

Міністерство освіти і науки України
Кам'янець-Подільський національний університет імені Івана Огієнка
Фізико-математичний факультет
Кафедра комп'ютерних наук

Кваліфікаційна робота магістра
з теми: «Розробка та застосування структурних моделей на
основі ланцюгових дробів для аналізу лінійних динамічних
систем з розподіленими параметрами»

Виконав:
здобувач вищої освіти групи KN1-M23
спеціальності 122 Комп'ютерні науки
Павлюк Богдан Олександрович

Керівник:
Іванюк Віталій Анатолійович,
доктор технічних наук, доцент,
завідувач кафедри комп'ютерних наук

Рецензент:
Ковальська Ірина Борисівна,
кандидат фізико-математичних наук,
доцент кафедри математики

Кам'янець-Подільський, 2024 р.

АНОТАЦІЯ

Павлюк Б.О. Розробка та застосування структурних моделей на основі ланцюгових дробів для аналізу лінійних динамічних систем з розподіленими параметрами. – Кваліфікаційна робота магістра, 2024.

Кваліфікаційна робота присвячена актуальній проблемі моделювання динамічних систем з розподіленими параметрами. Метою роботи є розробка ефективних методів структурного та математичного моделювання таких систем на основі ланцюгових дробів.

У роботі проведено детальний аналіз існуючих методів моделювання, розроблено алгоритми апроксимації передатних функцій ланцюговими дробами та створено програмні модулі для їх реалізації в середовищі MATLAB/Simulink.

Запропонований підхід дозволяє підвищити точність і ефективність моделювання складних систем, що мають широке застосування в різних галузях науки і техніки. Результати дослідження можуть бути використані для розробки нових алгоритмів ідентифікації та адаптації систем, а також для оптимізації робочих процесів в промисловості.

Ключові слова: динамічні системи, розподілені параметри, ланцюгові дроби, апроксимація, моделювання, MATLAB, Simulink.

ABSTRACT

Pavliuk B.O. Development and application of structural models based on continued fractions for the analysis of linear dynamic systems with distributed parameters. – Master's qualification work, 2024.

The qualification work is devoted to the current problem of modeling dynamic systems with distributed parameters. The aim of the work is to develop effective methods for structural and mathematical modeling of such systems based on continued fractions.

The work provides a detailed analysis of existing modeling methods, algorithms for approximating transfer functions by continued fractions are developed, and software modules are created for their implementation in the MATLAB/Simulink environment.

The proposed approach allows to increase the accuracy and efficiency of modeling complex systems that are widely used in various fields of science and technology. The results of the study can be used to develop new algorithms for identifying and adapting systems, as well as for optimizing work processes in industry.

Keywords: dynamic systems, distributed parameters, continued fractions, approximation, modeling, MATLAB, Simulink.

ЗМІСТ

ВСТУП.....	5
РОЗДІЛ 1. МЕТОДИ СТРУКТУРНОГО ТА МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ СИСТЕМ.....	8
1.1. Метод структурного моделювання	8
1.2. Передатні функції.....	14
1.3. Передатні функції об'єктів з розподіленими параметрами	19
Висновки до розділу 1	24
РОЗДІЛ 2. СЕРЕДОВИЩЕ МОДЕЛЮВАННЯ.....	26
2.1. Загальна характеристика MATLAB.....	26
2.2. Побудова та аналіз моделей динамічних систем в MATLAB	29
2.3. Середовище імітаційного моделювання Simulink	32
2.4. Інструменти апроксимації в Matlab.....	35
Висновки до розділу 2.	38
РОЗДІЛ 3. АЛГОРИТМИ АПРОКСИМАЦІЇ НА ОСНОВІ ЛАНЦЮГОВИХ ДРОБІВ.....	39
3.1. Алгоритми побудови ланцюгових дробів для апроксимації моделей ...	39
3.2. Алгоритми побудови правильних C -дробів	40
3.3. Оптимізований алгоритм побудови дробово-раціональних передатних функцій.....	41
3.4. Алгоритм отримання коефіцієнтів ланцюгових дробів	43
Висновок до розділу 3.	44
РОЗДІЛ 4. ПРОГРАМНІ ЗАСОБИ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ	45
4.1. Огляд розроблених модулів.....	45
4.2. Обчислювальні експерименти побудови ланцюгових дробів.....	46

4.3. Імітаційне моделювання розподілених ланок на основі ланцюгових дробів.....	48
4.4. Оптимізація структури моделей в Simulink	50
4.5. Модель "вхід-вихід" для ланцюгового дробу	52
4.6. Обчислювальні експерименти	53
Висновки до 4 розділу	61
ВИСНОВКИ	63
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ.....	64

ВСТУП

Дослідження динамічних систем з розподіленими параметрами набуває все більшої актуальності в сучасному світі, де технології розвиваються стрімкими темпами. Такі системи широко представлені в різних галузях, таких як автоматичне керування (наприклад, робототехніка, процеси виробництва), електротехніка (лінії електропередачі), механіка (вібрації конструкцій, теплові процеси) та інших.

Точне моделювання динамічних систем з розподіленими параметрами є критичним для досягнення кількох важливих цілей: оптимізації робочих процесів, зниження енергоспоживання та підвищення надійності систем, що веде до покращення їхньої ефективності; створення інноваційних продуктів та рішень на основі глибокого розуміння динамічних процесів, що сприяє розробці нових технологій; а також забезпечення безпеки шляхом прогнозування та запобігання аварійним ситуаціям, особливо в критичних системах.

Однак, точне моделювання систем з розподіленими параметрами є складним завданням через їхню високу розмірність та складність математичного опису. Існуючі методи часто мають обмеження щодо точності, швидкодії або універсальності.

Саме тому актуальним є завдання розробки ефективних методів структурного та математичного моделювання динамічних систем з розподіленими параметрами на основі ланцюгових дробів, що дозволить подолати зазначені обмеження та забезпечити більш точне і ефективне моделювання таких систем.

Мета роботи. Метою роботи є розробка методів структурного та математичного моделювання лінійних динамічних систем з розподіленими параметрами на основі ланцюгових дробів, а також створення програмних засобів для їх реалізації.

Завдання дослідження:

- провести аналіз існуючих методів моделювання лінійних динамічних систем;
- розробити алгоритми апроксимації передатних функцій систем з розподіленими параметрами ланцюговими дробами;
- створити програмні модулі для реалізації розроблених алгоритмів у середовищі MATLAB/Simulink;
- провести обчислювальні експерименти для оцінки точності та ефективності запропонованих методів.

Об'єкт дослідження. Об'єктом дослідження є лінійні динамічні системи з розподіленими параметрами, які описуються складними математичними моделями.

Предмет дослідження. Предметом дослідження є методи структурного та математичного моделювання лінійних динамічних систем з розподіленими параметрами на основі ланцюгових дробів, а також програмні засоби їх реалізації.

Методи дослідження. У роботі застосовано широкий спектр методів дослідження. Теоретичний аналіз літератури дозволив ознайомитися з існуючими підходами до моделювання динамічних систем. Математичне моделювання було використано для розробки формальних моделей досліджуваних об'єктів. Комп'ютерне моделювання в середовищі MATLAB/Simulink забезпечило можливість проведення чисельних експериментів та аналізу отриманих результатів. Статистична обробка даних дозволила оцінити точність і достовірність моделей.

Практичне значення. Запропоновані методи та алгоритми знайдуть своє застосування в різних галузях. Зокрема, вони дозволять підвищити точність моделювання складних технічних систем, що є критичним для їхнього ефективного функціонування. Крім того, результати дослідження спростять процедуру проектування систем автоматичного керування, зменшивши час та ресурси, необхідні для розробки нових рішень. Не менш важливим є потенціал розробки нових алгоритмів ідентифікації та адаптації систем, що відкриває

нові можливості для створення більш гнучких та адаптивних систем управління.

Елементи наукової новизни. Наукова новизна роботи полягає у розробці нового методу моделювання, який дозволяє підвищити ефективність та універсальність аналізу складних динамічних систем. Запропонований підхід базується на використанні апроксимаційних моделей у формі ланцюгових дробів та імітаційного моделювання, а його реалізація в середовищі MATLAB/Simulink забезпечує практичну значущість для широкого кола задач автоматичного керування.

Апробація результатів. Результати дослідження опубліковані у Віснику Кам'янець-Подільського національного університету імені Івана Огієнка. Фізико-математичні науки. Випуск 17. Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2024 [19].

РОЗДІЛ 1. МЕТОДИ СТРУКТУРНОГО ТА МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ СИСТЕМ

1.1. Метод структурного моделювання

Структурний метод моделювання є потужним інструментом для аналізу складних систем. Його суть полягає у розбитті системи на більш прості підсистеми, для кожної з яких будується окрема математична модель. Цей підхід дозволяє спростити аналіз системи та зрозуміти взаємозв'язки між її компонентами.

Розбиття системи на підсистеми є першим і найважливішим етапом. Критерії розбиття можуть бути різними: фізичні процеси, які відбуваються в системі, функціональні блоки, або комбінація обох. Наприклад, електричну схему можна розбити на окремі елементи (резистори, конденсатори, котушки індуктивності), а механічну систему – на тверді тіла, пружини та демпфери.

Після розбиття системи на підсистеми для кожної з них будується математична модель. Вибір математичного апарату залежить від конкретної підсистеми та її характеристик. Це можуть бути диференціальні рівняння, передатні функції, графи та інші математичні формалізми. Важливим етапом є ідентифікація параметрів моделей на основі експериментальних даних або теоретичних розрахунків.

З'єднання моделей підсистем дозволяє отримати модель всієї системи. Це може бути здійснено за допомогою алгебраїчних рівнянь, які описують взаємодію між підсистемами, або графічних методів, таких як блок-схеми. При цьому необхідно враховувати всі взаємодії між підсистемами, щоб отримати адекватну модель системи в цілому.

Аналіз моделі всієї системи дозволяє досліджувати її поведінку та оцінити її характеристики. Досліджуються як статичні характеристики системи (наприклад, стаціонарні режими), так і динамічні (реакція системи на зовнішні збурення). Важливими характеристиками є стійкість, керованість та

спостережуваність системи. Стійкість означає здатність системи повертатися до рівноважного стану після невеликих збурень. Керованість характеризує можливість зміни стану системи за допомогою зовнішніх впливів. Спостережуваність означає можливість визначення стану системи за доступними вимірами.

Отже, структурний метод моделювання є потужним інструментом, який широко застосовується в різних галузях науки і техніки для аналізу складних систем. Він дозволяє не тільки зрозуміти поведінку системи, але й розробляти ефективні алгоритми управління та оптимізації [1, 3, 4, 5, 14, 15].

Структурний метод є універсальним, оскільки може бути застосований до широкого спектра систем, незалежно від їх фізичної природи. Незалежно від того, чи йдеться про механічні, електричні, біологічні або соціальні системи, структурний підхід дозволяє розбити їх на складові частини та побудувати відповідні моделі. Це робить його потужним інструментом для дослідження різноманітних явищ.

Метод має значну гнучкість та дозволяє адаптувати рівень деталізації моделі до конкретних завдань. Для початкового аналізу можна використовувати спрощені моделі, а для більш детального дослідження – більш складні. Це дозволяє дослідникам фокусуватися на найбільш важливих аспектах системи та ігнорувати другорядні деталі.

Однією з ключових переваг структурного моделювання є можливість повторного використання моделей підсистем. Якщо в системі є подібні підсистеми, їх моделі можуть бути використані багаторазово, що значно скорочує час і зусилля на розробку моделі всієї системи. Це також спрощує внесення змін до моделі, оскільки достатньо змінити лише модель відповідної підсистеми.

Структурні моделі зазвичай представляються у вигляді графів або блок-схем. Це особливо корисно для розуміння складних систем і спілкування з іншими фахівцями. Візуалізація також полегшує виявлення помилок у моделі та внесення необхідних змін.

Незважаючи на всі свої переваги, метод структурного моделювання має також свої обмеження, які необхідно враховувати при його застосуванні.

Зі збільшенням складності системи зростає і складність її моделі. Побудова деталізованих моделей для великих систем може вимагати значних обчислювальних ресурсів та часу. Крім того, для складних систем може бути важко визначити оптимальний рівень деталізації моделі.

Параметри моделей часто визначаються на основі експериментальних даних, які можуть містити похибки. Це призводить до невизначеності в моделі та може вплинути на точність результатів моделювання.

Модель завжди є спрощеним представленням реальної системи. Вона не може врахувати всі фактори, які впливають на поведінку системи. Тому завжди існує ризик, що модель не буде адекватно відображати реальність. Для оцінки адекватності моделі проводять її валідацію на експериментальних даних.

Особливо складнощі проявляються при моделюванні систем із розподіленими параметрами, оскільки вони характеризуються тим, що їхні стани не можуть бути описані скінченним числом змінних. Наприклад, температура в тілі, напруження в пружному середовищі або концентрація речовини в розчині змінюються безперервно в просторі. Це ускладнює їхнє моделювання, оскільки вимагає розв'язання диференціальних рівнянь в частинних похідних.

До особливостей систем з розподіленими параметрами варто віднести те, стани системи безперервно змінюються в просторі та часі, для опису таких систем використовуються диференціальні рівняння в частинних похідних, які часто не мають аналітичних розв'язків, а для повного опису системи потрібно враховувати безліч змінних, що ускладнює аналіз моделі.

Для того щоб застосувати метод структурного моделювання до систем з розподіленими параметрами, необхідно дискретизувати простір і час. Це дозволяє замінити неперервні диференціальні рівняння системою звичайних диференціальних рівнянь або алгебраїчних рівнянь. Одними з найпоширеніших є методи дискретизації, а саме метод скінченних різниць та

метод скінченних елементів, які базуються на розбитті області дискретною сіткою або елементами (трикутники, чотирикутники тощо), для кожного з яких будується аппроксимація розв'язку. Метод скінченних елементів є більш універсальним і дозволяє моделювати складні геометрії та неоднорідні матеріали. Він широко використовується в інженерних розрахунках, таких як аналіз на міцність, теплопередача, дифузія. Метод скінченних різниць простіший у реалізації, але менш гнучкий. Він добре підходить для задач з простими геометріями.

Прикладами застосування є моделювання розподілу температури в тілі при різних умовах нагрівання або охолодження, процесів дифузії речовин в різних середовищах (газах, рідинах, твердих тілах), вібрацій мембран під дією зовнішніх сил, поширення електромагнітних хвиль в різних середовищах.

Переваги використання структурного моделювання для систем з розподіленими параметрами полягає в тому, що можна отримати графічне представлення розподілу змінних в просторі і часі, оцінити вплив різних параметрів на поведінку системи, знайти оптимальні параметри системи для досягнення заданих цілей.

Метод скінченних різниць та метод скінченних елементів – це потужні інструменти для чисельного розв'язання диференціальних рівнянь, які широко застосовуються в інженерних розрахунках. Однак, кожен з цих методів має значні складнощі, до яких можна віднести складність побудови сітки для складних геометрій, для складних задач методи вимагають значних обчислювальних ресурсів, які, часто є не виправданими, особливо коли розглядаються задачі типу «вхід-вихід» [8].

Ще одним видом математичних моделей, який зазвичай застосовують в структурному методі, є моделі подані у вигляді передатних функцій, які на відміну від методів скінченних різниць та скінченних елементів, мають ряд переваг.

Передатна функція – це математичний засіб, який описує поведінку лінійної стаціонарної системи. Вона являє собою відношення перетворення

Лапласа вихідного сигналу до перетворення Лапласа вхідного сигналу за нульових початкових умов. Іншими словами, передатна функція показує, як система перетворює вхідний сигнал у вихідний у частотній області.

Передатні функції є більш компактним і зручним для аналізу математичним апаратом, ніж системи диференціальних рівнянь, які отримуються в результаті застосування методів скінченних різниць або скінченних елементів. За допомогою передатних функцій легко проводити аналіз стійкості системи, використовуючи критерії стійкості (наприклад, критерій Найквіста). Передатні функції широко використовуються в задачах синтезу систем автоматичного управління та дозволяють проводити частотний аналіз системи, що є важливим для оцінки її динамічних властивостей.

Для застосування передатних функцій до систем з розподіленими параметрами необхідно проводити їх апроксимацію, тобто замінювати систему з розподіленими параметрами системою зі скінченним числом ступенів свободи. Така апроксимація завжди вводить деяку похибку, тому вибір методів апроксимації має важливе значення. В цьому випадку, одним із найкращих методів є метод ланцюгових дробів [9].

Ланцюговий дріб – це математичний вираз, який представляє число або функцію у вигляді нескінченної суми дробів. В контексті структурного моделювання, ланцюгові дроби можуть бути використані для представлення передатних функцій лінійних стаціонарних систем.

Ланцюгові дроби можуть апроксимувати довільну раціональну функцію з будь-якою заданою точністю. Це особливо корисно для представлення передатних функцій систем з розподіленими параметрами, які часто мають складні частотні характеристики.

Ланцюгові дроби мають модульну структуру, що дозволяє легко додавати нові елементи в модель і змінювати її структуру. Ланцюгові дроби дозволяють проводити аналіз стійкості, керованості та інших характеристик системи більш ефективно, ніж традиційні методи.

Як вже згадувалося, ланцюгові дроби дозволяють апроксимувати складні передатні функції з високою точністю. Це особливо важливо для систем з розподіленими параметрами, які часто мають нераціональні частотні характеристики.

Існує ряд критеріїв стійкості, які формулюються в термінах коефіцієнтів ланцюгового дроби. Це дозволяє легко перевіряти стійкість системи. Ланцюгові дроби широко використовуються для синтезу цифрових і аналогових фільтрів. Вони дозволяють отримувати фільтри з різними типами частотних характеристик (низькочастотні, високочастотні, смугові тощо).

Отже, ланцюгові дроби є потужним інструментом для моделювання систем з розподіленими параметрами. Вони дозволяють апроксимувати складні передатні функції, що дозволяє проводити аналіз їх стійкості та синтезувати фільтри. Завдяки своїй модульності та ефективності обчислень, ланцюгові дроби знаходять широке застосування в різних областях науки і техніки.

Таким чином, структурний метод є потужним інструментом для аналізу складних систем, включаючи ті, що мають розподілені параметри. Він дозволяє розбити систему на окремі елементи, описати їх поведінку математично і потім об'єднати в єдину модель. Системи з розподіленими параметрами ускладнюють аналіз через нескінченну кількість ступенів свободи. Передатні функції, хоча й зручні для опису лінійних стаціонарних систем, мають обмеження при застосуванні до систем з розподіленими параметрами. Ланцюгові дроби пропонують ефективний спосіб апроксимації передатних функцій таких систем, дозволяючи досягти високої точності апроксимації і маючи ряд переваг, таких як модульність і зручність для аналізу стійкості. Комбінація структурного методу, передатних функцій і ланцюгових дробів дозволяє проводити детальний аналіз і синтез складних систем [2, 16, 17, 18].

1.2. Передатні функції

Розглянемо детальніше поняття передатних функцій [17, 18].

Передатна функція – це математичний вираз, який показує, як система перетворює вхідний сигнал у вихідний. Вона розраховується як відношення перетворення Лапласа вихідного сигналу до перетворення Лапласа вхідного сигналу за умови, що система спочатку була в стані спокою (нульові початкові умови).

Перетворення Лапласа – це математична операція, яка переводить функцію часу в функцію комплексної змінної s . Це дозволяє перейти від диференціальних рівнянь, що описують динаміку системи, до алгебраїчних рівнянь, які простіше аналізувати. Формально, перетворення Лапласа функції $f(t)$ визначається як інтеграл виду:

$$F(s) = \int_{[0, \infty]} f(t) * e^{(-st)} dt .$$

Передатні функції тісно пов'язані з лінійними диференціальними рівняннями зі сталими коефіцієнтами. Такі рівняння часто використовуються для опису динамічних систем у різних областях науки і техніки. Наприклад, розглянемо загальне лінійне диференціальне рівняння n -го порядку:

$$\begin{aligned} a_n * y^{(n)} + a_{(n-1)} * y^{(n-1)} + \dots + a_1 * y' + a_0 * y = \\ = b_m * x^{(m)} + b_{(m-1)} * x^{(m-1)} + \dots + b_1 * x' + b_0 * x \end{aligned} \quad (1.1)$$

У цьому рівнянні:

- $y(t)$ – вихідна величина, яку ми хочемо знайти;
- $x(t)$ – вхідна величина, що впливає на систему;
- a_i та b_i – сталі коефіцієнти, які характеризують властивості системи;
- n та m – порядки похідних за часом t .

Це рівняння описує, як зміна вхідної величини $x(t)$ впливає на зміну вихідної величини $y(t)$ з урахуванням властивостей самої системи.

Рівняння (1.1) є неоднорідним, оскільки права частина рівняння містить члени, які не залежать від шуканої змінної $y(t)$. Ці члени, які ми перенесли в праву частину рівняння, називаються вільними членами або зовнішніми силами. Вони відображають вплив зовнішніх факторів на систему.

У контексті динамічних систем, вільний член $x(t)$ часто інтерпретується як вхідний сигнал, який впливає на систему, а шукана функція $y(t)$ – як вихідний сигнал, що є реакцією системи на цей вхідний сигнал.

Коли на систему не діють зовнішні сили або впливи, тобто права частина рівняння дорівнює нулю, таку систему називають вільною. Математично, вільна система описується однорідним лінійним диференціальним рівнянням:

$$a_n * y^{(n)} + a_{(n-1)} * y^{(n-1)} + \dots + a_1 * y' + a_0 * y = 0 \quad (1.2)$$

Це означає, що зміна вихідної величини $y(t)$ визначається лише внутрішніми властивостями системи, а не зовнішніми впливами.

Розглянемо загальний розв'язок однорідного рівняння (1.2). Для спрощення подальших міркувань, припустимо спочатку, що всі корені характеристичного рівняння різні.

У такому випадку загальний розв'язок рівняння (1.2) можна записати у вигляді лінійної комбінації експонент:

$$y(t) = C^1 e^{(\lambda^1 t)} + C^2 e^{(\lambda^2 t)} + \dots + C_n e^{(\lambda_n t)} \quad (1.3)$$

де:

- $C_1, C_2, \dots, C_n$ - довільні сталі, які визначаються з початкових умов;
- $\lambda_1, \lambda_2, \dots, \lambda_n$ - корені характеристичного рівняння.

Підставляючи цей розв'язок у рівняння (1.2), легко перевірити, що він задовольняє цьому рівнянню при будь-яких значеннях сталих $C_1, \dots, C_n$.

Якщо серед коренів характеристичного рівняння є кратні, то структура загального розв'язку дещо ускладнюється. Наприклад, якщо корінь λ_1 має

кратність k , то відповідній частині загального розв'язку відповідають не лише члени виду $C^1 e^{(\lambda^1 t)}$, але й члени виду $t e^{(\lambda^1 t)}, t^2 e^{(\lambda^1 t)}, \dots, t^{(k-1)} * e^{(\lambda^1 t)}$.

Таким чином, загальний розв'язок однорідного рівняння (1.2) завжди можна представити як лінійну комбінацію експонент та їх добутків на степені незалежної змінної t .

Важливо зауважити, що кожен доданок у загальному розв'язку відповідає певному кореню характеристичного рівняння, довільні сталі $C_1, C_2, \dots, C_n$ визначаються з початкових умов задачі, структура загального розв'язку залежить від характеру коренів характеристичного рівняння (дійсні, комплексні, кратні).

Аналіз коренів характеристичного полінома є ключовим інструментом для розуміння поведінки динамічних систем, описаних лінійними диференціальними рівняннями. Корені цього полінома містять важливу інформацію про характер розв'язків рівняння, а отже, і про поведінку системи з часом.

Якщо корінь характеристичного полінома є дійсним числом, то відповідна складова загального розв'язку описує експоненціальну зміну вихідної величини системи. Якщо цей корінь додатний, то вихідна величина з часом необмежено зростає, що свідчить про нестійкість системи. Наявність хоча б одного додатного дійсного кореня гарантує нестійкість системи.

Корені характеристичного полінома з дійсними коефіцієнтами можуть бути або дійсними числами, або комплексно-спряженими парами. Комплексні корені відповідають за коливальні процеси в системі. Їхні дійсні частини визначають швидкість затухання або наростання коливань, а уявні частини – їх частоту.

Таким чином, аналізуючи корені характеристичного полінома, можна зробити висновки про стійкість системи, характер перехідних процесів та інші важливі характеристики. Це дозволяє оцінити якість роботи системи, а також розробляти заходи для її покращення.

Важливим аспектом є також кратність коренів. Якщо корінь має кратність більшу за одиницю, то відповідна частина загального розв'язку містить не лише експоненту, але й її добуток на степені часу. Це призводить до більш складних перехідних процесів.

Крім того, нульові корені вказують на наявність постійної складової в розв'язку, що може бути пов'язано з наявністю інтегруючих ланок в системі.

Підсумовуючи, аналіз коренів характеристичного полінома є потужним інструментом для дослідження динамічних систем і широко використовується в різних областях науки і техніки, зокрема в теорії автоматичного керування, електротехніці, механіці та інших.

Поняття передатної функції тісно пов'язане з перетворенням Лапласа і використовується для опису лінійних систем, динаміка яких описується диференціальними рівняннями виду (1.1)-(1.2).

Перетворення Лапласа є математичним інструментом, який переводить функцію часу в функцію комплексної змінної s . Зображення функції $f(t)$ за Лапласом позначається $F(s)$ і визначається за інтегралом:

$$F(s) = L[f(t)] = \int_{[0, \infty]} e^{(-st)} * f(t) dt \quad (1.4)$$

Однією з важливих властивостей перетворення Лапласа є його вплив на похідні функцій. Якщо $F(s)$ – зображення функції $f(t)$, то зображення її n -ї похідної $f^{(n)}(t)$ виражається так:

$$L[f^{(n)}(t)] = s^n * F(s) - s^{(n-1)} * f(0) - s^{(n-2)} * f'(0) - \dots - f^{(n-1)}(0) \quad (1.5)$$

Застосуємо перетворення Лапласа до рівняння (1.1), враховуючи співвідношення (1.5) і припускаючи нульові початкові умови. Отримаємо:

$$\begin{aligned} a_n * s^n * Y(s) + a_{(n-1)} * s^{(n-1)} * Y(s) + \dots + a_0 * Y(s) = \\ = b_m * s^m * X(s) + b_{(m-1)} * s^{(m-1)} * X(s) + \dots + b_0 * X(s) \end{aligned} \quad (1.6)$$

де $Y(s)$ і $X(s)$ – відповідно зображення вихідного сигналу $y(t)$ та вхідного сигналу $x(t)$.

З рівняння (1.6) можна виразити відношення зображень вихідного і вхідного сигналів:

$$Y(s)/X(s) = G(s) \quad (1.7)$$

Величина $G(s)$, яка є відношенням зображення вихідного сигналу до зображення вхідного сигналу, називається передатною функцією системи. Її вираз має вигляд:

$$G(s) = \frac{b_ms^m + b_{m-1}s^{m-1} + \dots + b_0}{a_ns^n + a_{n-1}s^{n-1} + \dots + a_0} \quad (1.8)$$

Узагальнюючи поняття передатної функції, можна сказати, що передатна функція системи – це відношення зображень за Лапласом вихідної та вхідної величин за умови, що система спочатку перебувала у стані спокою (нульові початкові умови). Це відношення не залежить від конкретного виду вхідного сигналу, а визначається лише структурою та параметрами самої системи.

Важливою особливістю передатної функції є те, що вона може бути застосована лише до лінійних стаціонарних систем. Це означає, що поведінка такої системи описується лінійними диференціальними рівняннями зі сталими коефіцієнтами. Тобто, принцип суперпозиції виконується, а параметри системи не змінюються з часом.

Математично, передатна функція $G(s)$ є дробово-раціональною функцією комплексної змінної s (1.8). Знаменник передатної функції збігається з характеристичним поліномом диференціального рівняння, яке описує систему. Корені знаменника називаються полюсами передатної функції. Вони визначають стійкість системи та характер її перехідних процесів. Корені чисельника називаються нулями передатної функції і впливають на форму імпульсної характеристики системи.

Отже, передатна функція є потужним математичним інструментом для аналізу та синтезу лінійних стаціонарних систем. Вона дозволяє перейти від часової області до частотної, що значно спрощує дослідження динамічних властивостей систем.

Дробово-раціональні передатні функції є зручним інструментом для опису лінійних стаціонарних систем зі скінченним числом елементів, таких як електричні кола, механічні системи та системи автоматичного керування. Однак, для опису більш складних систем, зокрема систем з розподіленими параметрами, цей апарат виявляється недостатнім [16, 17, 18].

1.3. Передатні функції об'єктів з розподіленими параметрами

Передатні функції є універсальним інструментом для опису динамічних систем у різних галузях науки і техніки, зокрема в механіці, електродинаміці та системах автоматичного керування. Завдяки своїй наочності та зручності у використанні, передатні функції широко застосовуються в математичному моделюванні та аналізі систем. Більшість сучасних програмних засобів для моделювання систем орієнтовані саме на роботу з передатними функціями.

Однак, якщо для систем зі зосередженими параметрами, які описуються звичайними диференціальними рівняннями, побудова передатних функцій є досить стандартною процедурою, то для систем з розподіленими параметрами, які описуються диференціальними рівняннями з частинними похідними, ця задача має свої особливості.

Коли маємо справу з системами, що містять розподілені параметри (наприклад, теплові процеси, дифузія, коливання пружних тіл) і описуються диференціальними рівняннями в частинних похідних, традиційного поняття передатної матриці стає недостатнім для повної характеристики їхніх передатних властивостей, оскільки системи з розподіленими параметрами мають нескінченне число ступенів свободи, тобто їх стан не може бути описаний скінченним числом змінних. В таких системах параметри розподілені в просторі, а не зосереджені в окремих точках.

Двоточкова функція Гріна $G(x_2, x_1, s)$ характеризує відгук системи в точці $x_2 \in Q$ на одиничний імпульсний вплив, прикладений в точці $x_1 \in P$. Математично вона визначається як відношення перетворення Лапласа за часом від вихідного сигналу в точці x_2 до перетворення Лапласа за часом від вхідного сигналу в точці x_1 :

$$G(x_2, x_1, s) = \frac{Y(x_2, s)}{F(x_1, s)}$$

де:

- $G(x_2, x_1, s)$ – двоточкова функція Гріна;
- $Y(x_2, s)$ – перетворення Лапласа за часом від вихідного сигналу в точці x_2 ;
- $F(x_1, s)$ – перетворення Лапласа за часом від одиничного імпульсного впливу в точці x_1 ;
- s – комплексна змінна перетворення Лапласа;
- $x_1 \in P$ – точка прикладання вхідного впливу;
- $x_2 \in Q$ – точка спостереження вихідного сигналу.

Двоточкова функція Гріна показує, як розповсюджується вплив, прикладений в одній точці системи, в інші точки системи. Вона є узагальненням поняття імпульсної характеристики для систем зі зосередженими параметрами на системи з розподіленими параметрами.

Двоточкова функція Гріна широко використовується в різних галузях науки і техніки, таких як:

- теорія коливань;
- теорія пружності;
- теорія теплопровідності;
- електродинаміка;
- акустика.

За допомогою двоточнової функції Гріна можна аналізувати стійкість систем з розподіленими параметрами. Вона є основою для розробки систем

автоматичного керування для об'єктів з розподіленими параметрами. Цей інструмент використовується для проектування різних технічних пристроїв, таких як теплообмінники, реактори, антенні системи тощо.

Поняття двоточної функції Гріна є важливим доповненням до традиційного апарату передатних функцій і дозволяє ефективно аналізувати та синтезувати системи з розподіленими параметрами.

Для різних типів структур об'єктів з розподіленими параметрами існують різні методи визначення їх передатних функцій. Серед найпоширеніших можна виділити: прямий метод, інтегральний метод, метод функціональних перетворень та матричний метод.

Прямий метод є одним з найпростіших і найінтуїтивніших. Він полягає у безпосередньому застосуванні перетворення Лапласа до диференціального рівняння в частинних похідних, яке описує поведінку об'єкта.

Прямий метод знаходження передатних функцій для систем з розподіленими параметрами передбачає послідовне виконання таких кроків: спочатку формулюється диференціальне рівняння в частинних похідних, яке описує процес у системі; потім, застосовуючи перетворення Лапласа, це рівняння перетворюється на звичайне диференціальне рівняння відносно просторової змінної; далі, розв'язуючи це рівняння з урахуванням граничних умов, отримуємо вирази для зображень вихідної та вхідної величин; нарешті, обчислюючи відношення цих зображень, знаходимо передатну функцію системи, яка повністю описує її динамічні властивості в області зображень Лапласа.

Прямий метод є одним з найдоступніших інструментів для визначення передатних функцій систем з розподіленими параметрами завдяки своїй простоті та універсальності. Однак, для складних систем розв'язання отриманих рівнянь може бути досить громіздким, а для деяких типів систем можуть бути більш ефективними інші підходи. Тобто, хоча прямий метод є зручним інструментом, його застосування має свої обмеження, які залежать від конкретної задачі та складності моделі системи.

Прямий метод особливо зручний для систем, які описуються хвильовим рівнянням. Це пов'язано з тим, що при застосуванні перетворення Лапласа до хвильового рівняння отримується звичайне диференціальне рівняння четвертого порядку (або вищого порядку для складніших випадків). Розв'язання такого рівняння, хоча і є складним, але є стандартною задачею математичної фізики.

Інтегральний метод є особливо ефективним, коли потрібно визначити передатні функції між входом системи та різними точками виходу. Це особливо актуально, наприклад, при встановленні датчиків для реєстрації різних параметрів у різних ділянках досліджуваної системи. Для такого випадку достатньо побудувати функцію Гріна для всієї системи. Далі, підставляючи в цю функцію конкретні координати точки виходу ξ отримаємо передатну функцію між входом та цією точкою виходу. Таким чином, маючи одну функцію Гріна, можна визначити безліч передатних функцій для різних точок виходу.

Цей метод можна також застосовувати для багатовимірних систем, де диференціальні оператори залежать від кількох просторових координат. Однак, побудова функції Гріна для таких операторів є значно складнішою задачею. Тому для багатовимірних систем часто використовують інші, більш ефективні методи, які будуть розглянуті далі.

Ефективним математичним інструментом для аналізу процесів у системах з розподіленими параметрами є дискретні функціональні перетворення, основою яких є дискретне перетворення Фур'є. Цей метод дозволяє перейти від просторово-часової області до частотно-просторової, що значно спрощує аналіз складних диференціальних рівнянь, які описують такі системи. Застосування дискретного перетворення Фур'є дозволяє розкласти функції, що описують процеси в системі, на гармонічні складові, що часто дозволяє отримати більш наочну інтерпретацію результатів і спростити подальший аналіз.

Матричний метод продовження розв'язку виявляється надзвичайно ефективним при визначенні передатних функцій складних систем, що включають як елементи з розподіленими параметрами (наприклад, балки, пластини), так і елементи із зосередженими параметрами (маси, пружини).

Матричний метод продовження розв'язку є універсальним інструментом для аналізу складних систем, який дозволяє ефективно обчислювати передатні функції. Завдяки матричному запису, метод забезпечує системний підхід до розв'язання задач і спрощує обчислення. Крім того, він дозволяє досліджувати не тільки однорідні системи, але й ті, що складаються з різних матеріалів або мають змінні геометричні характеристики, що значно розширює область його застосування.

Для кожної однорідної ділянки системи (прогону) складається система диференціальних рівнянь, яка описує її поведінку. Цю систему можна записати у матричній формі, отримавши так звану матрицю прогону. Умови з'єднання різних прогонів (наприклад, у вузлах балки) дозволяють скласти матрицю переходів. Поєднуючи матриці прогонів та переходів, можна отримати загальну матрицю системи, яка пов'язує вхідні та вихідні величини.

Матричний метод продовження розв'язку особливо ефективний при аналізі складних інженерних конструкцій. Його можна застосовувати до різноманітних систем, таких як багатопрогінні балки, де кожен прогін розглядається як окремий елемент, а потім усі вони з'єднуються за допомогою матриць переходів. Для просторових стержньових систем спочатку будується балочна модель, до якої потім застосовується цей метод. Крім того, метод дозволяє включати в розрахунок елементи з зосередженими параметрами, такі як маси та пружини.

Основні переваги матричного методу для аналізу складних стержньових систем полягають у його систематичному підході, що забезпечується матричним записом рівнянь. Це робить процес аналізу більш структурованим і зрозумілим. Крім того, матричні рівняння легко програмуються для розв'язання на комп'ютері, що значно спрощує і пришвидшує обчислення. Ще

однією важливою перевагою є можливість узагальнення методу для аналізу тривимірних конструкцій, що розширює сферу його застосування.

Розглянуті методи дозволяють отримати спрощені моделі систем з розподіленими параметрами у вигляді передатних функцій. Це означає, що складні процеси, які відбуваються в таких системах, можна описати за допомогою більш простих математичних моделей.

Однак, передатні функції, що описують динамічні системи з розподіленими параметрами, мають ряд особливостей:

- трансцендентні передатні функції – містять експоненти, гіперболічні функції (косинус, синус, тангенс) та їх обернені значення. Прикладами таких функцій є: $e^{(-p)}$, $ch(p)$, $sh(p)$, $th(p)$, $ctg(p)$.
- ірраціональні передатні функції – функції, які містять корені, наприклад: $\sqrt{p}$, $1/\sqrt{p}$, $\sqrt{(1+p)}$ тощо.

Складність таких функцій обумовлена тим, що вони описують процеси, які відбуваються в різних точках простору одночасно. Це відрізняє їх від систем із зосередженими параметрами, де всі процеси вважаються зосередженими в одній точці.

Спрощення моделей систем з розподіленими параметрами до передатних функцій дозволяє застосовувати стандартні методи аналізу та синтезу систем автоматичного керування. Однак, при цьому необхідно враховувати обмеження та похибки, пов'язані з такими спрощеннями.

Метод ланцюгових дробів є потужним інструментом в математичному апараті і знаходить все ширше застосування в різних областях науки і техніки. Його особлива цінність полягає у можливості наближеного представлення складних функцій, зокрема, тих, що описують динамічні системи з розподіленими параметрами [16].

Висновки до розділу 1

По-перше, метод структурного моделювання дозволяє перейти від абстрактного уявлення про систему до її візуального представлення у вигляді

графів або блок-схем. Це надає інструмент для чіткого визначення компонентів системи, їхніх взаємозв'язків та ролі в загальному функціонуванні. Такий візуальний підхід значно спрощує розуміння структури системи та допомагає виявити потенційні проблеми.

По-друге, передатні функції забезпечують математичний опис динамічних властивостей лінійних систем. Вони дозволяють встановити кількісний зв'язок між вхідними та вихідними сигналами системи, а також проаналізувати її реакцію на різні впливи. За допомогою передатних функцій можна досліджувати такі важливі характеристики системи, як стійкість, керованість та спостережуваність.

По-третє, метод ланцюгових дробів є ефективним інструментом для апроксимації складних функцій, які часто зустрічаються при описі систем з розподіленими параметрами. Цей метод дозволяє замінити складні трансцендентні або ірраціональні функції на більш прості раціональні дроби, що значно спрощує подальший аналіз. Ланцюгові дроби знаходять широке застосування в різних областях, зокрема, в теорії керування та обробці сигналів.

Поєднання цих трьох методів дозволяє проводити комплексний аналіз складних систем. Структурне моделювання дає візуальне уявлення про систему, передатні функції забезпечують математичний опис, а метод ланцюгових дробів дозволяє спростити складні математичні моделі. Такий підхід відкриває широкі можливості для дослідження та проектування складних систем.

РОЗДІЛ 2. СЕРЕДОВИЩЕ МОДЕЛЮВАННЯ

В попередньому розділі розглянуто теоретичні основи структурного моделювання, методу передатних функцій та методу ланцюгових дробів. Для ефективного застосування цих методів на практиці необхідно глибоко розглянути можливості існуючих інструментальних засобів. Тому в цьому розділі буде проведено детальний аналіз функціоналу MATLAB, зокрема, його можливостей щодо створення та апроксимації математичних моделей, проведення симуляцій та візуалізації результатів.

2.1. Загальна характеристика MATLAB

MATLAB (Matrix Laboratory) – це потужний інструмент для наукових обчислень, який зародився як інтерактивне середовище для роботи з матрицями. Створений у 1970-х роках Клівом Молером в Університеті Нью-Мексико. MATLAB швидко завоював популярність серед науковців та інженерів завдяки своїй ефективності у вирішенні складних математичних задач.

MATLAB був розроблений з метою спрощення роботи з матрицями, які є основою багатьох наукових і технічних обчислень. Спочатку, MATLAB був задуманий як інтерактивне середовище, що дозволяло користувачам швидко вводити команди та отримувати результати, на відміну від традиційних мов програмування, які вимагали тривалої підготовки та компіляції коду. Ця особливість значно пришвидшувала процес розробки та тестування алгоритмів, роблячи MATLAB незамінним інструментом для науковців та інженерів.

Початково створений для роботи з матрицями, MATLAB сьогодні є універсальною платформою для наукових обчислень. Він дозволяє не лише виконувати складні математичні операції, але й будувати математичні моделі різних систем, аналізувати великі обсяги даних, розробляти власні алгоритми та створювати інтерактивні інтерфейси для користувачів.

Сучасний MATLAB – це високопродуктивна платформа для наукових обчислень, що характеризується простотою використання, потужною бібліотекою вбудованих функцій, ефективною візуалізацією даних та можливостями інтеграції з іншими програмними системами. Завдяки цим особливостям, MATLAB широко застосовується в різних галузях науки і техніки для вирішення складних математичних задач, моделювання систем, аналізу даних та розробки алгоритмів.

MATLAB базується на зручній для користувача матрично-орієнтованій мові програмування. Ця мова дозволяє ефективно виконувати складні математичні операції та аналізувати дані завдяки інтерактивному режиму роботи та простому синтаксису, який наближений до математичної нотації. Це робить MATLAB незамінним інструментом для науковців та інженерів, які потребують швидкого прототипування та аналізу результатів.

Середовище MATLAB забезпечує зручну платформу для роботи з програмним забезпеченням. Основними компонентами середовища є робочий простір, де користувач безпосередньо взаємодіє з системою, редактор для створення та редагування програмних кодів (М-файлів), командний рядок для введення індивідуальних команд та інструменти відладки для виявлення та усунення помилок у програмах. Ця інтегрована система дозволяє ефективно розробляти, виконувати та аналізувати різноманітні обчислювальні завдання.

MATLAB пропонує потужні інструменти для візуалізації даних, що дозволяють перетворювати цифрову інформацію на зрозумілі графіки, діаграми та навіть анімації. Ці графічні представлення даних не лише полегшують розуміння результатів обчислень, але й дозволяють користувачам інтерактивно досліджувати дані, змінюючи масштаб, обертаючи тривимірні моделі та застосовуючи різноманітні параметри. Це робить MATLAB незамінним інструментом для візуального аналізу та презентації результатів досліджень.

MATLAB пропонує багату бібліотеку вбудованих функцій для широкого спектру математичних операцій, статистичного аналізу, обробки

сигналів та лінійної алгебри. Ця колекція функцій постійно розширюється і доповнюється спеціалізованими інструментами та додатковими пакетами, такими як Control System Toolbox, Signal Processing Toolbox, Statistics and Machine Learning Toolbox, Optimization Toolbox, що дозволяють вирішувати більш складні завдання. Особливе місце серед цих інструментів займає Control System Toolbox, який є незамінним для інженерів-автоматиків. Він надає потужні можливості для проектування, аналізу та симуляції систем автоматичного керування. За допомогою Control System Toolbox можна будувати моделі систем у часовій та частотній областях, аналізувати стійкість систем, синтезувати регулятори та проводити ідентифікацію систем за експериментальними даними. Крім того, MATLAB містить численні бібліотеки для роботи зі структурними моделями, передатними функціями та іншими інструментами, що робить його незамінним інструментом для моделювання та аналізу складних систем. Наприклад, Simulink – це спеціалізований інструмент, що дозволяє створювати блок-схеми моделей динамічних систем і проводити їх симуляцію. Поєднання цих можливостей робить MATLAB потужним інструментом для розробки та аналізу систем автоматичного керування в різних галузях, від промислової автоматизації до робототехніки.

MATLAB не лише потужний інструмент для наукових обчислень, але й платформа для розробки власних додатків. Він дозволяє створювати інтуїтивно зрозумілі графічні інтерфейси, які полегшують взаємодію користувача з програмою. Крім того, MATLAB надає можливість генерувати код на інших мовах програмування, таких як C, C++ чи Java, що дозволяє оптимізувати швидкість виконання або інтегрувати розроблені додатки з іншими системами.

Отже, ключовими переваги MATLAB є ефективні матричні обчислення, якісні засоби візуалізації даних, можливість символьних обчислень та активна спільнота користувачів [6, 12, 13].

Розглянемо далі ключові можливості MATLAB для моделювання динамічних об'єктів.

2.2. Побудова та аналіз моделей динамічних систем в MATLAB

MATLAB є потужним інструментом для моделювання та аналізу динамічних систем, що має широкий спектр застосування в науці та техніці. Моделювання динамічних систем дозволяє передбачати поведінку систем у різних умовах, оптимізувати їх роботу, проектувати нові системи та аналізувати складні процеси, які важко досліджувати експериментально. Завдяки цим можливостям, MATLAB став незамінним інструментом для інженерів, науковців та розробників у різних сферах.

Control System Toolbox – це потужний інструмент, вбудований в MATLAB, який спеціалізується на моделюванні, аналізі та проектуванні систем автоматичного керування. Цей інструментарій надає користувачам широкий спектр функцій для роботи з лінійними стаціонарними системами, що робить його незамінним помічником для інженерів та науковців, які займаються розробкою та дослідженням різноманітних систем керування.

Основне призначення Control System Toolbox полягає в тому, щоб надати користувачам зручний та ефективний інструмент для: створення математичних моделей систем (Toolbox дозволяє задавати системи за допомогою різних математичних моделей: передатних функцій, простору станів, нулів, полюсів та коефіцієнтів підсилення), аналізу властивостей систем (можна досліджувати стійкість систем, аналізувати їхню частотну характеристику, будувати графіки перехідних процесів тощо), синтезу систем керування (надаються функції для проектування регуляторів, спостерігачів та інших елементів системи керування), симуляції систем (моделі, створені в Toolbox, можуть бути використані для симуляції поведінки системи в різних умовах).

Основні типи моделей, що підтримуються Control System Toolbox використовуються для опису лінійних стаціонарних систем: передатні функції

(найпростіший спосіб представлення системи, який описує співвідношення між виходом системи та її входом у вигляді раціональної дробової функції від оператора Лапласа), простір станів (модель описує систему за допомогою системи лінійних диференціальних рівнянь першого порядку), нулі, полюси та коефіцієнт підсилення (форма представлення моделі використовується для опису передатної функції через її нулі, полюси та коефіцієнт підсилення). Кожен з цих типів моделей має свої переваги та недоліки, і вибір конкретного типу залежить від особливостей модельованої системи та поставлених задач.

Control System Toolbox надає широкий спектр функцій для маніпуляції та аналізу моделей динамічних систем. Розглянемо деякі з найважливіших операцій.

Control System Toolbox дозволяє створювати складні моделі систем шляхом з'єднання більш простих компонентів. Функція *series* використовується для послідовного з'єднання моделей, коли вихід однієї системи стає входом для іншої. Для паралельного з'єднання, де обидві системи мають спільний вхід і вихід, застосовується функція *parallel*. Системи зі зворотним зв'язком, де частина виходу повертається на вхід, моделюються за допомогою функції *feedback*. Ці операції дозволяють точно відтворити поведінку реальних систем, що складаються з багатьох взаємопов'язаних елементів.

Control System Toolbox забезпечує гнучкість у роботі з моделями, дозволяючи легко переходити між різними формами представлення. Для адаптації моделей до конкретних завдань можна виконувати дискретизацію неперервних моделей за допомогою функції *c2d*, що є необхідним кроком для моделювання цифрових систем керування. Крім того, Toolbox надає функції для перетворення між передатними функціями, простором станів та представленням через нулі, полюси і коефіцієнт підсилення (наприклад, *tf2ss*, *ss2tf*, *zp2ss*, *ss2zp*), що дозволяє вибрати найбільш зручну форму для аналізу та синтезу системи.

Control System Toolbox надає потужні інструменти для всебічного аналізу моделей динамічних систем. Є можливість досліджувати стійкість системи за допомогою функцій *isstable* та *pole*, аналізувати її поведінку в часі за допомогою перехідної та імпульсної характеристик (функції *step* та *impulse*), а також оцінювати керованість та спостережуваність системи за допомогою функцій *ctrb* та *obsv* відповідно. Для вивчення частотних характеристик системи можна використовувати функції *bode* та *nyquist*. Ці інструменти дозволяють отримати детальну інформацію про поведінку системи і приймати обґрунтовані рішення щодо проектування систем керування.

Приклад:

```
% Створення двох моделей
sys1 = tf([1], [1 2 1]);
sys2 = tf([2 1], [1 3]);
% Послідовне з'єднання
sys_series = series(sys1, sys2);
% Аналіз стійкості
isstable(sys_series)
% Побудова частотної характеристики
bode(sys_series);
```

Цей приклад демонструє, як створити дві моделі, з'єднати їх послідовно і перевірити стійкість отриманої системи.

Давайте розглянемо кілька прикладів типових задач, які можна вирішувати за допомогою цього інструментарію.

За допомогою Control System Toolbox можна проектувати системи регулювання, аналізувати їх стійкість та динамічні характеристики, оптимізувати параметри, ідентифікувати математичні моделі реальних систем та проводити їх симуляцію, зокрема в таких галузях як робототехніка, авіація, автомобілебудування, енергетика, процесна промисловість та біомедицина.

Наприклад, у робототехніці він використовується для моделювання рухів роботів та створення систем керування, які забезпечують точність і

стабільність їх роботи. В авіації цей інструмент застосовується для аналізу систем керування літаками та розробки алгоритмів автоматичного пілотування. Автомобільна промисловість використовує Control System Toolbox для моделювання систем ABS, ESP та інших систем безпеки, а також для розробки систем адаптивного круїз-контролю. Енергетика, процесна промисловість та біомедицина – це лише деякі з інших галузей, де Control System Toolbox знаходить своє застосування для розробки ефективних систем керування та аналізу складних процесів.

Отже, Control System Toolbox є потужним інструментом для моделювання, аналізу та синтезу систем автоматичного керування, який знаходить широке застосування в різних галузях науки і техніки, і продовжує розвиватися, інтегруючи нові алгоритми та методи [1, 3, 4, 5, 10, 11].

2.3. Середовище імітаційного моделювання Simulink

Simulink – це потужний інструмент візуального програмування, який є додатком до пакету MATLAB. Він призначений для моделювання, симуляції та аналізу динамічних систем різного типу: від простих електричних схем до складних систем керування, які використовуються в авіації, автомобілебудуванні, робототехніці та інших галузях.

Ключовою особливістю Simulink є використання блок-схем для представлення моделей. Кожен блок у схемі відповідає певному елементу системи (наприклад, суматор, інтегратор, передатна функція), а з'єднання між блоками визначають взаємодію між цими елементами. Такий підхід робить процес моделювання більш інтуїтивним та зрозумілим, особливо для інженерів, які не є досвідченими програмістами.

Simulink пропонує низку переваг, серед них: інтуїтивно зрозумілий графічний інтерфейс для створення моделей, можливість моделювати різноманітні системи, включаючи аналогові, дискретні та гібридні, широка бібліотека готових блоків, тісна інтеграція з MATLAB для потужних обчислень і аналізу, а також можливість симуляції в реальному часі та

автоматичного генерування коду. Крім того, Simulink є відкритою платформою, що дозволяє користувачам створювати власні блоки та розширювати його функціональність.

Simulink, як невід'ємна частина MATLAB, забезпечує тісну інтеграцію з цим потужним інструментом математичних обчислень. Це дозволяє ефективно обмінюватися даними між середовищами, використовувати в моделях Simulink широкий спектр математичних функцій і алгоритмів, розроблених у MATLAB, а також візуалізувати результати симуляції за допомогою різноманітних графічних інструментів MATLAB. Така інтеграція значно розширює можливості моделювання та аналізу систем, надаючи користувачам єдине середовище для розв'язання складних інженерних задач.

Завдяки тісній інтеграції з MATLAB, Simulink забезпечує потужний інструментарій для моделювання, аналізу та розробки різноманітних систем.

Вікно Simulink пропонує зручний інтерфейс для створення та редагування моделей. Воно включає панель меню з основними командами, панель інструментів для швидкого доступу до функцій, браузер бібліотеки з різноманітними блоками, робоче поле для побудови моделі та панель властивостей для налаштування параметрів. Така організація інтерфейсу спрощує роботу з програмним забезпеченням і дозволяє користувачам ефективно створювати моделі різної складності.

Бібліотека Simulink містить різноманітні блоки, структуровані за категоріями (неперервні системи, дискретні системи, математичні операції, нелінійні елементи, сигнали, джерела та приймачі сигналів, підсистеми), що дозволяє легко знайти потрібні елементи для побудови моделі.

Створення моделі в Simulink відбувається у кілька етапів: створення нового файлу, додавання блоків з бібліотеки, з'єднання блоків між собою, налаштування параметрів, симуляція та візуалізація результатів.

Перед симуляцією в Simulink необхідно налаштувати такі параметри, як вибір солвера, інтервал часу симуляції та точність обчислень. Ці параметри впливають на ефективність та точність отриманих результатів.

Після симуляції важливо проаналізувати отримані результати, порівнявши їх з експериментальними даними, провівши частотний аналіз, дослідивши чутливість моделі до змін параметрів та оптимізувавши її характеристики.

Щоб зрозуміти результати симуляції в Simulink, можна використовувати різноманітні інструменти візуалізації. Найчастіше застосовують графіки для відображення зміни вихідних сигналів з часом. Крім того, результати можна представити у вигляді таблиць, 3D-графіків або навіть анімацій, що дозволяє детально дослідити динаміку моделі.

Створення користувацьких блоків у Simulink надає користувачам значну гнучкість та потужність. Це дозволяє розширити функціональність середовища моделювання за допомогою створення власних блоків, які точно відповідають специфічним потребам моделі. Процес створення передбачає написання MATLAB-функції, що реалізує логіку блоку, а також оформлення його зовнішнього вигляду та інтерфейсу для введення параметрів. Такий підхід сприяє модульності моделей, повторному використанню створених блоків та захисту інтелектуальної власності шляхом приховування внутрішньої логіки.

Simulink, як потужний інструмент моделювання, забезпечує глибоку інтеграцію з іншими програмними системами, розширюючи таким чином свої можливості. Це дозволяє підключати сторонні бібліотеки, взаємодіяти з базами даних, генерувати код для різних платформ та інтегруватися з іншими інструментами моделювання. Такі можливості досягаються за допомогою різних механізмів, серед яких MATLAB Function, S-функції, COM-об'єкти та Simulink Coder. Завдяки інтеграції, користувачі отримують доступ до широкого спектра інструментів та бібліотек, що значно спрощує процес моделювання та дозволяє створювати більш складні та реалістичні моделі. Крім того, інтеграція автоматизує рутинні операції та забезпечує безперервність процесу розробки від моделювання до реалізації на фізичному обладнанні.

Можливість створення користувацьких блоків та інтеграція з іншими програмними продуктами роблять Simulink потужним інструментом для моделювання складних систем. Це дозволяє адаптувати Simulink до конкретних завдань і створювати індивідуальні рішення [7, 10, 11].

2.4. Інструменти апроксимації в Matlab

Апроксимація в моделюванні динамічних систем є важливим інструментом, який дозволяє спрощувати складні математичні моделі, зберігаючи при цьому їх основні характеристики. Це особливо актуально при моделюванні реальних фізичних процесів, де точне математичне описання може бути надто складним або навіть неможливим [2].

Перелік основних функцій для апроксимації в Matlab:

- *polyfit*: будує поліноміальну апроксимацію даних методом найменших квадратів;
- *interpft*: здійснює інтерполяцію періодичних функцій за допомогою швидкого перетворення Фур'є;
- *icubic*: виконує кубічну інтерполяцію функцій однієї змінної;
- *spline*, *ppval*, *mkpp*, *unmkpp*: функції пов'язані між собою і використовуються для кубічної сплайн-інтерполяції;
- *interp1*: одновимірна таблична інтерполяція;
- *interp2*: двовимірна таблична інтерполяція;
- *griddata*: двовимірна таблична інтерполяція на нерівномірній сітці.

Хоча Matlab пропонує широкий спектр функцій для апроксимації даних та функцій, більшість з них орієнтовані на роботу з дійсними функціями однієї або декількох змінних. Це робить їх ефективними для апроксимації експериментальних даних та простих математичних моделей. Однак, при роботі зі складними динамічними системами, які описуються диференціальними рівняннями або передатними функціями, виникають обмеження. Справа в тому, що такі системи часто демонструють нелінійні характеристики, мають розподілені параметри або характеризуються високою

розмірністю. Для адекватної апроксимації таких систем потрібні більш спеціалізовані методи та інструменти, які виходять за рамки стандартних функцій Matlab.

Функції *balreal* та *modred* є потужними інструментами в Matlab для спрощення моделей динамічних систем. *balreal* обчислює збалансовану модель, де стани впорядковані за їхнім впливом на вихід системи. Це дозволяє виявити ті стани, які мають найменший вплив і можуть бути відкинуті без значної втрати точності моделі. Функція *modred* використовує результати *balreal* для зменшення порядку моделі шляхом відкидання малозначущих станів. Такий підхід дозволяє отримати спрощену модель, яка легше для аналізу та синтезу, але при цьому зберігає основні характеристики оригінальної системи.

Розглянемо модель системи керування, яка описується передатною функцією другого порядку. Для її спрощення виконаємо такі кроки в Matlab:

```
% Створення моделі системи керування
num = [1 2];
den = [1 5 4];
sys = tf(num, den);
% Обчислення збалансованої моделі
[sysb,g,t] = balreal(sys);
% Візуалізація сингулярних чисел для оцінки впливу
станів
plot(g);
% Зниження порядку моделі, відкидаючи стан з
найменшим впливом
sysr = modred(sysb, length(g)-1);
```

Функція *minreal* у MATLAB використовується для спрощення передатних функцій шляхом видалення спільних нулів та полюсів. Це дозволяє отримати більш компактну та обчислювально ефективну модель системи. Принцип роботи *minreal* полягає в тому, що функція аналізує

чисельник і знаменник передатної функції та виявляє спільні множники. Якщо ці множники знаходяться достатньо близько один до одного (враховуючи задану точність), вони скорочуються.

Розглянемо приклад, де передатна функція описує систему з декількома полюсами, деякі з яких є комплексно спряженими:

```
% Створення моделі з комплексно спряженими полюсами
num = [1 3 5];
den = conv([1 2 2], [1 4 13]); % Конволюція для
отримання полінома зі складними коренями
sys = tf(num, den);
% Спрощення моделі
sysr = minreal(sys);
```

Функція *pade* в MATLAB використовується для апроксимації запізнень у динамічних системах за допомогою раціональних функцій. Запізнення часто зустрічаються в реальних системах і можуть ускладнювати аналіз та синтез систем керування. Принцип роботи *pade* полягає в тому, що функція використовує розклад Паде для апроксимації експоненціальної функції, яка відповідає за запізнення. Порядок апроксимації задається користувачем і визначає точність апроксимації. Чим вищий порядок, тим точніше апроксимація, але тим складнішою стає передатна функція.

Приклад.

Розглянемо систему з запізненням 0.5 секунди. Для апроксимації цього запізнення раціональною функцією другого порядку виконаємо наступні дії в MATLAB:

```
% Апроксимація запізнення 0.5 з точністю до 2-го
порядку
[num, den] = pade(0.5, 2);
sys = tf(num, den);
```

Вибір функції для апроксимації залежить від конкретної задачі та характеристик моделі. Важливо розуміти, що спрощення моделі може

призвести до втрати точності, особливо в деяких областях частот. Тому необхідно проводити аналіз похибки апроксимації [7].

Висновки до розділу 2.

Підсумовуючи, MATLAB, Control System Toolbox та Simulink є потужними інструментами для аналізу та синтезу систем автоматичного керування. Однак, існують певні обмеження, особливо при роботі зі складними моделями, які описуються ірраціональними передатними функціями. В такому випадку потрібно застосовувати методи апроксимації.

Незважаючи на широкий спектр функцій для апроксимації, Matlab має певні обмеження при роботі зі складними моделями. Зокрема, недостатньо інструментів для апроксимації моделей з ірраціональними передатними функціями. Такі функції часто зустрічаються при описі нелінійних систем, систем з розподіленими параметрами або систем зі змінними параметрами. Стандартні методи апроксимації, реалізовані в Matlab, зазвичай орієнтовані на раціональні передатні функції, що обмежує їх застосування для більш складних випадків.

Для ефективної апроксимації складних динамічних систем, які описуються ірраціональними передатними функціями, необхідно розробляти додаткові методи та функції. Це може включати в себе розробку нових алгоритмів апроксимації, для роботи з ірраціональними функціями, наприклад, заснованих на методах нелінійної оптимізації або методів ланцюгових дробів.

РОЗДІЛ 3. АЛГОРИТМИ АПРОКСИМАЦІЇ НА ОСНОВІ ЛАНЦЮГОВИХ ДРОБІВ

3.1. Алгоритми побудови ланцюгових дробів для апроксимації моделей

Розглянемо складну передатну функцію $G(s)$. Розклавши її в степеневий ряд за степенями s , отримаємо:

$$G(s) = b_0 + b_1s + b_2s^2 + \dots = \sum_{k=0}^{\infty} b_k s^k. \quad (3.1)$$

Обмеживши цей ряд першими $N + 1$ членами, отримаємо поліноміальну апроксимацію:

$$G_N(s) = b_0 + b_1s + b_2s^2 + \dots + b_Ns^N = \sum_{k=0}^N b_k s^k. \quad (3.2)$$

Для полінома $G_N(s)$ можна знайти такий ланцюговий дріб, що розклад будь-якого його n -го підхідного дробу в степеневий ряд збігатиметься з вихідним рядом (3.2) до члена s^n включно.

Побудуємо ланцюговий дріб, який відповідає степеневому ряду (3.2):

$$G_N(s) = b_0 + b_1s + b_2s^2 + \dots + b_Ns^N = \sum_{k=0}^N b_k s^k. \quad (3.3)$$

Зауважимо, що всі подальші перетворення будуть формальними, тобто не будуть розглядатися питання збіжності отриманого ланцюгового дробу.

Перепишемо ряд (3.3) у вигляді:

$$G_N(s) = b_0 + s \cdot \left(b_1 + s \cdot \left(b_2 + s \cdot (\dots + s \cdot b_N) \right) \right),$$

де внутрішні дужки утворюють рекурсивну структуру.

Введемо заміну:

$$f_N(s) = b_N + s \cdot (b_{N-1} + s \cdot (\dots + s \cdot b_1)).$$

Помноживши обидві частини рівності на s і прирівнявши коефіцієнти при однакових степенях s , отримаємо систему рівнянь для визначення коефіцієнтів f^k :

$$\begin{cases} b_N = f_N(0) \\ b_{N-1} = f_{N-1}(0) \\ \vdots \\ b_1 = f_1(0) \end{cases}$$

Таким чином, ми можемо виразити b^k через $f_k(0)$.

Запишемо $G_N(s)$ у вигляді:

$$G_N(s) = b_0 + s \cdot f_N(s).$$

Введемо нову заміну:

$$f_{N-1}(s) = b_{N-1} + s \cdot (b_{N-2} + s \cdot (\dots + s \cdot b_1)).$$

Повторивши аналогічні дії отримаємо послідовність заміни, яка приведе до наступного ланцюгового дробу:

$$G_N(s) = b_0 + s \cdot \frac{1}{1 + \frac{s}{f_1(0) + \frac{s}{f_2(0) + \dots + \frac{s}{f_N(0)}}}}.$$

Коефіцієнти $f_k(0)$ визначаються через коефіцієнти вихідного ряду b_k .

3.2. Алгоритми побудови правильних С-дробів

Розглянемо степеневий ряд:

$$f(z) = \sum_{n=0}^{\infty} a_n z^n$$

Необхідно побудувати правильний С-дріб для цього ряду.

Для заданого ряду визначник Ганкеля порядку k визначається як:

$$H_k = \begin{vmatrix} a_0 & a_1 & \dots & a_{k-1} \\ a_1 & a_2 & \dots & a_k \\ \vdots & \vdots & \ddots & \vdots \\ a_{k-1} & a_k & \dots & a_{2k-2} \end{vmatrix}$$

Шукаємо правильний С-дріб у вигляді:

$$f(z) \sim \frac{c_1 z}{1 + \frac{c_2 z}{1 + \frac{c_3 z}{1 + \dots}}}$$

Коефіцієнти c_n обчислюються за наступними формулами:

$$c_1 = \frac{H_1}{H_0},$$

$$c_n = \frac{H_n H_{n-2}}{H_{n-1}^2}, \quad n \geq 2.$$

3.3. Оптимізований алгоритм побудови дробово-раціональних передатних функцій

Розглянемо функцію $f(s)$, яку ми бажаємо апроксимувати дробово-раціональною функцією.

Алгоритм можна представити у вигляді наступних шести кроків:

1. *Заміна змінної (якщо необхідно).* Якщо точка $s = 0$ є особливою точкою для функції $f(s)$, виконаємо заміну змінної:

$$s = \frac{1}{z}$$

Отримаємо нову функцію $g(z) = f(1/z)$.

2. *Розклад у степеневий ряд.* Розвинемо функцію $g(z)$ в степеневий ряд в околі точки $z = 0$:

$$g(z) = \sum_{n=0}^{\infty} a_n z^n$$

3. *Обрізання ряду.* Залишимо перші N членів степеневого ряду.
4. *Перетворення в ланцюговий дріб.* Застосуємо один з відомих алгоритмів перетворення степеневого ряду в ланцюговий дріб, що розглянуті вище:
 - Аналог формули Тейлора для ланцюгових дробів.
 - Алгоритм побудови правильних С-дробів.
5. *Побудова підхідного дробу.* За коефіцієнтами отриманого ланцюгового дробу побудуємо його підхідний дріб $R_N(z)$, який є дробово-раціональною функцією.
6. *Зворотна заміна.* Якщо виконувалася заміна змінної на кроці 1, виконаємо зворотну заміну $z = 1/s$ у функції $R_N(z)$, отримавши апроксимацію вихідної функції $f(s)$.

Для оцінки точності апроксимації використовуємо інтегральні норми похибок амплітудно-частотної характеристики (АЧХ) та фазо-частотної характеристики (ФЧХ):

$$\int_0^{\omega_m} |H(j\omega) - R_N(j\omega)|^2 d\omega \leq \varepsilon_A,$$

$$\int_0^{\omega_m} |\angle H(j\omega) - \angle R_N(j\omega)|^2 d\omega \leq \varepsilon_\phi$$

де:

- $H(j\omega)$ - точна частотна характеристика,
- $R_N(j\omega)$ - частотна характеристика підхідного дробу,
- ω_m - максимальна частота,
- ε_A та ε_ϕ - задані похибки для АЧХ та ФЧХ відповідно.

Порядок N підхідного дробу підбирається таким чином, щоб виконувалися нерівності для заданих похибок ε_A та ε_ϕ .

Запропонований підхід до побудови дробово-раціональних апроксимацій має ряд переваг. По-перше, він універсальний і може бути застосований до широкого спектра функцій. По-друге, він забезпечує гнучкість у доборі порядку апроксимації та точності наближення. І, нарешті, існують ефективні алгоритми для обчислення коефіцієнтів ланцюгових дробів, що робить цей метод обчислювально ефективним.

Цей алгоритм може широко використовуватись в теорії керування, обробці сигналів та інших областях для апроксимації передатних функцій систем.

Обираючи алгоритм перетворення степеневого ряду в ланцюговий дріб, слід враховувати властивості функції, бажану точність апроксимації та доступні обчислювальні ресурси. При цьому важливо пам'ятати про питання чисельної стійкості, особливо при обчисленні визначників Ганкеля, що можуть виникати під час реалізації алгоритму на комп'ютері.

3.4. Алгоритм отримання коефіцієнтів ланцюгових дробів

Розглянемо передатну функцію у вигляді:

$$G(s) = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_m}{a_0 s^n + a_1 s^{n-1} + \dots + a_n}$$

Позначимо:

$$b(s) = b_0 s^m + b_1 s^{m-1} + \dots + b_m$$

$$a(s) = a_0 s^n + a_1 s^{n-1} + \dots + a_n$$

Тоді передатну функцію можна записати як:

$$G(s) = \frac{b(s)}{a(s)}$$

Виконуючи довгий поділ поліномів, отримаємо:

$$G(s) = c_0 + \frac{r_1(s)}{a(s)}$$

де c_0 – частка від ділення, а $r_1(s)$ – остача.

Повторюючи цю процедуру для дробу $r_1(s)/a(s)$, отримаємо:

$$G(s) = c_0 + \frac{1}{c_1 + \frac{r_2(s)}{r_1(s)}}$$

Продовжуючи цей процес отримаємо ланцюговий дріб у вигляді:

$$G(s) = c_0 + \frac{1}{c_1 + \frac{1}{c_2 + \frac{1}{c_3 + \dots}}}$$

Коефіцієнти c_i ланцюгового дробу можуть бути знайдені за допомогою алгоритму Евкліда для поліномів.

Коефіцієнти c_i можна також подати у вигляді матриці:

$$\begin{bmatrix} a_0 & a_1 & a_2 & \dots & a_n \\ b_0 & b_1 & b_2 & \dots & b_m \end{bmatrix}$$

Ця матриця пов'язана з визначником Рауса і дозволяє ефективно обчислювати коефіцієнти ланцюгового дробу.

Висновок до розділу 3.

У третьому розділі були детально розглянуті різноманітні алгоритми, що дозволяють ефективно будувати апроксимації моделей за допомогою ланцюгових дробів. Представлені алгоритми охоплюють широкий спектр застосувань, від загальних методів побудови ланцюгових дробів до спеціалізованих алгоритмів для отримання правильних S -дробів та дробово-раціональних передатних функцій.

Особливу увагу було приділено алгоритму отримання коефіцієнтів ланцюгових дробів, який базується на довгому поділі поліномів та пов'язаний з визначником Рауса. Цей алгоритм дозволяє ефективно обчислювати коефіцієнти ланцюгового дробу для заданої передатної функції.

РОЗДІЛ 4. ПРОГРАМНІ ЗАСОБИ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ

4.1. Огляд розроблених модулів

На основі розглянутих у попередніх розділах алгоритмів побудови ланцюгових дробів були розроблені спеціальні модулі для середовища MATLAB. Ці модулі розширюють функціональність MATLAB, надаючи користувачам потужний інструмент для аналізу та синтезу систем на основі дробово-раціональних моделей.

Модуль cfrac_approx

Призначення: Побудова апроксимації заданої передатної функції за допомогою ланцюгового дробу.

Синтаксис:

```
[num, den] = cfrac_approx('func', st);
```

- *func* – символьний вираз, що описує передатну функцію;
- *st* – порядок апроксимації (ступінь ланцюгового дробу);
- *num, den* – чисельник та знаменник отриманого ланцюгового дробу відповідно.

Модуль реалізує алгоритм побудови ланцюгового дробу, описаний у розділі 3. На вході приймає передатну функцію у символьному вигляді та порядок апроксимації. На виході видає чисельник та знаменник ланцюгового дробу, який є найкращим наближенням заданої передатної функції в сенсі мінімізації норми похибки.

Модуль cfrac_coef

Призначення: Обчислення коефіцієнтів ланцюгового дробу за заданими чисельником та знаменником.

Синтаксис:

```
h = cfrac_coef(num, den);
```

- num, den – чисельник та знаменник передатної функції;
- h – вектор коефіцієнтів ланцюгового дробу.

Модуль використовує алгоритм, описаний у розділі 3, для обчислення коефіцієнтів ланцюгового дробу за заданими чисельником та знаменником. Отримані коефіцієнти можуть бути використані для подальшого аналізу та синтезу систем.

Модуль cfrac_param

Призначення: Налаштування параметрів ланцюгового дробу.

Синтаксис:

$$h = \text{cfrac_param}(za);$$

- za – значення параметра, що впливає на коефіцієнти ланцюгового дробу;
- h – коефіцієнти ланцюгового дробу.

Модуль призначений для автоматизації процесу обчислення коефіцієнтів ланцюгових дробів при зміні значень параметрів системи. На вході приймає значення параметра, на виході видає вектор коефіцієнтів ланцюгового дробу, обчислених для заданого значення параметра.

4.2. Обчислювальні експерименти побудови ланцюгових дробів

Розглянемо методику використання даних засобів на конкретному прикладі.

Нехай маємо передатну функцію:

$$G(s) = e^{-as}$$

Бажаємо отримати апроксимаційне наближення 3-го порядку за допомогою ланцюгового дробу.

Застосування модуля *cfrac_approx*:

```
syms a p;
```

```
func = exp(-a*p);
```

```
st = 3;
```

```
[num, den] = cfrac_approx(func, st);
```

У результаті отримаємо чисельник *num* та знаменник *den* апроксимуючого ланцюгового дробу, які будуть виражені через символ *a*.

Отримання коефіцієнтів ланцюгового дробу:

```
h = cfrac_coef(num, den);
```

Отримаємо вектор *h*, елементи якого є коефіцієнтами ланцюгового дробу і залежать від параметра *a*. Наприклад, можемо отримати:

```
h = [1, 1/a, -2, -3/a, 2, 5/a];
```

Побудований ланцюговий дріб:

$$G_{approx}(s) = 1 + \frac{1}{\frac{1}{a} + \frac{1}{-2 + \frac{1}{-\frac{3}{a} + \frac{1}{2 + \frac{1}{\frac{5}{a}}}}}}$$

Для $a = 1$:

$$G_{approx}(s) = 1 + \frac{1}{1 + \frac{1}{-2 + \frac{1}{-3 + \frac{1}{2 + \frac{1}{5}}}}}}$$

Для $a = 2$:

$$G_{approx}(s) = 1 + \frac{1}{0.5 + \frac{1}{-2 + \frac{1}{-1.5 + \frac{1}{2 + \frac{1}{2.5}}}}}}$$

Замість експоненти можна використовувати інші функції, такі як раціональні дроби, тригонометричні, ірраціональні передатні функції тощо. Змінюючи значення параметра *st*, можна отримувати апроксимації різного

порядку. Можна обчислити похибку апроксимації та проаналізувати її залежність від частоти та порядку апроксимації.

Отже, за допомогою розроблених модулів можна ефективно будувати апроксимації складних передатних функцій за допомогою ланцюгових дробів. Отримані моделі можуть бути використані для подальшого аналізу та синтезу систем.

4.3. Імітаційне моделювання розподілених ланок на основі ланцюгових дробів

Розглянемо застосування розроблених методів ланцюгово-дробової апроксимації для імітаційного моделювання розподілених ланок, що описуються ірраціональними та трансцендентними передатними функціями.

Нехай маємо ланцюговий дріб, отриманий в результаті апроксимації:

$$G(s) \approx 1 + \frac{1}{a_1 + \frac{s}{a_2 + \frac{s}{\ddots + \frac{s}{a_n}}}}$$

Обмежившись першими шістьма членами, отримаємо:

$$G(s) \approx 1 + \frac{1}{a_1 + \frac{s}{a_2 + \frac{s}{a_3 + \frac{s}{a_4 + \frac{s}{a_5 + \frac{s}{a_6}}}}}}$$

Цей ланцюговий дріб можна легко реалізувати у вигляді схеми в Simulink (рис. 3.1). Кожен член ланцюгового дробу відповідає послідовному з'єднанню інтегратора та підсилювача з коефіцієнтом, оберненим до відповідного коефіцієнта a_i [19].

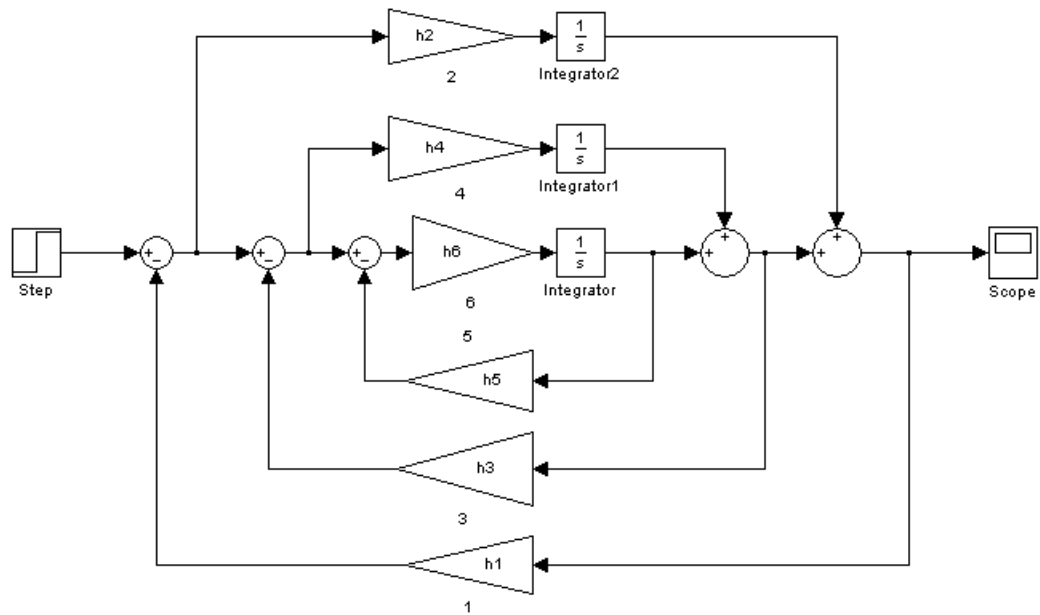


Рис. 3.1. Схема Simulink-моделі

Приклади.

Для передатної функції $G(s) = e^{(-as)}$ за умови дробово-раціональної апроксимації 32-го порядку отримано 64 коефіцієнти ланцюгового дробу:

$$h = [1, 1/a, -2, -3/a, 2, 5/a, \dots, 2, 61/a, -2, -63/a]$$

Ці коефіцієнти використовуємо для побудови відповідної схеми в Simulink.

Для передатної функції $G(s) = \frac{1}{\sqrt{as+1}}$ коефіцієнти апроксимаційного ланцюгового дробу мають такий вигляд:

$$h = [a^{(1/2)}, 2*a^{(1/2)}, 2*a^{(1/2)}, \dots, 2*a^{(1/2)}]$$

Запропонований підхід до моделювання за допомогою ланцюгових дробів має низку переваг. По-перше, він забезпечує гнучкість, дозволяючи моделювати широкий спектр систем завдяки різноманітності можливих ланцюгових дробів. По-друге, реалізація таких моделей у середовищі Simulink є досить простою та масштабованою для будь-якого порядку апроксимації. І, нарешті, при достатньо високому порядку апроксимації можна досягти

високої точності моделювання, що робить цей метод ефективним інструментом для дослідження та аналізу складних систем.

Незважаючи на переваги, метод моделювання за допомогою ланцюгових дробів має й певні обмеження. Зокрема, при використанні високих порядків апроксимації структура моделі може ставати надмірно складною, що ускладнює її аналіз та реалізацію.

4.4. Оптимізація структури моделей в Simulink

Побудова схеми в Simulink для великої кількості коефіцієнтів є трудомістким процесом і схильна до помилок. Використання підсистем дозволяє розбити складну модель на більш прості підмодулі, що значно полегшує її створення, розуміння та модифікацію. Наприклад, для моделі третього порядку достатньо п'яти підсистем (рис. 3.2-3.6).

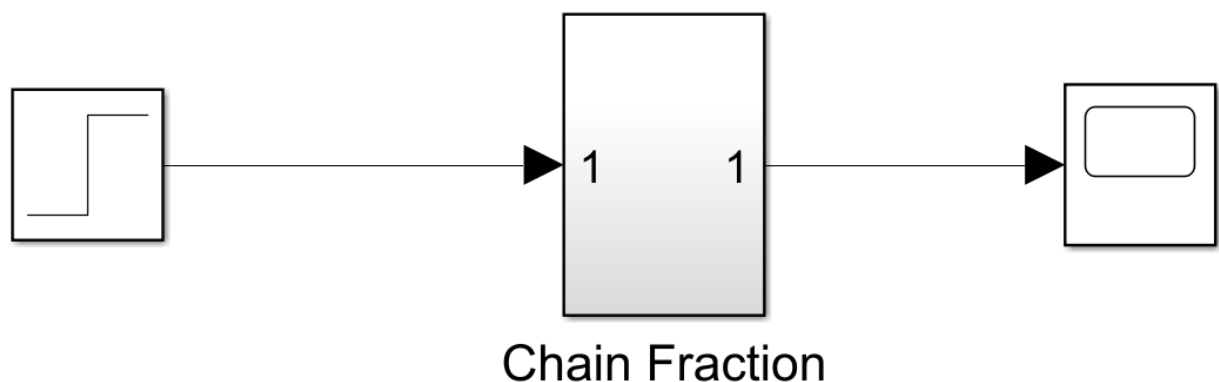


Рис. 3.2. Основна модель системи

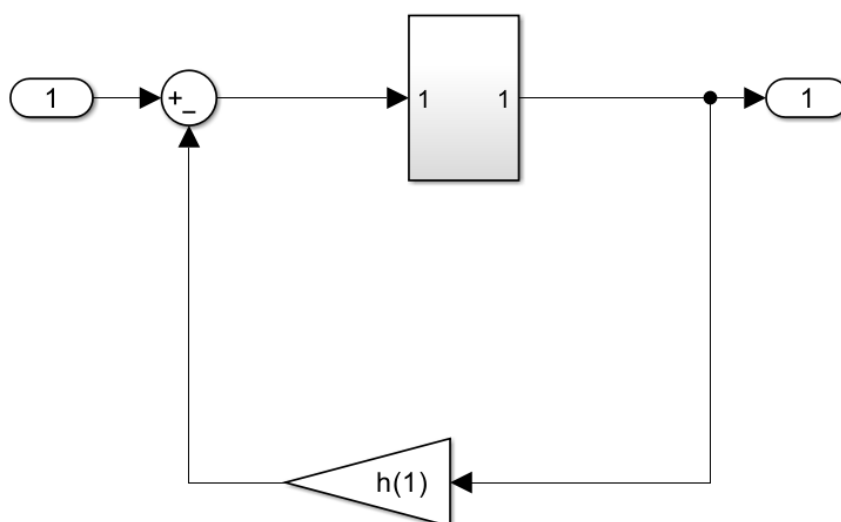


Рис. 3.3. Підсистема для обчислення елемента ланцюгового дробу

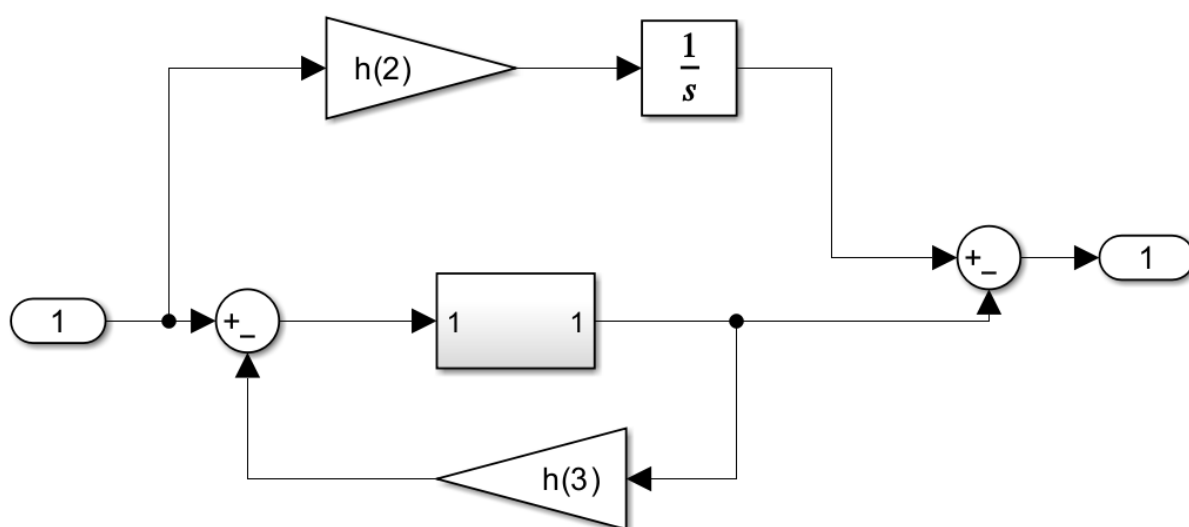


Рис. 3.4. Ієрархічна структура моделі для 2-3-го порядку апроксимації

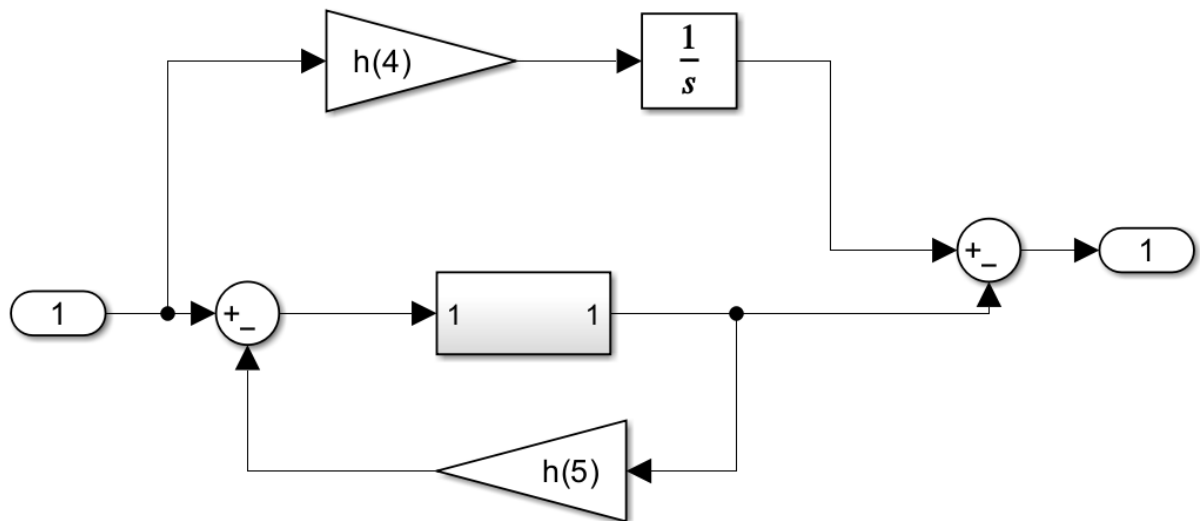


Рис. 3.5. Ієрархічна структура моделі для 4-5-го порядку апроксимації



Рис. 3.6. Ієрархічна структура моделі для 6-го порядку

Принцип масштабування цього підходу для моделей вищих порядків є інтуїтивно зрозумілим. Хоча використання великої кількості підсистем може ускладнювати загальну картину моделі, цей недолік легко компенсується підвищенням модульності та керованості.

4.5. Модель "вхід-вихід" для ланцюгового дробу

В середовищі Simulink була розроблена модель "Lanc_dr_h64", призначена для роботи з 64 коефіцієнтами ланцюгового дробу. Модель має

один вхідний параметр "h", який розглядається як внутрішня змінна. Загальна структура моделі "вхід-вихід" зображена на рис. 3.7.

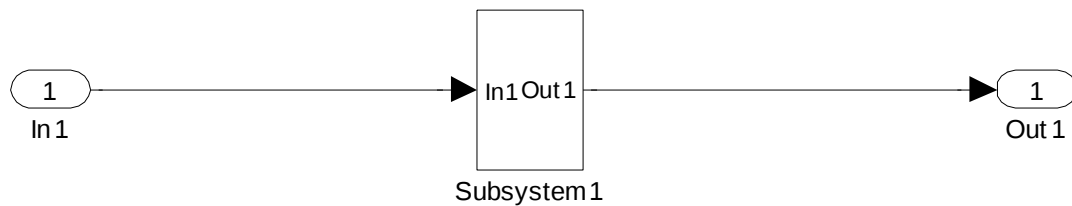


Рис. 3.7. Модель "вхід-вихід"

Для зручного використання цієї моделі в інших проектах створено блок "Model", який дозволяє передавати параметри ланцюгового дробу у вигляді вектора. Інтерфейс блоку представлений на рис. 3.8. Перед застосуванням розроблених блоків до моделей об'єктів з розподіленими параметрами необхідно провести апроксимацію цих об'єктів за допомогою запропонованих модулів, детально розглянутих у попередніх розділах.

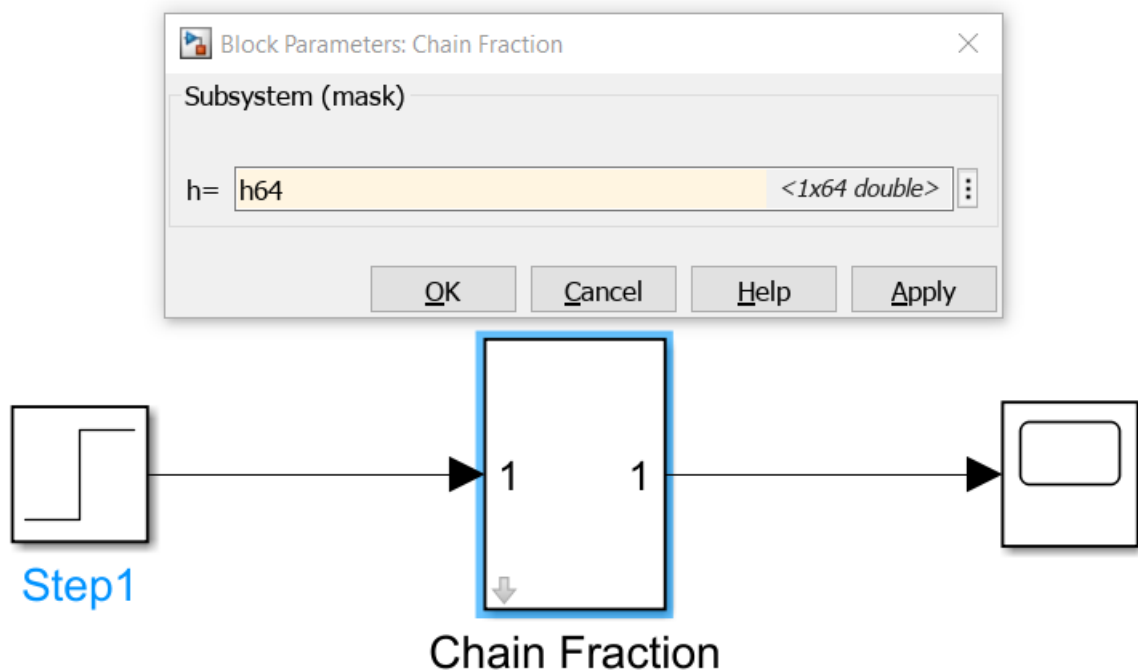


Рис. 3.8. Інтерфейс блоку Simulink-моделі

4.6. Обчислювальні експерименти

Для оцінки ефективності запропонованих методів побудови апроксимаційних моделей було проведено ряд обчислювальних експериментів.

Приклад 1.

В якості тестового прикладу була обрана система з передатною функцією

$$G_1(s) = \frac{1}{e^{as}}, \quad a = 1.$$

Аналітичний вираз для перехідної характеристики цієї системи

$$f_1(t) = \begin{cases} 0, & 0 < t < 1, \\ 1, & t > 1. \end{cases}$$

Результати моделювання та порівняння з аналітичним розв'язком представлені на рис. 3.9 та 3.10 відповідно. Рис. 3.9 демонструє отриману перехідну характеристику, а рис. 3.10 ілюструє абсолютну похибку апроксимації.

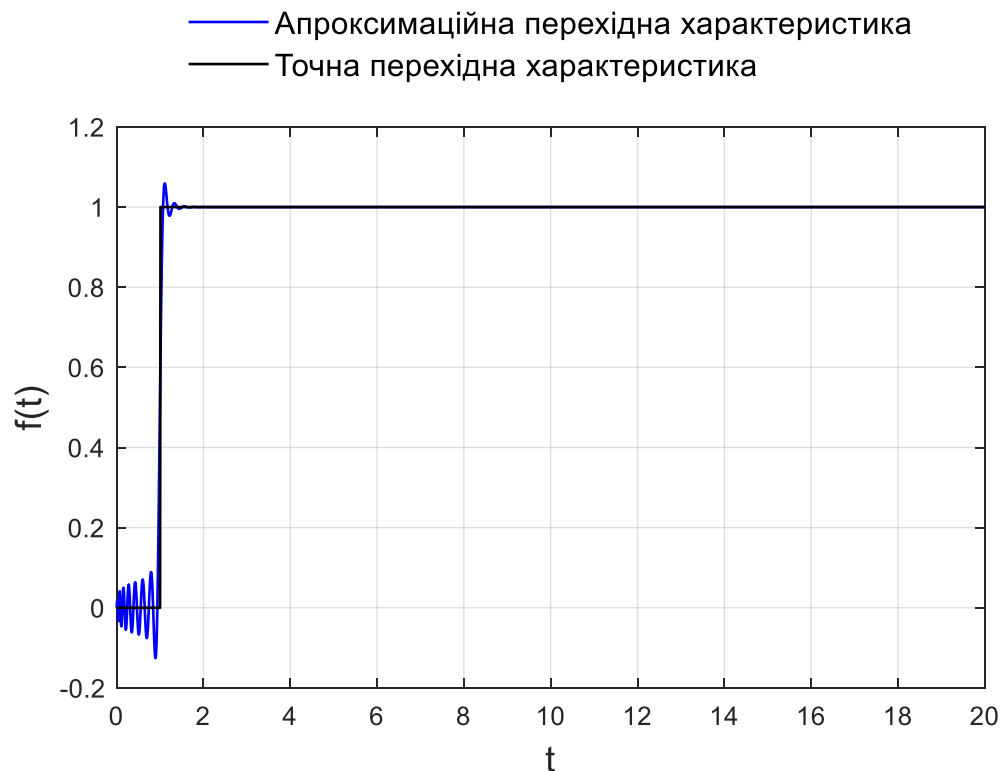


Рис. 3.9. Перехідна характеристика прикладу 1

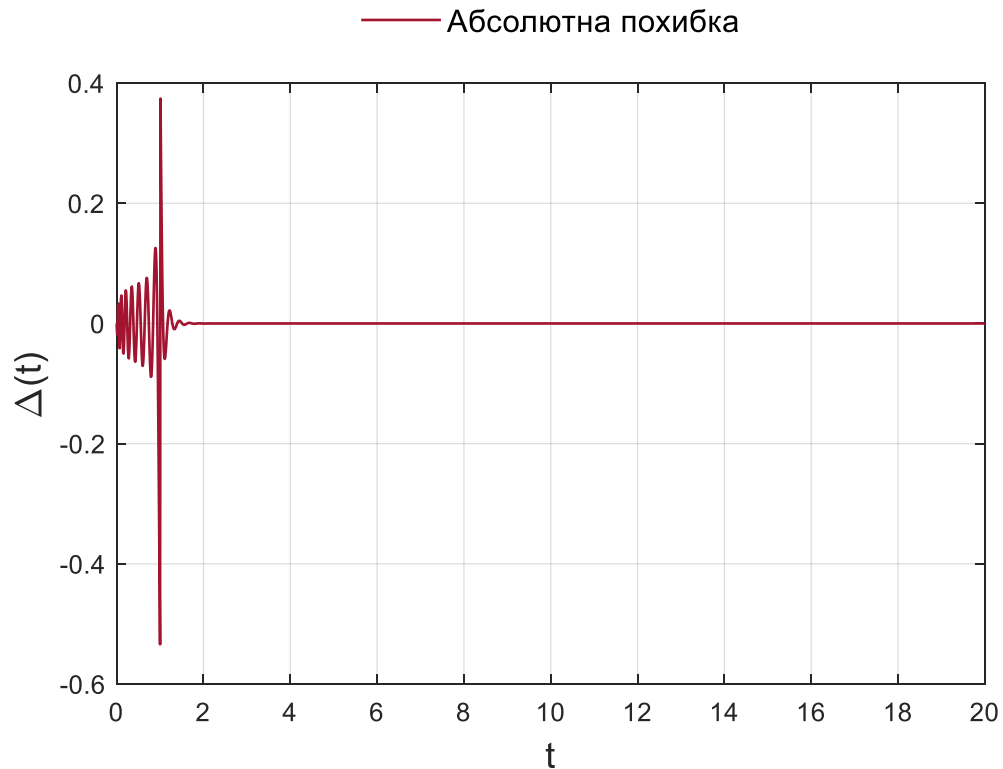


Рис. 3.10. Абсолютна похибка прикладу 1

Приклад 2.

Розглянемо інший приклад, де передатна функція має вигляд

$$G_2(s) = \frac{1}{\cosh(as)}, \quad a = 1$$

Аналітичний розв'язок для перехідної характеристики цієї системи представлений у виразі

$$f_2(t) = \begin{cases} 0, & 0.4n - 1 < \frac{t}{a} < 4n + 1 \\ 1, & 0.4n + 1 < \frac{t}{a} < 4n + 3 \end{cases}, n \in N$$

Застосувавши співвідношення

$$\cosh(as) = \frac{e^{as}(1 + e^{-2as})}{2}$$

та виконавши відповідні перетворення, отримаємо зручніший для апроксимації вираз

$$G_2(s) = \frac{2}{e^{as}(1 + e^{-2as})}.$$

Після побудови ланцюгового дробу 32-го порядку і моделювання в середовищі Simulink, було отримано результати, зображені на рис. 3.11 та 3.12. Рис. 3.11 демонструє порівняння точної та апроксимованої перехідних характеристик, а рис. 3.12 ілюструє абсолютну похибку апроксимації.

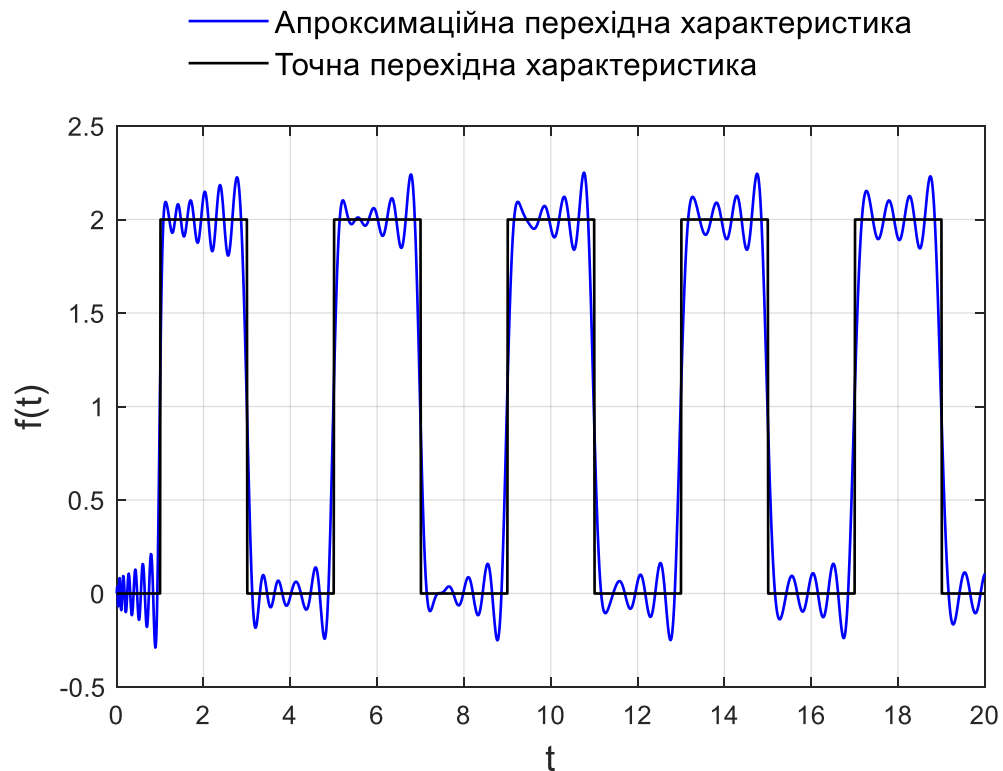


Рис. 3.11. Перехідна характеристика прикладу 2

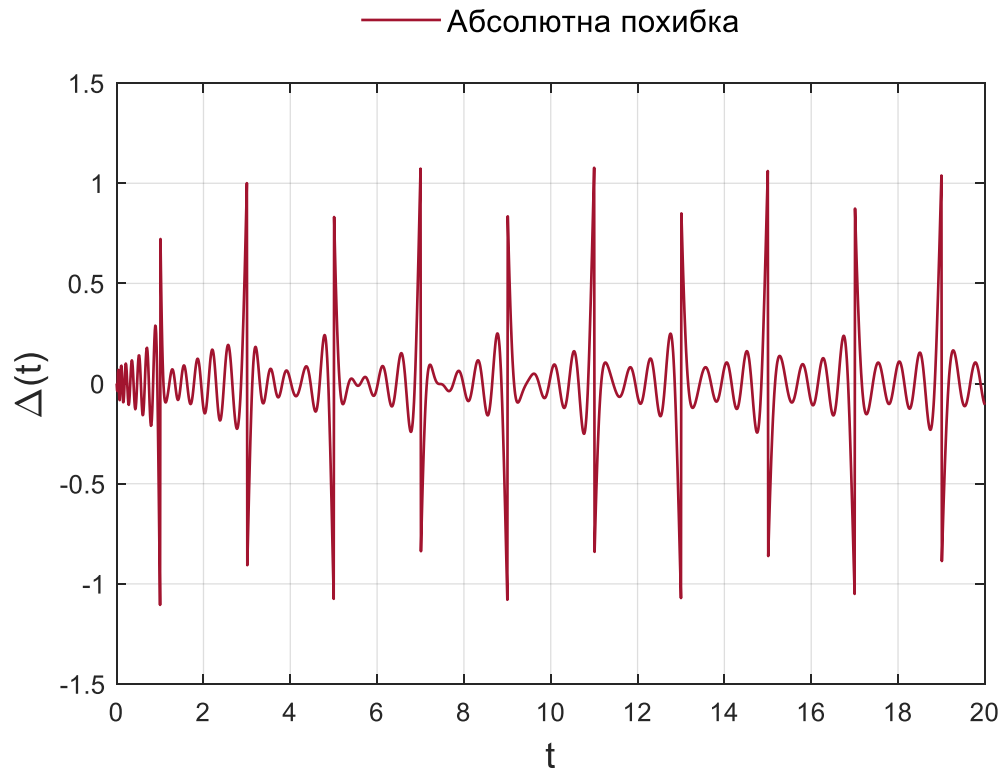


Рис. 3.12. Абсолютна похибка прикладу 2

Приклад 3.

Розглянемо ще один приклад, де передатна функція має вигляд.

$$G_3(s) = \frac{1}{\sinh(as)}, \quad a = 1$$

Аналітичний розв'язок для перехідної характеристики цієї системи представлений у виразі

$$f_3(t) = 2n, 2n - 1 < \frac{t}{a} < 2n + 1, n \in N$$

Застосувавши співвідношення

$$\sinh(as) = \frac{e^{as}(1 - e^{-2as})}{2}$$

та виконавши відповідні перетворення, отримаємо зручніший для апроксимації вираз

$$G_3(s) = \frac{2}{e^{as}(1 - e^{-2as})}.$$

Після побудови ланцюгового дробу і моделювання в середовищі Simulink, було отримано результати, зображені на рис. 3.13 та 3.14. Рис. 3.13 демонструє порівняння точної та апроксимованої перехідних характеристик, а рис. 3.14 ілюструє абсолютну похибку апроксимації.



Рис. 3.13. Перехідна характеристика прикладу 3

Приклад 4.

Розглянемо четвертий приклад, де передатна функція має вигляд

$$G_4(s) = \tanh(as), \quad a = 1$$

Аналітичний розв'язок для перехідної характеристики цієї системи представлений у виразі

$$f_4(t) = (-1)^{n-1}, n - 1 < \frac{1}{2a}t < n, n \in N.$$

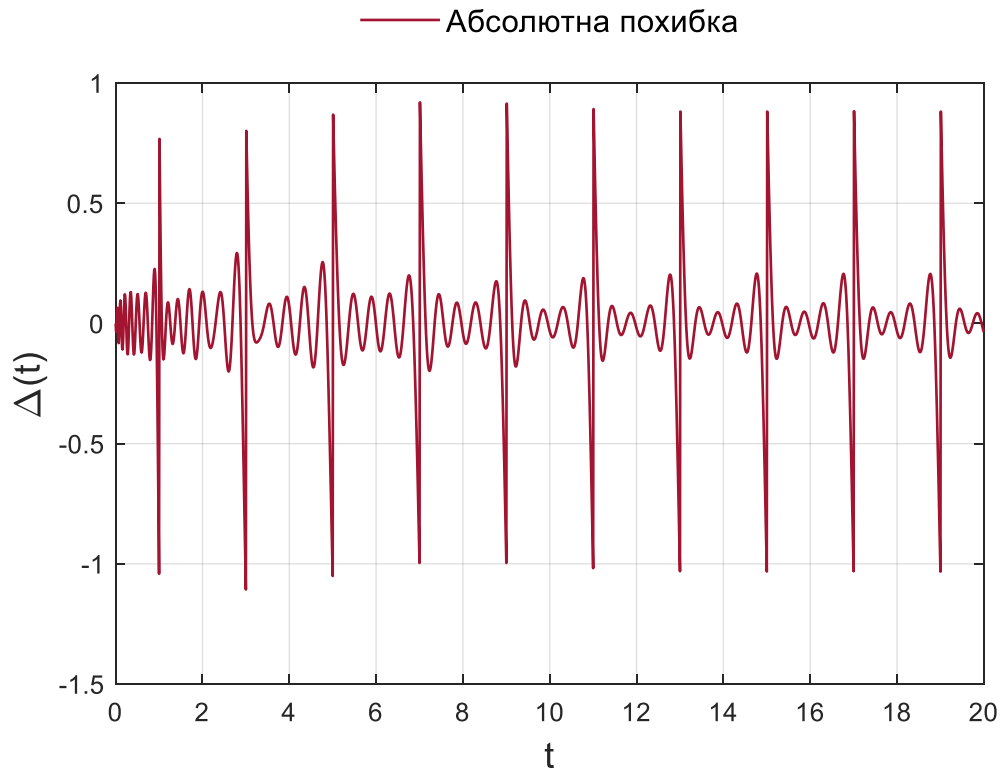


Рис. 3.14. Абсолютна похибка прикладу 3

Застосувавши відповідні перетворення отримаємо зручніший для апроксимації вираз

$$G_4(s) = 1 - 2 \frac{e^{-2as}}{1 + e^{-2as}}$$

Після побудови ланцюгового дробу 32-го порядку і моделювання в середовищі Simulink, було отримано результати, зображені на рис. 3.15 та 3.16. Рис. 3.15 демонструє порівняння точної та апроксимованої перехідних характеристик, а рис. 3.16 ілюструє абсолютну похибку апроксимації.

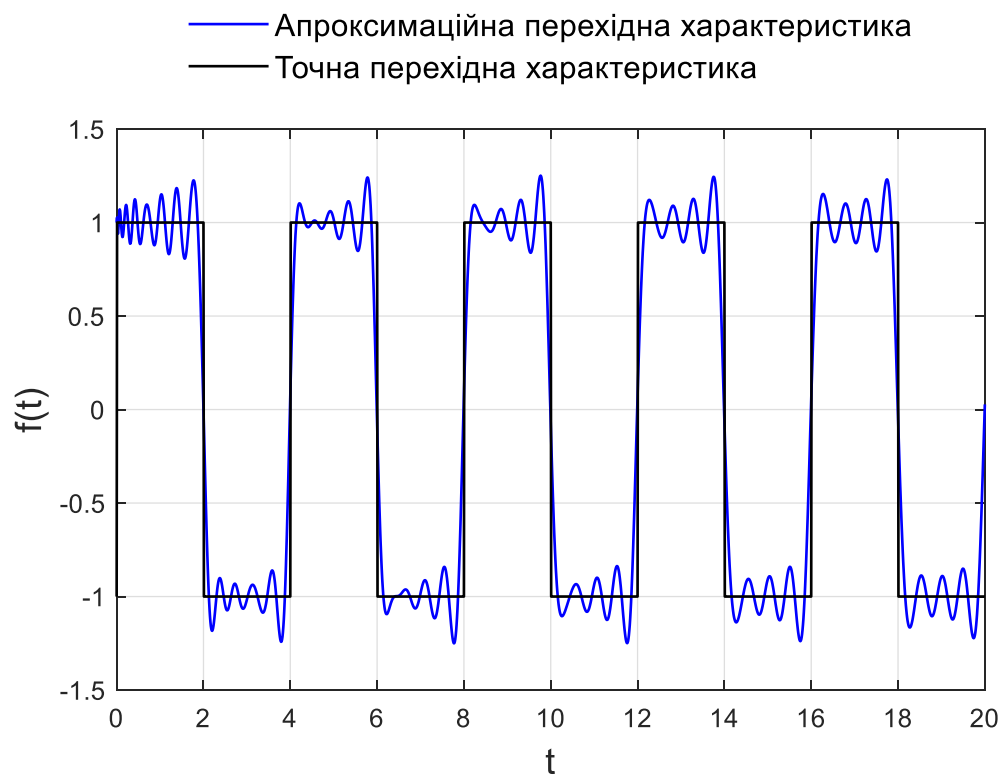


Рис. 3.15. Перехідна характеристика прикладу 4

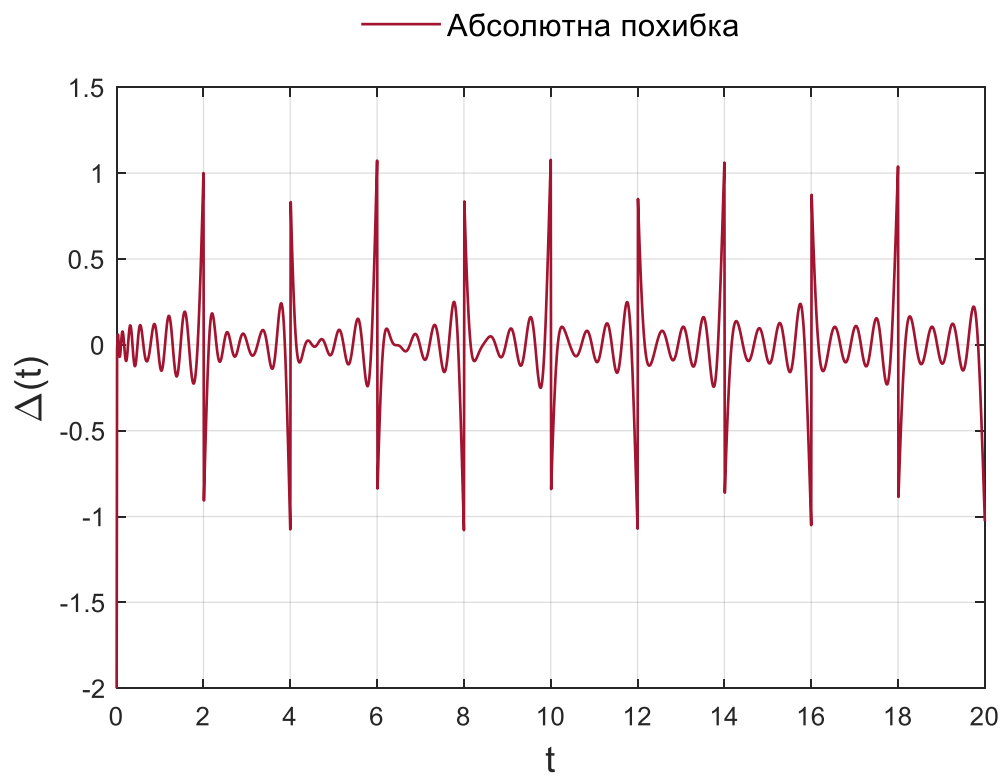


Рис. 3.16. Абсолютна похибка прикладу 4

Приклад 5.

Розглянемо п'ятий приклад, де передатна функція має вигляд

$$G_5(s) = \coth(as), \quad a = 1$$

Аналітичний розв'язок для перехідної характеристики цієї системи представлений у виразі

$$f_5(t) = 2n - 1, n - 1 < \frac{1}{2a}t < n, n \in N.$$

Застосувавши відповідні перетворення отримаємо зручніший для апроксимації вираз

$$G_4(s) = 1 + 2 \frac{e^{-2as}}{1 - e^{-2as}}$$

Після побудови ланцюгового дробу 32-го порядку і моделювання в середовищі Simulink, було отримано результати, зображені на рис. 3.17 та 3.18. Рис. 3.17 демонструє порівняння точної та апроксимованої перехідних характеристик, а рис. 3.18 ілюструє абсолютну похибку апроксимації.

Висновки до 4 розділу

У четвертому розділі роботи було детально описано розроблені програмні модулі, які дозволяють ефективно будувати імітаційні моделі систем на основі ланцюгових дробів. Проведені обчислювальні експерименти показали високу точність апроксимації динамічних об'єктів за допомогою запропонованих методів. Оптимізація моделей у середовищі Simulink дозволила значно скоротити час обчислень і підвищити стабільність роботи моделей. Розроблена модель "вхід-вихід" для ланцюгового дробу є універсальним інструментом для аналізу та синтезу різноманітних систем автоматичного керування.



Рис. 3.17. Перехідна характеристика прикладу 5

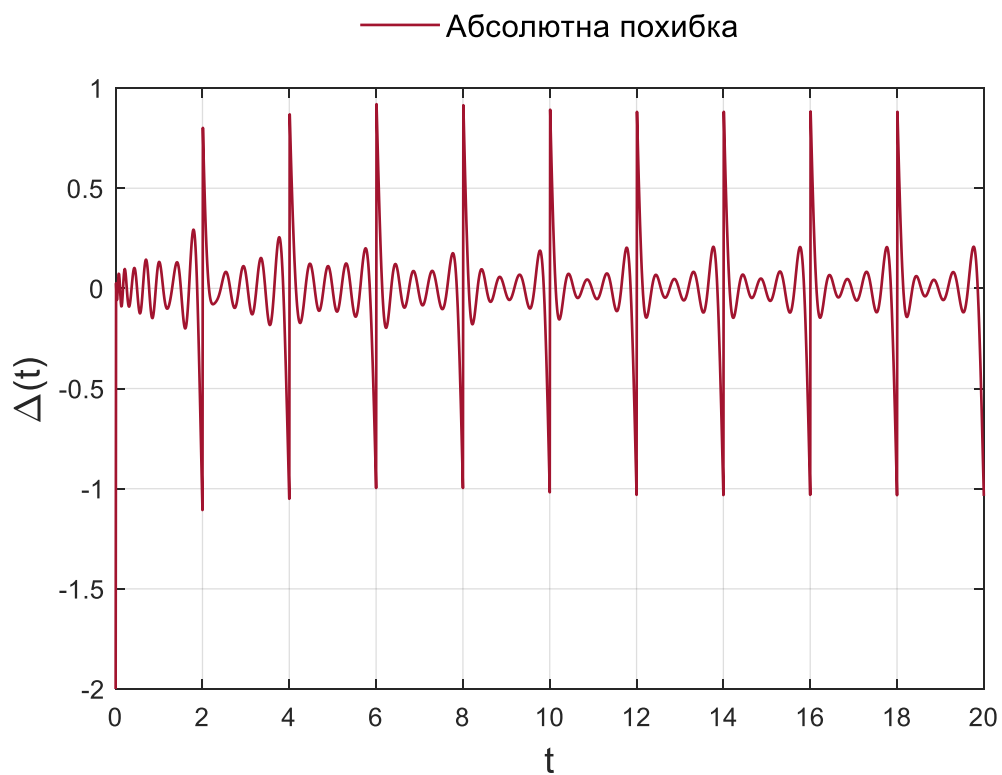


Рис. 3.18. Абсолютна похибка прикладу 5

ВИСНОВКИ

Проведені дослідження дозволили детально проаналізувати існуючі методи моделювання динамічних систем та виявити їхні обмеження. На основі цього аналізу були розроблені нові алгоритми апроксимації передатних функцій систем з розподіленими параметрами ланцюговими дробами. Ці алгоритми забезпечують більш точну та ефективну апроксимацію порівняно з існуючими методами.

Для реалізації розроблених алгоритмів було створено спеціалізоване програмне забезпечення на платформі MATLAB/Simulink. Розроблені програмні модулі дозволяють автоматизувати процес побудови імітаційних моделей, що значно спрощує процес моделювання та аналізу динамічних систем.

Проведені обчислювальні експерименти підтвердили високу точність і ефективність запропонованих методів. Отримані результати демонструють, що розроблені моделі адекватно описують динаміку реальних систем і можуть бути використані для вирішення широкого кола задач автоматичного керування.

Таким чином, у даній роботі було вирішено всі поставлені завдання. Розроблені методи та програмні засоби можуть бути успішно застосовані для моделювання та аналізу складних динамічних систем, що мають розподілені параметри.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ

1. B. Wayne Bequette. Process Control: Modeling, Design, and Simulation, 2nd edition. *Rensselaer Polytechnic Institute*. Pearson, 2023
2. Ben Adcock, Simone Brugiapaglia, Clayton G. Webster. Sparse Polynomial Approximation of High-Dimensional Functions, *Simon Fraser University, Concordia University, University of Texas*. SIAM, 2022.
3. Bingen Yang, Inna Abramova. Dynamic Systems: Modeling, Simulation, and Analysis. *University of Southern California; University of Southern California*. Cambridge University Press, 2022.
4. Bingen Yang. Stress, Strain, and Structural Dynamics: An Interactive Handbook of Formulas, Solutions, and MATLAB Toolboxes, 2nd edition. *University of Southern California*. Academic Press, 2022.
5. Clarence W. de Silva. Modeling of Dynamic Systems: with Engineering Applications, 2nd edition. *University of British Columbia*. CRC Press, Inc., 2022.
6. Dingyü Xue, Feng Pan. MATLAB and Simulink in Action: Programming, Scientific Computing and Simulation. *Northeastern University, Northeastern University*. Springer Singapore, 2024.
7. Dumitru Baleanu, Dr. Valentina Balas, Dr. Praveen Agarwal. Fractional Order Systems and Applications in Engineering. *Institute of Space Sciences, Magurele-Bucharest, "Aurel Vlaicu" University of Arad, Anand International College of Engineering*. Academic Press, 2022
8. Gang Li. Introduction to the Finite Element Method and Implementation with MATLAB. *Clemson University*. Cambridge University Press, 2020.
9. Gordon Blower. Linear Systems. Springer International Publishing, 2023.
10. Jongrae Kim. Dynamic System Modelling and Analysis with MATLAB and Python: For Control Engineers. *University of Leeds*. Wiley-IEEE Press, 2022.
11. Nader Jalili, Nicholas W. Candelino. Dynamic Systems and Control Engineering. Cambridge University Press, 2023.

12. Paul A. Gagniuc. Coding Examples from Simple to Complex: Applications in MATLAB. *National University of Science and Technology Politehnica Bucharest*. Springer International Publishing, 2024.
13. Sulaymon Eshkabilov. Beginning MATLAB and Simulink: From Beginner to Pro, 2nd edition. *North Dakota State University*. Apress, 2022.
14. Верлань А. Ф., Федорчук В. А. Моделі динаміки електромеханічних систем Київ: Наук. думка, 2013. 222 с.
15. Верлань А. А., Федорчук В. А. Моделювання комп'ютерно-інтегрованих силових енергетичних установок: монографія. Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2023. 296 с.
16. Верлань А. Ф., Федорчук В. А., Іванюк В. А. Комп'ютерне моделювання в задачах динаміки електромеханічних систем : монографія. Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова. Кам'янець-Подільський, 2010. 204 с.
17. Іванюк В. А., Тихоход В. О., Протасов С. О. Інтегральні моделі ірраціональних та трансцендентних ланок. Математичне та комп'ютерне моделювання. Серія: Технічні науки : зб. наук. праць. Кам'янець-Подільський, 2011. Вип. 5. С. 101–109.
18. Федорчук В. А., Іванюк В. А. Сучасні проблеми комп'ютерного моделювання: навчально-методичний посібник. [Електронний ресурс]. Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2023. 101 с.
19. Павлюк Б., Іванюк В. Застосування ланцюгових дробів для моделювання систем з розподіленими параметрами. Вісник Кам'янець-Подільського національного університету імені Івана Огієнка. Фізико-математичні науки. Випуск 17. Кам'янець-Подільський : Кам'янець-Подільський національний університет імені Івана Огієнка, 2024.