^{\infty} h(t - t_k) \cdot e^{-pt} dt. \quad (8.63)$$

Здійснимо у (8.63) заміну змінної. Нехай

$$t - t_k = \theta, \quad (8.64)$$

Тоді:

$$\begin{cases} dt = d\theta, \\ t = 0 \rightarrow \theta = -t_k. \end{cases} \quad (8.65)$$

З врахуванням (8.64) і (8.65) з (8.63) отримаємо:

$$\begin{aligned} L\{h(t - t_k)\} &= \int_{-t_k}^{\infty} h(\theta) \cdot e^{-p(\theta + t_k)} d\theta = \\ &= e^{-pt_k} \cdot \int_0^{\infty} h(\theta) \cdot e^{-p\theta} d\theta = \frac{W(p)}{p} e^{-pt_k}. \end{aligned} \quad (8.66)$$

Отримавши співвідношення (8.66), отримуємо водночас і право записати його з використанням оберненого перетворення Лапласа у такому вигляді:

$$h(t - t_k) = L^{-1} \left\{ \frac{W(p)}{p} e^{-pt_k} \right\}. \quad (8.67)$$

Якщо маємо зображення за Лапласом вихідного сигналу $Z(p)$, то, за означенням, його оригіналом у часовій області буде

$$z(t) = L^{-1} \{ Z(p) \}. \quad (8.68)$$

Підставляючи у (8.68) вираз (8.61), матимемо:

$$\begin{aligned} z(t) &= L^{-1} \left\{ y_0 \sum_{k=0}^{n-1} (-1)^k \cdot \left[\frac{W(p)}{p} e^{-pt_k} - \frac{W(p)}{p} e^{-pt_{k+1}} \right] \right\} = \\ &= y_0 \sum_{k=0}^{n-1} (-1)^k \cdot \left[L^{-1} \left\{ \frac{W(p)}{p} e^{-pt_k} \right\} - L^{-1} \left\{ \frac{W(p)}{p} e^{-pt_{k+1}} \right\} \right]. \end{aligned} \quad (8.69)$$

$t_k < t \leq t_{k+1}.$

З врахуванням (8.67) співвідношення (8.69) можна переписати так:

$$\begin{aligned} z(t) &= y_0 \cdot \sum_{k=0}^{n-1} (-1)^k \cdot [h(t - t_k) - h(t - t_{k+1})], \\ &\text{для } t_k < t \leq t_{k+1}, \end{aligned} \quad (8.70)$$

або

$$\begin{aligned} z(t) &= y_0 \cdot \sum_{k=0}^{n-1} (-1)^k \cdot [h(t - t_k) - h(t - t_{k+1})] + \\ &\quad + y_0 \cdot (-1)^n \cdot [h(t - t_n) - h(t - t_{n+1})] = \\ &= y_0 \cdot \sum_{k=0}^{n-1} (-1)^k \cdot [h(t - t_k) - h(t - t_{k+1})] + y_0 \cdot (-1)^n \cdot h(t - t_n), \end{aligned}$$

$\text{для } t_n < t \leq t_{n+1}.$

(8.71)

Враховуючи, що

$$t_0 = 0 \quad (8.72)$$

- це впливає з рис. 8.15, рівняння (8.71) можна переписати і так:

-

$$z(t) = y_0 \cdot \left[h(t) + 2 \sum_{k=1}^n (-1)^k \cdot h(t - t_k) \right],$$

для $t_n < t \leq t_{n+1}$.

(8.73)

Для наочності розпишемо вираз (8.73) для трьох значень n .
Нехай $n = 0$, тоді

$$z(t) = y_0 \cdot h(t),$$

$t_0 < t \leq t_1$.

(8.74)

Нехай $n = 1$, тоді:

$$z(t) = y_0 \cdot h(t) - 2y_0 \cdot h(t - t_1),$$

$t_1 < t \leq t_2$.

(8.75)

Нехай $n = 2$, тоді:

$$z(t) = y_0 \cdot h(t) - 2y_0 \cdot h(t - t_1) + 2y_0 \cdot h(t - t_2),$$

$t_2 < t \leq t_3$.

(8.76)

А знаючи вихідний сигнал лінійної частини системи $z(t)$, при заданому керувальному сигналі $u(t)$, за допомогою виразу (8.59) легко знаходимо значення $x(t)$ сигналу на вході релейного елемента РЕ (рис. 8.3) для будь-якого моменту часу t .

Усі викладки, що присвячені отриманню математичних моделей, необхідних для аналізу нелінійних динамічних систем з релейними елементами, зроблені за умови, що відомими є моменти переключення $t_k, k = 0, 1, \dots$ релейного елемента.

Тож, для побудови замкненого алгоритму аналізу даного класу систем необхідно цей алгоритм доповнити способом визначення моментів переключення релейного елемента.

Із рис. 8.15 ми бачимо, що моменти переключення $t_k, k = 0, 1, \dots$ релейного елемента, на вхід якого надходить сигнал $x(t)$, є коренями рівняння

$$x(t_k) = 0,$$
(8.77)

яке після підстановки в (8.77) виразу (8.59) набуває вигляду

$$u(t_k) = z(t_k).$$
(8.78)

В рівнянні (8.78), нагадаємо, $u(t)$ – керувальний сигнал, що подається на вхід системи, а $z(t)$ – вихідний сигнал її лінійної частини.

Із виразів (8.78) і (8.74) легко бачити, що момент t_1 першого переключення РЕ після включення системи в роботу шляхом подачі керувального сигналу $u(t)$ знаходиться з рівняння

$$u(t_1) = y_0 \cdot h(t_1), \quad (8.79)$$

в якому $h(t)$ – попередньо знайдена експериментально чи оберненим перетворенням Лапласа виразу (8.62) перехідна характеристика лінійної частини системи, що аналізується.

Оскільки в проміжку часу $t_1 < t \leq t_2$ для сигналу $z(t)$ справедливим є рівняння (8.75), то момент переключення t_2 можна знайти з рівняння

$$u(t_2) = y_0 \cdot h(t_2) - 2y_0 \cdot h(t_2 - t_1), \quad (8.80)$$

яке отримуємо підстановкою (8.75) у (8.78).

Аналогічно, момент переключення t_3 знайдемо з рівняння

$$u(t_3) = y_0 \cdot h(t_3) - 2y_0 \cdot h(t_3 - t_1) + 2y_0 \cdot h(t_3 - t_2), \quad (8.81)$$

яке отримуємо підстановкою виразу (8.76) у (8.78).

Узагальнюючи, можна стверджувати, що будь-який $(n+1)$ -ий момент переключення РЕ може бути знайдений з рівняння

$$u(t_{n+1}) = y_0 \cdot \left[h(t_{n+1}) + 2 \sum_{k=1}^n (-1)^k \cdot h(t_{n+1} - t_k) \right],$$

$$n = 0, 1, \dots, \quad (8.82)$$

яке отримуємо, підставляючи вираз (8.73) у (8.78).

Із рис. 8.15 ми бачимо, що вхідний сигнал $x(t)$ релейного елемента в момент t_1 першого переключення має похідну $\dot{x}(t_1)$ з від'ємним знаком, а в момент t_2 другого переключення – з додатним знаком. І далі знаки у похідній чергуються.

В загальному вигляді це чергування знаку похідної $\dot{x}(t_k)$ сигналу $x(t)$ в точках переключення можна задати умовою

$$\dot{x}(t_k) \cdot (-1)^k > 0,$$

$$k = 1, 2, 3, \dots, \quad (8.83)$$

або (з врахуванням виразу (8.59)) умовою

$$[\dot{u}(t_k) - \dot{z}(t_k)] \cdot (-1)^k > 0,$$

$$k = 1, 2, \dots, \quad (8.84)$$

яку називають *умовою належних напрямків переключень*.

Важливе зауваження: аби не пропустити якийсь момент переключення, рівняння (8.82) необхідно розв'язувати методом послідовних наближень з достатньо малим кроком ітерації.

Тепер розглянемо питання стосовно того, яких змін зазнає алгоритм аналізу релейних динамічних систем, якщо релейний елемент матиме характеристику, відмінну від показаної на рис. 8.15, б.

Почнемо розгляд з систем з релейною характеристикою, наведеною на рис. 8.17, б.

Легко бачити, що у цьому випадку знакозмінна імпульсна послідовність $y(t)$ на виході релейного елемента РЕ має такий самий характер, як і y вже розглянутому випадку. Відмінність лише в рівнянні, за допомогою якого знаходяться моменти переключень $t_k, k = 1, 2, \dots$

Із рис. 8.17, в видно, що рівняння для визначення моментів переключень $t_k, k = 1, 2, \dots$ у цьому випадку матиме вигляд

$$\begin{aligned} x(t_k) &= (-1)^k \cdot x_0, \\ k &= 1, 2, \dots, \end{aligned} \quad (8.85)$$

або (з врахуванням (8.59))

$$u(t_k) - z(t_k) = (-1)^k \cdot x_0, \quad (8.86)$$

що еквівалентно

$$\begin{aligned} u(t_k) - (-1)^k \cdot x_0 &= z(t_k), \\ k &= 1, 2, \dots \end{aligned} \quad (8.87)$$

Із того ж рис. 8.17 ми бачимо, що умова належних напрямків переключень залишається такою ж, як і в попередньому випадку і задається тими ж виразами (8.83), (8.84).

Що ж стосується сигналу $z(t)$ на виході лінійної частини релейної нелінійної системи даного класу, то для нього є справедливими ті ж вирази (8.73) – (8.76), що і для того випадку, для якого ці вирази отримані і які зображено на рис. 8.15.

Тепер перейдемо до аналізу релейних динамічних систем з релейними характеристиками, показаними на рис. 8.16, б і рис. 8.18, б.

Легко бачити, що знакозмінна імпульсна послідовність $y(t)$ на виході релейного елемента в обох цих випадках має один і той же характер, що не дивно, оскільки релейну характеристику, яка наведена на рис. 8.16, б, можна отримати з релейної характеристики, що наведена на рис. 8.18, б, при $\lambda = 1$.

Тож достатньо розглянути випадок, показаний на рис. 8.18. Отже, нехай $t_1, t_2, t_3, \dots$ – це моменти переключення, які виникають у моменти переходу сигналу $x(t)$ через пороги $\pm x_0$, а $t'_1, t'_2, t'_3, \dots$ – через пороги $\pm \lambda x_0$.

Із рис. 8.18 ми бачимо, що усі ці моменти є коренями рівнянь:

$$x(t_k) = (-1)^k \cdot x_0, \quad (8.88)$$

$$\begin{aligned} x(t'_k) &= (-1)^{k+1} \cdot \lambda x_0, \\ k &= 1, 2, \dots, \end{aligned} \quad (8.89)$$

які за допомогою виразу (8.59) трансформуються у рівняння:

$$u(t_k) - (-1)^k \cdot x_0 = z(t_k), \quad (8.90)$$

$$\begin{aligned} u(t'_k) + (-1)^k \cdot \lambda_0 x_0 &= z(t'_k), \\ k &= 1, 2, \dots \end{aligned} \quad (8.91)$$

Очевидно (дивись рис. 8.18, в), що умова належних напрямків переключень для обох цих випадків буде задаватись однаковими за структурою виразами, аналогічними (8.83) і (8.84), а саме:

$$\dot{x}(t_k) \cdot (-1)^k > 0, \quad (8.92)$$

$$\begin{aligned} \dot{x}(t'_k) \cdot (-1)^k &> 0, \\ k &= 1, 2, \dots, \end{aligned} \quad (8.93)$$

або (з врахуванням виразу (8.59)):

$$[\dot{u}(t_k) - \dot{z}(t_k)] \cdot (-1)^k > 0, \quad (8.94)$$

$$\begin{aligned} [\dot{u}(t'_k) - \dot{z}(t'_k)] \cdot (-1)^k &> 0, \\ k &= 1, 2, \dots \end{aligned} \quad (8.95)$$

Дивлячись на рівняння (8.90), (8.91), бачимо, що при заданих $u(t)$, x_0 , λ для їх розв'язання відносно t_k і t'_k , $k = 1, 2, \dots$ необхідно знати вихідний сигнал $z(t)$ лінійної частини (ЛЧ) релейної динамічної системи.

Перш ніж перейти до виведення розрахункових співвідношень для сигналу $z(t)$ звертаємо увагу на те, що він формується лише за рахунок впливу імпульсів висотою $(-1)^k \cdot y_0$ між моментами t_k і t'_{k+1} , оскільки між моментами t'_{k+1} і t_{k+1} вхідний сигнал $y(t)$ дорівнює нулю.

Як знайти реалізацію $z(t)$ ЛЧ системи на таку послідовність імпульсів уже відомо (дивись вирази (8.53) – (8.71), в яких усюди t_{k+1} необхідно замінити на t'_{k+1}). Тобто маємо право записати:

$$z(t) = y_0 \cdot \sum_{k=0}^{n-1} (-1)^k \cdot [h(t-t_k) - h(t-t'_{k+1})] + y_0 \cdot (-1)^n \cdot h(t-t_n),$$

$$t_n < t \leq t'_{n+1}. \quad (8.96)$$

А для проміжку часу $t'_{n+1} < t \leq t_{n+1}$ рівняння (8.96) набуває вигляду

$$z(t) = y_0 \cdot \sum_{k=0}^n (-1)^k \cdot [h(t-t_k) - h(t-t'_{k+1})],$$

$$t'_{n+1} < t \leq t_{n+1}. \quad (8.97)$$

Пам'ятаючи, що $t_0 = 0$, розкриваємо (8.96), (8.97) для трьох значень n .

Нехай $n = 0$, тоді

для $t_0 < t \leq t'_1$:

$$z(t) = y_0 \cdot h(t), \quad (8.98)$$

для $t'_1 < t \leq t_1$:

$$z(t) = y_0 \cdot h(t) - y_0 \cdot h(t-t'_1). \quad (8.99)$$

Нехай $n = 1$, тоді

для $t_1 < t \leq t'_2$:

$$z(t) = y_0 \cdot h(t) - y_0 \cdot h(t-t'_1) - y_0 \cdot h(t-t_1), \quad (8.100)$$

для $t'_2 < t \leq t_2$:

$$z(t) = y_0 \cdot h(t) - y_0 \cdot h(t-t'_1) - y_0 \cdot h(t-t_1) + y_0 \cdot h(t-t'_2). \quad (8.101)$$

Нехай $n = 2$, тоді

для $t_2 < t \leq t'_3$:

$$z(t) = y_0 \cdot h(t) - y_0 \cdot h(t-t'_1) - y_0 \cdot h(t-t_1) +$$

$$+ y_0 \cdot h(t-t'_2) + y_0 \cdot h(t-t_2), \quad (8.102)$$

для $t'_3 < t \leq t_3$:

$$z(t) = y_0 \cdot h(t) - y_0 \cdot h(t-t'_1) - y_0 \cdot h(t-t_1) +$$

$$+ y_0 \cdot h(t-t'_2) + y_0 \cdot h(t-t_2) - y_0 \cdot h(t-t'_3). \quad (8.103)$$

Для знаходження моменту переключення t'_1 необхідно у вираз (8.91) підставити значення $z(t)$ із (8.98) і розв'язати відносно t'_1 отримане рівняння

$$u(t'_1) - \lambda_0 x_0 = y_0 \cdot h(t'_1). \quad (8.104)$$

Аналогічно, для знаходження моменту переключення t_1 необхідно у вираз (8.90) підставити значення $z(t)$ із (8.99) і розв'язати відносно t_1 отримане рівняння

$$u(t_1) - x_0 = y_0 \cdot h(t_1) - y_0 \cdot h(t_1 - t'_1). \quad (8.105)$$

Для знаходження наступних моментів переключень необхідно рекурентно продовжити цей процес отримання і розв'язання рівнянь спочатку стосовно t'_k , а потім стосовно t_k , починаючи з $k = 2$.

Очевидно, що всі отримані в цьому випадку співвідношення будуть справедливими і при $\lambda = 1$. А саме при такому значенні λ релейний елемент, зображений на рис. 8.18, а, з релейною характеристикою, наведеною на рис. 8.18, б, перетворюється на релейний елемент, зображений на рис. 8.16, а, з релейною характеристикою, наведеною на рис. 8.16, б.

Тож, для аналізу релейних динамічних систем з релейною характеристикою, зображеною на рис. 8.18, б, необхідно використовувати всі отримані вище математичні моделі, починаючи з (8.88), поклавши $\lambda = 1$ в тих із них, які цей параметр мають у своїй структурі.

На завершення розділу зауважимо, що аналіз релейних динамічних систем з несиметричними релейними характеристиками, наприклад, такими, як показано на рис. 8.19, майже нічим не відрізняється від вищевикладеного.

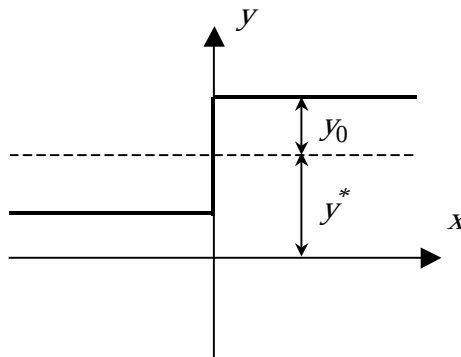


Рисунок 8.19 – Приклад несиметричної релейної характеристики

Єдине, чим доповнюється алгоритм аналізу таких систем, є те, що у вихідному сигналі $z(t)$ лінійної частини системи з'являється додатковий член

$$z_d(t) = y_0 \cdot y^* \cdot h(t), \quad (8.106)$$

який характеризує реакцію лінійної частини системи з перехідною характеристикою $h(t)$ на усталений вплив y^* релейного елемента, характеристика якого центрована цим значенням. Тобто у цьому випадку для сигналу $z(t)$ замість моделей (8.70) і (8.71) необхідно використовувати моделі, відповідно:

$$z(t) = y_0 \cdot \left\{ y^* \cdot h(t) + \sum_{k=0}^{n-1} (-1)^k \cdot [h(t - t_k) - h(t - t_{k+1})] \right\},$$

$$t_k < t \leq t_{k+1}; \quad (8.107)$$

$$z(t) = y_0 \cdot \left\{ y^* \cdot h(t) + \sum_{k=0}^{n-1} (-1)^k \cdot [h(t - t_k) - h(t - t_{k+1})] + (-1)^n \cdot h(t - t_n) \right\},$$

$$t_n < t \leq t_{n+1}. \quad (8.108)$$

Аналогічні поправки до отриманих моделей сигналу $z(t)$ будуть мати місце і при несиметризації інших із розглянутих симетричних релейних характеристик.

8.4 Завдання для самоперевірки

Що собою являють релейні динамічні системи і де вони знаходять застосування? Наведіть приклади.

Який вигляд має релейна характеристика без зони нечутливості і гістерезису? Як виглядає її математична модель?

Який вигляд має релейна характеристика з зоною нечутливості, але без гістерезису? Як виглядає її математична модель?

Який вигляд має релейна характеристика без зони нечутливості, але з гістерезисом? Як виглядає її математична модель?

Який вигляд має релейна характеристика з зоною нечутливості і гістерезисом? Як виглядає її математична модель?

Які дві характерні особливості релейних динамічних систем, пов'язані з вихідним сигналом їхнього релейного елемента, ви знаєте?

Як виглядають функціональна і структурна схеми релейної динамічної системи, придатні для аналізу процесів в ній?

Отримайте в загальному вигляді математичну модель вхідного сигналу релейного елемента релейної динамічної системи.

Отримайте в загальному вигляді математичну модель сигналу на виході лінійної частини релейної динамічної системи.

Продемонструйте процес отримання математичної моделі релейної динамічної системи на прикладі системи вібраційного регулювання електрорушійної сили генератора.

Продемонструйте процес отримання математичної моделі релейної динамічної системи на прикладі системи стабілізації кутової швидкості електродвигуна електропривода.

Побудуйте графіки вхідного та вихідного сигналів релейного елемента:

- без зони нечутливості і гістерезису;
- з зоною нечутливості, але без гістерезису;
- з гістерезисом, але без зони нечутливості;
- з зоною нечутливості і гістерезисом.

В чому суть аналізу релейної динамічної системи? Що треба знати?

Виведіть математичні співвідношення, за допомогою яких можна здійснити аналіз релейної динамічної системи, що має релейну характеристику без зони нечутливості і без гістерезису.

З яких рівнянь знаходяться моменти переключень релейного елемента?

Що собою являє умова належних напрямків переключень?

В чому відмінність аналізу релейних динамічних систем з релейними елементами без зони нечутливості і без гістерезису та без зони нечутливості, але з гістерезисом?

Яка особливість аналізу релейних динамічних систем з релейним елементом, що має і зону нечутливості, і гістерезис?

Як можна використати математичні моделі, що розроблені для випадку релейної характеристики з зоною нечутливості і гістерезисом, для аналізу релейних динамічних систем, що мають релейну характеристику з зоною нечутливості, але без гістерезису?

Продемонструйте знання алгоритму визначення моментів переключень релейного елемента, визначивши декілька перших моментів.

Як аналізувати релейні динамічні системи з несиметричними релейними характеристиками? Наведіть приклад.

ПІДСУМКИ

У першій частині навчального посібника показано, як побудувати математичні моделі лінійних детермінованих неперервних та дискретних динамічних систем із зосередженими параметрами.

Розкриті особливості синтезу математичних моделей у часовому просторі, на комплексній площині, у частотній області та у просторі змінних стану.

Моделі синтезуються у класі диференціальних та різницевих рівнянь, передаточних функцій та частотних характеристик із використанням як прямих алгоритмів обробки інформації, так і оптимальних.

Обґрунтовано і доведено до розрахункових співвідношень і алгоритму застосування Фур'є-інтегральний метод ідентифікації динамічних систем, який є єдиним методом ідентифікації, що дозволяє визначити одночасно і за одним алгоритмом як оптимальні значення параметрів математичної моделі, так і її оптимальну структуру.

У другій частині навчального посібника показано, як побудувати математичні моделі лінійних динамічних системах із зосередженими параметрами, в яких має місце високий рівень зовнішніх завад та внутрішніх шумів, що змушує відносити їх до класу стохастичних систем.

Розкрито особливості синтезу математичних моделей лінійних динамічних систем на основі кореляційної теорії стаціонарних випадкових процесів, для яких виконується умова ергодичності.

Показано, як будувати алгоритми ідентифікації стохастичних систем на основі Фур'є-інтегрального методу.

Запропоновано моделі часових рядів, за допомогою яких можна здійснювати прогнозування розвитку динамічних процесів, що належать до класу стохастичних.

У третій частині навчального посібника показано, як будувати математичні моделі нелінійних динамічних систем взагалі і релейних систем зокрема.

Викладено теорію і алгоритм застосування Фур'є-інтегрального методу для ідентифікації нелінійних динамічних систем з аналітичними нелінійностями.

Наведено алгоритми аналізу процесів в релейних динамічних системах з використанням перетворення за Лапласом характеристик їх лінійних частин.

Здійснено аналіз особливостей ідентифікації лінійних динамічних систем з розподіленими параметрами і синтезована найбільш загальна математична модель цього класу систем.

Побудовані також математичні моделі таких об'єктів з розподіленими параметрами, як довгі електричні лінії, кабелі зв'язку, гідравлічні та пневматичні трубопроводи, теплопроводи, троси, лінії затримки, конвеєри і хімічні реактори. Наведено алгоритми їх аналізу і як самодостатніх об'єктів, і як структурних ланок систем автоматичного регулювання в умовах дії як детермінованих, так і стохастичних вхідних сигналів.

Для більшості синтезованих математичних моделей створені комп'ютерні програми на мові Python, з використанням яких не лише розв'язуються задачі

ідентифікації цих моделей, але і здійснюється моделювання процесів у динамічних об'єктах з використанням синтезованих та ідентифікованих їх моделей. Саме завдяки цьому доповненню друге видання навчального посібника «Математичні методи ідентифікації динамічних систем» було вирішено назвати «Математичні методи ідентифікації та моделювання динамічних систем»

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мокін Б. І. Математичні методи ідентифікації динамічних систем: навчальний посібник / Б. І. Мокін, В. Б. Мокін, О. Б. Мокін. – Вінниця : ВНТУ, 2010. – 260 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/301>
2. Мокін Б. І. Методологія та організація наукових досліджень: підручник, вид. 3-є, змін. та доп. / Б. І. Мокін, О. Б. Мокін, В. Б. Мокін. – Вінниця: ВНТУ, 2023. – 230 с. – Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/view/805/1408/2677-1>.
3. Мокін Б. І. Системний аналіз процесу здобування знань в університетському середовищі./ Б. І. Мокін, О. О. Войцеховська, О. Б. Мокін – Вінниця: ВНТУ. – 2024. – 166 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/862>
4. Мокін Б. І. Навчальний посібник для опанування студентами способів розв'язання задач з функціонального аналізу мовою Python. Частина 1 / Б. І. Мокін, В. Б. Мокін, О. Б. Мокін. – Вінниця : ВНТУ, 2022. – 124 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/689>
5. Мокін Б. І. Навчальний посібник для опанування студентами способів розв'язання задач з функціонального аналізу мовою Python. Частина 2 : навчальний посібник / Б. І. Мокін, В. Б. Мокін, О. Б. Мокін. – Вінниця : ВНТУ, 2023. – 139 с. режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/768>
6. Мокін Б. І. Функціональний аналіз, адаптований до прикладних задач в галузі інформаційних технологій : навчальний посібник / Б. І. Мокін, В. Б. Мокін, О. Б. Мокін. – Вінниця : ВНТУ, 2020. – 192 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/611>
7. Мокін Б. І. Методи та засоби комп'ютерних обчислень: навчальний посібник / Б. І. Мокін, В. Б. Мокін, О. Б. Мокін, Вінниця : ВНТУ, 2024. – 129 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/870>
8. Python. [Електронний ресурс]. Режим доступу: <https://www.python.org/downloads/>.
9. Кветний Р. Н. Комп'ютерне моделювання систем та процесів. Методи обчислень. Част. 1 : навчальний посібник / Р. Н. Кветний та ін. Вінниця : ВНТУ, 2013. 191 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/156>
10. Кветний Р. Н. Комп'ютерне моделювання систем та процесів. Методи обчислень. Част. 2 : навчальний посібник / Р. Н. Кветний та ін. Вінниця : ВНТУ, 2013. 235 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/157>
11. Кветний Р. Н. Методи та алгоритми комп'ютерних обчислень. Теорія і практика : підручник / Р. Н. Кветний та ін. Вінниця : ВНТУ, 2023. 280 с. Режим доступу: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/767>
12. IDE Anaconda. Режим доступу: <https://anaconda.com>

13. Доля П. Г. Уведення в науковий Python. Харків : ХНУ ім. Каразіна, 2016. 265 с.

14. Мокін Б. І. Практикум для самостійної роботи студентів з навчальної дисципліни «Методологія та організація наукових досліджень». Частина 1: від постановки задачі до синтезу та ідентифікації математичної моделі./ Б. І. Мокін, О. Б. Мокін, В. Б. Мокін - Вінниця : ВНТУ, 2018. 179 с. - Режим доступу: <http://www.mokin.com.ua/pedagogical/posibn/6546.html>

15. Мокін О. Б. Метод ідентифікації моделі авторегресії-ковзного середнього АРКС(p,q) з довільними значеннями порядків p, q, який узагальнює методику Юла – Уокера./ О. Б. Мокін, В. Б. Мокін, Б. І. Мокін Б. І. - *Наукові праці Вінницького національного технічного університету*. 2014. № 2. С. 1–6. Режим доступу : <http://praci.vntu.edu.ua/article/view/3626/5339>.

16 Мокін О. Б., Мокін В. Б., Мокін Б. І. Алгоритм методу ідентифікації моделі авторегресії - ковзного середнього, який узагальнює методику Юла – Уокера, та його програмна Python-реалізація./ О. Б. Мокін, В. Б. Мокін, Б. І. Мокін - *Вісник Вінницького політехнічного інституту*. 2022. № 4. С. 41–55. DOI: <https://doi.org/10.31649/1997-9266-2022-163-4-41-55>

17. Box George E. P., Jenkins Gwilym M. TIME SERIES ANALYSIS. Forecasting and control. HOLDEN-DAY: San Francisco, Cambridge, London, Amsterdam, 1970. 532 p.

18. Войцеховська О. О., Мокін Б. І., Мокін О. Б.. Інформаційна технологія реалізації фур'є-інтегрального методу ідентифікації для відновлення вхідних сигналів інформаційно-вимірювальних систем/ О. О. Войцеховська, Б. І. Мокін, О. Б. Мокін - *Вісник Вінницького політехнічного інституту*. 2025. № 3. С. 90–100. DOI: <https://doi.org/10.31649/1997-9266-2025-180-3-90-100>

Електронне навчальне видання

**Борис Іванович Мокін
Віталій Борисович Мокін
Олександр Борисович Мокін**

МАТЕМАТИЧНІ МЕТОДИ ІДЕНТИФІКАЦІЇ ТА МОДЕЛЮВАННЯ ДИНАМІЧНИХ СИСТЕМ

Навчальний посібник

Рукопис оформив *Б. Мокін*

Оригінал виготовлено в *Редакційно-видавничому відділі ВНТУ*

Підписано до видання 30.06.2026 р.
Гарнітура Times New Roman.
Зам. № P2026-078.

Видавець та виготовлювач
Вінницький національний технічний університет,
Редакційно-видавничий відділ.
ВНТУ, ГНК, к. 114.
Хмельницьке шосе, 95, м. Вінниця, 21021.
press.vntu.edu.ua;
email: rvv.vntu@gmail.com.
Свідоцтво суб'єкта видавничої справи
серія ДК № 3516 від 01.07.2009 р.