

Міністерство освіти України
Львівський національний університет імені Івана Франка

Я. М. Матвійчук

**МАТЕМАТИЧНЕ МАКРОМОДЕЛЮВАННЯ
ДИНАМІЧНИХ СИСТЕМ:
ТЕОРІЯ ТА ПРАКТИКА**

Львівський національний університет імені Івана Франка

Львів 2000

УДК 621.3.

МАТВІЙЧУК Я.М.

Математичне макромодельовання динамічних систем: теорія та практика.

Наукове видання. -Львів: ЛНУ ім.Івана Франка, 2000. -236с.

У монографії розглянуто проблеми макромодельовання неперервних лінійних та нелінійних динамічних систем із зосередженими сталими параметрами. Обґрунтовано загальні моделюючі структури у вигляді систем звичайних диференціальних рівнянь та методи їхньої ідентифікації. Досліджено методи регуляризації ідентифікаційних процедур. Усі методи проілюстровано численними прикладами оригінальних макромоделей конкретних систем.

Для спеціалістів з математичного моделювання і теорії систем, аспірантів і студентів.

Рецензенти: д-р фіз.-мат. наук, проф. *Б.О.Попов* (Фіз.-мех. ін-т НАНУ);

д-р техн. наук, проф. *Л.А.Синицький* (ЛНУ ім.І.Франка);

д-р техн. наук, проф. *П.Г.Стахів* (ДУ “Львівська політехніка”).

Мовний редактор М.Мартиняк.

ВСТУП

Науково-технічний прогрес спричинює створення складних систем, особливо в галузі мікроелектроніки, які важко аналізувати в цілому на ЕОМ, однак які потрібно розглядати як цілісні функціональні блоки. Це зумовило створення математичних моделей високого рівня агрегування – макромоделей, які активно розробляються протягом останніх 25 років.

Переважає більшість дослідників називає макромоделями математичні моделі пристроїв та систем, які, з одного боку, помітно простіші від оригіналу, а з іншого, – достатньо точно відтворюють суттєві риси поведінки модельованих об'єктів. Менш відомими задачами, що посилили інтерес до макромоделей, є так звані обернені задачі нелінійної динаміки, що полягають у відтворенні математичного оператора об'єкта за заданими вхідними та вихідними сигналами. Спрощення макромоделі порівняно з оригіналом тут є несуттєвим. Однак формулювання задач та арсенал засобів їх розв'язування збігаються з традиційними задачами макромоделювання. До обернених задач нелінійної динаміки за формулюванням та методами розв'язування дуже близькі задачі математичного прогнозування.

Вивчення власне задач макромоделювання розпочав Грінбаум [1], що запропонував метод структурно-функціонального синтезу макромоделей. Важливі результати у математичному макромоделюванні одержали С.Директор, Р.Рорер [2], Л.Заде, Ч.Дезоер [3], Н.Вінер [4], Р.Калман [5], Л.О.Чуа [6], Ф.Такенс [7], а також С.А.Букашкін [8], А.Ф.Верлань [9], Л.В.Данілов [10], С.В.Денбновецький [11], А.Г.Івахненко, В.С.Степашко [12], В.Н.Ільїн [13], А.А.Ланне [14], В.Камінскас [15], В.І.Капалін [16], А.І.Петренко [17], Г.Є.Пухов [18], В.П.Сігорський [19], Л.А.Синицький [20], П.Г.Стахів [21, 22].

У цілому задача побудови макромоделі складається з трьох підзадач: вибір структури макромоделі; ідентифікація параметрів макромоделі за експериментальними даними; адекватний опис макромоделі для успішного функціонування у зовнішньому оточенні.

У монографії розглянуто, зокрема, такі проблеми: створення обґрунтованих структур макромоделей у вигляді систем диференціальних рівнянь визначеного класу; розробка стійких методів ідентифікації макромоделей визначеного класу; синтез макромоделей різних систем у формі, придатній для їх використання в системах автоматизованого проектування.

Усі методи ідентифікації, розглянуті в монографії, проілюстровано прикладами макромоделей конкретних систем. Оскільки макромоделі ідентифіковано за допомогою системи аналізу електронних схем та систем САНОС-ПК, ми коротко описали вхідну мову та структуру цієї системи. Короткий виклад мови системи САНОС-ПК потрібний ще й тому, що майже всі приклади в монографії наведені як завдання мовою САНОС-ПК. Це дозволило компактно та вичерпно описати численні ілюстрації методів ідентифікації.

Систему САНОС-ПК разом з усіма прикладами та детальними інструкціями можна переписати з INTERNET-адреси WWW.WUPS.LVIV.UA/ За цією ж адресою доступні всі програми апроксимації, на які є посилання в монографії як на додатки.

Попередньо читати монографію можна без детального розгляду ілюстрацій у вигляді завдань на мові САНОС-ПК. Однак короткого опису мови у п'ятому розділі достатньо для повного розуміння всіх наведених у монографії прикладів.

Автор висловлює щире подяку професору Л.А.Синицькому, численні бесіди з яким суттєво сприяли поліпшенню монографії, а також відзначає внесок Р.Гасько, М.Густі та А.Курганевича, яким належать макромоделі довгої лінії та частотного детектора, синтезовані за середньо-квадратичним та за мінімаксним критеріями.

Автор наперед вдячний усім читачам, що захочуть повідомити свої зауваження та враження на адресу видавництва, а також на адресу matv@ua.fm

РОЗДІЛ 1

ЗАГАЛЬНИЙ ОГЛЯД ПРОБЛЕМИ МАКРОМОДЕЛЮВАННЯ

1.1. ВИЗНАЧЕННЯ МАКРОМОДЕЛІ

У праці В.Ільїна [13] дано визначення макромоделі пристрою, як його математичної моделі, простішої за обсягом обчислень та затратами машинної пам'яті, ніж модель, що отримується безпосереднім об'єднанням математичних моделей компонент цього пристрою.

Звернімо увагу на два важливі моменти цього визначення.

По-перше, наголошено, що макромодель – це модель математична. Деякі автори пов'язують поняття макромоделі зі спрощеною схемою заміщення модельованого пристрою. Такий підхід значно звужує можливості макромодельовання та не дає нічого нового, бо в результаті машинного опрацювання схеми заміщення формується все та ж математична модель. Зручність макромоделей у вигляді схем заміщення полягає лише у простоті опису вхідними мовами систем автоматизованого проектування (САПР) та у легшому сприйнятті їх інженерами-проектувальниками. Інше трактування зводить поняття макромоделі до опису внутрішньою мовою САПР, що не можна вважати конструктивним.

По-друге, чітко визначено ієрархічний рівень макромоделі.

На рис.1.1 показані рівні макромодельовання для систем електроніки.

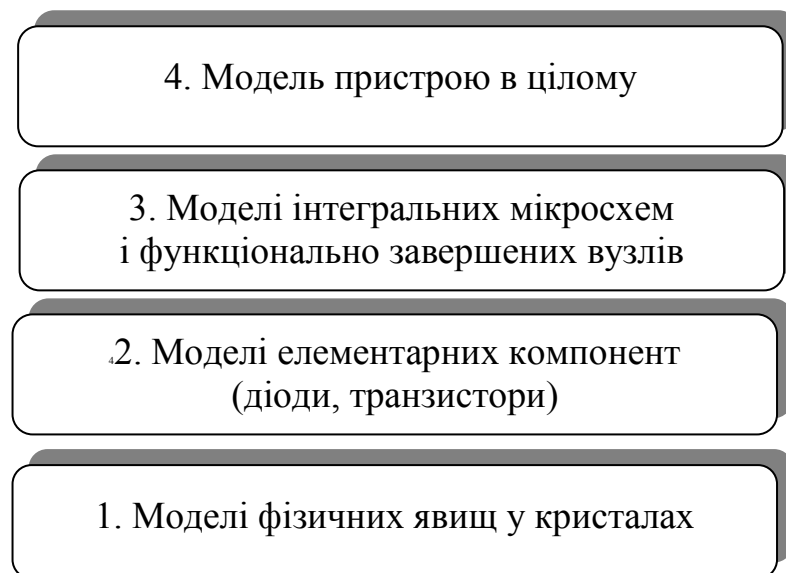


Рис.1.1. Ієрархія рівнів моделювання в електроніці

Згідно зі сформульованим вище визначенням, макромодель перебуває на рівнях 3 та 4. Саме таке визначення макромоделі прийнято в нашій монографії.

У праці [13] наведено перелік основних проблем макромоделювання, що є загальними і досі актуальними:

- 1) класифікація макромоделей з метою впорядкування основних ідей макромоделювання;
- 2) побудова ієрархічного ряду макромоделей;
- 3) побудова динамічних нелінійних макромоделей;
- 4) забезпечення алгоритмічної надійності макромоделей.

Ми досліджували третю й четверту проблеми як такі, що є ключовими та найважливішими у практиці.

Надалі під макромоделлю розумітимемо деяку математичну модель, що відтворює із заданою точністю співвідношення між заданими множинами вхідних та вихідних сигналів модельованого об'єкта. Завдання макромоделювання полягає у визначенні структури макромоделі (структурна ідентифікація), її параметрів (параметрична ідентифікація) та способів введення в математичну модель пристрою в цілому (стикування із середовищем).

Макромоделі можна класифікувати за такими ознаками: форма опису або структура макромоделі (математичні рівняння того чи іншого вигляду, еквівалентні схеми заміщення, підпрограми); спосіб отримання даних для ідентифікації макромоделі (за повною схемою, за зовнішніми характеристиками, комбінований); природа модельованих об'єктів; складність макромоделі (ієрархічна класифікація).

Що стосується першої ознаки, то оскільки макромоделі у прийнятому визначенні є моделлю математичною, то ми далі розглянемо лише структури макромоделей залежно від їхнього математичного опису.

За другою ознакою можна виділити два великі класи макромоделей: макромоделі, отримані на основі співвідношень вхідних та вихідних сигналів модельованого об'єкта; макромоделі, отримані спрощенням повної схеми заміщення. Класифікація за другою ознакою визначає методи ідентифікації макромоделей.

Третя ознака стосується фізичної природи об'єктів моделювання. Переважна більшість наших прикладів побудована на радіоелектронних колах та системах. Однак універсальність розроблених методів макромоделювання робить їх придатними для об'єктів будь-якої фізичної природи.

Четверта ознака співпадає з ієрархією макромоделей за рис.1.1.

1.2. ФОРМУЛЮВАННЯ ЗАДАЧІ МАКРОМОДЕЛЮВАННЯ ТА СИСТЕМНИЙ ОПИС ОБ'ЄКТІВ

Системою називають сукупність взаємодіючих об'єктів, поведінка яких підлягає математичному опису [24].

Величини або сигнали, що їх використовують для опису системи, прийнято поділяти на зовнішні, які можна задати чи виміряти, та внутрішні,

що не підлягають безпосередньому вимірюванню. Зовнішні величини, які можна задати, називають вхідними, або збуреннями, а ті, які лише вимірюють, називають вихідними, або реакціями.

У загальному випадку математичний опис системи – це оператор (функціонал), що пов’язує вхідні, вихідні та внутрішні величини згідно з рівнянням

$$\Phi(\bar{u}, \bar{y}, \bar{x}; \bar{p}) = 0, \quad (1.1)$$

де $\bar{u} = (u_1, \dots, u_r)$, $\bar{y} = (y_1, \dots, y_s)$, $\bar{x} = (x_1, \dots, x_n)$ – вектори відповідно вхідних, вихідних та внутрішніх величин, $\bar{p} = (p_1, \dots, p_N)$ – вектор параметрів оператора.

Рівняння (1.1) повинно задовольняти умови існування та єдиності розв’язку $\tilde{\bar{y}}$ при \bar{u} , \bar{x} та \bar{p} , що є в заданих областях. Крім того, повинна виконуватись умова точності моделі

$$\|\bar{y} - \tilde{\bar{y}}\| < \varepsilon, \quad (1.2)$$

де параметр $\varepsilon > 0$ задає точність відтворення вихідного вектора.

Вектори зовнішніх та внутрішніх величин залежать від деякого параметра t , яким найчастіше є час.

Ідентифікувати математичну модель системи означає за заданою структурою оператора в (1.1) та відомими векторами $\bar{u}(t \in [t_0, t_1])$ й $\bar{y}(t \in [t_0, t_1])$ знайти вектори $\bar{x}(t \in [t_0, t_1])$ та \bar{p} оператора моделі $\Phi(\bar{u}, \bar{y}, \bar{x}; \bar{p})$ такі, щоб виконувалась умова

$$\min_{\bar{p}} \|\bar{y}(t) - \tilde{\bar{y}}(t; \bar{p})\| \quad \text{для всіх } t \in [t_0, t_1], \quad (1.3)$$

де $\tilde{\bar{y}}(t; \bar{p})$ – розв’язок рівняння (1.1) для заданого \bar{p} .

На практиці вектори $\bar{u}(t)$ та $\bar{y}(t)$ задають на дискретній множині значень параметра $\{t_k\}$, $k=1, \dots, M$. У цьому випадку критерій оптимальності ідентифікації (1.3) набуває вигляду

$$\min_{\bar{p}} \|\bar{y}(t_k) - \tilde{\bar{y}}(t_k; \bar{p})\|, \quad k=1, \dots, M. \quad (1.4)$$

Вибір певної норми зумовлює метод розв’язування оптимізаційної задачі (1.4). Найпоширенішими [25] є метод найменших квадратів

$$\min_{\bar{p}} \sum_{k=1}^M (\bar{y}(t_k) - \tilde{\bar{y}}(t_k; \bar{p}))^2 \quad (1.5)$$

та метод мінімаксу

$$\min_{\bar{p}} \max_{k=1, M} |\bar{y}(t_k) - \tilde{\bar{y}}(t_k; \bar{p})| . \quad (1.6)$$

Якщо $\tilde{\bar{y}}(t_k; \bar{p})$ є лінійною функцією компонент вектора параметрів \bar{p} , то (1.5) зводиться до розв'язування s систем лінійних алгебричних рівнянь порядку N , а (1.6) можна розв'язати методами лінійного програмування.

Макромодельовання полягає у виборі структури оператора в (1.1) та ідентифікації параметрів обраної структури за заданими множинами значень $\{u(t_k)\}$, $\{y(t_k)\}$, $k=1, \dots, M$. Разом з тим оператор в (1.1) повинен бути якомога простішим.

Зручним є системний опис об'єктів макромодельовання. Розглянемо перехід до системного опису на прикладі електричних систем. Системний опис у теорії електричних кіл поширився завдяки розвитку теорії багатополосників. Електричну систему щодо зовнішнього середовища завжди можна зобразити у вигляді багатополосника із зовнішніми затискачами (полосами) та системою струмів і напруг щодо цих полюсів. Використовують рівноцінні системи: напруг сторін і струмів сторін; полюсних напруг і полюсних струмів (рис.1.2).

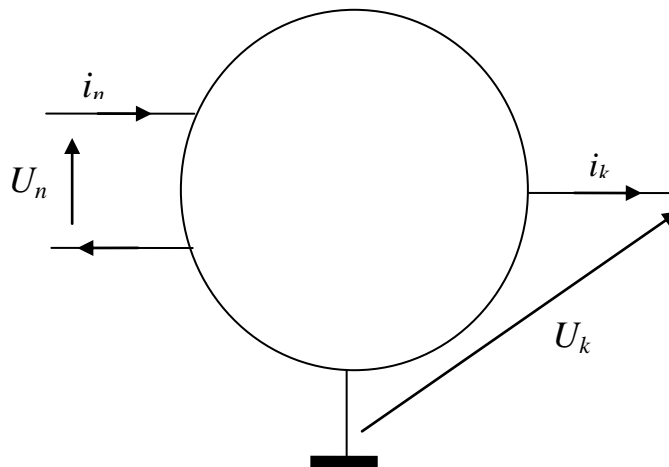


Рис.1.2. Напруга та струм n -ї сторони та k -го полюса багатополосника

До системного опису багатополосника завжди можна перейти, якщо розділити струми й напруги на вхідні та вихідні і сформулювати таким чином вектор вхідних сигналів \underline{u} та вектор вихідних сигналів \underline{y} . Потрібно лише стежити, щоб у вхідний або вихідний вектор не потрапили струм і напруга, що належать до однакової сторони або полюса. Такі величини пов'язані між собою внутрішніми співвідношеннями багатополосника і не можуть бути взаємно незалежними.

Системний підхід зручний для подальшого викладу і дає змогу рівноправно вводити у вектори сигналів змінні як електричної, так і неелектричної природи (температура, механічна сила, радіація). Крім того, системний опис надає отриманим результатам універсальності, загальності,

поширює їх на об'єкти довільної фізичної природи, що допускають такий опис. З огляду на викладене вище надалі послідовно проводиться системний підхід на всіх рівнях викладу.

1.3. СТРУКТУРИ МАКРОМОДЕЛЕЙ ТА МЕТОДИ ІДЕНТИФІКАЦІЇ

1.3.1. Методи спрощення та агрегування

Повну компонентну модель схеми спрощують, замінюючи групу елементів одним або кількома керованими елементами, у яких є елементи пасивні (керовані резистори або ємності) та активні (керовані джерела напруги або струму) із складними описами або навіть алгоритмами. В праці [18] розглянуто три підходи до такого спрощення.

Перший полягає в тому, що замість деякого каскаду схеми вмикають нелінійний опір, характеристика якого визначена процедурою оптимізації.

Другий підхід передбачає визначення (за допомогою методів аналізу чутливості) впливу окремих елементів на вихідні характеристики схеми. Якщо вплив незначний, то відповідний елемент або відкидають, або приєднують до інших шляхом еквівалентних перетворень.

Ці підходи для макромодельовання використовують порівняно мало. Частіше застосовують третій підхід, що ґрунтується на побудові еквівалентних схем окремих типових структур. У праці [26] наведено словник типових структур інтегральних мікросхем – найчастіше використовуваних схемотехнічних деталей мікросхем (диференціальний каскад, відбивач струму, підсилювальний каскад, вихідний каскад, тощо). Для кожного типового елемента будують макромодель, що містить, як звичайно, керовані джерела струму та напруги. В аналітичні залежності, які описують властивості модельованих каскадів, вводяться параметри, що дають змогу коректувати властивості елементів. На базі макромоделей елементів будують макромодель цілої інтегральної мікросхеми.

Очевидно, подібне спрощення розгорнутої компонентної моделі є досить складною задачею, яку може розв'язати лише досвідчений розробник, що добре розуміє функціонування об'єкта та може врахувати властивості, потрібні в кожному конкретному випадку.

Більш формалізованими є методи агрегування, що полягають у згортанні (редукції) рівнянь повної компонентної моделі шляхом аналітичних перетворень. Вони спрямовані на те, щоб усі напруги та струми, а також усі аргументи нелінійних залежностей, які входять у повний опис схеми, були виражені через зовнішні вхідні та вихідні змінні. Кількість рівнянь макромоделі в цьому випадку дорівнює кількості зовнішніх змінних.

На жаль, послідовне проведення таких перетворень до кінця можливе лише у виняткових випадках. Є відомості, що кількість диференціальних рівнянь у макромоделі можна зменшити у три-чотири рази, зберігаючи точність повної компонентної моделі [27]. Поширення такий підхід не набув.

1.3.2. Методи функціональної подібності

Перші макромоделі інтегральних мікросхем створені на початку 70-х років у США [1]. Формування таких макромоделей зводиться до виділення та заміни спрощеною еквівалентною схемою групи елементів, об'єднаних спільною функцією. Цей підхід набув загального визнання та поширення й отримав назву структурно-функціонального синтезу, а макромоделі, створені за його допомогою – функціональними. Методами структурно-функціонального синтезу створені широкі бібліотеки макромоделей мікросхем.

У випадку структурно-функціонального синтезу приймають, що функціональне призначення модельованого об'єкта відоме, а зовнішні характеристики та параметри використовують лише для оцінки недосконалості макромоделі під час виконання заданих функцій.

Узагальнена структура такої макромоделі показана на рис.1.3. Вона складається з трьох блоків послідовного вкладення: блоку зовнішніх характеристик; власне блоку функціонування; блоку зв'язків між ними.

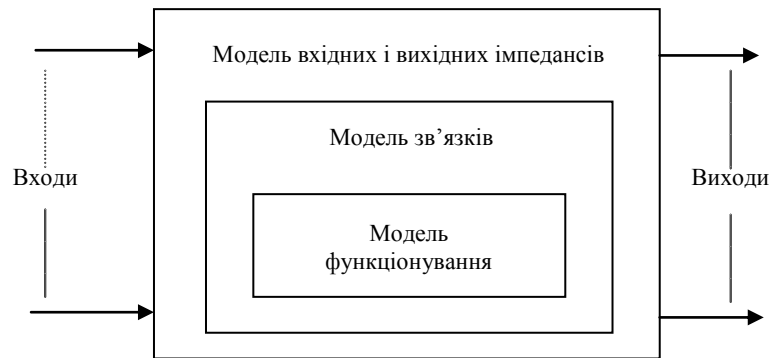


Рис.1.3. Узагальнена структура макромоделі у випадку структурно-функціонального синтезу

Частіше використовують окремий випадок структури у вигляді, показаному на рис.1.4.

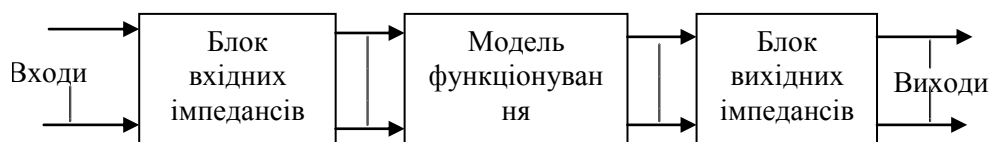


Рис.1.4. Структура функціональної макромоделі з послідовним розміщенням блоків

Внутрішній блок макромоделі відображає її головні функціональні властивості, його найчастіше зображають у такому вигляді: аналітичних залежностей; логічних рівнянь та таблиць істинності; часових діаграм; ідеалізованих еквівалентних схем.

Блоки вхідних та вихідних імпедансів слугують для зв'язку внутрішнього блоку з оточенням макромоделі.

Зазначимо, що розподіл макромоделі на блоки та належність того чи іншого елемента до певного блоку є умовними і в кожному конкретному випадку це можна виконувати по-різному.

Розглянемо макромодель операційного підсилювача (рис.1.6), побудовану за методом структурно-функціонального синтезу.

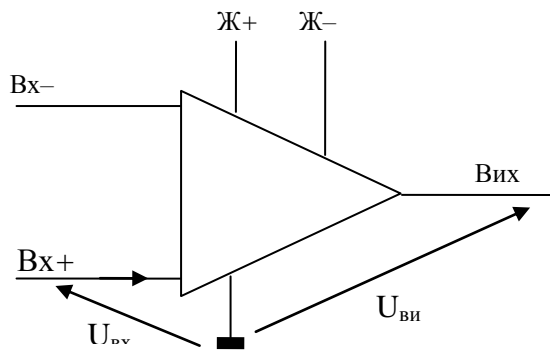
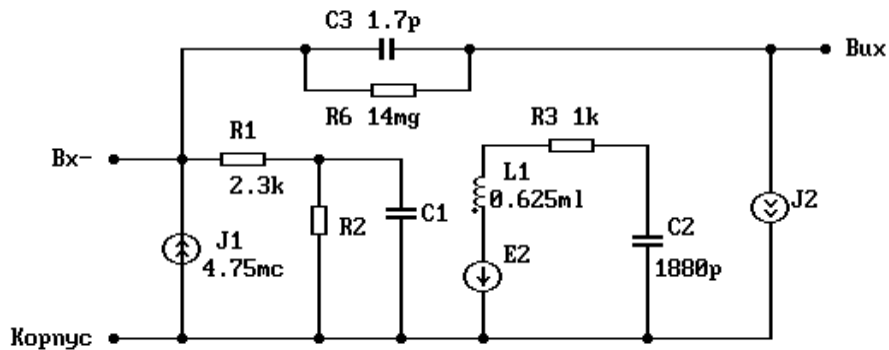


Рис.1.6. Зовнішні змінні операційного підсилювача

Схема простої нелінійної макромоделі операційного підсилювача К140УД1 [26] з потрібними додатковими співвідношеннями зображена на рис.1.7. Це набір каскадів, сукупність яких є простою схемою заміщення інтегральної мікросхеми.



$$E2=U_{Bx-}; \quad C1=35 \cdot (1+1.285/\sqrt{\text{ch}(29.16 \cdot (U_{Bx-}+8.5\text{ml})))}) \text{ [pF]}; \quad R2=(U_{Bx-}+8.5\text{ml})/ \\ (4.75\text{mc} \cdot \text{th}(29.16 \cdot (U_{Bx-}+8.5\text{ml}))) - 2.3\text{k}[\Omega]; \quad J2=\text{if}(U_{C2} \geq -5.4\text{ml}) \text{ then } (S \cdot (U_P - U_{C2})) \text{ else } \\ (S \cdot (U_P + 5.4\text{ml}))[\text{A}]; \quad S=\text{if}(U_{J2} \leq 5\text{ml}) \text{ then } (3) \text{ else } (3 - 0.0625 \cdot (U_{J2} - 5))[\text{A/V}]; \quad U_P=\text{if}(U_{J2} \leq 5\text{ml}) \text{ then } (- \\ 1.75\text{ml} + 0.415\text{ml} \cdot U_{J2}) \text{ else } (-1.75\text{ml} + 0.415\text{ml} \cdot U_{J2} + 0.05\text{ml} \cdot (U_{J2} - 5)^{2.5})[\text{V}].$$

Рис.1.7. Макромодель операційного підсилювача К140УД1, синтезована за структурно-функціональним методом

Макромодель складається з таких каскадів: нелінійна модель вхідного каскаду (J1,R1,R2,C1); лінійна інерційна модель передатної характеристики (E2,L1,R3,C2); нелінійна модель кінцевого каскаду (J2); лінійна модель зворотнього зв'язку (R6,C3).

Нелінійні залежності, які відображають процеси в операційному підсилювачі, є апроксимаціями експериментальних залежностей, отриманих для інтегральної схеми. Параметри цих апроксимацій визначають, вимірюючи вольт-амперні характеристики, вхідні та вихідні опори та ємності. Структура моделі інерційного лінійного кола визначена відгуком схеми на імпульс малої амплітуди. Точність макромоделі оцінюють, порівнюючи результати макромодельовання з експериментальними результатами.

Порівняння цієї макромоделі із математичною макромоделлю, яку ми розробили, описане в 4.2.2.

Аналогічну блочну структуру застосовано в усіх працях з макромодельовання операційних підсилювачів (ОП).

Вхідний блок залежно від потрібної точності змінюється від простої RC-ланки до компонентної схеми заміщення вхідного каскаду модельованого ОП. Проміжні блоки в усіх макромоделях однотипні, складаються з керованих джерел, RC- або RLC-ланок та нелінійних опорів (діодів). Вони моделюють лінійні властивості ОП, обмеження амплітуди та обмеження швидкості зміни вихідного сигналу. Вихідні блоки подібні до вхідних.

Очевидно, створення макромоделі за методом структурно-функціонального синтезу досить складне, потребує досконалого розуміння процесів у модельованому об'єкті та доступне досвідченим спеціалістам-схемотехнікам. Тому структурно-функціональний синтез не підлягає автоматизації, що можна оцінювати як його принциповий недолік.

1.3.3. Структури математичних макромоделей

Останніми роками активно розвиваються методи, що ґрунтуються на описі співвідношень між множинами вхідних та вихідних сигналів. У літературі їх називають макромоделями “чорної скриньки”, інформаційними макромоделями, кібернетичними або математичними макромоделями. Завдяки формалізації ідентифікаційних процедур ці методи містять передумови для автоматизації макромодельовання. Друга перевага полягає в тому, що один раз розроблену методику (алгоритми, програми) можна з успіхом застосовуватись для цілого класу об'єктів довільної природи.

Математичні оператори, що реалізують потрібні співвідношення між вхідними та вихідними сигналами, можна будувати на засадах різних міркувань. Можна виділити три загальні підходи.

Перший підхід запозичено з теорії синтезу нелінійних перетворювачів, де задачу формулюють так: збудувати фізично реалізовний оператор F такий, що відображає багатопараметричну множину дійсних сигналів $\{U\}$ в множину дійсних сигналів $\{Y\}$. У макромодельованні задача дещо спрощена: знімають умову фізичної реалізованості, а множини $\{U\}$ та $\{Y\}$ є відповідно множинами вхідних та вихідних сигналів об'єкта.

Другий підхід полягає в апріорному заданні структури оператора F . Це може бути інтегральне рівняння, диференціальне рівняння, функціональний ряд тощо. Далі виконують параметричну ідентифікацію,

тобто обчислюють коефіцієнти оператора. Вибір множин сигналів $\{U\}$ та $\{Y\}$ є підпорядкованим ідентифікації.

Легко помітити недолік першого підходу. Він полягає у тому, що складність макромоделі повністю залежить від зовнішніх сигналів. Це може бути причиною надлишкової складності та, як наслідок, некоректності макромоделі. Недолік другого підходу – в апіорності вибору структури. Якщо обрана структура макромоделі не відповідає задачі, то природним проявом такої ситуації буде некоректність макромоделі при зовнішній коректності процедури ідентифікації.

Кожний із цих підходів у “чистому вигляді” не реалізують, крім деяких спеціальних випадків (наприклад, задачі прогнозування процесів). Природно виникає третій підхід, комбінований, де структуру обирають на підставі попереднього вивчення задачі, а сигнали – з відповідних множин, визначених умовами задачі.

Перший підхід представлений методом, що ґрунтується на теорії розщеплення [14]. Коротко опишемо основну ідею цього методу для скалярного випадку. Нехай $u(t)$ – вхідний, $y(t)$ – вихідний сигнали, а побудова макромоделі полягає у відшукуванні оператора F такого, щоб

$$\|y(t) - Fu(t)\| \leq \varepsilon \quad \text{при} \quad 0 \leq t \leq t_m. \quad (1.7)$$

Безпосередньо застосувати добре розвинений апарат поліноміальної апроксимації не вдається, бо між $y(t)$ і $u(t)$, як звичайно, нема однозначної залежності. Пропонують замінити скаляр $u(t)$ вектором $\bar{u}(t) = (u_1(t), \dots, u_n(t))$, щоб кожному значенню $\bar{u}(t)$ при $0 < t < t_m$ відповідало одне і тільки одне значення $y(t)$. Тоді оператор F можна апроксимувати поліномом, зокрема багатовимірним степеневим поліномом, а задача (1.7) набуває вигляду

$$\left\| y(t) - \sum_{i,j,\dots,k=0}^r c_{ij\dots k} u_1^i(t) u_2^j(t) \dots u_n^k(t) \right\| \leq \varepsilon \quad \text{при} \quad 0 \leq t \leq t_m, \quad (1.8)$$

і її можна розв’язати стандартними програмами.

Вектор $\bar{u}(t)$ називають розщепленим сигналом щодо $u(t)$, а його побудувати можна, зокрема, багаторазовим диференціюванням або інтегруванням сигналу $u(t)$:

$$\bar{u}(t) = \left(u(t), \frac{du(t)}{dt}, \frac{d^2u(t)}{dt^2}, \dots, \frac{d^{n-1}u(t)}{dt^{n-1}} \right);$$

$$\bar{u}(t) = \left(u(t), \int_0^t u(\tau) d\tau, \int_0^t \int_0^t u(\tau) d\tau^2, \dots, \int_0^t (n-1) \int_0^t u(\tau) d\tau^{n-1} \right).$$

Суттєво, що розщеплення виконують лінійними інерційними операторами, а нелінійне перетворення є однозначною стаціонарною функцією багатьох аргументів.

Блок-схема макромоделі зображена на рис.1.8. Вона відповідає нерекурсивній структурі, обґрунтування якої вперше наведене в [28]. Згодом метод поширено на рекурсивні структури [14].

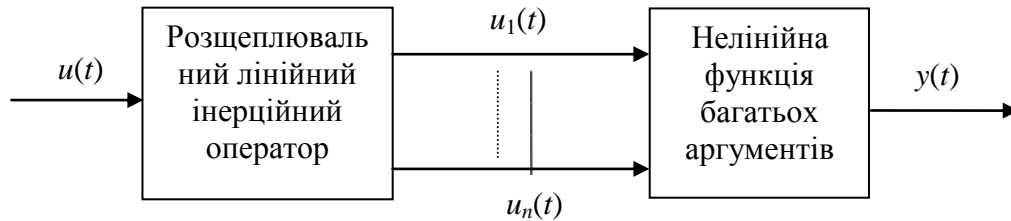


Рис.1.8. Блок-схема нерекурсивної макромоделі за теорією розщеплення

Структура макромоделі за методом розщеплення не є загальною. В розділі 3 доведено, що таку структуру, як і модель Вінера (див. нижче), не можна вивести із загального опису нелінійної динамічної системи в змінних стану, крім деяких спрощених випадків.

До другого підходу належать інтегральні рівняння, які широко використовують для макромодельовання [9].

Лінійні системи компактно описує інтеграл згортки

$$y(t) = \int_0^t G(t, \tau) u(\tau) d\tau, \quad (1.9)$$

який є чинним для широкого класу нестационарних систем з розподіленими параметрами.

Для опису нелінійних систем застосовують інтегральне рівняння Гамерштейна [15, 29]

$$y(t) = \int_0^t G(t, \tau) F(u(\tau), y(\tau)) d\tau, \quad (1.10)$$

якому відповідає блок-схема на рис.1.9.

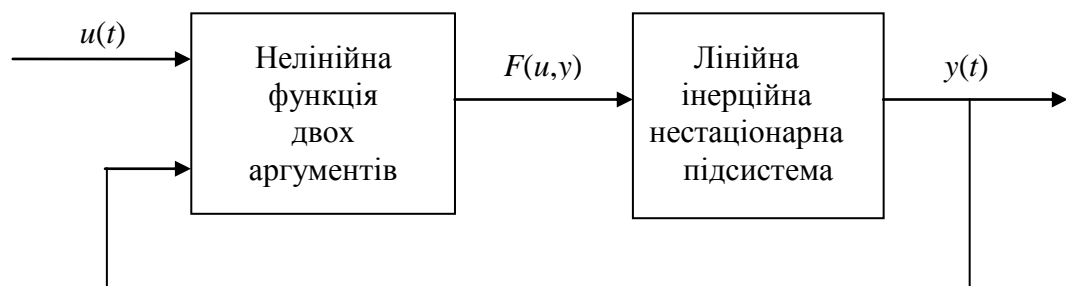


Рис.1.9. Блок-схема макромоделі Гамерштейна

У розділі 3 показано, що макромодель Гамерштейна є окремим випадком загального опису в системі змінних стану.

Використання макромоделей (1.9) та (1.10) обмежене із-за складностей їх опису в традиційних середовищах. Зокрема, нема відповідних засобів опису інтегральних операторів у вхідних мовах систем автоматизованого проектування. Цю проблему детально проаналізовано в розділі 2.

Узагальненням інтегралу згортки (1.9) на нелінійні системи є функціональний ряд Вольтерра [30, с.65]:

$$y(t) = \sum_{i=1}^{\infty} \int_0^t \dots \int_0^t G_i(t, \tau_1, \dots, \tau_i) \prod_{j=1}^i u(\tau_j) d\tau_1 \dots d\tau_i, \quad (1.11)$$

де $G_i(t, \tau_1, \dots, \tau_i)$ – ядро (вагова функція) i -го порядку.

Макромоделі у вигляді (1.11) справджуються для практично довільних систем. Необхідною умовою є лише неперервність залежності виходу від входу [16]. Прийнятні алгоритми експериментального визначення ядер $G_i(t, \tau_1, \dots, \tau_i)$ запропонував ще Н.Вінер [4]. Однак три обставини серйозно загальмували використання макромоделей (1.11). По-перше, функціональні вирази для ядер стрімко ускладнюються із збільшенням порядку ядра. По-друге, ряд (1.11) у загальному випадку є розбіжним, а швидко збігається лише для слабконелінійних систем. Нарешті, багатовимірні інтеграли складно вписати у традиційні середовища.

Суттєвий прогрес у використанні рядів (1.11) для макромодельовання пов'язаний із поміченою аналогією між рядами Вольтерра та рядами Вольтерра-Пікара.

Нехай рівняння системи можна записати у досить загальному вигляді

$$\begin{aligned} \dot{\bar{x}} &= A\bar{x} + \bar{f}(\bar{x}) + \bar{u}(t), \\ \bar{y} &= C\bar{x}, \end{aligned} \quad (1.12)$$

де A – матриця лінійної частини; $\bar{f}(\bar{x})$ – нелінійна поліноміальна вектор-функція. Тоді i -та ітерація Пікара ($i > 1$) розв'язку системи (1.12) має вигляд

$$\begin{aligned} \bar{x}_i(t) &= (pE - A)^{-1}(\bar{f}(\bar{x}_{i-1}(t)) + \bar{u}(t)), \\ \bar{y}_i(t) &= C\bar{x}_i(t), \end{aligned} \quad (1.13)$$

а загальний розв'язок є

$$\bar{y}(t) = C\bar{x}_0(t) + \sum_{i=1}^{\infty} (pE - A)^{-1}(\bar{f}(\bar{x}_{i-1}(t)) + \bar{u}(t)), \quad (1.14)$$

де $\bar{x}_0(t) = (pE - A)^{-1}\bar{u}(t)$.

Л.В.Данілов довів [10], що i -й член ряду (1.11) для системи (1.12) збігається з $(i-1)$ -м членом ряду (1.14). Отже, для систем (1.12) маємо спосіб відшукування членів функціонального ряду Вольтерра без громіздкого обчислення ядер, що значно спрощує задачу.

Наведена аналогія усуває перше і третє обмеження щодо застосування функціональних рядів Вольтерра, однак не вирішує проблеми збіжності. Як наслідок, ряди Вольтерра-Пікара є ефективним методом макромоделювання лише для слабконелінійних систем.

Суттєвий внесок у розуміння проблеми макромоделювання нелінійних динамічних систем зробили автори методу групового обліку аргументів (МГОА) [12]. Зокрема, у побудові макромоделей нелінійних динамічних систем ними застосовано багатомірні степеневі поліноми для апроксимації нелінійних вектор-функцій $\bar{f}_1(\cdot)$ та $\bar{f}_2(\cdot)$ у моделюючій системі вигляду

$$\bar{y} = \bar{f}_1(t, \bar{u}) + \bar{f}_2(\bar{y}, \dot{\bar{y}}, \ddot{\bar{y}}, \dots), \quad (1.15)$$

якій відповідає блок-схема на рис.1.10.

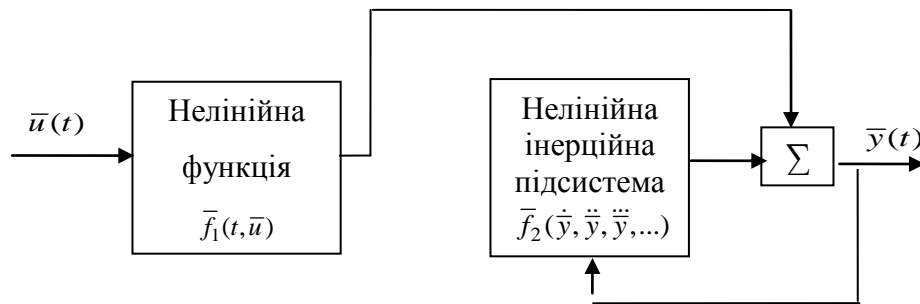


Рис.1.10. Блок-схема однієї з макромоделей за МГОА

Систему (1.15) названо такою, на якій може ґрунтуватись “найбільш загальний системний аналіз” [12, с.20]. Однак це твердження ніяк не обґрунтовано. Багато уваги в алгоритмах МГОА звернено на проблему некоректності задачі макромоделювання, хоча ніде саме так вона не названа. Алгоритми самоорганізації та забезпечення завадостійкості моделей, що зводяться до обмеження розмірності математичної моделі, є, по суті, конкретним розвитком методу регуляризації на компактi [31].

Важливий напрям макромоделювання пов’язаний зі структурою математичної моделі у вигляді системи звичайних алгебро-диференціальних рівнянь у формі Коші:

$$\begin{aligned} \dot{\bar{x}} &= \bar{f}(\bar{x}, \bar{u}, t), \\ \bar{y} &= \bar{\varphi}(\bar{x}, \bar{u}, t), \end{aligned} \quad (1.16)$$

де $\bar{f}(\bar{x}, \bar{u}, t)$ та $\bar{\varphi}(\bar{x}, \bar{u}, t)$ – дійсні вектор-функції дійсних аргументів.

Загальні методи ідентифікації рівнянь (1.16) невідомі. Теоретичні підходи до ідентифікації викладені в праці Ван дер Шафта [32]. У [5] запропоновано практичний метод ідентифікації рівнянь лінійного стаціонарного наближення (1.17), відомий як метод Хо-Калмана:

$$\begin{aligned}\dot{\bar{x}} &= A\bar{x} + B\bar{u}, \\ \bar{y} &= C\bar{x} + D\bar{u},\end{aligned}\tag{1.17}$$

де A, B, C, D – дійсні матриці відповідних розмірів.

Цей метод полягає в опрацюванні синхронних дискретних рівноінтервальних відліків вхідних та вихідних сигналів, сформованих у вигляді матриці Хенкеля:

$$H = \begin{bmatrix} \Gamma^{(1)} & \Gamma^{(2)} & \Gamma^{(3)} & \dots \\ \Gamma^{(2)} & \Gamma^{(3)} & \Gamma^{(4)} & \dots \\ \Gamma^{(3)} & \Gamma^{(4)} & \Gamma^{(5)} & \dots \\ \vdots & & & \end{bmatrix},\tag{1.18}$$

де елемент $\Gamma_{jk}^{(i)}$ підматриці $\Gamma^{(i)}$ визначає реакцію на j -му виході у разі подавання на k -й вхід одиничного стрибка в i -й момент часу. Розмір матриці H збільшують, доки зростає її ранг, що й визначає порядок макромоделі. Матриці системи (1.17) визначають із сингулярного розкладу $H = Q \cdot \Lambda \cdot R$, де Λ – діагональна матриця сингулярних чисел матриці Хенкеля.

У [33] описаний підхід узагальнено на білінійні системи

$$\begin{aligned}\dot{\bar{x}} &= A\bar{x} + N_1\bar{x} \cdot N_2\bar{x} + B\bar{u}, \\ \bar{y} &= C\bar{x} + D\bar{u},\end{aligned}\tag{1.19}$$

а в [34] аналогічний підхід запропоновано для систем більш загального вигляду з поліноміальними нелінійностями:

$$\begin{aligned}\dot{\bar{x}} &= A\bar{x} + \sum_{i,j} N_i \bar{x}^i \cdot N_j \bar{u}^j + B\bar{u}, \\ \bar{y} &= C\bar{x} + D\bar{u}.\end{aligned}\tag{1.20}$$

Викладений підхід вдало поєднує досить загальний вигляд структури (1.20) із процедурою ідентифікації, забезпечує мінімальну розмірність макромоделі. Однак ніде у працях, присвячених обґрунтуванню методу, не згадано проблему некоректності сформульованої задачі.

Чимало зусиль з удосконалення, практичної реалізації та перевірки останнього методу доклали автори [21, 22]. Отримані результати

переконливо свідчать про відсутність процедур регуляризації у методі Хо-Калмана та споріднених з ним. Наприклад, ідентифікація лінійної системи за даними з похибками приводить до нестійких результатів, що є типовим проявом некоректності.

Автори пропонують регуляризувати задачу застосуванням методів нелінійної оптимізації. Для нелінійних систем розроблені процедури обчислення матриць N_1 та N_2 з (1.19) методами нелінійної пошукової оптимізації.

У праці [35] зроблена спроба обґрунтувати білінійні моделі вигляду

$$\begin{aligned}\dot{y}_1 &= y_2, \\ \dot{y}_2 &= y_3, \\ &\vdots \\ \dot{y}_n &= \sum_{i=1}^n a_i y_i + \sum_{i=0}^m b_i u^{(i)} + \sum_{i=0}^m \sum_{j=0}^m c_{ij} u^{(i)} u^{(j)}\end{aligned}\tag{1.21}$$

або

$$\begin{aligned}\dot{y}_1 &= y_2, \\ \dot{y}_2 &= y_3, \\ &\vdots \\ \dot{y}_n &= \sum_{i=1}^n a_i y_i + \sum_{i=0}^m b_i u^{(i)} + \sum_{i=0}^m \sum_{j=1}^n c_{ij} u^{(i)} y_j\end{aligned}\tag{1.22}$$

з використанням апарату диференціальних перетворень Тейлора, запропонованого Г.Є.Пухова. Очевидно, системи (1.21) та (1.22) є окремими випадками систем (1.19) та (1.20).

Ще один окремий випадок системи (1.19) досліджено у [8], а саме систему вигляду

$$\begin{aligned}\dot{y}_1 &= y_2, \\ \dot{y}_2 &= y_3, \\ &\vdots \\ \dot{y}_n &= P(y_1, y_2, \dots, y_n, u),\end{aligned}\tag{1.23}$$

де $P(\cdot)$ – багатовимірний степеневий поліном степеня m :

$$P(\cdot) = \sum_{i=0}^m \sum_{j=0}^m \dots \sum_{k=0}^m c_{ij\dots k} y_1^i y_2^j \dots u^k, \quad i + j + \dots + k \leq m.\tag{1.24}$$

За структурою (1.23) вперше побудовані компактні макромоделі амплітудного детектора, подільників частоти та генераторів періодичних сигналів заданої форми [8]. Щодо вирішення проблеми некоректності, то у [8] до системи (1.23) уведено спеціальну регуляризуючу функцію, яку будують з використанням ідей другого методу Ляпунова.

Дещо згодом було опубліковано метод визначення розмірності простору вкладення та побудови ортогонального базиса незалежних змінних для нелінійного динамічного об'єкта за однією вихідною змінною [36], у якому використано структуру рівнянь моделі, подібну до (1.23). З цією метою для рівномірного кроку за даними τ формують коваріаційну матрицю

$$S = \begin{bmatrix} \sigma_{11} & \dots & \sigma_{1n} \\ \vdots & & \\ \sigma_{n1} & \dots & \sigma_{nn} \end{bmatrix} \quad (1.25)$$

з елементами $\sigma_{kk} = \sum_{j=0}^n y^2(t_k + j\tau)$; $\sigma_{ki} = \sigma_{ik} = \sum_{j=0}^n y(t_k + j\tau) \cdot y(t_i + j\tau)$.

Власні вектори S_i матриці S визначають ортогональний базис для нових змінних, перехід до яких виконується згідно з виразом

$$z_{ik} = y_k \cdot S_i / |S_i|, \quad (1.26)$$

де $y_k = (y(t_k), y(t_k + \tau), \dots, y(t_k + n\tau))$.

У праці [37] запропоновано метод відтворення системи диференціальних рівнянь об'єкта за однією вихідною змінною, де як базис прийнято аналог базису (1.26), а саме:

$$x_k = (y(t_k), \frac{dy(t_k)}{dt}, \dots, \frac{d^n y(t_k)}{dt^n}). \quad (1.27)$$

У такому разі використання (1.26) зводиться до обчислення потрібної кількості похідних вихідної функції в усіх заданих точках. Система рівнянь n -го порядку, для якої виконується відтворення в [37], має вигляд

$$\begin{aligned} \dot{y}_1 &= y_2, \\ \dot{y}_2 &= y_3, \\ &\vdots \\ \dot{y}_n &= F(y_1, y_2, \dots, y_n) / f(y_i), \end{aligned} \quad (1.28)$$

де: $F(y_1, y_2, \dots, y_n)$ – багатовимірний степеневий поліном; $f(y_i)$ – одна з функцій: 1; y_i ; y_i^2 ; $y_i y_j$; $i, j=1, \dots, n$.

Однак вибір системи (1.28) ніяк не обґрунтовано, а також ніде не згадано про некоректність сформульованих задач, зокрема, обчислення похідних (1.27).

Привертає увагу дослідників моделююча структура, запропонована Н.Вінером [38] (рис.1.11).

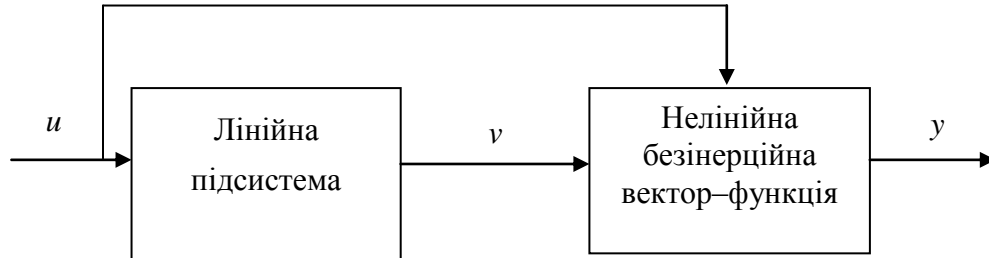


Рис.1.11. Структура моделі Вінера

Популярність моделі Вінера зумовлена простотою ідентифікації завдяки тому, що нема рекурсій. Проте використовують таку структуру зазвичай без належного визначення класу систем, на які вона поширюється.

Отже, обґрунтування моделюючої структури є актуальним і потребує подальших досліджень. Доцільно звернути увагу на обґрунтування розділення моделюючої структури на лінійну та нелінійну підсистеми. Як звичайно, дослідники ніяк не обґрунтовують таке розділення. Іноді складається враження, що потребу такого обґрунтування не завжди усвідомлюють.

1.3.4. Методи ідентифікації математичних макромоделей

Розглянемо тепер методи ідентифікації наведених структур математичних моделей.

Важливим питанням є вибір множин сигналів, на яких виконують ідентифікацію. Із загальних міркувань бажано обрати вхідний (збудувальний) сигнал, який здатний забезпечити якомога більш універсальну процедуру ідентифікації. Н.Вінер [4, с.11] як такий вхідний сигнал запропонував випадковий сигнал “білий шум”. Для синтезу моделі нелінійного пристрою у вигляді функціонального ряду Вольтерра (1.11) вихідний сигнал можна апроксимувати ортонормованим рядом базисних функцій на основі багаточленів Ерміта, а коефіцієнти апроксимації обчислювати як коефіцієнти узагальненого ряду Фур’є. Однак використання цієї ідеї нашою хується на серйозні труднощі у зв’язку з громіздкістю відповідних числових процедур та поганою збіжністю функціональних рядів Вольтерра.

На практиці перевагу надають детермінованим сигналам, які підбирають для кожного об’єкта ідентифікації зокрема. Лише для лінійних стаціонарних систем завдяки принципу суперпозиції немає проблеми вибору

сигналу ідентифікації, оскільки підходить довільний сигнал, аби лише він викликав зручний для реєстрації перехідний процес в об'єкті. Поширена ідентифікація лінійних систем за амплітудно-фазово-частотними характеристиками, що відображають встановлені процеси під час моногармонічного збурення. Однак така ідентифікація принципово неможлива для систем нелінійних, де взагалі нема поняття частотної характеристики.

Класичне формулювання задачі ідентифікації оминає проблему вибору сигналів, а зосереджене на методах визначення параметрів моделі для заданої структури моделі та заданих сигналів. У цьому випадку для кожної задачі сигнали, на яких виконується ідентифікація, формуються окремо, на підставі відомих особливостей об'єкта та обраної структури математичної моделі.

Суттєво, що для автономних систем взагалі нема збурювального сигналу, а ідентифікацію виконують лише за вихідним сигналом.

Як впливає з наведеного огляду, ідентифікація математичної моделі у вигляді системи диференціальних рівнянь полягає, як звичайно, у визначенні внутрішнього вектора моделі та апроксимації функцій, що формують лінійну та нелінійну підсистеми.

Природним є намагання розділити процедури ідентифікації лінійної та нелінійної підсистем. Це легко вдається, якщо можна окремо ідентифікувати статичну характеристику, яка відповідає нелінійній безінерційній підсистемі, та малосигнальну динамічну характеристику, що відповідає лінійній підсистемі. Методи, що розв'язують саме таку задачу ідентифікації, можна об'єднати спільною назвою “методи нелінійної статистики та малосигнальної динаміки”.

Окремі методи ідентифікації нерекурсивних систем, зокрема такі, що відповідають структурі Вінера та нерекурсивній структурі Гамерштейна, розглянуті в [15, 39]. Вичерпно підхід нелінійної статистики та малосигнальної динаміки досліджено у працях [10, 40, 41], де показано, що ідентифікації за цими методами підлягають лише системи з “коротким” перехідним процесом, який практично завершується раніше, ніж виявляють себе нелінійні властивості системи, викликані порівняно повільним вхідним сигналом. Клас таких систем досить вузький.

У параграфі 3.2 викладено та проілюстровано метод нелінійної статистики та малосигнальної динаміки для рекурсивних систем, поширений на деякі неоднозначні нелінійності.

Однак далеко не завжди можна виділити статичну характеристику системи. Наприклад, для автогенераторних систем такого поняття взагалі нема, однак нелінійну безінерційну підсистему все ж таки можна ідентифікувати.

Оригінальний метод ідентифікації таких систем (метод оберненої лінійної підсистеми), що поширюється на значно ширший клас систем, ніж досі, описано в параграфі 3.3.

Можливий граничний випадок, коли лінійна підсистема зводиться до ланцюжкової послідовності ідеальних інтеграторів. Саме така ситуація

виникає для систем (1.23) та (1.28). У цьому випадку лінійну підсистему вважають заданою, а ідентифікації підлягають лише нелінійні безінерційні підсистеми, що їх описують функції багатьох аргументів: $P(y_1, y_2, \dots, y_n, u)$ для системи (1.23) або $F(y_1, y_2, \dots, y_n)/f(y_i)$ для системи (1.28). Аналогічний підхід застосував автор у параграфі 4.1 для ідентифікації автогенераторів, що містять дивні атрактори.

Взагалі центральним завданням макромоделювання є ідентифікація нелінійної безінерційної підсистеми. Це значно складніше, ніж ідентифікація лінійної підсистеми і зводиться для окресленого класу систем до апроксимації нелінійної дійсної вектор-функції багатьох дійсних аргументів. Загальне формулювання такої задачі у разі макромоделювання наведено в розділі 4.

Вхідні та вихідні сигнали, за якими ідентифікують макромоделі, описані у вигляді дискретних функцій часу, причому далеко не обов'язково на рівномірній сітці. Отже, ідентифікація нелінійної підсистеми – це апроксимація нелінійної функції багатьох аргументів на нерівномірній сітці. Ця задача далека від повного вирішення. В розділі 4 ми спробували розв'язати її для окремих практичних задач макромоделювання.

Ідентифікаційні підходи, засновані на використанні матриці Хенкеля (1.18) або коваріаційної матриці (1.25), розраховані на рівномірний крок за даними. Це ускладнює їхнє використання, оскільки потребує принаймні інтерполяційного вирівнювання кроку з неминучим внесенням похибок.

Аналогічна ситуація із запропонованим [42] підходом до апроксимації функцій багатьох змінних з умовою рівномірного кроку за всіма аргументами. Така умова практично нереальна.

Макромоделювання в цілому належить до так званих обернених задач, унаслідок чого ідентифікація макромоделей значно ускладнена проявами некоректностей. Це означає, що розв'язок ідентифікаційних задач макромоделювання загалом є нестійким, хоча існує і є єдиним [31].

Типовим проявом некоректності є ситуація, коли макромоделі, побудована за бездоганною зовні процедурою ідентифікації, поводить себе зовсім не так, як очікували.

Усунення некоректностей полягає у визначенні джерел їхнього виникнення та розробці методів їх подолання. Загальна теорія регуляризації некоректних задач зводиться до заміни некоректної задачі близькою до неї коректною. Однак специфіка макромоделних задач потребує особливих прийомів використання відомих методів регуляризації та розробки оригінальних.

Застосування та розробка регуляризованих методів ідентифікації макромоделей ще не визнанні дослідниками. Водночас без регуляризації не можна розв'язати будь-яку серйозну задачу макромоделювання. Учені змушені регуляризувати свої задачі, однак роблять це без використання здобутків математиків, що не може сприяти успіху справи. Як звичайно, регуляризаційні прийоми навіть не називають такими.

Ідеї самоорганізації, незалежно запропоновані розробниками методу МГОА, перегукуються з ідеями регуляризації на компактні [31]. С.А.Букашкін запропонував використати другий метод Ляпунова для забезпечення стійкості макромоделей [8], що повністю відповідає теорії регуляризаційного функціонала. П.Г.Стахів був змушений застосувати нелінійну оптимізацію для одержання стійких розв'язків [22], що збігається з ітераційними схемами регуляризації.

Регуляризовані методи ідентифікації для задач макромодельовання розглянуті в розділі 4. Всі методи ілюстровані прикладами макромоделей динамічних систем різної складності та фізичної природи.

РОЗДІЛ 2

МАКРОМОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ

2.1. ІНТЕГРАЛЬНІ ОПЕРАТОРИ

Лінійну систему із скалярними входом та виходом іноді зручно описати інтегральним рівнянням

$$y(t) = \int_0^t g(t, \tau) u(\tau) d\tau; \quad y(0) = 0, \quad (2.1)$$

де $g(t, \tau)$ – вагова або імпульсна перехідна функція системи [3].

У багатовимірному випадку

$$\bar{y}(t) = \int_0^t G(t, \tau) \bar{u}(\tau) d\tau; \quad \bar{y}(0) = 0, \quad (2.2)$$

де $G(t, \tau)$ – матриця вагових функцій відповідної розмірності.

Для моделі (2.1) ідентифікація полягає у відшукуванні коефіцієнтів апроксимації \bar{p} вагової функції $g_{\bar{p}}(t, \tau)$, що забезпечать найменше відхилення реакції $\bar{y}(t; \bar{p}) = \int_0^t g_{\bar{p}}(t, \tau) u(\tau) d\tau$ від заданої реакції $y(t)$ із заданим збуренням $u(t)$.

Описи (2.1) та (2.2) є лаконічними та універсальними (справджуються для систем нестационарних та з розподіленими параметрами). Однак безпосереднє використання їх як макромоделей не завжди зручне. Річ у тім, що макромодель використовують, як звичайно, в оточенні інших елементів системи, для опису яких традиційним є апарат диференціальних рівнянь. Звідси випливає потреба, щоб опис макромоделі був зручним для аналізу стандартними методами розв'язування диференціальних рівнянь. Модель (2.1) цій потребі формально відповідає, якщо вагова функція сепарабельна, тобто $g(t, \tau) = g_1(t) \cdot g_2(\tau)$.

Справді, у цьому випадку (2.1) можна записати

$$y(t) = g_1(t) \int_0^t g_2(\tau) u(\tau) d\tau, \quad y(0) = 0, \quad (2.3)$$

і рівнянню (2.3) еквівалентна система алгебро-диференціальних рівнянь

$$\begin{aligned}\frac{dx(t)}{dt} &= g_2(t)u(t); \\ y(t) &= g_1(t)x(t); \\ x(0) &= 0.\end{aligned}$$

На жаль, сепарабельність вагової функції є скоріше винятком, ніж правилом.

Для стаціонарних систем із зосередженими параметрами природною апроксимацією вагової функції є сума зважених експонент

$$g(t) = \sum_{k=1}^n a_k \exp(b_k t). \quad (2.4)$$

У цьому випадку вагова функція є сепарабельною і (2.3) набуває вигляду

$$y(t) = \sum_{k=1}^n a_k \exp(b_k t) \int_0^t \exp(-b_k \tau) u(\tau) d\tau, \quad y(0) = 0,$$

а еквівалентна система звичайних диференціальних рівнянь є такою:

$$\begin{aligned}\dot{x}_k &= \exp(-b_k t) u(t); \quad k = \overline{1, n}; \\ y &= \sum_{k=1}^n a_k \exp(b_k t) x_k; \\ x_k(0) &= 0; \quad k = \overline{1, n}.\end{aligned} \quad (2.5)$$

Однак числове інтегрування системи (2.5) є невдячною справою внаслідок катастрофічного накопичення числових похибок обмеженої розрядності ЕОМ. Значно зручніше інтегрувати еквівалентну систему [3]

$$\begin{aligned}\dot{x}_k &= b_k x_k + a_k u(t); \quad k = \overline{1, n}; \\ y &= \sum_{k=1}^n x_k; \\ x_k(0) &= 0; \quad k = \overline{1, n};\end{aligned}$$

однак тоді нічого не залишається від інтегрального оператора (2.2)!

Опишемо, як можна зобразити інтеграл згортки (2.2) за допомогою диференціальних рівнянь, уникаючи потреби в сепарабельності вагової функції [43].

Обмежимося стаціонарними системами в скалярному випадку. Тоді інтеграл згортки (2.2) набуває вигляду

$$y(t) = \int_0^t g(t-\tau)u(\tau)d\tau, \quad y(0) = 0. \quad (2.6)$$

Інтегруванням за частинами отримуємо

$$\begin{aligned} y(t) &= \int_0^t \frac{d}{dt} \int_0^\tau u(\xi)d\xi g(t-\tau)d\tau = \\ &= g(t-\tau) \int_0^\tau u(\xi)d\xi \Big|_0^t + \int_0^t g^{(1)}(t-\tau) \int_0^\tau u(\xi)d\xi d\tau = \\ &= g(0) \int_0^t u(\xi)d\xi + g^{(1)}(t-\tau) \int_0^\tau u(\xi)d^2\xi \Big|_0^t + \int_0^t g^{(2)}(t-\tau) \int_0^\tau u(\xi)d^2\xi d\tau = \\ &= \dots = \sum_{i=0}^{\infty} g^{(i)}(0) \int_0^t (i+1) \int_0^\tau u(\xi)d^{i+1}\xi. \end{aligned} \quad (2.7)$$

Якщо скористатись іншою формою інтеграла згортки

$$y(t) = \int_0^t g(\tau)u(t-\tau)d\tau,$$

то розклад виглядатиме так:

$$y(t) = \sum_{i=0}^{\infty} (-1)^i g^{(i)}(t) \int_0^t (i+1) \int_0^\tau u(\xi)d^{i+1}\xi. \quad (2.8)$$

Ряди (2.7) та (2.8) дають змогу побудувати макромоделі, які не потребують сепарабельності вагових функцій для зручного зображення звичайними диференціальними рівняннями та справджуються для широкого класу лінійних систем, у тім числі систем з розподіленими параметрами. Відповідна система для ряду (2.7) є така:

$$\begin{aligned} \dot{x}_1 &= u; \quad \dot{x}_2 = x_1; \quad \dot{x}_i = x_{i-1}; \dots \\ y(t) &= \sum_{i=0}^{\infty} g^{(i)}(0)x_{i+1}; \\ x_i(0) &= 0; \quad i = \overline{1, \infty}. \end{aligned} \quad (2.9)$$

Аналогічний вигляд має диференціальна система для ряду (2.8):

$$\begin{aligned}\dot{x}_1 &= u; \quad \dot{x}_2 = x_1; \quad \dot{x}_i = x_{i-1}; \dots \\ y(t) &= \sum_{i=0}^{\infty} (-1)^i g^{(i)}(t) x_{i+1}; \\ x_i(0) &= 0; \quad i = \overline{1, \infty}.\end{aligned}\tag{2.10}$$

Практичне використання (2.9) або (2.10) можливе у випадку обрізання відповідних рядів. Якщо використати експоненціальну апроксимацію вагової функції (2.4), то ряди надто повільно збігаються, а їхні частинні суми істотно залежать від довжини розрядної сітки ЕОМ. Ці складності вдається оминати, застосовуючи інші апроксимації, наприклад, степеневим поліномом на деякому інтервалі $[0, \tau]$:

$$g(t) \cong \sum_{i=0}^n a_i t^i, \quad t \in [0, \tau].\tag{2.11}$$

У цьому випадку лише перші n похідних функції $g(t)$ відмінні від нуля, а точність апроксимації (2.11) визначає точність обчислення реакцій за $(n+1)$ першими членами рядів (2.9) або (2.10) на інтервалі $[0, \tau]$.

Особливо зручним є застосування розкладу (2.9) з апроксимацією (2.11). Похідні функції $g(0)$ обчислюють один раз для довільного вхідного сигналу за формулою

$$g^{(i)}(0) = a_i i!$$

Для розкладу (2.10) обчислення похідних є громіздкішим для кожного t :

$$\begin{aligned}g^{(0)}(t) &= \sum_{j=0}^n a_j t^j; & g^{(1)}(t) &= \sum_{j=0}^n j a_j t^{j-1}; \\ g^{(2)}(t) &= \sum_{j=2}^n j(j-1) a_j t^{j-2}; & \dots & \quad g^{(n)}(t) = n! a_n.\end{aligned}$$

Звичайно, залишається непросте завдання апроксимації вагової функції степеневим поліномом із заданою точністю. У зв'язку з цим заслуговують на увагу інші апроксимації, які хоча й не забезпечують кінцевої

кількості членів розкладу, однак є зручнішими для відтворення вагових функцій [44].

2.2. ДИФЕРЕНЦІАЛЬНІ ОПЕРАТОРИ

Якщо лінійна система стаціонарна, то $g(t, \tau) = g(t - \tau)$. Якщо, крім цього, система має зосереджені параметри, то перетворення Лапласа для вагової функції $g(t)$ дає дробово-раціональну передатну функцію:

$$\int_0^{\infty} \exp(-\lambda \tau) g(\tau) d\tau = \frac{\sum_0^m b_i \lambda^i}{\sum_0^n a_i \lambda^i} = W(\lambda); \quad m \leq n. \quad (2.12)$$

Передавальній функції $W(\lambda)$ відповідає диференціальне рівняння n -го порядку

$$\sum_0^n a_i \frac{d^i y(t)}{dt^i} = \sum_0^m b_i \frac{d^i u(t)}{dt^i} \quad (2.13)$$

або система n диференціальних рівнянь у формі Коші ($a_n = 1$)

$$\begin{aligned} \dot{y}_0 &= y_1; \\ \dot{y}_1 &= y_2; \\ &\vdots \\ \dot{y}_{n-1} &= \sum_0^m b_i u^{(i)}(t) - \sum_0^{n-1} a_i y_i(t). \end{aligned} \quad (2.14)$$

Системі рівнянь (2.14) завжди можна поставити у взаємну відповідність еквівалентну систему рівнянь щодо вектора змінних стану \bar{x} :

$$\begin{aligned} \dot{\bar{x}} &= A\bar{x} + Bu; \\ y &= C\bar{x} + Du. \end{aligned} \quad (2.15)$$

де: $\bar{x} = (x_1, x_2, \dots, x_n)$ – внутрішній вектор; A, B, C, D – матриці розмірністю $[n \times n]$, $[n \times r]$, $[s \times n]$, $[s \times r]$ відповідно [3].

Система рівнянь (2.15) є алгебро-диференціальним оператором, що пов'язує у загальному випадку вектори u , y та \bar{x} . Якщо відомі вектори

$\bar{x}(t_0), \bar{u}(t \in [t_0, t_1])$ та матриці A, B, C, D , то однозначно відомий вектор $y(t \in [t_0, t_1])$.

Загальний розв'язок системи рівнянь (2.15) має вигляд

$$y(t) = Ce^{A(t-t_0)}\bar{x}(t_0) + C \int_{t_0}^t e^{A(t-\tau)} Bu(\tau) d\tau + Du(\tau), \quad (2.16)$$

де $e^{At} = \sum_{i=0}^{\infty} A^i t^i / i!$ – матрична експонента [3].

Очевидно, (2.16) збігається з (2.1) при скалярних u, y та нульових початкових умовах $\bar{x}(0)=0$, причому

$$g(t, \tau) = g(t - \tau) = Ce^{A(t-\tau)}B + D\delta(t - \tau), \quad (2.17)$$

де $\delta(t)$ – дельта-функція Дірака.

Нагадаємо поняття спостережуваної системи, яке використовується далі.

Лінійна система з описом (2.15) називається спостережуваною, якщо з відомими A, C , заданим $\bar{u}(t \in [t_0, t_1])$ та відомим $\bar{y}(t \in [t_0, t_1])$ можна знайти $\bar{x}(t_0)$. Система (2.15) є спостережуваною тоді й тільки тоді, коли

$$\text{rank}(C^t \quad A^t C^t \quad (A^t)^2 C^t \quad (A^t)^{n-1} C^t) = n \quad (2.18)$$

Система (2.14) є зручною для ідентифікації за входом-виходом. Справді, потрібно лише обчислити похідні вхідного та вихідного сигналів. Ідентифікаційна задача (1.5) у цьому випадку набуває вигляду

$$\min_{\bar{a}, \bar{b}} \sum_{k=1}^M (y^{(n)}(t_k) - \sum_{i=0}^m b_i u^{(i)}(t_k) + \sum_{i=0}^{n-1} a_i y^{(i)}(t_k))^2, \quad M \geq m + n + 1, \quad (2.19)$$

і зводиться до розв'язування системи лінійних алгебричних рівнянь порядку $(n+m+1)$ щодо компонент векторів $\bar{a}=(a_0, \dots, a_{n-1})$ та $\bar{b}=(b_0, \dots, b_m)$. Якщо функції $u^{(i)}(t)$, $i=\overline{1, m}$ та $y^{(i)}(t)$, $i=\overline{0, n-1}$ утворюють систему лінійно-незалежних функцій, то система (2.19) завжди сумісна, її розв'язок завжди існує та є єдиним.

Отже, щоб збудувати макромодель лінійної системи у формі (2.14), треба знати сигнали $u(t)$ та $y(t)$ в точках $t_k, k=1, \dots, M$, а також похідні $u^{(i)}(t)$, $i=1, \dots, m$ та $y^{(i)}(t)$, $i=1, \dots, n$ у тих самих точках t_k . Далі методом найменших квадратів достатньо розв'язати задачу (2.19), тобто визначити $n+m+1$ коефіцієнтів a_i та b_i , що забезпечують найменшу середньоквадратичну похибку апроксимації функції $y_{n-1}(t)$ з (2.14) у точках t_k .

Очевидно, макромодель у вигляді (2.14) має надлишковий порядок $(n+m)$, бо крім n основних диференціальних рівнянь містить ще m

диференціальних рівнянь для похідних вхідного сигналу. Такої надлишковості легко уникнути, якщо після відшукування коефіцієнтів рівнянь у вигляді (2.14), перейти до рівнянь у вигляді (2.15) [3]. В одному з варіантів такого переходу при $m=n$ відповідні матриці можуть мати вигляд

$$A = \begin{pmatrix} -a_{n-1} & 1 & 0 & \dots & 0 \\ -a_{n-2} & 0 & 1 & \dots & 0 \\ \vdots & & & & \\ -a_0 & 0 & 0 & \dots & 0 \end{pmatrix}; \quad B = \begin{pmatrix} b_{n-1} & -a_{n-1} & b_n \\ b_{n-2} & -a_{n-2} & b_n \\ \vdots & \vdots & \vdots \\ b_0 & -a_0 & b_n \end{pmatrix}; \quad C = (10\dots 0); \quad D = b_n. \quad (2.20)$$

Отже, сформульована задача макромодельовання лінійних систем із зосередженими сталими параметрами має завершений вигляд, а основна задача (2.19) є лінійною стосовно шуканих векторів \bar{a} та \bar{b} , її розв'язок завжди існує та є єдиним.

2.3. НЕКОРЕКТНОСТІ МАКРОМОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ. РЕГУЛЯРИЗАЦІЯ ОБЧИСЛЕННЯ ПОХІДНИХ

Пряма обчислювальна реалізація описаного процесу макромодельовання пов'язана з суттєвими труднощами, що помітно стримують її поширення. Річ у тім, що процедури послідовного числового диференціювання, а також розв'язування задач (1.5) або (1.6) належать до так званих некоректних задач, уведених на початку сторіччя Адамаром (J.Hadamard) як такі, що не мають реального змісту і тому їх не потрібно розглядати.

За Адамаром, коректні задачі повинні задовольняти умови існування та єдиності розв'язку, а також стійкості розв'язку стосовно збурень умов задачі. Наведені вище задачі, а також усі розглянуті в монографії, крім найпростіших, мають ту властивість, що їхній розв'язок неприпустимо сильно залежить від малих змін (похибок) умов задачі (нестійкий стосовно збурень), хоча існує та є єдиним [31]. Унаслідок цього похідні вищих порядків у (2.19), а також розв'язки задач (1.5) або (1.6) виявляються недостовірними.

Загальна теорія регуляризації лінійних некоректних задач, до яких належать згадані задачі, розроблена школою Тихонова [31], а також американськими дослідниками [45]. Зокрема, один із підходів до регуляризації задачі (1.5) зводиться до заміни її коректною задачею

$$\min_{\bar{p}} \left(\sum_{k=1}^M (y(t_k) - \tilde{y}(t_k; \bar{p}))^2 + \alpha \sum_{i=1}^N p_i^2 \right), \quad (2.21)$$

де значення параметра регуляризації $\alpha > 0$ певним чином залежить від похибок даних $\delta u(t_k)$, причому $\alpha \rightarrow 0$ при $\delta u(t_k) \rightarrow 0$ для всіх $t_k, k = \overline{1, M}$.

Методам регуляризації головно присвячений розділ 4. Розглянемо тут лише некоректність диференціювання функцій дискретного аргументу, що має суттєве практичне значення у разі макромодельованні лінійних систем.

Похідні можна обчислювати різними методами [46]. У параграфі 4.1 описано використання інтерполяційних сплайнів. Ефективним є також метод диференціювання за допомогою ковзного апроксимаційного полінома (див. 2.4.3, Додаток.2). Тут лише проаналізовано похибки та вироблено практичні рекомендації щодо числового диференціювання за допомогою звичайних диференціальних рівнянь.

Неважко продемонструвати некоректність диференціювання. Нехай $u_1(t)$ – деяка неперервна функція, а $z_1(t)$ – її похідна. Функція $u_2(t) = u_1(t) + A \sin(\omega t)$ відрізняється від функції $u_1(t)$ в метриці C на величину $\rho_C(u_1, u_2) = |A|$ для довільних значень ω . Однак відстань між похідними $z_2(t) = du_2/dt$ та $z_1(t)$ дорівнює $\rho_C(z_1, z_2) = |A\omega|$ і може бути як завгодно великою при достатньо великому ω . Отже, у разі обмеженості відстані між функціями відстань між похідними функцій є необмеженою, що свідчить про нестійкість диференціювання.

Розглянемо детальніше похибки числового диференціювання. Задача числового диференціювання полягає у наближеному обчисленні похідних неперервної функції $u(t)$ за заданими в окремих точках значеннями цієї функції. Наближене значення похідної можна відшукувати за одним з таких різницевих співвідношень:

$$u_{t,i} = (u_i - u_{i-1})/h; \quad u_{\bar{t},i} = (u_{i+1} - u_i)/h; \quad u_{\tilde{t},i} = (u_{i+1} - u_{i-1})/2h;$$

де: h – крок дискретизації; $u_i = u(t_i)$.

Для першої формули оцінка похибки, що виникає внаслідок різницевої апроксимації, є такою [47]:

$$|u_{t,i} - \dot{u}(t_i)| = h/2 \cdot \ddot{u}(\xi_i), \quad (2.22)$$

де ξ_i – точка з інтервалу (t_{i-1}, t_i) .

Очевидно, чим менший крок h , тим меншою є різниця між наближеним значенням похідної та її точним значенням. Однак малість h якраз і є причиною некоректності. Справді, нехай замість точних значень u_i маємо наближені значення $\tilde{u}_i = u_i + \delta_i$. Тоді для першої з наведених різницевих формул маємо

$$\tilde{u}_{t,i} = (u_i - u_{i-1})/h + (\delta_i - \delta_{i-1})/h = u_{t,i} + (\delta_i - \delta_{i-1})/h.$$

Якщо задати межі $\delta \geq |\delta_i|$ та $\delta \geq |\delta_{i-1}|$, то для похибки різничевої похідної справджується оцінка

$$|\tilde{u}_{t,i} - u_{t,i}| \leq 2\delta/h. \quad (2.23)$$

Отже, оцінка похибки різничевої похідної $2\delta/h$ суттєво відрізняється від оцінки похибки δ самої функції. Якщо δ не залежить від h , то оцінка похибки різничевої похідної може необмежено зростати при $h \rightarrow 0$. Аналогічні оцінки є і для інших різницевих формул.

Для коректного використання різницевих формул диференціювання природно домагатися, щоб оцінки (2.22) та (2.23) були сумірні, наприклад,

$$\frac{h}{2} |\ddot{u}(\xi_i)| \geq 2\delta/h. \quad (2.24)$$

Тому числове диференціювання треба проводити не з як завгодно малим кроком h , а згідно із співвідношенням (2.24), яке задає нижню межу коректного кроку.

Розглянемо обчислення похідної за допомогою числового інтегрування відповідного диференціального рівняння

$$\frac{d\varphi(t)}{dt} = \frac{1}{\tau} (u(t) - \varphi(t)), \quad (2.25)$$

де $d\varphi(t)/dt \approx du(t)/dt$ з оцінкою абсолютної похибки

$$\Delta_{\max} = \tau \cdot \max \left| \frac{d^2\varphi(t)}{dt^2} \right|, \quad (2.26)$$

а стала часу τ не повинна бути надто малою з огляду на некоректність задачі.

Рівняння (2.25) збігається з рівнянням електричного кола на рис.2.1.

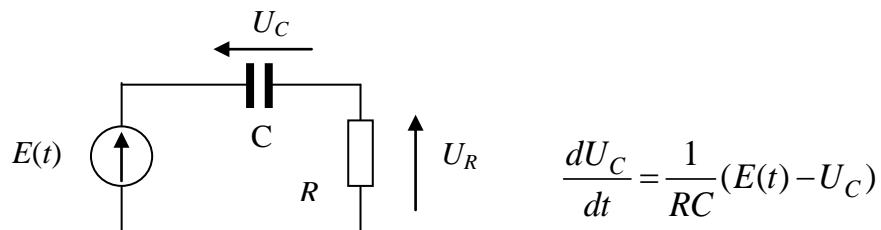


Рис. 2.1. Диференціююче електричне коло

Звідси

$$\frac{dE(t)}{dt} \approx \frac{U_R}{RC} \quad (2.27)$$

з оцінкою абсолютної похибки $\Delta_{\max} = RC \cdot \max |d^2 U_c / dt^2|$.

У разі обчислення похідних за допомогою сплайнів або ковзних інтерполюючих поліномів існують надійні методи регуляризації [46]. Проте у випадку використання (2.25) або (2.27) ситуація ускладнюється, оскільки зазвичай нема засобів безпосереднього впливу на крок інтегрування (програма числового інтегрування диференціальних рівнянь автоматично вибирає крок). Тому є корисними експерименти з оцінки таких параметрів рівняння (2.25), які забезпечують надійне обчислення похідних за заданими параметрами інтегратора.

Такі експерименти зроблені в системі схемотехнічного проектування САНОС-ПК, коротко описаній у розділі 5. Ці експерименти полягали в багаторазовому числовому диференціюванні гармонічної функції $u(t) = \cos(\omega t)$ з оцінкою точності диференціювання у разі зміни сталої часу $\tau = RC$ та максимального кроку інтегрування h . Вибір гармонічної функції зумовлений можливістю аналітичного обчислення точного значення похідної, простим зв'язком між оцінкою похідної і частотою, а також зручним узагальненням результатів.

Прийmemo для оцінки відносної похибки першої похідної вираз

$$\max |\varepsilon| = \Delta_{\max} / \max \left| \frac{du(t)}{dt} \right| = \tau \cdot \max \left| \frac{d^2 u(t)}{dt^2} \right| / \max \left| \frac{du(t)}{dt} \right|.$$

Тоді у випадку гармонічного сигналу $u(t) = \cos(\omega t)$ маємо

$$\max |\varepsilon| = \tau \omega^2 / \omega = \omega \tau, \quad (2.28)$$

тобто оцінка відносної похибки визначається добутком $\omega \tau$.

Формулу (2.28) підтверджено прямим експериментом.

Диференціююча ланка на рис.2.1 описана вхідною мовою системи САНОС-ПК (розділ 5) при $E(t) = \cos(\omega t)$. Наближене значення похідної визначали згідно з (2.27) шляхом числового інтегрування відповідного диференціального рівняння за методом ФДН (неявний метод другого порядку з автоматичним вибором кроку), причому крок інтегрування обмежували зверху параметром $h_{\max} = 2\pi/\omega$. Абсолютні похибки обчислення похідної визначали за виразом $\Delta = U_R/RC - \omega \sin(\omega t)$ для різних ω та $\tau = RC$.

На рис.2.2 показані результати розрахунків у вигляді залежностей $\lg(\omega/2\pi)$ від $\lg(\tau)$ у випадку сталих значень ε_{\max} . Експериментальні точки лягли на відрізки прямих, що повністю відповідає (2.28).

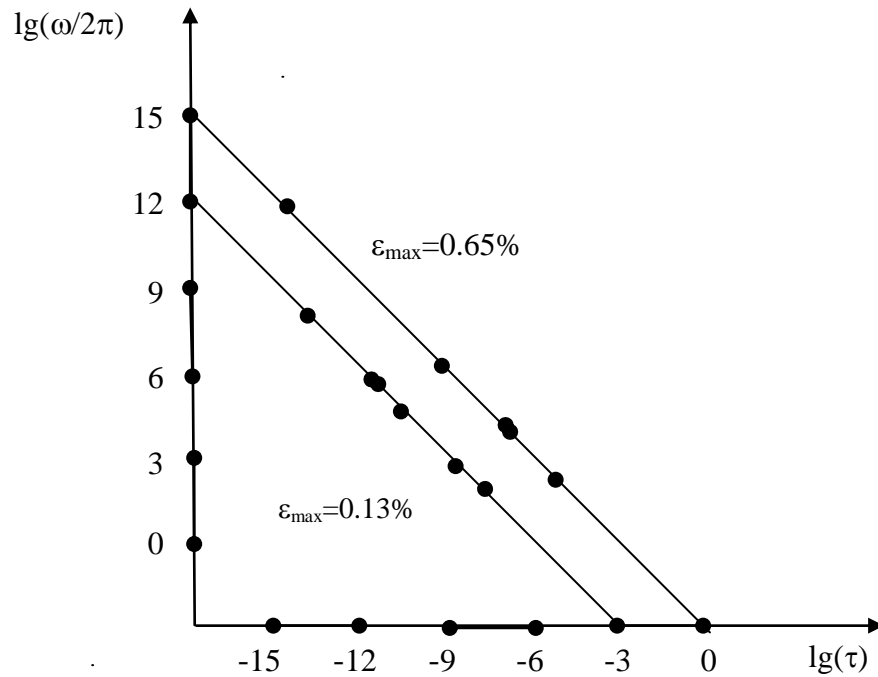


Рис.2.2. Оцінка відносної похибки чисельного диференціювання гармонічного сигналу

Отже, добуток $\omega\tau$ є визначальним параметром для диференціювання гармонічних сигналів, що особливо виявляється у випадку багаторазового послідовного диференціювання. Якщо $\omega\tau$ зavelike, то неприпустимо зростає похибка (2.28), якщо ж замале, то крок інтегрування занадто зменшується і процес диференціювання стає некоректним, що призводить до різкого збільшення похибки.

Виконано числові експерименти щодо визначення областей диференціювання з допустимим значенням похибки у разі зміни $\omega\tau$ та порядку похідної n . Сталі часу послідовних диференціюючих ланок приймали однаковими, а крок інтегрування обмежували зверху параметром h_{\max} .

Зведені результати для методу ФДН другого порядку (програма Gear'a) показані на рис.2.3, а для методу Рунге-Кутта з автоматичним вибором кроку – на рис.2.4.

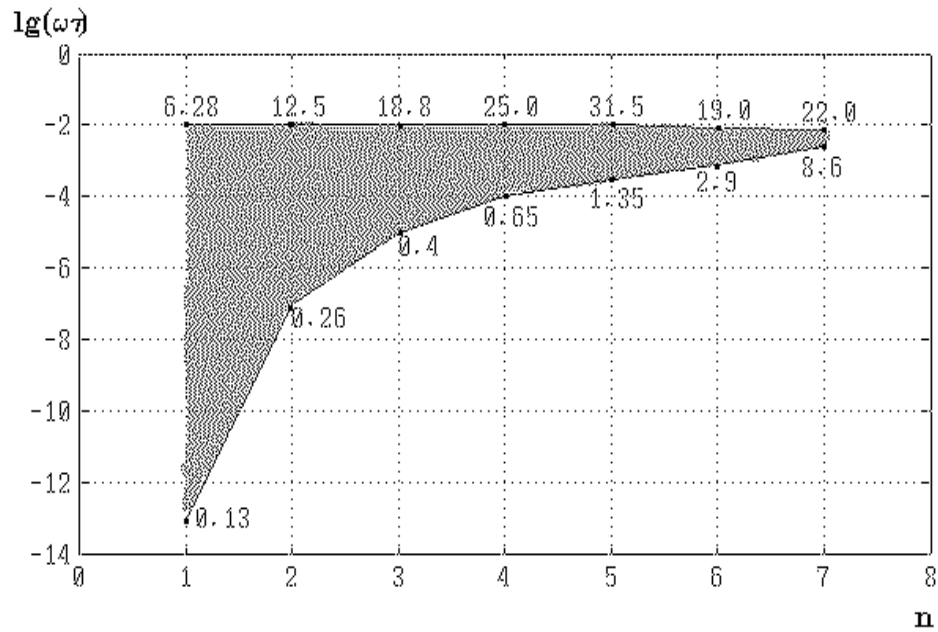


Рис.2.3. Дослідження похибок числового диференціювання гармонічного сигналу для неявного методу другого порядку

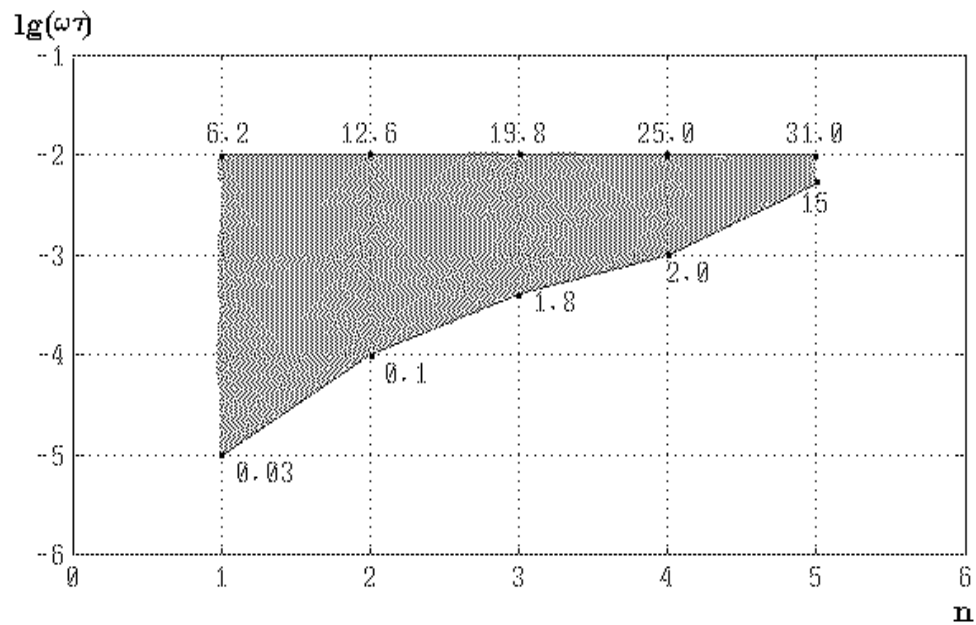


Рис.2.4. Дослідження похибок числового диференціювання гармонічного сигналу для явного методу четвертого порядку

Верхні криві відповідають значенню $h=2\pi \cdot 10^{-2}/\omega$, а нижні – значенню $h=2\pi \cdot 10^{-5}/\omega$. Похибки диференціювання означені числами біля точок. Між нижньою та верхньою кривими у заштрихованій ділянці значення похибки є у межах, визначених нижнім та верхнім числами для відповідного порядку похідної.

Отримані результати у вигляді ділянок стійкого диференціювання на рисунках 2.3 і 2.4 можна використати для вибору сталої часу

диференціюючих ланок та максимального кроку інтегрування у разі обчислення похідних сигналів зі складними спектрами.

Перш за все зазначимо, що оцінка абсолютної похибки похідної полігармонічного сигналу дорівнює сумі оцінок абсолютних похибок гармонік. А для кожної гармоніки оцінка абсолютної похибки повинна задовольняти співвідношення (2.26). Отже, для n -ї похідної полігармонічного сигналу з m гармоніками справджується

$$\Delta_{\max} = \tau^n \sum_{i=1}^m A_i \omega_i^{n+1}, \quad (2.29)$$

де A_i та ω_i – амплітуда й частота i -ї гармоніки.

Вираз (2.29) демонструє, що внесок окремих гармонік у загальну похибку може бути суттєво неоднаковий. Виділимо гармоніку з найбільшим значенням добутку $A_c \omega_c^{n+1}$. Тоді природно очікувати, що знайдеться така частота ω_b , для якої виконуватиметься нерівність $A_i \omega_i^{n+1} \ll A_c \omega_c^{n+1}$ при $\omega_i \leq \omega_b$. Аналогічно може існувати така частота ω_h , що при $\omega_i \geq \omega_h$ виконується $A_i \omega_i^{n+1} \ll A_c \omega_c^{n+1}$. Тобто основний вклад у абсолютну похибку n -ї похідної здійснюють гармоніки сигналу, частоти яких є в інтервалі $[\omega_b, \omega_h]$, що відповідає таким нерівностям:

$$A_b \omega_b^{n+1} \ll A_c \omega_c^{n+1}; \quad A_h \omega_h^{n+1} \ll A_c \omega_c^{n+1}. \quad (2.30)$$

Водночас повинні виконуватись співвідношення, що забезпечують потрапляння зображувальної точки сигналу у заштриховані області на рис.2.3 або 2.4:

$$\lg(\omega_b \tau) \geq \alpha_{nb}; \quad \lg(\omega_h \tau) \leq \alpha_{nh}, \quad (2.31)$$

де α_{nb} та α_{nh} відповідно нижня та верхня межі допустимих значень $\lg(\omega \tau)$ для n -ї похідної за рис.2.3 або 2.4.

Отже, для успішного числового обчислення n -ї похідної за (2.25) з допомогою системи САНОС-ПК необхідно і достатньо, щоб для обраного ω_c існували такі ω_b та ω_h , які одночасно задовольняють нерівності (2.30) та (2.31).

Сформульований критерій дає змогу надійно обчислювати похідні у системі САНОС-ПК до сьомого порядку за допомогою неявних методів інтегрування та до п'ятого порядку – за допомогою явних методів. Потрібно лише ретельно підбирати сталу часу $\tau = RC$, від якої залежить, з одного боку, точність диференціювання, а з іншого – коректність.

2.4. ОРГАНІЗАЦІЯ МАКРОМОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ

Загальна схема макромодельовання зображена на рис.2.5.



Рис.2.5. Загальна схема макромодельовання лінійних систем

Масиви початкових значень можна формувати по-різному. Це можуть бути експериментальні значення. Масиви може задавати замовник макромоделі для деякого уявного об'єкта, що його треба описати у вигляді математичної моделі. Іноді в розпорядженні розробника макромоделі є повний математичний опис об'єкта, за яким потрібно створити спрощену макромодель зниженого порядку та складності.

Порядок макромоделі визначають, головню, з апіорної інформації про модельований об'єкт. Однак є змога оцінювати необхідний порядок макромоделі кореляційним аналізом сигналу із застосуванням результатів Такенса (параграф 4.6, Додаток 7). Різні способи числового диференціювання дискретного сигналу розглянуто у параграфах 2.3, 2.4.3, 4.1.

Задачу (1.4) розв'язують програмою лінійної оптимізації (Додаток 1), у якій формують опис готової макромоделі мовою системи САНОС-ПК.

Порівняльний аналіз макромоделі та об'єкта є значною мірою суб'єктивним, хоча й ґрунтується на значеннях похибок відтворення заданих реакцій.

Описаний процес макромодельовання багаторазово повторюєть з метою досягнення адекватного відтворення заданих сигналів.

Розглянемо організацію макромодельовання з використанням системи аналізу та оптимізації САНОС-ПК на прикладі побудови лінійних макромоделей фільтра та довгої лінії.

2.4.1. Макромодель фільтра п'ятого порядку

Побудуємо макромодель лінійного об'єкта за його реакцією на вхідне збурення. Попередню демонстрацію макромодельовання виконаємо для об'єкта, що є лінійним пасивним фільтром нижніх частот (рис.2.6), а вхідне збурення має вигляд

$$E1(t) = \begin{cases} 10t & \text{при } 0 \leq t < 0.1; \\ 1 & \text{при } t \geq 0.1. \end{cases} \quad (2.32)$$

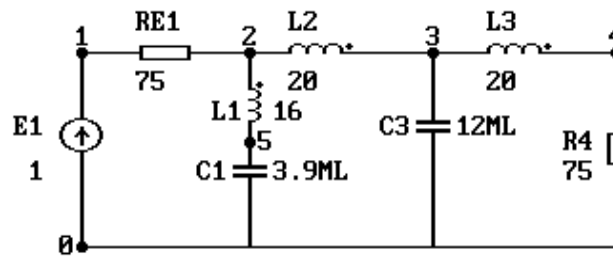


Рис.2.6. Схема фільтра п'ятого порядку.

У табл.2.1 наведено опис фільтра з потрібними даними щодо його моделювання вхідною мовою системи САНОС-ПК (див.розділ 5).

Таблиця 2.1. Завдання підготовки даних для макромодельовання лінійного фільтра

BEGI

Макромодельовання лінійного фільтра

DESC

* Опис фільтра ;

E1(0,1) 1; RE1(1,2) 75; L2(2,3) 20; L3(3,4) 20; R4(4,0) 75; L1(2,5) 16;

C1(5,0) 3.9ml; C2(3,0) 12ml;

* Вихідна напруга та її модуль ;

VR4(R4); MR4=M(VR4);

* Обчислення похідних вхідного сигналу; t=400;

U1`=DU1; U2`=DU2; U3`=DU3; U4`=DU4; U5`=DU5;

DU1=t*(E1-U1);DU2=t*(DU1-U2);DU3=t*(DU2-U3);DU4=t*(DU3-U4);

DU5=t*(DU4-U5);

* Обчислення похідних вихідного сигналу ;

V1`=DV1; V2`=DV2; V3`=DV3; V4`=DV4; V5`=DV5;

DV1=t*(VR4-V1);DU2=t*(DV1-V2);DV3=t*(DV2-V3);DV4=t*(DV3-V4); DV5=t*(DV4-V5);

* Кінець опису системи ;;;

AC

* Частотний аналіз та виведення MR4(F) ;

E1=(1,0); F=A(0.1,1)100; zero;

output: MR4, xmin=0.1,xmax=1,ymin=-1,ymax=1;;

```

DESC * Додатковий опис вхідного сигналу ;
      E1=if(T.lt.0.1)(10*T)(1);;;
TRAN * Аналіз перехідного процесу та запис у FILEAPR.SAN ;
      tmax=10; zero; hs=0.02; output: E1, DU1, DU2, DU3, DU4, DU5, VR4, DV1,
      DV2, DV3, DV4, DV5, tabl, xmin=0, xmax=10, ymin=-0.5, ymax=1;;
END

```

Якщо виконати компіляцію цього тексту в системі САНОС-ПК, то буде проведений спочатку частотний розрахунок із побудовою амплітудно-частотної характеристики (АЧХ) на екрані ПК, а потім розрахунок перехідного процесу із обчисленням п'яти похідних вхідного сигналу $E1$ та п'яти похідних вихідного сигналу VRN . У робочий файл результати аналізу будуть записані в такій послідовності: $u, u^{(1)}, \dots, u^{(5)}, y, y^{(1)}, \dots, y^{(5)}$ (див. речення `output:....;` у розділі TRAN табл. 2.1, а також позначення у (2.14)).

Програма апроксимації, що формує та розв'язує задачу (2.19), міститься в Додатку 1. В основу програми покладена вдала програма середньоквадратичної апроксимації функції багатьох змінних у лінійному просторі заданих базисних функцій [48], де здійснюється ортогоналізація апроксимаційного базису за Грамом-Шмідтом [49]. Програма з [48] перекладена на FORTRAN, доповнена блоками нормування даних, блоками регуляризації відповідно до (2.26) та блоком формування макромоделі мовою САНОС-ПК.

У режимі роботи без регуляризації програма потребує задання лише одного параметра: порядку макромоделі. На підставі даних, що записані в робочому файлі, та заданого порядку макромоделі програма відшукує коефіцієнти апроксимації $a_i, b_i, i=1,2,\dots,5$, будує систему алгебро-диференціальних рівнянь у формі Коші згідно з (2.14) та описує макромодель вхідною мовою системи САНОС-ПК у вигляді рівнянь (2.14) та (2.15) згідно з (2.20). Остаточний текст, повністю готовий для аналізу системою САНОС-ПК, записується в інший робочий файл.

У табл.2.2 наведено вміст такого файлу для макромоделі п'ятого порядку, збудованої за наведеною вище схемою.

Таблиця 2.2. Завдання для порівняння частотних характеристик та перехідних процесів у лінійному об'єкті та макромоделі

```

BEGI
Апроксимаційна макромодель п'ятого порядку
DESC
* Макромодель лінійного об'єкта ;
* Коефіцієнти апроксимації (обчислені програмою апроксимації);
B0=258.178; B1=-3.22718; B2=16.1264; B3=-0.120867; B4=3.02118E-04;
B5=-2.51748E-07;
A0=516.356; A1=439.118; A2=191.998; A3=56.2823; A4=12.2809;
* Макромодель у вигляді (2.14) ;

```

```

*Y`=Y1; *Y1`=Y2; *Y2`=Y3; *Y3`=Y4;
*Y4`=B0*U+B1*U1+B2*U2+B3*U3+B4*U4+B5*U5
-A0*Y-A1*Y1-A2*Y2-A3*Y3-A4*Y4;
* Обчислення похідних вхідного сигналу для макромоделі у вигляді (2.14);
*W`=U1; *U1=400*(U-W); *W1`=U2; *U2=400*(U1-W1); *W2`=U3;
*U3=400*(U2-W2); *W3`=U4; *U4=400*(U3-W3); *W4`=U5; *U5=400*(U4-
W4);
* Макромодель у вигляді (2.15) ;
X1`=(-A4*X1+X2+(B4-A4*B5)*U);
X2`=(-A3*X1+X3+(B3-A3*B5)*U);
X3`=(-A2*X1+X4+(B2-A2*B5)*U);
X4`=(-A1*X1+X5+(B1-A1*B5)*U);
X5`=(-A0*X1+(B0-A0*B5)*U);
Y=X1+B5*U; EY(0,100)Y; VY(EY);
* Лінійний об'єкт для порівняння;
E1(0,1); RE1(1,2) 75; L2(2,3) 20; L3(3,4) 20; R4(4,0) 75; L1(2,8) 16;
C1(8,0) 3.9ml; C2(3,0) 12ml; VRE1(RE1); VR4(R4);;
E1=0; MVR4=M(VR4); MY=M(VY); U=E1;;;
AC * Частотний аналіз та порівняння ;
E1=(1,0); F=A(0.1,1)100; zero;
output: MVR4,MY, xmin=0.1,xmax=1,ymin=0,ymax=1;;
DESC * Додатковий опис вхідного сигналу ;
E1=if(T.lt.0.1)(10*T)(1);;
TRAN * Аналіз перехідного процесу та порівняння ;
tmax=10; zero;
output: VR4,Y, xmax=10,ymin=-0.5,ymax=1;;
END

```

Компіляція тексту в табл.2.2 системою САНОС-ПК дає змогу порівняти частотні характеристики та перехідні процеси в об'єкті та в макромоделі. На рис.2.7 показані графіки частотних характеристик, а на рис.2.8 – перехідних процесів у модельованому фільтрі та в макромоделі. Відносна похибка відтворення перехідних процесів не перевищує 2%.

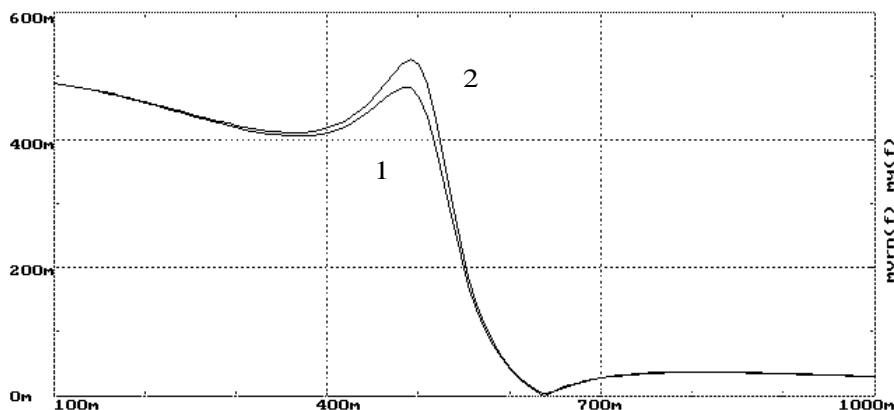


Рис.2.7. Частотні характеристики лінійного об'єкту (1) та макромоделі (2)

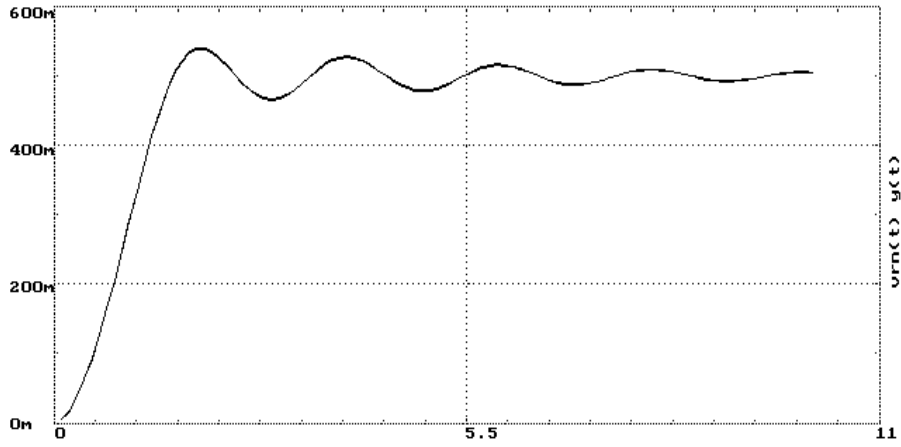


Рис.2.8. Перехідні процеси в лінійному об'єкті та макромоделі.

2.4.2. Макромоделі довгої лінії за середньоквадратичним та мінімаксним критеріями

Набагато складнішою є задача макромоделювання довгої лінії за реакцією на імпульс, близький до одиничного. Річ у тім, що для успішного моделювання довгої лінії звичайними диференціальними рівняннями з прийнятною похибкою відтворення реакції необхідно будувати макро-модель високого порядку. Послідовне обчислення похідних за (2.25) дає змогу отримати макромоделі до сьомого порядку, що може бути недостатньо. Для лінійних систем є спосіб, що дає змогу збільшити порядок макромоделі принаймні вдвічі.

Проінтегруємо рівняння (2.13) із змінною верхньою межею $(n-r)$ раз за нульових початкових умов та $m=n$. Отримаємо еквівалентне рівняння

$$\begin{aligned} \sum_{i=1}^r a_{n-r+i} \frac{d^i y(t)}{dt^i} + \sum_{i=1}^{n-r} a_{n-r-i} \int_0^t (i) \int y(\tau) d\tau^i = \\ = \sum_{i=0}^r b_{n-r+i} \frac{d^i u(t)}{dt^i} + \sum_{i=1}^{n-r} b_{n-r-i} \int_0^t (i) \int u(\tau) d\tau \end{aligned} \quad (2.33)$$

Для рівняння (2.33) задача ідентифікації, аналогічна до (2.19), має вигляд

$$\begin{aligned} \min_{\bar{a}, \bar{b}} \sum_{k=1}^M (y^{(r)}(t_k) - \sum_{i=0}^r b_{n-r+i} u^{(i)}(t_k) - \sum_{i=1}^{n-r} b_{n-r-i} \int_0^{t_k} (i) \int u(\tau) d\tau^i + \\ + \sum_{i=0}^{r-1} a_{n-r+i} y^{(i)}(t_k) + \sum_{i=1}^{n-r} a_{n-r-i} \int_0^{t_k} (i) \int y(\tau) d\tau^i)^2; \quad M \geq 2n+1. \end{aligned} \quad (2.34)$$

Для розв'язку задачі (2.34) треба обчислити r похідних та $(n-r)$ інтегралів сигналів $u(t)$ і $y(t)$. Похибка послідовного інтегрування зростає

аналогічно до похибки послідовного диференціювання. Тому максимальний порядок макромоделі з заданою похибкою для (2.34) вдвічі більший, ніж для (2.19).

У табл.2.3 наведено текст завдання для системи САНОС-ПК з описом заступної схеми довгої лінії і п'ятнадцяти диференціальних та інтегровальних ланок вхідного й вихідного сигналів.

Таблиця 2.3. Завдання підготовки даних
для макромодельовання довгої лінії

BEGI

Модель довгої лінії з обчисленням похідних та інтегралів

DESC

* Вхідний сигнал ;

E0(0,1)U; U='IMP';; table:'IMP'0.0/0.0,0.001/3,1/3;;

* Заступна схема довгої лінії як об'єкт для макромодельовання ;

L1(1,2)KL; L2(2,3)KL; L3(3,4)KL; L4(4,5)KL; L5(5,6)KL;

L6(6,7)KL; L7(7,8)KL; L8(8,9)KL; L9(9,10)KL; L10(10,11)KL;

C1(0,2)KC; C2(0,3)KC; C3(0,4)KC; C4(0,5)KC; C5(0,6)KC;

C6(0,7)KC; C7(0,8)KC; C8(0,9)KC; C9(0,10)KC; C10(0,11)KC;

R0(11,0) 642.9; V0(R0); U0=V0; KL=0.06; KC=0.9234E-6;

* Коефіцієнт нормування ; KN=1E3;

* Інтегрування вхідного сигналу ;

J53(53,0)U ; C53(0,53)1; V53(C53); R53(53,0)10MG; UI1=V53*KN;

J54(54,0)UI1; C54(0,54)1; V54(C54); R54(54,0)10MG; UI2=V54*KN;

J55(55,0)UI2; C55(0,55)1; V55(C55); R55(55,0)10MG; UI3=V55*KN;

J56(56,0)UI3; C56(0,56)1; V56(C56); R56(56,0)10MG; UI4=V56*KN;

J59(59,0)UI4; C59(0,59)1; V59(C59); R59(59,0)10MG; UI5=V59*KN;

52

* Інтегрування вихідного сигналу ;

J51(51,0)U0 ; C51(0,51)1; V51(C51); R51(51,0)10MG; UI1=V51*KN;

J52(52,0)UI1; C52(0,52)1; V52(C52); R52(52,0)10MG; UI2=V52*KN;

J57(57,0)UI2; C57(0,57)1; V57(C57); R57(57,0)10MG; UI3=V57*KN;

J58(58,0)UI3; C58(0,58)1; V58(C58); R58(58,0)10MG; UI4=V58*KN;

J60(60,0)UI4; C60(0,60)1; V60(C60); R60(60,0)10MG; UI5=V60*KN;

* Диференціювання вихідного сигналу ;

E11(0,21)U0; R11(21,22)1; C11(22,0)1MC; I11(R11); U1=I11/C11/KN;

E12(0,23)U1; R12(23,24)1; C12(24,0)1MC; I12(R12); U2=I12/C12/KN;

E13(0,25)U2; R13(25,26)1; C13(26,0)1MC; I13(R13); U3=I13/C13/KN;

E14(0,27)U3; R14(27,28)1; C14(28,0)1MC; I14(R14); U4=I14/C14/KN;

E15(0,29)U4; R15(29,30)1; C15(30,0)1MC; I15(R15); U5=I15/C15/KN;;;

TRAN * Розрахунок перехідного процесу та записування результатів ;

TMAX=0.03; HS=1E-4; ERMAX=0.005; zero; HMIN=1.0E-10;

output: UI5, UI4, UI3, UI2, UI1, U, tabl, xmax=0.03,ymin=-10,ymax=10;

output: U5I, U4I, U3I, U2I, U1I, U0, U1, U2, U3, U4, U5, tabl,

xmax=0.03, ymin=-10, ymax=10;;

END

За цим завданням формують масиви даних, що містять близько 300 синхронних часових відліків 17-ти змінних. Програма апроксимації, аналогічна до наведеної у Додатку 1, формує та розв'язує задачу (2.34) при $n=10$, $r=5$, а також готує завдання мовою САНОС-ПК для порівняльного аналізу перехідних процесів у заступній схемі довгої лінії та макромоделі. Текст такого завдання є у табл.2.4.

Таблиця 2.4. Завдання для порівняльного аналізу перехідних процесів у довгій лінії та її макромоделі 10-го порядку

BEGI

Макромодель довгої лінії за критерієм найменших квадратів

DESC

* Довга лінія ;

E0(0,1)U; U='IMP';; table:'IMP'0.0/0.0,0.001/3,1/3;; u=U;

L1(1,2) KL; L2(2,3) KL; L3(3,4) KL; L4(4,5) KL; L5(5,6)KL;

L6(6,7) KL; L7(7,8) KL; L8(8,9) KL; KL=0.06;

C1(0,2) KC; C2(0,3) KC; C3(0,4) KC; C4(0,5) KC; C5(0,6)KC;

C6(0,7) KC; C7(0,8) KC; C8(0,9) KC; KC=0.9234E-6;

R0(9,0) 642.9; V0(R0); U0=V0;

* Макромодель ; * кількість точок = 283;

A0=0.0;A1=9.86237;A2=479.3668;A3=2765.375; A4=-184.2789; A5=-312.1642;

53

A6=63.09724; A7=-9.870094; A8=-486.6061; A9=-3143.882; A10=-2836.485;

A11=-5293.941; A12=-944.0645; A13=-1207.467; A14=-62.29639;

A15=-71.68034; A16=-0.8290017;

y`=y1*KN; y1`=y2*KN; y2`=y3*KN; y3`=y4*KN; y4`=y5*KN;

y5`=y6*KN; y6`=y7*KN; y7`=y8*KN; y8`=y9*KN; y9`=y10*KN;

w1`=uu1; uu1=1e6*(u-w1); u1=uu1/KN; KN=1E3;

w2`=uu2; uu2=1e6*(u1-w2); u2=uu2/KN;

w3`=uu3; uu3=1e6*(u2-w3); u3=uu3/KN;

w4`=uu4; uu4=1e6*(u3-w4); u4=uu4/KN;

w5`=uu5; uu5=1e6*(u4-w5); u5=uu5/KN;

y10=A0+A1*u+A2*u1+A3*u2+A4*u3+A5*u4+A6*u5+A7*y+A8*y1+A9*y2+A10*y3+A11*y4+A12*y5+A13*y6+A14*y7+A15*y8+A16*y9;;;

TRAN

TMAX=0.04; HS=5.0E-5; ZERO; HMIN=1E-9;

output: U, U0, y, xmax=0.04, ymin=0, ymax=5;;

END

На рис.2.9 показано перехідні процеси у довгій лінії та її макромоделі, коефіцієнти якої обчислені за методом найменших квадратів. Максимальна відносна похибка відтворення реакції довгої лінії становить 9%.

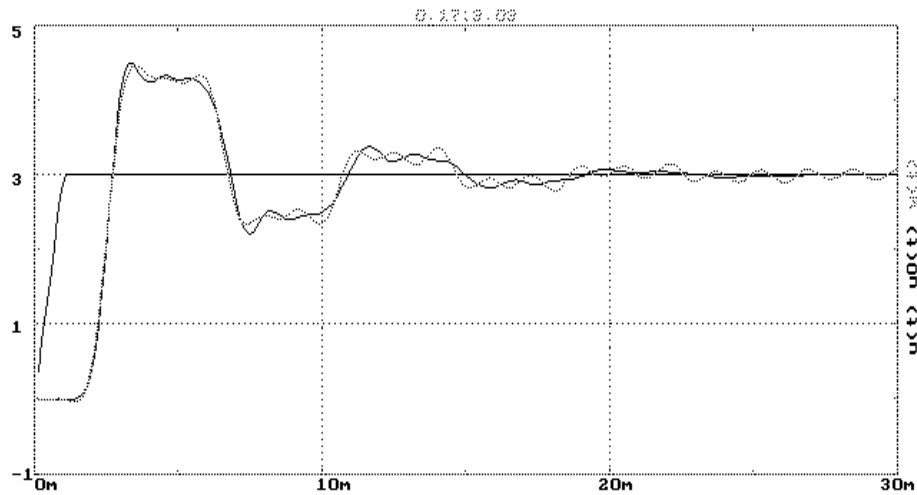


Рис.2.9. Перехідні процеси в довгій лінії та макромоделі.
Апроксимація методом найменших квадратів

Макромодель довгої лінії була збудована також за критерієм Чебишова як розв'язок задачі (2.35), яка є конкретним випадком загальної задачі (1.6):

$$\min_{\bar{a}, \bar{b}} \max_{k=1, \dots, M} \left| y^{(r)}(t_k) - \sum_{i=0}^r b_{n-r+i} u^{(i)}(t_k) - \sum_{i=1}^{n-r} b_{n-r-i} \int_0^{t_k} \binom{i}{\tau} u(\tau) d\tau^i + \right. \\ \left. + \sum_{i=0}^{r-1} a_{n-r+i} y^{(i)}(t_k) + \sum_{i=1}^{n-r} a_{n-r-i} \int_0^{t_k} \binom{i}{\tau} y(\tau) d\tau^i \right|; \quad M \geq 2n+1; \quad (2.35)$$

Програма, яка формує дані для задачі (2.35) та розв'язує її симплекс-методом, наведена в Додатку 2. В основі програми є стандартна програма мінімаксної апроксимації симплекс-методом із [50].

У табл.2.5 записане завдання для системи САНОС-ПК, яке містить макромодель довгої лінії, збудовану за (2.35) для тих самих даних, що й макромодель, описана у табл.2.4.

Таблиця 2.5. Макромодель довгої лінії за критерієм Чебишова

BEGI

Макромодель довгої лінії за мінімаксним критерієм.

DESC

* Довга лінія ;

E0(0,1)U; U='IMP';; table:'IMP'0.0/0.0,0.001/3,1/3;; u=U;

L1(1,2) KL; L2(2,3) KL; L3(3,4) KL; L4(4,5) KL; L5(5,6)KL;

L6(6,7) KL; L7(7,8) KL; L8(8,9) KL; KL=0.06;

C1(0,2) KC; C2(0,3) KC; C3(0,4) KC; C4(0,5) KC; C5(0,6)KC;

C6(0,7) KC; C7(0,8) KC; C8(0,9) KC; KC=0.9234E-6;

R0(9,0) 642.9; V0(R0); U0=V0;

* Макромодель , кількість точок = 283;

A0=0.0; A1=1.606421; A2=58.11726; A3=2746.657; A4=-140.6944;

```

A5=-308.8003; A6=58.98512; A7=-1.613959; A8=-59.19547; A9=-2794.165;
A10=-2022.4; A11=-5003.937; A12=-749.7761; A13=-1186.148; A14=-50.62755;
A15=-71.08282; A16=-0.6655368;
y`=y1*KN; y1`=y2*KN; y2`=y3*KN; y3`=y4*KN; y4`=y5*KN;
y5`=y6*KN; y6`=y7*KN; y7`=y8*KN; y8`=y9*KN; y9`=y10*KN;
w1`=uu1; uu1=1e6*(u-w1); u1=uu1/KN; KN=1E3;
w2`=uu2; uu2=1e6*(u1-w2); u2=uu2/KN; w3`=uu3; uu3=1e6*(u2-w3); u3=uu3/KN;
w4`=uu4; uu4=1e6*(u3-w4); u4=uu4/KN; w5`=uu5; uu5=1e6*(u4-w5);
u5=uu5/KN;
y10=A0+A1*u+A2*u1+A3*u2+A4*u3+A5*u4+A6*u5+A7*y+A8*y1+A9*y2+
A10*y3+A11*y4+A12*y5+A13*y6+A14*y7+A15*y8+A16*y9;;;
TRAN
TMAX=0.04; HS=1.2E-4; *ERMAX=0.005; zero; HMIN=1E-9;
output: U, U0, y, tabl, xmax=0.04, ymin=0, ymax=5;;
END

```

Результати порівняльного розрахунку для такої макромоделі показані на рис.2.10. Максимальна відносна похибка у цьому випадку становить 10%.

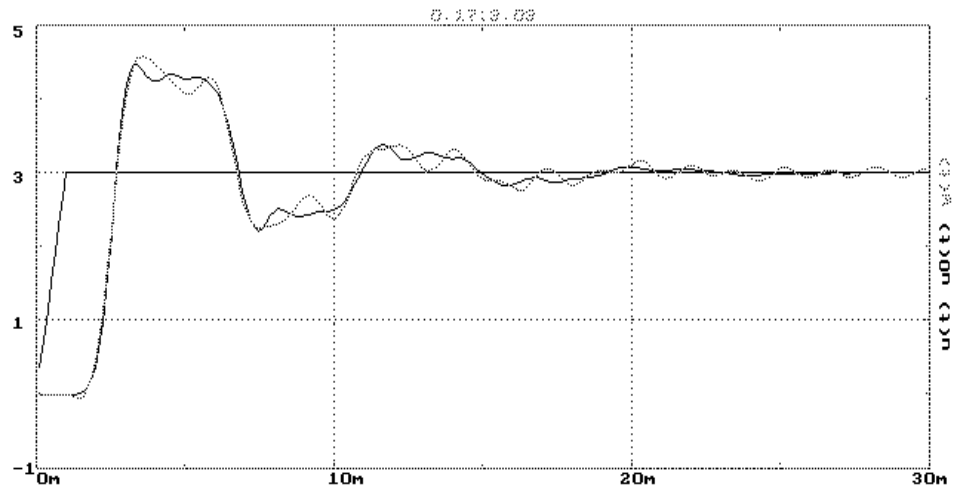


Рис.2.10. Перехідні процеси в довгій лінії та макромоделі.
Апроксимація за мінімаксним критерієм

Зазначимо, що точності відтворення реакції для макромоделей за обома критеріями практично збігаються. Натомість машинний час, потрібний для апроксимації за мінімаксним критерієм, для лінійних об'єктів більше ніж у тричі перевищує час апроксимації за середньо-квадратичним критерієм.

У параграфі 4.6 описано макромоделі нелінійного об'єкта, отримані за обома названими критеріями.

2.4.3. Макромодельовання за частотними характеристиками

Добре відома ідентифікація лінійних стаціонарних систем із зосередженими параметрами у частотній області, коли для вимірюваних частотних характеристик об'єкта $A(\omega)$ та $\varphi(\omega)$, де $\omega \in \Omega$, апроксимацію шукаємо у вигляді дробово-раціональної функції комплексної змінної. У випадку застосування середньоквадратичного критерію задача ідентифікації має вигляд

$$\min_{\bar{a}, \bar{b}} \int_{\Omega} \left(\frac{\sum_0^m b_i(j\omega)^i}{\sum_0^n a_i(j\omega)^i} - A(\omega) \exp(j\varphi(\omega)) \right)^2 d\omega. \quad (2.36)$$

Аналог задачі (2.36) в дискретному варіанті є таким

$$\min_{\bar{a}, \bar{b}} \sum_{k=1}^M \left(\frac{\sum_0^m b_i(j\omega_k)^i}{\sum_0^n a_i(j\omega_k)^i} - A(\omega_k) \exp(j\varphi(\omega_k)) \right)^2 \quad (2.37)$$

Задача (2.37) загалом є лінійною відносно шуканих параметрів апроксимації – векторів \bar{a} та \bar{b} . При $a_n=1$ та $A(\omega_k) \exp(j\varphi(\omega_k))=W(j\omega_k)$ вона зводиться до розв'язку за методом найменших квадратів комплексної прямокутної системи

$$\begin{pmatrix} -(j\omega_1)^{n-1}W(j\omega_1) & \cdots & -W(j\omega_1) & (j\omega_1)^m & \cdots & 1 \\ -(j\omega_2)^{n-1}W(j\omega_2) & \cdots & -W(j\omega_2) & (j\omega_2)^m & \cdots & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -(j\omega_M)^{n-1}W(j\omega_M) & \cdots & -W(j\omega_M) & (j\omega_M)^m & \cdots & 1 \end{pmatrix} \begin{pmatrix} a_{n-1} \\ \vdots \\ a_0 \\ b_m \\ \vdots \\ b_0 \end{pmatrix} = \begin{pmatrix} (j\omega_1)^n W(j\omega_1) \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ (j\omega_M)^n W(j\omega_M) \end{pmatrix}$$

Неважко встановити взаємозв'язок між задачами (2.37) та (2.19), що є задачами ідентифікації однакових лінійних систем, але поставлених в частотній та часовій областях.

Пропонуємо таку послідовність дій.

Нехай числово задана в деякій частотній області Ω амплітудно-фазо-частотна характеристика (АФЧХ) об'єкта $W(j\omega)=A(\omega) \cdot \exp(j\varphi(\omega))$; $\omega \in \Omega$. Як вхідний сигнал виберемо деякий сигнал $u(t)$, для якого існує Фур'є-зображення і він визначений у часовій області Θ , що відповідає частотній області Ω :

$$U(j\omega) = \int_{\Theta} u(t) \exp(-j\omega t) dt; \quad \omega \in \Omega. \quad (2.38)$$

Фур'є-зображення реакції системи в області Ω визначене виразом $Y(j\omega) = W(j\omega)U(j\omega)$, а часову реакцію системи знаходять зворотним перетворенням Фур'є

$$y(t) = \frac{1}{2\pi} \int_{\Omega} W(j\omega)U(j\omega) \exp(j\omega t) d\omega; \quad t \in \Theta. \quad (2.39)$$

Тепер ідентифікація лінійної моделі за знайденими функціями $u(t)$, $y(t)$, зводиться у дискретному варіанті до задачі (2.19).

Якщо задана не вся АФЧХ $W(j\omega)$, а лише АЧХ $A(\omega)$ або лише ФЧХ $\varphi(\omega)$, то, застосувавши гіпотезу мінімальнофазової системи, за допомогою перетворень Гільберта можемо відтворити частину АФЧХ, якої не вистачає:

$$\varphi(\omega) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\ln A(\xi)}{\xi - \omega} d\xi; \quad A(\omega) = \exp\left(-\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\varphi(\xi)}{\xi - \omega} d\xi\right). \quad (2.40)$$

Встановлений зв'язок між задачею (2.37) та задачею (2.19) зручно перевірити в системі MATLAB [51].

На рис.2.11 показана блок-схема відповідної програми у випадку, коли задана лише АЧХ об'єкта.

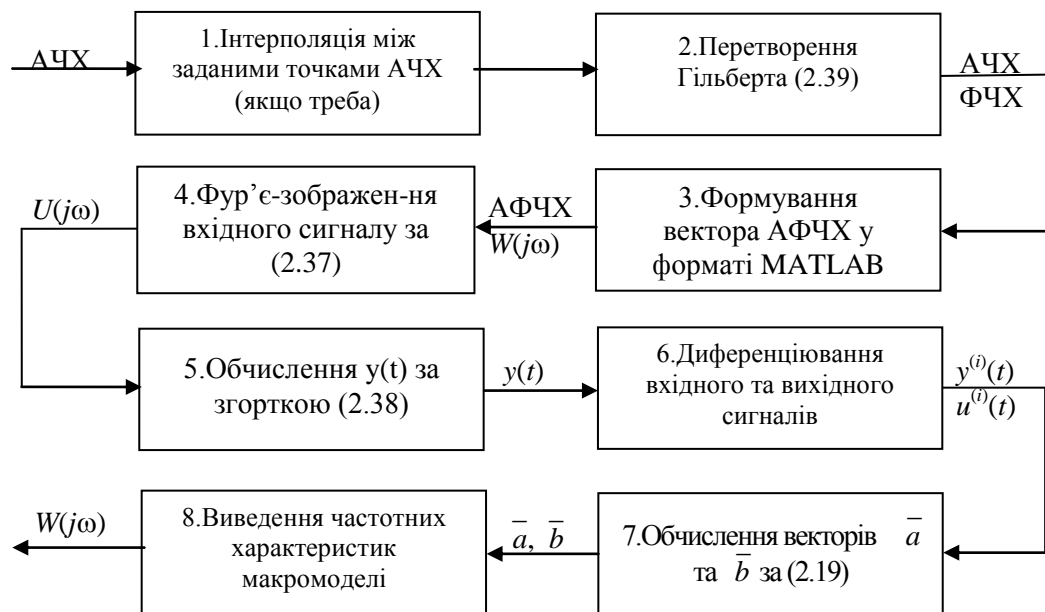


Рис.2.11. Блок-схема програми макромодельовання лінійного об'єкта за частотною характеристикою

У Додатку 3 наведено текст програми, яка реалізує описаний метод у вигляді зовнішньої функції MASMOD мовою системи MATLAB-3.5.

Всі обчислення в програмі виконуються для дискретних функцій за допомогою дискретних аналогів формул (2.38)-(2.40).

У блоці 1 задані точки АЧХ доповнюють до найближчого зверху степеня 2 за допомогою інтерполяції степеневими поліномами степеня $n-1$, де n – порядок макромоделі.

Блок 2 обчислює $\varphi(\omega)$ дискретним аналогом формули (2.40) за допомогою зовнішньої функції HILBERT.

У блоці 4 швидким дискретним перетворенням Фур'є (функція FFT) визначають зображення $U(j\omega)$ для обраного вхідного сигналу $u(t)$.

Блок 5 використовує зворотне швидке перетворення Фур'є (функція IFFT) для обчислення реакції $y(t)$.

Диференціювання вхідного та вихідного сигналів виконано в блоці 6 методом ковзного інтерполювального полінома [46, с.253] степеня $n+2$ з використанням функцій поліноміальної інтерполяції POLIFIT, POLIDER, POLIVAL. Якщо потрібна регуляризація процедури диференціювання, то застосовують апроксимуючий степеневий поліном степеня $n+2$, який будують за кількістю точок, що перевищує $n+3$ [46].

Вектори коефіцієнтів \bar{a} та \bar{b} АФЧХ макромоделі $W_m(j\omega)$ обчислюють за методом найменших квадратів у блоці 7.

Постає питання про зв'язок між похибками задач (2.37) і (2.19) та вплив цих похибок на результат ідентифікації.

Без порушення загальності для спрощення викладу можемо вважати, що реакція системи $y(t)$ є імпульсною функцією системи $g(t)$, пов'язаною з передатною функцією $W(\lambda)$ перетворенням Лапласа:

$$\int_0^{\infty} g(t) \exp(-\lambda t) dt = W(\lambda) .$$

Нехай похибка $\Delta(t)$ імпульсної функції $g(t)$ є адитивною функцією часу. Завдяки лінійності перетворення Лапласа

$$\int_0^{\infty} (\Delta(t) + g(t)) \exp(-\lambda t) dt = \Delta(\lambda) + W(\lambda); \quad \text{де} \quad \int_0^{\infty} \Delta(t) \exp(-\lambda t) dt = \Delta(\lambda) .$$

Отже, адитивна залежна від часу похибка $\Delta(t)$ та адитивна залежна від частоти похибка $\Delta(\lambda)$ практично однаковим чином входять в ідентифікаційні задачі, поставлені в частотній області Ω

$$\min_{\bar{a}, \bar{b}} \int_{\Omega} \left(\frac{\sum_0^m b_i(j\omega)^i}{\sum_0^n a_i(j\omega)^i} - A(\omega) \exp(j\varphi(\omega)) - \Delta(j\omega) \right)^2 d\omega$$

і в часовій області Θ

$$\min_{\bar{a}, \bar{b}} \int_{\Theta} \left(\sum_{i=0}^m b_i \delta^{(i)}(t) - \sum_{i=0}^{n-1} a_i (g(t) + \Delta(t))^{(i)} - (g(t) + \Delta(t))^{(n)} \right)^2 dt.$$

Похибки $\Delta(\lambda)$ та $\Delta(t)$ пов'язані перетворенням Лапласа.

Звичайно, системи, отримані за двома процедурами ідентифікації, будуть різні. Однак похибки обох підходів, якщо не враховувати похибки розв'язування задач (2.19) і (2.37), є сумірними.

Покажемо застосування програми MACMOD для тестової лінійної системи з передавальною функцією

$$W(\lambda) = \frac{-0.252 \cdot 10^{-6} \lambda^5 + 0.302 \cdot 10^{-3} \lambda^4 - 0.121 \lambda^3 + 16.13 \lambda^2 - 3.227 \lambda + 258.2}{\lambda^5 + 12.28 \lambda^4 + 56.28 \lambda^3 + 192.0 \lambda^2 + 439.1 \lambda + 516.4}$$

Для запуску ідентифікації за описаною програмою MACMOD потрібно задати АЧХ та ФЧХ модельованої системи. Графіки АЧХ та ФЧХ, обчислені за передатною функцією $W(\lambda)$, показані на рис.2.12 і 2.13.

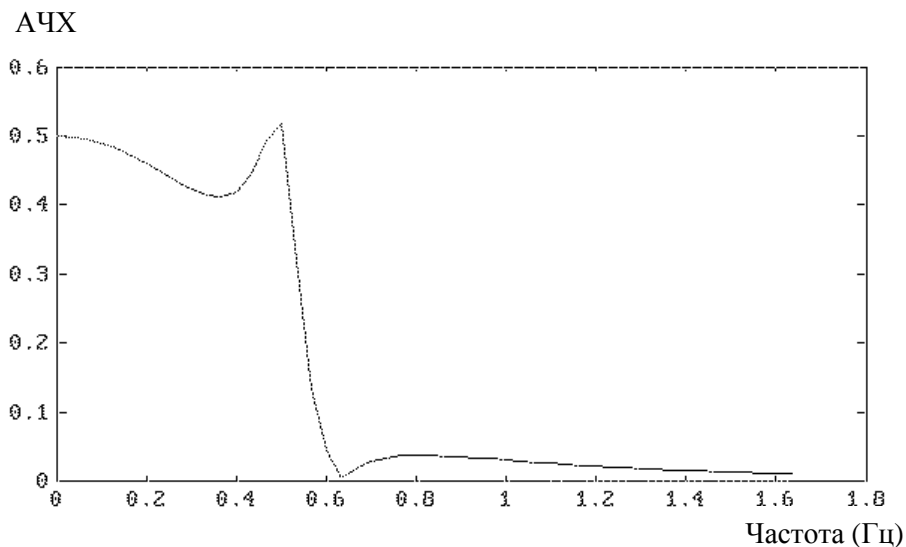


Рис.2.12. Задана ампліудно-частотна характеристика.

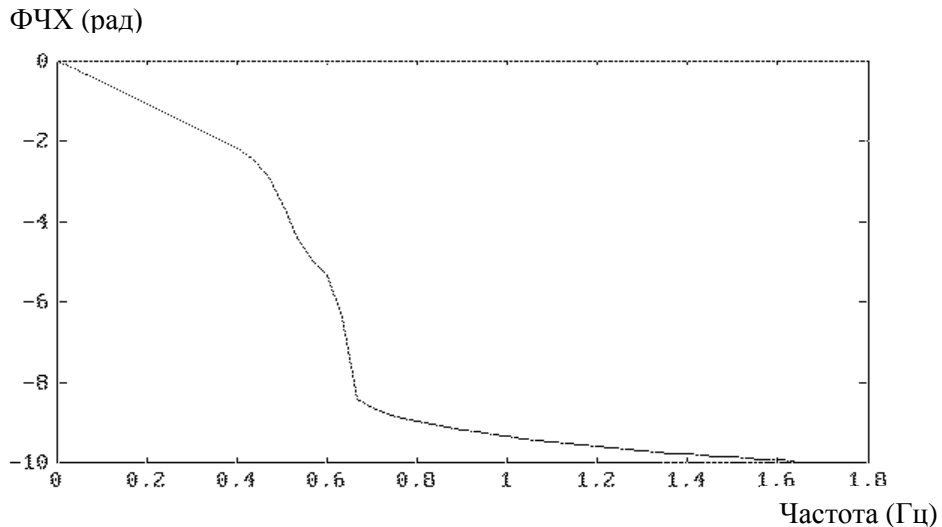


Рис.2.13. Задана фазо-частотна характеристика.

Текст запускаючої програми мовою MATLAB для функції MACMOD наведено в табл.2.6.

Таблица 2.6. Запускающая программа для функции макромоделювання лінійної системи MACMOD

```
% ***** Transfer function of object *****
nu=[ -2.52e-7  3.02e-4  -1.21e-1  16.13  -3.227  258.2 ];
de=[ 1  12.28  56.28  192.0  439.1  516.4 ];
m=300;           % Number of points.
wm=10;           % Maximal frequency in Hz.
N=5;             % Order of system.
dw=2*pi*wm/m;    % Frequency step.
w=[0:dw:dw*m];   % Frequency array.
[mag,phase]=bode(nu,de,w); phase=phase*(pi/180);
plot(w(1:50)/(2*pi),mag(1:50),'g'); pause;
plot(w(1:50)/(2*pi),phase(1:50),'g'); pause;
[num,den]=macmod(N,mag,phase,w), nu, de, pause;
% ***** Grafic comparison of macromodel and object *****
t=[0:1/wm:2*pi/dw]; tau=2*pi/(dw*100); u=1-exp(-t/tau);
plot(t(1:m),lsim(nu, de, u(1:m),t(1:m)),'w'); hold on;
plot(t(1:m),lsim(num,den,u(1:m),t(1:m)),'g'); hold off;
w=w/5; [acho,fcho]=bode(nu,de,w); [achm,fchm]=bode(num,den,w);
plot(w/(2*pi),acho(1:m+1),'w',w/(2*pi),achm(1:m+1),'g'); pause;
plot(w/(2*pi),fcho(1:m+1),'w',w/(2*pi),fchm(1:m+1),'g');
pause; exit;
```

Триста відліків частотних характеристик з рівномірним кроком в інтервалі $[0,10]$ Гц утворюють початкові масиви для ідентифікації.

Функція МАСМОД (табл.2.6) за цими масивами обчислює коефіцієнти передатної функції макромоделі:

$$W_m(p) = \frac{-0.7 \cdot 10^{-3} p^5 + 0.27 \cdot 10^{-1} p^4 - 0.36 p^3 + 17.08 p^2 - 4.45 p + 263.5}{p^5 + 12.23 p^4 + 56.92 p^3 + 192.5 p^2 + 445.01 p + 527.02}$$

Максимальні відносні похибки відтворення АЧХ та ФЧХ, а також перехідної характеристики на рис 2.14, не перевищують 3%.

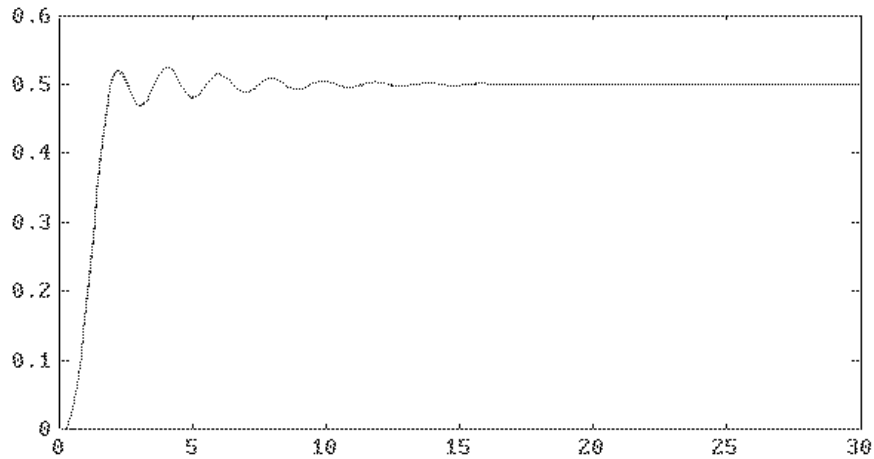


Рис.2.14. Перехідна характеристика модельованої системи.

РОЗДІЛ 3

СТРУКТУРИ МАКРОМОДЕЛЕЙ ТА МЕТОДИ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ СИСТЕМ ІЗ ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ

3.1. СТРУКТУРИ МАКРОМОДЕЛЕЙ НЕЛІНІЙНИХ СИСТЕМ

Макромоделі нелінійних динамічних систем за заданими множинами вхідних і вихідних сигналів будують для різноманітних структур моделюючих операторів. Однак зазвичай нема обґрунтування обраних структур. Спробуємо обґрунтувати низку структур моделюючих операторів за відомим описом у системі змінних стану.

Спочатку визначимо поняття структури моделюючого оператора. Першим кроком у побудові макромоделі є вибір загального вигляду оператора. Під час ідентифікації уточнюють його вигляд і параметри складових частин (наприклад, вигляд ядра оператора, вигляд нелінійної функції). Однак якщо процедури ідентифікації обґрунтовують вигляд складових частин оператора, то його загальну структуру вибирають, як звичайно, без будь-яких обґрунтувань.

Розпочинаючи обґрунтування загальної структури макромоделі нелінійної системи із зосередженими параметрами, скористаємось такими положеннями. По-перше, структура повинна містити лінійну підсистему, для якої існують надійні процедури ідентифікації (розділ 2), завдяки чому можна помітно спростити побудову макромоделі. По-друге, структура повинна допускати побудову ідентифікаційних методів, тобто бути ідентифікованою. Нарешті, структура повинна бути досить загальною – поширюватись на значний клас нелінійних систем.

3.1.1. Структура макромоделі нелінійної системи з нестационарною лінійною підсистемою

Доведемо теорему [23].

Теорема 1. Нехай нелінійна система описана системою звичайних алгебро-диференціальних рівнянь

$$\begin{aligned}\dot{\bar{x}} &= \bar{f}(\bar{x}, \bar{u}, t), \\ \bar{y} &= \bar{\varphi}(\bar{x}, \bar{u}, t),\end{aligned}\tag{3.1}$$

де $\bar{u}(t)$, $\bar{x}(t)$, $\bar{y}(t)$ – дійсні вектор-функції відповідно вхідних сигналів (збурень), стану та вихідних сигналів (реакцій), а дійсна вектор-функція $\bar{\varphi}(\bar{x}, \bar{u}, t)$ є диференційовною за t при $t \in [t_0, T]$.

Такій системі еквівалентна за входом-виходом система, що складається з лінійної нестационарної динамічної та нелінійної нестационарної

безінерційної підсистем відповідно до рис.3.1, де $\bar{\chi}(\bar{v}, \bar{y}, \bar{u}, \dot{\bar{u}}, t)$ – дійсна нелінійна вектор-функція, а \bar{v} – дійсний внутрішній вектор.

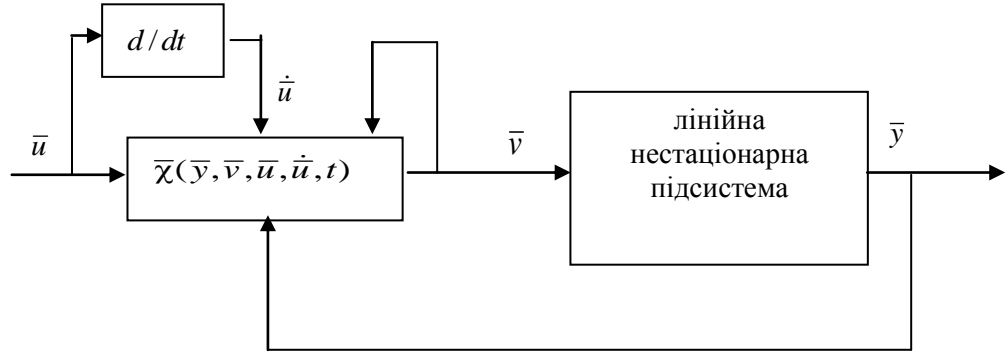


Рис.3.1. Структура макромоделі нелінійної системи до теореми 1

Доведення. Опишемо систему (3.1) у розширеному координатному базисі $\bar{z}(t) = [\bar{x}(t), \bar{y}(t)]$

$$\begin{aligned}\dot{\bar{z}} &= \bar{f}_p(\bar{z}, \bar{u}, \dot{\bar{u}}, t), \\ \bar{y} &= C\bar{z},\end{aligned}\tag{3.3}$$

де $\bar{f}_p(\bar{z}, \bar{u}, \dot{\bar{u}}, t) = [\bar{f}(\bar{x}, \bar{u}, t), \frac{d\bar{\phi}}{dt}(\bar{x}, \bar{u}, \dot{\bar{u}}, t)]$; C – діагональна матриця з нульовими та одиничними елементами.

Систему (3.3) завжди можна зобразити лінійною нестаціонарною підсистемою з матрицею рівнянь стану $A(t)$ і нелінійною підсистемою $\bar{\psi}(\bar{z}, \bar{u}, \dot{\bar{u}}, t)$

$$\begin{aligned}\dot{\bar{z}} &= A(t)\bar{z} + \bar{\psi}(\bar{z}, \bar{u}, \dot{\bar{u}}, t), \\ \bar{y} &= C\bar{z},\end{aligned}\tag{3.4}$$

де $\bar{\psi}(\bar{z}, \bar{u}, \dot{\bar{u}}, t) = \bar{f}_p(\bar{z}, \bar{u}, \dot{\bar{u}}, t) - A(t)\bar{z}$.

Допустимо, що для системи $\dot{\bar{z}} = A(t)\bar{z}$ існує перехідна матриця $\Phi(t, t_0)$ [3]. Тоді рівняння руху системи (3.4) набудуть вигляду

$$\begin{aligned}\bar{z}(t) &= \Phi(t, t_0)\bar{z}(t_0) + \int_{t_0}^t \Phi(t, \tau)\bar{\psi}(\bar{z}, \bar{u}, \dot{\bar{u}}, \tau)d\tau, \\ \bar{y}(t) &= C\bar{z}(t),\end{aligned}\tag{3.5}$$

де $\int_{t_0}^t d\tau$ - матричний інтеграл.

Розглянемо лінійну підсистему

$$\begin{aligned} z &= A(t)z + v, \\ y &= Cz, \end{aligned} \quad (3.6)$$

де $\bar{v}(t) = \bar{\psi}(\bar{z}, \bar{u}, \dot{\bar{u}}, t)$ – деякий вхідний вектор, розмірність якого збігається з розмірністю вектора $\bar{z}(t)$.

Допустимо, що система (3.6) спостережувана на інтервалі $[t_0, T]$, тобто знання реакції $\bar{y}(t), t \in [t_0, T]$ на відоме управління (збурення) $\bar{v}(t), t \in [t_0, T]$ достатньо для визначення стану системи $\bar{z}(t_0)$ [3]. Тоді існує перетворення – функціонал

$$\bar{z}(t_0) = G_{t_0}(\bar{v}(t), \bar{y}(t)), \quad t \in [t_0, T], \quad (3.7)$$

яке ставить у відповідність в кожний момент часу t_0 значенням векторів $\bar{v}(t)$ та $\bar{y}(t)$ значення вектора стану $\bar{z}(t_0)$. З урахуванням перетворення (3.7) запишемо

$$\bar{v}(t) = \bar{\psi}(G_t(\bar{v}, \bar{y}), \bar{u}, \dot{\bar{u}}, t) = \bar{\chi}(\bar{v}, \bar{y}, \bar{u}, \dot{\bar{u}}, t). \quad (3.8)$$

Відповідно до (3.7) та (3.8) рівняння (3.5) набувають вигляду

$$\bar{y}(t) = C\Phi(t_0, t)G_{t_0}(\bar{v}(t), \bar{y}(t)) + \int_{t_0}^t C\Phi(t, \tau)\bar{\chi}(\bar{v}, \bar{y}, \bar{u}, \dot{\bar{u}}, \tau)d\tau. \quad (3.9)$$

Рівняння (3.9) описують систему, що складається з двох підсистем: лінійної нестационарної з перехідною матрицею $C\Phi(t_0, t)$, початкові умови для якої визначені перетворенням (3.7), виходом є вектор $\bar{y}(t)$, а входом – вектор $\bar{v}(t)$; нелінійної з виходом $\bar{v}(t) = \bar{\chi}(\bar{v}, \bar{y}, \bar{u}, \dot{\bar{u}}, t)$, входами $\bar{u}, \dot{\bar{u}}, \bar{y}, \bar{v}$.

Отже, система (3.9) відповідає структурі на рис.3.1.

Теорему доведено.

Зазначимо, що вибір лінійної підсистеми досить вільний, бо під час доведення на неї накладені слабкі умови: спостережуваність та вимога існування перехідної матриці.

Недоліком отриманої структури є обмеженість інформації про нелінійну динамічну підсистему, що ускладнює побудову алгоритмів ідентифікації. Інша теорема обґрунтовує структуру, що значною мірою позбавлена цього недоліку.

3.1.2. Структура макромоделі нелінійної системи зі стаціонарною лінійною підсистемою

Теорема 2 [23]. Нехай нелінійна система допускає опис звичайними алгебро-диференціальними рівняннями

$$\begin{aligned}\dot{\bar{x}} &= \bar{f}(\bar{x}, \bar{u}, t), \\ \bar{y} &= \bar{\varphi}(\bar{x}, \bar{u}, t),\end{aligned}\quad (3.10)$$

де: $\bar{u}(t)$, $\bar{x}(t)$, $\bar{y}(t)$ – дійсні вектори вхідних впливів, стану та вихідних реакцій відповідно; дійсна вектор-функція $\bar{\varphi}(\bar{x}, \bar{u}, t)$ диференційовна за t ; сумарна розмірність векторів $\bar{x}(t)$, $\bar{y}(t)$ дорівнює k .

Тоді такій системі еквівалентна за входом-виходом система, що складається з лінійної стаціонарної динамічної та нелінійної нестационарної безінерційної підсистем згідно з рівнянням

$$a(\lambda)\bar{y} = D(\lambda)\bar{\chi}(\bar{y}, \bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}, \bar{v}^{(1)}, \dots, \bar{v}^{(k-2)}, \bar{u}, \dot{\bar{u}}, t). \quad (3.11)$$

Матриця передавальних функцій $W(\lambda) = D(\lambda)/a(\lambda)$ описує лінійну підсистему з вихідними векторами $\bar{y}, \dots, \bar{y}^{(k-1)}, \bar{v}, \dots, \bar{v}^{(k-2)}$ та вхідним вектором \bar{v} , а нелінійна вектор-функція $\bar{\chi}(\cdot)$ відповідає нелінійній підсистемі з векторними входами $\bar{y}, \dots, \bar{y}^{(k-1)}, \bar{v}, \dots, \bar{v}^{(k-2)}, \bar{u}, \dot{\bar{u}}$ і вихідним вектором \bar{v} .

Систему (3.11) зручно зобразити блок-схемою (рис.3.2).

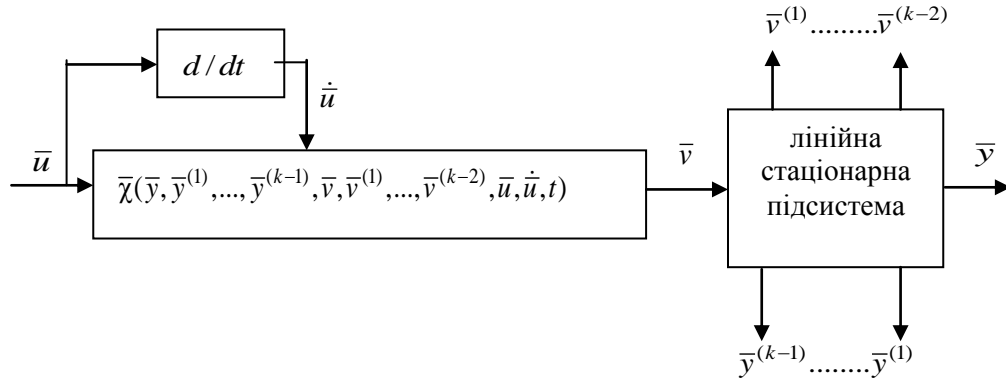


Рис.3.2. Структура макромоделі нелінійної системи до теореми 2

Доведення. Введемо розширений координатний базис $\bar{z}(t) = [\bar{x}(t), \bar{y}(t)]$ та виділимо деяку лінійну стаціонарну підсистему з матрицею рівнянь стану A

$$\begin{aligned}\dot{\bar{z}} &= A\bar{z} + \bar{\psi}(\bar{z}, \bar{u}, \dot{\bar{u}}, t), \\ \bar{y} &= C\bar{z}.\end{aligned}\quad (3.12)$$

Розглянемо лінійну підсистему

$$\begin{aligned}\dot{\bar{z}} &= A\bar{z} + \bar{v}, \\ \bar{y} &= C\bar{z},\end{aligned}\tag{3.13}$$

де $\bar{v} = \bar{\psi}(\bar{z}, \bar{u}, \dot{\bar{u}}, t)$ – вхідний вектор підсистеми.

Вилучимо вектор \bar{z} , позначивши оператор диференціювання символом λ , а одиничну матрицю – I :

$$\lambda\bar{z} = A\bar{z} + \bar{v}; \quad (\lambda I - A)\bar{z} = \bar{v}; \quad \bar{z} = (\lambda I - A)^{-1}\bar{v}; \quad \bar{y} = C(\lambda I - A)^{-1}\bar{v}.\tag{3.14}$$

Елементи матриці $(\lambda I - A)^{-1}$, відповідно до визначення оберненої матриці, є дробово-раціональними функціями λ , причому знаменники цих функцій однакові та дорівнюють визначнику $|\lambda I - A|$, а чисельники – це відповідні мінори матриці $(\lambda I - A)$.

Позначимо $a(\lambda) = |\lambda I - A|$; $B(\lambda)$ – матриця, поліноміальні елементи якої – мінори матриці $(\lambda I - A)$; $D(\lambda) = CB(\lambda)$.

Тоді рівняння (3.14) можна записати у вигляді

$$a(\lambda)\bar{y} = D(\lambda)\bar{v}.\tag{3.15}$$

Нехай k – степінь полінома $a(\lambda)$, яка у загальному випадку збігається з розмірністю вектора \bar{z} , а m – максимальний степінь поліномів у матриці $D(\lambda)$, причому $m = k - 1$. Як доведено в [3], стан лінійної підсистеми (3.15) визначений вектором $\bar{r} = (\bar{y}, \bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}, \bar{v}^{(1)}, \dots, \bar{v}^{(m-1)})$.

Покажемо таке: якщо система (3.13) спостережувана [3], то існує лінійне перетворення

$$\bar{z} = G\bar{r}.\tag{3.15a}$$

Продиференціюємо $(k-1)$ раз векторне рівняння $\bar{y} = C\bar{z}$:

$$\begin{aligned}\bar{y} &= C\bar{z}, \\ \bar{y}^{(1)} &= C\bar{z}^{(1)} = CA\bar{z} + C\bar{v}, \\ \bar{y}^{(2)} &= CA^2\bar{z} + CA\bar{v} + C\bar{v}^{(1)}, \\ &\vdots \\ \bar{y}^{(k-1)} &= CA^{k-1}\bar{z} + \sum_{i=0}^{k-2} CA^i\bar{v}^{(k-i-2)}.\end{aligned}\tag{3.16}$$

Система рівнянь (3.16) у матричному записі має вигляд

$$\begin{bmatrix} \bar{y} \\ \bar{y}^{(1)} \\ \vdots \\ \bar{y}^{(k-1)} \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{bmatrix} \bar{z} + \begin{bmatrix} 0 & \dots & \dots & 0 \\ C & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ CA^{k-2} & CA^{k-3} & \dots & C \end{bmatrix} \begin{bmatrix} \bar{v} \\ \bar{v}^{(1)} \\ \vdots \\ \bar{v}^{(k-2)} \end{bmatrix}. \quad (3.17)$$

Умова існування розв'язку системи (3.17) щодо вектора \bar{z} збігається з умовою спостережуваності системи (3.13)

$$\text{rank} \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{bmatrix} = k, \quad (3.18)$$

а матриця G з (3.15а) має вигляд

$$G = R^{-1} \begin{bmatrix} & 0 & 0 & \dots & 0 \\ I & -C & 0 & \dots & 0 \\ & \vdots & \vdots & & \vdots \\ & -CA^{k-2} & -CA^{k-3} & \dots & -C \end{bmatrix}. \quad (3.19)$$

Матриця R в (3.19) утворена k лінійно незалежними стрічками матриці у (3.18), а матриця-співмножник містить лише відповідні k стрічок.

Отже, вираз для вектора \bar{v} набуває вигляду

$$\bar{v} = \bar{\psi}(G\bar{r}, \bar{u}, \dot{\bar{u}}, t) = \bar{\chi}(\bar{r}, \bar{u}, \dot{\bar{u}}, t). \quad (3.20)$$

Враховуючи (3.20), рівняння (3.15) можна остаточно записати так:

$$a(p)\bar{y} = D(p) \cdot \bar{\chi}(\bar{y}, \bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}, \bar{v}^{(1)}, \dots, \bar{v}^{(k-2)}, \bar{u}, \dot{\bar{u}}, t). \quad (3.21)$$

Тут матриця передатних функцій $W(\lambda) = D(\lambda)/a(\lambda)$ описує лінійну підсистему з вихідним вектором \bar{y} та вхідним вектором \bar{v} , а нелінійна вектор-функція $\bar{\chi}(\cdot)$ відповідає нелінійній підсистемі з векторними входами $\bar{y}, \dots, \bar{y}^{(k-1)}, \bar{v}, \dots, \bar{v}^{(k-2)}, \bar{u}, \dot{\bar{u}}$ та вихідним вектором \bar{v} . Без порушення загальності можна вважати, що обчислення похідних $\bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}^{(1)}, \dots, \bar{v}^{(k-2)}$ виконується у лінійній підсистемі з входом \bar{v} та виходами $\bar{y}, \bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}^{(1)}, \dots, \bar{v}^{(k-2)}$. Отримана структура повністю відповідає структурі на рис.3.4. Теорему доведено.

3.1.3. Перетворення рівнянь Лоренца згідно зі структурою макромоделі (3.11)

Розглянемо приклад перетворення системи відповідно до теореми 2. Як звичайно, аналітичне перетворення нездійсненне, однак у деяких простих випадках це можливо.

Розглянемо відомі рівняння, що описують дивний атрактор Е.Лоренца [52].

$$\begin{aligned}\dot{x}_1 &= 10x_2 - 10x_1; \\ \dot{x}_2 &= 40x_1 - x_2 - x_1x_3; \\ \dot{x}_3 &= -8/3 \cdot x_3 + x_1x_2.\end{aligned}$$

Нехай вихідний сигнал збігається з однією з фазових змінних: $y=x_1$. У цьому випадку немає потреби розширювати координатний базис і можна безпосередньо записати рівняння системи у вигляді (3.13)

$$\begin{aligned}\dot{z}_1 &= -10z_1 + 10z_2 + v_1; \\ \dot{z}_2 &= 40z_1 - z_2 - z_3 + v_2; \\ \dot{z}_3 &= -8/3z_3 + v_3; \\ y &= z_1,\end{aligned}$$

де $z_i = x_i, i=1,2,3; v_1=0; v_2 = z_3 - z_1z_3; v_3 = z_1z_2;$

$$A = \begin{bmatrix} -10 & 10 & 0 \\ 40 & -1 & -1 \\ 0 & 0 & -8/3 \end{bmatrix}; \quad \bar{v} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^t; \quad C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}.$$

У друге рівняння додатково введені два члени $+z_3$ та $-z_3$ для виконання умови спостережуваності лінійної підсистеми (3.18).

Перетвореннями (3.14) і (3.15) утворюємо рівняння лінійної підсистеми

$$\ddot{y} + 41/3\ddot{y} - 1082/3\dot{y} - 1040y = 10\dot{v}_2 + 80/3v_2 - 10v_3.$$

Для нашого прикладу $k=3$, а стан визначений вектором $\bar{r} = (y, \dot{y}, \ddot{y}, v_1, v_2, v_3, \dot{v}_1, \dot{v}_2, \dot{v}_3)$.

Виконаємо перетворення $\bar{z} = G\bar{r}$:

$$R^{-1} = \begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -10 & 10 & 0 \\ 500 & -110 & -10 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0.1 & 0 \\ 39 & -1.1 & -0.1 \end{bmatrix};$$

$$H = \begin{bmatrix} & 0 & 0 \\ I & -C & 0 \\ & -CA & -C \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 10 & -10 & 0 & -1 & 0 & 0 \end{bmatrix};$$

$$G = R^{-1}H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0.1 & 0 & -0.1 & 0 & 0 & 0 & 0 & 0 \\ 39 & -1.1 & -0.1 & 0.1 & 1 & 0 & 0.1 & 0 & 0 \end{bmatrix};$$

$$z_1 = y;$$

$$z_2 = y + 0.1\dot{y} - 0.1v_1;$$

$$z_3 = 39y - 1.1\dot{y} - 0.1\ddot{y} + 0.1v_1 + v_2 + 0.1\dot{v}_1.$$

Підставивши замість v_1 та v_2 відповідні вирази, можна визначити z_i щодо y, \dot{y}, \ddot{y} :

$$z_1 = y;$$

$$z_2 = y + 0.1\dot{y};$$

$$z_3 = 39 - 1.1\dot{y}/y - 0.1\ddot{y}/y.$$

Змінні v_2, v_3, \dot{v}_2 записані вище через змінні стану z_1, z_2, z_3 . Тепер можемо остаточно записати змінні v_2, v_3, \dot{v}_2 через вихідну змінну y та її похідні:

$$v_2 = 39 - 1.1\dot{y}/y - 0.1\ddot{y}/y - 39y + 1.1\dot{y} + 0.1\ddot{y};$$

$$v_3 = y^2 + 0.1y\dot{y};$$

$$\dot{v}_2 = 1.1\dot{y}^2/y - 0.1\ddot{y}y^2 + 0.1y\ddot{y} - 1258/30\dot{y} + 0.1\ddot{y}\ddot{y}/y + 88/30\dot{y}/y - y^3 + y^2 + 104y - 8/30\ddot{y} - 104 + 8/30\ddot{y}/y.$$

Останні співвідношення сумісно з рівнянням лінійної підсистеми утворюють шукану структуру моделі системи Лоренца, яка зображена на рис.3.3.

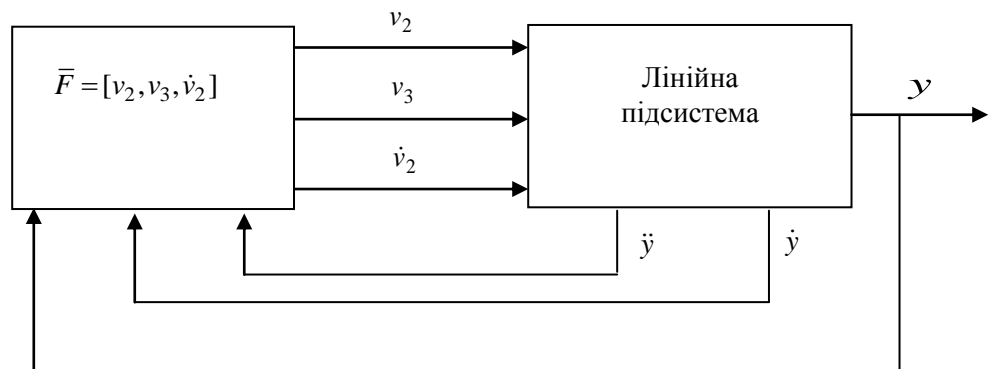


Рис.3.3. Структура макромоделі системи Лоренца

Рівняння Лоренца, перетворені згідно зі структурою (3.11), використані для макромодельовання дивних атракторів у параграфі 4.1.

3.1.4. Окремі випадки загальної моделюючої структури (3.11)

Доведені теореми визначають загальні структури моделей нелінійних динамічних систем з виділеною лінійною частиною.

Практична цінність отриманих структур залежить від наявності методів ідентифікації. Такі методи побудовані для структури з рис.3.2 і розглянуті нижче.

Суттєво, що лінійні підсистеми є досить довільними. Із доведення теореми 2 впливає лише одна вимога до лінійної стаціонарної підсистеми: спостережуваність. Таке незначне обмеження забезпечує вільну побудову методів ідентифікації.

Наведемо деякі окремі випадки структури на рис.3.2.

Нехай вихідний сигнал \bar{y} і внутрішній вектор \bar{v} – скаляри. Тоді матриця $D(\lambda)$ в (3.21) перетворюється в поліном $d(\lambda)$, вектор-функція $\bar{\chi}(\cdot)$ – у функцію-скаляр, передатній функції $w(\lambda)=d(\lambda)/a(\lambda)$ відповідає вагова функція $g(t)$, а рівняння (3.21) зводяться до одного інтегрального рівняння

$$y(t) = \int_0^t g(t-\tau) \chi(y, y^{(1)}, \dots, y^{(k-1)}, v, \dots, v^{(k-2)}, u, \dot{u}, \tau) d\tau \quad (3.22)$$

Якщо тепер у списку аргументів функції $\chi(\cdot)$ залишити тільки y, u, t , то отримаємо рівняння Гамерштейна [9], яке часто застосовують для моделювання нелінійних динамічних систем:

$$y(t) = \int_0^t g(t-\tau) \chi(y, u, \tau) d\tau \quad (3.23)$$

Використовують, крім того, для моделювання нелінійних динамічних систем диференціальні рівняння (3.24) [8]

$$\begin{aligned} \dot{y} &= y_1; \\ \dot{y}_1 &= y_2; \\ &\vdots \\ \dot{y}_n &= \sum_{i_1=0}^{k_1} \dots \sum_{i_{n+3}=0}^{k_{n+3}} c_{i_1 i_2 \dots i_{n+2}} y_1^{i_1} y_2^{i_2} \dots y_n^{i_n} u^{i_{n+2}} \dot{u}^{i_{n+3}}; \end{aligned} \quad (3.24)$$

де $c_{i_1 i_2 \dots i_{n+2}}$ – коефіцієнти багатовимірною степеневому полінома.

Така структура є окремим випадком структури на рис.3.2, якщо \bar{y}, \bar{u} – скаляри, а рівняння лінійної та нелінійної підсистем мають вигляд

$$\frac{d}{dt} \begin{bmatrix} y \\ y_1 \\ \vdots \\ y_n \\ u \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} y \\ y_1 \\ \vdots \\ y_n \\ \dot{u} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ v_n \\ 0 \end{bmatrix};$$

$$v_n = \chi_n(y, y_1, \dots, y_n, u) = \sum_{i_1=0}^{k_1} \dots \sum_{i_{n+3}=0}^{k_{n+3}} c_{i_1 \dots i_{n+2}} y^{i_1} \dots y_n^{i_{n+1}} u^{i_{n+2}} \dot{u}^{i_{n+3}}.$$

Зазначимо, що в цьому випадку нелінійна безінерційна функція обов'язково однозначна, тоді як з теореми 2 така умова не впливає.

У теоремах 1, 2 обґрунтовані структури, що різняться типом лінійної підсистеми (відповідно нестационарної та стаціонарної). Можливих варіантів загальних структур нелінійних систем, очевидно, дуже багато, наприклад, структури на основі теорії розщеплення [14], рядів Вольтерра [16], рядів Вольтерра-Пікара [10], або нерекурсивна структура Вінера, розглянута нижче. Однак стосовно вихідних посилань на початку розділу та опису (3.1) теореми 1 і 2 вичерпують варіанти загальних структур.

Структури на рис.3.1 та 3.2 є рекурсивними, тобто містять, користуючись кібернетичним терміном, зворотні зв'язки. Це деколи ускладнює ідентифікацію, оскільки внаслідок внутрішніх процесів у самій макро-моделі її поведінка може сильно відрізнятися від потрібної, незважаючи на зовнішню коректність ідентифікації (макромодель нестійка). З огляду на це, а також зважаючи на складність ідентифікації, становлять інтерес структури нерекурсивні.

3.1.5. Нерекурсивна структура Вінера

Нелінійні динамічні системи часто моделюють за допомогою структури, зображеної на рис.3.4, що її називають моделлю Вінера.

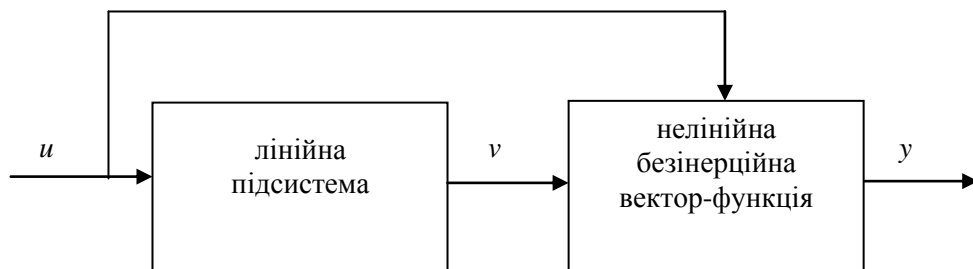


Рис.3.4. Структура моделі Вінера

Цю структуру описують рівняння

$$\begin{aligned}\dot{v} &= Av + Bu, \\ y &= \chi(v, u, t).\end{aligned}\tag{3.25}$$

Чи можна отримати (3.25) з опису (3.1)?

Одразу зазначимо, що загальне доведення нам невідоме. Однак для деяких часткових випадків можна вивести (3.25) з (3.1).

Теорема 3. Нехай система першого порядку допускає опис

$$\begin{aligned}\dot{x} &= f(x, u, t), \\ y &= \varphi(x, u, t),\end{aligned}\tag{3.26}$$

де x – скаляр, а функції $f(\cdot)$ та $\varphi(\cdot)$ – неперервні.

Тоді існує система першого порядку

$$\begin{aligned}\dot{v} &= av + bu, \\ y &= \chi(v, u, t),\end{aligned}\tag{3.27}$$

еквівалентна системі (3.26) за входом-виходом.

Доведення. Покажемо, що існує диференційовне перетворення $v = \psi(x)$, яке задовольняє рівняння

$$\frac{d\psi}{dx} \dot{x} = a\psi + bu.\tag{3.28}$$

Згідно з (3.26) рівняння (3.28) можна записати так:

$$\frac{d\psi}{dx} = \frac{1}{f(x, u, t)}(a\psi(x) + bu).$$

Розв'язок останнього рівняння, тобто функція $\psi(x)$, існує, бо права частина неперервна щодо $\psi(x)$ [53].

Оскільки (3.26) та (3.27) є описами еквівалентних систем і перетворення $v = \psi(x)$ існує, то є зворотне перетворення $x = \phi(v)$.

Тоді можна записати

$$y = \varphi(\phi(v), u, t) = \chi(v, u, t).$$

Теорему доведено.

Теорема 4. Нехай система допускає опис

$$\begin{aligned}\dot{\bar{x}} &= \bar{f}(\bar{x}, \bar{u}), \\ \bar{y} &= \bar{\varphi}(\bar{x}, \bar{u}, t),\end{aligned}\tag{3.29}$$

де вектор-функція $\bar{f}(\bar{x}, \bar{u})$ хоча б двічі диференційовна в ε -околі точки $(\bar{x}_0, \bar{u}_0, t_0)$.

Тоді існує система (3.25), що відтворює поведінку системи (3.29) за входом-виходом з будь-якою наперед заданою точністю в ε -околі точки $(\bar{x}_0, \bar{u}_0, t_0)$.

Доведення. Розкладемо вектор-функцію $\bar{f}(\bar{x}, \bar{u})$ в ряд Тейлора в околі точки $(\bar{x}_0, \bar{u}_0, t_0)$, відкинувши члени другого та вищих порядків малості:

$$\bar{f}(\bar{x}, \bar{u}) \approx \bar{f}(\bar{x}_0, \bar{u}_0) + \left. \frac{\partial \bar{f}}{\partial \bar{x}} \right|_{\bar{x}_0, \bar{u}_0} (\bar{x} - \bar{x}_0) + \left. \frac{\partial \bar{f}}{\partial \bar{u}} \right|_{\bar{x}_0, \bar{u}_0} (\bar{u} - \bar{u}_0).$$

Допустимо існування оберненої матриці $\left[\left. \frac{\partial \bar{f}}{\partial \bar{x}} \right|_{\bar{x}_0, \bar{u}_0} \right]^{-1} = F$ і введемо

позначення $\bar{v} = \bar{x} - \bar{x}_0 + F \cdot (\bar{f}(\bar{x}_0, \bar{u}_0) - \left. \frac{\partial \bar{f}}{\partial \bar{u}} \right|_{\bar{x}_0, \bar{u}_0} \cdot \bar{u}_0).$

Враховуючи, що $\frac{d\bar{v}}{dt} = \frac{d\bar{x}}{dt}$ і позначивши $A = \left. \frac{\partial \bar{f}}{\partial \bar{x}} \right|_{\bar{x}_0, \bar{u}_0}$, $B = \left. \frac{\partial \bar{f}}{\partial \bar{u}} \right|_{\bar{x}_0, \bar{u}_0}$,

отримуємо

$$\begin{aligned}\dot{\bar{v}} &= A\bar{v} + B\bar{u}, \\ \bar{y} &= \bar{\chi}(\bar{v}, \bar{u}, t),\end{aligned}$$

що й треба було довести.

Теорема 3 та 4 стосуються тільки систем першого порядку або систем зі слабконелінійною динамічною частиною, унаслідок чого використовувати структуру (3.25) треба з певною обережністю. В наступному параграфі наведено приклад використання структури Вінера для побудови багатовимірної макромоделі операційного підсилювача.

Ідентифікація структур нелінійних систем полягає в ідентифікації лінійної динамічної підсистеми, визначенні внутрішнього вектора \bar{v} та апроксимації нелінійної вектор-функції $\bar{\chi}(\cdot)$.

Загальні процедури ідентифікації нам невідомі. У наступних параграфах розглянуто ідентифікаційні методи для спрощених варіантів запропонованих структур, а також труднощі, пов'язані з некоректностями ідентифікаційних задач.

3.2. ІДЕНТИФІКАЦІЯ ЗА МЕТОДОМ НЕЛІНІЙНОЇ СТАТИКИ ТА МАЛОСИГНАЛЬНОЇ ДИНАМІКИ

3.2.1. Загальні положення

Допустимо, що список аргументів функції $\bar{\chi}(\cdot)$ у структурі (3.11) не містить похідних функцій \bar{y}, \bar{u} . Тоді така структура, як було показано вище, є моделлю Гамерштейна (3.23).

Природний підхід до ідентифікації системи, що складається з лінійної динамічної та нелінійної безінерційної підсистем, полягає в малосигнальній ідентифікації лінійної підсистеми та ідентифікації нелінійної підсистеми у статичному режимі. Відповідні методи розроблені [15, 39], однак вони стосуються лише нерекурсивних моделей Гамерштейна, де $\bar{\chi}(\cdot) = \chi(\bar{u})$. Далі наведені методи ідентифікації рекурсивних моделей Гамерштейна, зокрема з неоднозначними нелінійними функціями.

Розглянемо ідентифікацію одновимірної рекурсивної структури на прикладі рекурсивної моделі Гамерштейна.

Нехай у моделі Гамерштейна (3.23) нелінійна функція стаціонарна та має вигляд $\chi(u, y) = u - F(y)$:

$$y(t) = \int_0^t g(t - \tau)(u - F(y))d\tau \quad (3.30)$$

Якщо ваговій функції $g(t)$ відповідає дробово-раціональна передатна функція $w(\lambda) = b(\lambda)/a(\lambda)$ (див. 2.3), то за нульових початкових умов інтегральному рівнянню (3.30) відповідає диференціальне рівняння

$$a(\lambda)y = b(\lambda)(u - F(y)), \quad \lambda = d/dt. \quad (3.31)$$

Параметри такої системи можна ідентифікувати у нелінійному статичному та малосигнальному динамічному режимах

У статичному режимі функції $u(t)$ та $y(t)$ є константами u_c та y_c , а інтегральне рівняння (3.30) перетворюється в алгебричне

$$y_c = k_c(u_c - F(y_c)), \quad (3.32)$$

де $k_c = \lim_{t \rightarrow \infty} \int_0^t g(t - \tau) d\tau = b_0 / a_0$ – статичний коефіцієнт передачі лінійної

підсистеми $w(\lambda) = \sum_{i=0}^n b_i \lambda^i / \sum_{i=0}^n a_i \lambda^i$.

Функцію $F(y_c)$ апроксимуємо степеневим поліномом

$$F(y_c) = f_0 + \sum_{j=2}^m f_j y_c^j, \quad (3.33)$$

а коефіцієнти апроксимації f_j знайдемо сумісним розв'язком $M \geq m$ рівнянь для пар значень u_{ck} та y_{ck}

$$\sum_{j=0}^m f_j y_{ck}^j = u_{ck}, \quad k = \overline{1, M}, \quad (3.34)$$

відносно коефіцієнтів f_j , причому $f_1 = 1/k_c$. За методом найменших квадратів це відповідає лінійній задачі щодо компонент вектора $\bar{f} = (f_0, f_2, \dots, f_m)$:

$$\min_{\bar{f}} ((f_0 + \sum_{j=2}^m f_j y_{ck}^j) - (u_{ck} - y_{ck} / k_c))^2; \quad k = \overline{1, M}. \quad (3.35)$$

Функцію $w(\lambda)$ ідентифікуємо в режимі малих відхилень від деякого стійкого положення рівноваги u_0, y_0 . Позначимо $\tilde{u} = u - u_0$; $\tilde{y} = y - y_0$; $\tilde{k} = dF / dy$ при $y = y_0$. Запишемо рівняння (3.31) стосовно малих приростів в околі точки (u_0, y_0) :

$$\sum_{i=0}^n a_i \tilde{y}^{(i)} + \sum_{i=0}^n b_i \tilde{k} \tilde{y}^{(i)} = \sum_{i=0}^n b_i \tilde{u}^{(i)}. \quad (3.36)$$

Описаною у попередньому розділі процедурою ідентифікації лінійної системи (2.19) можемо визначити коефіцієнти малосигнальної макромоделі в околі точки u_0, y_0 :

$$\sum_{i=0}^n c_i \tilde{y}^{(i)} = \sum_{i=0}^n b_i \tilde{u}^{(i)}. \quad (3.37)$$

Очевидно, шукані коефіцієнти (3.36) визначені формулами

$$a_i = c_i - \tilde{k}b_i; \quad i = \overline{1, n}. \quad (3.38)$$

Остаточно макромодель має вигляд

$$\sum_{i=0}^n (c_i - \tilde{k}b_i) y^{(i)} = \sum_{i=0}^n b_i (u - f_0 - \sum_{j=2}^m f_j y^j)^{(i)}, \quad (3.39)$$

де повинне виконуватись співвідношення $b_0 / (c_0 - \tilde{k}b_0) = k_c$.

Макромодель без надлишкового порядку отримуємо, якщо лінійну підсистему перетворити згідно з (2.20).

Описаний метод ідентифікації є методом ідентифікації рекурсивної макромоделі Гамерштейна для обраного вигляду нелінійної функції, відображеного в (3.30). Можливі варіанти цього методу залежно від вигляду нелінійної функції.

3.2.2. Ідентифікація рекурсивної макромоделі. Макромодель тригера

Спочатку ідентифікуємо нелінійну безінерційну підсистему за статичною характеристикою тригера з використанням системи САНОС-ПК.

Схема симетричного тригера показана на рис.3.5.

Текст завдання для САНОС-ПК з описом симетричного тригера та формуванням масивів значень E2 і V2, потрібних для ідентифікації статичної характеристики, наведено у табл.3.1.

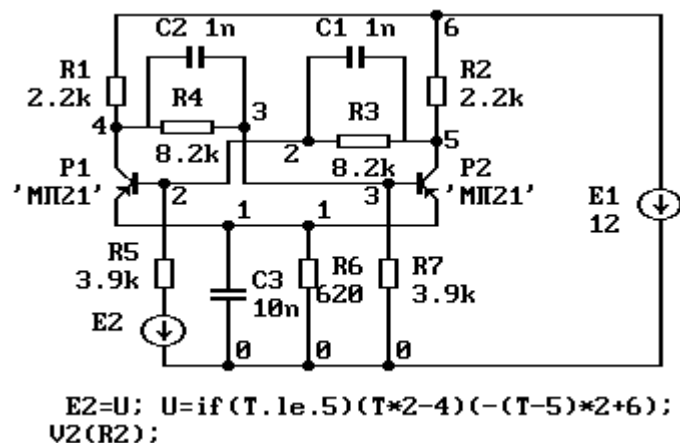


Рис.3.5. Схема тригера з входним сигналом $U=E2$ та вихідним сигналом $Y=V2(R2)$

Таблиця 3.1. Текст завдання для вимірювання статичної характеристики тригера

BEGI

Зняття статичних характеристик тригера

DESC

R1(6,4)2.2K; R2(6,5)2.2K; R3(2,5)8.2K; R4(4,3)8.2K;
 R5(2,7)3.9K; R6(1,0)620; R7(3,0)3.9K; V2(R2);
 C1(2,5)1n; C2(4,3)1n; C3(1,0)10n; E1(6,0)12;
 E2(7,0) IF(T.LE.5)(-4+T*2)(6-(T-5)*2); U=E2;
 P1(1,2,4)'МП21'; P2(1,3,5)'МП21';;

TRAN

TMAX=10; ZERO; HS=0.02; HMIN=1E-15; ERMAX=0.01;
 OUTPUT: E2, V2, TABL, XMIN=0,XMAX=10,YMIN=-15,YMAX=15;;

END

Згідно із завданням у табл.3.1, дані для розв'язування апроксимаційної задачі (3.35) формують у вигляді таблиці в робочому файлі системи САНОС-ПК як результат розрахунку перехідного процесу в тригері, де вхідна напруга $E2$ повільно лінійно змінюється від $-4V$ до $+6V$ та від $+6V$ до $-4V$. У цьому випадку в тригері двічі відбувається перекидання між положеннями стійкої рівноваги, що відображене стрибкоподібними змінами вихідної напруги $V2$, як це показано на рис.3.6.

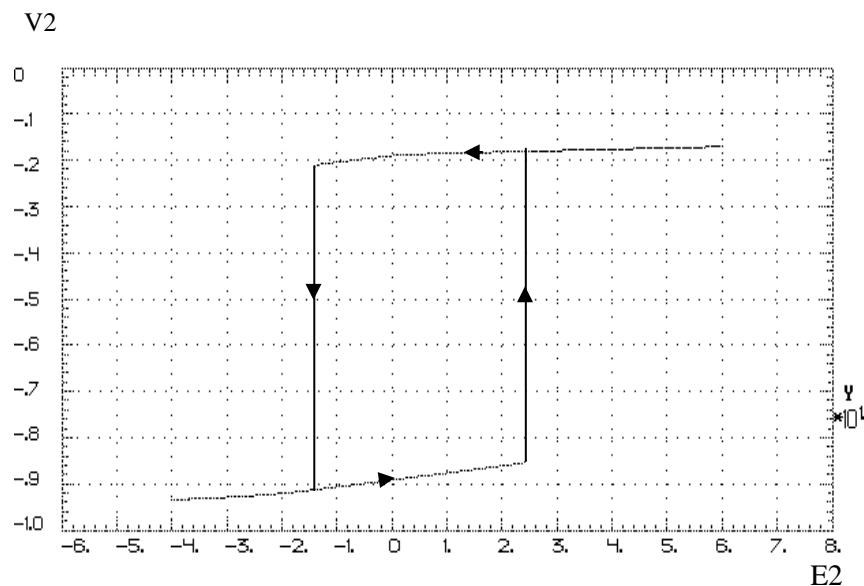


Рис.3.6. Статична характеристика тригера, зображеного на рис.3.5

Нехай треба апроксимувати статичну характеристику $V2(E2)=Y(U)$ степеневим поліномом. Безпосередньо за даними з рис.3.6 (або з табл.3.1) це виконати неможливо, бо залежність $Y(U)$ є неоднозначною і не може бути апроксимованою степеневим поліномом вигляду $Y=\sum f_i U^i$. Однак можлива апроксимація у вигляді неявної залежності $Y=Y(U,Y)$.

Для цього поміняємо місцями аргумент і функцію на рис.3.6. Функція $U(Y)$ є вже однозначною і допускає апроксимацію у вигляді $U=\sum f_i Y^i$. Далі треба розв'язати отримане рівняння щодо Y :

$$Y = 1/f_1 * (U - f_0 - \sum_{i=2}^n f_i Y^i) . \quad (3.40)$$

Однак лише даних з табл.3.1 недостатньо для задовільної апроксимації. Це очевидно, бо вузли апроксимації задані лише на майже горизонтальних ділянках, показаних на рис.3.6, і поведінка апроксимуючого полінома не контролюється в області нестійких положень рівноваги. На рис.3.7 показано результат апроксимації за даними завдання у табл.3.1, де вектор \bar{f} знайдено методом найменших квадратів для $n=8$ і $m=777$ за допомогою програми, текст якої міститься в Додатку 4:

$$\min_{\bar{f}} \sum_{j=1}^m (U_j - \sum_{i=0}^n f_i Y_j^i)^2 \quad (3.40a)$$

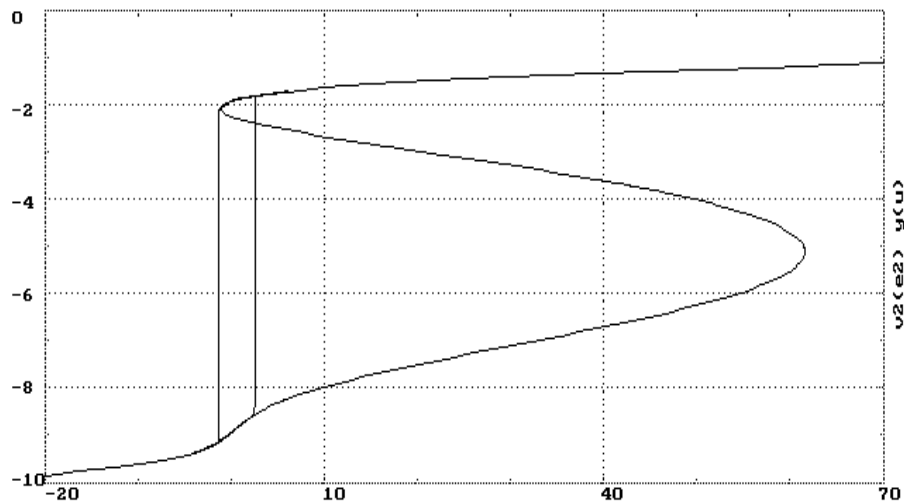


Рис.3.7. Незадовільна апроксимація статичної характеристики поліномом 8-го степеня без додаткових точок у зоні нестійкості

Апроксимуючий степеневий поліном восьмого степеня добре відтворює задані фрагменти статичної характеристики, проте поза ними поводить себе зовсім не так, як потрібно для якісної апроксимації в цілому. Очевидно, що у випадку такої апроксимації неможливе перекидання тригера відповідно до рис.3.6. Така ситуація є наслідком того, що експериментальних значень в області нестійких положень рівноваги тригера нема.

В області значень, що обмежена прямокутником $E2=(-1.18, +2.23)$ та $V2=(-2.2,-8.5)$, де нема стійких положень рівноваги тригера і тому нема прямих експериментальних даних, необхідно задати якісь умовні вузли апроксимації. Пропонуємо створити ці додаткові дані як точки відрізка прямої, що проходить через зазначену область. Положення прямої підбираємо емпірично залежно від якості отриманої апроксимуючої функції.

У табл.3.2 наведено завдання мовою САНОС-ПК, що доповнює робочий файл апроксимації додатковими вузлами.

Таблиця 3.2. Створення додаткових вузлів апроксимації в області нестійкості

```

BEGI
  Доповнення відрізком прямої між точками перекидання (двічі)
DESC
  w'=1; *Допоміжне рівняння;
  V21=-8.5; U1=2.1; V22=-2.2; U2=1.0; * Координати майже вертикального
  відрізка;
  U=(U2-U1)/(V22-V21)*V2+(U1*V22-U2*V21)/(V22-V21); * Рівняння прямої;
  V2=if(T.le.-2.2)(T)(-4.4-T);;;
TRAN
  tin=-8.5; tmax=4.1; hs=0.03;
  output: U, V2, tabs, xmin=-9,xmax=5,ymin=-15,ymax=15;;
END

```

Коефіцієнти апроксимації згідно із задачею (3.40а) обчислює FORTRAN-програма з Додатку 4, яка формує ще й файл завдання мовою САНОС-ПК для порівняльного розрахунку статичних характеристик тригера та макромоделі. Вміст такого файлу для тригера наведено в табл.3.3.

Таблиця 3.3. Порівняння статичних характеристик тригера та безінерційної нелінійної підсистеми макромоделі

```

BEGI
  Поліноміальна апроксимація статики тригера.
DESC
  * Схема тригера ;
  R1(6,4)2.2K; R2(6,5)2.2K; R3(2,5)8.2K; R4(4,3)8.2K;
  R5(2,7)3.9K; R6(1,0)620; R7(3,0)3.9K;
  C1(2,5)1n; C2(4,3)1n; C3(1,0)10n; E1(6,0)12;
  E2(7,0) IF(T.LE.5)(6-T*2)(-4+(T-5)*2); U=E2;
  P1(1,2,4)'МП21'; P2(1,3,5)'МП21'; V2(R2);
  * Апроксимація статики, кількість експериментальних точок – 777;
  f0=854.8; f1=1553.8; f2=1171.9; f3=480.9; f4=118.01; f5=17.806; f6=1.6193;
  f7=8.1402E-02; f8=1.7365E-03; * Коефіцієнти апроксимації;
  YY=1/f1*(U-f0-Y*Y*(f2+Y*(f3+Y*(f4+Y*(f5+Y*(f6+Y*(f7+Y*f8))))));
  * Рівняння, що перетворює неоднозначну нелінійність в однозначну;
  Y`=1e6*(YY-Y);
  * Демонстрація апроксимаційної кривої ;
  u=f0+y*(f1+y*(f2+y*(f3+y*(f4+y*(f5+y*(f6+y*(f7+y*f8)))))); y=-T;;
TRAN
  TMAX=10; ZERO; HS=4E-2; HMIN=1E-9; allstep;
  OUTPUT: V2(E2), Y(U), y(u), XMIN=-5, XMAX=7, YMIN=-10, YMAX=0;;
END

```

У завданні табл.3.3 неоднозначну нелінійну функцію у вигляді (3.40) замінено алгебро-диференціальним рівнянням (3.41), що дає змогу застосовувати стандартні методи числового інтегрування для відтворення швидких рухів у разі перекидання тригера. Ця ситуація детально проаналізована в класичній праці А.А.Андропова, А.А.Вітта, С.Е.Хайкіна [54, с.787].

$$\begin{aligned}\dot{Y} &= 1/\tau (YU - Y), \\ YU &= 1/f_1 (U - f_0 - \sum_{i=2}^n f_i Y^i).\end{aligned}\quad (3.41)$$

Вплив допоміжного диференціального рівняння в (3.41) на форму перехідного процесу в динамічній макромоделі тригера можна зробити як завгодно малим, якщо вибрати достатньо малою сталу часу τ .

Компіляція в системі САНОС-ПК завдання, що міститься в табл.3.3, дає змогу порівняти статичні характеристики тригера та побудованої поліноміальної апроксимації. Результат показано на рис.3.8.

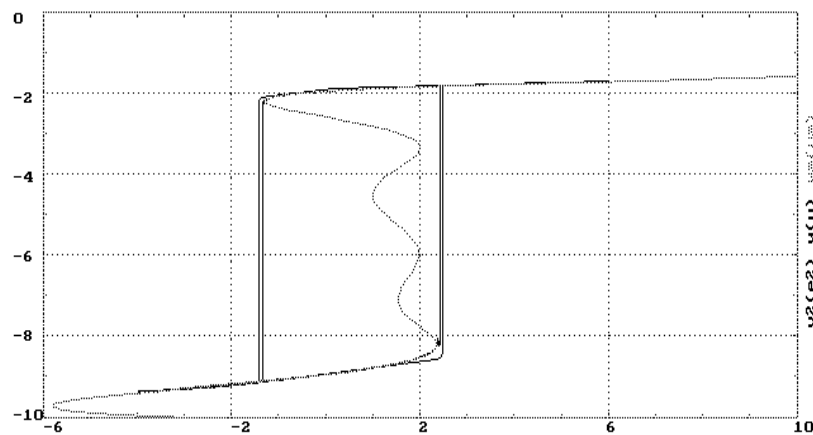


Рис.3.8. Статичні характеристики тригера та поліноміальної макромоделі за завданням у табл.3.3

Можлива також кусково-лінійна апроксимація статичної характеристики. Вона складається з відрізків двох прямих, що апроксимують майже горизонтальні ділянки статичної характеристики, та відрізка прямої, що проходить через зону нестійкості. Для створення кусково-лінійної апроксимації не треба застосовувати спеціальні апроксимаційні програми, оскільки рівняння прямих формуються безпосередньо з координат обраних вузлових точок статичної характеристики. Треба лише записати умови переходу від однієї ділянки апроксимації до іншої.

Нехай вузлові точки на статичній характеристиці мають координати (U_1, Y_1) , (U_2, Y_2) , (U_3, Y_3) , (U_4, Y_4) , показані на рис.3.9.

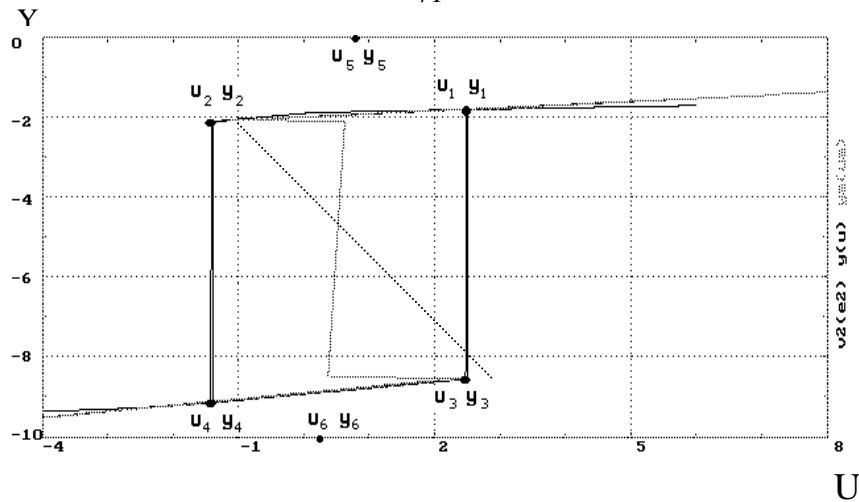


Рис.3.9. Статичні характеристики тригера та кусково-лінійної макромоделі (табл.3.4)

Верхня майже горизонтальна пряма проходить через точки (U_1, Y_1) , (U_2, Y_2) , нижня майже горизонтальна пряма – через точки (U_3, Y_3) , (U_4, Y_4) . Що стосується середнього відрізка, то природно отримати неперервну апроксимацію, провівши середній відрізок через точки (U_2, Y_2) , (U_3, Y_3) (штрихова лінія на рис.3.9). Проте в такому випадку під час інтегрування відповідних диференціальних рівнянь виникає “зависання” у точках стрибків з верхньої на нижню та з нижньої на верхню прямі (точки (U_2, Y_2) та (U_3, Y_3) відповідно).

Подібні явища у системах з кусково-лінійною апроксимацією детально вивчені [55]. Добре відомий загальний рецепт боротьби із “зависанням”, що полягає у введенні в інтегратор спеціальних “умов стрибків”, які дозволяють інтегратору “вирватись з області зависання”. Однак введення “умов стрибків” в інтегратор складно, бо він, як правило, являє собою завершений модуль.

Проаналізуємо причини згаданого “зависання”.

Умови переходу від однієї лінійної ділянки до іншої мають вигляд

$$\text{if } (Y \geq Y_2) \text{ then } (A_1 \cdot U + B_1) \text{ else } (\text{if } (Y > Y_3) \text{ then } (A_2 \cdot U + B_2) \text{ else } (A_3 \cdot U + B_3)),$$

де A_1, B_1 , A_2, B_2 , A_3, B_3 – параметри верхньої, середньої та нижньої лінійних ділянок апроксимації відповідно.

Нехай зображувальна точка рухається вздовж відрізка $(U_1, Y_1) - (U_2, Y_2)$ під впливом зменшення координати U . У випадку потрапляння в точку (U_2, Y_2) відбувається перемикання з відрізка $(U_1, Y_1) - (U_2, Y_2)$ на відрізок $(U_2, Y_2) - (U_3, Y_3)$. Однак у разі подальшого зменшення координати U вздовж прямої, до якої належить відрізок $(U_2, Y_2) - (U_3, Y_3)$, може бути лише зростання координати Y , що викликає зворотне перемикання на відрізок $(U_1, Y_1) - (U_2, Y_2)$. Циклічне перемикання з відрізка на відрізок при майже незмінному значенні $U \cong U_2$ виглядає як “зависання” інтегратора в точці (U_2, Y_2) .

Тепер зрозуміло, як позбутись “зависання”. Для цього середній відрізок треба провести так, щоб пряма, якій він належить, перетнула пряму $(U2, Y2)-(U4, Y4)$ в точці, ордината якої не перевищує $Y4$. Аналогічні міркування справджуються щодо “зависання” в околі точки $(U3, Y3)$.

На рис.3.9 показано середній відрізок прямої, що проходить через точки $(U5, Y5)$ і $(U6, Y6)$. Для такого середнього відрізка не виникає “зависання” в точках $(U2, Y2)$ і $(U3, Y3)$. У цьому випадку, щоправда, порушується неперервність кусково-лінійної апроксимації. Однак тут вимога неперервності апроксимації має естетичний, а не математичний характер.

Отже, запропонований метод уникнення “зависання” полягає у відмові від неперервної апроксимації. Координати вузлових точок для середнього відрізка апроксимації $(U5, Y5)-(U6, Y6)$ потрібно обирати так, щоб його кут нахилу був менше 90° , однак більший за кут нахилу прямої, що проходить через точки з координатами $(U1, Y1)$, $(U4, Y4)$. Це відповідає виконанню умов $\infty > A2 > (Y1 - Y4) / (U1 - U4)$. Тоді кусково-лінійна апроксимація стає розривною, проте лише за цих умов не виникає “зависання” у разі перекидання тригера.

Зауважимо, що всі точки так збудованого середнього відрізка є стійкими положеннями рівноваги. Стійкі положення рівноваги у внутрішній області нехарактерні для тригера. Однак область цих положень рівноваги можна зробити як завгодно малою, наближаючи параметр $A2$ до $+\infty$.

Завдання для САНОС-ПК, що дозволяє порівняти статистику тригера та макромоделі з кусково-лінійною апроксимацією, наведено у табл.3.4.

Таблиця 3.4. Порівняння статистики тригера та нелінійної підсистеми макромоделі з кусково-лінійною апроксимацією

BEGI

Кусково-лінійна апроксимація статистики тригера

DESC

* Схема тригера ;

$R1(6,4)2.2K$; $R2(6,5)2.2K$; $R3(2,5)8.2K$; $R4(4,3)8.2K$; $V2(R2)$;

$R5(2,7)3.9K$; $R6(1,0)620$; $R7(3,0)3.9K$;

$C1(2,5)1000P$; $C2(4,3)1000P$; $C3(1,0)100n$;

$E1(6,0)12$;

$E2(7,0) IF(T.LE.5)(6-T*2)(-4+(T-5)*2)$; $U=E2$;

$P1(1,2,4)'МП21'$; $P2(1,3,5)'МП21'$;

* Кусково-лінійна розривна апроксимація статистики ;

* Координати точок перемикання ;

$U1=2.5$; $Y1=-1.8$; $U2=-1.4$; $Y2=-2.1$; $U3=2.5$; $Y3=-8.5$; $U4=-1.4$; $Y4=-9.1$;

$U5=0.7$; $Y5=0$; $U6=0.3$; $Y6=-10$; * Точки для відрізка в зоні нестійкості;

* Коефіцієнти кусково-лінійної апроксимації ;

$A1=(Y2-Y1)/(U2-U1)$; $B1=(Y1*U2-Y2*U1)/(U2-U1)$;

$*A2=(Y3-Y2)/(U3-U2)$; $*B2=(Y2*U3-Y3*U2)/(U3-U2)$; * $A2<0$;

$A2=(Y6-Y5)/(U6-U5)$; $B2=(Y5*U6-Y6*U5)/(U6-U5)$; * $(Y1-Y4)/(U1-$

$U4)<A2<\infty$;

$A3=(Y4-Y3)/(U4-U3)$; $B3=(Y3*U4-Y4*U3)/(U4-U3)$;

$YY=A*U+B$; * Рівняння прямої ;


```

Y`=1e6*(YY-Y); * Рівняння, що усуває неоднозначну нелінійність ;
* Умови перемикавання з відрізка на відрізок;
A=if(Y.gt.Y2)(A1)(if(Y.gt.Y3)(A2)(A3));
B=if(Y.gt.Y2)(B1)(if(Y.gt.Y3)(B2)(B3));
* Демонстрація апроксимаційних відрізків ;
ua=(ya-b)/a; ya=-T;
* Умови перемикавання для демонстрації ;
a=if(ya.gt.Y2)(A1)(if(ya.gt.Y3)(A2)(A3));
b=if(ya.gt.Y2)(B1)(if(ya.gt.Y3)(B2)(B3));;
TRAN
TMAX=10; HMIN=1E-10; ERMAX=0.01; allstep;
OUTPUT: V2(E2), Y(U), ya(ua), XMIN=-5, XMAX=7, YMIN=-10, YMAX=0;;
END

```

На рис.3.9 показані відповідні статичні характеристики, отримані за завданням в табл.3.4.

У табл.3.4 є стрічка операторів зі значеннями коефіцієнтів A2 та B2 (вузлові точки (U2,Y2),(U3,Y3)), за яких нелінійна залежність є неперервною, але виникає “зависання”.

Запропонований тут метод усунення “зависання” не вимагає введення до інтегратора умов “стрибків” і дає змогу користуватись звичайними програмами інтегрування диференціальних рівнянь. Це суттєво, бо сучасні інтегратори є завершеними програмними блоками, що ускладнює зовнішні втручання в їхню роботу.

Тепер дещо змінимо структуру макромоделі порівняно з (3.31), щоб мати змогу ідентифікувати лінійну підсистему у разі апроксимації перехідних процесів перекидання тригера.

Нехай у моделі (3.23) нелінійна функція має вигляд $\chi = v - u - F(v)$. Запишемо систему, аналогічну до (3.31):

$$\begin{aligned} a(\lambda)y &= b(\lambda)v, \\ v &= u - F(v). \end{aligned} \quad (3.42)$$

Тут нелінійна функція залежить лише від внутрішнього сигналу v , завдяки чому рекурсія охоплює тільки нелінійний блок.

У статичному режимі (3.42) набуває вигляд:

$$\begin{aligned} y_c &= k_c v_c, \\ v_c &= u_c - F(v_c). \end{aligned} \quad (3.43)$$

Нехай без порушення загальності $k_c=1$. Тоді апроксимація нелінійної функції степеневим поліномом

$$F(v_c) = f_0 + \sum_{j=2}^m f_j v_c^j \quad (3.44)$$

виконується за $M \geq m$ парами значень u_{ck} та y_{ck} при $v_{ck} = y_{ck}$:

$$\sum_{j=0}^m f_j y_{ck}^j = u_{ck}, \quad k = \overline{1, M}, \quad (3.45)$$

де обов'язково $f_1 = 1$. За методом найменших квадратів це відповідає лінійній задачі щодо компонент вектора $\bar{f} = (f_0, f_2, \dots, f_m)$:

$$\min_{\bar{f}} ((f_0 + \sum_{j=2}^m f_j y_{ck}^j) - (u_{ck} - y_{ck}))^2; \quad k = \overline{1, M}. \quad (3.46)$$

Лінійну підсистему $w(\lambda) = b(\lambda)/a(\lambda)$ ідентифікуємо за реакцією системи (3.42) на деяке збурення u , причому внутрішній сигнал v , необхідний для ідентифікації, є вихідним сигналом нелінійного блоку

$$v = u - f_0 - \sum_{j=2}^m f_j v^j, \quad (3.47)$$

параметри якого f_0 та $f_j, j=2, m$ є розв'язком задачі (3.46). Отже, параметри лінійної підсистеми визначаємо, розв'язуючи задачу (2.19) для системи

$$\sum_{i=0}^n a_i y_k^{(i)} = \sum_{i=0}^n b_i v_k^{(i)}, \quad (3.48)$$

де сигнал v знаходимо з нелінійного рівняння (3.47), а його похідні та похідні сигналу y – числовим диференціюванням.

Очевидно, ідентифікація системи (3.48) не залежить від способу апроксимації нелінійної функції $F(v)$, і можна використати всі попередні результати, наприклад, показані на рис.2.7 та 2.8.

Виконаємо ідентифікацію лінійної підсистеми у випадку кусково-лінійної апроксимації нелінійної функції. У табл.3.5 є текст завдання мовою системи САНОС-ПК, що формує дані для ідентифікації системи (3.48) у випадку малих відхилень в околі стійкого положення рівноваги з координатами $(U_0, Y_0) = (3.0, -1.7942)$.

Таблиця 3.5. Завдання для ідентифікації макромоделі тригера при малих відхиленнях від положення рівноваги

BEGI

```

Зняття динамічних характеристик тригера в околі точки (3.0,-1.7942)
DESC
R1(6,4)2.2K; R2(6,5)2.2K; R3(2,5)8.2K; R4(4,3)8.2K;
R5(2,7)3.9K; R6(1,0)620; R7(3,0)3.9K; VR2(R2);
C1(2,5)1n; C2(4,3)1n; C3(1,0)10n;
E1(6,0)12;
E2(7,0) (3.0+0.01*(1-exp(-1e6*T))); tau=1e5;
U=E2-3.0;
P1(1,2,4)'МП21'; P2(1,3,5)'МП21';
Y=VR2+1.7942; t=T*tau;;
TABLE: 'YTAB' 0/0 (500);;
HMAX=1e-9+5e-7*(1-exp(-0.05*1e6*T));;;
TRAN
TMAX=1.01E-4; HMIN=1E-19; ERMAX=0.01; allstep;
OUTPUT: Y(t)'YTAB', XMIN=0,XMAX=1e1,YMIN=0,YMAX=1e-3;;
DESC
-P1; -P2; -C1; -C2; -C3;      * Відімкнення тригера на транзисторах;
U=0.01*(1-exp(-10*T)); Y='YTAB'; * Нові значення входу та виходу;
* Кусково-лінійна апроксимація статки ;
* Координати точок перемикання ;
U1=3;Y1=-1.7942; U2=-1.4;Y2=-1.92; U3=2.5;Y3=-8.5; U4=-1.4;Y4=-9.1;
* Коефіцієнти апроксимації ;
A1=(Y2-Y1)/(U2-U1); A3=(Y4-Y3)/(U4-U3);
V=A*U;      * Рівняння прямої відносно приростів змінних ;
A=if(Y.gt.Y2)(A1)(A3); * Умова перемикання ;
*Обчислення похідних вих.сигн.нелінійної підсистеми V та вих.сигн.тригера
Y;
V21`=DY1; V22`=DY2;
DY1=ky*(Y-V21); DY2=ky*(DY1-V22);
v21`=DV1;
DV1=ky*(V-v21); ky=3e3;
HMAX=4e-4+20e-2*(1-exp(-0.05*10*T));;;
TRAN
TMAX=1e1; HMIN=1E-19; ERMAX=0.01; allstep;
OUTPUT:V,DV1,Y,DY1,DY2,TABL,XMIN=0,XMAX=1e1,YMIN=-1e-3,YMAX=1e-3;;
END

```

Завдання складається з двох підзавдань. У першому прирости вихідної напруги Y записують у внутрішню таблицю з іменем $YTAB$, причому масштаб часу змінений у τ разів. Зміна масштабу часу потрібна для точного обчислення похідних. У другому підзавданні визначають внутрішній сигнал V , який разом із похідними сигналів V та Y записують у робочий файл.

Коефіцієнти лінійної підсистеми другого порядку обчислює програма, аналогічна до наведеної у Додатку 1, яка одночасно готує завдання мовою системи САНОС-ПК для перевірки точності апроксимації лінійної підсистеми.

Складніша ситуація із суттєво нелінійним перехідним процесом у разі перекидання тригера. В рамках структур (3.42) або (3.31) із статичною нелінійною підсистемою принципово неможливо ідентифікувати динамічну нелінійність перекидання тригера, а лінійне наближення буде наперед неточним. Однак іншої можливості цей метод не передбачає. Тому збудовано лінійне наближення перехідних процесів під час перекидання тригера з одного стійкого положення рівноваги в інше та назад. Два коефіцієнти апроксимуючої системи першого порядку знайдені тією ж програмою (Додаток 1). Відповідне завдання наведене в табл.3.6.

Таблиця 3.6. Завдання для ідентифікації лінійної підсистеми у разі перекидання тригера

BEGI

Зняття динамічних характеристик тригера у разі перекидання

DESC

R1(6,4)2.2K; R2(6,5)2.2K; R3(2,5)8.2K; R4(4,3)8.2K;
R5(2,7)3.9K; R6(1,0)620; R7(3,0)3.9K;
C1(2,5)1n; C2(4,3)1n; C3(1,0)10n;
E1(6,0)12; E2(7,0); E2=if(T.lt.0.001)(3)(if(T.lt.0.02)(-2)(3));
P1(1,2,4)'МП21'; P2(1,3,5)'МП21';
VR2(R2); Y=VR2+1.7942; U=E2;;
TABLE: 'YTAB' 0/0 (200);;

TRAN

TMAX=4e-2; HMIN=1E-19; *ERMAX=0.1;
OUTPUT: Y'YTAB', XMAX=4e-2, YMIN=-10, YMAX=0;;

DESC

-P1; -P2; -C1; -C2; -C3; * Відімкнення тригера на транзисторах;
Y='YTAB'; * Нові значення вихідного сигналу;
* Кусково-лінійна розривна апроксимація статички ;
* Координати точок перемикання ;
U1=3; Y1=-1.7942; U2=-1.4; Y2=-1.92; U3=2.5; Y3=-8.5; U4=-1.4; Y4=-9.1;
U5=0.51; Y5=0; U6=0.5; Y6=-10; * Точки для середнього відрізка;
* Коефіцієнти апроксимації ;
A1=(Y2-Y1)/(U2-U1); B1=(Y1*U2-Y2*U1)/(U2-U1);
A2=(Y6-Y5)/(U6-U5); B2=(Y5*U6-Y6*U5)/(U6-U5);
A3=(Y4-Y3)/(U4-U3); B3=(Y3*U4-Y4*U3)/(U4-U3);
v=A*U+B; * Рівняння прямої ;
vv`=1e7*(v-vv); * Рівняння, що перетворює неоднозначну нелінійність;
V=vv+1.79; * в однозначну;
* Умови перемикання для статичної нелінійності ;
A=if(vv.gt.Y2)(A1)(if(vv.gt.Y3)(A2)(A3));
B=if(vv.gt.Y2)(B1)(if(vv.gt.Y3)(B2)(B3));
* Обчислення похідної DY1 вихідного сигналу тригера Y;
vy`=DY1; DY1=ky*(Y-vy); ky=4e3;;;

TRAN

TMAX=4e-2; HMIN=1E-19;
OUTPUT: V, Y, DY1, TABL, XMAX=4e-2, YMIN=-10, YMAX=10;;

END

Оскільки поведінка тригера у випадку малих відхилень та перекидання суттєво різна, то макромодель повинна об'єднувати обидві апроксимації, перемикаючись на вихідний сигнал тієї з них, яка відповідає біжучому стану. Критерієм перемикання лінійної підсистеми обрано значення вихідного сигналу нелінійної підсистеми стосовно вузлових точок Y_2 та Y_3 (рис.3.9). На рис.3.10 зображено блок-схему повної макромоделі тригера.

У табл.3.7 наведено текст завдання, що описує кінцеву макромодель тригера в рамках структури (3.42). Ця макромодель відтворює статику, малосигнальну динаміку та процес перекидання тригера, і містить лише чотири диференціальні рівняння, а не сім, як за схемою тригера.

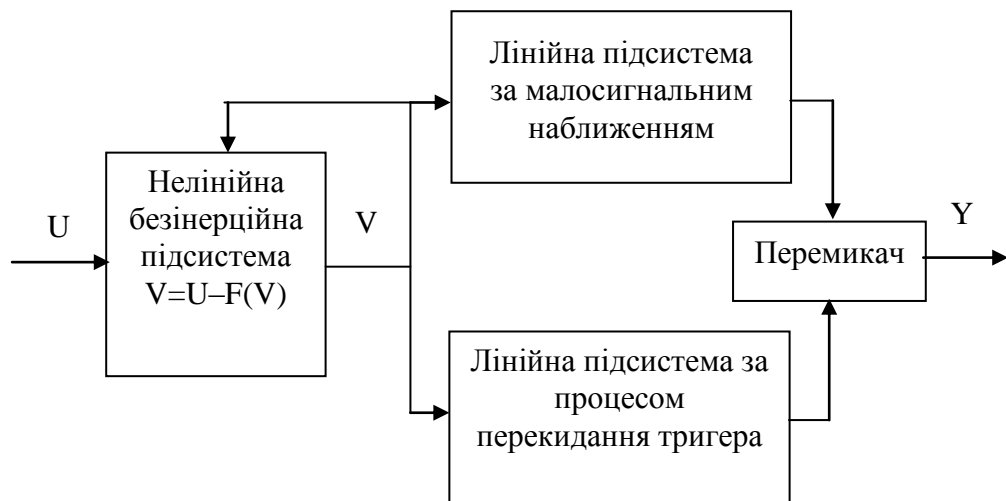


Рис.3.10. Блок-схема повної макромоделі тригера

Таблиця 3.7. Повна макромодель тригера, що відповідає структурі на рис.3.10

BEGI

Перевірка повної макромоделі тригера (перемикання у лінійній підсистемі)

DESC

R1(6,4)2.2K; R2(6,5)2.2K; R3(2,5)8.2K; R4(4,3)8.2K;

R5(2,7)3.9K; R6(1,0)620; R7(3,0)3.9K;

C1(2,5)1n; C2(4,3)1n; C3(1,0)10n;

E1(6,0)12; E2(7,0);

$E2=3.0+0.1*(1-\exp(-1e6*T))$; * Вхідний сигнал малої амплітуди;

P1(1,2,4)'МП21'; P2(1,3,5)'МП21';

VR2(R2); Y=VR2; U=E2;

* Макромодель зі статичною нелінійною та динамічною лінійною підсистемами;

* Кусково-лінійна розривна апроксимація статички;

* Координати точок перемикання ;

```

U1=3;Y1=-1.7942; U2=-1.4;Y2=-1.92; U3=2.5;Y3=-8.5; U4=-1.4;Y4=-9.1;
U5=0.51;Y5=0; U6=0.5;Y6=-10; * Точки для середнього відрізка;
* Коефіцієнти апроксимації ;
A1=(Y2-Y1)/(U2-U1); B1=(Y1*U2-Y2*U1)/(U2-U1);
A2=(Y6-Y5)/(U6-U5); B2=(Y5*U6-Y6*U5)/(U6-U5);
A3=(Y4-Y3)/(U4-U3); B3=(Y3*U4-Y4*U3)/(U4-U3);
v=A*U+B; * Рівняння прямої ;
V`=1e7*(v-V); * Рівняння, що перетворює неоднозначну нелінійність в
однозначну;
* Умови перемикавання для статичної нелінійності ;
A=if(V.gt.Y2)(A1)(if(V.gt.Y3)(A2)(A3));
B=if(V.gt.Y2)(B1)(if(V.gt.Y3)(B2)(B3));
* Динамічна підсистема ;
* Високочастотна модель (малосигнальна); tau=1e5;
b0=-1126.798000*tau*tau; b1=-1350.992000*tau;
a0=-1092.294000*tau*tau; a1=-895.991300*tau;
c0=a0; c1=b1/b0*c0; * Перерахунок коефіцієнтів чисельника для  $k_c=1$  ;
X1`=(-a1*X1 +X2 +c1*V); X2`=(-a0*X1 +c0*V);
* Низькочастотна модель (у разі перекидання тригера);
YN`=2.173e+02*(V-YN);
* Умови перемикавання лінійної підсистеми ;
YA=if(YN.gt.Y2+0.05)(X1)(if(YN.gt.Y3-0.07)(YN)(X1));;;
TRAN
TMAX=1E-4; HMIN=1E-19; ERMAX=0.01; allstep;
OUTPUT: Y, YA, XMIN=0, XMAX=1E-4, YMIN=-1.795, YMAX=-1.79;;
DESC
E2=0.5+3.5*cos(2*3.141592*2.25*T); * Вхідний сигнал великої амплітуди;;;
TRAN
tmax=10; hmin=1e-19; *ermax=0.01; allstep;
output: Y(U),YA(U), xmin=-3,xmax=4,ymin=-10,ymax=-1;;
END

```

На рис.3.11 показаний перехідний процес $\Delta Y(\Delta U)$ у тригері та макромоделі в околі точки $(U_0, Y_0) = (3.0, -1.7942)$ у випадку збурення функцією $\Delta U = 0.1 \cdot (1 - \exp(-1e6 \cdot t))$, яка близька до функції $\Delta U = 0.1 \cdot 1(t)$. Відносна похибка відтворення реакції не перевищує 7%.

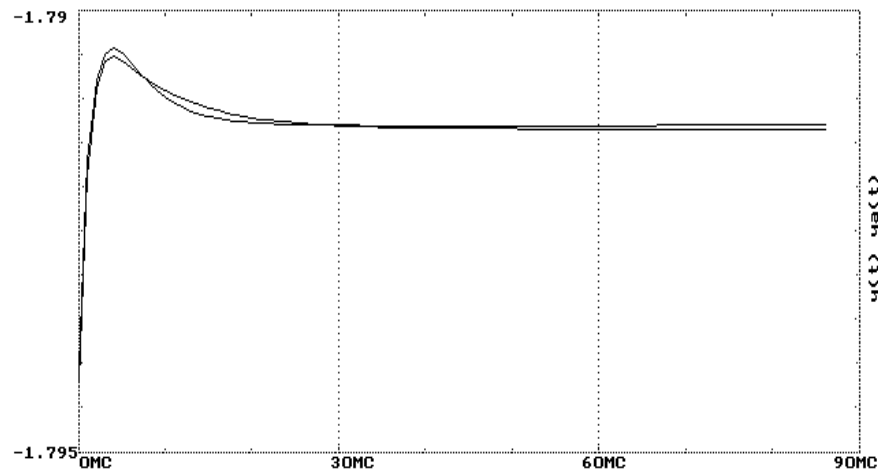


Рис.3.11. Перехідні процеси в тригері та макромоделі у разі малих відхилень від положення рівноваги (3.0, -1.7942)

Реакцію тригера та макромоделі на синусоїдний вхідний сигнал великої амплітуди, що викликає перекидання між положеннями рівноваги, зображено на рис.3.12.

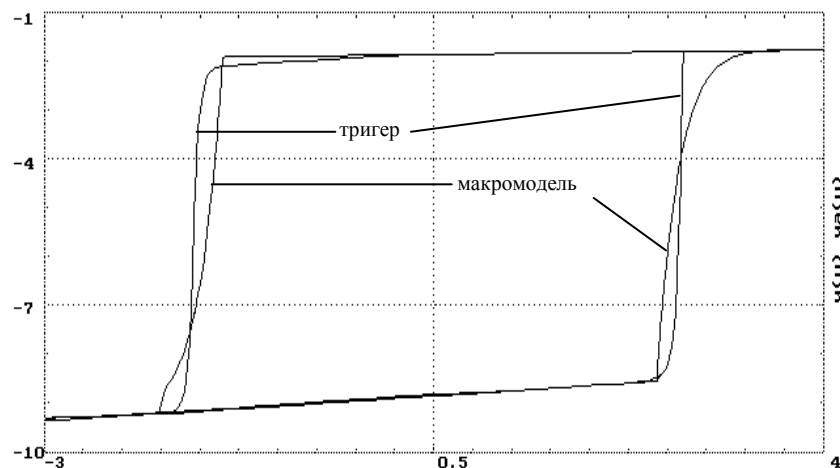


Рис.3.12. Перехідні процеси у випадку перекидання тригера та макромоделі (табл.3.7)

Як було вказано у першому розділі, в рамках методу нелінійної статистики та малосигнальної динаміки не можна врахувати нелінійні динамічні ефекти, а лише статичні. Тому, як і треба було очікувати, перехідний процес у разі перекидання тригера відтворюється лише в загальних рисах.

3.2.3. Ідентифікація нерекурсивних макромоделей.

Макромоделі стабілізатора та операційного підсилювача

Найпростіше виглядає ідентифікація нерекурсивної стаціонарної структури Вінера (рис.3.2):

$$\begin{aligned} b(\lambda)u &= a(\lambda)v, \\ y &= F(v, u). \end{aligned} \quad (3.49)$$

де u, v, y – скаляри. Наведений далі метод відрізняється від відповідного метода з [15] лише характером ідентифікаційних сигналів та узагальненням на багатовимірний випадок.

Нелінійну підсистему апроксимуємо за значеннями сигналів у статичному режимі. Якщо прийняти для статичного коефіцієнта передачі лінійної підсистеми $k_c = b_0/a_0 = 1$, то $v_c = u_c$. Ідентифікацію статичної характеристики виконуємо за співвідношенням $y_c = F(u_c)$.

За методом найменших квадратів апроксимацію степеневим поліномом степеня m у M заданих точках статичної характеристики зведемо до задачі

$$\min_f \left(\sum_{j=1}^m f_j u_{ck}^j - y_{ck} \right)^2; \quad k = \overline{1, M}. \quad (3.50)$$

Лінійну підсистему ідентифікуємо за перехідним процесом $y(t)$, що спричинений малим збурювальним сигналом $u(t)$ в околі деякого положення рівноваги (u_0, y_0) , причому сигнал $v(t)$ обчислюють за співвідношенням $v(t) = y(t)/k_p$, де $k_p = dF/du$ при $u = u_0$. Отже, параметри лінійної підсистеми визначаємо, розв'язуючи задачу (2.19) для системи

$$\sum_{i=0}^n a_i y_k^{(i)} / \tilde{k} = \sum_{i=0}^n b_i u_k^{(i)}, \quad (3.51)$$

Іноді лінійну підсистему зручніше ідентифікувати за задачею (2.19) для системи

$$\sum_{i=0}^n a_i y_k^{(i)} = \sum_{i=0}^n b_i u_k^{(i)}, \quad (3.52)$$

а умову $k_c = 1$ забезпечити переобчисленням коефіцієнтів чисельника передатної функції за формулами $c_i = b_i/b_0 * a_0$, $i = \overline{0, n}$. Тоді рівняння лінійної підсистеми має такий вигляд:

$$\sum_{i=0}^n a_i y_k^{(i)} = \sum_{i=0}^n c_i u_k^{(i)}, \quad (3.53)$$

Структуру Вінера доцільно застосовувати для моделювання підсилювачів з нескладними статичними нелінійностями. Опишемо можливості цієї структури на прикладі макромоделі компенсаційного

стабілізатора напруги (рис 3.13), опис якого мовою САНОС-ПК наведено в табл.3.8.

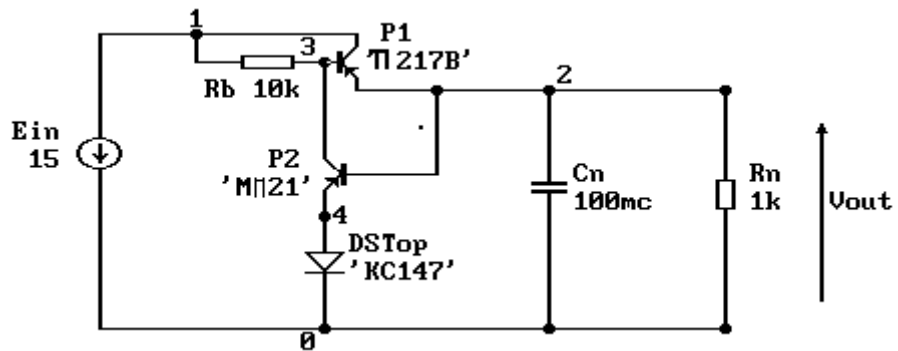


Рис.3.13. Схема компенсаційного стабілізатора напруги

Таблиця 3.8. Завдання для аналізу статичної характеристики та перехідного процесу стабілізатора напруги

BEGI

Компенсаційний стабілізатор напруги на PNP-транзисторах

DESC * Схема стабілізатора;

Ein(1,0)15; * Вхідна напруга ;
 Pcon(2,3,1)'П217В'; * Регулювальний транзистор ;
 Pamp(4,2,3)'МП21'; Rb(1,3)10k; * Підсилювальний транзистор ;
 DSTop(4,0)'КС147'; * Стабілітрон ;
 Rn(0,2)1k; Cn(0,2)100mc; * Навантаження ;
 Vout(0,2); * Вихідна напруга ;;;

DC * Аналіз статичної характеристики ;

Ein=A(15,0.5)50; * Від 15в до 0.5в 50 точок ;
 OUTPUT: Vout(Ein), tabl, xmax=15, ymin=0, ymax=10;;

DESC * Доповнення опису схеми ;

Ein=10+(1-exp(-1e3*T)); * Стрибок величиною 1в ;
 U=Ein-10; Y=Vout-5.4352; * Прирости на вході та виході ;
 * Диференціювання вхідного та вихідного сигналів ;
 V21`=DY1; V22`=DY2; u21`=DU1; ky=1e4;
 DY1=ky*(Y-V21); DY2=ky*(DY1-V22); DU1=ky*(U-u21);;;

TRAN * Аналіз перехідного процесу ;

TMAX=0.2;
 OUTPUT: U, DU1, Y, DY1, DY2, tabl, xmax=0.2, ymin=0, ymax=0.03;;

END

Виконавши завдання з табл.3.8, система САНОС-ПК формує у робочому файлі числові масиви, що відповідають статичній характеристиці та перехідному процесу з похідними вхідного та вихідного сигналів. Статичну нелінійність зручно апроксимувати кусково-лінійною залежністю. Коефіцієнти апроксимації динамічної лінійної підсистеми другого порядку отримано за допомогою програми, що міститься у Додатку 1.

Процедура ідентифікації досить тривіальна, тому наведемо одразу остаточні рівняння макромоделі.

Рівняння лінійної підсистеми, апроксимаційні коефіцієнти якої перераховано для забезпечення одиничної передачі у статиці, мають вигляд:

$$X1' = (-a1 * X1 + X2 + a0 * Ein); \quad X2' = (-a0 * X1 + b1 / b0 * a0 * Ein);$$

$$a0 = 38440; \quad a1 = 69.86; \quad b0 = 733.3; \quad b1 = 0.5629;$$

Рівняння нелінійної функції такі (вказані вузлові точки кусково-лінійної апроксимації $(U1, Y1)$, $(U2, Y2)$, $(U3, Y3)$):

$$Ym = A * X1 + B;$$

$$A = \text{if}(X1 > U2) \text{ then } (Y2 - Y1) / (U2 - U1) \text{ else } (Y3 - Y2) / (U3 - U2);$$

$$B = \text{if}(X1 > U2) \text{ then } (Y1 * U2 - Y2 * U1) / (U2 - U1) \text{ else } (Y2 * U3 - Y3 * U2) / (U3 - U2);$$

$$U1 = 10; \quad Y1 = 5.4352; \quad U2 = 6.7; \quad Y2 = 5.372; \quad U3 = 0.5; \quad Y3 = 0.1.$$

У табл.3.9 наведено текст завдання мовою САНОС-ПК, що дає змогу порівняти статику і динаміку повної схеми стабілізатора та її макромоделі. Результати порівняння зображені на рис.3.14 та 3.15.

Таблиця 3.9. Завдання для порівняння статики і динаміки стабілізатора та макромоделі

BEGI

Макромодель компенсаційного стабілізатора напруги

DESC * Описи схеми стабілізатора та макромоделі ;

* Схема стабілізатора ;

Ein(1,0)15; * Вхідна напруга ;

Pcon(2,3,1)'П2176'; Pamp(4,2,3)'МП21'; Rb(1,3)10k;

DSTop(4,0)'KC147'; Rn(0,2)1k; Cn(Rn)100mc;

Vout(Rn); * Вихідна напруга ;

* Макромодель за структурою Вінера ;

* Лінійна підсистема ;

$X1' = (-a1 * X1 + X2 + c1 * Ein); \quad X2' = (-a0 * X1 + c0 * Ein); \quad V = X1;$

* Коефіцієнти апроксимації; $b0 = 733.3; \quad b1 = 0.5629; \quad a0 = 38440; \quad a1 = 69.86;$

* Переобчислення коефіцієнтів чисельника для забезпечення $Kc = 1$;

$c0 = a0; \quad c1 = b1 / b0 * a0;$

* Нелінійна підсистема з кусково-лінійною апроксимацією ;

$Ym = A * V + B;$ * Рівняння прямої ;

* Умови перемикавання для статичної нелінійності ;

$A = \text{if}(V.gt.U2)(A1)(A2); \quad B = \text{if}(V.gt.U2)(B1)(B2);$

* Коефіцієнти апроксимації ;

$A1 = (Y2 - Y1) / (U2 - U1); \quad B1 = (Y1 * U2 - Y2 * U1) / (U2 - U1);$

$A2 = (Y3 - Y2) / (U3 - U2); \quad B2 = (Y2 * U3 - Y3 * U2) / (U3 - U2);$

* Координати вузлів апроксимації ;

$U1 = 10; \quad Y1 = 5.4352; \quad U2 = 6.7; \quad Y2 = 5.372; \quad U3 = 0.5; \quad Y3 = 0.1;;;$

DC * Аналіз статичної характеристики ;

Ein=A(15,0.5)50; * Від 15в до 0.5в 50 точок ;

```

OUTPUT: Vout(Ein), Ym(Ein), xmax=15,ymin=0,ymax=10;;
DESC * Доповнення опису схеми ;
Ein=10+(1-exp(-1e3*T)); * Стрибок вхідної напруги на 1в ;
U=Ein-10; Y=Vout-5.4352; Yp=Ym-5.4352; * Прирости на вході та на
виході;;;
TRAN * Аналіз перехідного процесу ;
TMAX=0.2; OUTPUT: U,Y,Yp, xmax=0.2,ymin=0,ymax=0.03;;
END

```

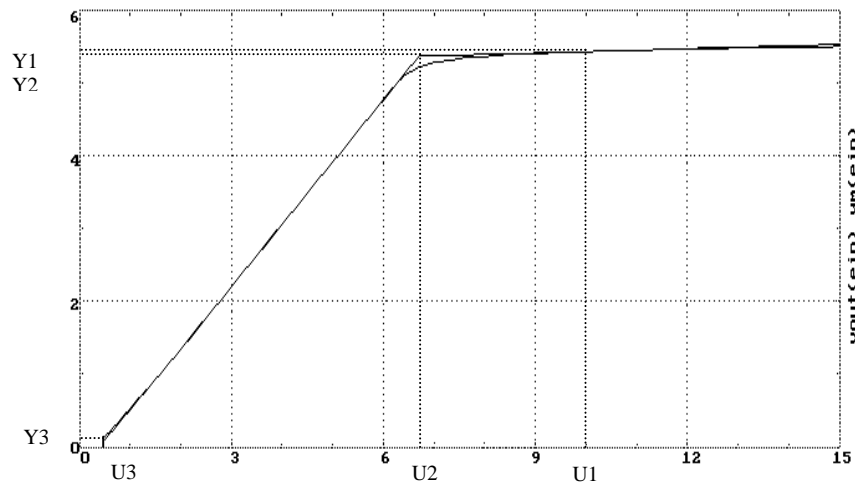


Рис.3.14. Статичні характеристики стабілізатора та макромоделі з вузлами кусково-лінійної апроксимації

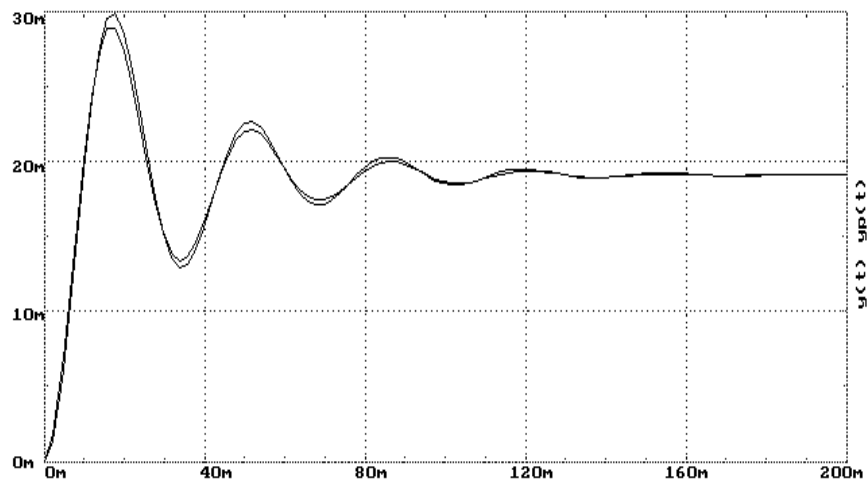


Рис.3.15. Перехідні процеси в стабілізаторі та макромоделі

Відносна похибка відтворення не перевищує 4% для статичних характеристик та 7% для перехідних процесів. Однак макромодель містить лише два диференціальні рівняння, а не шість, як у повній схемі стабілізатора на рис.3.13.

Усі описані вище методи ідентифікації узагальнені для багатовимірних задач. Розглянемо багатовимірну макромодель операційного підсилювача (ОП), побудовану за структурою Вінера.

На рис.3.16 зображена принципова схема ОП 140УД1. В якості моделей інтегральних транзисторів використані близькі до них моделі дискретних транзисторів KT312B.

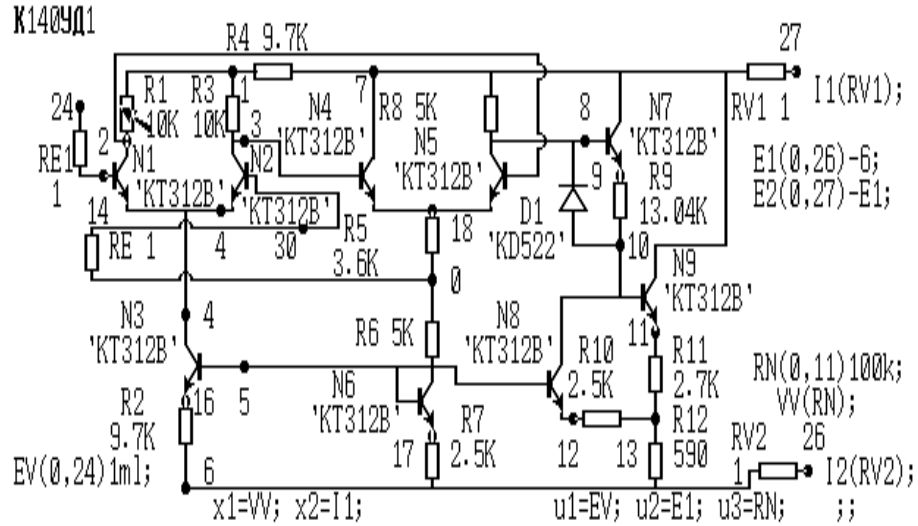


Рис.3.16. Принципова схема операційного підсилювача

Макромодель синтезували за трьома вхідними сигналами та двома вихідними. Відповідно до позначень на рис.3.16, компонентами вхідного вектора U було обрано вхідну напругу $EV=U_1$, напругу джерела живлення $E1=U_2$ та опір навантаження $RN=U_3$, а компонентами вихідного вектора Z – вихідну напругу $VV=Z_1$ та струм живлення $I1=Z_2$. Суттєвою нелінійністю було визнано залежність вихідної напруги від вхідної та від опору навантаження: $Z_1=Z_1(U_1, U_3)$.

Загальні рівняння макромоделі, відповідно до структури на рис.3.2, мають такий вигляд:

$$\begin{aligned}
 X_1' &= X_3; \quad X_3' = \sum_{i=1}^4 A_{1i} X_i + \sum_{i=1}^9 B_{1i} U_i; \\
 X_2' &= X_4; \quad X_4' = \sum_{i=1}^4 A_{2i} X_i + \sum_{i=1}^9 B_{2i} U_i; \\
 U_1' &= U_4; \quad U_4' = U_7; \quad U_2' = U_5; \quad U_5' = U_8; \quad U_3' = U_6; \quad U_6' = U_9; \\
 Y_1 &= F_1(X_1); \quad Y_4 = F_2(U_3); \quad Z_1 = \sum_{i,j=0}^5 C_{ij} Y_1^i Y_4^j; \quad Z_2 = X_2.
 \end{aligned}$$

Повний опис макромоделі мовою САНОС-ПК наведено в табл.3.10.

Таблиця 3.10. Макромодель операційного підсилювача

BEGI

Апроксимаційна макромодель ОП. Перевірка за великим сигналом

DESC

```

* Лінійні підсистеми ;
X1`=X5; X5`=X9; X2`=X6; X6`=X10;
* Перша підсистема ;
A11=-5.575E+14; A12=-1.812E+18; A14=-1.980E+07; A15= 0.000E+00;
B11=-9.468E+16; B12=-9.210E+14; B13=-1.351E+07;
B14= 6.263E+09; B15= 8375105.0; B16= -1.295;
B17= -89.62; B18=-2.505E-01; B19= 5.127E-08;
X9=A11*X1+A12*X2+A14*X5+A15*X6+B11*U1+B12*U2+B13*U3+B14*U5
+
B15*U6+B16*U7+B17*U8+B18*U9+B19*U10;
* Друга підсистема ;
A21= 0.000E+00; A22=-2.521E+13; A24= 0.000E+00; A25=-582557.00;
B21= 6.705E+12; B22=-1.281E+10; B23= 1949.935;
B24= 127699.10; B25= -2310.955; B26= 1.829E-04;
B27=-2.326E-02; B28=-4.865E-04; B29= 3.600E-13;
X10=A21*X1+A22*X2+A24*X5+A25*X6+B21*U1+B22*U2+B23*U3+
B24*U5+B25*U6+B26*U7+B27*U8+B28*U9+B29*U10;
* Малі збурення, за якими ідентифіковані лінійні підсистеми ;
* U1=IF(T.gt.0.1E-6)(IF(T.GT.1E-6)(0.0)(0.00005))(0.0);
* U2=IF(T.LT.4E-6)(-6)(if(T.ge.5E-6)(-6)(-5.94));
* U3=if(T.ge.2E-6)(IF(T.ge.3E-6)(1k)(10k))(1k) ;
* Диференціювання (двічі) вхідного вектора ;
E1(0,50)U1; R1(50,51)1; C1(51,0)KD; I1(R1); U5=I1/C1; KD=4E-8;
E4(0,80)U2; R4(80,81)1; C4(81,0)KD; I4(R4); U6=I4/C4;
E5(0,90)U3; R5(90,91)1; C5(91,0)KD; I5(R5); U7=I5/C5;
E8(0,100)U5; R8(100,101)1; C8(101,0)KD; I8(R8); U8=I8/C8;
E9(0,110)U6; R9(110,111)1; C9(111,0)KD; I9(R9); U9=I9/C9;
E10(0,120)U7; R10(120,121)1; C10(121,0)KD; I10(R10); U10=I10/C10;
* Нелінійна функція ;
Y1=if(abs(X1).lt.3.714)(1.008*X1+0.092)(if(X1.lt.-2.4)(-3.834-0.564)(3.834));
Y4=-1.136+3.186*exp(-0.129*log(U3)); Y2=Y1*Y1*Y1*Y1;
Y5=Y4*Y4*Y4*Y4;
FN=F0 +F1*Y1 +F2*Y4 +F3*Y1*Y1 +F4*Y4*Y4 +F5*Y1*Y4+F6*Y1*Y1*Y1
+F7*Y4*Y4*Y4+F8*Y1*Y1*Y4+F9*Y1*Y4*Y4+F10*Y2+F11*Y5+F12*Y1*Y1*Y1*
Y4
+F13*Y1*Y1*Y4*Y4 +F14*Y1*Y4*Y4*Y4 +F15*Y1*Y2 +F16*Y4*Y5 +F17*Y2*Y4
+F18*Y1*Y1*Y1*Y4*Y4 +F19*Y1*Y1*Y4*Y4*Y4 +F20*Y1*Y5;
F0=-1.118E-01; F1= 1.117; F2=-1.580; F3= 4.799E-03; F4= 1.341; F5=-1.430;
F6=-2.309E-02; F7= 9.214; F8=-1.206E-01; F9=-8.623E-01; F10=-1.880E-03;
F11=-14.435; F12= 8.164E-03; F13= 1.102E-01; F14= 2.319; F15= 2.957E-04;
F16= 5.637; F17= 3.105E-03; F18= 8.060E-03; F19=-2.687E-02; F20=-1.048;
Z1=FN; Z2=X2; * Нелінійний вихід ;
U1=10ml*sin(2*3.141592*1E5*T); U2=-6; * Збурення ;
U3='Rntab'; TABLE:'Rntab' 0/100K,0.3e-4/10,1e-4/10k;;;

```

TRAN

TMAX=1e-4; HS=5E-7; HMIN=1E-15;

OUTPUT: X1, Z1, U1, U3, XMIN=0,XMAX=1E-4,YMIN=-6,YMAX=6;;

END

Лінійні підсистеми ідентифіковано за малими стрибками вхідних сигналів в околах значень $U1=0$, $U2=6$, $U3=1k$, як це описано в завданні табл.3.10.

Безпосередня ідентифікація функції $Z1(U1,U3)$ як багатовимірного степеневого полінома від $U1$ та $U3$ виявилась дуже складною навіть з урахуванням змінних $X1$ та $X2$. Тому до апроксимаційного базису для функції $Z1(U1,U3)$ було залучено апроксимації статичних залежностей $Z1(X1)$ при $U3=1k$ та $Z1(U3)$ при $X1=0$. Загальний опис такого методу вибору апроксимаційного базису є у параграфі 4.2.

Апроксимаційні вирази для функцій $Y1=Z1(X1)$ та $Y4=Z1(U3)$ наведено у табл.3.10. Функція $Y1=Z1(X1)$ апроксимована кусково–лінійною залежністю з трьома лінійними ділянками. Для функції $Y4=Z1(U3)$ найкраще підійшла степенева залежність. Відповідно, функція $Z1(U1,U3)$ ідентифікована як двовимірний степеневий поліном п'ятого степеня від функцій $Y1=Z1(X1)$ та $Y4=Z1(U3)$.

На рис.3.17 показані графіки вхідних та вихідних сигналів макромоделі ОП відповідно до опису у табл.3.10.

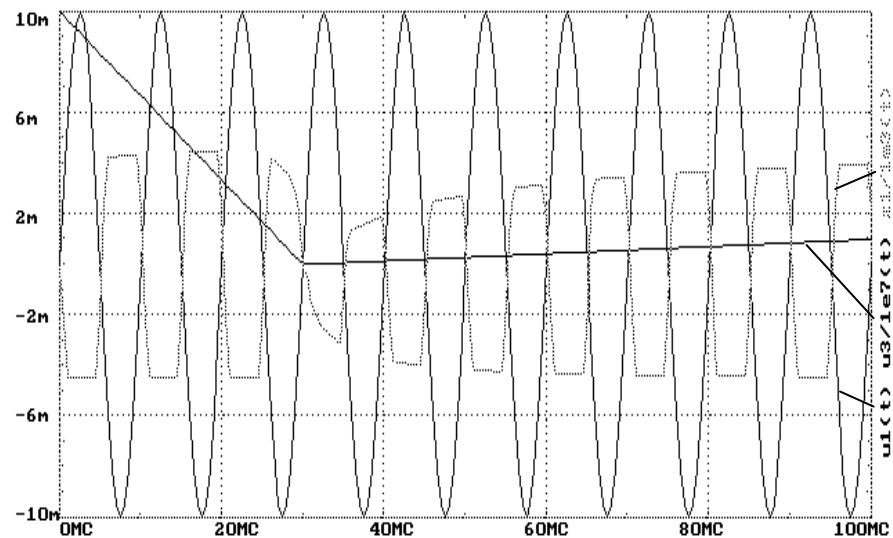


Рис.3.17. Вхідні та вихідні сигнали математичної макромоделі ОП

Вхідна напруга є гармонічним сигналом великої амплітуди, при якій суттєво виявляються нелінійні спотворення. Опір навантаження змінюється кусково–лінійно з різною швидкістю в діапазоні від 100 к Ω до 10 Ω . Відносна похибка відтворення реакцій повної схеми ОП на такі ж збурення не перевищує 11%. У цьому випадку макромодель є системою лише 8-го порядку, тоді як повна схема відповідає системі 19-го порядку.

Описану макромодель порівняли із макромоделлю, створеною методом структурно-функціонального синтезу та описаною у параграфі 1.3.2, (див. рис.1.7). Реакції останньої макромоделі на ті ж вхідні сигнали відображені на рис.3.18. Максимальна відносна похибка відтворення вихідних сигналів повної схеми ОП досягає 56%.

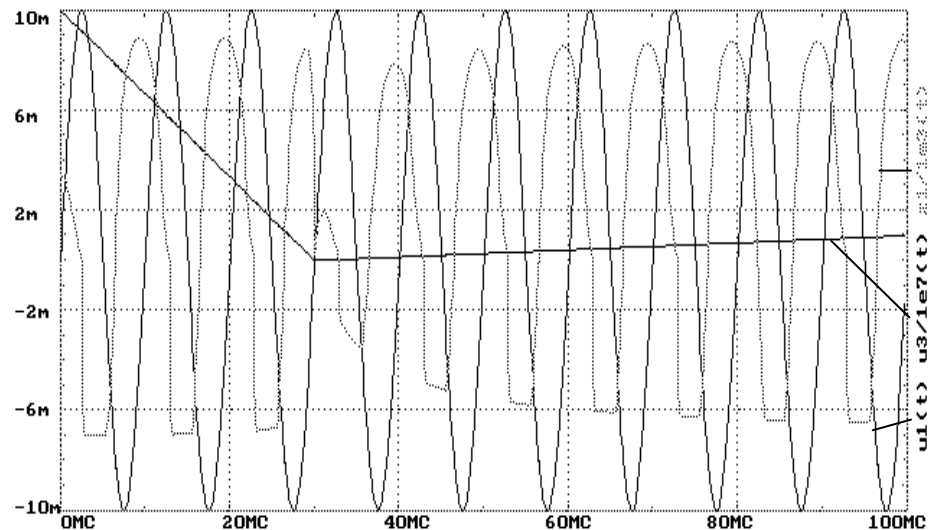


Рис.3.18. Вхідні та вихідні сигнали структурно-функціональної макромоделі ОП

Час розрахунку зазначених перехідних процесів для функціональної макромоделі становив лише 3% від часу розрахунку для повної схеми ОП, і 10% – для математичної макромоделі з табл.3.10.

Водночас перевірка макромоделей у режимах, близьких до лінійних, засвідчила цілком задовільну точність. Частотні характеристики фільтра низьких частот, зображеного на рис.3.19, відрізняються не більше ніж на 7%, якщо ОП замінювати повною схемою (див. рис.3.16), функціональною (див. рис.1.7) та математичною макромоделями (див. табл.3.10). Час аналізу у разі заміщення макромоделями зменшується на порядок.

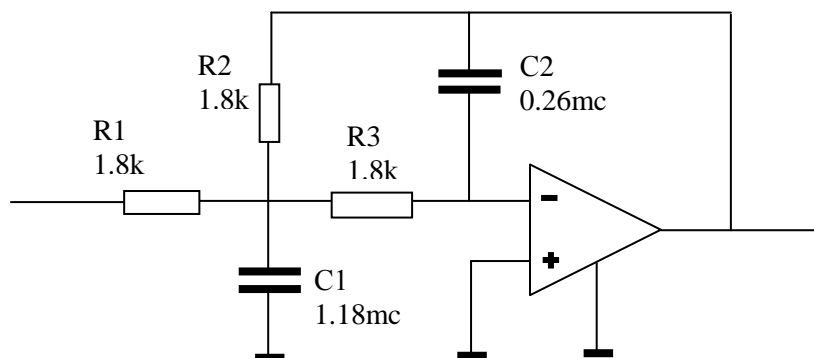


Рис.3.19. Активний фільтр на ОП

Отже, хоча функціональна макромодель (див. рис.1.7) компактніша, ніж математична (див. табл.3.10), однак за точністю відтворення нелінійних ефектів перша значно поступається другій.

З огляду на сказане можна зробити такий висновок: основний недолік методу нелінійної статички і малосигнальної динаміки той, що нелінійності повинні міститись лише у статичних характеристиках, тобто для ідентифікації нелінійностей повинно бути достатньо вимірювань у статичному режимі. Це різко обмежує клас систем, що можуть бути ідентифіковані за цим методом. Зокрема, для електричних кіл всі нелінійності повинні бути омичного (резистивного) характеру і мати гальванічні зв'язки (за постійним струмом) з входами та виходами кола. Очевидно, така ситуація можлива далеко не завжди. Результати ідентифікації макромоделі тригера, наведені вище, підтверджують цей висновок. Тонкі деталі динаміки перекидання тригера не вдалось змодельовати в рамках прийнятого методу ідентифікації макромоделі.

Спроби вдосконалити метод, наприклад, виконувати ідентифікацію ще й на сталих значеннях перших похідних вхідних сигналів, призводять до значних ускладнень методу і, як наслідок, зростання похибок. Нам невідомі успішні спроби таких удосконалень.

Однак для визначеного класу систем метод нелінійної статички та малосигнальної динаміки дає порівняно стійкі й легко досяжні результати.

Названих недоліків позбавлений метод оберненої лінійної підсистеми, розглянутий нижче.

3.3. ІДЕНТИФІКАЦІЯ ЗА МЕТОДОМ ОБЕРНЕНОЇ ЛІНІЙНОЇ ПІДСИСТЕМИ

3.3.1. Загальні положення

Основна ідея методу полягає у визначенні внутрішнього вектора \bar{v} шляхом обчислення реакції оберненої лінійної підсистеми на сигнал \bar{u} як вхідний [56].

Нехай у структурі (3.11), зображеній на рис.3.2, матриця передатних функцій лінійної підсистеми $W(\lambda)$ є вже визначеною або за малосигнальними випробуваннями, або за апріорною інформацією. Зазначимо

лише, що вдалий вибір лінійної підсистеми може значно спростити подальшу ідентифікацію нелінійної вектор-функції багатьох аргументів $\bar{v} = \bar{\chi}(\bar{y}, \bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}, \bar{v}^{(1)}, \dots, \bar{v}^{(k-2)}, \bar{u}, \dot{\bar{u}}, t)$.

Остання задача зводиться до звичайної апроксимаційної задачі функції багатьох аргументів, якщо відомі вектори \bar{u} , \bar{y} , \bar{v} та їхні похідні. Вектори \bar{u} та \bar{y} ми можемо отримати безпосередньо з експерименту, досліджуючи систему, яка моделюється. Натомість вектор \bar{v} не підлягає вимірюванню в модельованій системі, бо його в ній просто не

існує. Опишемо метод, що дає змогу обчислити вектор \bar{v} за відомим вектором \bar{y} , якщо відома матриця $W(\lambda)$.

У теорії лінійних систем є поняття системи, оберненої до заданої [3]. Це така система $W(\lambda)^{-1}$, для якої при $\bar{y} = W(\lambda)\bar{v}$ виконується $\bar{v} = W(\lambda)^{-1}\bar{y}$ з відповідним перерахунком початкових умов. Інакше кажучи, для оберненої системи вхід та вихід міняються місцями порівняно з даною системою.

Обчислення оберненої підсистеми в загальному випадку є складним завданням. Однак для лінійних стаціонарних систем із зосередженими параметрами елементи матриці $W(\lambda) = \{W_{ij}(\lambda)\}; i=1, \dots, n; j=1, \dots, m$ є дробово-раціональними функціями зі сталими коефіцієнтами й обернену систему обчислюють найпростіше.

Зокрема, якщо \bar{y} та \bar{v} – скаляри, то $W(\lambda)^{-1} = 1/W(\lambda)$. Умови фізичної реалізованості не враховані, оскільки йдеться лише про математичне обчислення сигналу \bar{v} . Якщо матриця $W(\lambda)$ квадратна, то матриця оберненої підсистеми обчислюється її обертанням як матриці з символьними елементами [3].

Складніше ситуація для прямокутної матриці, коли $n \neq m$. Тоді треба скористатись одним із означень оберненої прямокутної матриці, залежно від обраної метрики. Зокрема, для середньоквадратичної метрики $W(\lambda)^{-1}$ співпадає з псевдооберненою матрицею $W(\lambda)^+$ [57, с.32].

Отже, метод оберненої лінійної підсистеми складається з послідовності таких дій:

- побудова лінійної підсистеми $W(\lambda)$ за результатами експериментального вивчення системи або за апіорною інформацією;
 - побудова оберненої лінійної підсистеми $W(\lambda)^{-1}$;
 - обчислення внутрішнього вектора $\bar{v} = W(\lambda)^{-1}\bar{y}$;
 - апроксимація нелінійної вектор-функції
- $$\bar{v} = \bar{\chi}(\bar{y}, \bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}, \bar{v}^{(1)}, \dots, \bar{v}^{(k-2)}, \bar{u}, \dot{\bar{u}}, t).$$

Привабливість методу полягає в апроксимації нелінійного функціонала (1.1) лінійним функціоналом $\bar{y} = W(\lambda)\bar{v}$ та нелінійною вектор-функцією $\bar{v} = \bar{\chi}(\bar{y}, \bar{y}^{(1)}, \dots, \bar{y}^{(k-1)}, \bar{v}, \bar{v}^{(1)}, \dots, \bar{v}^{(k-2)}, \bar{u}, \dot{\bar{u}}, t)$. Крім того, вектор-функція $\bar{\chi}(\cdot)$ має в загальному випадку повний список аргументів, на відміну від нелінійної функції у попередньому методі.

3.3.2. Макромодель генератора майже гармонічного сигналу

Застосуємо метод оберненої лінійної підсистеми до побудови макромоделі генератора сигналу, близького до гармонічного.

Спочатку виберемо лінійну підсистему. Її можна вибрати у вигляді консервативної лінійної системи з комплексними власними числами, що попарно відповідають відомим частотним складовим сигналам модельованого генератора. Тоді, з огляду на вимоги перехідного процесу, потрібної форми

сигналу, стійкості граничного циклу або інші, залишається підібрати нелінійну вектор-функцію.

Аналітичний метод реалізації такого підходу, поширений на автогенератори з несумірними власними частотами, описаний у [20]. У праці [8] запропонований метод числового синтезу, що ґрунтується на апроксимації багатовимірним степеневим поліномом за мінімакним критерієм правої частини системи диференціальних рівнянь у формі Коші (3.24) з подальшою стабілізацією періодичних рішень спеціальними функціями, створеними за допомогою ідей другого методу Ляпунова.

Збудуємо макромодель перехідного процесу в генераторі сигналу, близького до гармонічного, користуючись методом оберненої лінійної підсистеми. Схему RC-генератора на одному транзисторі з трьома фазоповертальними ланками, що слугував об'єктом моделювання, зображено на рис.3.20 та описано у табл.3.11.

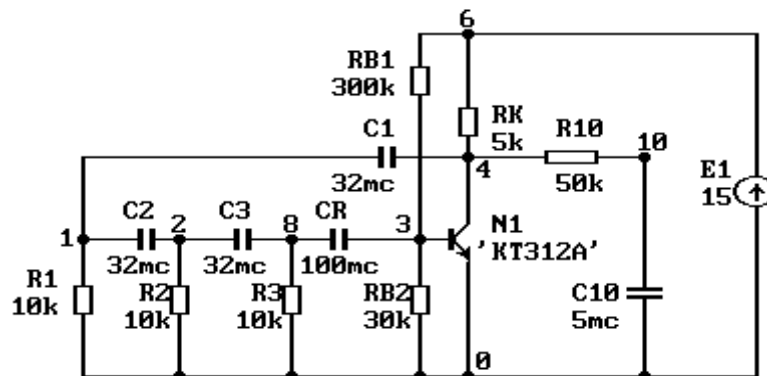


Рис.3.20. Схема RC-генератора

Таблиця 3.11. Опис схеми RC-генератора та макромоделі

BEGI

Макромодельовання RC-генератора

DESC

* Схема RC-генератора ;

R1(1,0)10k; R2(2,0)10k; R3(8,0)10k;

* Фазоповертальні ;

C1(4,1)32mc; C2(2,1)32mc; C3(2,8)32mc;

* ланки ;

RK(4,6)5k; RB(6,3)300k; RB1(3,0)30k;

E1(0,6)15; CR(8,3)100mc; N1(0,3,4)'KT312A';

VOUT(10,0); C10(10,0)5mc; R10(4,10)50k; * Фільтр НЧ;

* Макромодель ;

Y`=Y1; Y1`=FN-W*W*Y; W=2.0924;

FN=29.31+2.153*Y-0.2633*Y1-0.1114*Y*Y-

0.008486*Y1*Y1+0.1487*Y*Y1-0.0008418*Y*Y*Y-0.001248*Y1*Y1*Y1-

0.003446*Y*Y1*Y1-0.01128*Y1*Y*Y;

INIT: Y1(8), C3(6);;

TRAN

TMAX=25; INIT; HS=0.05; HMIN=1E-12;

OUTPUT: VOUT, Y, TABL, XMAX=25, YMIN=-2, YMAX=18;;

END

За вихідним сигналом генератора V_{OUT} (напруга між вузлами 10 і 0 на рис.3.20) визначена частота встановлених коливань після завершення перехідного процесу. Цей параметр є єдиним для лінійної консервативної підсистеми другого порядку $\ddot{y} + \omega^2 y = v$. Для обчислення внутрішнього сигналу v як вихідного оберненої підсистеми, послідовним диференціюванням (3.20) знайдено другу похідну \ddot{y} . За m синхронними відліками сигналів y, \dot{y}, \ddot{y} та $v = \ddot{y} + \omega^2 y$ розв'язано апроксимаційну задачу

$$\min_a \sum_{i=1}^m (v_i - (a_0 + a_1 y_i + a_2 \dot{y}_i + a_3 y_i^2 + a_4 \dot{y}_i^2 + a_5 y_i \dot{y}_i + a_6 y_i^3 + a_7 \dot{y}_i^3 + a_8 y_i \dot{y}_i^2 + a_9 y_i^2 \dot{y}_i))^2$$

за допомогою програми, зразок якої наведено у Додатку 1.

Для $m=300$ отримано такі значення коефіцієнтів:

$$a_0=29.31; a_1=2.153; a_2=-0.2633; a_3=-0.1114; a_4=-0.008486; a_5=0.1487; \\ a_6=-0.0008418; a_7=-0.001248; a_8=-0.003446; a_9=-0.01128.$$

Загальні рівняння макромоделі мають вигляд

$$\dot{y} = y_1; \quad \dot{y}_1 = -\omega^2 y + v; \quad v = f(y, y_1);$$

$$f(y, y_1) = a_0 + a_1 y + a_2 y_1 + a_3 y^2 + a_4 \dot{y}_1^2 + a_5 y y_1 + a_6 y^3 + a_7 y_1^3 + a_8 y y_1^2 + a_9 y^2 y_1,$$

що відповідає блок-схемі на рис.3.21.

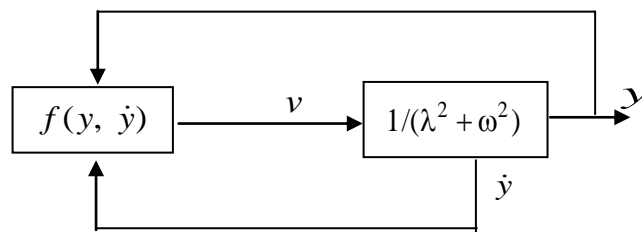


Рис.3.21. Блок-схема макромоделі RC-генератора

Очевидно, блок-схема на рис.3.21 є окремим випадком загальної блок-схеми з рис.3.2.

На рис.3.22 показано перехідний процес у генераторі та в макромоделі згідно із завданням у табл.3.11.

Встановлений режим відтворюється з максимальною відносною похибкою 3%. Помітну похибку відтворення перехідного процесу пояснюють неможливістю адекватного перерахунку початкових умов генератора та макромоделі. Адже схема RC-генератора містить сім диференціальних рівнянь, тоді як макромодель – лише два.

Отримана макромодель демонструє збіжність до встановленого періодичного режиму, якщо початкові умови є у межах: $y(0) \in [-25, 10^7]$; $y_1(0) \in [-10^7, 10^7]$.

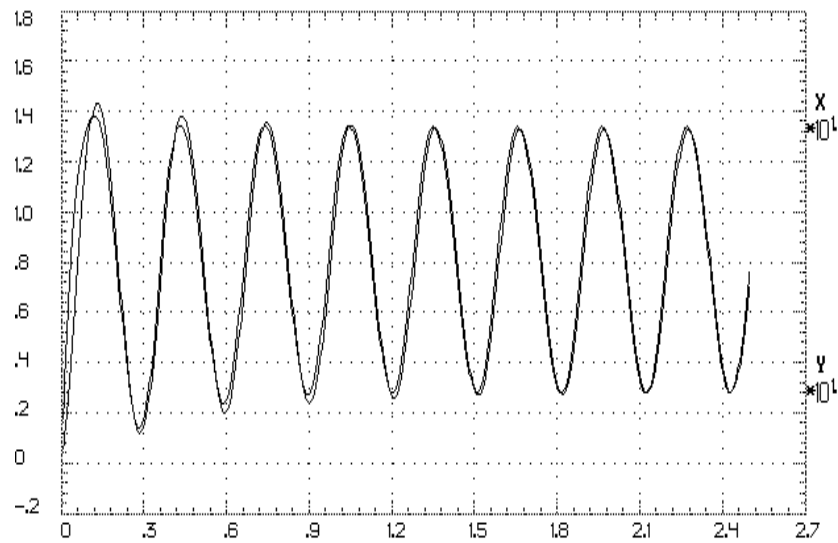


Рис.3.22. Перехідні процеси в RC-генераторі та макромоделі

3.3.3. Макромодель мультивібратора

Продemonструємо метод оберненої лінійної підсистеми на прикладі синтезу математичної моделі автогенератора з неоднозначною нелінійністю [58].

Як об'єкт макромодельовання візьмемо простий двотранзисторний мультивібратор (рис.3.23).

Макромодельовання виконаємо для сигналу мультивібратора $y(t) = V5(t) + 10$, де $V5$ – напруга на ємності $C5$. Форма сигналу $y(t)$ показана на рис.3.24.

Шукатимемо нелінійну функцію у формі, близькій до релейної залежності з гістерезисом. Тоді вихідний сигнал нелінійної функції $v(t)$, тобто вхідний сигнал лінійної підсистеми, повинен бути близьким до прямокутних коливань з основною частотою, що збігається з основною частотою заданих коливань $y(t)$ (рис.3.24).

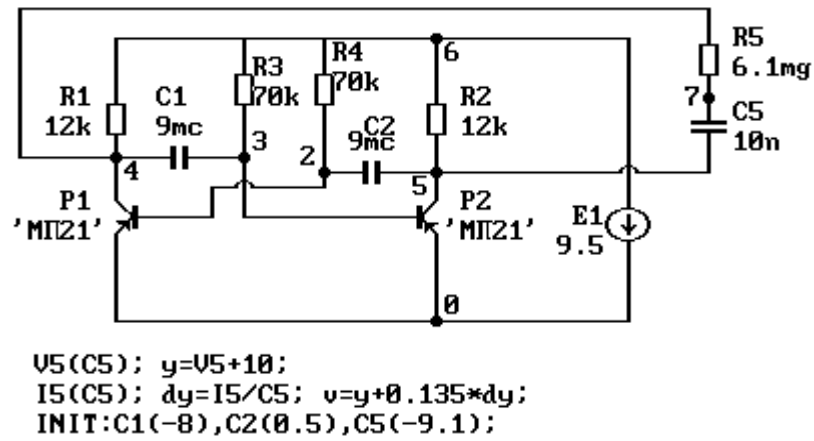


Рис.3.23. Схема мультивібратора та вирази для вихідного (y) і внутрішнього (v) сигналів макромоделі

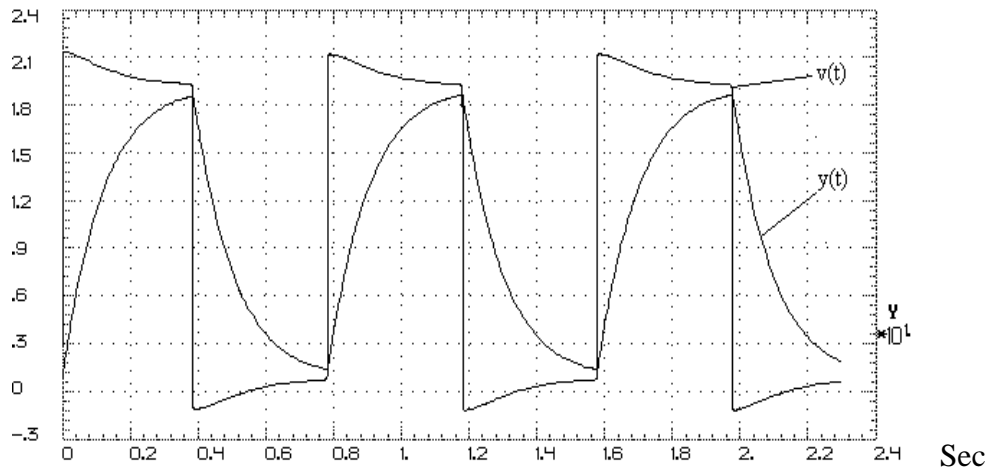


Рис.3.24. Форма коливань мультивібратора $y(t)$ та сигнал $v(t)$, обчислений оберненою лінійною підсистемою

Відповідно до алгоритму оберненої лінійної підсистеми знайдемо передатну функцію лінійної підсистеми, обернена до якої забезпечить потрібну форму $v(t)$ для заданої форми $y(t)$. У цьому випадку вже система першого порядку $T\dot{y} + y = v$ виконує таке завдання. Щоб визначити єдиний невідомий параметр T , чисельно продиференціюємо функцію $y(t)$, яка є вхідною для оберненої підсистеми, задамо ідеально прямокутну форму функції $v(t)$ та розв'яжемо задачу лінійної оптимізації відносно T :

$$\min_T \sum_{i=1}^n (T\dot{y}_i - (v_i - y_i))^2, \quad (3.54)$$

де середньоквадратичну похибку обчислимо за n відліками функцій $v_i = v(t_i)$, $y_i = y(t_i)$ та $\dot{y}_i = \dot{y}(t_i)$.

Значення $T=0.135$ сек. є розв'язком (3.54) за допомогою програми з Додатка 1 при $n=100$ на часовому інтервалі, що показаний на рис.3.24.

Отже, передатна функція лінійної підсистеми має вигляд

$$W(\lambda)=1/(0.135\cdot\lambda+1).$$

Оскільки лінійна підсистема першого порядку, то відповідно до загальної структури на рис.3.2 задача ідентифікації полягає тепер в апроксимації нелінійної функції $f(y,v)$ в системі

$$\begin{aligned} T\dot{y} + y &= v, \\ v &= f(y, v), \end{aligned} \quad (3.55)$$

що відповідає блок-схемі на рис.3.25 і є окремим випадком загальної структури на рис.3.2.

Для задачі ідентифікації нелінійної функції $v=f(y,v)$ значення $y(t)$ задані, а значення $v(t)$ обчислені як вихідний сигнал оберненої лінійної системи з передатною функцією $w^{-1}(\lambda)=0.135\cdot\lambda+1$. Вигляд функції $v(t)$, обчисленої за описом схеми на рис.3.23, показано на рис.3.24.

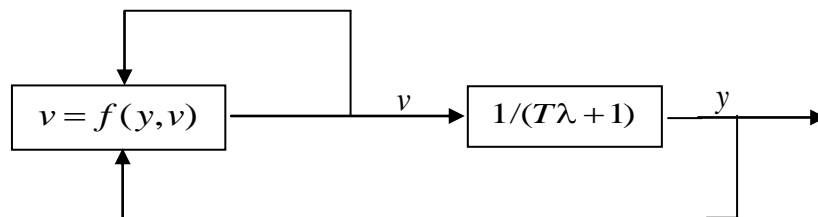
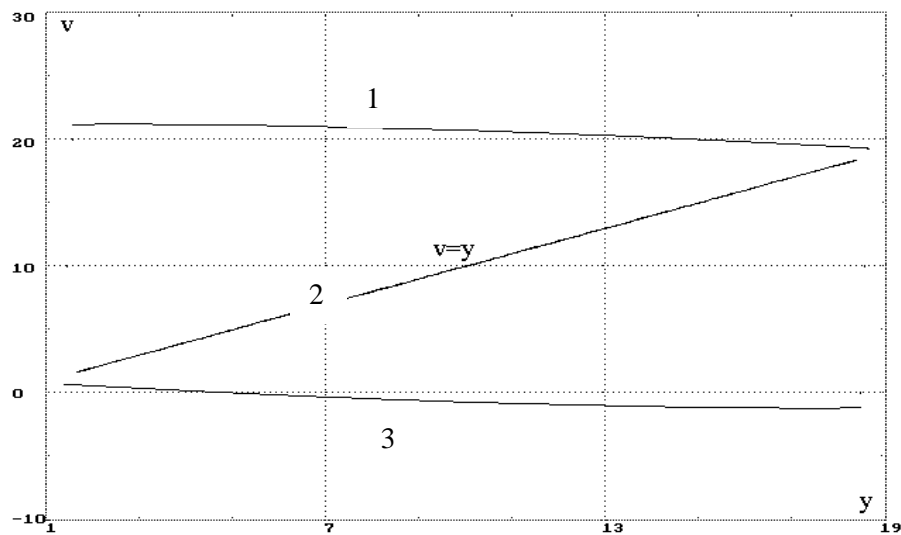
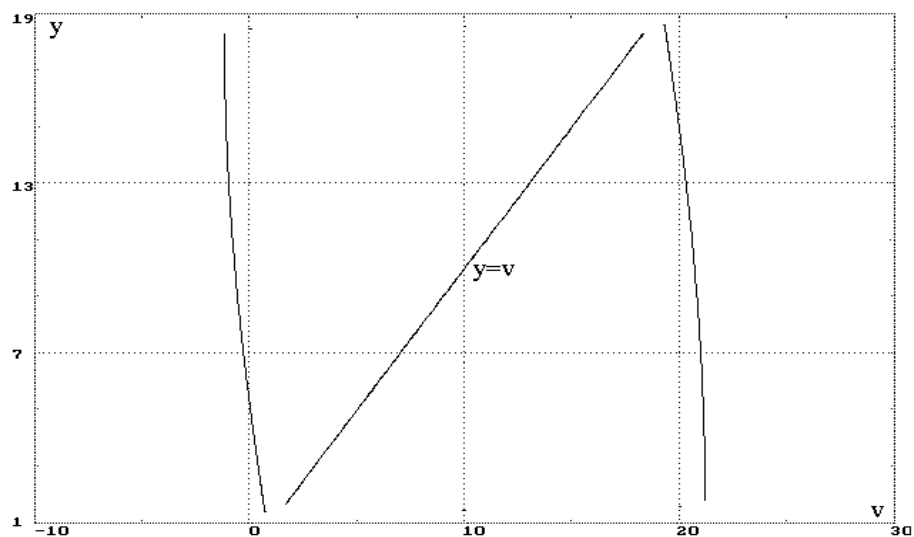


Рис.3.25. Блок-схема макромоделі мультавбратора

Як бачимо на рис.3.26, залежність v від y , збудована за парами миттєвих значень відповідних функцій, є неоднозначною.

Рис.3.26. Графік нелінійної функції $v=f(v,y)$

Натомість обернена залежність y від v є однозначною і допускає апроксимацію степеневим поліномом (рис.3.27).

Рис.3.27. Графік оберненої функції $y=y(v)$

Однак експериментальні значення утворюють лише два майже вертикальні відрізки, яких явно недостатньо для успішної апроксимації. Ситуація подібна до тієї, що виникла у параграфі 3.3 у випадку степеневій апроксимації неоднозначної статичної характеристики тригера.

Аналогічно до апроксимації неоднозначної статичної характеристики тригера доповнимо експериментальні значення точками на виділеному відрізку прямої $y=v$ і виконаємо апроксимацію степеневим поліномом

$y = \sum_{i=0}^n a_i v^i$. Шукана залежність має вигляд $v = 1/a_1 (y - a_0 - \sum_{i=2}^n a_i v^i)$, а систему (3.55) запишемо так:

$$T\dot{y} + y = v$$

$$v = 1/a_1(y - a_0 - \sum_{i=2}^n a_i v^i) \quad (3.56)$$

Проаналізуємо положення рівноваги в системі (3.56), які визначає рівність $v=y$. Розглянемо автоколивний рух системи, користуючись рис. 3.26. У режимі автоколивань зображуюча точка рухається ділянками 1 та 3, швидко переходячи з однієї на другу і не потрапляючи на ділянку 2. Отже, на ділянках 1 та 3 не повинно бути стійких положень рівноваги $v=y$, бо тоді в системі не буде режиму автоколивань. Графічно це означає, що ділянки 1 та 3 не повинні перетинатись із прямою $v=y$, бо будь-яка точка перетину відповідає стійкій системі. Формально ця вимога зводиться до виконання нерівностей

$$\begin{aligned} \sum_{i=0}^n a_i v^i &< v \quad \text{для всіх точок ділянки 1;} \\ \sum_{i=0}^n a_i v^i &> v \quad \text{для всіх точок ділянки 3.} \end{aligned} \quad (3.57)$$

Неважко переконатись, що положення рівноваги на ділянці 2 можуть бути стійкими і нестійкими. Справді, у разі апроксимації степеневим поліномом значення похідної dv/dy на ділянці 2 міститься в інтервалі $[0, \infty]$, тоді як значення $dv/dy=1$ є граничним між стійкою та нестійкою рівновагою. Отже, потрапляння зображувальної точки на ділянку 2 може спричинити припинення автоколивань у системі (3.56).

Система (3.56) є завершеним розв'язком поставленої задачі макромодельовання. Однак спроби знайти розв'язок (3.56) стандартними програмами, не розрахованими на числове інтегрування диференціальних рівнянь з неоднозначними нелінійностями, приречені на невдачу. Необхідно або модифікувати програми, або змінити модель так, щоб позбутись неоднозначності, зберігши основні властивості моделі. Класичний розв'язок останньої задачі є у праці [54, с.787]. Ідея полягає у введенні в модель ще одного допоміжного диференціального рівняння, яке перетворює неоднозначну нелінійність в однозначну. Для нашої моделі це дає систему

$$T\dot{y} + y = v$$

$$\tau\dot{v} + v = w$$

$$w = 1/a_1(y - a_0 - \sum_{i=2}^n a_i v^i) \quad (3.58)$$

де нелінійність є однозначною, тобто будь-якій парі значень y та v з області визначення відповідає одне й тільки одне значення w .

Унаслідок розширення системи (3.56) до системи (3.58) потрібно сформулювати умови, за яких введення допоміжного рівняння не приводить до якісних змін поведінки системи (3.58) порівняно з системою (3.56). Перш за все, сталу часу τ додаткового рівняння треба вибирати достатньо малою, щоб впливом її на перехідні процеси можна було знехтувати. Оскільки $\tau \ll T$, то можна, зафіксувавши y , розглянути стійкість положень рівноваги в системі

$$\begin{aligned} \tau \dot{v} + v &= w, \\ w &= 1/a_1 (\tilde{y} - a_0 - \sum_{i=2}^n a_i v^i), \end{aligned} \quad (3.59)$$

де \tilde{y} є зовнішньою повільною дією, а v – виходом. Положення рівноваги системи (3.59) \tilde{v} є розв'язками рівняння

$$\tilde{v} = 1/a_1 (\tilde{y} - a_0 - \sum_{i=2}^n a_i \tilde{v}^i).$$

Необхідна та достатня умова стійкості системи (3.59)

$$\left. \partial(1/a_1 (\tilde{y} - a_0 - \sum_{i=2}^n a_i v^i) - v) / \partial v \right|_{\tilde{v}} < 0 \quad (3.60)$$

повинна виконуватись для всіх допустимих положень рівноваги.

Обчислення похідної в (3.60) дає умову

$$-\sum_{i=1}^n i a_i \tilde{v}^{i-1} < 0 \quad (3.61)$$

Тепер можна остаточно сформулювати задачу апроксимації неоднозначної функції, яка полягає у визначенні $(n+1)$ коефіцієнтів a_i як розв'язку системи рівнянь

$$\tilde{y} = \sum_{i=0}^n a_i \tilde{v}^i \quad (3.62)$$

для всіх заданих значень \tilde{y} та відповідних значень \tilde{v} , обчислених оберненою лінійною підсистемою, у разі виконання нерівностей (3.57) для

всіх точок ділянок 1 та 2 на рис.3.25, а також нерівностей (3.61) для всіх значень \tilde{v} . Так сформульовану задачу можна розв'язати як типову задачу лінійного програмування з мінімаксним критерієм оптимальності та вказаними обмеженнями-нерівностями:

$$\min_{\tilde{a}} \max_{\tilde{y}, \tilde{v}} \left| \tilde{y} - \sum_{i=0}^n a_i \tilde{v}^i \right| \quad (3.63)$$

У табл.3.12 наведено текст завдання для системи САНОС-ПК із результатами розв'язування задачі (3.54) та дискретного аналога задачі (3.63). Задача (3.54) розв'язана програмою з Додатка 1, а дискретний аналог задачі (3.63) – програмою мінімаксної апроксимації (Додаток 2).

Таблиця 3.12. Опис макромоделі мультивібратора з поліноміальною нелінійністю

BEGI

Макромодель мультивібратора з поліноміальною апроксимацією

DESC

* Генератор сигнала (мультивібратор);

R3(6,3)70k; R4(6,2)70k; R1(6,4)12k; R2(6,5)12k; C1(4,3)9mc; C2(5,2)9mc;

P1(0,2,4)'МП21'; P2(0,3,5)'МП21'; E1(6,0)9.5;

R5(4,7)6.1mg; C5(7,5)10n; V5(C5);

Y=V5+10; * Сигнал мультивібратора;

I5(C5); DY=I5/C5; V=Y+0.135*DY; * Внутрішній сигнал макромоделі;

INIT: C1(-8), C2(0.5), C5(-9.1);

* Макромодель мультивібратора ;

YM=1/0.135*(VM-YM); * Лінійна підсистема ;

VM=1e6*(W-VM); * Нелінійна функція UM=F(YM,UM) ;

W=1/A1*(YM-A0-

VM*VM*(A2+VM*(A3+VM*(A4+VM*(A5+VM*(A6+VM*A7)))));

A0=5.15824; A1=-6.83848; A2=4.22528; A3=-1.05855; A4=0.138004;

A5=-0.00970633; A6=0.000348532; A7=-0.00000500631;

init: VM(0.5), YM(0.5);

YA=A0+VA*(A1+VA*(A2+VA*(A3+VA*(A4+VA*(A5+VA*(A6+VA*A7)))));

);

VA=T*10-1.5;

* Демонстрація нелінійної функції ;;;

TRAN

TMAX=2.3; init; ermax=0.0005; hmin=1e-12; hs=20e-4;

OUTPUT: Y,V, YM,VM, tabl, XMAX=2.3,YMIN=-3,YMAX=24;;

TRAN

TMAX=2.3; init; ermax=0.0005; hmin=1e-12; hs=5e-4;

output: V(Y), VM(YM), VA(YA), xmin=-3,xmax=24,ymin=-3,ymax=24;;

END

На рис.3.28 показано коливний процес у макромоделі $Y_M(t)$ та заданий сигнал $Y(t)$ згідно із завданням у табл.3.12, а на рис.3.29 –

залежності заданих внутрішнього сигналу від вихідного $V(Y)$, аналогічних сигналів макромоделі $VM(YM)$ і графік апроксимаційного полінома $VA(YA)$.

Можлива кусково-лінійна апроксимація залежності $V(Y)$. Як і для макромоделі тригера, кусково-лінійну апроксимацію будують без застосування спеціальних апроксимаційних програм, оскільки для визначення лінійних відрізків досить задати лише певні опорні точки та умови перемикання з однієї лінійної ділянки на іншу.

Залежність $V(Y)$ (рис.3.29) містить необхідну інформацію для побудови кусково-лінійної апроксимації. Координати точок $(V1, Y1)$, $(V2, Y2)$, $(V3, Y3)$, $(V4, Y4)$ (рис.3.29) визначають верхню та нижню прямі апроксимації:

$$VM = (V2 - V1) / (Y2 - Y1) * YM + (V1 * Y2 - V2 * Y1) / (Y2 - Y1);$$

$$VM = (V3 - V4) / (Y3 - Y4) * YM + (V4 * Y3 - V3 * Y4) / (Y3 - Y4).$$

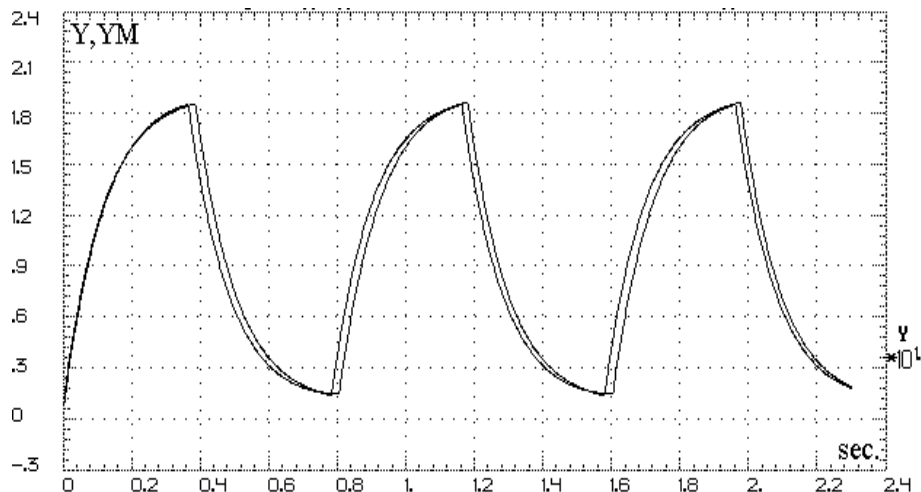


Рис.3.28. Заданий сигнал та коливний процес у поліноміальній макромоделі мультівібратора

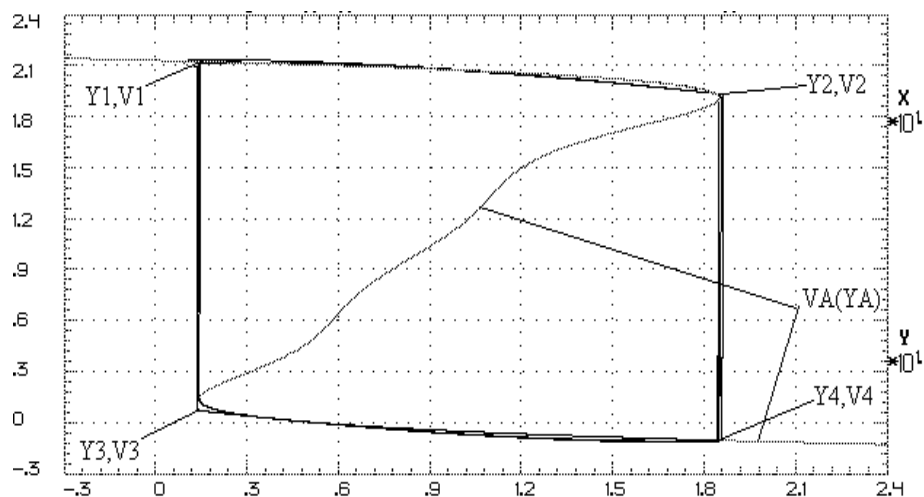


Рис.3.29. Апроксимація неоднозначної нелінійності
степеневим поліномом

Умови перемикавання лінійних ділянок мають вигляд:

$\text{if}(V \geq V_2) \text{ then } (L_1) \text{ else } (\text{if}(V > V_3) \text{ then } (L_2) \text{ else } (L_3));$

де L_1, L_2, L_3 , позначають верхню, середню та нижню лінійні ділянки апроксимації.

Складніша ситуація з середньою (внутрішньою) частиною, де нема експериментальних результатів. Напрошується з'єднання точок (V_2, Y_2) і (V_3, Y_3) відрізком прямої, але така неперервна апроксимація приводить до “зависання” під час інтегрування, як це сталося в кусково-лінійній моделі тригера (див. рис.3.9 та подальший текст).

Щоб не вводити умов “стрибків” в інтегратор, застосуємо метод, використаний для синтезу моделі тригера. Перейдемо до розривної кусково-лінійної апроксимації, причому середній відрізок візьмемо з кутом нахилу понад 90° до осі абсцис, проте меншим, ніж кут нахилу прямої, що проходить через точки з координатами (V_1, Y_1) , (V_4, Y_4) . Цей прийом детально пояснено в параграфі 3.2.2.

Блок-схема макромоделі збігається з рис.3.25, лише нелінійний блок містить кусково-лінійну апроксимацію нелінійності.

Опис завдання, що дає змогу порівняти заданий сигнал та сигнали моделі мультивібратора з кусково-лінійною апроксимацією, наведено в табл.3.13.

Таблиця 3.13. Опис макромоделі мультивібратора
з кусково-лінійною апроксимацією

BEGI

Macromodel of the multivibrator (piecewise-linear approximation)

DESC

* Generator (multivibrator);

R3(6,3)70k; R4(6,2)70k; R1(6,4)12k; R2(6,5)12k; C1(4,3)9mc; C2(5,2)9mc;

P1(0,2,4)'МП21'; P2(0,3,5)'МП21'; E1(6,0)9.5;

R5(4,7)6.1mg; C5(7,5)10n; V5(C5); Y=V5+10;

I5(C5); DY=I5/C5; U=Y+0.135*DY;

init:C1(-8),C2(0.5),C5(-9.1);

* Macromodel of generator ;

YM=1/0.131*(UM-YM); * Linear subsystem ;

UM=1e5*(W-UM); * Nonlinear function ;

Y1=1.42; U1=21.1; Y2=18.6; U2=19.3; Y3=1.37; U3=0.68; Y4=18.5; U4=-1.0;

A1=(U2-U1)/(Y2-Y1); B1=(U1*Y2-U2*Y1)/(Y2-Y1);

A3=(U3-U4)/(Y3-Y4); B3=(U4*Y3-U3*Y4)/(Y3-Y4); * UM=A*YM+B ;

W=if(UM.ge.U2)(A1*YM+B1)(if(UM.gt.U3)(-8*YM+90)(A3*YM+B3));

INIT:YM(1),UM(20);

```

YA=if(UA.ge.U2)(1/A1*(UA-B1))(if(UA.gt.U3)(90/8-1/8*UA)(1/A3*(UA-B3)));
UA=T*10-1.5;                                * Nonlinear function demonstration;;
TRAN
TMAX=2.3; init; ermax=5e-4; hmin=1e-12; hs=1e-3;
OUTPUT: U,Y, UM,YM, XMAX=2.3,YMIN=-3,YMAX=24;;
TRAN
TMAX=2.4; init; ermax=5e-4; hmin=1e-12; hs=1e-3;
output: U(Y), UM(YM), UA(YA), xmin=-3,xmax=24,ymin=-3,ymax=24;;
END

```

На рис.3.30 показані заданий сигнал Y та внутрішній V , обчислений оберненою лінійною підсистемою, а також відповідні сигнали кусково-лінійної макромоделі (YM , VM).

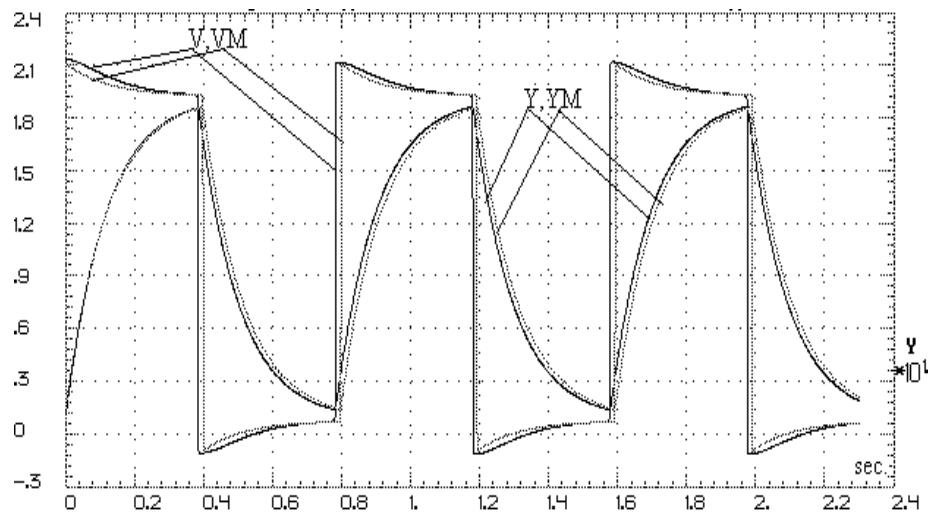


Рис.3.30. Сигнали задані (V, Y) та згенеровані кусково-лінійною макромоделлю мультивібратора (VM, YM)

Залежність $V(Y)$ для вхідного сигналу, а також відповідна залежність $VM(YM)$ для макромоделі, згідно із завданням у табл.3.13, показані на рис.3.31. Там же зображена кусково-лінійна апроксимація $VA(YA)$ із середнім відрізком, що створений з метою запобігання “зависанням”.

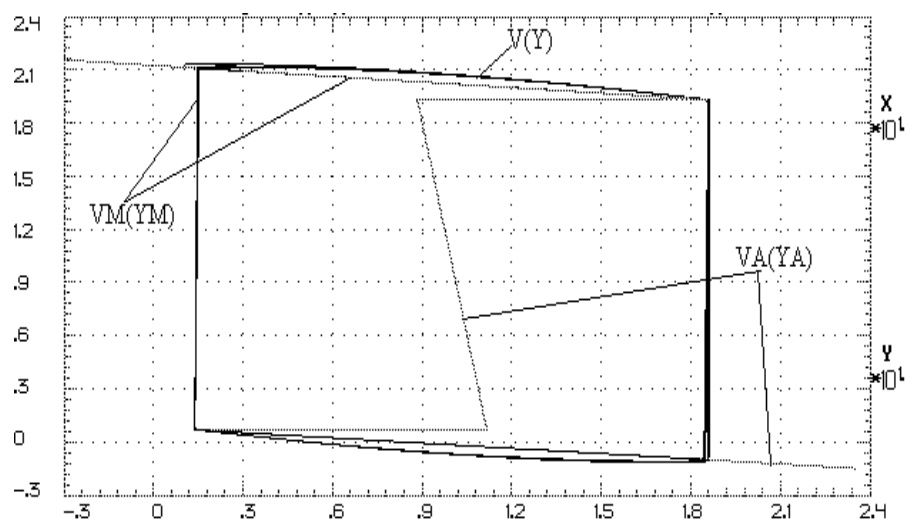


Рис.3.31. Кусково-лінійна апроксимація неоднозначної нелінійності в макромоделі муьтивібратора

РОЗДІЛ 4

РЕГУЛЯРИЗАЦІЯ АПРОКСИМАЦІЙНИХ ЗАДАЧ
МАКРОМОДЕЛЮВАННЯ

Загальні моделюючі структури для систем із зосередженими параметрами спільно з методами ідентифікації (розділ 3) дають змогу звести задачу макромодельовання до апроксимації лінійної динамічної системи та нелінійної вектор-функції багатьох аргументів. Якщо апроксимація лінійної динамічної системи доволі проста (розділ 2), то апроксимація нелінійної функції є складним завданням. Як зазначив Г.І. Марчук [49, с.187], теоретичні аспекти апроксимації функцій багатьох аргументів на хаотичних сітках ще не розроблені й висвітлені евристично.

Наведемо загальне формулювання задачі [63].

Нехай $\bar{x}_t \in R^n$ – вектор аргументів апроксимованої вектор-функції $\bar{f}(\bar{x}_t) \in R^m$, де $t \in R$ – деякий параметр (у наших задачах t – час). Нехай також $\bar{v}_t \in R^m$ – вектор значень функції $\bar{f}(\bar{x}_t)$, які залежать від цього ж параметра t . Тоді для складових $\bar{f}(\bar{x}_t)$ і \bar{v}_t векторів $\bar{f}(\bar{x}_t)$ і \bar{v}_t для кожного i -го значення параметра t справджується рівність

$$f(\bar{x}_{t_i}) = v_{t_i}, \quad i = \overline{1, N}.$$

Якщо апроксимувати $\bar{f}(\bar{x}_t)$ в лінійному просторі дійсних базисних функцій $\phi_j(\bar{x}_t)$ розмірності M , то отримуємо систему лінійних алгебричних рівнянь (СЛАР) щодо коефіцієнтів апроксимації a_j ,

$$\sum_{j=1}^M a_j \phi_j(\bar{x}_{t_i}) = v_{t_i}, \quad i = \overline{1, N}, \quad (4.1)$$

яку при $N > M$ можна розв'язувати методом найменших квадратів.

Можливе мінімаксне формулювання задачі. Для цього введемо, крім вектора \bar{a} , ще один коефіцієнт $\varepsilon > 0$:

$$-\varepsilon \leq \sum_{j=1}^M a_j \phi_j(\bar{x}_{t_i}) - v_{t_i} \leq \varepsilon, \quad i = \overline{1, N}. \quad (4.2)$$

Знайти вектор \bar{a} , що забезпечує мінімум функції мети $z = \varepsilon$ за обмежень-нерівностей (4.2) – це типова задача лінійного програмування.

Методи розв'язування обох задач розроблені до стандартних програм математичного забезпечення. Зазначимо, що методу найменших квадратів надають перевагу за наявності похибок даних \bar{x}_t і \bar{v}_t , тоді як апроксимація за Чебишовим дає найліпше рівномірне наближення. Деякі результати

порівняння обох підходів стосовно макромодельовання лінійних систем наведені в параграфі 2.4, а в параграфі 4.6 порівняно результати застосування обох критеріїв у разі макромодельовання нелінійної системи.

Дуже важливим є вибір базису апроксимації – функцій $\varphi_j(\bar{x}_t)$. Поширені багатовимірні степеневі поліноми

$$f(\bar{x}) \approx \sum_{i=0}^r \sum_{j=0}^r \dots \sum_{k=0}^r c_{ij\dots k} x_1^i x_2^j \dots x_n^k, \quad i + j + \dots + k \leq r. \quad (4.3)$$

Основу для цього дає відома теорема Стоуна-Вейерштраса [53], яка визначає принципову досяжність будь-якої точності апроксимації у разі підвищення степеня r багатовимірного степеневого полінома (4.3). Однак безпосередній розв'язок задач (4.1) або (4.2) за апроксимації (4.3) приводить до появи некоректності в сенсі Адамара [31] для достатньо великих r (див. параграфи 1.3.4, 2.3). Це виявляється, з одного боку, в тому, що малі зміни векторів \bar{x}_t і \bar{v}_t спричиняють значні зміни значень $c_{ij\dots k}$. З іншого боку, малі зміни $c_{ij\dots k}$ зумовлюють значні порушення умов (4.1) або (4.2).

В параграфах 4.2 та 3.2.3 продемонстровано спеціальний вибір апроксимаційного базису, що дозволяє регуляризувати деякі некоректні задачі макромодельовання.

Існують прийоми усунення некоректностей, що ґрунтуються на класичній теорії регуляризації некоректних задач.

4.1. РЕГУЛЯРИЗУЮЧИЙ ФУНКЦІОНАЛ. МАКРОМОДЕЛЬ СИСТЕМИ З ВИПАДКОВИМ СИГНАЛОМ

У працях школи А.М. Тіхонова для регуляризації розв'язку СЛАР запропоновано, зокрема, мінімізувати функціонал Тіхонова, який у наших позначеннях має вигляд

$$M_{\alpha, h} = \|(\bar{a}, \bar{\varphi}(\bar{x}_t)) - \bar{v}_t\|^2 + \alpha \|\bar{a}\|^2, \quad (4.4)$$

де вектор $\bar{a} \in R^m$ залежить, у загальному випадку, і від похибки h в значеннях \bar{x}_t і \bar{v}_t . Перший член в (4.4) забезпечує найкращу точність розв'язку СЛАР (4.1) в обраній метриці, а другий член – власне регуляризуючий – мінімальну норму вектора \bar{a} .

Якщо метрика простору апроксимації евклідова, то функціонал Тіхонова набуває вигляду

$$M_{\alpha, h} = \sum_{i=1}^N \left(\sum_{j=1}^M a_j \varphi_j(\bar{x}_{t_i}) - v_{t_i} \right)^2 + \alpha \sum_{j=1}^M a_j^2. \quad (4.5)$$

Відомі алгоритми, які дають змогу визначити α за відомим h [31]. Однак похибка початкових даних, як звичайно, не відома або її важко визначити. У цьому випадку залишається емпірично підбирати параметр α , стежачи за коректністю розв'язку СЛАР.

До задачі, регуляризованої за (4.5), найпростіше перейти шляхом доповнення (4.1) рівняннями

$$\alpha a_j^2 = 0, \quad j = \overline{1, M}. \quad (4.5a)$$

Аналогічний регуляризаційний додаток у випадку чебишовської метрики доповнює систему (4.2):

$$-\varepsilon \leq \sqrt{\alpha} a_j \leq \varepsilon, \quad j = \overline{1, M}.$$

У Додатках 1 та 4 містяться тексти FORTRAN-програм, що мінімізують регуляризований функціонал (4.5). В їхній основі є алгоритм із [48], доповнений блоками нормування даних, формування регуляризаційних додатків (4.5a) та завдання мовою САНОС-ПК. Додаток 2 містить програму розв'язку регуляризованої задачі в мінімаксній метриці, де використано симплекс-алгоритм із [50].

Застосування наведеної методики регуляризації розглянемо на прикладі макромоделі системи з випадковим сигналом [59, 60].

Опишемо синтез систем звичайних диференціальних рівнянь, розв'язки яких відтворюють статистичні властивості заданих випадкових послідовностей. Спочатку з'ясуємо синтез таких систем на прикладі моделювання вихідного сигналу дивного атрактора Лоренца, а потім – на прикладі послідовності псевдовипадкових чисел, рівномірно розподілених в інтервалі $[0, 1]$.

У відомій статті Е.Лоренца [52] вперше доведено можливість отримання одновимірних сигналів, що відповідають хаотичним рухам деяких неперервних детермінованих систем третього порядку. Сформулюємо обернену задачу: синтезувати неперервну систему, сигнал якої у певному сенсі еквівалентний заданій одновимірній випадковій послідовності.

Задано деякий об'єкт, на виході якого з'являється послідовність випадкових чисел. Треба створити автономну неперервну дисипативну систему із зосередженими параметрами, яка б певним чином відтворювала статистичні властивості заданої випадкової послідовності. Відомо, що така система повинна містити дивний атрактор та її порядок не може бути менший за три [60].

$$\begin{aligned}
\dot{y}_1 &= y_2; \\
\dot{y}_2 &= y_3; \\
\dot{y}_3 &= 1040y_1 - \frac{88}{3}y_2 - \frac{41}{3}y_3 + v; \\
v &= 11\frac{y_2^2}{y_1} + \frac{y_2y_3}{y_1} - y_1^2y_2 - 10y_1^3; \\
y_1(0) &= 1; \quad y_2(0) = -10; \quad y_3(0) = 500.
\end{aligned} \tag{4.6}$$

У параграфі 3.1 доведено теорему 2, що визначає загальну структуру математичної моделі із зосередженими параметрами, та наведено аналітичне перетворення рівнянь Лоренца до вигляду рівнянь (4.6). Блок-схема цих рівнянь зображена на рис.3.3.

Рівняння (4.6) дають змогу збудувати нескладний алгоритм відновлення коефіцієнтів системи рівнянь за одним лише вихідним сигналом y_1 . Для цього треба обчислити похідні сигналу y_1 та вибрати метрику, у якій мінімізувати норму

$\|\ddot{y}_1 - (a_1y_1 + a_2\dot{y}_1 + a_3\ddot{y}_1 + c_1\dot{y}_1^2/y_1 + c_2\dot{y}_1\ddot{y}_1/y_1 + c_3y_1^2\dot{y}_1 + c_4y_1^3)\|$ в деякій заданій області. Якщо метрика евклідова та сигнал $y_1(t_m) \equiv y_m$ заданий дискретно в M точках, то задача набуває вигляду

$$\min_{\bar{a}, \bar{c}} \sum_{m=1}^M (\ddot{y}_m - (a_1y_m + a_2\dot{y}_m + a_3\ddot{y}_m + c_1\dot{y}_m^2/y_m + c_2\dot{y}_m\ddot{y}_m/y_m + c_3y_m^2\dot{y}_m + c_4y_m^3))^2 \tag{4.7}$$

і зводиться до розв'язування системи семи лінійних алгебричних рівнянь відносно семи невідомих компонент векторів \bar{a} і \bar{c} .

Задача (4.7) розв'язана в [37]. Похідні обчислено за методом з [36] через власні вектори коваріаційної матриці, збудованої на дискретних значеннях сигналу y_1 . Значення коефіцієнтів відновлювались з точністю 0.0002 у разі достатньо малих похибок даних. Там же розв'язані й інші задачі зворотної нелінійної динаміки, у тому числі при наявності адитивних шумів у вихідних даних.

Добре відомо, що числове диференціювання дискретно заданої функції є некоректною процедурою, зокрема нестійкою до збурень даних. Саме тому для отримання стійких результатів потрібно застосовувати регуляризаційні алгоритми.

У праці [37] використані числові процедури згладжування сигналу. Це дало змогу усунути некоректності, пов'язані з шумами сигналу. Однак некоректність виявляє себе під час розв'язування задач типу (4.7), навіть якщо дані "безшумні". Тому регуляризація повинна охоплювати всі етапи формулювання та розв'язування вказаних задач.

У [36] немає вказівок щодо регуляризаційних заходів. Не застосовували регуляризацію й автори вказаної вище публікації, розв'язуючи задачу (4.7) та аналогічні. Відносну коректність отриманих результатів можна пояснити малою розмірністю задач.

Числове диференціювання можна виконувати різними регуляризованими методами. В параграфі 2.3 похідні обчислено регуляризованими диференціальними рівняннями. У параграфі 2.4.3 використано ковзні інтерполяційні та апроксимаційні поліноми. Розглянемо тепер застосування сплайн-функцій.

На множині значень y_m , $m=1, \dots, M$ завжди можна збудувати інтерполяційний сплайн степені $n < M$ з числом коефіцієнтів $p=M$ і неперервністю похідних до $(n-1)$ -го порядку включно [62]. Для задачі (4.7) необхідно обчислити похідні до третього порядку. Якщо домагатися неперервності похідних хоча б четвертого порядку, то інтерполяційний сплайн повинен бути п'ятого степеня.

Регуляризації процедури побудови сплайн-функції, а отже й аналітичного обчислення похідних за сплайном, досягають застосуванням апроксимаційного сплайна [62] з кількістю коефіцієнтів $p < M$. Апроксимаційний сплайн здатний фільтрувати шуми сигналу.

У праці [62] опубліковані тексти ФОРТРАН-програм, що реалізують усі потрібні алгоритми в алгебрі В-сплайнів. Саме цими програмами ми користувались для побудови сплайн-функцій та обчислення їхніх значень. У Додатку 5 наведено повний текст програми, що виконує сплайн-інтерполяцію заданого степеня та обчислення похідних заданого порядку.

Під час обчислення похідних у задачі (4.7) за допомогою сплайн-інтерполяції п'ятого степеня точність відновлення коефіцієнтів системи (4.6) не гірша, ніж у [37].

Задача (4.7) є нескладною, оскільки відомий точний апроксимаційний базис нелінійної функції. Але якщо базис невідомий, то апроксимацію виконують, як звичайно, багатовимірним степеневим поліномом [8, 12, 35, 63].

У нашому випадку степеневий поліном тривимірний:

$$\begin{aligned} \dot{y}_1 &= y_2; \\ \dot{y}_2 &= y_3; \\ \dot{y}_3 &= \sum_{i,j,k=0}^r a_{ijk} y_1^i y_2^j y_3^k. \end{aligned} \tag{4.8}$$

Структура (4.8) є окремим випадком загальної структури, обґрунтованої теоремою 2 з розділу 3. Обмежимося умовами $r=3$ та $i+j+k \leq 3$. Тоді кількість невідомих коефіцієнтів a_{ijk} дорівнює 20, а задача, аналогічна до (4.7), набуває вигляду

$$\min_{\bar{a}} \sum_{m=1}^M (\ddot{y}_m - \sum_{i,j,k=0}^3 a_{ijk} y_m^i \dot{y}_m^j \ddot{y}_m^k)^2, \quad i+j+k \leq 3. \quad (4.9)$$

Однак отримати стійкі результати безпосереднім розв'язуванням задачі (4.9) виявилось неможливим, оскільки вона некоректна. Відповідно до ідеї регуляризаційного функціонала замість задачі (4.9) розв'яжемо

$$\min_{\bar{a}} \left(\sum_{m=1}^M (\ddot{y}_m - \sum_{i,j,k=0}^3 a_{ijk} y_m^i \dot{y}_m^j \ddot{y}_m^k)^2 + \alpha \sum_{i,j,k=0}^3 a_{ijk}^2 \right), \quad i+j+k \leq 3, \quad (4.10)$$

де параметр регуляризації α підберемо емпірично залежно від некоректності задачі.

Нижче наведено коефіцієнти апроксимації, що є розв'язком задачі (4.10), отриманим за допомогою програми, аналогічної до наведеної в Додатку 4, при $M=500$ та $\alpha=1.0$:

$a_{000}=3.4120_{10} 01$; $a_{100}=5.8604_{10} 02$; $a_{010}=1.0572_{10} 02$; $a_{001}=-5.6501_{10} 00$;
 $a_{200}=-1.7036_{10} 01$; $a_{020}=-5.8876_{10} 03$; $a_{002}=1.4002_{10} 04$; $a_{110}=2.1736_{10} 02$;
 $a_{101}=1.5655_{10} 02$; $a_{011}=7.8429_{10} 04$; $a_{300}=-5.3826_{10} 00$; $a_{030}=2.4692_{10} 03$;
 $a_{003}=5.6768_{10} 07$; $a_{210}=-2.2706_{10} 00$; $a_{201}=6.6169_{10} 03$; $a_{120}=7.7593_{10} 02$;
 $a_{021}=3.4417_{10} 04$; $a_{102}=1.1896_{10} 04$; $a_{012}=-8.3498_{10} 06$; $a_{111}=-3.4414_{10} 03$.

На рис.4.1 показано проекцію фазової траєкторії системи (4.6) на площину x_1-x_1' , а на рис.4.2 – проекцію на площину y_1-y_2 фазової траєкторії системи (4.8) з вище наведеними коефіцієнтами за початкових умов $y_1(0)=1$; $y_2(0)=-10$; $y_3(0)=500$.

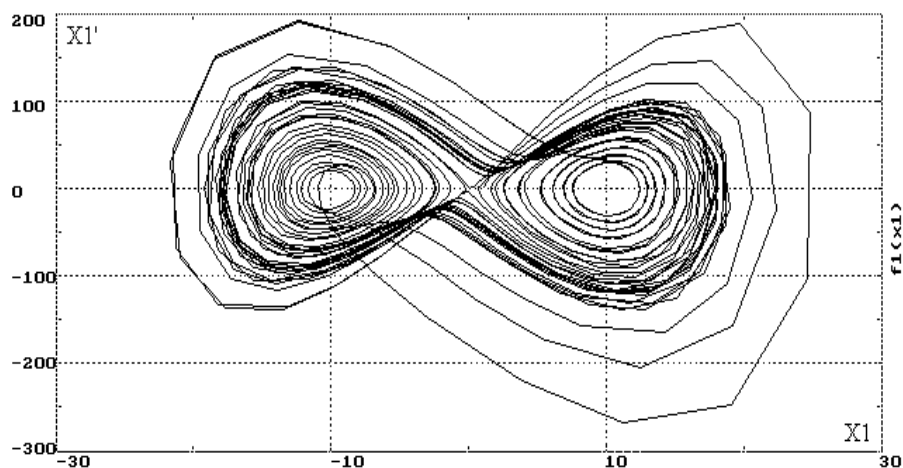


Рис.4.1. Проекція фазового портрета атратора Лоренца

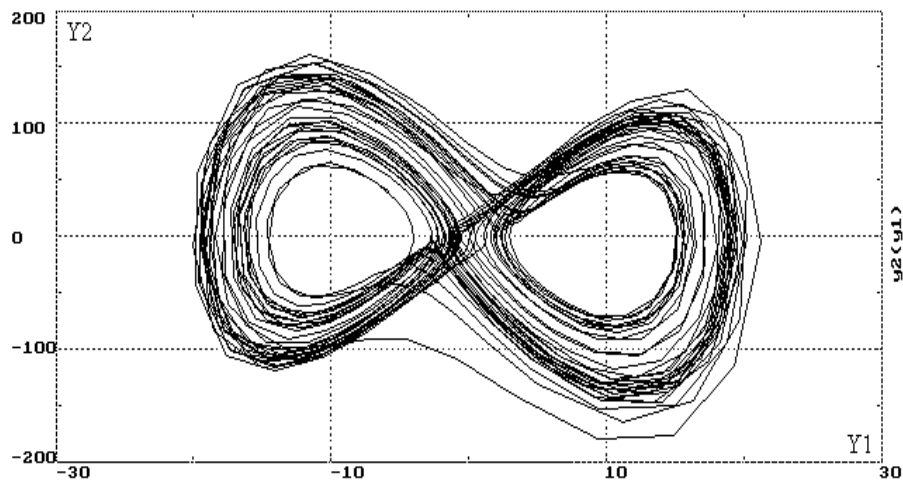


Рис.4.2. Проекція фазового портрета реконструкції атрактора Лоренца

Фрактальна розмірність [61] для системи (4.8) дорівнює 2.11, що відповідає значенню фрактальної розмірності 2.07 для системи (4.6). Проекції фазових портретів на рис.4.1 і рис.4.2 відображають топологічну подібність систем.

Застосуємо викладений метод конструювання моделі за однією вихідною змінною до послідовності випадкових чисел, рівномірно розподілених у інтервалі $[0,1]$. Псевдовипадкову послідовність з періодом повторення $>10^{10}$ відліків генерує стандартна програма (Додаток 5). Ця послідовність прив'язана до умовного часового масштабу. Прийнято, що випадкові числа з'являються рівномірно з часовим інтервалом 1 с.

Похідні обчислені програмою з Додатка 5 за допомогою інтерполяційного сплайну. На відрізку послідовності побудовано інтерполяційний сплайн п'ятого степеня й обчислено значення трьох похідних сплайну у вузлах інтерполяції, а також у дев'яти проміжних рівновіддалених точках для кожних двох сусідніх вузлів інтерполяції. Залежність значень першої похідної сплайну y_1' від значень самого сплайну y_1 зображено на рис.4.3.

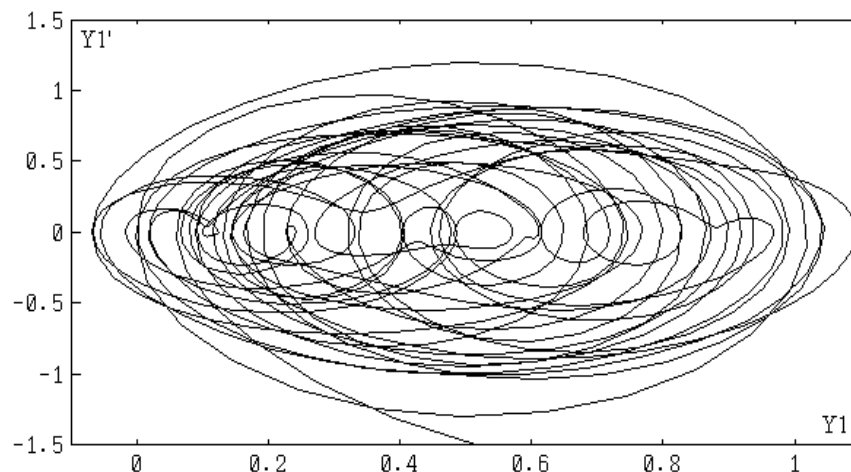


Рис.4.3. Інтерполяція випадкової послідовності.

Рис.4.3 можна розглядати як проекцію на площину y_1-y_1' фазового портрета деякої неперервної системи третього порядку, яка відповідає реалізації випадкового процесу в сенсі проведеної сплайн-інтерполяції.

На множинах значень сплайну та його похідних розв'язано регуляризовану задачу (4.10), тобто знайдено коефіцієнти моделювальної системи (4.8). Задовільні результати отримані вже для досить коротких відрізків випадкових послідовностей завдовжки від 250 до 1000 чисел та для значень параметра регуляризації α від 0.001 до 0.42.

Отримано понад десять систем з дивними атракторами для різних довжин послідовностей та різних значень параметра α . Коефіцієнти однієї із систем диференціальних рівнянь (4.8), отриманої як розв'язок (4.10) для послідовності завдовжки 290 чисел при $\alpha=0.001$, такі:

$$\begin{aligned} a_{000} &= -0.11817; a_{100} = 6.55425; a_{010} = -11.0487; a_{001} = -0.38934; a_{200} = -9.41420; \\ a_{020} &= -4.23326; a_{002} = -0.03089; a_{110} = 38.6623; a_{101} = 2.12705; a_{011} = 9.24564; \\ a_{300} &= 2.82030; a_{030} = -6.76449; a_{003} = -0.08485; a_{210} = -34.6874; a_{201} = -2.19388; \\ a_{120} &= 3.47065; a_{021} = 0.46197; a_{102} = -0.47460; a_{012} = -3.26361; a_{111} = -16.9036; \end{aligned}$$

Систему (4.8) з наведеними вище коефіцієнтами інтегрували неявним методом другого порядку з кроком інтегрування не більше 0.1 секунди та максимальною локальною похибкою 0.0001 за таких початкових умов: $y_1(0)=0.5$; $y_2(0)=0.2$; $y_3(0)=0$. Проекція отриманого фазового портрета на площину y_1-y_2 зображена на рис.4.4.

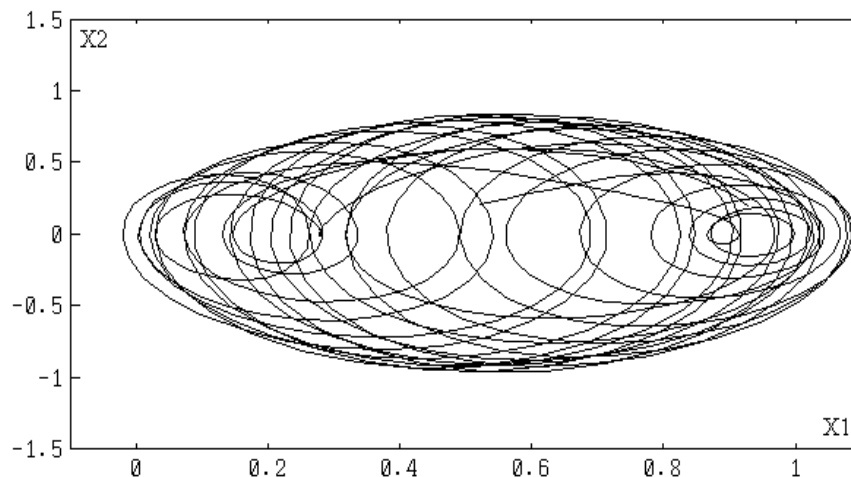


Рис.4.4. Проекція фазового портрета
макромоделі випадкової послідовності

Фазовий портрет з рис.4.4 відповідає дивному атрактору, існування якого підтверджене позитивним значенням старшого характеристичного показника Ляпунова (див. розділ 5), що дорівнює 0.23, а також дробовим значенням кореляційної розмірності 2.27 [61].

Послідовність значень вихідного сигналу макромоделі y_1 , взятих із відповідним часовим інтервалом (1 секунда) та відображених в інтервал

значень $[0,1]$, моделює випадкову послідовність. Деякі результати порівняння вихідного сигналу макромоделі та випадкової послідовності показані на рис.4.5.

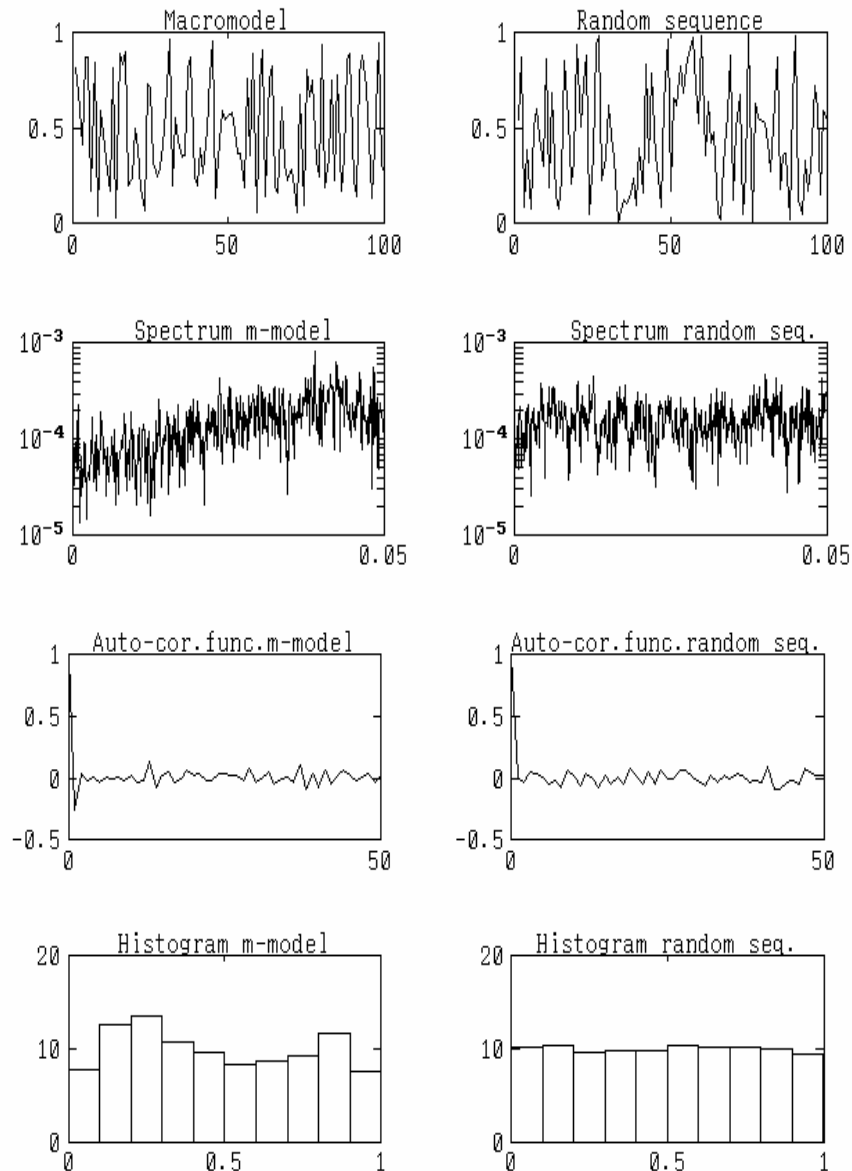


Рис.4.5. Порівняння макромоделі та випадкової послідовності

Спектральні густини обчислено для 4096 точок відповідних послідовностей, автокореляційні функції – для 500 точок (50 початкових показані на рис.4.5). Вісь ординат гістограм відображає відсотки.

У цілому статистичні властивості випадкової та моделювальної послідовностей подібні.

Зазначимо, що всі отримані автором дивні атрактори виявились досить вразливими. Зокрема, вони руйнувались у разі застосування явних методів інтегрування.

Наведений приклад моделювання випадкової послідовності числово реалізує зворотну задачу синтезу дивного атрактора із заданими

властивостями. Досягти позитивних результатів ми змогли лише завдяки застосуванню регуляризації за Тіхоновим.

4.2. ВИБІР АПРОКСИМАЦІЙНОГО БАЗИСУ. МАКРОМОДЕЛЬ СИНХРОННОГО ГЕНЕРАТОРА

Ефективний засіб уникнення проявів некоректності – підібрати такі базисні функції $\varphi_j(\bar{x})$, що відрізняються від степеневих поліномів і відображають деякі суттєві особливості апроксимованої функції. Перебирання різних можливих апроксимаційних базисів (степеневого, гармонічного, функцій Бесселя, функцій Уолша та ін.) чи їхніх комбінацій є громіздким та, найголовніше, не гарантує успіху. Загальні теоретичні засади такого апроксимаційного підходу розроблені мало [46]. Лише вивчення особливостей кожного конкретного випадку дає змогу підібрати базисні функції, що суттєво полегшують апроксимаційну задачу. Зокрема, до базисних функцій варто залучати характерні функції об'єкта.

Для систем, що дають змогу виміряти статичну характеристику $\bar{y} = \bar{S}(\bar{x})$, доцільно застосовувати вектор-функцію $\bar{S}(\bar{x})$ як базисну під час апроксимацій.

Опишемо регуляризацію з використанням статичної характеристики як однієї з базисних функцій на прикладі синтезу макромоделі електромашинного синхронного генератора [64]. Синтезуємо макромодель корабельного синхронного генератора типу ТМВ2-2 з такими основними параметрами: $S=2500\text{кВН}$, $U=400\text{В}$, $I_\phi=3609\text{А}$, $f=50\text{Гц}$, $\cos\phi=0.8$, $R_{\text{НОМ}}=0.8\text{ Ом}$. Об'єкт описано системою алгебро-диференціальних рівнянь типу рівнянь Парка-Горєва:

$$\begin{aligned} \psi_f' &= 0.24U_f - 0.22i_f; \\ i_d' &= 523(U_d - R_N i_d) - 314i_q; \\ i_q' &= 523(U_q - R_N i_q) + 314i_d; \\ i_D' &= -42.7\psi_f' - 0.71i_d' - 11.3i_D; \\ i_Q' &= -0.8i_q' - 0.82i_Q; \\ i_f &= 1.59\psi_f - 0.47i_d - 0.473i_D; \\ U_d &= 0.18R_N i_d - 1.29i_d + 0.0026i_D - 0.0036\psi_f' - 1.32i_Q; \\ U_q &= 0.19R_N i_q + 1.27i_d + 0.0034i_Q + 2.61i_f + 1.3i_D; \\ U_N &= \sqrt{U_d^2 + U_q^2}; \end{aligned} \tag{4.11}$$

де i_f, i_d, i_q, i_D, i_Q – відповідно діючі значення струмів кіл збудження, статора за осями d, q , демпферів за осями d, q ; U_f, ψ_f, U_d, U_q – діючі значення напруг та потокозчеплень тих же кіл.

Керуючими (вхідними) змінними об'єкта є напруга контуру збудження U_f та опір навантаження R_N , а регульованою (вихідною) змінною – напруга U_N . Вхідні змінні набувають значення у межах $0 \leq U_f \leq 10$; $0.8 \leq R_N \leq 100$.

Початкові умови розрахунку перехідних процесів такі:

$$\psi_f(0)=0.282; i_d(0)=-1.36; i_q(0)=0.47; i_D(0)=0; i_Q(0)=0.$$

Мета макромодельовання полягає у зменшенні порядку диференціальних рівнянь моделі та жорсткості цих рівнянь при збереженні адекватної поведінки моделі на заданих множинах вхідних сигналів.

Процес ідентифікації засвідчує, що вдалий вибір змінних апроксимаційного базису має вирішальне значення.

Макромодель ідентифікували для кусково-постійних вхідних функцій U_f та R_N , зображених на рис.4.6.

Спочатку макромодель синтезували за рекурсивною структурою на рис.3.2 з використанням апроксимації нелінійних функцій багатовимірними степеневими поліномами від вказаних вхідних та вихідних змінних з мінімізацією суми середньоквадратичних відхилів у дискретні моменти часу. Однак спроби отримати прийнятні макромоделі цим шляхом виявились невдалими внаслідок суттєвої некоректності задачі, яку не вдалось усунути традиційним методом, застосованим у параграфі 4.1.

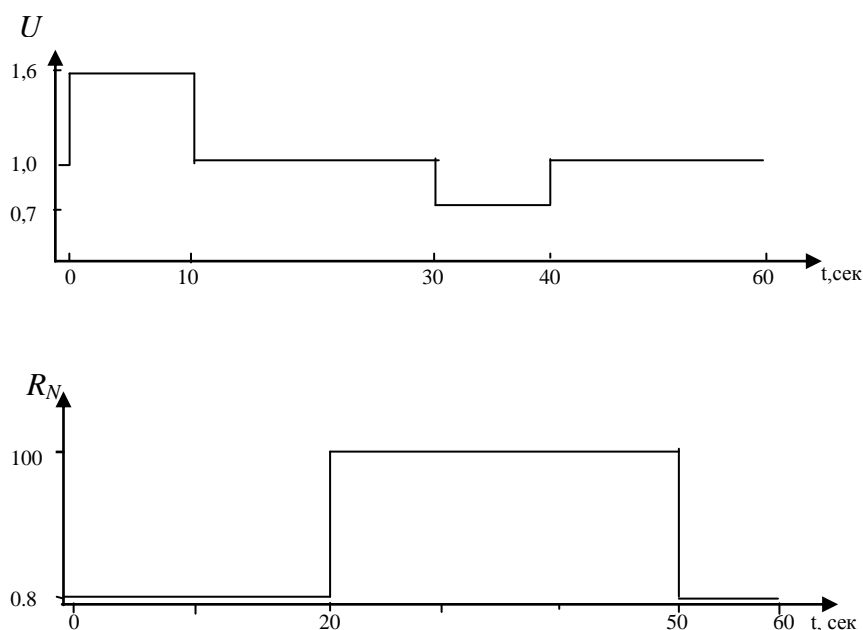


Рис.4.6. Вхідні функції об'єкта

Задачу було успішно розв'язано лише внаслідок переходу до іншого апроксимаційного базису, що враховував особливості об'єкта.

Апроксимацію статичної характеристики об'єкта визначено шляхом нелінійної мінімізації середньоквадратичної похибки згідно із задачею

$$\min_{\vec{c}} \sum_{i=0}^{100} (U_{Ni} - (c_1 + c_2 \exp(c_3 (R_{Ni} - 0.8))))^2; \quad (4.12)$$

де $R_{Ni}=0.8+i$; U_{Ni} – відповідне значення U_N як розв’язок системи (4.11) при $U_f=1$, $\psi'_f=i'_d=i'_q=i'_D=i'_Q=0$; $\bar{c}=(c_1, c_2, c_3)$. Апроксимація статичної характеристики із максимальною відносною похибкою 8% має вигляд

$$U_N \approx F_{UR} = U_f (3.54 - 2.1 \exp(-0.37(R_N - 0.8))). \quad (4.13)$$

Функцію F_{UR} використано як базисну в подальших апроксимаціях.

Оскільки перехідні процеси в об’єкті є досить складними, то застосовано регуляризаційну декомпозицію вихідного сигналу в часовій області на повільні та швидкі компоненти, як це обґрунтовано у параграфі 4.5.

Макромодельовання виконували окремо для повільних U_{NN} та швидкозмінних U_{NV} складових вихідного сигналу U_N , виділених лінійними фільтрами першого порядку зі сталими часу 0.1 с.

Низькочастотну макромодель ідентифікували на множинах дискретних часових відліків змінних U_{NN} , U'_{NN} , F_{UR} , F_{UR}/U_f , причому апроксимаційні багатовимірні степеневі поліноми будували щодо цих змінних. Відповідна лінійна апроксимаційна задача має вигляд

$$\min_a \sum_{i=0}^{600} ((U'_{NN})^i - (a_1 U_{NN}^i + a_2 F_{UR}^i + a_3 U_{NN}^i F_{UR}^i / U_f^i + a_4 (F_{UR}^i)^2 / U_f^i))^2, \quad (4.14)$$

де верхній індекс i позначає відлік відповідної змінної у момент часу $(0.1*i)$ секунд.

Максимальної відносної похибки реакції 5% досягнуто вже у рекурсивній макромоделі першого порядку при $a_1=-a_2$ та $a_3=-a_4$:

$$U'_{MN} = (-0.7348 + 0.1415 \cdot F_{UR} / U_f)(U_{MN} - F_{UR}). \quad (4.15)$$

Зауважимо, що в таким чином збудованій макромоделі відтворюється статична характеристика (4.13): $U_{MN}=F_{UR}$ при $U'_{MN}=0$.

Структура макромоделі (4.15) повністю відповідає теоремі 2 з розділу 3 для вхідного сигналу F_{UR} та вихідного сигналу U_{MN} .

Аналогічно формулюють апроксимаційну задачу для високочастотної макромоделі щодо змінних U_{NV} , U'_{NV} та $F'_{UR}=dF_{UR}/dt$ в околах моментів часу 1, 10, 20, 30, 40 та 50 секунд:

$$\min_b \sum_{i=in}^{in+100} ((U'_{NV})^i - (b_1 U_{NV}^i + b_2 U_{NV}^i F_{UR}'^i + b_3 (F_{UR}'^i)^2))^2, \quad (4.16)$$

де $in=100, 1000, 2000, 3000, 4000, 5000$, а верхній індекс i позначає відлік відповідної змінної у момент часу $(0.01*i)$ секунд.

Для високочастотної макромоделі максимальна відносна похибка реакції у вказаних ділянках апроксимації становить 12%:

$$U'_{MV} = -395.3U_{MV} + 0.2341U_{MV}F'_{UR} + 0.3254F'_{UR} - 0.0003196(F'_{UR})^2. \quad (4.17)$$

Структура макромоделі (4.17) також відповідає загальній структурі з теореми 2 (розділ 3) для вхідного сигналу F'_{UR} та вихідного U_{MV} . Повна макромодель утворена об'єднанням (4.13), (4.15) та (4.17) за блок-схемою, показаною на рис.4.7.

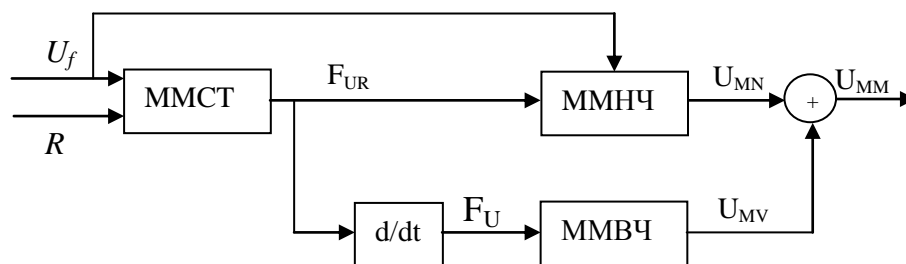


Рис.4.7. Блок-схема макромоделі синхронного генератора:
 ММСТ – статична макромодель (4.13); ММНЧ – макромодель з повільними рухами (4.15); ММВЧ – макромодель зі швидкими рухами (4.17).

Порівняльний розрахунок отриманої макромоделі та системи (4.11) засвідчив значно меншу жорсткість макромоделі. Максимальний крок інтегрування за методом Рунге-Кутта для макромоделі з рис.4.7 у 40 разів більший, ніж для системи (4.11) з однаковим значенням локальної похибки.

Кількість диференціальних рівнянь макромоделі – три, що на два менше, ніж у модельованому об'єкті.

У табл.4.3 наведено опис об'єкта та макромоделі вхідною мовою системи САНОС-ПК.

Таблиця 4.3. Завдання для порівняльного аналізу макромоделі та повного опису синхронного генератора рівняннями Парка-Горєва BEGI

Макромодельовання синхронного генератора
 DESC

* Опис генератора рівняннями Парка-Горєва ;

$$PSI = 0.24 * U_F - 0.22 * I_F;$$

$$I_F = 1.59 * PSI - 0.473 * I_D - 0.473 * I_{DV};$$

$$U_D = 0.185 * R_N * I_D - 1.293 * I_Q + 0.256E-2 * I_D - 0.364E-2 * D * PSI - 1.321 * I_{QV};$$

$$I_D' = 523.3 * U_D - 523.3 * R_N * I_D - 314 * I_Q;$$

$$U_Q = 1.269 * I_D + 0.195 * R_N * I_Q + 0.341E-2 * I_{QV} + 2.612 * I_F + 1.305 * I_{DV};$$

$$I_Q' = 523.3 * U_Q - 523.3 * R_N * I_Q + 314 * I_D;$$

$$I_{DV}' = -42.7 * D * PSI - 0.71 * D * I_D - 11.3 * I_{DV};$$

$$I_{QV}' = -0.799 * D * I_Q - 0.825 * I_{QV};$$

$$U_N = \text{SQRT}(U_D * U_D + U_Q * U_Q); \quad * \text{ Вихідний сигнал генератора ;}$$

```

* Початкові умови та вхідні сигнали ;
INIT: PSI(0.282),ID(-1.36),IQ(0.47),IDV(0),IQV(0) ;;
TABLE:'RN'0/0.8,20.0/0.8,20.1/100,50/100,50.1/0.8,60/0.8;
TABLE:'UF'0/1,1/1,1.1/1.6,10/1.6,10.1/1,30/1,30.1/0.8,40/0.8,/ 40.1/1,60/1;;
RN='RN'; UF='UF';
* Статична макромодель ;
FR=UF*(3.54-2.1*EXP(-0.37*(RN-0.8)));
* Низькочастотна макромодель ;
UMN'=(-0.7348 +0.1415*FR/UF)*(UMN-FR);
* Високочастотна макромодель ;
UMV'=(0.3254*DFR -0.0003196*DFR*DFR -395.3*UMV +0.2341*UMV*DFR);
UR1`=DFR; DFR=10k*(FR-UR1);
* Сумарна макромодель ;
UMM=UMN+UMV;;;
TRAN
TMAX=60; HS=0.01; ERMAX=0.01;
OUTPUT: RN,UF,UN,UMM, XMAX=60,YMIN=0.5,YMAX=4;;
END

```

Перехідні процеси в об'єкті та макромоделі, розраховані системою САНОС-ПК за завданням у табл.4.3, показані на рис.4.8.

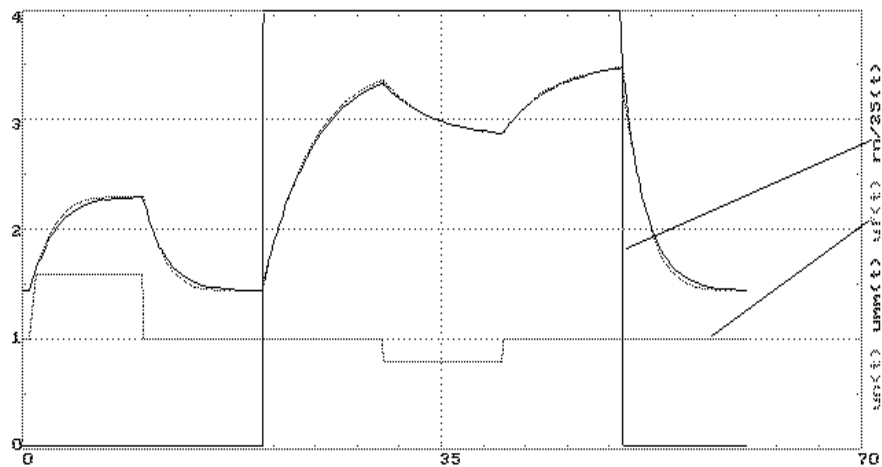


Рис.4.8. Перехідні процеси в об'єкті (електромашинний генератор) та його макромоделі.

На графік виведені часові залежності змінних U_F , U_N , U_{MM} , R_N . Вхідні змінні збігаються із зображеними на рис.4.6, лише змінна R_N показана зменшеною у 25 разів. Як бачимо, графіки змінних U_N та U_{MM} практично співпадають. Відносна похибка відтворення сигналу U_N не перевищує 4%.

На рис.4.9 у збільшеному масштабі відображені швидкі рухи в об'єкті та макромоделі в околі моменту часу 20 с.

Статичні характеристики як базисні функції з успіхом використані з метою регуляризації для апроксимації нелінійностей у багатовимірній нерекурсивній макромоделі операційного підсилювача (див. 3.2.2).

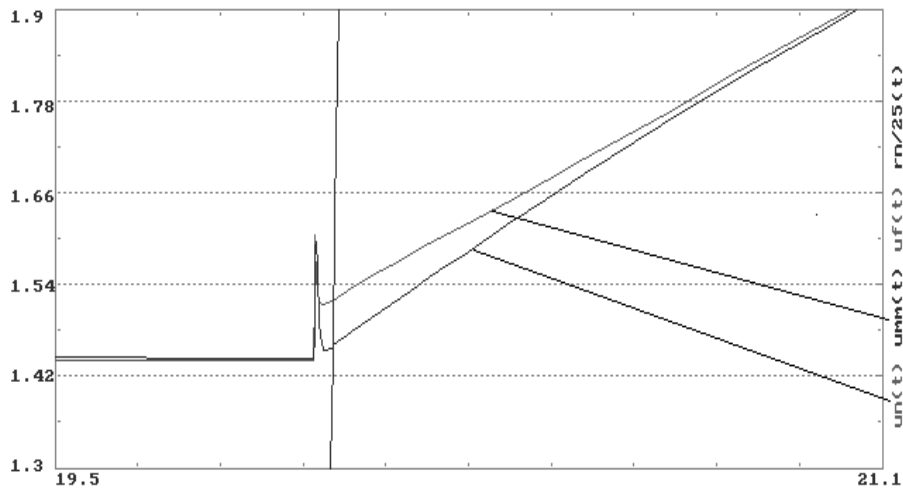


Рис.4.9. Фрагмент швидких рухів в об'єкті U_N та макромоделі U_{MM} .

4.3. ФУНКЦІОНАЛЬНЕ РОЗБИТТЯ ЗАДАЧІ. МАКРОМОДЕЛЬ БЛОКА КОЛЬОРУ ТЕЛЕВІЗІЙНОГО ПРИЙМАЧА

Будь-яка апріорна інформація про об'єкт моделювання може суттєво полегшити процес побудови макромоделі. Зокрема, завжди доцільним є розбиття задачі на підзадачі за функціональними ознаками, якщо це можливо. Саме такий підхід дав змогу збудувати макромодель основної частини сучасного телевізійного приймача – блока кольору.

Під час моделювання телевізійних приймачів та їхніх окремих частин на ЕОМ виникає задача перетворення повного кольорового телевізійного сигналу (*ПКТС*) в сигнали керування кольоровим кінескопом. Ми опишемо деякі результати макромодельовання чотирьох надвеликих інтегральних схем (НВІС), що опрацьовують *ПКТС* у стандарті SECAM.

Вхідними сигналами є *ПКТС* і сигнал синхронізації *SSC*. Дані для моделювання взяті з [65], а також із результатів прямих розрахунків повних принципових схем фрагментів НВІС у системі САНОС-ПК. Загальна функціональна схема всіх чотирьох НВІС показана на рис.4.10.

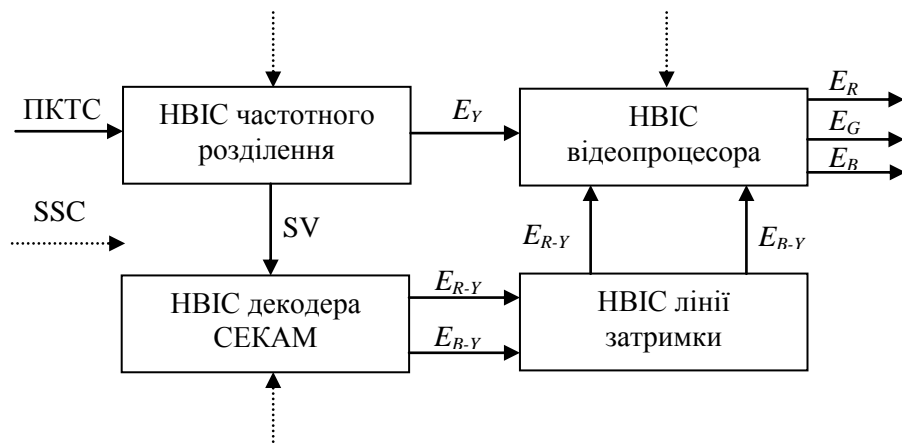


Рис.4.10. Схема зв'язків надвеликих інтегральних схем (НВІС), що опрацьовують повний кольоровий телевізійний сигнал (*ПКТС*).

Опишемо коротко роботу НВІС .

ПКТС – це суперпозиція сигналу яскравості E_Y та сигналу кольору SV , що несе інформацію про кольорові складові сигналу у вигляді частотно-модульованих коливань з різними значеннями несучої частоти в сусідніх стрічках. НВІС частотного розділення лінійною фільтрацією виділяє з *ПКТС* сигнал кольору SV та сигнал яскравості E_Y . НВІС декодера розпізнає “колір” стрічки та виконує частотне детектування сигналу кольору SV . НВІС лінії затримки вирівнює в часі кольорорізницеві сигнали E_{R-Y} та E_{B-Y} . НВІС відеопроцесора із сигналів E_Y , E_{R-Y} та E_{B-Y} формує сигнали кольору E_R , E_G і E_B .

Розглянемо роботу блока кольору детальніше на функціональному рівні.

ПКТС складається із стрічок двох видів – так званих “синіх” і “червоних”, що розрізняються частотою несучої сигналу кольору. Як входні дані для блока кольору взяті дві послідовні стрічки *ПКТС* типу “кольорових смуг”, які містять по 2048 відліків сигналу [66].

На рис.4.11 і 4.12 наведені часові діаграми цих двох стрічок разом із відповідними кольорорізницевими частотно-модулюючими сигналами, а на рис.4.13 – сигнал “червоної” стрічки разом із сигналом яскравості. Кольорорізницеві сигнали містять значні перехідні викиди, призначені для компенсації перехідних процесів у частотному детекторі.

На виході блока кольору утворюються сигнал яскравості E_Y , три кольорорізницеві сигнали E_{R-Y} , E_{G-Y} , E_{B-Y} , а також сигнали трьох основних кольорів – E_R , E_G , E_B .

ПКТС надходить у два канали – канал яскравості та канал кольору. Структурні схеми цих двох каналів зображені відповідно на рис.4.14 і 4.15.

У каналі яскравості (рис.4.14) сигнал проходить через широкосмугову лінію затримки (ШЛЗ), яка слугує для вирівнювання часу поширення сигналів у широкосмуговому каналі яскравості та вузькосмуговому каналі кольору. Час затримки залежить від ширини смуг пропускання каналів і змінюється в межах 0.4-0.8 мкс. Далі сигнал надходить у режекторний фільтр (РФ), призначений для придушення сигналу кольору в каналі яскравості.

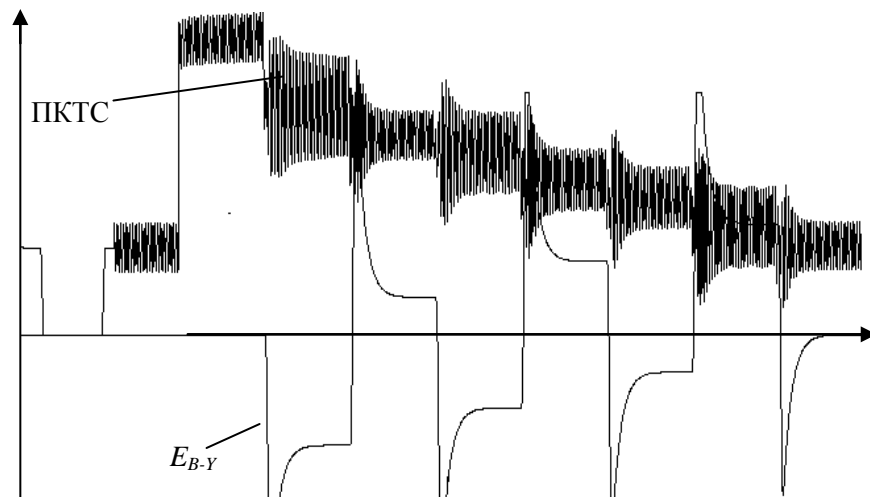


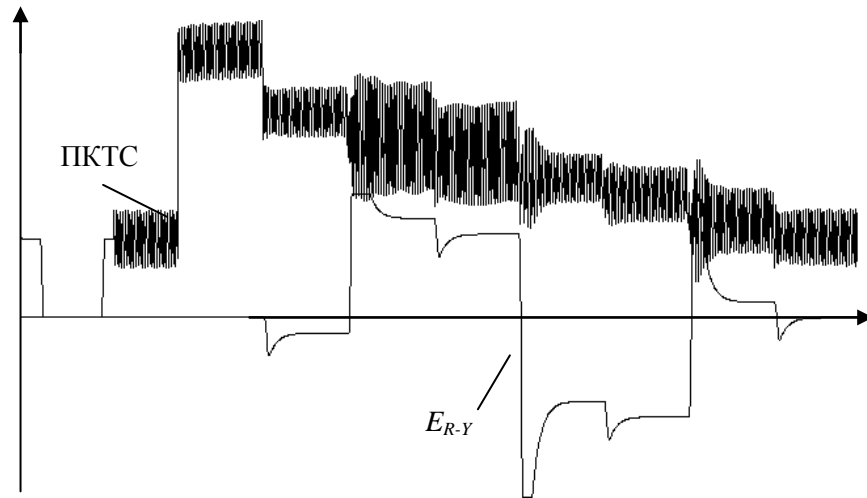
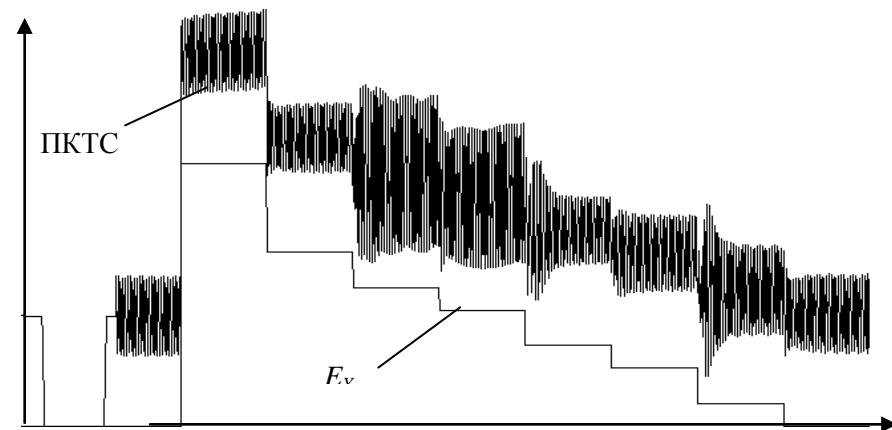
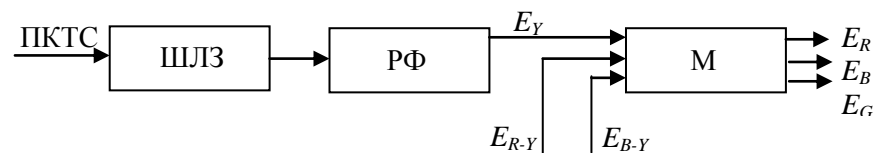
Рис.4.11. ПКТС “синьої” стрічки з сигналом E_{B-Y} Рис.4.12. ПКТС “червоної” стрічки з сигналом E_{R-Y} Рис.4.13. ПКТС “червоної” стрічки з сигналом E_Y .

Рис.4.14. Блок-схема формування сигналів $E_Y(t)$, $E_R(t)$, $E_G(t)$, $E_B(t)$:
 ШЛЗ – широкопasmовна лінія затримки на 0.4–0.8 мкс; РФ - режекторний фільтр для придушення піднесучих сигналів кольору; М – схема “матрицювання”.

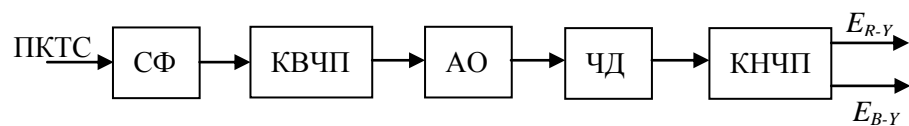


Рис.4.15. Блок-схема формування сигналів $E_{R-Y}(t)$ і $E_{B-Y}(t)$:
 СФ – смуговий фільтр для виділення піднесучих сигналів кольору;
 КВЧП, КНЧП – коректори попередніх високо- та низькочастотних
 спотворень; АО – амплітудний обмежувач; ЧД – частотний детектор.

До каналу кольору (рис.4.15) сигнал надходить після відеодетектора через смуговий фільтр (СФ), який виділяє частоти піднесучих сигналів кольору. Далі сигнал проходить коректор високочастотних попередніх спотворень (КВЧП) з комплексною функцією передачі

$$K_{\text{КВЧП}} = (1 + j \cdot 1.26F) / (1 + j \cdot 16F); \quad (4.18)$$

де $F = f/f_0 - f_0/f$; f – частота сигналу в кГц; $f_0 = 4286 \pm 20$ кГц.

Після КВЧП сигнал потрапляє на амплітудний обмежувач (АО) для компенсації нерівномірності АЧХ і ліквідації імпульсних завад, а далі – на частотний детектор (ЧД) і коректор низькочастотних попередніх спотворень (КНЧП) з комплексною функцією передачі

$$K_{\text{КНЧП}} = (1 + j \cdot f/f_1) / (1 + j \cdot f/(k \cdot f_1)); \quad (4.19)$$

де f – частота сигналу; $f_1 = 85$ кГц; $k = 3$.

На виході КНЧП утворюються два кольорорізницевої сигнали E_{R-Y} і E_{B-Y} . Третій кольоро-різницевий сигнал E_{G-Y} можна отримати, використовуючи таку формулу:

$$E_{G-Y} = -(0.51E_{R-Y} + 0.19E_{B-Y}). \quad (4.20)$$

Сигнали E_Y , E_{R-Y} , E_{G-Y} , E_{B-Y} опрацьовують у схемі “матрицювання” (М), яку описують співвідношення:

$$\begin{aligned} E_R &= E_{R-Y} + E_Y; \\ E_G &= E_{G-Y} + E_Y; \\ E_B &= E_{B-Y} + E_Y; \end{aligned} \quad (4.21)$$

На виході блока М отримують сигнали трьох основних кольорів E_R , E_G , E_B , які подають на відповідні катоди електронних прожекторів кінескопа.

Макромодель блока кольору створювали на основі макромоделей окремих функціональних блоків.

Математичні моделі функціональних блоків синтезовані у вигляді алгебро-диференціальних рівнянь методами математичного макромодельовання за множинами вхідних та вихідних сигналів блока. Отриману модель у цілому описано вхідною мовою системи схемотехнічного проектування САНОС-ПК, а часові залежності для будь-яких сигналів знаходимо сумісним числовим інтегруванням сукупної системи алгебро-

диференціальних рівнянь. Функціональна блок-схема макромоделі зображена на рис.4.16.

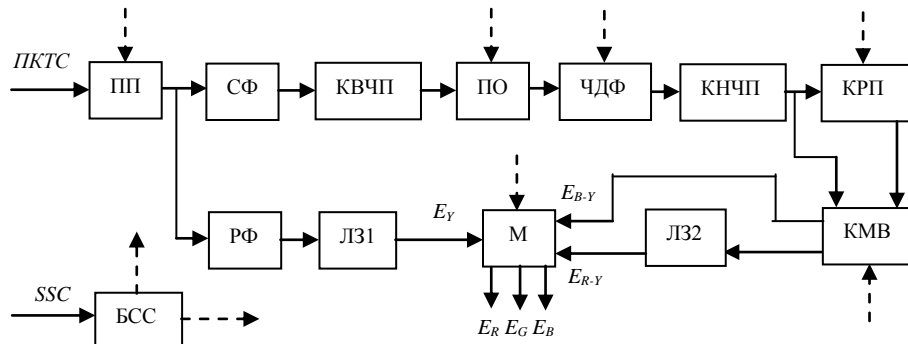


Рис.4.16. Блок-схема макромоделі блока кольору

Наведемо рівняння всіх функціональних блоків з рис.4.16, причому позначатимемо вхідний сигнал кожного блоку – U , вихідний сигнал – Y , внутрішні сигнали – X та Z .

Робота всіх блоків синхронізована імпульсами, що утворюються із стандартного синхроімпульсу SSC у блоці синхронізації БСС. Часові діаграми всіх синхроімпульсів показані на рис.4.17.

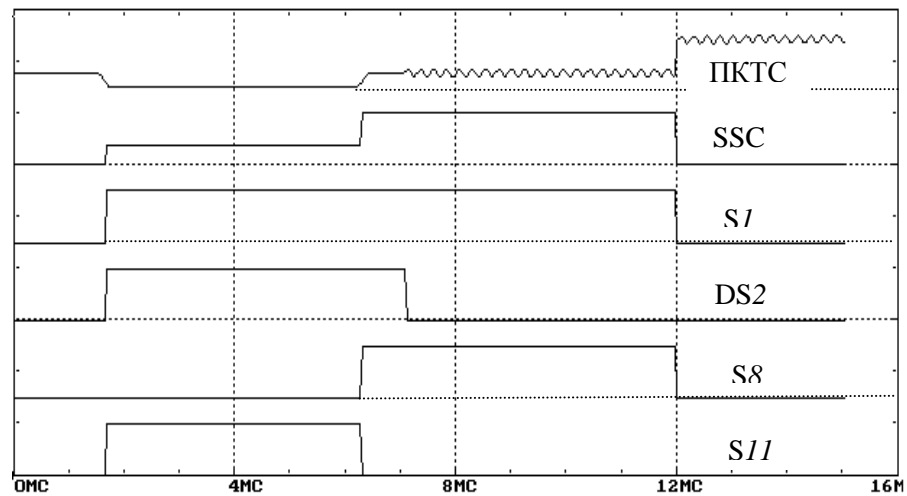


Рис.4.17. Часові діаграми синхроімпульсів

Блок синхронізації БСС містить одне диференціальне рівняння для виділення переднього фронту імпульсу $S1$ та формування імпульсу $DS2$:

$$\dot{Y} = 5 \cdot 10^5 (S1 - Y); \quad DS2 = 0.5(\text{sinh}(Y - 10^5) + 1). \quad (4.22)$$

Співвідношення (4.22) забезпечують тривалість імпульсу $DS2$ 1мкс. Решта співвідношень блока БСС алгебричні (див. табл.4.4).

Підсилювач з прив'язкою ПП попередньо підсилює $PKTC$ у 10 разів та віднімає деяку складову (рівень прив'язки) U_p , сталу на протязі однієї стрічки, однак змінну від стрічки до стрічки. Значення рівня прив'язки U_p

з'ясовується на початку стрічки за час дії імпульсу $DS2$. Рівняння підсилювача з прив'язкою мають вигляд

$$\begin{aligned}\dot{Y} &= 1/\tau_1 \cdot (k(U - X) - Y); \\ \dot{X} &= DS2/\tau_2 \cdot (Y - U_p); \end{aligned} \quad (4.23)$$

де $\tau_1 = 0.25 \cdot 10^{-7}$ – стала часу підсилювача; $\tau_2 = 10^{-7}$ – стала часу встановлення рівня прив'язки; $k=10$ – коефіцієнт підсилення.

Числові значення параметрів ПП отримані під час дослідження схеми ПП на ЕОМ.

Сигнал яскравості виділяють з *ПКТС* режекторним фільтром РФ. Відповідне рівняння, розраховане за АЧХ з [65], має вигляд

$$\ddot{Y} + 5.421 \cdot 10^6 \dot{Y} + 7.402 \cdot 10^{14} Y = \ddot{U} + 7.402 \cdot 10^{14} U. \quad (4.24)$$

Смуговий фільтр СФ реалізує відповідну АЧХ з [65] та виділяє сигнали піднесучих кольору:

$$\ddot{Y} + 5.421 \cdot 10^6 \dot{Y} + 7.402 \cdot 10^{14} Y = 5.421 \cdot 10^6 \dot{U}. \quad (4.25)$$

Фільтр високочастотної корекції попередніх спотворень КВЧП відповідає АФЧХ (4.18), його описує рівняння

$$\ddot{Y} + 1.682 \cdot 10^6 \dot{Y} + 7.252 \cdot 10^{14} Y = 7.875 \cdot 10^{-2} \ddot{u} + 1.682 \cdot 10^6 \dot{U} + 5.711 \cdot 10^{13} U. \quad (4.26)$$

Підсилювач-обмежувач ПО виконує безінерційне обмеження сигналу до амплітуди 1.0 зі збереженням сталої складової U_p :

$$Y = (1 - S1)(\text{sign}(U - U_p) + U_p). \quad (4.27)$$

Частотний детектор сигналу кольору ЧДФ є основним та функціонально найскладнішим блоком. У сучасних телевізійних приймачах цей блок є системою фазового автоналаштування частоти (ФАНЧ), що працює в режимі частотного детектора.

Рівняння безфільтрової системи ФАНЧ з ідеальним перемножувачем, що працює як фазовий детектор, та керованим генератором прямокутних імпульсів, модель якого містить ідеальний інтегратор, мають такий вигляд:

$$\begin{aligned}Y &= UX; \\ X &= \text{sign}(\sin(2\pi \cdot 4.33 \cdot 10^6 t + Z)); \\ \dot{Z} &= 1.44 \cdot 10^6 Y. \end{aligned} \quad (4.28)$$

Вихідний сигнал ФАНЧ фільтрується ФНЧ другого порядку (4.29) та режекторним фільтром на частоту 8.66 МГц (4.30):

$$\ddot{Y} + 2 \cdot 10^7 \dot{Y} + 2 \cdot 10^{14} Y = 2 \cdot 10^{16} U; \quad (4.29)$$

$$\ddot{Y} + 2.42 \cdot 10^8 \dot{Y} + 2.96 \cdot 10^{15} Y = \ddot{U} + 2.96 \cdot 10^{15} U. \quad (4.30)$$

Фільтр корекції низькочастотних попередніх спотворень КНЧП відповідає АФЧХ (4.19):

$$\begin{aligned} \dot{X} &= -5.34 \cdot 10^5 X + 3.56 \cdot 10^5 U; \\ Y &= X + 0.333U. \end{aligned} \quad (4.31)$$

Зауважимо, що основна частота керованого генератора ФАНЧ при $Z=0$ становить 4.33 МГц. Унаслідок цього вихідний сигнал при детектуванні сигналу кольору містить сталу складову, знак якої відповідає різниці основної частоти піднесучої кольору та частоти 4.33 МГц. Для “червоної” стрічки ця складова від’ємна (4.25 МГц), а для “синьої” – додатна (4.406 МГц). Сталу складову виділяє компаратор під час “спалаху” частоти піднесучої у кінці імпульсу стрічкової синхронізації (синхроімпульс $S8$).

Компаратор з пам’яттю КРП працює як ФНЧ першого порядку, стала часу якого дорівнює 1мкс при $S8=1$ та 1с при $S8=0$:

$$\dot{Y} = (1 + 10^6 \cdot S8)(U - Y); \quad U_k = Y. \quad (4.32)$$

Отже, компаратор “запам’ятовує” рівень U_k сталої складової вихідного сигналу ЧД на момент закінчення синхроімпульсу $S8$ та “пам’ятає” його до початку наступної стрічки.

Комутатор з відніманням КМВ із вихідного сигналу ЧД віднімає сталу складову U_k та подає на вихід E_{R-Y} або E_{B-Y} , залежно від знаку U_k :

$$\begin{aligned} E_{R-Y} &= 0.5(1 + \text{sign}(U_k))(1 - S1)(U - U_k); \\ E_{B-Y} &= 0.5(1 - \text{sign}(U_k))(1 - S1)(U - U_k). \end{aligned} \quad (4.33)$$

Лінію затримки різницевого кольорового сигналу ЛЗ2 моделює запис сигналу E_{R-Y} у бібліотеку системи САНОС-ПК. У випадку моделювання обробки сигналу E_{B-Y} з бібліотеки викликається сигнал E_{R-Y} та використовується в обчисленнях одночасно із сигналом E_{B-Y} . Завдяки цьому імітується затримка різницевого сигналу на час тривалості однієї стрічки 64 мкс.

Нарешті, блок “матрицювання” М реалізує арифметичні співвідношення (4.20) та (4.21), а також затримку сигналу E_Y у ЛЗ1 на 0.6 мкс згідно з рівнянням

$$Y(t) = U(t - 0.6 \cdot 10^{-6}) . \quad (4.34)$$

Сукупна система алгебро-диференціальних рівнянь описана вхідною мовою системи САНОС-ПК і наведена у табл.4.4.

Таблиця 4.4. Макромодель блока кольору телевізійного приймача
 BEGI
 Опрацювання повного телевізійного сигналу системи SECAM.
 DESC

```
PTS='PTS_RR'/100; * Повний телевізійний сигнал із бібліотеки ;
* Блок селекції основного синхроімпульсу SSC ;
SSC=IF(T.LT.12mc)(SSC1)(0); SSC1=IF(T.LT.4mc)(SSC2)(SSC3);
SSC2=IF(PTS.LT.1)(4)(0); SSC3=IF(PTS.GT.1)(SSC4)(4);
SSC4=if(T.lt.11mc)(10.2)(4);
* Блок селекції допоміжних синхроімпульсів ;
S1=IF(SSC.GT.2)(1)(0); * Синхроімпульс S1 ;
XS`=DS; DS=KD*(S1-XS); KD=5E5*S1+1; KDP=KD/1.3; KDR=KD/5;
DS1=IF(DS.GT.KDP)(1)(0); * Передній фронт S1 ;
DS2=IF(DS.GT.KDR)(1)(0); * Розширений передній фронт S1 ;
S8=IF(SSC.GT.6)(1)(0); * Строб спалаху ;
S18=S1-S8; S1I=1-S1; * Інверсний S1 ;
S81`=5E5*(S8-S81); S2=IF(S81.GT.0.05)(S18)(0); * Строб компаратора;
S9=S8+S2; * Строб спалах+компаратор;
S11=S1-S9; S12=1-S11; * S1 до спалаху та інверсний S1 до спалаху;
* Блок частотного розділення ;
* Підсилювач з прив'язкою ;
uc`=DS2*1E7*(vp-up); up=1mc; * Рівень прив'язки ;
vp`=4E7*(1*(PTS-uc)-vp); * vp - вихідний сигнал ;
* Режекторний фільтр частот піднесучих кольору ;
YS=vp-SV; * для ліпшого виділення сигналу яскравості ;
* Фільтр нижніх частот (виділення сигналу яскравості Y) ;
Y`=Y1; Y1`=Y2; Y2=4.02e14*YS-2.904e7*Y1-4.02e14*Y;
* Смуговий фільтр(виділення піднесучих кольору) ;
SV1`=SV; SV`=5.421e6*vp-5.421e6*SV-7.402e14*SV1;
* Блок декодера сигналів кольору ;
* Підсилювач з прив'язкою ;
UC`=DS2*1E7*(VP-UP); UP=1mc; * Рівень прив'язки ;
VP`=4E7*(2*(SV-UC)-VP); * VP - сигнал кольору ;
* Підсилювач-обмежувач (UUA-обмежений сигнал кольору) ;
UU1=S12*30*(VPUP); UUA=ABS(UU1);
UUA=IF(UUA.GT.1)(UU1/UUA+UP)(UU1+UP);
* Детектор сигналу кольору – ФАНЧ ;
UDV=UUA*UPG; * Ідеальний перемножувач ;
UPG1=S12*10*SIN(ARG+FIV); * Сигнал автогенератора ;
ARG=2*PI*4.33mg*(T-6.4375e-6);
```

```

UPG2=IF(UPG1.GT.1)(1)(UPG1);
UPG=IF(UPG1.LT.-1)(-1)(UPG2); * Обмежений сигнал автогенератора;
UF1`=1e7*((UDV-sv)-UF1); UF`=2e7*(UF1-UF); * Фільтр другого порядку ;
UD=KI*2e-6*UF; * Вихідний сигнал детектора - UD;
sv1`=sv; FF=2*PI*8.66mg; * Смуговий фільтр другого порядку на 8.66 МГц;
sv`=2.421e8*(UDV-sv)-FF*FF*sv1;
FIV`=S12*(UDV) *KI; KI=2e7; * Інтегратор фази автогенератора;
UBI`=(1ml+1mg*S1) *(UD-UBI); * Компаратор ;
YB=IF(UBI.LT.0)(S1I*(UD-UBI))(0); * Комутатор ;
YR=IF(UBI.GT.0)(S1I*(UD-UBI))(0);;
TRAN
ZERO; HS=0.1mc; TMAX=64mc; ERMAX=0.05; hmax=0.05mc; HMIN=1E-20;
output: PTS,SSC,UD,Y,YB,YR, XMIN=0,XMAX=64mc,YMIN=-
10,YMAX=10;;
END

```

Загалом макромодель блока кольору складається з 15 диференціальних рівнянь, на відміну від 300 у повній схемі. Результати виконання завдання з табл.4.4 наведені на рис.4.18-4.20.

На рис.4.20 показано вихідний сигнал яскравості E_Y макромоделі, а на рис.4.18, 4.19 – сигнали E_{R-Y} та E_{B-Y} . Відносна похибка відтворення показаних сигналів модуляції не перевищує 15%, крім ділянок перехідних викидів сигналів E_{R-Y} та E_{B-Y} . на рис.4.11 та 4.12.

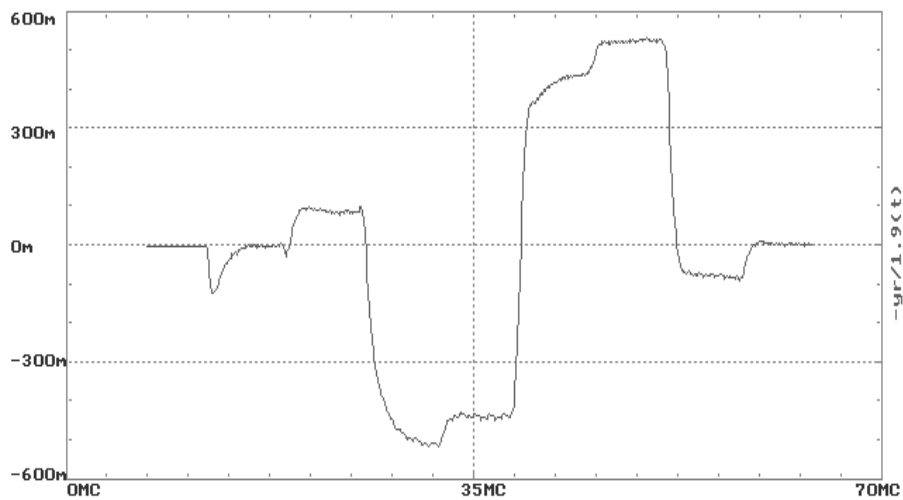
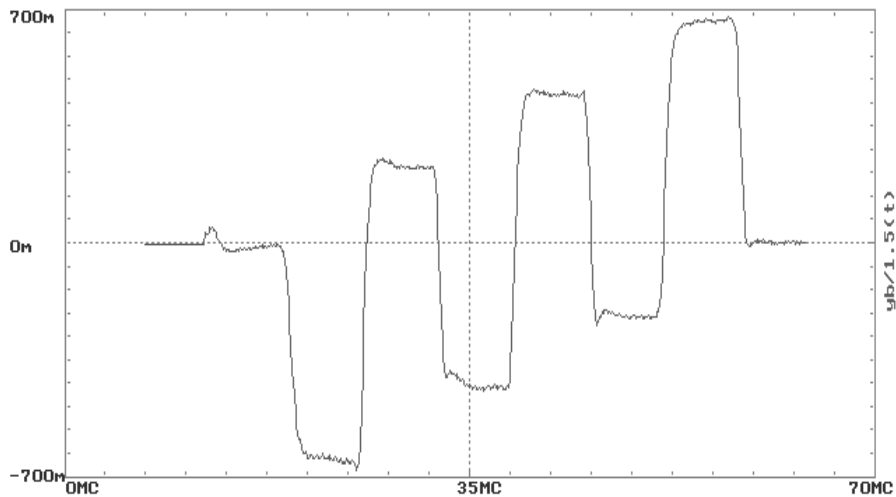
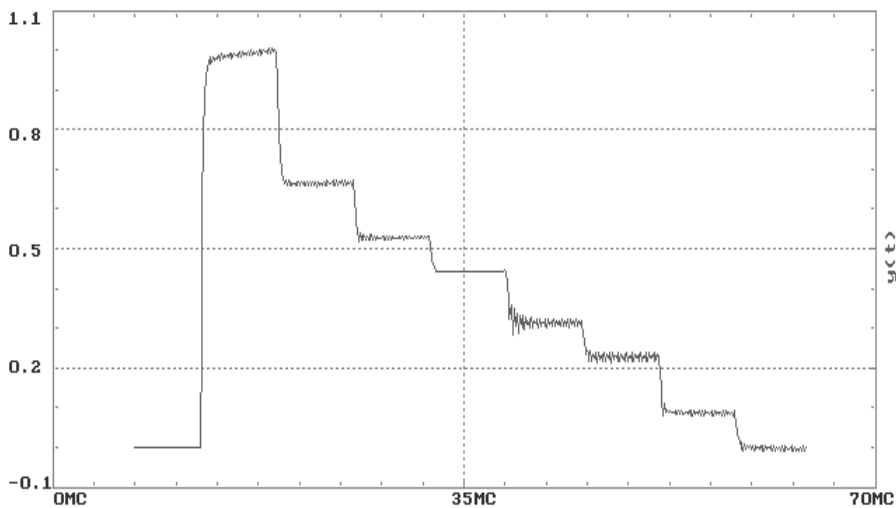


Рис.4.18. Кольорорізницевий сигнал E_{R-Y}

Рис.4.19. Кольорорізницевий сигнал E_{B-Y} Рис.4.20. Сигнал яскравості E_Y

4.4. ДЕКОМПОЗИЦІЯ АПРОКСИМОВАНОЇ ФУНКЦІЇ В ПРОСТОРІ АРГУМЕНТІВ

Ще один метод уникнення некоректності полягає в розбитті складної апроксимаційної задачі на низку простіших в області аргументів апроксимації, їх послідовному розв'язуванні і об'єднанні отриманих результатів.

Саме так чинять у разі багатовимірної сплайн-апроксимації. Однак безпосереднє використання сплайн-апроксимації степеневими поліномами невисокого порядку у випадку, коли кількість розбиттів велика, непридатне, оскільки у вхідних мовах систем схемотехнічного проектування нема адекватних засобів опису сплайн-апроксимацій. Тому розроблено алгоритм автоматичного розбиття області апроксимації на невелику – не більше п'яти – кількість підобластей, апроксимації в кожній із підобластей окремо, стикування отриманих апроксимацій та автоматичного вибору потрібної апроксимації залежно від поточного значення вектора аргументів.

Основою методу стали такі положення:

підобласті мають вигляд гіперкуль у багатовимірному евклідовому просторі базисних функцій $\bar{\varphi}(\bar{x})$ розмірності M ;

гіперкулі можуть мати різні радіуси, однак похибка апроксимації в усіх підобластях не перевищує заданої;

стикування апроксимацій у сусідніх підобластях забезпечене взаємним накладанням країв підобластей.

Кожну гіперкулю описують центр і радіус. На рис.4.21 показане схематичне розбиття області апроксимації на дві підобласті з центрами C_1 , C_2 та частковим накладанням країв підобластей.

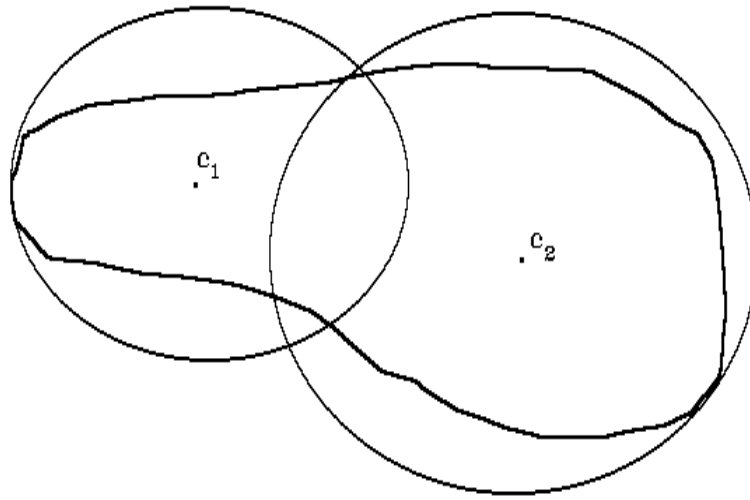


Рис.4.21. Розбиття області апроксимації на дві підобласті

Координати $(\xi_{i1}, \dots, \xi_{iM})$ центра C_i підобласті Ω_i визначають як середнє максимального і мінімального значень базисних функцій $\xi_j = \varphi_j(\bar{x})$ у межах підобласті за кожною із базисних функцій:

$$\xi_{ij} = 1/2(\max_{\Omega_i}(\xi_j) + \min_{\Omega_i}(\xi_j)); \quad j = \overline{1, M}. \quad (4.35)$$

Радіус i -ї гіперкулі – це максимальна відстань від центра в межах підобласті

$$R_i = \max_{\Omega_i} \sqrt{\sum_{j=1}^M (\xi_j - \xi_{ij})^2}, \quad (4.36)$$

де максимум шукають за всіма точками, які утворюють підобласть Ω_i .

Поточна точка з координатами (ψ_1, \dots, ψ_M) належить до i -ї підобласті, якщо її відстань від центра C_i менша R_i :

Використання описаного методу усунення некоректності дозволяє провести апроксимацію складних нелінійних функцій багатьох аргументів. У Додатку 8 наведено текст програми, що автоматично розбиває задану область у просторі базисних функцій на чотири підобласті, апроксимує регуляризованим методом найменших квадратів у кожній із підобластей і формує описи апроксимацій у вигляді числових типів відповідної моделі мовою САНОС-ПК у бібліотеці системи. Опис моделі мовою системи САНОС-ПК, що відповідає розбиттю простору апроксимації на чотири підобласті з автоматичним вибором апроксимації для найближчої підобласті, є в Додатку 9.

4.5. ДЕКОМПОЗИЦІЯ АПРОКСИМОВАНОЇ ФУНКЦІЇ В ЧАСОВІЙ ТА ЧАСТОТНІЙ ОБЛАСТЯХ

Ще один метод уникнення некоректності – розбиття складної апроксимаційної задачі на низку простіших шляхом декомпозиції апроксимованого сигналу в часовій та (або) частотній областях.

Загальна схема послідовної декомпозиції в часовій області зображена на рис.4.23.

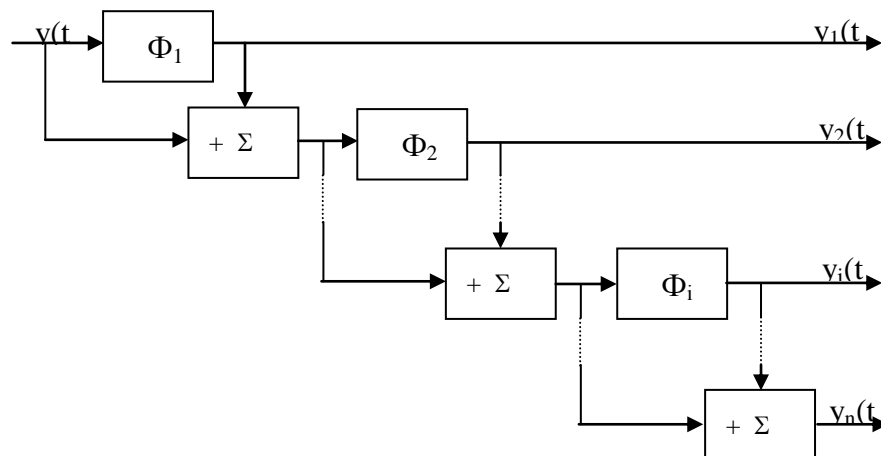


Рис.4.23. Декомпозиція сигналу в часовій області:
 Φ_i – оператор; Σ – алгебричний суматор

Очевидно, для схеми з рис.4.23 виконується рівність

$$y(t) = \sum_{i=1}^n y_i(t), \quad (4.39)$$

а отже сума апроксимацій для $y_i(t)$ утворює апроксимацію для $y(t)$, похибка якої дорівнює сумі похибок для $y_i(t)$. Залишається так підібрати оператори Φ_i , щоб збудувати наближення для всіх $y_i(t)$, $i=1, \dots, n$ було простіше, ніж для $y(t)$ одразу.

Іноді природніше розділяти сигнал у частотній області. На рис. 4.24 зображено схему такої декомпозиції, що, по суті, відрізняється від попередньої наявністю блоків прямого (FT) та зворотнього (IFT) перетворень Фур'є.

Якщо знехтувати похибками перетворення Фур'є, то внаслідок його лінійності з очевидної рівності $Y(\omega) = \sum_{i=1}^n Y_i(\omega)$ випливає рівність (4.39).

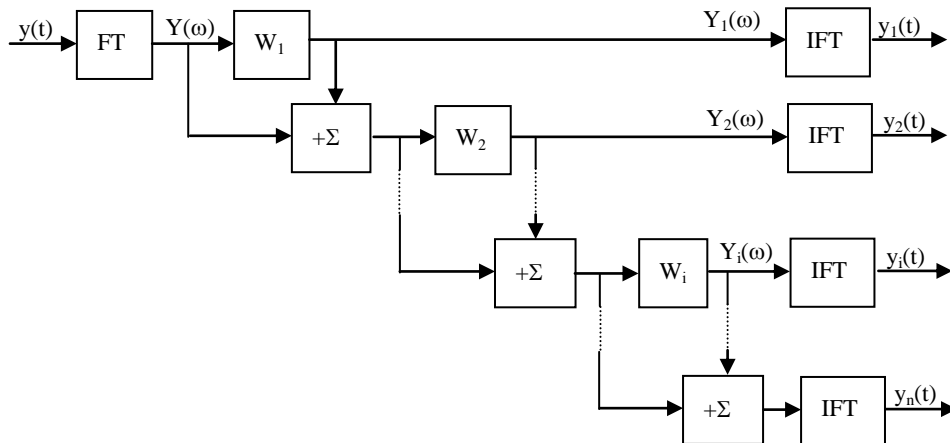


Рис.4.24. Декомпозиція сигналу в частотній області:

W_i – довільні вікна, що виділяють фрагменти відповідних Фур'є-зображень.

Для обох схем нехтування будь-якими складовими робить рівність (4.39) наближеною, тобто дає змогу контролювано спрощувати сигнал. Можливі комбінації обох способів декомпозиції на довільному етапі послідовної процедури залежно від характеру сигналу.

Якщо в сигналі явно вирізняються швидкі та повільні рухи, то декомпозиція виглядає вельми природно. У випадку макромодельовання синхронного генератора (див. параграф 4.2) декомпозицію вихідного сигналу в часовій області на дві компоненти виконано найпростішим фільтром першого порядку. Роздільний синтез низькочастотної та високочастотної макромоделей суттєво спростив задачу.

4.6. РЕДУКЦІЯ АПРОКСИМУЮЧОГО ПОЛІНОМУ. МАКРОМОДЕЛЬ ЧАСТОТНОГО ДЕТЕКТОРА

Надлишкова розмірність вектора невідомих параметрів апроксимації \bar{a} може бути причиною некоректності задачі ідентифікації. З метою зменшення розмірності запропоновано метод вилучення зайвих членів (редукції) апроксимуючого поліному, тобто зменшення M у системі (4.1).

По суті, редукція апроксимаційного базису повинна наближати задачу до точної згідно з методом регуляризації на компактї, запропонованим та

розробленим школою Тіхонова [67]. Суттєва особливість полягає в тому, що для наших задач невідома компактна множина, яка містить шуканий розв'язок. Отже, запропонований далі метод редукції апроксимуючого полінома можна інтерпретувати як спосіб визначення деякої компактної множини базисних функцій, на якій задача апроксимації наближається до коректної.

Метод ґрунтується на емпіричному припущенні, що “зайві” базисні функції повинні виявляти себе найбільшою зміною відповідних коефіцієнтів апроксимації при невеликих збуреннях апроксимаційної задачі. Послідовне вилучення таких “зайвих” компонент задачі апроксимації може приводити до більш стійкої задачі, тобто регуляризувати її.

Метод зводиться до послідовності таких дій.

1. Знайти розв'язок системи (4.1) методом найменших квадратів:

$$\min_{\bar{a}} \sum_{i=1}^M (a_i \varphi_i(\bar{x}_j) - v_j)^2; \quad j = \overline{1, N}. \quad (4.40)$$

2. Ввести невеликі випадкові збурення у вектор \bar{v} і визначити збурений вектор \tilde{a} , що є розв'язком задачі

$$\min_{\tilde{a}} \sum_{i=1}^M (\tilde{a}_i \varphi_i(\bar{x}_j) - \tilde{v}_j)^2; \quad j = \overline{1, N}. \quad (4.41)$$

3. Оцінити відносні відхилення компонент вектора \bar{a} та знайти індекс r найбільшого з них за модулем:

$$|(a_r - \tilde{a}_r) / a_r| = \max_{k=1, M} |(a_k - \tilde{a}_k) / a_k|. \quad (4.42)$$

4. Обчислити діапазон відносних відхилень компонент вектора \bar{a} :

$$\delta = \max_{k=1, M} |(a_k - \tilde{a}_k) / a_k| - \min_{k=1, M} |(a_k - \tilde{a}_k) / a_k|.$$

5. Якщо $\delta < \delta_{\min}$, то припинити процес редукції.

6. Видалити з вектора базисних функцій $\bar{\varphi}$ компоненту φ_r і розв'язати редуктовану задачу:

$$\min_{\bar{a}} \sum_{i=1}^M (a_i \varphi_i(\bar{x}_j) - v_j)^2; \quad j = \overline{1, N}; \quad i \neq r. \quad (4.43)$$

7. Продовжити процес редукції з пункту 2.

Як бачимо, метод підлягає повній автоматизації. Треба лише вдало задати параметр δ_{\min} , що є критерієм некоректності апроксимаційної задачі.

У Додатку 6 наведено текст FORTRAN-програми, що реалізує метод редукції для вектора базисних функцій, компоненти якого є членами повного багатовимірного степеневого полінома третього степеня.

Опишемо застосування методу на прикладі макромодельовання частотного детектора. Частотний детектор може бути, з одного боку, простою задачею за своєю схемою та за порядком диференціальних рівнянь, що його описують. Однак з функціонального погляду він є складною системою. Особливо суттєвими є прояви некоректностей при макромодельованні частотного детектора. Зважаючи на ці особливості, побудову макромоделі частотного детектора може вважати тестовою задачею макромодельовання.

У праці [68] опубліковані результати побудови дискретних макромоделей амплітудного та частотного детекторів за ідеальними вхідними та вихідними сигналами з використанням мінімаксного критерію. Прояви некоректності, що названі “нестійкістю рішень”, нейтралізовані введенням додаткових обмежень на коефіцієнти апроксимації. Ці обмеження формують у вигляді нерівностей із застосуванням другого методу Ляпунова. Отримані макромоделі чутливі до кроку дискретизації.

З огляду на ці зауваження ми побудували неперервні макромоделі частотного детектора для реальних схем із застосуванням описаної вище процедури регуляризації [69]. Одна з найпростіших реальних схем частотного детектора показана на рис.4.25.

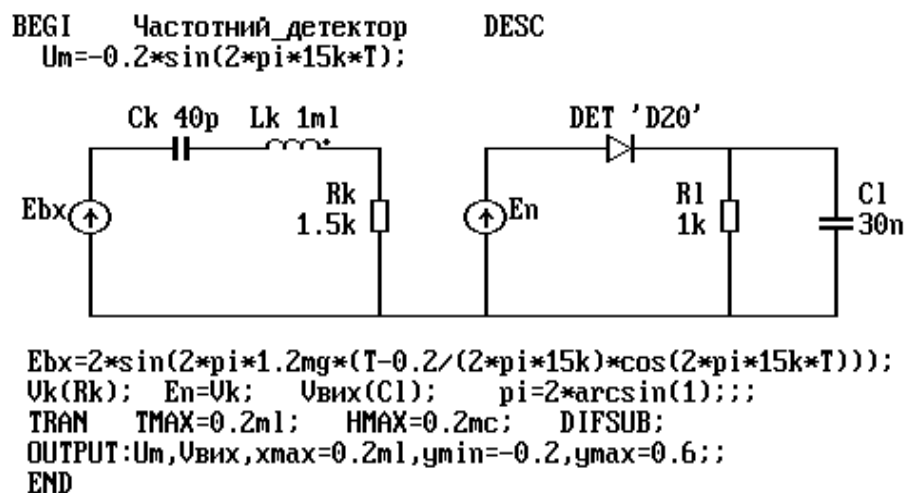


Рис.4.25. Схема частотного детектора

Частотно-модульований вхідний сигнал з несучою частотою 1.2 МГц та частотою модуляції 15 кГц заданий джерелом напруги $E_{вх}$, а вихідним сигналом є напруга на ємності $C1$. Струм у частотно-вибірному контурі $Ск$ -

Lk-Rk виділяється на опорі Rk та через повторювач Еп перетворюється детектором DET. Сигнал частотної модуляції виділяється у фільтрі Rl-Cl.

На рис.4.26 зображені сигнал модуляції U_m та вихідний сигнал детектора $V_{вих}$, розраховані системою САНОС-ПК за схемою з рис. 4.25.

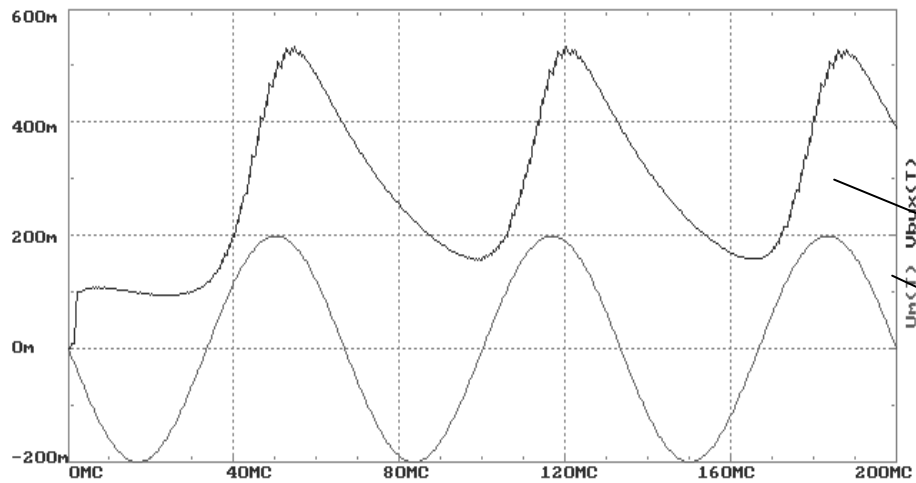


Рис.4.26. Сигнал модуляції U_m
та вихідний сигнал частотного детектора $V_{вих}$

Схема має помітну нелінійність декодування, що спотворює сигнал модуляції і створює додаткові труднощі під час макромодельовання.

Використання рекурсивної структури, що доведена теоремою 2 (рис.3.2), дало змогу суттєво спростити задачу ідентифікації. Вхідним сигналом макромоделі є напруга $E_{вх}=u$, компонентами вихідного вектора обрано вхідний струм $I_{вх}=y_1$ та вихідну напругу $V_{вих}=y_2$.

Загальна система рівнянь макромоделі має вигляд:

$$\begin{aligned}\dot{y}_1 &= y_3 / k_t; \\ \dot{y}_3 &= k_t^2 \cdot f_1(u_1, y_1, y_2, y_3); \\ \dot{y}_2 &= k_t \cdot f_2(u_1, y_1, y_2, y_3); \\ u_1 &= \dot{u}.\end{aligned}\tag{4.44}$$

Уже перші експерименти з ідентифікації макромоделі (4.44) засвідчили, що функція $f_1(u_1, y_1, y_2, y_3)$ з високою точністю є лінійною і структуру можна зведена до простішої:

$$\begin{aligned}\dot{y}_1 &= y_3 / k_t; \\ \dot{y}_3 &= k_t^2 \cdot (A_1 \cdot u_1 + A_2 \cdot y_1 + A_3 \cdot y_3); \\ \dot{y}_2 &= k_t \cdot f(u_1, y_1, y_2, y_3); \\ u_1 &= \dot{u}.\end{aligned}\tag{4.45}$$

де функція $f(\cdot)$ відповідає нелінійній безінерційній підсистемі, внутрішній вектор має компоненти $v_1 = u_1$ та $v_2 = f(\cdot)$, а k_t – нормувальний коефіцієнт.

Рівняння (4.45), блок-схема яких зображена на рис.4.27, повністю відповідають загальній моделювальній структурі з рис.3.2.

Рівняння лінійної підсистеми мають вигляд

$$\begin{aligned}\dot{y}_1 &= y_3 / k_t; \\ \dot{y}_3 &= k_t^2 (A_1 v_1 + A_2 y_1 + A_3 y_3); \\ \dot{y}_2 &= k_t v_2.\end{aligned}\tag{4.46}$$

Нелінійну безінерційну підсистему описують рівняння

$$\begin{aligned}v_1 &= u_1; \\ v_2 &= f(u_1, y_1, y_2, y_3).\end{aligned}\tag{4.47}$$

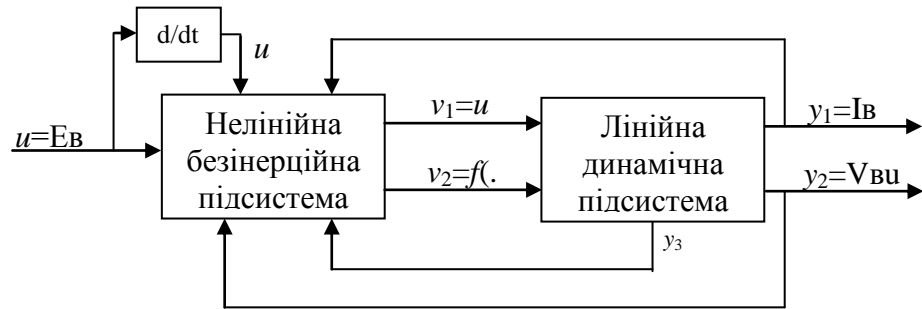


Рис.4.27. Блок-схема макромоделі частотного детектора

Систему (4.45) ідентифікували за алгоритмом оберненої лінійної підсистеми (див. параграф 3.3).

Спочатку було ідентифіковано лінійну підсистему (4.46).

Методом найменших квадратів для показаного на рис.4.25 вхідного сигналу $u_1 = E_v$, відповідного вихідного сигналу y_1 та його похідної y_3 при $k_t = 2 \cdot \pi \cdot 1e6$ розв'язком задачі (4.48)

$$\min_{A_1, A_2, A_3} \sum_{i=1}^{800} (\dot{y}_3(i) - k_t^2 (A_1 u_1(i) + A_2 y_1(i) + A_3 y_3(i)))^2\tag{4.48}$$

із максимальною відносною похибкою 3% для восьмиста дискретних значень змінних $u_1, y_1, y_3, dy_3/dt$ обчислено параметри лінійної підсистеми:

$$A_1 = 1.5915e-4; A_2 = -6.3326e-1; A_3 = -2.3613e-1.$$

Далі постає проблема апроксимації багатовимірним степеневим поліномом нелінійної функції чотирьох аргументів як розв'язок задачі

$$\min_{\bar{B}} \sum_{i=1}^{800} \left\| \dot{y}_2(i) - k_t \sum_{\substack{j,k,l,m=0 \\ j+k+l+m \leq d}}^d B_{jklm} u_1^j(i) y_1^k(i) y_2^l(i) y_3^m(i) \right\|. \quad (4.49)$$

171

Ця задача є основною для ідентифікації макромоделі. За методом найменших квадратів (4.49) набуває вигляду

$$\min_{\bar{B}} \sum_{i=1}^{800} (\dot{y}_2(i) - k_t \sum_{\substack{j,k,l,m=0 \\ j+k+l+m \leq d}}^d B_{jklm} u_1^j(i) y_1^k(i) y_2^l(i) y_3^m(i))^2. \quad (4.50)$$

Цю ж задачу за мінімаксним методом формулюють так:

$$\min_{\bar{B}} \max_i \sum_{i=1}^{800} \left| \dot{y}_2(i) - k_t \cdot \sum_{\substack{j,k,l,m=0 \\ j+k+l+m \leq d}}^d B_{jklm} \cdot u_1^j(i) \cdot y_1^k(i) \cdot y_2^l(i) \cdot y_3^m(i) \right|. \quad (4.51)$$

Повні рівняння макромоделі (4.45) з апроксимацією нелінійної функції багатовимірним степеневим поліномом такі:

$$\begin{aligned} \dot{y}_1 &= y_3 / k_t; \\ \dot{y}_3 &= k_t^2 (A_1 u_1 + A_2 y_1 + A_3 y_3); \\ \dot{y}_2 &= k_t \sum_{\substack{j,k,l,m=0 \\ j+k+l+m \leq d}}^d B_{jklm} u_1^j(i) y_1^k(i) y_2^l(i) y_3^m(i); \\ u_1 &= \dot{u}. \end{aligned} \quad (4.52)$$

Задачу (4.50) розв'язувала програма з Додатка 4, а задачу (4.51) – програма з Додатка 2.

Розмірність вектора коефіцієнтів \bar{B} комбінаторно зростає зі збільшенням степеня полінома d . Наприклад, при $d=2$ розмірність \bar{B} дорівнює 15, а при $d=4$ зростає до 70.

Результати ідентифікації при $d=2$ показані на рис.4.28.

Регуляризацію тут ще не застосовували. Максимальна відносна похибка відтворення сигналу детектора для методу найменших квадратів становить 10%, а для методу мінімаксу – 42%.

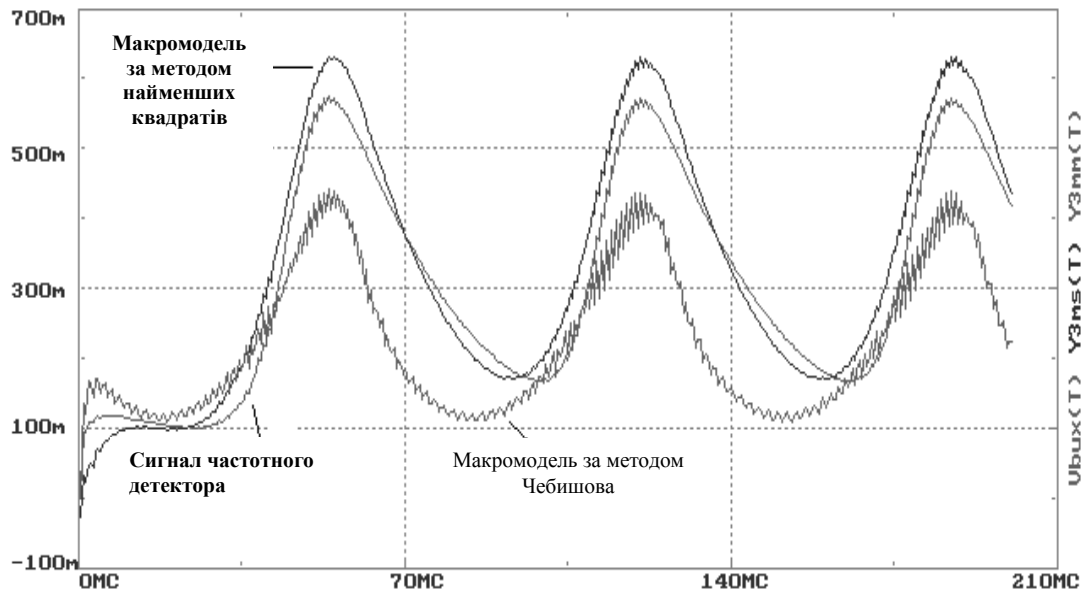


Рис.4.28. Вихідні сигнали частотного детектора та макромоделей у разі апроксимації нелінійної функції поліномом другого степеня

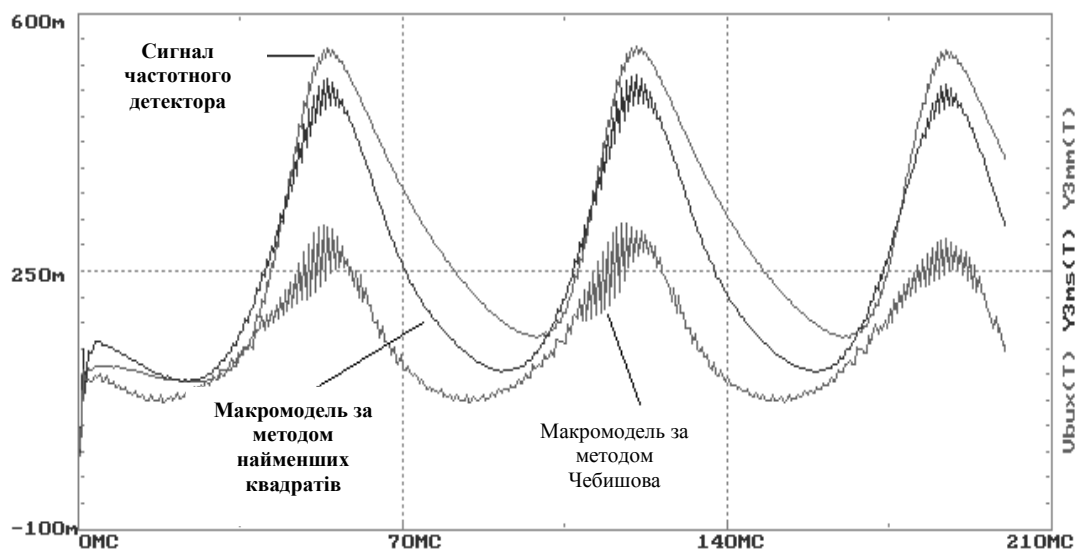


Рис.4.29. Вихідні сигнали частотного детектора та макромоделей у разі апроксимації нелінійної функції поліномом третього степеня

При $d=3$ ще вдалось отримати результати без регуляризації (рис.4.29). Максимальна відносна похибка відтворення вихідного сигналу для методу найменших квадратів дорівнює 17%, а для методу мінімаксу – 68%. Отже, збільшення степеня полінома призвело не до поліпшення якості макромоделі, а навіть до деякого погіршення. Це типовий прояв некоректності із збільшенням степеня полінома.

Зауважимо, що процесорний час одного розв'язування задачі (4.51) за методом мінімаксу виявився приблизно втричі більшим, ніж час розв'язування задачі (4.50) за методом найменших квадратів. Разом з тим якість макромоделі за методом мінімаксу помітно гірша.

Треба окремо означити проблему якості макромоделей, отриманих за різними методами апроксимації. Тут йдеться про *якість макромоделі*, а не про *якість апроксимації нелінійної функції*. Відомо, що чебишовське (мінімаксне) наближення дає в загальному випадку ліпшу якість апроксимації функцій, ніж метод найменших квадратів. Однак макромодель – це складна система із зворотними зв'язками (рекурсіями), до якої апроксимовані функції входять як складові частини. Можливо, згладжувальні властивості методу найменших квадратів є причиною ліпшої якості макромоделей у задачі, яку розглядаємо. В усякому разі для узагальнених висновків поки що замало даних і можна лише говорити про результати макромодельовання частотного детектора.

Подальше збільшення степеня апроксимуючого полінома без застосування регуляризації призводило до нестійких результатів. Для $d=4$ вдалось отримати працездатну макромодель лише із застосуванням редукції апроксимуючого полінома за методом, викладеним у цьому параграфі. На рис.4.30 показано фрагмент процесу зміни максимальної відносної похибки відтворення вихідного сигналу у міру редукції полінома. По осі абсцис відкладено номер k -ї ітерації за описаним вище методом редукції апроксимуючого полінома (тобто кількість коефіцієнтів полінома дорівнює $70-k$), а по осі ординат – максимальну відносну похибку відтворення вихідного сигналу у відсотках.

Найліпший отриманий результат у разі редукції кількості членів апроксимуючого полінома від 70 до 25 показано на рис.4.31 ($k=35$ на рис.4.30). Найбільша відносна похибка відтворення вихідного сигналу у встановленому режимі становить лише 4% (рис.4.31).

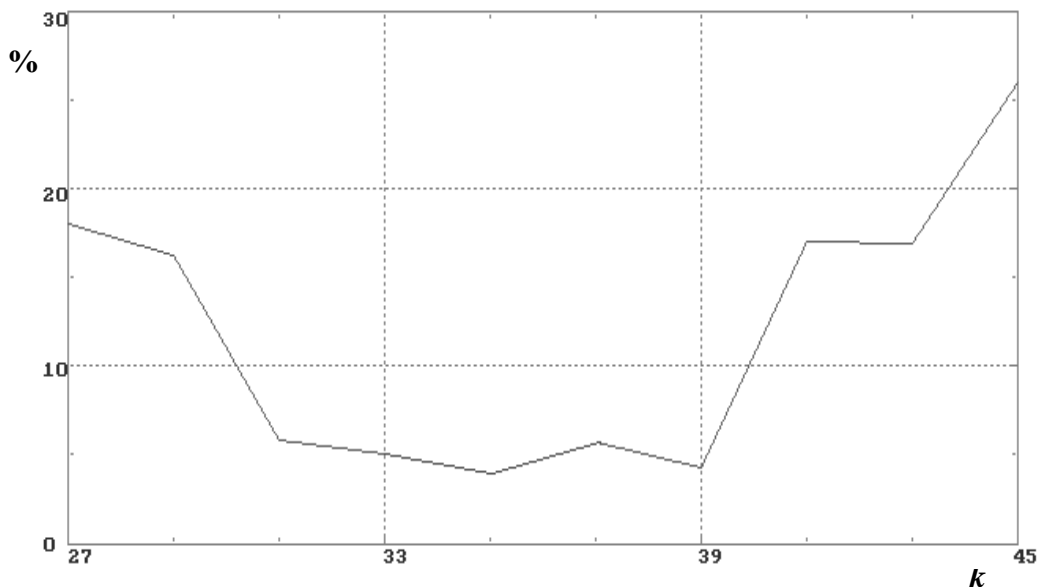


Рис.4.30. Залежність максимальної відносної похибки макромоделі (%) від кроку редукції апроксимуючого полінома (k)

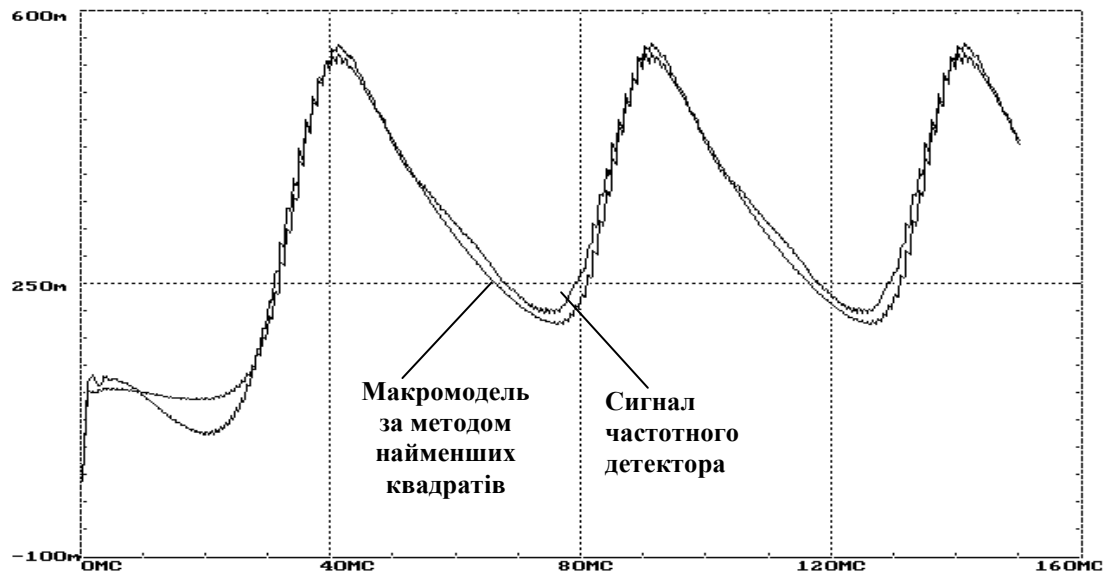


Рис.4.31. Вихідні сигнали частотного детектора та макромоделі після редукції апроксимуючого полінома від 70 до 35 членів

Відповідне завдання мовою системи САНОС-ПК наведене в табл.4.5. З тексту завдання видно, які саме коефіцієнти виявились суттєвими за методом редукції.

Таблиця 4.5. Завдання для порівняльного аналізу частотного детектора та макромоделі, отриманої методом редукції апроксимуючого полінома BEGI

Порівняння частотного детектора та макромоделі
DESC

Схема частотного детектора ;
 MODEL: DE(1,2) I0,U0; * Модель діода ;
 J1(1,2); V1(J1); J1=I0*(EXP(V1/U0)-1);;
 TYPE: 'D20'(DE) 2E-5,0.116;;
 E1(0,1) E0*SIN(W0*(T-km/WM*COS(WM*T))); * Вхідний сигнал ;
 f1=f0*(1+km*SIN(WM*T)); * Функція частотної модуляції ;
 E0=2; f0=1.2MG; fm=20K; km=0.2; PI=4*arctan(1); * Параметри ;
 W0=2*PI*f0; WM=2*PI*fm; WN=2*PI*1MG; * Константи ;
 L1(1,2)1ML; C1(2,30)40P; R1(30,0)1.5K; * Схема ЧД ;
 VR1(R1); E2(0,3) VR1;
 DE1(3,4)'D20'; R2(4,0)1K; C2(4,0)30N; Vвих(4,0);
 * Макромодель ;
 wc`=5e9*(E1-wc); * Диференціювання ;
 pE1=5e9*(E1-wc)/WN; * вхідного сигналу ;
 y1`=y2; * Лінійна підсистема ;
 y2`=(A1*pE1+A2*y1+A3*y2n)*WN**2; y2n=y2/WN;
 A1=1.591545E-04; A2=-6.332558E-01; A3=-2.361329E-01;
 y3`=f2*WN; * Нелінійна підсистема ;
 pE1c2=pE1*pE1; pE1c3=pE1*pE1*pE1; pE1c4=pE1*pE1*pE1*pE1;

```

y1c2=y1*y1; y1c3=y1*y1*y1; y1c4=y1*y1*y1*y1;
y2nc2=y2n*y2n; y2nc3=y2n*y2n*y2n; y2nc4=y2n*y2n*y2n*y2n;
y3c2=y3*y3; y3c3=y3*y3*y3; y3c4=y3*y3*y3*y3;
f2=B0000+B0100*y1+B0001*y3+B2000*pE1c2+B1100*pE1*y1+B1010*pE1*y2n+
B1001*pE1*y3+B0110*y1*y2n+B0101*y1*y3+B0020*y2nc2+B0011*y2n*y3+
B3000*pE1c3+B2100*pE1c2*y1+B2010*pE1c2*y2n+B2001*pE1c2*y3+B1200*pE1*y
1c2
+B1110*pE1*y1*y2n+B1002*pE1*y3c2+B0300*y1c3+B0210*y1c2*y2n+
B0201*y1c2*y3+B0120*y1*y2nc2+B0102*y1*y3c2+B0030*y2nc3+B0021*y2nc2*y3+
B0012*y2n*y3c2+B2200*pE1c2*y1c2+B2002*pE1c2*y3c2+B1210*pE1*y1c2*y2n+
B1120*pE1*y1*y2nc2+B0400*y1c4+B0301*y1c3*y3+B0211*y1c2*y2n*y3+
B0130*y1*y2nc3+B0121*y1*y2nc2*y3;
B0000=-1.83627E-03; B0100=-68.0413; B0100=2.3578E-03; B2000=3.15826E-04;
B1100=30.5748; B1010=-1.55837; B1001=-0.0404764; B0110=-44934.9; B0101=-
327.935;
B0020=5154.99; B0011=59.3577; B3000=1.86768E-03; B2100=39.505; B2010=-
8.75685;
B2001=1.13971E-03; B1200=221239.6; B1110=-125787; B1002=9.36891E-02;
B0300=9.62965E+08; B0210=-3.78511E+08; B0201=-219868; B0120=3.95986E+08;
B0102=1123.46; B0030=1.59719E+07; B0021=-12031.6; B0012=-143.449;
B2200=33813.5;
B2002=1.30725E-03; B1210=-9.99167E+07; B1120=-1.47303E+08;
B0400=1.77466E+11;
B0301=-1.39703E+09; B0211=6.94395E+07; B0130=2.27937E+11; B0121=-
8.51381E+08;;;
TRAN
ERMAX=0.001; TMAX=2E-4; HMAX=1E-7; HMIN=1E-33; HS=2E-7;
OUTPUT: Vвих, y3, XMIN=0,XMAX=0.2ML,YMIN=0,YMAX=1;;
END

```

Виконано експерименти з оцінки області адекватної поведінки макромоделі у разі зміни параметрів вхідного сигналу – амплітуди і частоти функції модуляції. Максимальна відносна похибка не перевищує 10%, якщо вказані параметри не виходять за межі 30% відхилення від значень, наведених у табл.4.5.

Отже, отримана макромодель не лише добре відтворює реакцію частотного детектора на заданому сигналі, але й є помітно стійкою щодо зміни параметрів вхідного сигналу.

4.7. ВИЗНАЧЕННЯ РОЗМІРНОСТІ ДИНАМІЧНИХ СИСТЕМ ЗА ВИХІДНИМ СИГНАЛОМ

Під час побудови математичних моделей нелінійних динамічних систем дуже корисною є попередня оцінка розмірності системи. Вона дає змогу уникнути надмірної складності моделі та пов'язаної з нею некоректності. Далі описано метод, що дає змогу отримати надійну оцінку

розмірності системи лише за одним скалярним вихідним сигналом, причому оцінка практично не залежить від того, який саме вихідний сигнал системи використано.

Нехай вихідний сигнал системи залежить від вектора змінних стану $\bar{x} = (x_1, x_2, \dots, x_N)$. Оскільки фазові змінні системи x_1, x_2, \dots, x_N зазвичай недоступні для прямого спостереження, то розглянемо поведінку системи у псевдофазовому просторі з координатами

$$y(t), dy(t)/dt, d^2y(t)/dt^2, \dots, d^{M-1}y(t)/dt^{M-1}.$$

Відомо, що фазові траєкторії у псевдофазовому просторі зберігають свої топологічні властивості у просторі змінних $y(t), y(t+\tau), y(t+2\tau), \dots, y(t+(M-1)\tau)$, де крок τ можна оцінити за мінімумом автокореляційної функції [60]:

$$\min_{\xi=\tau} \int_{-\infty}^{+\infty} y(t)y(t-\xi)dt \quad (4.53)$$

Нехай вихідний сигнал $y(t)$ задається послідовністю дискретних значень $\{y(k\tau)\}$ зі сталим у часі кроком τ , де k – ціле невід'ємне число. Тоді для сигналу $\{y(k\tau)\}$ псевдофазовою траєкторією буде послідовність точок y_k з координатами

$$y_k = (y(k\tau), y(k\tau+\tau), y(k\tau+2\tau), y(k\tau+3\tau), \dots, y(k\tau+(M-1)\tau)),$$

де M – розмірність псевдофазового простору.

Зв'язок між розмірністю фазового простору N і розмірністю збудованого псевдофазового простору M дають теореми, сформульовані і доведені Ф.Такенсом [7]. Основний для нас результат цих теорем виглядає так: псевдофазовий простір системи топологічно ідентичний фазовому, якщо $M = 2N+1$.

Оцінити розмірність системи можна за допомогою поняття фрактальної розмірності для заданої фазової траєкторії [60]. Є різні підходи до визначення фрактальної розмірності, що дають приблизно однакові результати. Скористаємось поняттям кореляційної розмірності d_m для заданої розмірності m псевдофазового простору. Позначимо

$$s_{ij} = \|y_i - y_j\|$$

і визначимо кореляційну функцію так:

$$C(r) = \lim_{K \rightarrow \infty} 1/K^2 \cdot (\text{кількість пар точок, для яких } s_{ij} < r),$$

де K – загальна кількість точок на фазовій траєкторії.

На практиці для визначення $C(r)$ використовують формулу

$$C(r) = \lim_{K \rightarrow \infty} \frac{1}{K^2} \sum_{i=1}^K \sum_{\substack{j=1 \\ i \neq j}}^K H(r - s_{ij}),$$

де $H(s) = \begin{cases} 1, & s > 0 \\ 0, & s < 0 \end{cases}$, а K обмежене зверху K_{\max} .

Переважає $\lim_{r \rightarrow 0} C(r) \approx r^{d_m}$, тому кореляційну розмірність d_m можна визначити за нахилом прямої у системі координат $(\ln C(r), \ln r)$:

$$d_m = \lim_{r \rightarrow 0} \frac{\ln C(r)}{\ln r}. \quad (4.54)$$

Такенс довів, що розмірність d_m зростає зі збільшенням m , доки не досягнуто верхньої межі при $m=M$.

Отже, для оцінки розмірності системи за вихідним сигналом треба:

- визначити необхідний крок дискретизації за (4.53);
- збудувати на дискретних значеннях сигналу псевдофазові простори різної розмірності m ;
- обчислити значення кореляційної розмірності d_m за (4.54) для низки значень розмірності m псевдофазового простору;
- визначити $m=M$, при якому d_m стає максимальним;
- оцінити розмірність системи N за формулою $N=(M-1)/2$.

Розглянемо застосування такого методу оцінки розмірності.

Дослідимо модель генератора, описану у параграфі 4.2. Модель містить п'ять диференціальних рівнянь. Залежність кореляційної розмірності d_m від розмірності псевдофазового простору m для вихідного сигналу синхронного генератора при $K_{\max} = 2000$, $\tau = 0.03$ с така:

m	2	3	4	5	6	7	8	9	10	11
d _m	0.960	0.963	1.036	1.062	1.096	1.145	1.145	1.143	1.140	1.164

Як бачимо, кореляційна розмірність досягає локального максимуму, якщо розмірність псевдофазового простору $m=7$. Це відповідає системі третього порядку. Отже, модель генератора, описана системою диференціальних рівнянь п'ятого порядку (4.11), поводить ся головню як система третього порядку. Результати макромодельовання, наведені в параграфі 4.2, підтверджують цей висновок. Однак для $m=11$ значення d_m знову помітно зростає. Це означає, що передати всі тонкощі реакції системи можна лише моделлю п'ятого порядку, що збігається з розмірністю диференціальних рівнянь повної моделі генератора.

Значення кореляційної розмірності обчислювали програмою, текст якої міститься у Додатку 7. На рис.4.32 показані залежності $\ln(C(r))$ від $\ln(r)$ для різних значень розмірності псевдофазового простору m . За цими залежностями обчислені значення кореляційної розмірності, наведені вище.

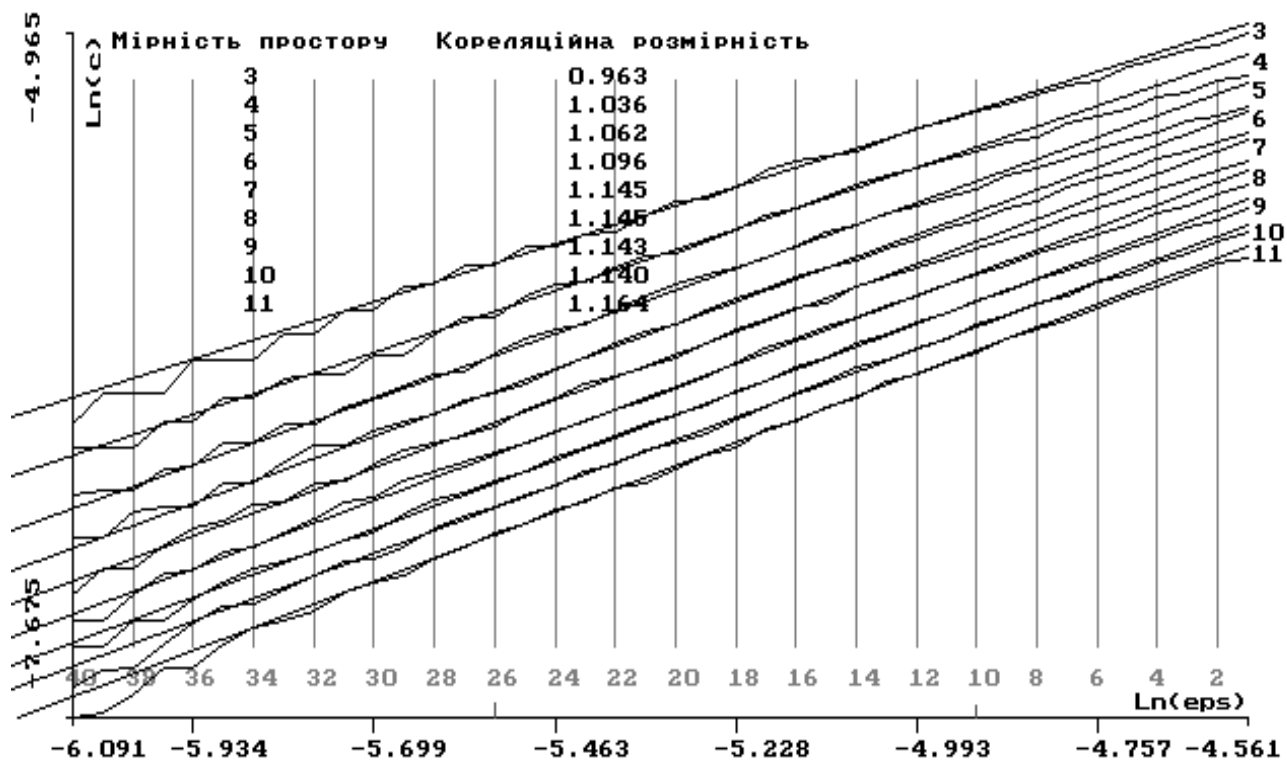


Рис.4.32. Залежності $\ln(C(r))$ від $\ln(r)$ для синхронного генератора

На близькій до лінійної ділянці, виділеній позначками на осі абсцис, обчислені значення кореляційних розмірностей як коефіцієнтів нахилу відповідних прямих апроксимації, визначених за методом найменших квадратів.

Програма у Додатку 7 використана також для обчислення кореляційних розмірностей дивних атракторів у параграфі 4.1 та схеми Чуа у розділі 5.

РОЗДІЛ 5

ВИКОРИСТАННЯ У МАКРОМОДЕЛЮВАННІ СИСТЕМИ САНОС-ПК

5.1. ЗАГАЛЬНИЙ ОПИС СИСТЕМИ

САНОС-ПК – це складна система програм схемотехнічного проектування радіоелектронних схем, об'єднаних спільною мовою. Вона призначена для аналізу та оптимізації радіоелектронних схем із зосередженими параметрами, а також нелінійних динамічних систем, описаних звичайними диференціальними рівняннями в формі Коші. Функціональні можливості системи САНОС-ПК дають змогу виконувати аналіз схем та систем у трьох режимах (по постійному струму, в часовій та частотній областях), розрахунок чутливостей вихідних сигналів до зміни параметрів складових частин схеми або системи, параметричну оптимізацію схем та систем в усіх трьох режимах аналізу.

Перша версія системи САНОС створена протягом 1977–1983 рр. для ЕС-ЕВМ [70]. В основу розробки була покладена французька система IMAG3. У 1990 р. запрацювала перша версія системи для IBM-подібних персональних ЕОМ САНОС-ПК.

Основна мова програмування системи – FORTRAN-5. Окремі модулі створені мовою ASSEMBLER. Графічний препроцесор, діалогова оболонка та постпроцесор системи виконані мовами Turbo-PASCAL та C++. Загальна кількість підпрограм системи – понад 100.

Система САНОС-ПК має просту та гнучку мову опису схеми і завдання аналізу, що дає змогу описувати широкий клас схем, а також обчислювати довільні функції від змінних схеми. Бібліотеку моделей, типів і таблиць створено та підтримувано на рівні вхідної мови.

В основі системи – модульний принцип організації, тобто система складається з модулів, що виконують різні функції: синтаксичний аналіз та редагування завдання; складання математичної моделі схеми; розв'язок математичної моделі згідно завдання. Основні ієрархічні рівні та відповідне їм програмне забезпечення показані на рис.5.1, а функціональні зв'язки між блоками системи САНОС-ПК – на рис.5.2.

Загальне керування роботою системи виконує *супервізор*, який визначає порядок опрацювання даних та результатів блоками системи відповідно до заданих керівних команд.

Початкова функція системи полягає в опрацюванні опису схеми. Її виконує *компілятор*, що складається з *синтаксичного аналізатора*, *редактора*, набору *семантичних функцій* та *інтерпретатора*. Діяльність компілятора повністю відповідає класичним принципам синтаксичного аналізу та редагування. В результаті опрацювання речень опису схеми мовою САНОС компілятор формує масив, що відображає топологію схеми та рівняння для компонент, описаних виразами.

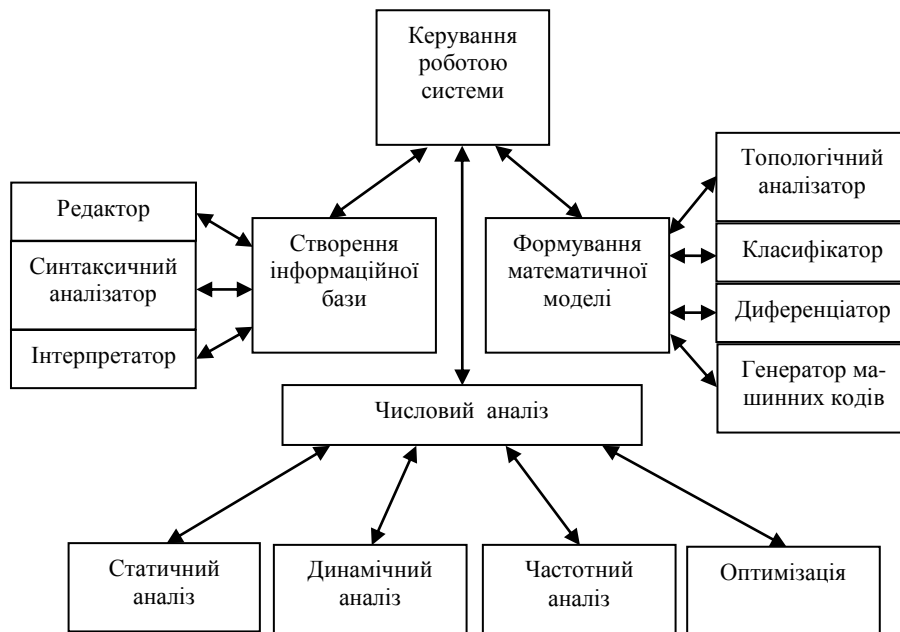


Рис.5.1. Основні рівні ієрархії програмного забезпечення САНОС-ПК.

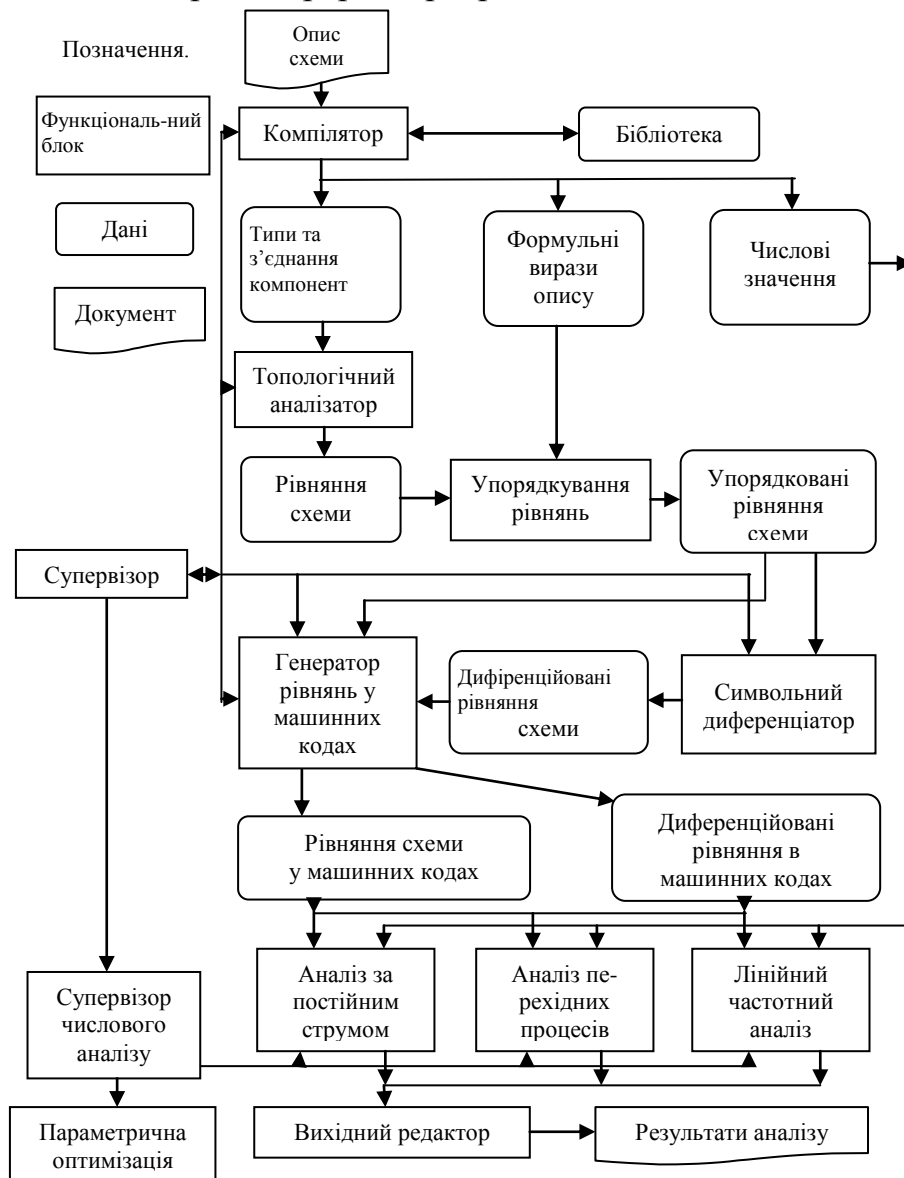


Рис.5.2. Функціональна схема зв'язків між блоками САНОС-ПК.

Редактор читає речення за реченням опис схеми вхідною мовою, виділяє синтаксичні одиниці та групує їх для синтаксичного аналізатора, а також упорядковує в окремих масивах ідентифікатори змінних та числові значення. *Синтаксичний аналізатор* перевіряє відповідність синтаксичних одиниць граматиці мови і викликає *семантичні функції*, які власне перекладають речення вхідної мови внутрішньою мовою системи САНОС. Нарешті, *інтерпретатор* формує масив з'єднань компонент і масив відповідних їм виразів у інверсній польській формі запису.

Результати роботи компілятора використовує *топологічний аналізатор*, який згідно з вмістом масиву з'єднань компонент схеми визначає вектор незалежних змінних – змінних стану, тобто напруг на конденсаторах та струмів у індуктивностях з урахуванням можливих вироджень, та складає відповідну систему алгебро-диференціальних рівнянь. На першому етапі вибирається модифіковане дерево схеми в сенсі Брента. Далі за ним формується система алгебро-диференціальних рівнянь вигляду

$$B \frac{dx}{dt} = F(\bar{x}, \bar{y}, t);$$

$$A\bar{y} = G(\bar{y}, t),$$

де: A, B – матриці; F, G – вектор-функції; \bar{x} – вектор змінних стану; \bar{y} – вектор допоміжних змінних.

Складені рівняння сумісно з рівняннями компонент упорядковуються. Оскільки в процесі розв'язування математичної моделі необхідно багаторазово обчислювати складені вирази, то *генератор* перекладає їх в еквівалентну послідовність команд машинною мовою. Цей прийом дає змогу помітно зменшити машинний час, потрібний для розв'язування рівнянь математичної моделі.

Диференціатор виконує символічне диференціювання рівнянь щодо будь-якого параметра. Диференційовані рівняння *генератор* також перекладає у послідовність машинних команд, що дає змогу ефективно обчислювати якобіан та коефіцієнти чутливості.

Блоки *числового аналізу* містять набір числових методів розв'язування завдань аналізу.

Інтегрування системи диференціальних рівнянь для обчислення перехідних процесів виконується неявними методами Гіра другого порядку та Гіра-Нордсіка змінного порядку, а також явним методом Рунге-Кутта-Фельдберга четвертого порядку. Систему нелінійних алгебричних рівнянь для знаходження режиму за постійним струмом розв'язують методом Ньютона-Рафсона або модифікованим методом Ньютона. Розв'язування системи лінійних алгебричних рівнянь у разі частотного аналізу лінеаризованої моделі, під час розрахунку коефіцієнтів чутливості, на кожній ітерації чисельного інтегрування тощо виконують за допомогою LU-розкладу з урахуванням розрідженості.

Чутливість у статичному та лінеаризованому режимах визначають символічним диференціюванням математичної моделі схеми за вектором параметрів варіацій.

Блок *оптимізації* містить набір програм, що реалізують методи параметричної оптимізації схеми для всіх трьох видів аналізу: статичного, динамічного та частотного. Вхідна мова дає змогу задавати функцію мети у вигляді алгебричних та логічних виразів, що залежать від довільних електричних змінних та параметрів. Градієнт функції мети визначають символічним диференціюванням з високою точністю та швидкодією. Це дало змогу застосувати ефективні методи спряжених градієнтів, Девідона-Флетчера-Пауела, Гольдфарба.

Метод аналізу та метод оптимізації задає користувач вхідною мовою або система вибирає автоматично за замовчуванням.

Редактор вихідної інформації організує обробку результатів розрахунку у вигляді графіків та таблиць.

Усі модулі системи виконані у вигляді оверлейних структур з перекриттям і надходять в оперативну пам'ять за потребою. Зв'язок між модулями відбувається через області спільної оперативної пам'яті та файли на магнітному диску.

5.2. КОРОТКИЙ ВИКЛАД ВХІДНОЇ МОВИ

Алфавіт мови складається з великих та малих кирилических і латинських літер, десяткових цифр та спеціальних знаків: () . ; : ` + - / * **.

Числа мають такі формати:

цілий: послідовність цифр із знаком (знака + може не бути).

Наприклад: -289;

десятковий: два цілі, розділені точкою. Наприклад: -2.89;

експоненційний: ціле або десяткове з цілим степенем числа 10, що позначений літерою E або e. Наприклад: 3E-3 або -2.8e-12;

десятковий з розмірністю, що позначена великими або малими літерами P (10^{-12}), N (10^{-9}), MC (10^{-6}), ML (10^{-3}), K (10^3), MG (10^6). Наприклад: 2P, -3.14mg.

Усі розмірні величини задані в системі СІ (опір – в омах, ємність – у фарадах, струм – в амперах тощо). Значення розмірностей впливає з контексту і окремо не вказується.

Ідентифікатор – це послідовність не більше восьми буквенно-цифрових символів. Ними позначають компоненти схеми, вузли, змінні, параметри. Певні ідентифікатори зарезервовані для спеціальних позначень (елементарні функції sin, cos, tan, exp, log, sqrt, arcsin, arctg, abs, max, min, mod; логічні функції "більше".gt., "більше-рівно".ge., "менше".lt., "менше-рівно".le.; модуль M і фаза P при частотному аналізі; біжучий час T та біжуча частота F) та для службових слів і команд. Ідентифікатори простих компонент схеми починаються з певних літер: для опорів – R; для ємностей – C; для індуктивностей – L; для джерел напруги – E; для джерел струму – J;

для взаємних індуктивностей – М. Крім того, ідентифікатори струмів схеми розпочинаються з літери І, а напруг – з V.

Опис схеми складається з речень. Кожне речення завершує розділювач “;”. Розглянемо основні речення опису.

Речення розміщення

$R_{vx}(1, BX);$

означає, що опір R_{vx} є між вузлами 1 та BX.

Речення розміщення

$S_{vx}(R_{vx});$

означає, що ємність S_{vx} розміщена між тими ж вузлами, що й опір R_{vx} .

Речення присвоєння

$R_{vx}=V_{vx}+AMP*\sin(\omega*T+fi);$

означає, що опір R_{vx} в омах дорівнює сумі напруги V_{vx} та гармонічної функції часу з амплітудою AMP, частотою ω та початковою фазою fi .

Речення розміщення повинно передувати реченню присвоєння, інакше ідентифікатор лівої частини речення присвоєння трактується як безрозмірний параметр.

Речення розміщення та присвоєння можна об’єднувати в одному реченні. Зокрема, речення

$R_{vx}(1, BX) = V_{vx}+AMP*\sin(\omega*T+fi);$

еквівалентне двом попереднім реченням.

Струми та напруги описують реченнями розміщення, наприклад

$V_{vx}(0,1);$ або $I_{vx}(R_{vx});$

Параметр позначають довільним ідентифікатором і, на відміну від електричних змінних та простих компонент схеми, від безпосередньо не пов’язаний з вузлами схеми. Речення присвоєння

$power=I_{out}*V_{out};$

описує параметр $power$, який дорівнює добутку миттєвих значень струму I_{out} та напруги V_{out} .

Умовне речення має вигляд:

$A=IF(<вираз><логічна\ функція><вираз>)(<вираз1>)(<вираз2>);$

де А набуває значення $<вираз1>$, якщо результат логічної функції – "істина", і значення $<вираз2>$ – у протилежному випадку. Наприклад: $Імпульс=IF(T.ge.10ml)(10*T)(0);$ де параметр Імпульс дорівнює $10*T$, якщо значення поточного часу $T \geq 1e-2$ с, і 0, якщо $T < 1e-2$ с.

Речення

$x'=a*x+b;$

відповідає диференціальному рівнянню

$dx/dt=ax+b,$

початкові умови для якого задає речення

INIT: $x(5.2e-3);$

Початкові умови схеми можна задати також як напруги на конденсаторах та струми в індуктивностях за допомогою речення

INIT: $C4(-1.5), L8(1ml);$

що задає початкову напругу $-1.5V$ на ємності $C4$ та початковий струм $1mA$ в індуктивності $L8$. Початкові умови, не задані явно, вважають нульовими.

Речення-коментар починається символом `*` і містить довільну символічну послідовність до розділювача.

Символ пропуску ігнорується.

Складні компоненти схеми описують моделями, які є описами заступних схем цих компонент.

Опис моделі починається реченням

`;MODEL:<ім'я моделі>(<список вузлів>)<список параметрів>;`

за яким ідуть речення, що описують внутрішню структуру моделі, пов'язану з ідентифікаторами зовнішніх вузлів та ідентифікаторами параметрів моделі. Опис моделі завершують два розділювача. У вигляді моделей описують заступні схеми діодів, транзисторів, мікросхем тощо.

Числові значення параметрам моделі присвоюють за допомогою звертань до типів. Тип описують реченням

`;TYPE:<ім'я типу>(<ім'я моделі>)<список чисел>;`

Якщо модель та відповідний тип були попередньо записані у *Бібліотеку* системи, то в описі схеми достатньо лише до них звернутись, наприклад, реченням

`NPN15(0,4,7)'KT312A';`

Це речення означає, що компонента схеми з моделлю `NPN` та відповідним типом `'KT312A'` розміщена між вузлами схеми 0, 4 та 7.

Вхідною мовою зручно задавати кусково-лінійні функції за допомогою речення

`;TABLE:'<ім'я функції>' A1/F1,A2/F2,...,An/Fn;;`

де $A1, \dots, A_n$ – числа, що відповідають значенням аргументу, а $F1, \dots, F_n$ – числа, що відповідають значенням функції. Звертання до таблично заданої функції в описі схеми може мати вигляд

`E1(k,15)'impulse'(T);`

де джерело напруги $E1$, що розміщене між вузлами k та 15 , набуває значення відповідно до опису таблиці з іменем `impulse`, аргументом якої є поточний час. Описи таблиць можуть бути у *Бібліотеці* системи.

Будь-яке завдання починається командою `BEGI` в окремій стрічці, за якою обов'язковою є стрічка з довільним текстом – заголовком. Опис схеми починають командою `DESC` в окремій стрічці, а завершують потрібним розділювачем.

Опис даних для аналізу починають в окремій стрічці однією з команд:

`TRAN` – аналіз перехідних процесів;

`DC` – аналіз за постійним струмом;

`AC` – частотний аналіз лінеаризованої схеми;

`OPTI` – параметрична оптимізація;

і завершується подвійним розділювачем.

Опис даних для аналізу містить алфавітно-числові параметри, що можуть задавати метод аналізу, особливості його виконання та деякі числові характеристики. Наприклад,

TRAN

TMAX=10ml; HS=5e-5; GEAR; ERMAX=0.005;

задає аналіз перехідного процесу протягом 10 мс (TMAX) неявним методом змінного порядку (GEAR) з оцінкою максимальної похибки 0.5% (ERMAX) та кроком виведення результатів 50мкс (HS).

Список змінних та параметрів, що виводяться на екран у вигляді графіків у міру просування розрахунку схеми, задають реченням

OUTPUT:<список>,<масштабні константи графіка>;

Система САНОС має розвинуту наскрізну підсистему діагностування помилок із рекомендаціями щодо їх виправлення.

5.3. МОДЕЛЮВАННЯ ДИВНИХ АТРАКТОРІВ СХЕМИ ЧУА

Опишемо деякі можливості системи САНОС-ПК на прикладі дослідження схеми Чуа. Розглянемо відому схему, запропоновану Чуа (L.Chua) як проста детермінована система, що здатна реалізувати хаотичні рухи. Схема зображена на рис.5.3. Вона складається з двох ємностей $C1$ і $C2$, однієї індуктивності L , опору R і нелінійного від'ємного опору $Rn=fn(Uc1)$, що має вольт-амперну характеристику (ВАХ), зображену на рис.5.4.

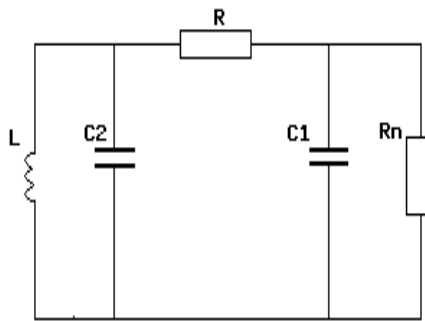


Рис.5.3. Схема Чуа.

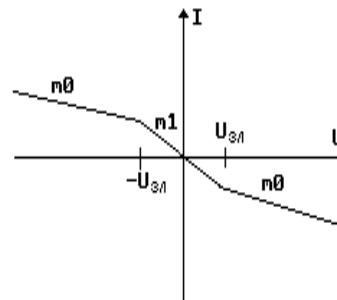


Рис.5.4. ВАХ опору Rn .

Рівняння стану цієї схеми мають вигляд

$$C1 \frac{dUc1}{dt} = (Uc2 - Uc1) / R - fn(Uc1);$$

$$C2 \frac{dUc2}{dt} = (Uc1 - Uc2) / R + IL;$$

$$L \frac{dIL}{dt} = -Uc2;$$

де функція $fn(Uc1) = m0 \cdot Uc1 + 1/2 \cdot (m1 - m0)(|Uc1 + U_{3l}| - |Uc1 - U_{3l}|)$ має такі параметри: $m0$ та $m1$ – коефіцієнти нахилу лінійних ділянок ВАХ; U_{3l} – напруга зламу ВАХ.

Фізична реалізація схеми Чуа полягає у створенні нелінійного елемента, що мав би потрібну кусково-лінійну характеристику. М.Кенеді [71] запропонував реалізацію нелінійності на двох операційних підсилювачах (ОП). В [72] описана реалізація обмеженого від'ємного опору схеми Чуа на одному ОП і двох кремнієвих діодах малої потужності. На рис.5.5 зображена схема Чуа з такою реалізацією від'ємного опору.

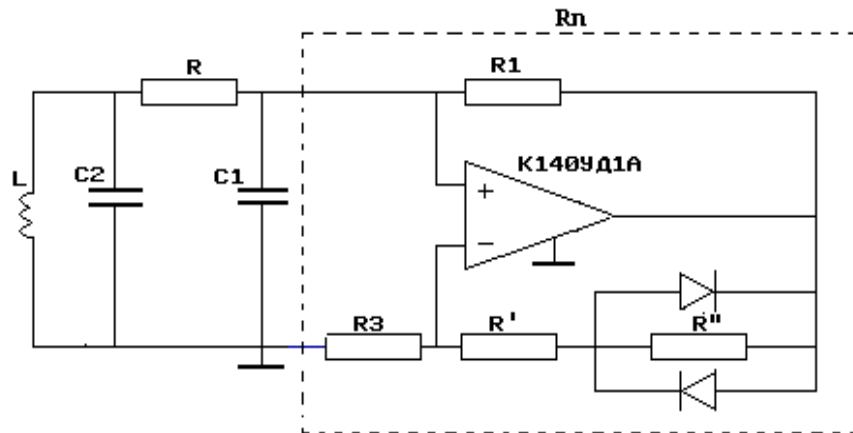


Рис.5.5. Електрична модель схеми Чуа

У [72] доведено, що за умови ідеальності ОП та діодів

$$R1 = \frac{-U_{\text{в}}}{U_{\text{зл}}(m1 - m0)}; \quad R' = R3 \frac{-m0 \cdot U_{\text{в}}}{(m1 - m0) \cdot U_{\text{зл}}}; \quad R'' = R3 \frac{U_{\text{в}}}{U_{\text{зл}}}.$$

Прийmemo $R3 = 2 \text{ кОм}$. Згідно з [71] $m1 = -0.8 \text{ мСім}$, $m0 = -0.5 \text{ мСім}$, $U_{\text{зл}} = 1 \text{ В}$. Для малопотужного кремнієвого діода напруга відкривання $U_{\text{в}} = 0.6 \text{ В}$. Отже $R1 = 2 \text{ кОм}$, $R' = 2 \text{ кОм}$, $R'' = 1.2 \text{ кОм}$.

Значення інших елементів схеми Чуа візьmemo з [71]: $C1 = 10 \text{ нФ}$; $C2 = 100 \text{ нФ}$; $L = 18 \text{ мГн}$; $R = 0 \div 2 \text{ кОм}$.

Схема визначення ВАХ нелінійного елемента показана на рис.5.6, де $R_{\text{р}} = 10 \text{ Ом}$ – малий пробний опір, X та Y – входи осцилографа.

ВАХ нелінійного елемента з від'ємним опором зображена на рис.5.7, що є комп'ютерною обробкою фотографії, зробленої з екрану осцилографа. Тут X приблизно рівне напрузі на опорі $R_{\text{п}}$, а Y пропорційне струмові через той же опір.

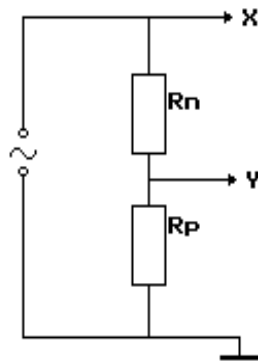


Рис.5.6. Схема визначення ВАХ нелінійного елемента

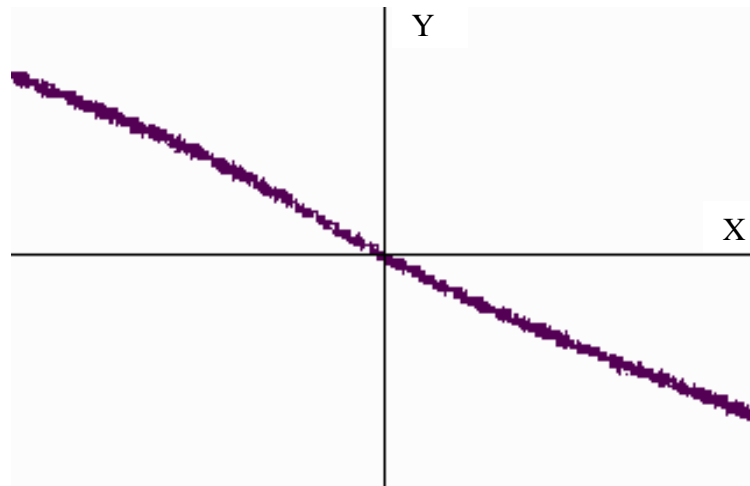


Рис.5.7. ВАХ нелінійного елемента
(по горизонталі 4 В, по вертикалі 3 мА)

Фазовий портрет схеми Чуа досліджували в проекції на площину (U_{c1} , U_{c2}). Для цього напруги на ємностях $C1$ та $C2$ подані відповідно на входи осцилографа X та Y . Зміна опору R зумовлює проходження через точки біфуркацій.

На рис.5.8-5.11 показані комп'ютерно опрацьовані осцилограми характерних режимів.

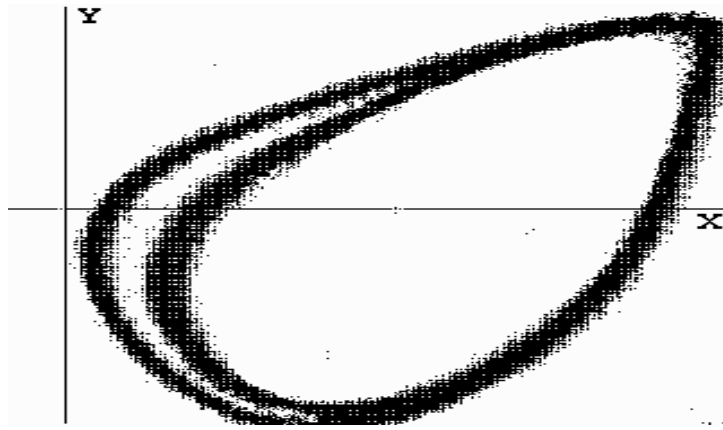


Рис.5.8. Роздвоєння періоду при $R=1580 \text{ Ом}$
(по горизонталі 3 В, по вертикалі 1 В)

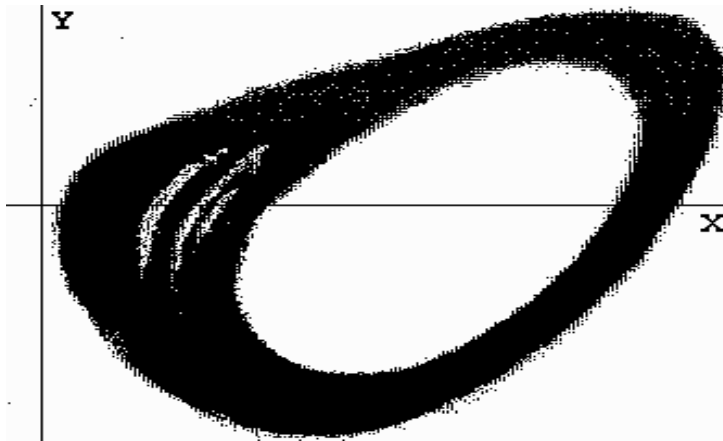


Рис.5.9. Атрактор Реслера [71] при $R=1540 \text{ Ом}$
(по горизонталі 3 В, по вертикалі 1 В)

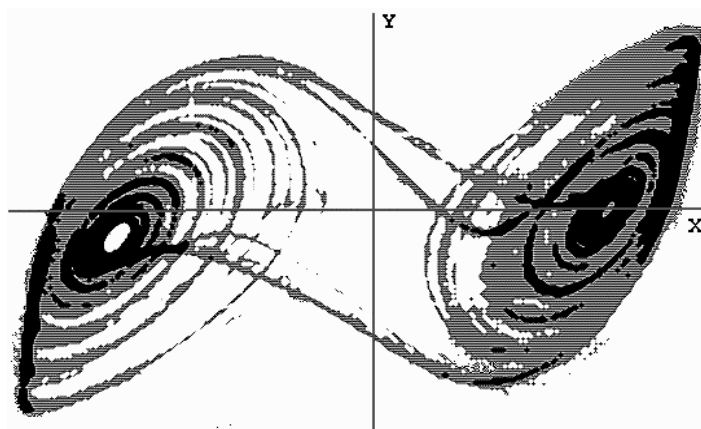


Рис.5.10. “Подвійна прокрутка”(“double scroll”) при $R=1500 \text{ Ом}$
(по горизонталі 6 В, по вертикалі 1 В)

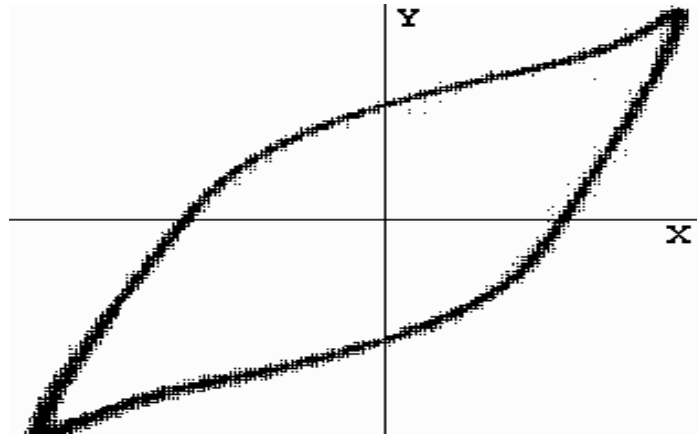


Рис.5.11. Граничний цикл при $R < 1300 \text{ Ом}$
(по горизонталі 9 В, по вертикалі 6 В)

Опис електричної схеми з рис.5.5 мовою системи САНОС-ПК разом із завданням для аналізу перехідних процесів наведено у табл.5.1. Ідеальний операційний підсилювач моделює вхідна ланка CIN-RIN та ідеальне джерело напруги з нелінійним коефіцієнтом керування $K \cdot VINSAT$. Схема на рис.5.5 доповнена омичним опором RL індуктивності $L1$.

Таблиця 5.1. Завдання для аналізу схеми Чуа у системі САНОС-ПК
BEGI

Електрична модель схеми Чуа мовою системи САНОС-ПК (рис.5.5)

DESC

```
MODEL:D(1,2)C1,KI,KV;                                     * Модель діода ;
J1(1,2); VJ(J1); C1(J1); J1=KI*(EXP(KV*VJ)-1);;
TYPE:'KD'(D)1P,1E-10,30.0;;
L1(10,1) 18ML; RL(0,10) 32.0; C2(0,1) 100N; C1(0,2) 10N;* Опис схеми ;
R_(1,2); *R_=1600; R_=1550; *R_=1450; * R_ еквівалентний опору R;
VC1(C1); VC2(C2); IL1(L1);
INIT:C1(1);
CIN(2,4) 0.1N; VIN(CIN); RIN(2,4) 4000;* Опис моделі нелінійного елемента
;
```

```
R1(2,3) 2000; R3(4,0) 2000; R21(4,6) 2000; R22(6,3) 1200;
```

```
D1(6,3) 'KD'; D2(3,6) 'KD';
```

```
ROUT(5,0) 700; K=500; SAT=9.0/K;
```

```
EOP(5,3) K*VINSAT;
```

```
VINSAT=IF(ABS(VIN).LE.SAT)(VIN)(VIN/ABS(VIN)*SAT);;;
```

TRAN

```
TMAX=40ML; INIT; HS=20MC; HMIN=0.0001MC;
```

```
OUTPUT: VC2(VC1), XMIN=-5, XMAX=5, YMIN=-2.0, YMAX=2.0;;
```

END

На рис.5.12-5.15 відтворені результати комп'ютерного моделювання схеми з рис.5.5 відповідно до завдання у табл.5.1.

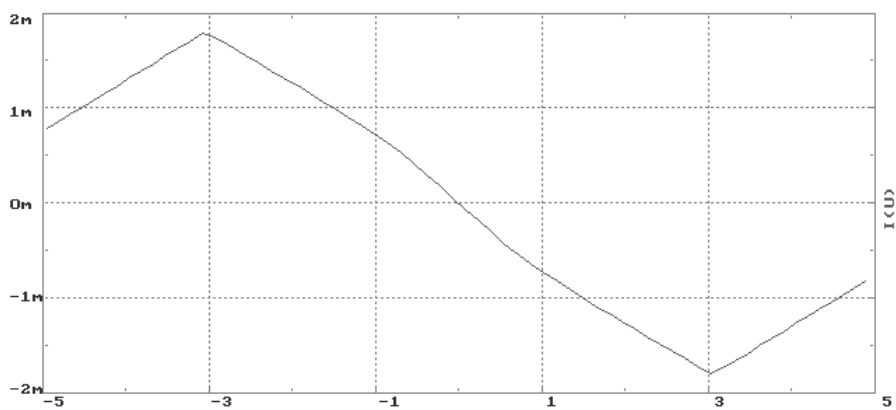
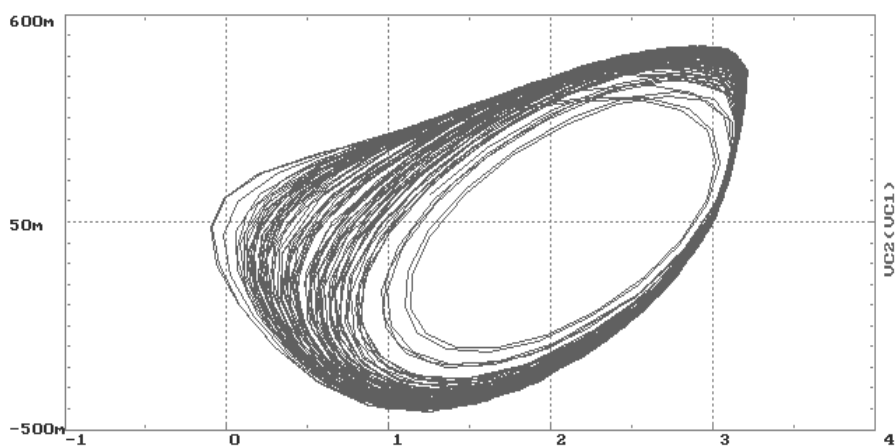
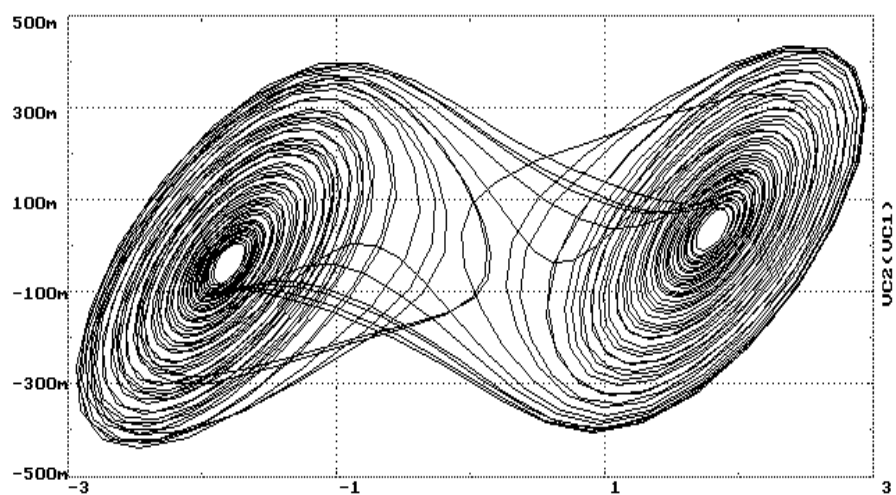


Рис.5.12. Характеристика нелінійного елемента

Рис.5.13. Атрактор Ресслера ($R=1600$ Ом)

201

Рис.5.14. "Подвійна прокрутка" ($R=1550$ Ом)

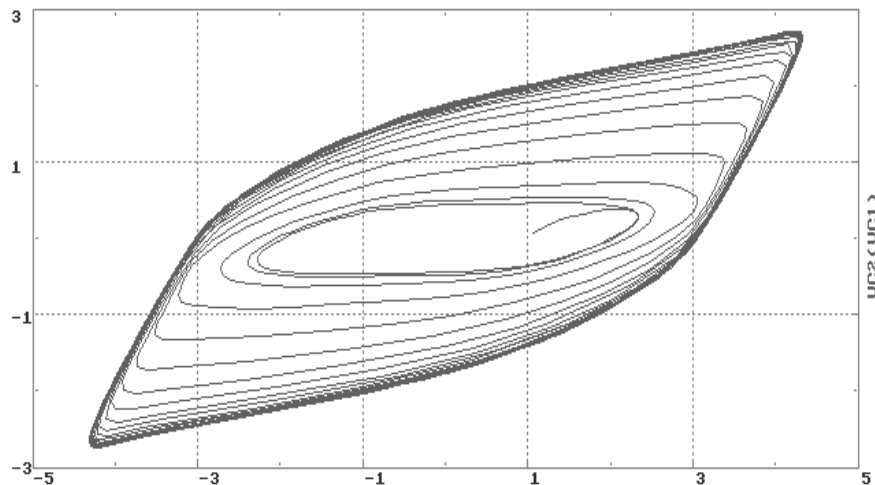


Рис.5.15. Граничний цикл (R=1450 Ом)

Відхилення значень опору $R_{\text{м}}=R$ порівняно з фізичним експериментом пояснюють похибками фізичного моделювання.

Для схеми Чуа обчислено деякі параметри, що характеризують хаотичні рухи, тобто наявність дивного атрактора.

Старший показник Ляпунова обчислювали за методом, описаним у [60]. Суть методу полягає в одночасному інтегруванні двох математичних моделей, що відрізняються лише початковими умовами. Інтегрування відбувається вздовж обраної фазової траєкторії з t_0 до t_N короткими інтервалами Δt , протягом яких рух моделей мало відрізняється від руху в лінійному наближенні. На початку кожного інтервалу задають деяку відстань d_0 між точками, що зображають моделі у фазовому просторі. В кінці i -го інтервалу відстань між точками стає d_i . Старший показник Ляпунова обчислюють за формулою

$$\lambda = \frac{1}{t_N - t_0} \sum_{i=1}^N \ln \frac{d_i}{d_0}. \quad (5.4)$$

Якщо старший показник Ляпунова додатний, то це підтверджує існування дивного атрактора.

Вдалось описати процес обчислення (5.4) вхідною мовою системи САНОС-ПК. У табл.5.2 міститься завдання для системи САНОС-ПК, яке дає змогу обчислювати старший показник Ляпунова для схеми Чуа різними методами інтегрування.

Таблиця 5.2. Завдання для визначення старшого показника Ляпунова
BEGI

Схема Чуа для характеристичних показників Ляпунова
DESC

- * Дослідження старшого показника Ляпунова ;
- * Генератор переривань, t-період, t_i -сумування в ild ,

```

(tii-ti)-запам'ятовування початкових та кінцевих значень за (tt-tii);
t=0.1ML; tii=t/2; kii=if(tt.lt.tii)(kt)(0); tt=mod(T,t); kt=10/tii;
ti=tii/2; ki=if(tt.lt.ti)(kt)(0); HMAX=0.5/kt;
* Основна і збурена системи, (tt-tii)-вільний рух,
kb-призупиняє вільний рух на час вимірювання tii ;
* Основний опис схеми ; kb=(1-kii/kt);
vc1`=kb*1/C1*(1/R*(vc2-vc1)-fv); C1=10N; C2=100N;
vc2`=kb*1/C2*(1/R*(vc1-vc2)+ilv); L=18ML; R=1540; RL=32;
ilv`=kb*(-1/L*(vc2+RL*ilv)); m0=-0.5ML; m1=-0.8ML; uz=1;
fv=m0*vc1+0.5*(m1-m0)*(abs(vc1+uz)-abs(vc1-uz));
init:vc1(3.02249),vc2(0.709851),ilv(-0.000317725);
* Опис збуреної схеми ;
uc1`=kb*1/C1*(1/R*(uc2-uc1)-fu)+kii*(vc1-uc1+dx1);
uc2`=kb*1/C2*(1/R*(uc1-uc2)+ilu)+kii*(vc2-uc2+dx2);
ilu`=kb*(-1/L*(uc2+RL*ilv))+kii*(ilv-ilu+dx3);
fu=m0*uc1+0.5*(m1-m0)*(abs(uc1+uz)-abs(uc1-uz));
init:uc1(3.12249),uc2(0.709851),ilu(-0.000317725);
* Запам'ятовування початкового (Zn) та кінцевого (Zk) значень за (tt-tii);
Y1k`=(kt-kii)*(vc1-uc1-Y1k); Y2k`=(kt-kii)*(vc2-uc2-Y2k);
Y3k`=(kt-kii)*(ilv-ilu-Y3k);      init:Y1k(0),Y2k(0),Y3k(0);
Y1n`=(kii-ki)*(Y1k-Y1n); Y2n`=(kii-ki)*(Y2k-Y2n);
Y3n`=(kii-ki)*(Y3k-Y3n);      init:Y1n(0),Y2n(0),Y3n(0);
dx1=Y1n/dk*dd; dx2=Y2n/dk*dd; dx3=Y3n/dk*dd; dd=0.1;
* Обчислення старшого показника Ляпунова liap сумуванням прямокутних
імпульсів амплітудою ldv та тривалістю ti, що реалізує формулу:
liap=1/(Ткін-Тпоч)*(log2(dkпоч/dnпоч)+...+log2(dkкін/dnкін)).
t0-початковий перехідний процес ;
* Деякі результати для різних tmax і різних методів інтегрування ;
* Для tmax=7ML, t=0.04,R=1705,RL=15,IMPLI:liap=1950(ат.RESSLER'a);
* Для tmax=9ML, t=0.08,R=1705,RL=15,IMPLI:liap=2060(ат.RESSLER'a);
* Для tmax=43ML,t=0.08,R=1705,RL=15, RK:liap=600 (ат.RESSLER'a);
* Для tmax=50ML,t=0.08,R=1618,RL=32, RK:liap=850 (ат.RESSLER'a);
* Для tmax=50ML,t=0.08,R=1540,RL=32, RK:liap=420 (double-scroll);
* Для tmax=100ML,t=0.1,R=1540,RL=32, RK:liap=980 (double-scroll);
t0=10*t; dk=sqrt(Y1k*Y1k+Y2k*Y2k+Y3k*Y3k+1e-9);
dn=sqrt(dx1*dx1+dx2*dx2+dx3*dx3+1e-9);
dv=if(tt.lt.ti)(du)(1); du=if(T.gt.t0)(dk/dn)(1); ldv=log(dv)/log(2);
ild`=ldv; liap=ild/(T*ti*tii/t); init: ild(0.0);
ttt=T/10ML-5; lttt=liap*1e-4;;
TRAN
TMAX=100ML; INIT; HS=0.1ML; rk;
OUTPUT: vc2(vc1), lttt(ttt), liap, T, tabl, xmin=-5,xmax=8,ymin=-1,ymax=1;;;
END

```

При $\Delta t = 0.1$ мс, $t_N - t_0 = 0.05$ с отримані такі значення старшого показника Ляпунова: $\lambda = 850$ (атрактор Ресслера на рис.5.13); $\lambda = 420$ (“подвійна прокрутка” на рис.5.14).

На рис.5.16 показано процес обчислення старшого показника Ляпунова відповідно до завдання у табл.5.2.

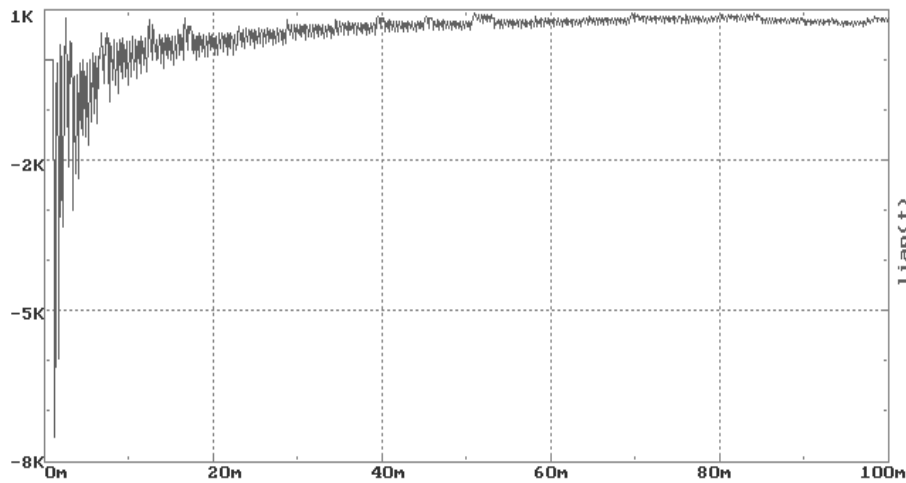


Рис.5.16. Процес обчислення старшого показника Ляпунова

Іншим параметром, що характеризує дивний атрактор, є фрактальна розмірність [60].

Поточкову фрактальну розмірність обчислювали таким методом. Вибирали випадковим чином M точок із загальної кількості N на фазовій траєкторії системи і визначали відстань S_{nm} від n -ї до m -ї. Імовірність потрапляння поточної точки в окіл ε n -ї точки визначається формулою:

$$P_n(\varepsilon) = \frac{1}{N} \sum_{m=1}^N H(\varepsilon - S_{nm}),$$

де $H(\cdot)$ – функція Хевісайда [60], N – загальна кількість точок.

Тоді усереднена поточкова фрактальна розмірність d

$$d = \lim_{\varepsilon \rightarrow 0} \frac{\ln C(\varepsilon)}{\ln \varepsilon}, \quad \text{де } C(\varepsilon) = \frac{1}{M} \sum_{n=1}^M P_n(\varepsilon).$$

Останню границю можна знайти за нахилом прямої у системі координат $\ln \varepsilon - \ln C(\varepsilon)$.

Існування дивного атрактора підтверджує дробове значення фрактальної розмірності.

У Додатку 7 наведена PASCAL-програма, що обчислює зокрема фрактальну розмірність.

Для схеми Чуа фрактальну розмірність обчислювали при $N=2500$ і $M=500$. Отримано такі значення фрактальної розмірності: $d = 2.03$ (атрактор Ресслера з рис.5.13); $d = 2.07$ (“подвійна прокрутка” з рис.5.14).

Перевірено можливість опису ВАХ нелінійного елемента у схемі з рис.5.3 степеневим поліномом третього порядку

$$I = a_3 \cdot U^3 + a_2 \cdot U^2 + a_1 \cdot U + a_0 . \quad (5.5)$$

З умов непарності та проходження ВАХ через початок координат випливає $a_2=0$ та $a_0=0$. Нахил ВАХ у початку координат дорівнює $-m_1$ (рис.5.3), тому $a_1=-m_1$.

Нехай нахил кривої, яку описує поліном, у деякій точці U_0 становить $-m_0$. Тоді $a_3=(m_1-m_0)/(3 \cdot U_0^2)$ і формула (5.5) набуває вигляду

$$I = \frac{m_1 - m_0}{3 \cdot U_0^2} U^3 - m_1 \cdot U .$$

У табл.5.3 міститься текст завдання, де схема Чуа (рис.5.3) описана мовою системи САНОС-ПК диференціальними рівняннями з кусково-лінійною та кубічною поліноміальною апроксимаціями вольт-амперної характеристики нелінійного елемента.

Таблиця 5.3. Завдання для аналізу схеми Чуа з кусково-лінійною та кубічною нелінійністю

BEGI

Схема Чуа, описана диференціальними рівняннями

DESC

* Опис схеми Чуа ;

$vc1' = 1/C1 \cdot (1/R \cdot (vc2 - vc1) - fv)$; $C1=10N$; $C2=100N$;

$vc2' = 1/C2 \cdot (1/R \cdot (vc1 - vc2) + ilv)$; $L=18mH$; $R=1440$; $RL=30$;

$ilv' = (-1/L \cdot (vc2 + RL \cdot ilv))$;

INIT: $vc1(1)$;

* Кусково-лінійна апроксимація нелінійного елемента;

$fv = m_0 \cdot vc1 + 0.5 \cdot (m_1 - m_0) \cdot (abs(vc1 + 1) - abs(vc1 - 1))$; $m_0=-0.5mH$; $m_1=-0.8mH$;

208

* Апроксимація нелінійного елемента кубічним поліномом ;

$fv = a_3 \cdot vc1^3 + a_1 \cdot vc1$; $a_3=0.03mH$; $a_1=-0.79mH$;;;

TRAN

TMAX=60mH; INIT; HS=5mH; RK;

OUTPUT: $vc2(vc1)$, XMIN=-5, XMAX=5, YMIN=-1, YMAX=1;;

END

Результати аналізу схеми Чуа з кубічною апроксимацією на ЕОМ дуже подібні до результатів, отриманих у випадку кусково-лінійної апроксимації.

5.4. ЗАГАЛЬНА СХЕМА МАКРОМОДЕЛЮВАННЯ З ВИКОРИСТАННЯМ СИСТЕМИ САНОС-ПК

Спочатку визначимо опис макромоделей у системах автоматизованого проектування.

Математичне макромодельовання, яке ми розглянули, стосується макромоделей у формі систем диференціальних рівнянь першого порядку, розв'язаних щодо похідної.

Способи зображення таких рівнянь еквівалентними електричними схемами добре відомі. Наприклад, системі $n+k$ алгебро-диференціальних рівнянь загального вигляду

$$\begin{aligned}
 \dot{x}_1 &= f_1(x_1, \dots, x_n, u_1, \dots, u_m); \\
 &\vdots \\
 \dot{x}_n &= f_n(x_1, \dots, x_n, u_1, \dots, u_m); \\
 y_1 &= g_1(x_1, \dots, x_n, u_1, \dots, u_m); \\
 &\vdots \\
 y_k &= g_k(x_1, \dots, x_n, u_1, \dots, u_m); \\
 x_1(0) &= x_{10}; \dots; x_n(0) = x_{n0}
 \end{aligned} \tag{5.6}$$

відповідає еквівалентна електрична схема з ідеальними керованими джерелами струму та напруги, зображена на рис.5.17. Змінні стану цієї схеми – напруги на конденсаторах, а початкові умови задано початковими напругами на конденсаторах.

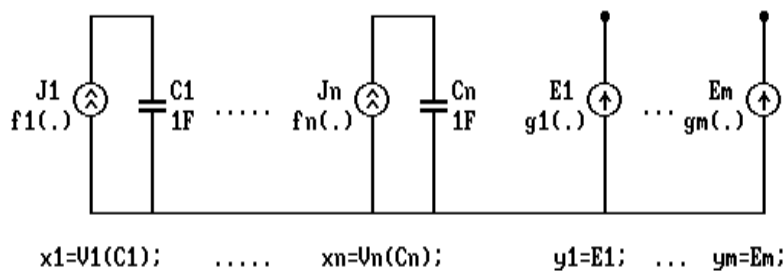


Рис.5.17. Електрична схема, еквівалентна рівнянням (5.6)

Опис таких схем вхідними мовами САПР радіоелектроніки не становить ніяких труднощів.

Система САНОС-ПК має вхідну мову, що є типовою для відомих САПР електроніки. Помітним доповненням є можливість записувати диференціальне рівняння першого порядку, розв'язане щодо похідної, безпосередньо вхідною мовою.

На рис.5.18 показана блок-схема процесу макромодельовання з використанням програм, на які ми посилаємось.



Рис.5.18. Загальна схема макромоделювання з використанням системи САНОС-ПК та програм апроксимації

Як бачимо, система САНОС-ПК використана в трьох блоках з п'яти. Дані від системи САНОС-ПК до програм апроксимації передаються через робочий файл, у якому в стандартній табличній формі накопичуються результати числового аналізу.

Реалізовані та досліджені на практиці два способи зворотного автоматичного передавання даних від програм апроксимації до системи САНОС-ПК.

Перший спосіб полягає у формуванні програмою апроксимації опису макромоделі вхідною мовою САНОС-ПК у вигляді моделі та відповідних типів. Такий спосіб реалізовано в програмі, наведеній у Додатку 8.

За другим способом макромоделі формується програмою апроксимації у вигляді готового завдання мовою САНОС-ПК. Саме такий спосіб, як зручніший, реалізовано в інших апроксимаційних програмах, на які є посилання (Додатки 1, 4).

Застосування системи САНОС-ПК суттєво полегшує процес макромоделювання, автоматизує передавання даних між блоками, дає змогу оперативно оцінити якість макромоделі.

ДОДАТКИ

За адресою автора matv@ua.fm міститься система САНОС-ПК з усіма прикладами, на які є посилання у монографії.

Додаток 1. FORTRAN-програма апроксимації лінійної макромоделі за середньо-квадратичним критерієм.

Додаток 2. FORTRAN-програма апроксимації за мінімаксним критерієм.

Додаток 3. MATLAB-програма макромодельовання лінійних систем за частотними характеристиками.

Додаток 4. FORTRAN-програма апроксимації статичної характеристики тригера.

Додаток 5. FORTRAN-програма сплайн-інтерполяції та обчислення похідних для послідовності псевдовипадкових чисел.

Додаток 6. FORTRAN-програма апроксимації з редукцією апроксимуючого поліному.

Додаток 7. PASCAL-програма обрахунку кореляційної розмірності та розмірності вкладення за дискретними послідовностями.

Додаток 8. FORTRAN-програма автоматичного розбиття на підобласті у просторі аргументів та регуляризованої апроксимації у межах кожної підобласті.

Додаток 9. Модель мовою САНОС-ПК, яка відповідає макромоделі з розбиттям на 4 підобласті та автоматичним вибором підобласті, найближчої до поточних координат.

Додаток 1
 FORTRAN-програма апроксимації лінійної макромоделі за
 середньо-квадратичним критерієм.

```

с      Апроксимація макромоделі з обчисленими похідними
с      та заданим порядком 1<=n<=5.
с      Початкові дані для апроксимації містяться у fileapr.san
с      по порядку:  u,u`,u`i,...,u`n,y,y`,...,y`n.
с      Результати записуються у maklinob.mod як готове завдання
с      для САНОСа у вигляді одного дифрівняння n-го порядку
с      та системи рівнянь у змінних стану n-го порядку.
с      Мінімізується середньоквадратична похибка по вектору Y
с      Здійснюється регуляризація по  $\min(|F*W-Y|^2 + \alpha * |W|^2)$ 
с      Жорсткість регуляризації Більше, якщо alfa Більше.
с      Якщо alfa = 0.0 , то регуляризація відсутня.
      SUBROUTINE APRLIN(n)
      INTEGER B,E,e1,M,n
      REAL*8 A,S,G,EPS/1.D-18/,W(30),F(13,500),L(13,13)
      REAL T(500),X(500),X1(500),X2(500),X3(500),X4(500),X5(500),
*Y(500),U(500),U1(500),U2(500),U3(500),U4(500),U5(500),U6(500)
      REAL BW(30),v(30),c(60)
      b=1
      e1=500
      alfa=0
      open (1, file='con')
      open (10, file='fileapr.san')
      open (2, file='rez1.dat')
      open (4, file='maklinob.mod')
      DO 119 I=1,30
119  W(I)=0.0
      read(10,'(I4)') M
      read(10,'(I4)') M
      read(10,'(I4)') M
      M=M-1
      nm=(M-2)/2
      if(n.gt.5.or.n.lt.1) then
      write(*,*)' Помилка :'
      write(*,*)' порядок макромоделі поза допустимими межами 1-5.'
      n=-1
      goto 1000
      endif
      if(n.gt.nm) then
      write(*,*)' Помилка :'
      write(*,*)' для заданого порядку макромоделі замало похідних.'
      n=-2
      goto 1000
      endif
      e=1
120  continue
      if(nm.eq.1) read(10,'(15E15.7)',end=110) T(e),U(e),U1(e),X(e),Y(e)
      if(nm.eq.2) read(10,'(15E15.7)',end=110) T(e),U(e),U1(e),U2(e),
* X(e),X1(e),Y(e)
      if(nm.eq.3) read(10,'(15E15.7)',end=110) T(e),U(e),U1(e),U2(e),
* U3(e),X(e),X1(e),X2(e),Y(e)
      if(nm.eq.4) read(10,'(15E15.7)',end=110) T(e),U(e),U1(e),U2(e),
* U3(e),U4(e),X(e),X1(e),X2(e),X3(e),Y(e)
      if(nm.eq.5) read(10,'(15E15.7)',end=110) T(e),U(e),U1(e),U2(e),
* U3(e),U4(e),U5(e),X(e),X1(e),X2(e),X3(e),X4(e),Y(e)
      if(nm.eq.6) read(10,'(15E15.7)',end=110) T(e),U(e),U1(e),U2(e),
* U3(e),U4(e),U5(e),U6(e),X(e),X1(e),X2(e),X3(e),X4(e),X5(e),Y(e)
      e=e+1
      if(e.le.e1) goto 120
110  e=e-1

```

```

        if(n.lt.nm) then
        select case (n)
        case(1)
            do 111 i=1,e
111      Y(i)=X1(i)
        case(2)
            do 112 i=1,e
112      Y(i)=X2(i)
        case(3)
            do 113 i=1,e
113      Y(i)=X3(i)
        case(4)
            do 114 i=1,e
114      Y(i)=X4(i)
        case(5)
            do 115 i=1,e
115      Y(i)=X5(i)
        end select
        endif
        M=n*2+2
        e1=e
        if(alfa.ne.0.0) e=e+m*2
c      Регуляризаційна доБавка від e1+1 до e, всього m*2 рівнянь.
        do 121 i=1,2
        write(i,*)' file          beg.arr. end arr. nomb.coef.',
*      '      EPS'
121      write(i,'(A13,3I10,E14.1)') name, B, E, M, EPS
            if(alfa.eq.0.0) goto 126
            do 124 i=e1+1,e
124      y(i)=0.0
c      Формування регуляризаційних констант.
            do 125 i=1,m*2,2
            c(i)=alfa
125      c(i+1)=-alfa
126      continue
            do 151 i=1,e
151      t(i)=i
            BY=0
            DO 131 I=B,E
131      BY=BY+ABS(Y(I))
            BY=BY/(E-B+1)
            IF(BY.LE.1E-50) BY=1
            DO 132 I=B,E
            Y(I)=Y(I)/BY
            IF(ABS(Y(I)).LT.EPS) Y(I)=0.0
132      CONTINUE
            DO 101 I=1,M
            BW(I)=0
101      W(I)=0
            DO 1 I=1,M
            DO 2 J=1,M
2      W(J)=0          W(I)=1          G=0
            DO 3 K=B,E1
        select case (n)
        case(1)
310      Z1=U(K)
            Z2=U1(K)
            Z3=X(K)
            F(I,K)=W(1) +W(2)*Z1 +W(3)*Z2 +W(4)*Z3
        case(2)
320      Z1=U(K)
            Z2=U1(K)
            Z3=U2(K)
            Z4=X(K)

```

```

      Z5=X1 (K)
      F(I,K)=W(1) +W(2)*Z1 +W(3)*Z2 +W(4)*Z3 +W(5)*Z4 +W(6)*Z5
      case(3)
330      Z1=U(K)
          Z2=U1 (K)
          Z3=U2 (K)
          Z4=U3 (K)
          Z5=X (K)
          Z6=X1 (K)
          Z7=X2 (K)
          F(I,K)=W(1) +W(2)*Z1 +W(3)*Z2 +W(4)*Z3 +W(5)*Z4 +W(6)*Z5
          *   +W(7)*Z6 +W(8)*Z7
          case(4)
340      Z1=U(K)
          Z2=U1 (K)
          Z3=U2 (K)
          Z4=U3 (K)
          Z5=U4 (K)
          Z6=X (K)
          Z7=X1 (K)
          Z8=X2 (K)
          Z9=X3 (K)
          F(I,K)=W(1) +W(2)*Z1 +W(3)*Z2 +W(4)*Z3 +W(5)*Z4 +W(6)*Z5
          *   +W(7)*Z6 +W(8)*Z7 +W(9)*Z8 +W(10)*Z9
          case(5)
350      Z1=U(K)
          Z2=U1 (K)
          Z3=U2 (K)
          Z4=U3 (K)
          Z5=U4 (K)
          Z6=U5 (K)
          Z7=X (K)
          Z8=X1 (K)
          Z9=X2 (K)
          Z10=X3 (K)
          Z11=X4 (K)
          F(I,K)=W(1) +W(2)*Z1 +W(3)*Z2 +W(4)*Z3 +W(5)*Z4 +W(6)*Z5
          *   +W(7)*Z6 +W(8)*Z7 +W(9)*Z8 +W(10)*Z9 +W(11)*Z10 +W(12)*Z11
      end select
3      G=G+DABS(F(I,K))
с      Регуляризаційна доБавка.
          if(alfa.eq.0.0) goto 333
          do 332 j=e1+1,e
332      f(i,j)=0.0
          f(i,e1+i*2-1)=c(i*2-1)
          f(i,e1+i*2)=c(i*2)
          g=g+dabs(f(i,e1+1))+dabs(f(i,e1+2))
333      continue
      BW(I)=G/(E-B+1)
      IF(G.LE.1E-50) BW(I)=1
1      CONTINUE
      DO 141 I=1,M
      DO 141 K=B,E
      F(I,K)=F(I,K)/BW(I)
      IF(DABS(F(I,K)).LT.EPS) F(I,K)=0.0
141      CONTINUE
      DO 20 I=1,M
      L(I,I)=1
      I1=I-1
      IF(I1.LT.1) GOTO 41
      DO 4 J=1,I1
          IF(W(J).LT.EPS) GOTO 5
      S=0
      DO 6 K=B,E

```

```

      G=F(I,K)*F(J,K)
6      S=S-G
      L(I,J)=S/W(J)
      GO TO 4
5      L(I,J)=0
4      CONTINUE
41     DO 8 K=B,E
      S=0
      DO 11 J=1,I
      G=L(I,J)*F(J,K)
11     S=S+G
      8     F(I,K)=S
      I2=I-2
      IF(I2.LT.1) GOTO 12
      DO 9 J=1,I2
      S=0
      DO 7 K=J,I1
      G=L(I,K)*L(K,J)
7      S=S+G
9      L(I,J)=S
12     S=0
      DO 10 K=B,E
      G=F(I,K)*F(I,K)
10     S=S+G
20     W(I)=S
      DO 21 I=1,M
      IF(W(I).LT.EPS) GO TO 22
      S=0
      DO 23 K=B,E
      G=Y(K)*F(I,K)
23     S=S+G
      W(I)=S/W(I)
      GO TO 21
22     W(I)=0
21     CONTINUE
      A=0
      DO 25 K=B,E
      S=0
      DO 26 J=1,M
26     S=S+W(J)*F(J,K)
      X1(K)=S*BY
25     A=A+(Y(K)-S)**2
      A=DSQRT(A/(E-B+1))
      DO 27 I=1,M
      S=0
      DO 28 K=I,M
      G=W(K)*L(K,I)
28     S=S+G
27     W(I)=S
      DO 30 I=1,M
30     W(I)=W(I)*BY/BW(I)
      VE=0.0
      Z=0.0
      DO 31 I=B,E
      Y(I)=Y(I)*BY
      IF(ABS(Y(I)-X1(I)).GT.VE) VE=ABS(Y(I)-X1(I))
      if(abs(Y(I)).ne.0.0) VY=VE/ABS(Y(I))
31     IF(VY.GT.Z.AND.VY.LT.100) Z=VY
      do 401 j=1,2
      write(j,*)' mid.qr.err. max.abs.err. max.rel.err.'
      write(j,'(6D13.4)') A, VE, Z
      write(j,*)' Коефіцієнти апроксимації '
401    write(j,'(6D13.4)') (W(I),I=1,M)
      do 501 i=1,m

```

```

501      v(i)=W(I)
c***** Завдання для SANOSa *****
      write(4, '(A4)') 'BEGI'
601      write(4, *) 'Апроксимаційна макромодель', n, '-го порядку.'
      write(4, '(A4)') 'DESC'
      write(4, *) '* Макромодель лінійного об'єкта ;'
      write(4, *) '* Кількість точок =', el-b+1, ';'
      write(4, *) '* Коефіцієнти апроксимації ;'
      select case (n)
      case(1)
701      write(4, *) ' B0=', v(2), '; B1=', v(3), '; A0=', -v(4), ';'
      write(4, *) '* Рівняння 1-го степеня ;'
      write(4, *) ' *Y`=B0*U +B1*U1 -A0*Y;'
      write(4, *) '* Рівняння у змінних стану ;'
      write(4, *) ' X1`=(-A0*X1 +(B0-A0*B1)*U);'
      write(4, *) ' Y=X1 +B1*U; EY(0,100)Y; VY(EY);'
      case(2)
702      write(4, *) ' B0=', v(2), '; B1=', v(3), '; B2=', v(4), ';'
      write(4, *) ' A0=', -v(5), '; A1=', -v(6), ';'
      write(4, *) '* Рівняння 2-го степеня ;'
      write(4, *) ' *Y`=Y1;'
      write(4, *) ' *Y1`=B0*U +B1*U1 +B2*U2 -A0*Y0 -A1*Y1;'
      write(4, *) '* Рівняння у змінних стану ;'
      write(4, *) ' X1`=(-A1*X1 +X2 +(B1-A1*B2)*U);'
      write(4, *) ' X2`=(-A0*X1 +(B0-A0*B2)*U);'
      write(4, *) ' Y=X1 +B2*U; EY(0,100)Y; VY(EY);'
      case(3)
703      write(4, *) ' B0=', v(2), '; B1=', v(3), '; B2=', v(4), ';'
      write(4, *) ' B3=', v(5), '; A0=', -v(6), '; A1=', -v(7), ';'
      write(4, *) ' A2=', -v(8), ';'
      write(4, *) '* Рівняння 3-го степеня ;'
      write(4, *) ' *Y`=Y1; *Y1`=Y2;'
      write(4, *) ' *Y2`=B0*U+B1*U1+B2*U2+B3*U3-A0*Y-A1*Y1-A2*Y2;'
      write(4, *) '* Рівняння у змінних стану ;'
      write(4, *) ' X1`=(-A2*X1 +X2 +(B2-A2*B3)*U);'
      write(4, *) ' X2`=(-A1*X1 +X3 +(B1-A1*B3)*U);'
      write(4, *) ' X3`=(-A0*X1 +(B0-A0*B3)*U);'
      write(4, *) ' Y=X1 +B3*U; EY(0,100)Y; VY(EY);'
      case(4)
704      write(4, *) ' B0=', v(2), '; B1=', v(3), '; B2=', v(4), ';'
      write(4, *) ' B3=', v(5), '; B4=', v(6), '; A0=', -v(7), ';'
      write(4, *) ' A1=', -v(8), '; A2=', -v(9), '; A3=', -v(10), ';'
      write(4, *) '* Рівняння 4-го степеня ;'
      write(4, *) ' *Y`=Y1; *Y1`=Y2; *Y2`=Y3;'
      write(4, *) ' *Y3`=B0*U +B1*U1 +B2*U2 +B3*U3 +B4*U4 -A0*Y'
      write(4, *) ' -A1*Y1 -A2*Y2 -A3*Y3;'
      write(4, *) '* Рівняння у змінних стану ;'
      write(4, *) ' X1`=(-A3*X1 +X2 +(B3-A3*B4)*U);'
      write(4, *) ' X2`=(-A2*X1 +X3 +(B2-A2*B4)*U);'
      write(4, *) ' X3`=(-A1*X1 +X4 +(B1-A1*B4)*U);'
      write(4, *) ' X4`=(-A0*X1 +(B0-A0*B4)*U);'
      write(4, *) ' Y=X1 +B4*U; EY(0,100)Y; VY(EY);'
      case(5)
705      write(4, *) ' B0=', v(2), '; B1=', v(3), '; B2=', v(4), ';'
      write(4, *) ' B3=', v(5), '; B4=', v(6), '; B5=', v(7), ';'
      write(4, *) ' A0=', -v(8), '; A1=', -v(9), '; A2=', -v(10), ';'
      write(4, *) ' A3=', -v(11), '; A4=', -v(12), ';'
      write(4, *) '* Рівняння 5-го степеня ;'
      write(4, *) ' *Y`=Y1; *Y1`=Y2; *Y2`=Y3; *Y3`=Y4;'
      write(4, *) ' *Y4`=B0*U +B1*U1 +B2*U2 +B3*U3 +B4*U4 +B5*U5'
      write(4, *) ' -A0*Y -A1*Y1 -A2*Y2 -A3*Y3 -A4*Y4;'
      write(4, *) '* Рівняння у змінних стану ;'
      write(4, *) ' X1`=(-A4*X1 +X2 +(B4-A4*B5)*U);'
      write(4, *) ' X2`=(-A3*X1 +X3 +(B3-A3*B5)*U);'

```

```

write(4,*)' X3`=(-A2*X1          +X4          + (B2-A2*B5)*U);'
write(4,*)' X4`=(-A1*X1          +X5          + (B1-A1*B5)*U);'
write(4,*)' X5`=(-A0*X1          + (B0-A0*B5)*U);'
write(4,*)' Y=X1 +B5*U; EY(0,100)Y; VY(EY);'
end select
write(4,*)'* ОБчислення похідних вхідного сигналу '
write(4,*)' для рівняння n-го степеня ;'
write(4,*)' *w`=U1; *U1=400*(U-w);'
write(4,*)' *w1`=U2; *U2=400*(U1-w1);'
write(4,*)' *w2`=U3; *U3=400*(U2-w2);'
write(4,*)' *w3`=U4; *U4=400*(U3-w3);'
write(4,*)' *w4`=U5; *U5=400*(U4-w4);'
write(4,*)'* Лінійний об'єкт ;'
write(4,*)' E1(0,1); RE1(1,2)75; L2(2,3)20; L3(3,4)20; RN(4,0)75;'
write(4,*)' L1(2,8)16; C1(8,0)3.9ml; C2(3,0)12ml;'
write(4,*)' VRE1(RE1); VRN(RN);;'
write(4,*)' E1=0; MVRN=M(VRN); MY=M(VY); U=E1;;;'
write(4,*(A4))'AC '
write(4,*)'E1=(1,0); F=A(0.1,1)100; zero;'
write(4,*)'output: MVRN,MY, xmin=0.1,xmax=1,ymin=0,ymax=1;;;'
write(4,*(A4))'DESC'
write(4,*)' E1=if(T.lt.0.1)(10*T)(1);;'
write(4,*(A4))'TRAN'
write(4,*)'tmax=10; zero;'
write(4,*)'output: VRN,Y, xmax=10,ymin=-0.5,ymax=1;;;'
write(4,*(A4))'END '
1000 return
end
c Main program
write(*,*)' Введіть порядок макромоделі:'
read(*,*) in
CALL APRLIN(in)
end

```

Додаток 2

FORTTRAN-програма апроксимації за мінімаксним критерієм.

```

C АПРОКСИМАЦІЯ ПО ЧЕЮИШЕВУ
INTEGER M,N,K,IHE(4460),ITER(1150)
REAL Z1(1000),SX1/0/,SX2/0/,SX3/0/,SDE1/0/,
*G(1000),X1(1000),X2(1000),Z3(1000),
*DE1(1000),X3(1000),PIVN(200),NORM(200)
DOUBLE PRECISION GD(1000),T(23104),TOP(3000),PIV(460),Y1(200),
*X1D(1000),X2D(1000),X3D(1000),DE1D(1000),Z3D(1000),H,X4D(1000),
*Y(200)
EXTERNAL FCT1
C ЗАДАННЯ К?ЛЬКОСТ? КОЕФ. ? ТОЧОК
M=35
N=850
K=M-1
BB=5
C АВ- КІЛЬКІСТЬ ДОДАТКОВИХ РЕГ.УМОВ НА ПЕРШІ АВ КОЕФІЦІЄНТИ
AB=10
N=N+AB
C НАЗВИ ЗОВНІШНІХ ФАЙЛІВ
OPEN (1,FILE='COEFN352.DAT') OPEN (3,FILE='FILEAPR2.SAN')
READ(3,*) VV
READ(3,*(15A4)) VVV
READ(3,*) VVVV
C ЧИТАННЯ ВХІДНИХ ЗМІННИХ
DO 11 I=BB,N
READ(3,*(15E15.7)',END=10) G(I-BB+1),X1(I-BB+1),X2(I-BB+1),

```

```

      *X3 (I-BB+1) , DE1 (I-BB+1) , Z3 (I-BB+1)
      II=I-BB+1
      GD (II) =DBLE (G (II) )
      X1D (II) =DBLE (X1 (II) )
      X2D (II) =DBLE (X2 (II) )
      X3D (II) =DBLE (X3 (II) )
      DE1D (II) =DBLE (DE1 (II) )
11  Z3D (II) =DBLE (Z3 (II) )
      X4D (II) =DE1D (II)
10  CONTINUE
      N=N-BB+1
      DO 101 J=1,M
      Y (J) =0.D0
      NORM (J) =0
101  Y1 (J) =0.D0
C  ВИРАХУВАННЯ НОРМУЮЧИХ КОЕФ.
      N=N-AB
      DO 12 I=1,N
      CALL FCT1 (Y, I, X1D, X2D, X3D, DE1D)
      DO 122 J=2,M
122  NORM (J) =NORM (J) +SNGL (DABS (Y (J) ) )
      12  SZ3=SZ3+DABS (Z3D (I) )
      DO 121 J=1,M-1
121  NORM (J) =NORM (J+1) /N
      NORM (M) =SZ3/N
      N=N+AB
      DO 96 I=1,M
      IF (NORM (I) ) 94, 93, 94
93  NORM (I) =1.E-40
      GOTO 96
94  IF (1.E38-NORM (I) ) 95, 96, 96
95  NORM (I) =1.E38
96  CONTINUE
      CALL MINMAX (NORM, M, FMIN, FMAX)
      DO 13 I=1,N
      X1D (I) =X1D (I) /NORM (1)
      X2D (I) =X2D (I) /NORM (4)
      X3D (I) =X3D (I) /NORM (10)
      DE1D (I) =DE1D (I) /NORM (20)
13  Z3D (I) =Z3D (I) /NORM (M)
C  ПідГОТОВКА ВЕКТОРА TOP
      DO 1 I=1,N
      1  TOP (I+N) =GD (I)
      DO 111 I=1,N
111  TOP (I) =Z3D (I)
      DO 112 I=N-AB+1,N
112  TOP (I) =0
C  ВИКЛИК РОБОЧОЇ ПідПРОГРАМИ
      CALL АРММ (FCT1, N, M, TOP, IHE, PIV, T, ITER, IER, X1D, X2D, X3D, DE1D)
C  ДЕНОРМУВАННЯ ПОРАХОВАНИХ КОЕФ.
      PIV (1) =PIV (1) *NORM (M)
      DO 14 J=2,M
14  PIV (J) =PIV (J) *NORM (M) /NORM (J-1)
      DO 140 J=1,M
      IF (ABS (PIV (J) ) -1.E38) 145, 145, 141
141  IF (PIV (J) ) 142, 143, 144
142  PIVN (J) =-1.E38
      GOTO 140
143  PIVN (J) =0.E0
      GOTO 140
144  PIVN (J) =1.E38
      GOTO 140
145  PIVN (J) =SNGL (PIV (J) )
140  CONTINUE

```



```

C          ВИВІД РЕЗУЛЬТАТІВ
      DO 2 I=1,M
      WRITE(*,'(A3,I2,A2,E11.5)') ' A(',I,')=',PIVN(I)
2  WRITE(1,'(A2,I2,A1,E11.5,A1)') ' A',I, '=',PIVN(I),';'
      WRITE(1,*) '-----'

      WRITE(1,'(A11,I2)') 'ERROR CODE=',IER
      IF(IER) 7,6,8
6  WRITE(1,*) '*****'
      GOTO 9
C          ПОВІДОМЛЕННЯ ПРО ПОМИЛКИ В РОБОЧІЙ ПІДПРОГРАМІ
7  WRITE(*,*) 'ERROR IN INPUT PARAMETR M OR N OR SINSE AT SOME
*ITERATION NO SUITABLE PIVOT COULD BE FOUND,PROGRAM ABORTED'
      GO TO 9
8  WRITE(*,*) 'ERROR,CRITICAL NUMBER OF ITERATIONS > (N+M) '
      WRITE(*,*) ' PROGRAM ABORTED'
      GO TO 9
9  CONTINUE
C  ПОВУДОВА АПРОКСИМУЮЧОЇ ФУНКЦІЇ
      N=N-AB
      DO 5 I=1,N
      H=0.D0
      DO 4 J=2,M
      CALL FCT1(Y1,I,X1D,X2D,X3D,DE1D)
4  H=H+PIVN(J)*SNGL(Y1(J))*NORM(J-1)
      H=H+PIVN(1)
      IF (ABS(H)-1.E38) 42,42,41
41 IF(H) 412,413,411
411 Z1(I)=1.E38
      GOTO 5
412 Z1(I)=-1.E38
      GOTO 5
413 Z1(I)=0.E0
      GOTO 5
42 Z1(I)=SNGL(H)
5  CONTINUE
C  ВИРАХУВАННЯ ВІДНОСНОЇ ПОХИБКИ
      ACS=0
      DO 3 I=1,N
      IF (Z3(I)) 032,031,032
31 CONTINUE
      Z3(I)=1.E-18
32 AC=(Z3(I)-Z1(I))
      B=AC/Z3(I)*100
      ACS=ACS+ABS(AC)
3  WRITE(1,'(A2,I4,A2,E12.5,A6,E12.5,A7,E12.5,A7,E12.5)')
*'F(',I,')=',Z3(I),'PR=',SNGL(TOP(I)),'REL=',B,'ABS=',AC
      WRITE(1,'(A5,E12.5)') 'SABS=',ACS/N
      WRITE(*,'(A5,E12.5)') 'SABS=',ACS/N
      GO TO 99
CC          РОБОЧА ПІДПРОГРАМА      SUBROUTINE APRMM
CC  Апроксимація функції,табульованої в N точках,Будь-якою лінійною
CC  комбінацією M даних неперервних функцій в сенсі Чебишева.
CC  ВИКОРИСТАННЯ
CC  CALL APRMM(FCT,N,M,TOP,INE,PIV,T,ITER,IER)
CC  Параметр FCT вимагає зовнішнього задання
CC  ОПИС ПАРАМЕТРІВ
CC  FCT-назва підпрограми,що задається користувачем.Вираховує
CC  значення M даних неперервних функцій для значень аргумента X
CC          ВИКОРИСТАННЯ
CC          CALL FCT(Y,X,K)
CC          Y-результуючий вектор виміру M,який містить значення даних
CC          неперервних функцій для даного аргумента X
CC          X-значення аргумента

```

```

CC      K-ціле число рівне M-1
CC      N-кількість точок, в яких визначена функція, що апроксимується
CC      M-кількість даних неперервних функцій, з яких конструюється
CC      апроксимаційна функція
CC      TOP-вектор виміру 3*N
CC      на вході має містити з TOP(1) до TOP(N) N даних значень
CC      функції і з TOP(N+1) до TOP(2*N) значення аргументу у
CC      відповідних вузлах
CC      на виході з TOP(1) до TOP(N) помилки в цих вузлах
CC      інші значення випадкові
CC      IHE-цілочисельний вектор виміру 3*M+4*N+6
CC      PIV-вектор виміру 3*M+6
CC      на виході містить коефіцієнти лінійної апроксимації
CC      з PIV(1) до PIV(M)
CC      T-допоміжний вектор розмірністю (2*M)*(2*M)
CC      ITER-результуюче ціле число, кількість зроблених ітерацій

CC      IER-код помилки
CC      IER=0-немає помилок
CC      IER=1-кількість ітерацій досягла внутрішнього максимуму N+M
CC      IER=-1-немає результату, бо неправильно введено параметр M
CC      або N чи на деякій ітерації не може бути знайдений базис,
CC      що підходить
CC      МЕТОД
CC      Задача апроксимації табульованої функції будь-якою лінійною
CC      комбінацією даних функцій в сенсі Чебишева (тобто мінімізувати
CC      максимальне відхилення) зводиться до задачі лінійного програм.
CC      АРММ використовує симплекс метод для розв'язування відповідної
CC      дуальної задачі.
CC      Для уточнення дивіться
CC      I.BARODALE/A.YOUNG, ALGORITHMS FOR BEST L-SUB-ONE AND
CC      L-SUB-INFINITY LINEAR APPROXIMATIONS ON A DISCRETE SET,
CC      NUMERISHE MATHEMATIK, VOL.8.ISS.3, (1966), PP.295-306C.
CC
CC      SUBROUTINE APMM(FCT1,N,M,TOP,IHE,PIV,T,ITER,IER,X1D,X2D,X3D,DE1D)
CC      DOUBLE PRECISION DSUM,TOL,HELP,SAVE,TOP,T,REPI,PIV,X1D,X2D,
CC      *X3D,DE1D
CC      DIMENSION IHE(1),T(1),TOP(1),PIV(1),
CC      *X1D(1000),X2D(1000),X3D(1000),DE1D(1000),NORM(200)
CC      IER=-1
CC      IF (N-1) 81,81,1
1  IF(M) 81,81,2
2  IER=0
CC      DO 3 I=1,N
CC      K=I+N
CC      J=K+N
CC      TOP(J)=TOP(K)
3  TOP(K)=-TOP(I)
CC      L=M+2
CC      LL=L*L
CC      DO 4 I=1,LL
4  T(I)=0.D0
CC      K=1
CC      J=L+1
CC      DO 5 I=1,L
CC      T(K)=1.D0
5  K=K+J
CC      DO 6 I=1,L
CC      K=I+L
CC      J=K+L
CC      IHE(I)=0
CC      IHE(K)=I
6  IHE(J)=1-I
CC      NAN=N+N

```

```

      K=L+L+L
      J=K+NAN
      DO 7 I=1,NAN
      K=K+1
      IHE(K)=I
      J=J+1
7    IHE(J)=I
      ITER=-1
8    ITER=ITER+1
      IF(N+M-ITER) 9,9,10
9    IER=1
      GO TO 69
10   ISE=0
      IPIV=0
      K=L+L+L
      SAVE=0.D0
      DO 14 I=1,NAN
      IDO=K+I
      HELP=TOP(I)
      IF(HELP-SAVE) 12,12,11
11   SAVE=HELP
      IPIV=I
12   IF(IHE(IDO)) 14,13,14
13   ISE=I
14   CONTINUE
      IF(IPIV) 69,69,15
15   ILAB=1
      IND=0
      J=ISE
      IF(J) 21,21,34
16   K=(K-1)*L
      DO 17 I=1,L
      J=L+I
      K=K+1
17   PIV(J)=T(K)
18   IF(ISE) 22,22,19
19   ISE=-ISE
      J=L+1
      IDO=L+L
      DO 20 I=J,IDO
      K=I+L
20   PIV(K)=PIV(I)
21   J=IPIV
      GO TO 34
22   SAVE=1.D38
      IDO=0
      K=L+1
      LL=L+L
      IND=0
      DO 29 I=K,LL
      J=I+L
      HELP=PIV(I)
      IF(HELP) 29,29,23
23   HELP=-HELP
      IF(ISE) 26,24,26
24   IF(IHE(J)) 27,25,27
25   IDO=I
      GO TO 29
26   IF(HELP) 262,261,262
261  PAUSE
      HELP=1.D-30
262  HELP=-PIV(J)/HELP
27   IF(HELP-SAVE) 28,29,29
28   SAVE=HELP

```

```

      IND=I
29  CONTINUE
      IF(IND) 30,30,32
30  IF(IDO) 68,68,31
31  IND=IDO
32  IF(PIV(IND)) 322,321,322
321 PAUSE
      PIV(IND)=1.D-20
322 REPI=1.D0/PIV(IND)
      IF(1.D38-REPI) 323,324,324
323 REPI=1.D38
324 IND=IND-L
      ILAB=0
      SAVE=-TOP(IPIV)*REPI
      TOP(IPIV)=SAVE
      J=NAN
33  IF(J-IPIV) 34,53,34
34  K=0
      DO 36 I=1,L
      IF(IHE(I)-J) 36,35,36
35  K=I
      IF(ILAB) 50,50,16
36  CONTINUE
      I=L+L+L+NAN+J
      I=IHE(I)-N
      IF(I) 37,37,38
37  I=I+N
      K=1
38  I=I+NAN
      I=I-NAN
      CALL FCT1(PIV,I,X1D,X2D,X3D,DE1D)
      DSUM=0.D0
      IDO=M
      DO 41 I=1,M
      HELP=PIV(IDO)
      IF(K) 39,39,40
39  HELP=-HELP
40  DSUM=DSUM+HELP
      PIV(IDO+1)=HELP
41  IDO=IDO-1
      PIV(L)=-DSUM
      PIV(1)=1.D0
      IDO=IND
      IF(ILAB) 44,44,42
42  K=1
43  IDO=K
44  DSUM=0.D0
      HELP=0.D0
      DO 46 I=1,L
      DSUM=DSUM+PIV(I)*T(IDO) TOL=DABS(DSUM)
      IF(TOL-HELP) 46,46,45
45  HELP=TOL
46  IDO=IDO+L
      TOL=1.D-5*HELP
      IF(DABS(DSUM)-TOL) 47,47,48
47  DSUM=0.D0
48  IF(ILAB) 51,51,49
49  I=K+L
      PIV(I)=DSUM
      K=K+1
      IF(K-L) 43,43,18
50  I=(K-1)*L+IND
      DSUM=T(I)
51  DSUM=DSUM*SAVE

```

```

      TOL=1.D-5*DABS(DSUM)
      TOP(J)=TOP(J)+DSUM
      IF(DABS(TOP(J))-TOL) 52,52,53
52 TOP(J)=0.D0
53 J=J-1
      IF(J) 54,54,33
54 I=IND+L
      PIV(I)=-1.D0
      DO 55 I=1,L
      J=I+L
55 PIV(I)=-PIV(J)*REPI
      J=0
      DO 57 I=1,L
      IDO=J+IND
      SAVE=T(IDO)
      T(IDO)=0.D0
      DO 56 K=1,L
      ISE=K+J
56 T(ISE)=T(ISE)+SAVE*PIV(K)
57 J=J+L
      J=0
      K=0
      ISE=0
      IDO=0
      DO 61 I=1,L
      LL=I+L
      ILAB=IHE(LL)
      IF(IHE(I)-IPIV) 59,58,59
58 ISE=I
      J=ILAB
59 IF(ILAB-IND) 61,60,61
60 IDO=I
      K=IHE(I)
61 CONTINUE
      IF(K) 62,62,63
62 IHE(IDO)=IPIV
      IF(ISE) 67,67,65
63 IF(IND-J) 64,66,64
64 LL=L+L+L+NAN
      K=K+LL
      I=IPIV+LL
      ILAB=IHE(K)
      IHE(K)=IHE(I)
      IHE(I)=ILAB
      IF(ISE) 67,67,65
65 IDO=IDO+L
      I=ISE+L
      IHE(IDO)=J
      IHE(I)=IND
66 IHE(ISE)=0
67 LL=L+L
      J=LL+IND
      I=LL+L+IPIV
      ILAB=IHE(I)
      IHE(I)=IHE(J)
      IHE(J)=ILAB
      GO TO 8
68 IER=-1
69 SAVE=0.D0
      HELP=0.D0
      K=L+L+L
      DO 73 I=1,NAN
      IDO=K+I
      J=IHE(IDO)

```

```

      IF(J) 71,70,73
70  SAVE=--TOP(I)
71  IF(M+J+1) 73,72,73
72  HELP=TOP(I)
73  CONTINUE
      T(1)=SAVE
      IDO=NAN+1
      J=NAN+N
      DO 74 I=IDO,J
74  TOP(I)=SAVE
      DO 75 I=1,M
75  PIV(I)=HELP
      DO 79 I=1,NAN
      IDO=K+I
      J=IHE(IDO)
      IF(J) 76,79,77
76  J=-J
      PIV(J)=HELP-TOP(I)
      GO TO 79
77  IF(J-N) 78,78,79
78  J=J+NAN
      TOP(J)=SAVE+TOP(I)
79  CONTINUE
      DO 80 I=1,N
      IDO=NAN+I
80  TOP(I)=TOP(IDO)
81  RETURN
      END

```

C

ПідПРОГРАМА ФУНКЦІЙ РОЗКЛАДУ

```

SUBROUTINE FCT1(Y1,I,X1D,X2D,X3D,DE1D)
DIMENSION Y1(200),X1D(1000),X2D(1000),X3D(1000),X4D(1000),
*DE1D(1000),A(200)
DOUBLE PRECISION Y1,X1D,X2D,X3D,DE1D,X4D,A
N=850
M=35
AB=5
BB=5
N=N-BB+1
X4D(I)=DE1D(I)
IF(I-N)1,1,2
1 Y1( 1)=1
Y1( 2)=X1D(I)
Y1( 3)=X1D(I)*X1D(I)
Y1( 4)=X1D(I)*X1D(I)*X1D(I)
Y1( 5)=X2D(I)
Y1( 6)=X1D(I)*X2D(I)
Y1( 7)=X1D(I)*X1D(I)*X2D(I)
Y1( 8)=X2D(I)*X2D(I)
Y1( 9)=X1D(I)*X2D(I)*X2D(I)
Y1(10)=X2D(I)*X2D(I)*X2D(I)
Y1(11)=X3D(I)
Y1(12)=X1D(I)*X3D(I)
Y1(13)=X1D(I)*X1D(I)*X3D(I)
Y1(14)=X2D(I)*X3D(I)
Y1(15)=X1D(I)*X2D(I)*X3D(I)
Y1(16)=X2D(I)*X2D(I)*X3D(I)
Y1(17)=X3D(I)*X3D(I)
Y1(18)=X1D(I)*X3D(I)*X3D(I)
Y1(19)=X2D(I)*X3D(I)*X3D(I)
Y1(20)=X3D(I)*X3D(I)*X3D(I)
Y1(21)=X4D(I)
Y1(22)=X1D(I)*X4D(I)
Y1(23)=X1D(I)*X1D(I)*X4D(I)
Y1(24)=X2D(I)*X4D(I)

```

```

Y1(25)=X1D(I)*X2D(I)*X4D(I)
Y1(26)=X2D(I)*X2D(I)*X4D(I)
Y1(27)=X3D(I)*X4D(I)
Y1(28)=X1D(I)*X3D(I)*X4D(I)
Y1(29)=X2D(I)*X3D(I)*X4D(I)
Y1(30)=X3D(I)*X3D(I)*X4D(I)
Y1(31)=X4D(I)*X4D(I)
Y1(32)=X1D(I)*X4D(I)*X4D(I)
Y1(33)=X2D(I)*X4D(I)*X4D(I)
Y1(34)=X3D(I)*X4D(I)*X4D(I)
Y1(35)=X4D(I)*X4D(I)*X4D(I)
GOTO 7
C                                РЕГУЛЯРИЗАЦІЯ
2 K=I-N
  A(1)=1.D-1
  A(2)=1.D0
  A(3)=1.D1
  A(4)=1.D2
  A(5)=1.D0
  A(6)=1.D0
  A(7)=1.D0
  A(8)=1.D0
  A(9)=1.D0
  A(10)=1.D0
  DO 6 J=1,M
6 Y1(J)=0
  Y1(K)=A(K)
7 CONTINUE
  RETURN
  END

```

Додаток 3
MATLAB-програма макромодельовання лінійних систем
за частотними характеристиками.

```

function [num,den]=macmod(N,mag,phase,w);
%*****
% MACMOD - macromodel of continuous-time linear systems.      *
% [NUM,DEN]=MACMOD(N,MAG,PHASE,W) define N-order transfer func. *
% using frequency response MAG(W) and PHASE(W) of "Black Box". *
% NUM and DEN contain the polynomial coefficients of transfer *
% function in descending powers.                               *
% Linear system identification is executed in time range.      *
% Input and Output derivatives are evaluated using approx.    *
% in shifting polinomial method.                               *
% Composed be MATVICHUK Ja., Lviv State University, 06.93.    *
%*****
if nargin==3, w=phase; end;
m=max(size(w));
if m<100, error(' Not enough input data for identification'),end;
dw=(w(m)-w(1))/512;  wnew=[w(1):dw:w(m)];
%***** Linear interpolation for the new frequency points *****
mag=table1([w;mag'],'',wnew)';
if nargin>3, phase=table1([w;phase'],'',wnew)'; end;
w=wnew; clear wnew;  tmax=2*pi/dw;  m=512;
%***** Phase response is calculated using Hilbert transformation **
if nargin==3,
for i=1:2*m+2,if i<=m+1,a(i)=mag(i);else a(i)=mag(2*m-i+3);end;end;
phase=unwrap(imag(hilbert(log(a))));  phase=-phase(1:m+1);
clear a; plot(w/(2*pi),phase); end;
%***** Vector c(jw)=mag*cos(phase)+j*mag*sin(phase) *****
j=sqrt(-1);  c(1)=mag(1)*cos(phase(1));

```

```

for i=2:m; c(i)=mag(i)*cos(phase(i))+j*mag(i)*sin(phase(i));end;
c(m+1)=c(m);
for i=m+2:2*m; k=2*m+2-i;
c(i)=mag(k)*cos(phase(k))-j*mag(k)*sin(phase(k)); end;
clear mag phase;
%***** Time vector and input signal *****
tau=tmax/50; % tau is time constant of exponential input
ht=tmax/(2*m);
t=(0:ht:tmax); t=t(1:m);
for i=1:m, u(i)=0; end;
for i=m+1:2*m, u(i)=1-exp(-t(i-m)/tau); end;
u(m+1:2*m)=u(m+1:2*m)+mean(u(m+1:2*m))*1e-3*(rand(u(m+1:2*m))-0.5);
%***** Direct and back FT we evaluate the output signal *****
fu=fft(u); fy=fu.*c; clear c fu;
y=real(ifft(fy)); clear fy;
y=y(m+1:2*m); u=u(m+1:2*m);
%***** EVALUATION OF INPUT & OUTPUT DERIVATIONS *****
n=N+2;
A=zeros(2*N+1,m); B=zeros(1,m); p=zeros(1,n+1);
%***** Input Derivatives *****
n1=round(n/2); n2=round(n1/2); l=n+n1; l=n*2; l1=round(l/2);
for i=l1+1:n1:m-l1;
il=i-l1;
kt=t(il+1-l)-t(il); % aprox.interv.[t(il),t(il+1-l)]
tt=(t(il+1-l)-t(il))/kt; % into interval [0,1] (array tt).
p=polyfit(tt,u(il+1-l),n); % Aprox.by n-order pol.on l p-ts.
for j=-n2:n2; A(N+1,i+j)=polyval(p,tt(l1+1+j)); end;
for k=1:N; p=polyder(p); % Похідна k-го порядку.
for j=-n2:n2; A(N-k+1,i+j)=polyval(p,tt(l1+1+j))/kt^k; end;
end;
end;
%***** Output Derivatives *****
n1=round(n/2); n2=round(n1/2); l=n+n1; l=n*2; l1=round(l/2);
p=zeros(1,n+1); pd=zeros(1,n+1);
for i=l1+1:n1:m-l1;
il=i-l1;
kt=t(il+1-l)-t(il); % aprox.interv.[t(il),t(il+1-l)]
tt=(t(il+1-l)-t(il))/kt; % into interval [0,1] (array tt).
p=polyfit(tt,y(il+1-l),n); % Aprox.by n-order polin.on l p-ts
for j=-n2:n2; A(2*N+1,i+j)=polyval(p,tt(l1+1+j)); end;
for k=1:N-1; p=polyder(p); % k-order derivation.
for j=-n2:n2; A(2*N-k+1,i+j)=polyval(p,tt(l1+1+j))/kt^k; end;
end;
p=polyder(p); % N-order derivation.
for j=-n2:n2; B(i+j)=polyval(p,tt(l1+1+j))/kt^N; end;
end;
A=A(:,l1+1-n2:m-l1)'; B=B(l1+1-n2:m-l1)';
%***** Creating of Macromodel *****
K=(A\B)'; clear A B;
num=[K(1:N+1)]; % Numerator of transfer function
den=[1 K(N+2:2*N+1)]; % Denominator of transfer function
%***** Stability of obtained macromodel *****
roots_macr=roots(den);
for i=1:max(size(roots_macr)),
if real(roots_macr(i))>=0, if real(roots_macr(i))>0.1,
error('The macromodel is unstable, try to decrease the order');
end;end;end;

```

Додаток 4

FORTTRAN-програма апроксимації статичної характеристики тригера.

с Апроксимація статички тригера з регуляризациєю.


```

с Мінімізується середньоквадратична похибка по вектору Y
с Здійснюється регуляризація мінімізацією  $\min(|F*W-Y|^2 + \alpha * |W|^2)$ 
с Жорсткість регуляризації Більше, якщо  $\alpha$  Більше.
с Якщо  $\alpha = 0.0$ , то регуляризація відсутня.
      INTEGER B/1/,E,e1/2000/,M
      REAL*8 A,S,G,EPS/1.D-20/,W(30),F(20,2000),L(20,20),WK
      REAL T(2000),X1(2000),X(2000),Y(2000)
      REAL BW(30),v(30),c(60)
      write(*,*)' Введіть степінь апроксимуючого полінома(>2,<10):'
      read(*,*) M
      M=M+1
      alfa=0.0
      open (1, file='con')
      open (10, file='fileapr.san')
      open (2, file='rez1.dat')
      open (4, file='trigstat.mod')
      DO 119 I=1,30
119  W(I)=0.0
      read(10,'(I4)') ii
      do 1190 i=1,2
1190  read(10,'(A4)') E
      DO 120 E=1,e1
120  read(10,'(10E15.7)',end=110) T(E),Y(E),X(E)
110  E=E-1
      e1=e
      if(alfa.ne.0.0) e=e+m
с Регуляризаційна доБавка від e1+1 до e, всього m рівнянь.
      do 121 i=1,2
      write(i,*)' file beg.arr. end arr. nomb.coef.',
* ' EPS'
121  write(i,'(A13,3I10,E14.1)') name, B, E, M, EPS
      DO 122 I=B,E
122  CONTINUE
      if(alfa.eq.0.0) goto 126
      do 124 i=e1+1,e
124  y(i)=0.0
с Формування регуляризаційних констант.
      do 125 i=1,m
      c(i)=alfa
125  continue
126  continue
      BY=0
      DO 131 I=B,E
131  BY=BY+ABS(Y(I))
      BY=BY/(E-B+1)
      IF(BY.LE.1E-50) BY=1
      DO 132 I=B,E
      Y(I)=Y(I)/BY
      IF(ABS(Y(I)).LT.EPS) Y(I)=0.0
132  CONTINUE
      DO 101 I=1,M
      BW(I)=0
101  W(I)=0
      DO 1 I=1,M
      DO 2 J=1,M
      IF(I.GT.1) W(I)=1
      W(I)=1
      G=0
      DO 3 K=B,E1
      Z1=X(K)
      Z2=Z1*X(K)
      Z3=Z2*X(K)
      Z4=Z3*X(K)
      Z5=Z4*X(K)

```

```

      Z6=Z5*X(K)
      Z7=Z6*X(K)
      Z8=Z7*X(K)
      Z9=Z8*X(K)
      F(I,K)=W(1) +W(2)*Z1 +W(3)*Z2 +W(4)*Z3 +W(5)*Z4
      * +W(6)*Z5 +W(7)*Z6 +W(8)*Z7 +W(9)*Z8 +W(10)*Z9
3      G=G+DABS(F(I,K))
с      Регуляризаційна доБавка.
      if(alfa.eq.0.0) goto 333
      do 332 j=e1+1,e
332    f(i,j)=0.0
      f(i,e1+i)=alfa
333    continue
      BW(I)=1
1      CONTINUE
      DO 141 I=1,M
      DO 141 K=B,E1
      F(I,K)=F(I,K)/BW(I)
      IF(DABS(F(I,K)).LT.EPS) F(I,K)=0.0
141    CONTINUE
      DO 20 I=1,M
      L(I,I)=1
      I1=I-1
      IF(I1.LT.1) GOTO 41
      DO 4 J=1,I1
      IF(dabs(W(J)).LT.EPS) GO TO 5
      S=0
      DO 6 K=B,E
      G=F(I,K)*F(J,K)
6      S=S-G
      L(I,J)=S/W(J)
      GO TO 4
5      L(I,J)=0
4      CONTINUE
41     DO 8 K=B,E
      S=0
      DO 11 J=1,I
      G=L(I,J)*F(J,K)
11     S=S+G
8      F(I,K)=S
      I2=I-2
      IF(I2.LT.1) GOTO 12
      DO 9 J=1,I2
      S=0
      DO 7 K=J,I1
      G=L(I,K)*L(K,J)
7      S=S+G
9      L(I,J)=S
12     S=0
      DO 10 K=B,E
      G=F(I,K)*F(I,K)
10     S=S+G
20     W(I)=S
      DO 21 I=1,M
      IF(dabs(W(I)).LT.EPS) GO TO 22
      S=0
      DO 23 K=B,E
      G=Y(K)*F(I,K)
23     S=S+G
      W(I)=S/W(I)
      GO TO 21
22     W(I)=0
21     CONTINUE
      A=0

```

```

DO 25 K=B,E
S=0
DO 26 J=1,M
26 S=S+W(J)*F(J,K)
X1(K)=S*BY
25 A=A+(Y(K)-S)**2
A=DSQRT(A/(E-B+1))
DO 27 I=1,M
S=0
DO 28 K=I,M
G=W(K)*L(K,I)
28 S=S+G
27 W(I)=S
DO 30 I=1,M
30 W(I)=W(I)*BY/BW(I)
VE=0.0
Z=0.0
DO 31 I=B,E
Y(I)=Y(I)*BY
IF(ABS(Y(I)-X1(I)).GT.VE) VE=ABS(Y(I)-X1(I))
if(y(i).ne.0.0) VV=VE/ABS(Y(I))
31 IF(VV.GT.Z.AND.VV.LT.100) Z=VV
do 401 j=1,2
write(j,*)' '
write(j,*)' mid.qr.err. max.abs.err. max.rel.err.'
write(j,'(6D13.4)') A, VE, Z
write(j,*)' '
write(j,*)' Коефіцієнти апроксимації '
401 write(j,'(6D13.4)') (W(I),I=1,M)
do 501 i=1,m
501 v(i)=W(I)
C***** Завдання для SANOSa *****
write(4,'(A4)')'BEGI'
write(4,*)' Апроксимація статистики тригера.'
write(4,'(A4)')'DESC'
write(4,*)'*****;'
write(4,*)' * Схема тригера ;'
write(4,*)' R1(6,4)2.2K;R2(6,5)2.2K;R3(2,5)8.2K;R4(4,3)8.2K;'
write(4,*)' R5(2,7)3.9K; R6(1,0)620; R7(3,0)3.9K;'
write(4,*)' C1(2,5)1n; C2(4,3)1n; C3(1,0)10n;'
write(4,*)' E1(6,0)12;'
write(4,*)' E2(7,0) IF(T.LE.5) (6-T*2) (-4+(T-5)*2); U=E2;'
write(4,*)' P1(1,2,4)'МП21''; P2(1,3,5)'МП21'';'
write(4,*)' V2(R2); VK2(0,5);'
write(4,*)' * Апроксимація статистики ;'
write(4,*)' * кільк.точок =',el-b+1,' ;'
write(4,*)' f0=',v(1),'; f1=',v(2),'; f2=',v(3),';'
write(4,*)' f3=',v(4),'; f4=',v(5),'; f5=',v(6),';'
write(4,*)' f6=',v(7),'; f7=',v(8),'; f8=',v(9),';'
write(4,*)' f9=',v(10),';'
write(4,*)' YY=1/f1*(U-f0-Y*Y*(f2+Y*(f3+Y*(f4+Y*(f5+Y*(f6+'
write(4,*)' Y*(f7+Y*(f8+Y*f9))))))));'
write(4,*)' Y`=1e6*(YY-Y);'
write(4,*)' * Демонстрація апроксимаційної кривої ;'
write(4,*)' u=f0+y*(f1+y*(f2+y*(f3+y*(f4+y*(f5+y*(f6+y*(f7+'
write(4,*)' +y*(f8+y*f9))))))));'
write(4,*)' y=-T;'
write(4,*)'*****;;;'
write(4,'(A4)')'TRAN'
write(4,*)' TMAX=10; *ZERO; HS=4E-2; HMIN=1E-9; allstep;'
write(4,*)'OUTPUT: V2(E2),Y(U),y(u),XMIN=-5,XMAX=7,'
write(4,*)' YMIN=-10,YMAX=0;;'
write(4,'(A4)')'END '
END

```

Додаток 5

FORTRAN-програма сплайн-інтерполяції та обчислення похідних
для послідовності псевдовипадкових чисел.

```

с ПРОГРАМА ІНТЕРПОЛЯЦІЇ БАЗИСНИМИ В-СПЛАЙНАМИ К-ГО ПОР.ПО N ЗАДАНИМ
с ТОЧКАМ З АБСЦИСАМИ TAU(I) ТА ОРДИНАТАМИ GTAU(I); ОБЧИСЛЕННЯ
с ЗНАЧЕННЯ СПЛАЙНА ТА ЙОГО JDERIV-ї ПОХІДНОЇ В ТОЧЦІ X (K<21).
с (ПІДПРОГРАМИ SPLINT, SPLOPT, BSPLVB, BANFAC, BANSLV, BVALUE).
с ДЕТАЛЬНО ДИВИСЬ: К.ДЕ БОР, "ПРАКТИЧЕСКОЕ РУКОВОДСТВО ПО СПЛАЙНАМ"
с М., РАДИО И СВЯЗЬ, 1985. НАБРАНО ТА ВІДЛАГОДЖЕНО 09.1992
с доц. Матвійчук Я.М., кафедра ТОЕР Львівського університету
с SUBROUTINE SPLINT (TAU, GTAU, T, N, K, BCOEF, IFLAG)
с** ОБЧИСЛЕННЯ ІНТЕРПОЛЯЦІЙНИХ В-СПЛАЙНІВ К-ГО ПОРЯДКУ З ВУЗЛАМИ **
с***** В T(I). КОЕФІЦІЄНТИ ІНТЕРПОЛЯЦІЇ В BCOEF.*****
      INTEGER IFLAG, K, N, I, ILP1MX, J, JJ, KM1, KPKM2, LEFT, LENQ, NP1
      REAL BCOEF(N), GTAU(N), Q(21000), T(N+K), TAU(N), TAU1
с      DIMENSION Q((2*K-1)*N), T(N+K), BCOEF(N)
      IF(K.GT.20) GOTO 999
      NP1=N+1
      KM1=K-1
      KPKM2=2*KM1
      LEFT=K
      LENQ=N*(K+KM1)
      DO 5 I=1, LENQ
5      Q(I)=0
      DO 30 I=1, N
          TAU1=TAU(I)
          ILP1MX=MIN0(I+K, NP1)
          LEFT=MAX0(LEFT, I)
          IF(TAU1.LT.T(LEFT)) GOTO 998
15      IF(TAU1.LT.T(LEFT+1)) GOTO 16
          LEFT=LEFT+1
          IF(LEFT.LT.ILP1MX) GOTO 15
          LEFT=LEFT-1
          IF(TAU1.GT.T(LEFT+1)) GOTO 998
16      CALL BSPLVB(T, K, 1, TAU1, LEFT, BCOEF)
          JJ=I-LEFT+1+(LEFT-K)*(K+KM1)
          DO 30 J=1, K
              JJ=JJ+KPKM2
30      Q(JJ)=BCOEF(J)
          CALL BANFAC(Q, K+KM1, N, KM1, KM1, IFLAG)
          GOTO (40, 999), IFLAG
40      DO 41 I=1, N
41      BCOEF(I)=GTAU(I)
          CALL BANSLV(Q, K+KM1, N, KM1, KM1, BCOEF)
          RETURN
998      IFLAG=2
999      WRITE (*, *) ' ЛІНІЙНА СИСТЕМА НЕ ОВЕРТАЄТЬСЯ '
          RETURN
      END

      SUBROUTINE BSPLVB ( T, JHIGH, INDEX, X, LEFT, BIATX )
с***** ОБЧИСЛЕННЯ В-СПЛАЙНІВ *****
      INTEGER INDEX, JHIGH, LEFT, I, J, J1, JP1
      REAL BIATX(JHIGH), T(1), X, DELTAL(21), DELTAR(21), SAVED, TERM
      GOTO (10, 20), INDEX
10      J=1
      BIATX(1)=1
      IF(J.GE.JHIGH) GOTO 99
20      JP1=J+1
      DELTAR(J)=T(LEFT+J)-X

```

```

      DELTAL(J)=X-T(LEFT+1-J)
      SAVED=0.0
      DO 26 I=1,J
      TERM=BIATX(I)/(DELTAR(I)+DELTAL(JP1-I))
      BIATX(I)=SAVED+DELTAR(I)*TERM
26      SAVED=DELTAL(JP1-I)*TERM
      BIATX(JP1)=SAVED
      J=JP1
      IF(J.LT.JHIGH) GOTO 20
99      RETURN
      END

      SUBROUTINE SPLOPT (TAU, N, K, T, IFLAG)
C***** ОПТИМАЛЬНЕ РОЗМІЩЕННЯ ВУЗЛІВ ІНТЕРПОЛЯЦІЇ *****
      INTEGER IFLAG,K,N,I,ID,INDEX,J,KM1,KPK,KPKM1,KPN,KP1,L,LEFT,
      *      LEFTMK,LENW,LL,LLMAX,LLMIN,NA,NB,NC,ND,NEWTMX/10/,
      *      NEWTON,NMK,NMKM1,NX
      REAL SCR(23000),T(1),TAU(N),DEL,DELMAX,FLDATK,SIGN,SIGNST,
      *      SUM,TOL,TOLRTE/0.000001/,XIJ
C      ПОЗМІП SCR=(N-K)(2K+3)+5K+3
      IF(K.GT.20) GOTO 999
      NMK=N-K
      IF(NMK) 1,56,2
1      WRITE(*,*)' ЧИСЛО ТОЧОК',N,' < ПОРЯДКА ІНТЕРПОЛЯЦІЇ',K
      GOTO 999
2      IF(K.GT.2) GOTO 3
      WRITE(*,*)' ПОРЯДОК ІНТЕРПОЛЯЦІЇ',K,' < 3'      GOTO 999
3      NMKM1=NMK-1
      FLOATK=K
      KPK=K+K
      KP1=K+1
      KM1=K-1
      KPKM1=KPK-1
      KPN=K+N
      SIGNST=-1.0
      IF(NMK.GT.(NMK/2)*2) SIGNST=1.0      NX=N+KPK+1      NA=NX+NMK+1
      ND=NA+N
      NB=ND+NMK
      NC=NB+KP1
      LENW=KPKM1*NMK
      DO 5 J=1,K
      SCR(J)=TAU(1)
5      SCR(KPN+J)=TAU(N)
      DO 6 J=1,N
6      SCR(K+J)=TAU(J)
      SCR(NX)=TAU(1)
      SCR(NMK+1+NX)=TAU(N)
      DO 10 J=1,NMK
      SUM=0.0
      DO 9 L=1,KM1
9      SUM=SUM+TAU(J+L)
10     SCR(J+NX)=SUM/FLOAT(KM1)
      SCR(N+NA)=0.5
      NEWTON=1
      TOL=TOLRTE*(TAU(N)-TAU(1))/FLOAT(NMK)
20     DO 21 I=1,LENW
21     SCR(I+NC)=0.0
      DO 22 I=2,N
22     SCR(I-1+NA)=0.0
      SIGN=SIGNST
      LEFT=KP1
      DO 28 J=1,NMK
      XIJ=SCR(J+NX)
23     IF(XIJ.LT.SCR(LEFT+1)) GOTO 25

```

```

LEFT=LEFT+1
IF (LEFT.LT.KPN) GOTO 23
LEFT=LEFT-1
25 CALL BSPLVB (SCR,K,1,XIJ,LEFT,SCR(1+NB))
LEFTMK=LEFT-K
INDEX=LEFTMK-J+(J-1)*KPKM1+NC
LLMIN=MAX0(1,K-LEFTMK)
LLMAX=MIN0(K,N-LEFTMK)
DO 26 LL=LLMIN,LLMAX
26 SCR(LL+INDEX)=SCR(LL+NB)
CALL BSPLVB (SCR,KP1,2,XIJ,LEFT,SCR(1+NB))
ID=MAX0(0,LEFTMK-KP1)
LLMIN=1-MIN0(0,LEFTMK-KP1)
DO 27 LL=LLMIN,KP1
ID=ID+1
27 SCR(ID+NA)=SCR(ID+NA)-SIGN*SCR(LL+NB)
28 SIGN=-SIGN
CALL BANFAC (SCR(1+NC),KPKM1,NMK,KM1,KM1,IFLAG)
GOTO(45,44) IFLAG
44 WRITE(*,*) ' В SPLOPT НЕ ОУЕПТАЕТСЯ МАТРИЦА '
RETURN
45 DO 46 I=N,2,-1
46 SCR(I-1+NA)=SCR(I-1+NA)+SCR(I+NA)
DO 49 I=1,NMK
49 SCR(I+ND)=SCR(I+NA)*(TAU(I+K)-TAU(I))/FLOATK
CALL BANSLV (SCR(1+NC),KPKM1,NMK,KM1,KM1,SCR(1+ND))
DELMAX=0.0
SIGN=SIGNST
DO 53 I=1,NMK
DEL=SIGN*SCR(I+ND)
DELMAX=AMAX1(DELMAX,ABS(DEL))
IF(DEL.GT.0) GOTO 51
DEL=AMAX1(DEL,(SCR(I-1+NX)-SCR(I+NX))/3.0)
GOTO 52
51 DEL=AMIN1(DEL,(SCR(I+1+NX)-SCR(I+NX))/3.0)
52 SIGN=-SIGN
53 SCR(I+NX)=SCR(I+NX)+DEL
IF(DELMAX.LT.TOL) GOTO 54
NEWTON=NEWTON+1
IF(NEWTON.LE.NEWTMX) GOTO 20
WRITE(*,*) ' НЕ ЗБІГАЄТЬСЯ МЕТОД НЬЮТОНА В SPLOPT. ЧЕКАЙТЕ!'
54 DO 55 I=1,NMK
55 T(K+I)=SCR(I+NX)
56 DO 57 I=1,K
T(I)=TAU(1)
57 T(N+I)=TAU(N)
RETURN
999 IFLAG=2
RETURN
END

SUBROUTINE BANFAC ( W, NROWW, NROW, NBANDL, NBANDU, IFLAG )
C***** LU-ПОЗКЛАД МАТРИЦІ А.*****
INTEGER IFLAG,NBANDL,NBANDU,NROW,NROWW,I,IPK,J,JMAX,K,KMAX,
* MIDDLE,MIDMK,NROWM1
REAL W(NROWW,NROW), FACTOR,PIVOT
IFLAG=1
MIDDLE=NBANDU+1
NROWM1=NROW-1
IF(NROWM1) 999,900,1
1 IF(NBANDL.GT.0) GOTO 10
DO 5 I=1,NROWM1
IF(W(MIDDLE,I).EQ.0) GOTO 999
5 CONTINUE

```

```

      GOTO 900
10    IF (NBANDU.GT.0) GOTO 20
      DO 15 I=1,NROWM1
      PIVOT=W(MIDDLE,I)
      IF (PIVOT.EQ.0.0) GOTO 999
      JMAX=MIN0 (NBANDL,NROW-I)
      DO 15 J=1,JMAX
15    W(MIDDLE+J,I)=W(MIDDLE+J,I)/PIVOT
      GOTO 900
20    DO 50 I=1,NROWM1
      PIVOT=W(MIDDLE,I)
      IF (PIVOT.EQ.0.0) GOTO 999
      JMAX=MIN0 (NBANDL,NROW-I)
      DO 32 J=1,JMAX
32    W(MIDDLE+J,I)=W(MIDDLE+J,I)/PIVOT
      KMAX=MIN0 (NBANDU,NROW-I)
      DO 40 K=1,KMAX
      IPK=I+K
      MIDMK=MIDDLE-K
      FACTOR=W(MIDMK,IPK)
      DO 40 J=1,JMAX
40    W(MIDMK+J,IPK)=W(MIDMK+J,IPK)-W(MIDDLE+J,I)*FACTOR
50    CONTINUE
900    IF (W(MIDDLE,NROW).NE.0)
      RETURN
999    IFLAG=2
      RETURN
      END

      SUBROUTINE BANSIV ( W, NROWW, NROW, NBANDL, NBANDU, B )
C***** ПОЗВ'ЯЗОК ЛІНІЙНОЇ СИСТЕМИ *****
      INTEGER NBANDL,NBANDU,NROW,NROWW,I,J,JMAX,MIDDLE,NROWM1
      REAL W(NROWW,NROW),B(NROW)
      MIDDLE=NBANDU+1      IF (NROW.EQ.1) GOTO 49
      NROWM1=NROW-1
      IF (NBANDL.EQ.0) GOTO 30
      DO 21 I=1,NROWM1
      JMAX=MIN0 (NBANDL,NROW-I)
      DO 21 J=1,JMAX
21    B(I+J)=B(I+J)-B(I)*W(MIDDLE+J,I)
30    IF (NBANDU.GT.0)
      GOTO 40
      DO 31 I=1,NROW
31    B(I)=B(I)/W(1,I)
      RETURN
40    DO 45 I=NROW,2,-1
      B(I)=B(I)/W(MIDDLE,I)
      JMAX=MIN0 (NBANDU,I-1)
      DO 45 J=1,JMAX
45    B(I-J)=B(I-J)-B(I)*W(MIDDLE-J,I)
49    B(1)=B(1)/W(MIDDLE,1)
      RETURN
      END

      REAL FUNCTION BVALUE ( T, BCOEF, N, K, X, JDERIV )
C***** ОБЧИСЛЕННЯ ЗНАЧЕННЯ JDERIV-ї ПОХІДНОЇ СПЛАЙНУ В ТОЧЦІ X **
      INTEGER JDERIV,K,N,I,ILO,IMK,J,JC,JCMIN,JCMAX,JJ,KMJ,KM1,NMI
      REAL BCOEF(N),T(1),X,AJ(21),DL(21),DR(21),FKMJ
      BVALUE=0
      IF (K.GT.20) GOTO 99
      IF (JDERIV.GE.K) GOTO 99
      IF (X.GE.T(1).AND.X.LE.T(N+K)) GOTO 102
      WRITE(*,*) ' Шукане значення сплайну за межами області '
      WRITE(*,*) ' визначення: ',T(1),'-',T(N+K)          GOTO 99

```

```

102 DO 100 I=1,N+K-1
100 IF(T(I).LE.X.AND.
* (T(I+1).GT.X.OR.(T(I+1).GE.X.AND.T(I+1).EQ.T(N+K))))GOTO 101
101 CONTINUE
    KM1=K-1
    IF(KM1.GT.0) GOTO 1
    BVALUE=BCOEF(I) GOTO 99
1    JCMIN=1
    IMK=I-K
    IF(IMK.GE.0) GOTO 8
    JCMIN=1-IMK
    DO 5 J=1,I
5    DL(J)=X-T(I+1-J)
    DO 6 J=I,KM1
    AJ(K-J)=0
6    DL(J)=DL(I)
    GOTO 10
8    DO 9 J=1,KM1
9    DL(J)=X-T(I+1-J)
10   JCMAX=K
    NMI=N-I
    IF(NMI.GE.0) GOTO 18
    JCMAX=K+NMI
    DO 15 J=1,JCMAX
15   DR(J)=T(I+J)-X
    DO 16 J=JCMAX,KM1
    AJ(J+1)=0
16   DR(J)=DR(JCMAX)
    GOTO 20
18   DO 19 J=1,KM1
19   DR(J)=T(I+J)-X
20   DO 21 JC=JCMIN,JCMAX
21   AJ(JC)=BCOEF(IMK+JC)
    IF(JDERIV.EQ.0) GOTO 30
    DO 23 J=1,JDERIV
    KMJ=K-J
    FKMJ=FLOAT(KMJ)
    ILO=KMJ
    DO 23 JJ=1,KMJ
    AJ(JJ)=(AJ(JJ+1)-AJ(JJ))/(DL(ILO)+DR(JJ))*FKMJ
23   ILO=ILO-1
30   IF(JDERIV.EQ.KM1) GOTO 39
    DO 33 J=JDERIV+1,KM1
    KMJ=K-J
    ILO=KMJ
    DO 33 JJ=1,KMJ
    AJ(JJ)=(AJ(JJ+1)*DL(ILO)+AJ(JJ)*DR(JJ))/(DL(ILO)+DR(JJ))
33   ILO=ILO-1
39   BVALUE=AJ(1)
99   RETURN
    END

```

```

SUBROUTINE FRAND(RAND)
C*** ГЕНЕРАТОР ПСЕВДОВИПАДКОВИХ ЧИСЕЛ З РІВНОМІРН.РОЗПОД. У [0,1]
    INTEGER IY/1234567/
    REAL RAND, S/0.4656613E-9/
    INTEGER A/843314861/,C/453816693/,M/1073741824/,O/1073741824/
    IY=IY*A+C
    IF(IY.LT.0) IY=IY+M+O
    RAND=FLOAT(IY)*S
    RETURN
    END

```

integer e, em, ps


```

real t(10000),x(10000),x1(10000),x2(10000),y(10000)
real tt(1000),yy(1000),u(1020),bcoef(1000)
write(*,*)' Програма інтерполяції послідовності випадкових'
write(*,*)' чисел, обчислення похідних до третього порядку'
с е - кількість точок( 4<e<801 ), ps - порядок сплайну( 2<ps<9 )
write(*,*)' Між випадковими точками обчислюються 9 '
write(*,*)' проміжних по сплайну.'
write(*,*)'Результати записуються в файли randn0,..,randn3.'
write(*,*)' Введіть 2 цілих числа:'
write(*,*)' кільк.точок(>4,<801); порядок сплайну(>2,<9) '
read(*,*) e,ps
do 601 i=1,e
call frand(yy(i))
601 tt(i)=i
call splopt(tt,e,ps,u,iflag)
call splint(tt,yy,u,e,ps,bcoef,iflag)
em=(e-1)*10+1
do 609 i=1,em
t(i)=1.0+(i-1)*0.1
x(i)=bvalue(u,bcoef,e,ps,t(i),0)
x1(i)=bvalue(u,bcoef,e,ps,t(i),1)
x2(i)=bvalue(u,bcoef,e,ps,t(i),2)
y(i)=bvalue(u,bcoef,e,ps,t(i),3)
609 continue
j=0
do 610
i=10,em-10
j=j+1
t(j)=t(i)
x(j)=x(i)
x1(j)=x1(i)
x2(j)=x2(i)
610 y(j)=y(i)
em=j
open(1,file='randr0.')
open(2,file='randr1.')
open(3,file='randr2.')
open(4,file='randr3.')
write(1,'(1000E11.4)') (x(i),i=1,em)
write(2,'(1000E11.4)') (x1(i),i=1,em)
write(3,'(1000E11.4)') (x2(i),i=1,em)
write(4,'(1000E11.4)') (y(i),i=1,em)
END

```

Додаток 6

FORTRAN-програма апроксимації
з редукцією апроксимуючого поліному.

```

с***** ПРОГРАМА АПРОКСИМАЦІЇ МЕТОДОМ НАЙМЕНШИХ КВАДРАТІВ *****
с***** З РЕДУКЦІЄЮ АПРОКСИМУЮЧОГО ПОЛІНОМА *****

SUBROUTINE APR3(R,W,Z1,Z2,Z3)
real*8 R,W(40)
real z1,z2,z3
R=W(1)+W(2)*Z1+W(3)*Z2+W(4)*Z3+W(5)*Z1*Z1+W(6)*Z2*Z2
* +W(7)*Z3*Z3+W(8)*Z1*Z2+W(9)*Z1*Z3+W(10)*Z2*Z3+W(11)*Z1**3
* +W(12)*Z2**3+W(13)*Z3**3+W(14)*Z1*Z1*Z2+W(15)*Z1*Z1*Z3
* +W(16)*Z2*Z2*Z1+W(17)*Z2*Z2*Z3+W(18)*Z3*Z3*Z1+W(19)*Z3*Z3*Z2
* +W(20)*Z1*Z2*Z3
end

```

```

      SUBROUTINE APPROX(b,e,m,w,sh,a,q,z)
      INTEGER B,E,M
      REAL*8 A,S,G,EPS/1.D-17/,W(1),F(36,1000),L(40,40)
      REAL BW(40),sh(1)
      COMMON /ARRAY/ Y(1000),X(1000),X1(1000),X2(1000),ya(1000)
      BY=0
      DO 131 I=B,E
131  BY=BY+ABS(Y(I))
      BY=BY/(E-B+1)
      DO 132 I=B,E
132  Y(I)=Y(I)/BY
      DO 1 I=1,M
        DO 2 J=1,M
          2  W(J)=0
          W(I)=sh(i)
          G=0
          DO 3 K=B,E
            call apr3(a,w,X(K),X1(K),X2(K))
            F(I,K)=a
          3  G=G+DABS(F(I,K))
          BW(I)=G/(E-B+1)
          1  CONTINUE
      DO 141 I=1,M
      DO 141 K=B,E
      F(I,K)=F(I,K)/BW(I)
      IF(DABS(F(I,K)).LT.EPS) F(I,K)=0.0
141  CONTINUE
      DO 20 I=1,M
      L(I,I)=1
      I1=I-1
      IF(I1.LT.1) GOTO 41
      DO 4 J=1,I1
      IF(W(J).LT.EPS) GO TO 5
      S=0
      DO 6 K=B,E
      G=F(I,K)*F(J,K)
          6  S=S-G
      L(I,J)=S/W(J)
      GO TO 4
          5  L(I,J)=0
          4  CONTINUE
41  DO 8 K=B,E
      S=0
      DO 11 J=1,I
      G=L(I,J)*F(J,K)
          11  S=S+G
          8  F(I,K)=S
      I2=I-2
      IF(I2.LT.1) GOTO 12
      DO 9 J=1,I2
      S=0
      DO 7 K=J,I1
      G=L(I,K)*L(K,J)
          7  S=S+G
          9  L(I,J)=S
          12  S=0
      DO 10 K=B,E
      G=F(I,K)*F(I,K)
          10  S=S+G
          20  W(I)=S
      DO 21 I=1,M
      IF(W(I).LT.EPS) GO TO 22
      S=0
      DO 23 K=B,E

```

```

      G=Y(K)*F(I,K)
23      S=S+G
      W(I)=S/W(I)
      GO TO 21
22      W(I)=0
21      CONTINUE
      A=0
      DO 25 K=B,E
      S=0
      DO 26 J=1,M
26      S=S+W(J)*F(J,K)
      ya(K)=S*BY
25      A=A+(Y(K)-S)**2
      A=DSQRT(A/(E-B+1))
      DO 27 I=1,M
      S=0
      DO 28 K=I,M
      G=W(K)*L(K,I)
28      S=S+G
27      W(I)=S
      DO 30 I=1,M
30      W(I)=W(I)*BY/BW(I)
      q=0.0
      Z=0.0
      DO 31 I=B,E
      Y(I)=Y(I)*BY
      IF (ABS(Y(I)-X1(I)).GT.q) q=ABS(Y(I)-ya(I))
      VV=q/ABS(Y(I))
31      IF (VV.GT.Z.AND.VV.LT.100) Z=VV
      END

      INTEGER B/1/,E,M,em,e1
      REAL*8 A,S,G,EPS/1.D-17/,W(40)
      REAL T(1000),BW(40),sh(40),sh1(40),v(40),v1(40),v2(40),crit
      COMMON /ARRAY/ Y(1000),X(1000),X1(1000),X2(1000),ya(1000)
      open (10, file='fileapr.san')
      open (5, file='rez.dat')
      write(*,*) 'Введіть: кільк.точок апрокс.та кільк.коэф-ів'
      read(*,*) em,m
      e=1
120      continue
      read(10,'(15E15.7)',end=110)T(e),X(e),X1(e),X2(e),Y(e)
      e=e+1
      if(e.le.em) goto 120
110      e=e-1
      do 120 i=1,em
      t(i)=i
120      continue
      do 130 i=1,m
130      sh(i)=1.0
      call approx(b,e,m,w,sh,a,q,z)
      write(5,*) ' сер.-кв.пох. макс.аБс.пох. макс.відн.пох.'
      write(5,'(3E13.4)') A, q, Z
      write(5,*) ' Коефіцієнти апроксимації '
      write(5,'(6D13.4)') (W(I),I=1,M)
      do 151 i=1,m
151      v1(i)=w(i)
      e1=e
      e=e-int(e/5)
      call approx(b,e,m,w,sh,a,q,z)
      write(5,*) ' сер.-кв.пох. макс.аБс.пох. макс.відн.пох.'
      write(5,'(3E13.4)') A, q, Z
      write(5,*) ' Коефіцієнти апроксимації '
      write(5,'(6D13.4)') (W(I),I=1,M)

```

```

do 152 i=1,m
152  v2(i)=w(i)
      ddmin=1E22
      ddmax=0.0
do 153 i=1,m
      dd=abs((v1(i)-v2(i))/v1(i))
      if(dd.lt.ddmin) ddmin=dd
      if(dd.gt.ddmax) ddmax=dd
153  sh1(i)=dd
600  write(*,*) ' ПохиБки коефіцієнтів апроксимації '
      write(1,'(6E13.4)') (sh1(i),i=1,m)
      dd=(ddmax+ddmin)/5
      write(*,*) ' Рекомендований критерій редукції =',dd
      write(*,*) ' Введіть критерій редукції ( 0 - кінець ) '
      read(*,*) dd
      if(dd.eq.0.0) goto 601
      do 154 i=1,m
      if(sh1(i).gt.dd) then
        sh(i)=0.0
      else
        sh(i)=1.0
      end if
154  continue
      e=e1
      call approx(b,e,m,w,sh,a,q,z)
      write(5,*) ' сер.-кв.пох. макс.аБс.пох. макс.відн.пох.'
      write(5,'(3E13.4)') A, q, Z
      write(5,*) ' Коефіцієнти апроксимації '
      write(5,'(6D13.4)') (W(I),I=1,M)
do 155 i=1,m
155  v(i)=w(i)
      goto 600
601  do 412 j=B,E,8
      write(5,'(8E9.3)') (Y(I),I=j,j+7)
412  write(5,'(8E9.3)') (ya(I),I=j,j+7)
      END

```

Додаток 7

PASCAL-програма обрахунку кореляційної розмірності та розмірності вкладення по дискретним послідовностям.

```

{$N+}
Program Fractdim;
{ ОБчислення розмірності вкладення для дискретної послідовності
  значень вихідного сигналу y[i] через кореляційну розмірність d2
  для ряду значень розмірності простору nr. (1.01.93).
  Реалізовано випадковий виБір точок. }
Uses Crt, Graph ;
type
      ar = Array[1..9000] of Single;
      ai = Array[1..9000] of Longint;
Var      finput,fres,fresl : Text;
          st : string[23];
          y : ^ar;
          yi : ^ai;
          z : Array[1..13] of Char;
          z1 : Array[1..11] of Char;
          ne : Array[1..40,1..40] of Longint;
          a,d2apr,lneps,eps,c : Array[0..40] of Real;
          lnci : Array[1..40,1..40] of Real;
          ei : Array[0..40] of Longint;

```

```

    nt,nrmin,nrmax,i,j,k,nr,nem,nk : Integer;
    ant,gd,gm,yy,color,pole,Gmx,Gmy : Integer;
    ntmax,nmmax,n,n1,n2,n0,m,nm,di,dd : Longint;
    lne,lnc,slne,slnc,slnce,slne2,si : Real;
    kx,ky,maxx,maxy,minx,miny : Real;
    rnd,dt,dlnc,dlne,k0,kr,d,d2,km : Real;
    Label l1,l2,l3,l4,l1,l3l,l5,l1l,l;
Begin
{ Початкові дані *****}
{ Крок по даним : } nk:=1;
{ Мінімальна розмірність 1<=nrmin<20 : } nrmin:=5;
{ Максимальна розмірність 1<nrmax<=20 : } nrmax:=9;
{ Кратність зменшення критичної відстані : } kr:=1.07;
{ Перше значення критичної відстані
  в %% від макс.модуля d даних : } k0:=100;
{ Максимальна кількість точок
  на кривій ln(c) (ln(l/lo)) ntmax<=40 : } ntmax:=40;
{ Довжина вихідного масиву nmmax : } nmmax:=2500;
{ Радіус околу обраної випадкової точки : } nt:=400;
{ Кількість випадкових точок : } ant:=500;
{ *****}
  New(y); New(yi);
{ Відкривання файлів зчитування finput та запису fres,fresl }
  Assign(finput,'chua. '); Reset(finput);
  Assign(fresl,'reschua.pas');
{ Читання дискретних значень вихідної змінної y[i] }
  nm:=0; d:=0.0;
  while nm<nmmax do
  Begin
    Read(finput,z[1]); Read(finput,z[1]); If z[1]=#10 then goto l1;
    For j:=1 to 13 do Read(finput,z[j]);
    nm:=nm+1; Val(z,y^[nm],k);
    if abs(y^[nm])>d then d:=abs(y^[nm]);
    For j:=1 to 17 do Read(finput,z[j]);
  End;
l1: Close(finput);
{ Формування цілочисельного масиву вихідної змінної yi[i] }
  km:=10000/d; { Коефіцієнт перерахунку }
  For i:=1 to nm do begin yi[i]:=Round(y^[i]*km); end;
{ Формування масиву критичних значень eps[i] та його
  цілочисельного еквіваленту ei[i] }
  k0:=k0/100; eps[0]:=sqr(d*k0*kr); m:=0; nem:=0;
  for i:=1 to ntmax do
  begin
    for j:=1 to ntmax do ne[j,i]:=0;
    eps[i]:=eps[i-1]/(kr*kr); ei[i]:=Round(eps[i]*km*km);
    if ei[i]>0 then nem:=i; write(ei[i],' ');
  end;
  if nem=0 then goto l2;
{ Підрахунок пар точок в просторі розмірністю nr+1, відстань між
  якими di не більша за ei[i]. Запис результатів у ne[nr,i]. }
  n0:=0;
  While n0<ant do Begin {n1-перша координата i-ї точки}
    n1:=nrmax+1+round(random*(nmmax-1-nrmax));
    n0:=n0+1;
    n2:=n1-nt; if n2<nrmax then n2:=nrmax;
    While n2<=n1+nt do Begin {n2-перша коорд.ii-ї точки}
      if n2=n1 then goto l;
      di:=0;
      for i:=0 to nrmin-2 do di:=di+sqr(yi^[n1-i]-yi^[n2-i]);
    { Цикл по розмірності простору nr }
    For nr:=nrmin to nrmax do Begin
      di:=di+sqr(yi^[n1-nr+1]-yi^[n2-nr+1]);
      For i:=1 to nem do

```



```

    pole:=25;
    kx:=(GetMaxX-pole*2)/(maxx-minx);
    ky:=(GetMaxY-pole*2)/(maxy-miny);
{ Вивід графіків та супутньої інформації }
    SetTextStyle(0,0,1); {тип Букв : шрифт,положення,розмір}
    SetColor(White);
    OutTextXY(80,30,'мірність простору');
    OutTextXY(240,30,'кореляційна розмірність');
{ Осі координат, назви змінних та оцифровки }
    Gmx:=(GetMaxX-pole); Gmy:=(GetMaxY-pole);
    Line(pole,pole,pole,Gmy+3); Line(pole-3,Gmy,Gmx,Gmy);
    Line(pole-3,pole,pole,pole); Line(Gmx,Gmy,Gmx,Gmy+3);
    OutTextXY(Gmx-150,Gmy+10,'Ln(1/lo)');
    SetTextStyle(0,1,1);
    OutTextXY(pole+15,pole,'Ln(c)');
    str(maxy,st); delete(st,6,12); delete(st,8,2);
    OutTextXY(15,pole-10,st);
    str(miny,st); delete(st,6,12); delete(st,8,2);
    OutTextXY(15,Gmy-60,st);
    SetTextStyle(0,0,1);
    str(minx,st); delete(st,6,12); delete(st,8,2);
    OutTextXY(pole-10,Gmy+10,st);
    str(maxx,st); delete(st,6,12); delete(st,8,2);
    OutTextXY(Gmx-60,Gmy+10,st);
{ Побудова графіків }
    color:=16;
    for nr:=nrmin to nrmax do begin
        color:=color-1; SetColor(color);
        for i:=1 to nem-1 do begin
            if ne[nr,i+1]=0 then goto l5;
{ Ділянка лінійно-інтерпольованої залежності
                ln(c[i]) від ln(eps[i]/eps[1]),i=1,...,nem }
                Line(round((lneps[i]-minx)*kx+pole),
                    Gmy-round((lnci[nr,i]-miny)*ky),
                    round((lneps[i+1]-minx)*kx+pole),
                    Gmy-round((lnci[nr,i+1]-miny)*ky));
                end;
l5:    if ne[nr,2]<>0 then begin
{ Графік середньо-квадратичної апроксимації }
                yy:=Gmy-round((d2apr[nr]*lneps[1]+a[nr]-miny)*ky);
                Line(round((lneps[1]-minx)*kx+pole),
                    yy,
                    round(Gmx-(Gmy-yy)/ky/d2apr[nr]*kx),
                    Gmy);
{ Значення мірності простору nr та кореляційної розмірності d2apr }
                str(nr,st); OutTextXY(145,45+12*(nr-nrmin),st);
                OutTextXY(Gmx+3,yy,st);
                str(d2apr[nr],st); delete(st,6,12); delete(st,8,2);
                OutTextXY(288,45+12*(nr-nrmin),st);
                end; end;

    ReadLn;
    CloseGraph;
End.

```

Додаток 8

FORTRAN-програма автоматичного розбиття на підобласті у просторі аргументів та регуляризованої апроксимації у межах кожної підобласті.

```

    INTEGER B/1/,E,M/15/,MT(300),BLAN/'      '/,MB(300),NOMM
    INTEGER APR1/'APR1'/,APR2/'APR2'/,APR3/'APR3'/,NOMT,
    *   APR4/'APR4'/,APR5/'APR5'/,APR6/'APR6'/,APR7/'APR7'/,

```

```

* CH1(410),CH(410),KNS,PR,MM(10)/15,15,15,15,15,15,11,31,31,21/
  INTEGER NOMM1/'FS41'/,NOMM2/'4' '/',  NOMT1/'TS41'/,
*  NOMT2/'4' '/',APR8/'APR8'/,APR9/'APR9'/,AP10/'AP10'/
*  ,AP11/'AP11'/,AP12/'AP12'/,      LR(15),      MQ/ 4/
REAL HMAX/3E-4/,DHM/1E-4/,OTN/1E20/,DOTN/999.9/,PREG/1E-3/
INTEGER NS/4/,DNS/6/,ECR/1/,KONN/101/,KOM/202/,NAP/1/,REG/0/
REAL MXCT(37,10)/370*0.0/,MYCT(10)/10*0.0/,RX(37,10),RY(10)
REAL*16 A,S,      G,EPS/1.Q-34/,W(200),F(37,500),L(37,37)
REAL*16 SMR
REAL B1(808),B2(808),B3(808),B4(808), Y(500),U(500), C(15),
*  XMIN(37),XMAX(37), W4(600),RA(37), RAXY(10),SQRADX(10),
*  Z1IN/0.0/,Z/0.0/,Z1/0.0/,V/0.0/,      BY, KY,MKS(11),
*  KKK(37), XC(37), XCT(37), YY(500), X(37,500),KKY
  INTEGER*2 AM(990)
C  NS-кількість оБластей,
C  DNS-число точок перекривання,
C  ECR=0-немає запису до Бібліотеки,
C  KONN-число точок, з яких Будуються підоБласті,
C  KOM-число точок у вихідних масивах,
C  NAP=0-немає автоматичного пошуку HMAX для заданого NS,
C  HMAX-макс.значення середньо-квадратичної похиБки апроксимації,
C  DHM-приріст HMAX при NAP=1,
C  OTN-макс.значення відн.значення функції у межах підоБласті,
C  DOTN-приріст OTN при NAP=1.
C  REG=0-немає регуляризації,
C  PREG-жорсткість регуляризації (максимальна при PREG=1.0)
  REWIND 8
  READ( 8)NFICH
  IF(NFICH.EQ.0) GOTO 1000
  READ( 8) LLB
  IF(LLB.EQ.0) GOTO 110
  READ( 8) (MB(I),I=1,LLB)
110  READ( 8) LLT
  IF(LLT.EQ.0) GOTO 1001
  READ( 8) (MT(I),I=1,LLT)
  IL=1
  N=0
118  IB=MT(IL+5)
  LLV=MT(IL+7)-1
  IF(LLV.NE.202) GOTO 121
  IB=IB-1
  IF(IB.LE.0) GOTO 120
  REWIND 7
  DO 119 I=1,IB
  READ( 7) ICA
119  CONTINUE
120  NOM=MT(IL+1)
  IF(NOM.EQ.APR1) READ(7) N, (B1(I),I=1,202)
  IF(NOM.EQ.APR2) READ(7) N, (B2(I),I=1,202)
  IF(NOM.EQ.APR3) READ(7) N, (B3(I),I=1,202)
121  IL=MT(IL)+1
  IF(IL.GT.1) GOTO 118
  IF(N.EQ.0) GOTO 1002
C  НОРМИРОВКА МАССИВОВ
  DO 123 I=1,M
123  KKK(I)=0.0
  KKY=0.0
  KOMM=KOM/2
  DO 1231 J=1,KOMM
  DO 1232 I=1,M
1232  X(I,J)=0.0
1231  YY(J)= 0.0
  IF(NS.EQ.1) DNS=0
  DO 598 I=1,KOMM

```



```

J=I*2-1
X1=B1(J)
X2=B2(J+1)
X3=B2(J)
X4=B3(J+1)
1239 FORMAT(' X1IN, X, X1, U      ')
X(1,I)=X1
X(2,I)=X2
X(3,I)=X3
X(4,I)=X4
X(5,I)=X1*X1
X(6,I)=X2*X2
X(7,I)=X3*X3
X( 8,I)=X4*X4
X( 9,I)=X1*X2
X(10,I)=X1*X3
X(11,I)=X1*X4
X(12,I)=X2*X3
X(13,I)=X2*X4
X(14,I)=X3*X4
M1=M-1
DO 5981 IJ=1,M1
  IF(KKX(IJ).LT.ABS(X(IJ,I))) KKX(IJ)=ABS(X(IJ,I))
5981 CONTINUE
  YY(I)=B1(J+1)
  IF(KKY.LT.ABS(YY(I))) KKX(IJ)=ABS(YY(I))
598 CONTINUE
DO 1221 I=1,M1
1221 IF(KKX(I).LT.1.0E-60) KKX(I)=1E25
  IF(KKY.EQ.0) KKX(I)=1E31
DO 599 I=1,KOMM
DO 597 J=1,M1
  X(J,I)=X(J,I)/KKX(J)
597 IF(ABS(X(J,I)).LT.EPS) X(J,I)=0.0
  YY(I)=YY(I)/KKX
599 IF(ABS(YY(I)).LT.EPS) YY(I)=0.0
C шкала меж груп у апрокс.полін. (LR(I)-ост.номер у (I-1)-й групі)
  LR(1)=0
  LR(2)=5
  LR(3)=9
  LR(4)=14
  LR(5)=20
  LR(6)=24
  LR(7)=35
C константи регуляризації (наБл.значення коеф. у межах груп)
  C(2)=-1E3
  C(3)=-1E3
  C(4)=-1E1
  C(5)=-1E0
  C(6)=-1E-1
C регуляризуюча доБавка
  KOMR=KOMM
  IF(REG.EQ.0) GOTO 902
DO 900 K=2,6
  JN=LR(K)+1
  JK=LR(K+1)
DO 900 J=JN,JK
  KOMR=KOMR+1
DO 901 I=1,M
901 X(I,KOMR)=0.0
  X(J,KOMR)=PREG/C(K)
900 YY(KOMR)=PREG
902 CONTINUE
  DO 596 I=1,M1

```

```

596 W4(I)=KKX(I)
    W4(M)=KKY
    PRINT 650
650 FORMAT(/'коефіцієнти нормування вихідних масивів')
    PRINT 1047, ( W4(I), I=1, M )
1047 FORMAT(10E10.3)
    PRINT 48
    JW=M
    NA=0
C   центри тяжіння по координатам
634 DO 635 I=1, M
    DO 6351 J=1, 10
    RY(J)=0.0
6351 RX(I, J)=0.0
    XCT(I)=0.0
635 XC(I)=0.0
    YC=0.0
    H=0.0
    DO 601 I=1, KOMM
601 AM(I)=0
    DO 6012 I=1, MQ
6012 XC(I)=XC(I)/KOMM
    YC=YC/KOMM
    DO 600 I=1, 410
600 CH(I)= 0
    KS0=1
    KS=0
    MKS(1)=0
    PR=0
    LSN=1
    KD=1
    KDNS=0
C   найближча (або найдальша, для ост. області) точка від ц.т.
640 M1=MM(KD)-1
    M=M1+1
    KON=KONN+(KD-1)*DNS
    KONR=KON
    IF(REG.NE.0) KONR=KON+(KOMR-KOMM)*KD
    SM=1E70
    IF(KD.GE.NS) SM=0.0
    DO 602 I=1, KOMM
    IF(AM(I).GT.0) GOTO 602
    S1=0.0
    DO 6021 J=1, MQ
6021 S1=S1+(X(J, I)-XC(J))**2
    S=SQRT(S1)
    IF(KD.LT.NS.AND.S.GE.SM) GOTO 602
    IF(KD.GE.NS.AND.S.LT.SM) GOTO 602
    SM=S
    IM=I
602 CONTINUE
    IF(NA.EQ.0) PRINT 44, IM, SM
C   початкове значення Біжучого ц.т.
    DO 6022 I=1, MQ
    XMIN(I)=X(I, IM)
    XMAX(I)=X(I, IM)
6022 XCT(I)=X(I, IM)
    YMIN=YY(IM)
    YMAX=YY(IM)
    YCT=YY(IM)
C   заповнення області
    TMAX=ABS(YCT)
    IF(TMAX.EQ.0.0.AND.KS.GT.0) TMAX=ABS(YY(CH(KS)))
    IF(TMAX.EQ.0.0.AND.KS.GT.1) TMAX=ABS(YY(CH(KS-1)))

```

```

      IF (TMAX.EQ.0.0.AND.KS.GT.2) TMAX=ABS (YY (CH (KS-2)))
      IF (TMAX.EQ.0.0) TMAX=1E-10
      TMIN=TMAX
608 SM=1E70
      M1=MM(KD)-1
C   пошук найБлижчої точки
      DO 603 I=1,KOMM
      IF (AM(I).GE.KD) GOTO 603
      IF (KD.NS.LT.1.AND.AM(I).GT.0) GOTO 603
      S1=0.0
      DO 6031 J=1,MQ
6031 S1=S1+(X(J,I)-XCT(J))**2
      S=SQRT(S1)
      IF (S.GE.SM) GOTO 603
      SM=S
      IM=I
603 CONTINUE
      IP=IM*2-1
      IF (NA.EQ.0) PRINT 4700,IM,SM,YY(IM),X(1,IM),X(2,IM),X(3,IM),
*   X(4,IM),X(5,IM),X(6,IM),X(7,IM),X(8,IM),X(9,IM)
4700 FORMAT(12G09.2)
      KS=KS+1
      CH(KS)=IM
      IF (KD.NS.LT.1) AM(IM)=KD
      IF (KD.NS.GT.0) AM(IM)=AM(IM)+10
C   Біжучі ц.т. та радіус
      DO 6071 JP=1,MQ
      IF (X(JP,IM).LT.XMIN(JP)) XMIN(JP)=X(JP,IM)
      IF (X(JP,IM).GT.XMAX(JP)) XMAX(JP)=X(JP,IM)
6071 CONTINUE
      IF (YY(IM).LT.YMIN) YMIN=YY(IM)
      IF (YY(IM).GT.YMAX) YMAX=YY(IM)
607 CONTINUE
621 DO 6211 I=1,MQ
      XCT(I)=(XMIN(I)+XMAX(I))/2
6211 RA(I)=XMAX(I)-XCT(I)
      YCT=(YMIN+YMAX)/2
      RS=YMAX-YCT
      LSK=KS
      M=MM(KD)
      IF (M.EQ.0) STOP 3
      T1=ABS (YY (CH (KS)))
      IF (T1.EQ.0.0) T1=TMIN
      IF (T1.LT.TMIN) TMIN=T1
      IF (T1.GT.TMAX) TMAX=T1
      IF (KS-KS0.LT.M.AND.KS.LT.KON) GOTO 608
      IF (NS.GT.1) GOTO 800
801 CONTINUE
      IF (KD.NS.GT.0) GOTO 8012
      IF (H.LT.HMAX.AND.KS.LT.KON.AND.TMAX/TMIN.LT.OTN) GOTO 608
      IF (H.LT.HMAX/1000.AND.KS.LT.KON) GOTO 608
8012 KDNS=KD.NS+1
      IF (KD.NS.LE.DNS) GOTO 608
      KDNS=0
      DO 8011 I=1,KOMM
      IF (AM(I).GE.10) AM(I)=AM(I)-10
8011 CONTINUE
      KD=KD+1
      MKS(KD)=KS
C   Ц.т. області, радіус області, радіус по координатам-по мітці 621
      JN=MKS(KD-1)+1
      JK=MKS(KD)
610 SM1=0.0
      SM=0.0

```

```

DO 606 J=JN,JK
I=CH(J)
S=0.0
S1=0.0
DO 6061 JP=1,MQ
6061 S1=S1+(X(JP,I)-XCT(JP))**2
S= SQRT(S1)
IF(S.GT.SM) SM=S
606 CONTINUE
RAD=1.0*SM*SM
SQRAD=0.9*SM
K=KD-1
DO 6221 I=1,MQ
RX(I,K)=RA(I)
6221 MXCT(I,K)=XCT(I)
MYCT(K)=YCT
RY(K)=RS
RAXY(K)=RAD
SQRADX(K)=SQRAD
IF(NA.EQ.0) PRINT 4700,XCT(1),XCT(2),XCT(3),XCT(4),XCT(5),
*      XCT(6),XCT(7),XCT(8),XCT(9),XCT(10),YCT,RAD,SQRAD,H
IF(NA.EQ.0) PRINT 48
C  запис регуляризуючої доБавки
IF(REG.EQ.0) GOTO 911
JR1=KOMM+1
DO 910 JR=JR1,KOMR
KS=KS+1
910 CH(KS)=JR
911 MKS(KD)=KS
C  Блок автомат. пошуку HMAX та OTN для заданого NS
IF(KD.LE.NS.AND.KS.LT.KONR) GOTO 633
PRINT 44,NA,KD,KS,HMAX,OTN
IF(NAP.EQ.0) GOTO 604
IF(KD.GT.NS.AND.KS.GE.KONR) GOTO 604
NA=NA+1
IF(NA.GT.10) STOP 4
IF(KD.GT.NS.AND.KS.LT.KONR.AND.DHM.LT.0) GOTO 623
IF(KD.LE.NS.AND.KS.GE.KONR.AND.DHM.GT.0) GOTO 623
GOTO 624
623 DHM=-DHM/2
DOTN=-DOTN/2
624 HMAX=HMAX+DHM
OTN=OTN+DOTN
GOTO 634
C  початковий ц.т. наступної оБласті,визначається по мітці 6022
633 DO 6331 I=1,MQ
RA(I)=0.0
6331 XCT(I)=0.0
YCT=0.0
RS=0.0
KS0=KS+1
LSN=KS0
GOTO 640
604 CONTINUE
C  Сплайн-підобласті не рівні по кількості точок (MKS, макс.число
C  сплайнів-NS)
C  Розмір підобласті вибирається автоматично по заданій середньоквадр.
C  похиБці HMAX аБо макс. відношенню крайніх знач. ф-ції OTN
C  Значення ординат нормуються в інтервалі /1,3/
C  Стикування сплайнів завдяки перекриванню підобластей (DNS)
C  Сплаيني неоднакового порядку (масив MM)
C  Апроксимація сплайнів
LS=1
PR=1

```

```

700  LSK=MKS (LS+1)
      LSN=MKS (LS) +1
      M=MM (LS)
800  E=0
      DO 122 J=LSN, LSK
      E=E+1
      I=CH (J)
      CH1 (E)=I
122  Y (E)=YY (I)
      IF (PR.EQ.1) PRINT 44,E,M,EPS,LS,NS,DNS
      KY=0
      DO 131 I=B,E
      IF (ABS (Y (I)) .LT. KY) GOTO 131
      KY=ABS (Y (I))
      IM=I
131  CONTINUE
      KY=Y (IM)
      DO 132 I=B,E
132  Y (I)=Y (I) /KY
      BY=0.0
      DO 101 I=1,M
101  W (I)=0
      DO 2 K=B,E
      F (1,K)=1.0
2    IF (LSN.EQ.1) F (1,K)=0.0
      DO 1 I=2,M
      DO 1 K=B,E
      KI=CH1 (K)
      F (I,K)= X (I-1,KI)
1    CONTINUE
      40  FORMAT ( 6Q13.5)
      44  FORMAT (6G11.3)
      47  FORMAT (8G11.3)
      48  FORMAT (' ')
      49  FORMAT (6E11.3)
      DO 20 I=1,M
      L (I,I)=1
      I1=I-1
      IF (I1.LT.1) GOTO 41
      DO 4 J=1,I1
      IF (W (J) .LT. EPS) GO TO 5
      S=0
      DO 6 K=B,E
      G=F (I,K) *F (J,K)
      IF (PR.NE.0) CALL CON (1,S,G)
6    S=S-G
      L (I,J)=S/W (J)
      GO TO 4
5    L (I,J)=0
4    CONTINUE
41  DO 8 K=B,E
      S=0
      DO 11 J=1,I
      G=L (I,J) *F (J,K)
      IF (PR.NE.0) CALL CON (2,S,G)
11  S=S+G
8    F (I,K)=S
      I2=I-2
      IF (I2.LT.1) GOTO 12
      DO 9 J=1,I2
      S=0
      DO 7 K=J,I1
      G=L (I,K) *L (K,J)
      IF (PR.NE.0) CALL CON (3,S,G)

```

```

7      S=S+G
9      L(I,J)=S
12     S=0
        DO 10 K=B,E
        G=F(I,K)**2
        IF(PR.NE.0) CALL CON(4,S,G)
10     S=S+G
20     W(I)=S
        DO 21 I=1,M
        IF(W(I).LT.EPS) GO TO 22
        S=0
        DO 23 K=B,E
        G=Y(K)*F(I,K)
        IF(PR.NE.0) CALL CON(5,S,G)
23     S=S+G
        W(I)=S/W(I)
        GO TO 21
22     W(I)=0
21     CONTINUE
        A=0
        DO 25 K=B,E
        S=0
        DO 26 J=1,M
26     S=S+W(J)*F(J,K)
        U(K)=(S-BY)*KY*KKY
25     A=A+(Y(K)-S)**2
        H=A
        H= SQRT(H/(E-B+1))
        IF(PR.EQ.0) GOTO 801
        DO 27 I=1,M
        S=0
        DO 28 K=I,M
        G=W(K)*L(K,I)
        IF(PR.NE.0) CALL CON(6,S,G)
28     S=S+G
27     W(I)=S
        W(1)=(W(1)-BY)*KY
        DO 30 I=2,M
        W(I)=W(I)*KY
30     CONTINUE
        DO 31 I=B,E
31     Y(I)=(Y(I)-BY)*KY*KKY
        DO 3102 I=B,E
        U(I)=W(1)*KKY
        DO 3102 J=2,M
3102  U(I)=U(I)+W(J)*X(J-1,CH1(I))*KKY
        DO 3101 I=1,M1
3101  W(M+I)=MXCT(I,LS)
        W(M+M1+1)=MYCT(LS)
C      Запис радіусу та його кореня      MS=M+M1+3      W(MS-1)=RAXY(LS)
        W(MS)=SQRADX(LS)
        PRINT 1239
        PRINT 49,H
        PRINT 40,(W(I),I=1,MS)
        PRINT 48
        PRINT 49,(Y(I),I=1,E)
        PRINT 48
        PRINT 49,(U(I),I=1,E)
        DO 32 I=1,MS
32     W4(JW+I)=W(I)
        JW=JW+MS
        LS=LS+1
        IF(LS.LE.NS) GOTO 700
        IF(ECR.EQ.0) GOTO 1004

```

```

29  IB=NFICH
    NFICH=NFICH+1
    DO 502 I=2,LLT,10
    IF (MT(I).EQ.NOMT1.AND.MT(I+1).EQ.NOMT2) GOTO 1003
502  CONTINUE
    MT(LLT-9)=LLT
    IL=LLT+1
    LLT=LLT+10
    MT(IL)=0
    MT(IL+1)=NOMT1
    MT(IL+2)=NOMT2
    MT(IL+3)=NOMM1
    MT(IL+4)=NOMM2
504  MT(IL+5)=NFICH
    MT(IL+6)=0
    MT(IL+7)=JW+1
    MT(IL+8)=0
    MT(IL+9)=BLAN
    REWIND 8
    WRITE(8) NFICH
    WRITE(8) LLB
    IF(LLB.EQ.0) GOTO 500
    WRITE(8) (MB(I),I=1,LLB)
500  WRITE(8) LLT
    WRITE(8) (MT(I),I=1,LLT)
    REWIND 7
    DO 501 I=1,IB
    READ(7) ICA
501  CONTINUE
    WRITE(7) JW, (W4(I),I=1,JW)
    PRINT 47,JW, (W4(I),I=1,JW)
    PRINT 305,NOMT1,NOMT2,NOMM1,NOMM2
305  FORMAT(/' Записано ',A4,A4,'(',A4,A4,')')
    STOP
1003 PRINT 303
    303 FORMAT(/' Цей тип вже Був записаний, тому -')
1004 PRINT 304
    STOP
    304 FORMAT(/' запису немає')
1000 PRINT 300
    300 FORMAT(/' ЮіБліотека пуста')
    STOP
1001 PRINT 301
    301 FORMAT(/' ТаБлиць немає')
    STOP
1002 PRINT 306
    STOP
    306 FORMAT(/' ПотріБних таБлиць немає')
    END

```

Додаток 9

Модель у форматі вхідної мови системи САНОС^ що відповідає макромоделі з розбиттям на 4 підобласті та автоматичним вибором апроксимації у найближчій підобласті.

```

BEGI
MOS 4.14
DESC
MODEL:FS414(0,0)K1,K2,K3,K4,K5,K6,K7,K8,K9,K10,K11,K12,K13,K14,KY,
  A01,A11,A21,A31,A41,A51,A61,A71,A81,A91,A101,A111,A121,A131,A141,Ц11,
  Ц21,Ц31,Ц41,Ц51,Ц61,Ц71,Ц81,Ц91,Ц101,Ц111,Ц121,Ц131,Ц141,ЦY1,SQ1,RA1,
  A02,A12,A22,A32,A42,A52,A62,A72,A82,A92,A102,A112,A122,A132,A142,Ц12,
  Ц22,Ц32,Ц42,Ц52,Ц62,Ц72,Ц82,Ц92,Ц102,Ц112,Ц122,Ц132,Ц142,ЦY2,SQ2,RA2,

```

```

A03,A13,A23,A33,A43,A53,A63,A73,A83,A93,A103,A113,A123,A133,A143,Ц13,
Ц23,Ц33,Ц43,Ц53,Ц63,Ц73,Ц83,Ц93,Ц103,Ц113,Ц123,Ц133,Ц143,ЦY3,SQ3,RA3,
A04,A14,A24,A34,A44,A54,A64,A74,A84,A94,A104,A114,A124,A134,A144,Ц14,
Ц24,Ц34,Ц44,Ц54,Ц64,Ц74,Ц84,Ц94,Ц104,Ц114,Ц124,Ц134,Ц144,ЦY4,SQ4,RA4;
*A05,A15,A25,A35,A45,A55,A65,A75,A85,A95,A105,A115,A125,A135,A145,Ц15,
Ц25,Ц35,Ц45,Ц55,Ц65,Ц75,Ц85,Ц95,Ц105,Ц115,Ц125,Ц135,Ц145,ЦY5,SQ5,RA5;
Z1=X1/K1;Z2=X2/K2;Z3=X3/K3;Z4=X4/K4;Z5=X1*X1/K5;Z6=X2*X2/K6;Z7=X3*X3/
K7;Z8=X4*X4/K8;Z9=X1*X2/K9;Z10=X1*X3/K10;Z11=X1*X4/K11;Z12=X2*X3/K12;
Z13=X2*X4/K13;Z14=X3*X4/K14;
Y1=A01+A11*Z1+A21*Z2+A31*Z3+A41*Z4+A51*Z5+A61*Z6+A71*Z7+A81*Z8+A91*Z9+
A101*Z10+A111*Z11+A121*Z12+A131*Z13+A141*Z14;      YM=YS/KY;
Y2=A02+A12*Z1+A22*Z2+A32*Z3+A42*Z4+A52*Z5+A62*Z6+A72*Z7+A82*Z8+A92*Z9+
A102*Z10+A112*Z11+A122*Z12+A132*Z13+A142*Z14;
Y3=A03+A13*Z1+A23*Z2+A33*Z3+A43*Z4+A53*Z5+A63*Z6+A73*Z7+A83*Z8+A93*Z9+
A103*Z10+A113*Z11+A123*Z12+A133*Z13+A143*Z14;
Y4=A04+A14*Z1+A24*Z2+A34*Z3+A44*Z4+A54*Z5+A64*Z6+A74*Z7+A84*Z8+A94*Z9+
A104*Z10+A114*Z11+A124*Z12+A134*Z13+A144*Z14;
D11=Z1-Ц11; D21=Z2-Ц21; D31 =Z3 -Ц31 ; D41 =Z4 -Ц41 ; D51 =Z5 -Ц51 ;
D61=Z6-Ц61;D71=Z7-Ц71;D81=Z8-Ц81;D91=Z9-Ц91;D101=Z10-Ц101;DY1=YM-ЦY1;
D111=Z11-Ц111; D121=Z12-Ц121; D131=Z13-Ц131; D141=Z14-Ц141;
D12=Z1-Ц12; D22=Z2-Ц22; D32 =Z3 -Ц32 ; D42 =Z4 -Ц42 ; D52 =Z5 -Ц52 ;
D62=Z6-Ц62;D72=Z7-Ц72;D82=Z8-Ц82;D92=Z9-Ц92;D102=Z10-Ц102;DY2=YM-ЦY2;
D112=Z11-Ц112; D122=Z12-Ц122; D132=Z13-Ц132; D142=Z14-Ц142;
D13=Z1-Ц13; D23=Z2-Ц23; D33 =Z3 -Ц33 ; D43 =Z4 -Ц43 ; D53 =Z5 -Ц53 ;
D63=Z6-Ц63;D73=Z7-Ц73;D83=Z8-Ц83;D93=Z9-Ц93;D103=Z10-Ц103;DY3=YM-ЦY3;
D113=Z11-Ц113; D123=Z12-Ц123; D133=Z13-Ц133; D143=Z14-Ц143;
D14=Z1-Ц14; D24=Z2-Ц24; D34 =Z3 -Ц34 ; D44 =Z4 -Ц44 ; D54 =Z5 -Ц54 ;
D64=Z6-Ц64;D74=Z7-Ц74;D84=Z8-Ц84;D94=Z9-Ц94;D104=Z10-Ц104;DY4=YM-ЦY4;
D114=Z11-Ц114; D124=Z12-Ц124; D134=Z13-Ц134; D144=Z14-Ц144;
DU1=(D11*D11+D21*D21+D31*D31+D41*D41);
DU2=(D12*D12+D22*D22+D32*D32+D42*D42);
DU3=(D13*D13+D23*D23+D33*D33+D43*D43);
DU4=(D14*D14+D24*D24+D34*D34+D44*D44);
YN=IF(R1.LT.R4) (P1) (P2);
R1=(D1.LT.0.0) (-D1) (D1*1E10);      P1=IF(R1.LT.R3) (P3) (P4);
R2=IF(D2.LT.0.0) (-D2) (D2*1E10);      P2=IF(R2.LT.R4) (P4) (P5);
R3=IF(D3.LT.0.0) (-D3) (D3*1E10);      P3=IF(R1.LT.R2) (Y1) (Y2);
R4=IF(D4.LT.0.0) (-D4) (D4*1E10);      P4=IF(R2.LT.R3) (Y2) (Y3);
P5=IF(R3.LT.R4) (Y3) (Y4);
Y=KY*YN;;;
BIBL
DELETE:FS414 ;
WRITE:FS414 ;
PRINT;;;
END

```


СПИСОК ЛІТЕРАТУРИ

(за порядком посилань у тексті)

1. Гринбаум А. (Миллер Г.) Модели цифровых ИС для машинного проектирования// Электроника. 1973. №25. С.46-52; №26. С.63-69; 1974. №2. С.49-53; №3. С.55-59; №5. С.56-59.
2. Директор С., Рорер Н. Введение в теорию систем/ Пер. с англ. М., 1974.
3. Заде Л., Дезоер Ч.. Теория линейных систем/ Пер. с англ. М., 1979.
4. Винер Н. Новые главы кибернетики/ Пер. с англ. М., 1963.
5. Калман Р., Фалб П., Арбиб М. Очерки по математической теории систем/ Пер. с англ. М., 1971.
6. Чуа Л.О., Пен-Мин Лин. Машинный анализ электронных схем/ Пер. с англ. М., 1980.
7. Takens F. Detecting Strange Attractor in Turbulence// Lect. Notes in Math., 1981. Vol.898, P.366-381.
8. Букашкин С.А. Адаптивные алгоритмы синтеза нелинейных электронных схем. К., 1989.
9. Верлань А.Ф., Москалюк С.С. Математическое моделирование непрерывных динамических систем. К., 1988.
10. Данилов Л.В. Функциональные ряды Вольтерра-Пикара в теории нелинейных электрических цепей. М., 1987.
11. Денбновецкий С.В., Писаренко Л.Д., Резниченко В.К. Основы автоматизированного проектирования электронных приборов. К., 1987.
12. Ивахненко А.Г., Степашко В.С. Помехоустойчивость моделирования. К., 1985.
13. Ильин В.Н. Проблемы макро моделирования// Автоматизация проектирования в радиоэлектронике и вычислительной технике. М., 1981. С.23-30.
14. Ланнэ А.А. Нелинейные динамические системы: синтез, оптимизация, идентификация. Л., 1985.
15. Каминскас В. Идентификация динамических систем по дискретным наблюдениям. Вильнюс, 1985.
16. Пупков К.А., Капалин В.И., Ющенко А.С. Функциональные ряды в теории нелинейных систем. М., 1976.
17. Петренко А.И., Бобин В.В., Романов В.В. Макромоделирование цифровых и аналоговых интегральных схем// Зарубежная радиоэлектроника. 1981. №7. С.3-27.
18. Пухов Г.Е. Приближенные методы математического моделирования, основанные на применении дифференциальных Т-преобразований. К., 1988.
19. Сигорский В.П., Фесечко В.А., Мельник А.С. Моделирование и идентификация компонентов и узлов электронной техники. К., 1981.
20. Синицкий Л.А., Фелыштын О.И. Математическая модель генератора квазипериодических колебаний// Радиотехника и электроника. 1986. Т.31. №8. С.1598-1604.

21. *Стахів П.Г., Мельник Б.К., Джала В.Р.* Використання оптимізаційного підходу при побудові нелінійних макромоделей багатополосних компонент електронних кіл в базисі змінних стану// Теоретична електротехніка. 1996. Вип.53. С.29-35.
22. *Стахів П.Г., Мельник Б.К., Джала В.Р.* Построение дискретных макромоделей нелинейных динамических объектов с многими входами// Электронное моделирование. 1996. Т.18. №3. С.89-91.
23. *Матвийчук Я.Н.* Общие структуры макромоделей нелинейных динамических систем// Электронное моделирование. 1994. №4. С.42-47.
24. *Садовский В.Н.* Основания общей теории систем. М., 1974.
25. *Попов Б.А., Теслер Г.С.* Приближение функций для технических приложений. К., 1980.
26. *Алексенко А.Г.* Макромоделирование аналоговых интегральных микросхем. М., 1983.
27. *Heydemann M.* Functional macromodelling of electrical circuits// Proc.IEEE Int. Symp. Circuits and Systems. 1977. P.692-695.
28. *Ланнэ А.А.* Синтез нелинейных систем// Электронное моделирование. 1980. №1. С.60-68.
29. *Данилов Л.В.* Электрические цепи с нелинейными элементами. М., 1974.
30. *Вольтерра В.* Теория функционалов, интегральных и интегродифференциальных уравнений/ Пер. с англ. М., 1982.
31. *Тихонов А.Н., Арсенин В.Я.* Методы решения некорректных задач. М., 1979.
32. *Шафт ван дер.* К теории реализации нелинейных систем // Новое в зарубежной науке. Математика. Теория систем. Математические методы и моделирование/ Пер. с англ. М., 1989.
33. *Isidori A.* Direct construction of minimal bilinear realization from nonlinear input-output maps// IEEE Trans. 1973. Vol.AC-18. №6. P.626-631.
34. *Hinamoto T., Mackava S.* Approximation of polynomial state-affine discret-time systems// IEEE Trans.Cir. & Syst. 1984. Vol.CAS-31. №8. P.713-721.
35. *Войтенков И.Н.* Методы и средства дифференциального оценивания и идентификации моделей. К., 1989.
36. *Broomhead D.S., King G.P.*// Physika D. 1986. Vol.20. №2. P.217-221.
37. *Грибков Д.А., Грибкова В.В. и др.* Восстановление дифференциальных уравнений автостохастических систем по временной реализации одной динамической переменной процесса// Журн. техн. физ. 1994. Т.64. №3. С.1-12.
38. *Винер Н.* Кибернетика или управление и связь в животном и машине/ Пер. с англ. М., 1963.
39. *Стахів П.Г.* Анализ динамических режимов в электронных схемах с многополосниками. Львов: 1988.
40. *Беянин А.Н., Золотницкий В.М., Парасочкин В.А.* Аналитический метод расчета нелинейных электрических цепей при медленно

изменяющихся воздействиях// Тр. 29 междунар. коллоквиума. Ильменау, ГДР, 1984. Т.3. С.31-32.

41. *Бороненко Т.А., Парасочкин В.А.* Приближенное аналитическое исследование нелинейных цепей с коротким переходным процессом. Л., 1985. 6с. Депонировано в ИНИОН ВИНТИ, №2009-85.

42. *Лозовский Ю.Н., Парасочкин В.А., Ткаченко В.Т.* Аппроксимация и восстановление многомерных функций при макро моделировании// Изв. ВУЗов. Радиоэлектроника. 1988. №8. С.43-47.

43. *Матвийчук Я.Н.* Макромодель линейной электронной цепи во временной области// Теоретическая электротехника. 1982. Вып.33. С.157-160.

44. *Верлань А.Ф., Сизиков В.С.* Интегральные уравнения: Методы, алгоритмы, программы. К., 1986.

45. *Лоусон Ч., Хенсон Р.* Численное решение задач метода наименьших квадратов/ Пер. с англ. М., 1986.

46. *Иванов В.В.* Методы вычислений на ЭВМ. К., 1986.

47. *Самарский А.А., Гулин А.В.* Численные методы. М., 1989.

48. *Волох В.Д.* Аппроксимация методом наименьших квадратов функций многих переменных// Математическое обеспечение для ЭВМ серии МИР. К., 1974.

49. *Марчук Г.И.* Методы вычислительной математики. М., 1989.

50. *Barrodale I., Young A.* Algorithms for best L-sub-one and L-sub-infinity linear approximation on a discrete set// Numerische Mathematik. 1966. Vol.8. ISS.3. P.295-306.

51. *Дьяконов В.П.* Справочник по применению системы PC MATLAB. М., 1993.

52. *Странные аттракторы:* Пер. с англ. Новое в зарубежной науке. Математика. Вып.22. М., 1981.

53. *Рудин У.* Основы математического анализа. М., 1966.

54. *Андронов А.А., Витт А.А., Хайкин С.Э.* Теория колебаний. М., 1959.

55. *Лисицкая И.Н., Сеницкий Л.А., Шумков Ю.М.* Расчет электрических цепей с магнитными и полупроводниковыми элементами. К., 1969.

56. *Матвийчук Я.Н.* Идентификация макромоделей нелинейных динамических систем методом обратной линейной подсистемы// Теоретическая электротехника. 1987. №42. С.81-83.

57. *Гантмахер Ф.Р.* Теория матриц. М., 1967.

58. *Матвийчук Я.Н.* Численный синтез математической модели автогенератора с нелинейным гистерезисным элементом// Теоретическая электротехника. 1992. №51. С.32-38.

59. *Матвийчук Я.М.* Математичне моделювання хаотичних рухів у детермінованих системах// Вісн. Львів. ун-ту. Сер. фіз. 1993. Вип.26. С.61-66.

60. *Матвийчук Я.Н.* Синтез детерминированных систем, воспроизводящих заданные случайные последовательности// Электронное моделирование. 1998. Т.20. С.25-31.

61. *Мун Ф.* Хаотические колебания/ Пер. с англ. М., 1990.

62. *К. де Бор*. Практическое руководство по сплайнам/ Пер. с англ. М., 1985.
63. *Матвійчук Я.М.* Апроксимація функцій багатьох аргументів у задачах синтезу макромоделей нелінійних динамічних систем// Теоретична електротехніка. 1994. Вип.52. С.45-50.
64. *Матвійчук Я.М.* Математичне макромодельовання синхронного генератора// Теоретична електротехніка. 1996. Вип.53. С.156-161.
65. *Новаковский С.В.* Стандартные системы цветного телевидения. М., 1976.
66. *Злобін Г.Г., Петрович О.В., Слободян А.Л.* Один підхід до моделювання телевізійних кодуєчих пристроїв// Теоретична електротехніка. 1994. Вип.52. С.56-61.
67. *Тихонов А.Н., Гончарский А.В., Степанов В.В., Ягола А.Г.* Численные методы решения некорректных задач. М., 1990.
68. *Букашкин С.А., Кузиев Э.М.* Синтез алгоритмов цифровых рекурсивных демодуляторов АМ- и ЧМ-сигналов// Изв. ВУЗов. Радиоэлектроника. 1989. Т.32. №12. С.34-41.
69. *Курганевич А., Матвійчук Я.* Дослідження некоректності задачі ідентифікації макромоделей нелінійних динамічних систем на прикладі частотного детектора// Сб. докл. Междунар. науч.-технич. конф. "Проблемы физической и биомедицинской электроники". К., 1995. С.67.
70. *Матвійчук Я.Н., Федорчук Е.Н.* Система анализа и оптимизации электронных схем// Теоретическая электротехника. 1980. Вып.28. С.139-143.
71. *Kennedy M.P.* Robust OPAMP realization of Chua's circuit. Frequenz, 1992. 46(34). P.66-80.
72. *Матвійчук Я.М., Хараба М.В.* Електрична та математичні реалізації схеми Чуа// Теоретична електротехніка. 1994. Вип.52. С.169-179.

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1. ЗАГАЛЬНИЙ ОГЛЯД ПРОБЛЕМИ	5
1.1. ВИЗНАЧЕННЯ МАКРОМОДЕЛІ	5
1.2. ФОРМУЛЮВАННЯ ЗАДАЧІ МАКРОМОДЕЛЮВАННЯ ТА СИСТЕМНИЙ ОПИС ОБ'ЄКТІВ	6
1.3. СТРУКТУРИ МАКРОМОДЕЛЕЙ ТА МЕТОДИ ІДЕНТИФІКАЦІЇ	9
1.3.1. Методи спрощення та агрегування	9
1.3.2. Методи функціональної подібності	10
1.3.3. Структури математичних макромоделей	12
1.3.4. Методи ідентифікації математичних макромоделей	20
РОЗДІЛ 2. МАКРОМОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ	24
2.1. ІНТЕГРАЛЬНІ ОПЕРАТОРИ	24
2.2. ДИФЕРЕНЦІАЛЬНІ ОПЕРАТОРИ	28
2.3. НЕКОРЕКТНОСТІ МАКРОМОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ. РЕГУЛЯРИЗАЦІЯ ОБЧИСЛЕННЯ ПОХІДНИХ	30
2.4. ОРГАНІЗАЦІЯ МАКРОМОДЕЛЮВАННЯ ЛІНІЙНИХ СИСТЕМ	37
2.4.1. Макромодель фільтра p 'ятого порядку	38
2.4.2. Макромоделі довгої лінії за середньоквадратичним та мінімаксним критеріями	41
2.4.3. Макромоделювання за частотними характеристиками	46
РОЗДІЛ 3. СТРУКТУРИ МАКРОМОДЕЛЕЙ ТА МЕТОДИ ІДЕНТИФІКАЦІЇ НЕЛІНІЙНИХ СИСТЕМ ІЗ ЗОСЕРЕДЖЕНИМИ ПАРАМЕТРАМИ	52
3.1. СТРУКТУРИ МАКРОМОДЕЛЕЙ НЕЛІНІЙНИХ СИСТЕМ	52
3.1.1. Структура макромоделі нелінійної системи з нестационарною лінійною підсистемою	52
3.1.2. Структура макромоделі нелінійної системи зі стаціонарною лінійною підсистемою	55
3.1.3. Перетворення рівнянь Лоренца згідно зі структурою макромоделі	58
3.1.4. Окремі випадки загальної моделювальної структури	60
3.1.5. Нерекурсивна структура Вінера	61
3.2. ІДЕНТИФІКАЦІЯ ЗА МЕТОДОМ НЕЛІНІЙНОЇ СТАТИКИ ТА МАЛОСИГНАЛЬНОЇ ДИНАМІКИ	64
3.2.1. Загальні положення	64
3.2.2. Ідентифікація рекурсивної макромоделі. Макромодель тригера	66
3.2.3. Ідентифікація нерекурсивної макромоделі. Макромоделі стабілізатора та операційного підсилювача	79
3.3. ІДЕНТИФІКАЦІЯ ЗА МЕТОДОМ ОБЕРНЕНОЇ ЛІНІЙНОЇ	

ПІДСИСТЕМИ	88
3.3.1. Загальні положення	88
3.3.2. Макромодель генератора майже гармонічного сигналу	90
3.3.3. Макромодель мультивібратора	92
РОЗДІЛ 4. РЕГУЛЯРИЗАЦІЯ АПРОКСИМАЦІЙНИХ ЗАДАЧ МАКРОМОДЕЛЮВАННЯ	103
4.1. ЗГЛАДЖУВАЛЬНИЙ ФУНКЦІОНАЛ. МАКРОМОДЕЛЬ СИСТЕМИ З ВИПАДКОВИМ СИГНАЛОМ	104
4.2. ВИБІР АПРОКСИМАЦІЙНОГО БАЗИСУ. МАКРОМОДЕЛЬ СИНХРОННОГО ГЕНЕРАТОРА	112
4.3. ФУНКЦІОНАЛЬНЕ РОЗБИТТЯ ЗАДАЧІ. МАКРОМОДЕЛЬ БЛОКА КОЛЬОРУ ТЕЛЕВІЗІЙНОГО ПРИЙМАЧА	117
4.4. ДЕКОМПОЗИЦІЯ АПРОКСИМОВАНОЇ ФУНКЦІЇ В ПРОСТОРІ АРГУМЕНТІВ	126
4.5. ДЕКОМПОЗИЦІЯ АПРОКСИМОВАНОЇ ФУНКЦІЇ В ЧАСОВІЙ ТА ЧАСТОТНІЙ ОБЛАСТЯХ	129
4.6. РЕДУКЦІЯ АПРОКСИМУЮЧОГО ПОЛІНОМА. МАКРОМОДЕЛЬ ЧАСТОТНОГО ДЕТЕКТОРА	130
4.7. ВИЗНАЧЕННЯ РОЗМІРНОСТІ ДИНАМІЧНИХ СИСТЕМ ЗА ВИХІДНИМ СИГНАЛОМ	139
РОЗДІЛ 5. ВИКОРИСТАННЯ В МАКРОМОДЕЛЮВАННІ СИСТЕМИ САНОС-ПК	143
5.1. ЗАГАЛЬНИЙ ОПИС СИСТЕМИ	143
5.2. КОРОТКИЙ ОПИС ВХІДНОЇ МОВИ	146
5.3. МОДЕЛЮВАННЯ ДИВНИХ АТРАКТОРІВ СХЕМИ ЧУА	149
5.4. ЗАГАЛЬНА СХЕМА МАКРОМОДЕЛЮВАННЯ З ВИКОРИСТАННЯМ СИСТЕМИ САНОС-ПК	159
ДОДАТКИ	161
СПИСОК ЛІТЕРАТУРИ	201

Н а у к о в е в и д а н н я

Матвійчук
Ярослав Миколайович

**МАТЕМАТИЧНЕ МАКРОМОДЕЛЮВАННЯ
ДИНАМІЧНИХ СИСТЕМ: ТЕОРІЯ ТА ПРАКТИКА**

Підп. до друку 13.06.2000. Формат 60x84/16.
Офс. друк. Обл.-вид. арк. 13.00. Тираж 400 прим.
Зам. 286.

Видавничий центр Львівського національного університету
імені Івана Франка. 79000 Львів, вул.Д.Дорошенка,41.