

**Міністерство освіти і науки України**  
**Центральноукраїнський національний технічний університет**  
**Факультет автоматики та енергетики**  
**Кафедра автоматизації виробничих процесів**

**ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ СКЛАДНИХ БАГАТОВИМІРНИХ  
РУХОМИХ ОБ'ЄКТІВ**

**Методичні вказівки до виконання практичних занять**

**з курсу: «Ідентифікація та моделювання складних багатовимірних рухомих об'єктів»  
для аспірантів 151 «Автоматизація та комп'ютерно-інтегровані технології»**

**Кропивницький 2021**

**Міністерство освіти і науки України**  
**Центральноукраїнський національний технічний університет**  
**Факультет автоматики та енергетики**  
**Кафедра автоматизації виробничих процесів**

**ІДЕНТИФІКАЦІЯ ТА МОДЕЛЮВАННЯ СКЛАДНИХ БАГАТОВИМІРНИХ  
РУХОМИХ ОБ'ЄКТІВ**

Методичні вказівки до виконання практичних занять  
з курсу: «Ідентифікація та моделювання складних багатовимірних рухомих  
об'єктів»  
для аспірантів 151 «Автоматизація та комп'ютерно-інтегровані технології»

Затверджена  
на засіданні кафедри АВП  
протокол № 5 від 01.11.2021 р.

**Кропивницький 2021**

Ідентифікація та моделювання складних багатовимірних рухомих об'єктів. Методичні вказівки до виконання практичних занять з курсу: «Ідентифікація та моделювання складних багатовимірних рухомих об'єктів» для аспірантів 151 «Автоматизація та комп'ютерно-інтегровані технології» / Укл. : В.А. Зозуля.- Кропивницький: ЦНТУ, 2021. – 33с.

Укладачі: В.А. Зозуля - к.т.н, доцент кафедри АВП

Рецензент: С.І. Осадчий– д.т.н., професор АВП

Центральноукраїнський національний технічний університет, 2021р.

## ЗМІСТ

1. Практичне заняття №1. Дослідження лінійних стаціонарних систем	5
2. Практичне заняття № 2. Аналітичні методи моделювання об'єкта з зосередженими параметрами	9
3. Практичне заняття №3. Моделювання об'єкта регулювання рівня рідини в резервуарі	12
4. Практичне заняття №4. Ідентифікація об'єкта за допомогою передавальної функції	14
5. Практичне заняття № 5. Визначення параметрів узагальненої динамічної моделі одним з методів статичного оцінювання	16
6. Практичне заняття № 6 Визначення параметрів регресійної моделі з використанням рекурентних методів оцінювання	18
7. Практичне заняття № 7. Параметрична ідентифікація лінійних систем	22
8. Практичне заняття № 8. Непараметрична ідентифікація динамічних об'єктів	23
9. Практичне заняття № 9. Методи ідентифікації із застосуванням настроювання адаптивних моделей	27
10. Практичне заняття № 10. Ідентифікація із застосуванням методу вагових функцій	30
11. Практичне заняття № 11. Кореляційний метод ідентифікації	31

## Практичне заняття №1. Дослідження лінійних стаціонарних систем

**Мета роботи:** створення моделей динамічних систем в командному вікні MatLab; аналіз моделей засобами пакета Control System Toolbox.

### 1.1 Основні види моделей лінійних динамічних об'єктів

В цілому ряді випадків реакція лінійної стаціонарної системи на ту чи іншу досить просте вплив визначається її теоретичними характеристиками. Основними з них є:

- передавальна функція  $W(p)$  (визначається за допомогою перетворення Лапласа  $L\{\cdot\}$ );
- імпульсна характеристика  $g(t)$  - реакція попередньо незбуреного об'єкта на вхідний сигнал у вигляді  $\delta$ -функції.

При нульових початкових умовах зв'язок між вихідним і вхідним сигналами описується інтегралом згортки

$$y(t) = \int_{-\infty}^{+\infty} g(t-\tau)u(\tau)d\tau,$$

- *перехідна функція*  $h(t)$  - реакція попередньо незбуреного об'єкта на вхідний сигнал у вигляді одиничного стрибка.

З теорії керування відомі наступні співвідношення між цими характеристиками

$$L\{h(t)\} = \frac{W(p)}{p};$$

$$L\{w(t)\} = W(p), g(t) = h'(t),$$

- *частотні характеристики* об'єкта, які визначаються комплексним коефіцієнтом передачі  $W(j\omega) = W(p)|_{p=j\omega}$ .

Під моделлю будемо розуміти інформацію про найбільш суттєві характеристики об'єкта, виражену в тій чи іншій формі.

Наведемо основні види моделей лінійних безперервних стаціонарних динамічних об'єктів.

### 1. Диференціальні рівняння.

Найбільш універсальна модель, заснована на диференціальних рівняннях, описується виразом

$$\sum_{i=0}^{na} a_i y^{(i)} = \sum_{j=0}^{nb} b_j u^{(j)}$$

де  $na$  - порядок моделі ( $na > nb$ );

$a_i$  та  $b_j$  - постійні коефіцієнти (параметри моделі);

$u^{(j)}(t)$  та  $y^{(i)}(t)$  - похідні відповідно вхідного і вихідного сигналів.

### 2. Рівняння змінних стану.

При виборі  $n$  координат системи (об'єкта) в якості змінних її стану (такими координатами, наприклад, можуть бути вихідний сигнал  $y(t)$  і  $n-1$  його похідних)  $x_i(t)$ ,  $i=1,2,\dots,n$ , цю систему можна описати рівняннями для змінних стану

$$\dot{X}(t) = AX(t) + Bu(t)$$

$$y(t) = CX(t) + Du(t),$$

де  $X(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T$  - вектор-стовпець змінних стану;  $A$ ,  $B$ ,  $C$  і  $D$  при скалярних  $u(t)$  і  $y(t)$  - відповідно матриця розміру  $n \times n$ , вектори розміру  $n \times 1$  і скаляр (при векторних  $u(t)$  і  $y(t)$  - матриці відповідних розмірів).

Наведені моделі є еквівалентними, тобто, знаючи будь-яку з них, можна отримати всі інші.

### 3. Разностные рівняння.

Для об'єктів, функціонування яких представляється для дискретного часу  $t_k = kT$  (в даному випадку  $T$  - інтервал дискретизації), тобто для дискретних об'єктів, найбільш загальним видом опису є різницеве рівняння (аналог диференціального)

$$y_k + a_1 y_{k-1} + \dots + a_m y_{k-na} = b_1 u_k + b_2 u_{k-1} + b_3 u_{k-2} + \dots + b_{nb} u_{k-nb+1}$$

де  $y_{k-i} = y[(k-i)T]$ ,  $u_{k-j} = u[(k-j)T]$ .

### 4. Z- перетворення.

Зв'язок між сигналами може бути відображена також через дискретну згортку

$$y_k = \sum_{i=0}^k g_i u_{k-i}$$

де  $g_i$  – ординати ваговий частоті функції об'єкта, або, з використанням апарату Z-перетворення

$$Y(z) = \sum_{k=0}^{\infty} y_k z^{-k}$$

де  $z = e^{pT}$ , через дискретну передавальну функцію

$$W(z) = \frac{Y(z)}{U(z)} = \frac{B(z)}{A(z)},$$

яка визначається на підставі різницевого рівняння після застосування до обох частин цього рівняння Z-перетворення

$$(1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{na} z^{-na}) Y(z) = (b_1 + b_2 z^{-1} + b_3 z^{-2} + \dots + b_{nb} z^{-nb+1}) U(z)$$

Оператор  $z^{-1} = e^{-pT}$  є оператор затримки, тобто

$$z^{-1} u_k = u_{k-1}, z^{-2} u_k = u_{k-2} \text{ та тодіб.}$$

Відзначимо, що на практиці в більшості випадків вимір безперервних сигналів проводиться в дискретні моменти часу, що представляє певну зручність при подальшій обробці даних на комп'ютері. Безперервні об'єкти можна наближено відображати дискретними моделями. При цьому для переходу від безперервних моделей до дискретним, застосовується або Z-перетворення, або заміна похідних в диференціальному рівнянні, що описує безперервний об'єкт, кінцевими різницями (цей підхід дає прийнятну точність тільки при малих інтервалах дискретизації і невеликому порядку рівняння).

Нижче наведено кілька поширених дискретних моделей дискретних об'єктів для тимчасової області, які враховують дію шуму спостереження (моменти дискретного часу позначені тим же символом  $t$ , що і безперервне час, тут  $t = 0, 1, 2 \dots$ ).

#### 5. Моделі авторегресії.

Модель авторегресії AR (AutoRegressive) вважається найпростішим описом

$$A(z) y(t) = e(t),$$

де

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{na} z^{-na}.$$

Складніша - ARX-модель (AutoRegressive with eXternal input)

$$A(z) y(t) = B(z) u(t) + e(t)$$

або в розгорнутому вигляді

$$y(t) + a_1 y(t-1) + \dots + a_{na} y(t-n) = b_1 u(t) + b_2 u(t-1) + \dots + b_{nb} u(t-m) + e(t).$$

Тут і нижче  $e(t)$  – дискретний білий шум,  $B(z) = b_1 + b_2 z^{-1} + \dots + b_{nb} z^{-nb+1}$ .

ARMAX- модель (AutoRegressive-MovingAverage with eXternal input) - модель авторегресії змінного середнього

$$A(z) y(t) = B(z) u(t - nk) + C(z) e(t)$$

де  $nk$  - величина затримки (запізнювання),

$$C(z) = 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + b_{nc} z^{-nc}.$$

Модель «вхід-вихід» (в англійських джерелах така модель називається «Output-Error», тобто «вихід-помилка», скорочено OE)

$$y(t) = \frac{B(z)}{F(z)} u(t - nk) + e(t)$$

де  $F(z) = 1 + f_1 z^{-1} + f_2 z^{-2} + \dots + f_{nf} z^{-nf}$ .

Модель Боксу-Дженкіс (BJ)

$$y(t) = \frac{B(z)}{F(z)} u(t - nk) + \frac{C(z)}{D(z)} e(t)$$

де  $D(z) = 1 + d_1 z^{-1} + d_2 z^{-2} + \dots + d_{nd} z^{-nd}$ .

Всі ці моделі можна розглядати як окремі випадки узагальненої параметричної лінійної структури

$$A(z)y(t) = \frac{B(z)}{F(z)}u(t - nk) + \frac{C(z)}{D(z)}e(t)$$

при цьому всі вони допускають розширення для багатовимірних об'єктів (що мають кілька входів і виходів).

#### 6. Модель для змінних стану (State Space).

Вона має вигляд

$$\begin{aligned}x(t+1) &= Ax(t) + Bu(t) + Ke(t), \\ y(t) &= Cx(t) + Du(t) + v(t)\end{aligned}$$

де A, B, C, D - матриці відповідних розмірів,  $v(t)$  - корельований шум спостереження.

#### 1.2 Дослідження лінійних стаціонарних систем

В системі MatLab для дослідження лінійних стаціонарних систем використовується пакет Control System Toolbox. Пакет реалізує методи дослідження динамічних систем, засновані на використанні передавальних функцій і моделей змінних стану. Введення моделі в середу пакету можливий в трьох формах - у формі матриць простору стану, у вигляді коефіцієнтів чисельників і знаменників передавальних функцій і в формі завдання нулів, полюсів і коефіцієнта передачі системи. Треба відзначити, що тут лінійні, інваріантні в часі системи з постійними параметрами (в російській версії ЛСС - лінійні стаціонарні системи) названі LTI-об'єктами (Linear Time Invariant).

Деякі процедури створення LTI-моделей:

- *ss* - створює модель простору стану по заданих матрицях рівнянь стану системи;
- *dss* - створює аналогічну модель за описом простору стану більш загального вигляду, коли рівняння змінних стани не дозволені щодо похідних;
- *tf* - створює модель за заданими передавальним функціям системи;
- *zpk* - створює модель за заданими нулями, полюсами і коефіцієнтам передачі системи;
- *filt* - створює модель по дискретним передавальним функціям, записаним у формі поліномів від  $z^{-1}$ ;
- *set* - привласнює значення деяким іншим полям об'єктів (назв входів і виходів, назвам системи і т.д.).

Зазначені процедури дозволяють створювати як безперервні, так і дискретні моделі. Крім цього існує інструментарій перетворення моделей з однієї зазначеної форми в іншу.

Зручним засобом перегляду тимчасових і частотних характеристик моделі є інтерактивний оглядач LTI-Viewer. Для виклику графічного інтерфейсу з командного вікна використовується функція *ltiview*, при цьому на екрані з'являється робоче вікно оглядача. Для обміну даними між робочими середовищами MatLab і LTI-Viewer використовуються команди меню вікна оглядача *File/Import* і *File/Export*. Після завантаження моделі у вікні оглядача відобразиться графік перехідної функції моделі. За допомогою оглядача легко обчислюються і відображаються у вигляді графіків і діаграм частотні і тимчасові характеристики, нулі і полюси системи.

#### Хід виконання практичного заняття

У командному вікні MatLab введіть вектора коефіцієнтів чисельника *num* і знаменника *den* передавальної функції об'єкта першого порядку.

Використовуючи команду *model=tf(num, den)* створіть модель об'єкта.

Перетворіть безперервну модель в дискретну

$$disc=c2d(model, 0.01),$$

тут 0.01 - крок дискретизації і назад в безперервну

$$cont=d2c(disc).$$

Переконайтеся, що при зворотному перетворенні отриманий вихідний варіант моделі. В результаті цих дій в робочій області збережені три моделі: *model*, *disc*, *cont*.

Завантажте файл з демонстраційними LTI-моделями, набравши в командному рядку *load LTIexamples*.

У командному рядку наберіть *ltiview*. Виберіть у вікні *File/Import*. Відкриється діалогове вікно вибору моделі для завантаження. Ваші моделі теж будуть в цьому списку.

Завантажте свою модель *model*. З'явиться вікно з графіком перехідної функції.

Використовуючи покажчик миші, клацання мишею на графіку, контекстні меню, виконайте різні операції (з огляду на їх очевидності, опис цих операцій не наводиться).

Активізуйте меню *Edit/Plot Configuration*, це призведе до появи діалогового вікна, що дозволяє перейти до групи цікавлять характеристик для їх відображення. Можливий вибір до 6 характеристик одночасно. На рисунку 1.1 показані чотири характеристики.

Використовуючи контекстні меню, встановіть: написи на графіку; розміри графіка; стиль графіка і інші можливі властивості.

Виконайте наступні дії:

- вибором меню *File/Toolbox* встановіть параметри інструментарію;
- вибором меню *Preferences File/LineStyle* встановіть стиль графічного зображення;
- вибором меню *Preferences File/LTI Viewer Preferences* встановіть параметри векторів часу і частот.

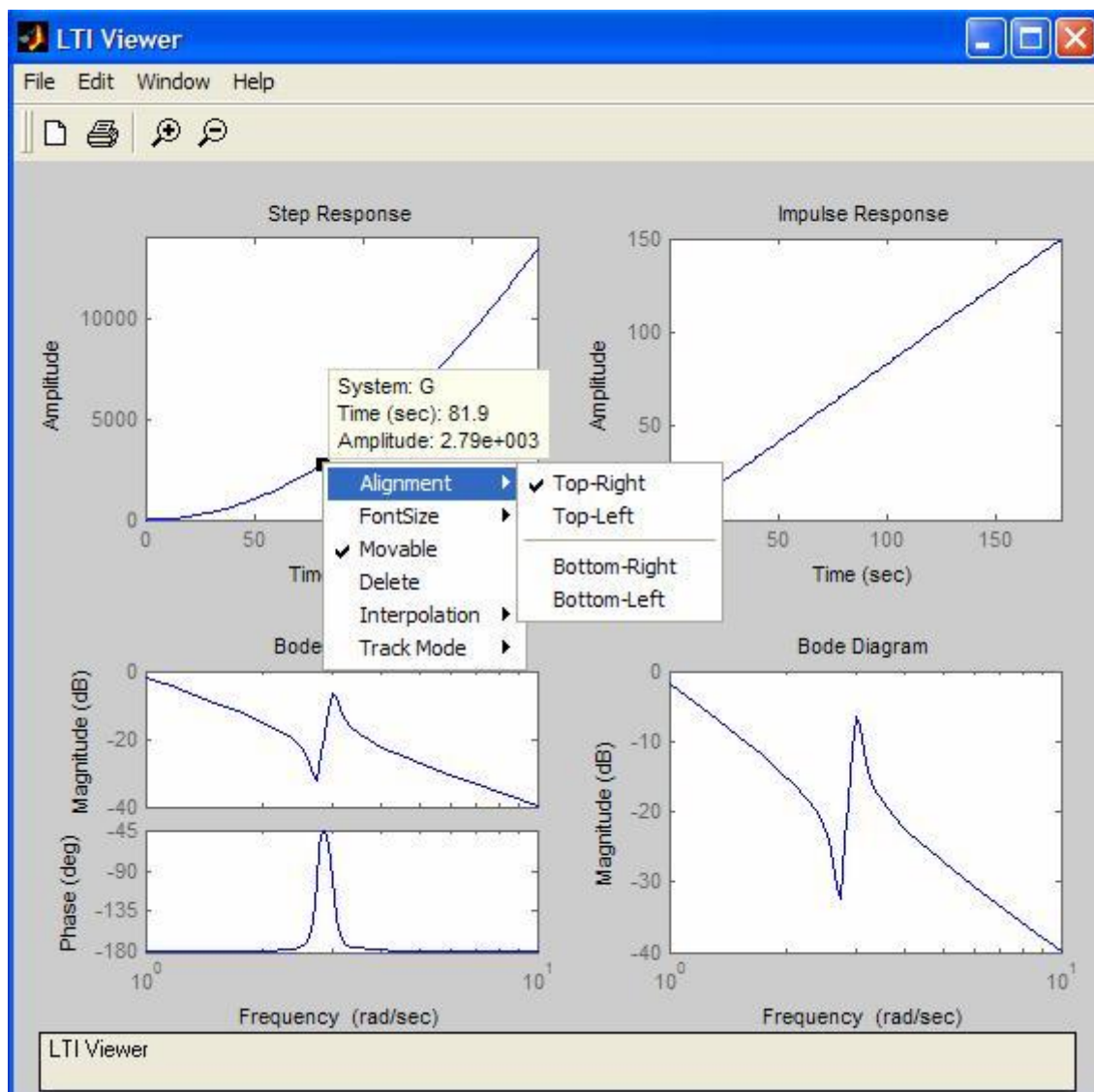


Рисунок 1.1 - Графіки перехідної і частотних характеристик

**Варіанти завдань:** Параметри передавальної функції об'єкта першого порядку вибираються студентом самостійно.

### Контрольні питання

- 1 Що собою являє система Matlab?
- 2 Назвіть основні елементи командного вікна системи Matlab.
- 3 Які режими роботи в цій системі ви вивчили?
- 4 Що є основним об'єктом системи?
- 5 Які зарезервовані змінні в системі Matlab ви знаєте?



- 6 Призначення пакета Control System Toolbox?
- 7 Що таке LTI-об'єкт?
- 8 Перерахуйте відомі вам процедури створення LTI-моделей.
9. Перерахуйте відомі вам процедури перетворення моделей.

## Практичне заняття № 2. Аналітичні методи моделювання об'єкта з зосередженими параметрами

**Мета роботи:** прищеплення навичок застосування основних законів збереження при моделюванні об'єктів з зосередженими параметрами; освоєння прийомів моделювання та аналізу об'єктів в пакеті Simulink.

### 2.1 Опис процесу

Об'єктом дослідження є млин, в яку надходить вхідний потік вихідного матеріалу  $\Phi_0$ , що переробляється, що містить великі фракції. Вихідний потік з млина  $\Phi_{\text{вих}}$  пропорційний масі матеріалу в млині  $M$

$$\Phi_{\text{вих}} = \alpha M,$$

де  $\alpha$  - константа пропорційності.

У млині відбувається подрібнення великої фракції зі швидкістю  $W$ , пропорційній масі великої фракції

$$W = k(M)(1 - C)M,$$

де  $C = \frac{m}{M}$ ,  $k(M) = \frac{k_0(2M^+ - M)}{M^+}$ .

Тут  $m$  - маса дрібної фракції в млині;  $C$  - частка дрібної фракції;  $k_0$  (1/год) - константа швидкості подрібнення;  $M^+$  - характерна маса, відповідна максимальній швидкості подрібнення (максимальної продуктивності млина). Значення  $M^+ = 10m$ ;  $k_0 = 1,5$  (1/год).

Математична модель процесу у вигляді системи диференціальних рівнянь виводиться на основі аналітичних методів моделювання. При складанні рівнянь математичної моделі процесу використовуються наступні рівняння балансу:

а) баланс загальної маси в млині

$$\frac{dM}{dt} = \Phi_0 - \Phi_{\text{вих}};$$

б) баланс маси дрібних фракцій (готового продукту) в млині

$$\frac{d(CM)}{dt} = W - C\Phi_{\text{вих}}.$$

Система диференціальних рівнянь, що описують даний процес має вигляд:

$$\begin{aligned} \frac{dM}{dt} &= \Phi_0 - \Phi \\ \frac{dm}{dt} &= W - \alpha^* m \end{aligned}$$

У лабораторній роботі досліджується продуктивність млина як функція вихідного потоку  $P = C\Phi_{\text{вих}}$ .

### 2. 2 Моделювання систем в пакеті Simulink

Пакет моделювання динамічних систем Simulink призначений для математичного моделювання лінійних і нелінійних динамічних систем і пристроїв, представлених своєї функціональної блок-схемою, що називається моделлю.

Для побудови функціональної блок-схеми модельованих пристроїв пакет Simulink системи MatLab має велику бібліотеку блокових компонентів і зручний редактор блок-схем. Введення характеристик досліджуваних систем проводиться в діалоговому режимі, шляхом графічної збірки схеми з'єднань стандартних елементарних ланок. В результаті такої збірки утворюється модель

досліджуваної системи, яку називають S-моделлю. Модель зберігається у файлі з розширенням *.mdl*.

Simulink автоматизує наступний найбільш трудомісткий етап моделювання: він становить і вирішує складні системи алгебраїчних і диференціальних рівнянь, що описують задану функціональну схему (модель), забезпечуючи зручний і наочний візуальний контроль за поведінкою створеного користувачем віртуального пристрою.

Щоб запустити пакет необхідно з командного вікна MatLab вибрати команду *File/New/Model* або натиснути відповідну кнопку на панелі інструментів. При цьому відкривається вікно Simulink Library Browser з переліком розділів бібліотеки Simulink. Вибравши в цьому вікні *File/New/Model* створюємо порожнє вікно *untitled* - заготовку для створення нової моделі. Вид екрану наведено на рисунку 2.1.

Для побудови S-моделі використовуються блоки з бібліотеки Simulink. Бібліотека блоків Simulink є набором візуальних об'єктів, використовуючи які можна збирати, як з кубиків, довільну конструкцію. Для будь-якого блоку можна отримувати необхідну кількість копій і використовувати кожен з них автономно. Більш того, практично для всіх блоків існує можливість індивідуальної настройки: користувач може змінити як внутрішні параметри блоків (наприклад, кількість входів), так і зовнішнє оформлення (розмір, колір, ім'я та т. Д.). На порядок з'єднання блоків між собою теж не накладається ніяких обмежень. Звичайно, при зв'язуванні блоків необхідно дотримуватись певних правил, які обумовлені в основному логікою роботи самої моделі, а не спеціальними вимогами Simulink.

Склад бібліотеки може бути поповнений призначеними для користувача блоками.

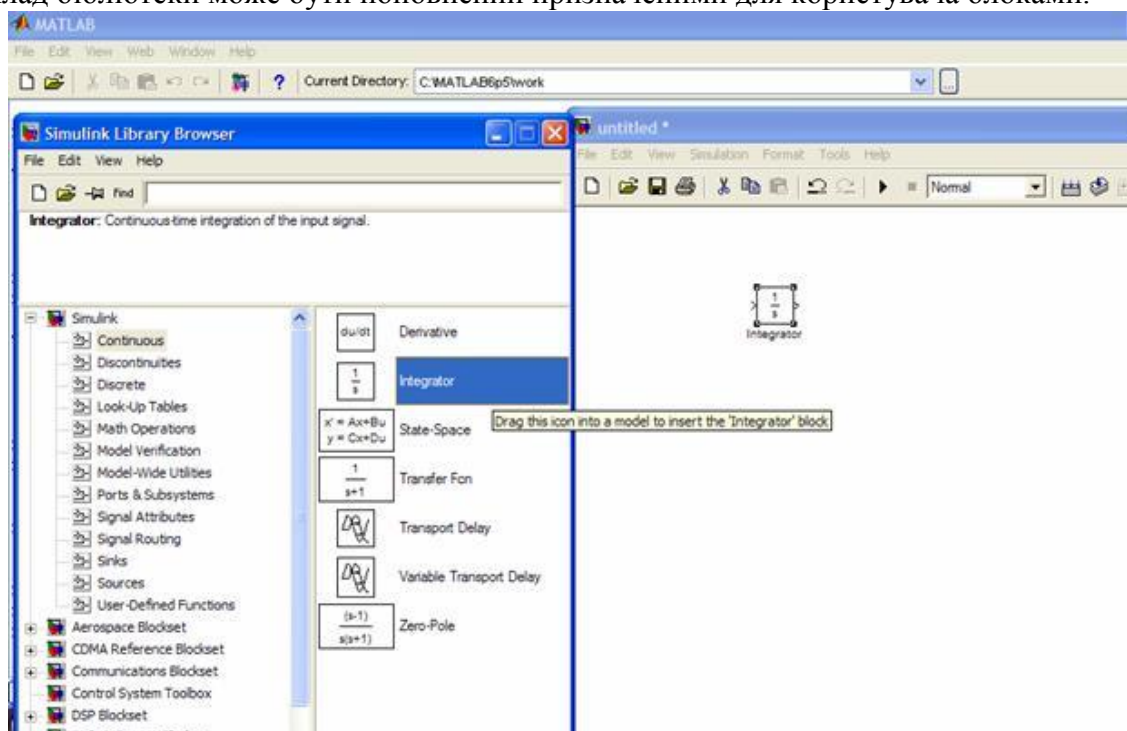


Рисунок 2.1 - Вікно пакету Simulink для створення нової моделі

### 2.3 Реалізація моделі процесу

Модель процесу являє собою систему диференціальних рівнянь другого порядку для визначення мас великої  $M(t)$  і дрібної фракцій  $m(t)$ . Для перетворення цих рівнянь в блок-схеми використовується наступна ідея. Спочатку формуються праві частини рівнянь, вважаючи, що  $M$  і  $m$  відомі, в результаті виходять  $dM/dt$  і  $dm/dt$ . Про інтегрувавши ці вирази, отримуємо значення  $M(t)$  і  $m(t)$ . Отримані величини тепер можна використовувати для формування правих частин рівнянь (зворотним зв'язком). Отже, формування блок-схеми складається з наступних кроків:

а) в основу блок-схеми кожного рівняння покладено блок Integrator, на його вхід подаються величини, отримані в результаті розрахунку правих частин рівнянь; виходи цих блоків представляють шукані значення  $M(t)$  і  $m(t)$ ;

б) сформувати окремим блоком у вигляді підсистеми (блок *Subsystem*) праву частину першого рівняння, використовуючи в якості вхідних процесів отримані значення  $M(t)$ , а також функцію вхідного потоку  $\Phi_0$ ; друга підсистема (для другого рівняння) в якості вхідних процесів використовує отримані значення  $M(t)$  і  $m(t)$ .

У підсистемах для формування виразів правих частин рівнянь використовується блок *fcn*. Цей блок має тільки один вхідний порт. Якщо треба подати на вхід блоку кілька сигналів, їх треба об'єднати в один вектор блоком *Mux*. Вхідний сигнал (скаляр або вектор) блоку *fcn* позначається буквою  $u$ , якщо цей сигнал є вектором, то його компоненти позначаються  $u(1)$ ,  $u(2)$ , ..., нумерація компонентів вектора в тому порядку, в якому вони подані на блок *Mux*; вираз у вікні настройки блоку формується за правилами мов програмування;

в) використовуючи створені підсистеми, зв'язати сформовані процеси  $M(t)$  і  $m(t)$  з відповідними входами підсистем, а вихід підсистем зв'язати з вікном перегляду результатів моделювання *Scope*;

г) вихід другої підсистеми  $m(t)$  пов'язати з блоком *Gain* і отримати досліджувану продуктивність  $P = C \Phi_{\text{вх}} = \alpha * m(t)$ ; відобразити  $P$  в оглядовому вікні *Scope*;

д) тепер необхідно сформувати вхідний потік; тут також використовується блок *fcn*, в якому формується вираз для вхідного потоку (по варіанту); так як вхідний потік заданий як явна функція часу, то на вхід цього блоку *fcn* треба подати сигнал з блоку *Clock*.

### Хід виконання практичного заняття

У пакеті Simulink системи MatLab побудувати блок-діаграму динамічної моделі процесу. Зберегти файл з моделлю.

Провести імітаційні експерименти на моделі (за варіантом). Знайти найбільш продуктивні режими.

#### Варіанти завдань:

У варіантах 1-8 коефіцієнт  $\alpha = 0.5$  (1 / год).

Дослідити продуктивність млина як функцію вхідного потоку  $\Phi_0(t) = a * \exp(-bt)$ . Знайти найбільш продуктивні режими.

Дослідити продуктивність млина як функцію вхідного потоку  $\Phi_0(t) = a * \exp(bt)$ . Знайти найбільш продуктивні режими.

Дослідити продуктивність млина як функцію вхідного потоку  $\Phi_0(t) = a_1 * t^2 + a_2 t + a_3$ . Знайти найбільш продуктивні режими.

Дослідити продуктивність млина як функцію вхідного потоку  $\Phi_0(t) = a_1 t + a_2$ . Знайти найбільш продуктивні режими.

Дослідити продуктивність млина як функцію вхідного потоку  $\Phi_0(t) = \sqrt{a_1 t + a_2}$ . Знайти найбільш продуктивні режими.

Дослідити продуктивність млина як функцію вхідного потоку  $\Phi_0(t) = a_1 / (a_2 t + a_3)$ . Знайти найбільш продуктивні режими.

Дослідити вплив на продуктивність млина частоти  $\omega$  періодично змінюється вхідного потоку  $\Phi_0(t) = \Phi_0^* + 2.5 \cos(\omega t)$ , де  $\Phi_0^* = 5$  т/час. Потрібно знайти найбільш «небезпечні» частоти коливань вхідного потоку.

Дослідити вплив на продуктивність млина амплітуди  $A$  періодично змінюється вхідного потоку  $\Phi_0 = \Phi_0^* + A \cos(0.2t)$ , де  $\Phi_0^* = 5$  т/час. Потрібно знайти амплітуди найбільш «небезпечних» коливань вхідного потоку.

Дослідити вплив на продуктивність млина частоти  $\omega$  періодично змінюється коефіцієнта  $\alpha$ :  $\alpha = \alpha_0 + 0.25 \cos(\omega t)$ , де  $\alpha_0 = 0.5$  (1/час). Вибрати оптимальний режим зміни  $\alpha$ .

Дослідити вплив на продуктивність млина амплітуди  $\alpha_1$  періодично змінюється коефіцієнта  $\alpha$ :  $\alpha = \alpha_0 + \alpha_1 \cos(0.2t)$ , де  $\alpha_0 = 0.5$  (1/час). Вибрати оптимальний режим зміни  $\alpha$ .

#### Контрольні питання

1. Які основні типи математичних моделей?

2. У чому відмінність аналітичних методів моделювання від

експериментальних?

3. Які закони збереження використовуються для виведення рівнянь моделі розглянутого процесу?
4. До якого класу належить модель розглянутого процесу?
5. Як по виду моделі визначити, чи є вона динамічною?
6. Поясніть призначення пакета *Simulink*?
7. Що собою представляє підсистема в *Simulink*?
8. Поясніть призначення блоків *Clock*, *Integrator*, *fcn*, *Gain*?
9. Які варіанти перегляду результатів експериментів на моделі ви знаєте?
10. Який буквою позначається вхідний сигнал блоку *fcn*?

### Практичне заняття №3. Моделювання об'єкта регулювання рівня рідини в резервуарі

**Мета роботи:** освоєння пакета подієвого моделювання *Stateflow* системи *MatLab*.

#### 3.1 Опис об'єкта

Об'єктом дослідження є резервуар з незалежним  $G_n(t)$  припливом рідини і залежним стоком  $G_c(t)$ . Потік на виході визначається величиною рівня рідини над зливним отвором  $H$  і площею прохідного перерізу зливного отвору  $f_c$ . Кількість рідини в резервуарі, що знаходиться над площиною зливного отвору, визначається зі співвідношення

$$M = F \cdot H \cdot \rho,$$

де  $F$  – площа поперечного перерізу резервуара,  $\rho$  - щільність.

При складанні рівнянь математичної моделі об'єкта використовуються:

- рівняння матеріального балансу системи

$$\frac{dM(t)}{dt} = G_n(t) - G_c(t); \quad (3.1)$$

- рівняння, що відображає закон збереження руху

$$G_c(t) = \mu \cdot f_c \cdot \sqrt{2 \cdot g \cdot \rho \cdot H}. \quad (3.2)$$

Прийmemo за координати стану системи рівень рідини в резервуарі  $H$  і потік на виході  $G_c$ .

Статична модель системи матиме вигляд

$$H = \frac{1}{2g\rho\mu^2} \cdot \frac{G_n^2}{f_c^2},$$

$$G_c = G_n = const. \quad (3.3)$$

Динаміка системи описується наступним диференціальним рівнянням:

$$\frac{dH(t)}{dt} = \frac{1}{F\rho} \left[ G_n(t) - \mu \sqrt{2g\rho} \cdot \sqrt{H(t)} \cdot f_c(t) \right], \quad (3.4)$$

тут для  $G_c$  залишається справедливим співвідношення (3.2).

У лабораторній роботі потрібно в різні часові відрізки загального інтервалу моделювання провести розрахунки по статичній (3.3) або динамічній (3.4) моделям. Перехідні процеси в системі виникають в результаті зміни вхідних змінних  $G_n(t)$  і  $f_c$ .

Для перемикання між цими моделями, а також для передачі нових значень параметрів в модель необхідно використовувати підсистему *Stateflow*.

#### 3.2 Підсистема *Stateflow*

Основне призначення пакету *Simulink* полягає в здійсненні імітаційного моделювання систем. При цьому вплив зовнішніх подій (ситуацій) на поведінку систем не враховується або враховується на рівні тимчасових залежностей джерел впливу. Тим часом багато систем працюють в умовах, коли виникнення тих чи інших подій може істотно вплинути на їх роботу. Моделювання систем, поведінка яких залежить від зовнішніх подій (ситуацій) називається подієвим або ситуаційним. У пакеті *Simulink* системи *MatLab* таке моделювання реалізується за допомогою

спеціального розширення *Stateflow*. Пакет має свій графічний інтерфейс користувача, що дозволяє створювати графічні діаграми, звані SF-діаграмами. Ці діаграми служать для візуального представлення роботи модельованої системи. Це досягається аналізом всіх стадій її роботи із зазначенням активних і пасивних в даний час блоків і переходів між ними за результатами аналізу тих чи інших умов.

Для створення *Stateflow*-діаграми необхідно спочатку створити модель в *Simulink* або відкрити вже існуючу. Потім потрібно вибрати в бібліотеці блок, що позначає діаграму *Stateflow*, і перенести його в свою модель. Після подвійного клацання по ньому мишею, відкриється вікно *Stateflow (chart)* (рисунок 3.1), в якому, користуючись меню і панеллю інструментів, можна описувати стану, зв'язку між ними, визначати змінні, умови перемикавання та інші, необхідні для управління елементи.

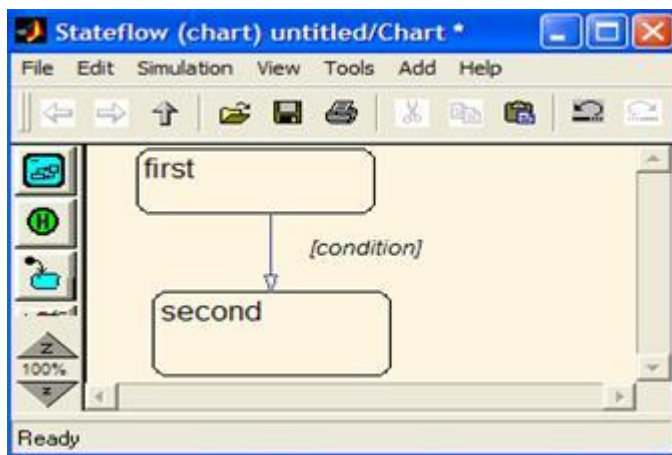


Рисунок 3.1 - Приклад простий SF-діаграми

На діаграмі *Stateflow* розрізняють дві основні групи елементів: графічні та неграфічні. Всі графічні елементи діаграми представлені на лівій панелі основного вікна графічного редактора карт станів.

Важливим об'єктом SF-діаграми є стан (*state*). Кожне стан описує один режим роботи подієво-керованої системи. Стан стає активним, якщо отримує значення "істина" умова переходу, що веде до нього, або якщо цей стан є початковим. Кожне стан на діаграмі має свого батька і може мати нащадків. Якщо стан є єдиним, то його батьком є сама SF-діаграма (коренева діаграма).

Для створення графічного образу стану необхідно, вибравши на панелі відповідну кнопку, натиснути кнопку миші в тому місці, де його необхідно розмістити.

Стан має текстові мітки, які визначають дії, що виконуються під час його активності. Ім'я стану вводиться першим. При описі самого стану можуть бути визначені наступні дії:

- *name* – ім'я стану;
- *entry* – дія, що виконується на вході в стан;
- *during* - дія, що виконується, поки стан активно;
- *exit* – дія, що виконується на виході зі стану;
- *on*: {ім'я події} - дія, що виконується в момент появи події (назва якого є в кутових дужках) за умови, що система буде перебувати в даному стані.

Ще один важливий графічний об'єкт SF-діаграми - переходи (*Transition*). Переходи відображають зв'язок одного об'єкта з іншим і представляються зазвичай стрілками, початок яких відноситься до стану-джерела, а кінець - до стану-адресату. Переходи не мають своєї кнопки на панелі інструментів. Для того щоб намалювати перехід, досить встановивши миша у стану-джерела, натиснути її ліву кнопку і, утримуючи її, протягнути з'явилася стрілку до стану-адресата. Переходи мають мітки, що описують умови, при яких відбувається перехід від одного стану до іншого. Текст мітки мають такий вигляд

*Event [condition] {condition\_action}/transition\_action*

де *Event* – визначає події, які ініціює перехід. Якщо подія не вказано, то перехід ініціюється при виконанні булевого вираження. Якщо подій, що ініціюють цей перехід, кілька, то вони все вказуються, при цьому поділяються оператором OR;

*condition* – булевское вираз, що ініціює перехід, стаючи справжнім;

*condition\_action* – дія, що виконується після того, як стало справжнім умова переходу, але до того, як весь перехід визначився істинним (і визначилося стан-адресат);

*transition\_action* – визначає дії, що здійснюються при переході, якщо всі описані раніше умови істинні і вже визначено стан-адресат.

Найважливіше поняття пакета *Stateflow* - *події (Event)* – це неграфічні об'єкти, керуючі діаграмою. Для візуалізації події можна використовувати мітки переходів. Всі події мають відношення до діаграми *Stateflow*, повинні бути визначені. Розрізняють такі види подій: вхідні; вихідні; локальні; імпортовані з джерела, зовнішнього щодо діаграм *Stateflow* і *Simulink*; експортовані адресату, зовнішньому щодо діаграм *Stateflow* і *Simulink*.

Редактор SF-діаграми має меню *Add/Event*, за допомогою якого можна задати тип події і вказати його властивості.

Дані - це неграфічні об'єкти на діаграмі *Stateflow*, призначені для зберігання числових даних. Змінні можна використовувати на будь-якому рівні ієрархії. Для створення і модифікації даних слід використовувати пункт меню *Add/Data*.

### Хід виконання практичного заняття

У пакеті *Simulink* системи *MatLab* побудуйте блок-діаграми статичної та динамічної моделі об'єкта, оформіть їх як підсистеми.

Підключіть блок *Chart* до моделі в *Simulink*.

Розробіть SF-діаграму для проведення імітаційних експериментів на моделі:

- тимчасові інтервали  $(t_0 - t_1)$ ,  $(t_2 - t_3)$ ,  $(t_4 - t_5)$  відображають три стану рівноваги резервуара;
- інтервал  $(t_1 - t_2)$  відповідає перехідному процесу в резервуарі, викликаному зміною

припливу рідини  $G_n$  в момент  $t_1$ ;

- перехідний процес на інтервалі  $(t_3 - t_4)$  викликаний зміною прохідного перетину зливного отвору в момент  $t_3$ .

### Варіанти завдань:

№ варіанту	1	2	3	4	5	6	7	8	9	10
F	2.0	1.5	1.0	2.2	1.8	2.4	1.2	1.9	2.1	1.2
$f_c$	0.6	0.3	0.2	0.5	0.4	0.6	0.2	0.3	0.7	0.1
$G_n$	350	300	250	270	330	390	270	340	460	370

моменти часу  $t_0$ ,  $t_1$ ,  $t_2$ ,  $t_3$ ,  $t_4$ ,  $t_5$  вибираються студентом самостійно.

### Контрольні питання

1. До якого класу належить розроблена модель?
2. Які змінні об'єкта, що моделюється є входами моделі?
3. Які змінні об'єкта, що моделюється є виходами моделі?
4. Дайте визначення статичної та динамічної моделі.
5. Через що може виникнути перехідний процес в даному об'єкті?
6. Для чого призначений пакет *Stateflow*?
7. Що таке SF-діаграма?
8. Поясніть поняття «стан» в пакеті *Stateflow*.
9. Що таке переходи?
10. Як створити або змінити дані в пакеті *Stateflow*?

## Практичне заняття №4. Ідентифікація об'єкта за допомогою передавальної функції

**Мета роботи:** освоєння методів графічної ідентифікації лінійних динамічних об'єктів.

### 4.1 Графічний метод ідентифікації за допомогою перехідної функції

Перші реалізовані в системах управління методи ідентифікації були засновані на використанні частотних, східчастих і імпульсних впливів. Ці методи (прямі методи ідентифікації) вимагають спеціальних вхідних сигналів, а саме: східчастих сигналів для ідентифікації по

перехідній функції (ступінчастою перехідною функцією), імпульсних вхідних сигналів для ідентифікації по імпульсній перехідній функції і синусоїдальних вхідних сигналів з різними частотами для визначення частотної характеристики. З трьох типів вхідних сигналів, про які говорилося вище, ступінчастий вхідний сигнал є найбільш простим для застосування.

У багатьох випадках для визначення передавальної функції системи можна використовувати запис її перехідної функції. Такий спосіб застосовується до більшості типів лінійних систем (1 і 2 порядків і до аперіодичним системам вищого порядку). Найбільш коректно графічний метод ідентифікації з використанням перехідних функцій застосовується до процесів першого порядку.

Нехай дано графік перехідного процесу. У момент часу  $t_0 = 0$  вхідна величина  $x$  стрибком змінюється на величину  $a$ . Треба записати рівняння об'єкта. Шукане рівняння для об'єкта першого порядку має вигляд

$$T \frac{dy(t)}{dt} + y(t) = ka(t) \quad \text{або} \quad W(p) = \frac{k}{Tp+1}$$

Загальна вираз для вирішення цього рівняння

$$y(t) = k \cdot a \cdot \left(1 - e^{-\frac{t}{T}}\right)$$

Треба визначити параметри  $T$  і  $k$ . Це можна виконувати таким чином:

а) при  $t \rightarrow \infty$ , маємо  $y(t) = k \cdot a$ , тобто через ординату  $y(t)$  асимптоти (ордината асимптоти  $K=ka$ ) можна визначити  $k$ . Коефіцієнт  $k$  являє собою співвідношення між усталеною величиною вихідного сигналу системи і амплітудою вхідного сигналу.

При  $t = T$  функція  $y(t) = K(1 - e^{-1}) = K(1 - 0.37) = 0.63K$ . Постійна часу  $T$  системи першого порядку дорівнює відрізку часу, за яке перехідна функція досягає 63% своєї усталеної величини;

б) про диференціюємо рішення

$$\frac{dy(t)}{dt} = \frac{k \cdot a \cdot e^{-\frac{t}{T}}}{T}$$

тоді

$$\lim_{t \rightarrow 0} \frac{dy}{dt} = \frac{ka}{T} = \frac{b}{T} = \operatorname{tg} \alpha,$$

тут  $\alpha$  - кут нахилу дотичної до графіка функції при  $t = 0$ . Тоді  $T = \frac{b}{\operatorname{tg} \alpha}$ . Отже, величина  $T$  - відрізок осі, що дорівнює відстані від початку координат до точки, в якій дотична перетинається з асимптотою. Коефіцієнт  $k$  визначається, як описано вище.

#### Хід виконання практичного заняття

У вікно редактора моделей Simulink завантажте з папки *Work/Objects* свій «об'єкт» (за варіантом).

Встановіть в якості вхідного сигналу ступінчастий сигнал, графік вихідного сигналу відобразить в оглядовому вікні.

Використовуючи отриманий графік, визначте обома способами параметри передавальної функції об'єкта першого порядку.

За відновленим моделям, подаючи на їх входи той же ступінчастий сигнал, отримаєте вихідні сигнали (загальний вигляд вийшла діаграми наведено на рисунку 4.1).

Порівняйте графіки перехідних функцій «об'єкта» і моделей. Яка з моделей дає більш точний результат? З чим це пов'язано?

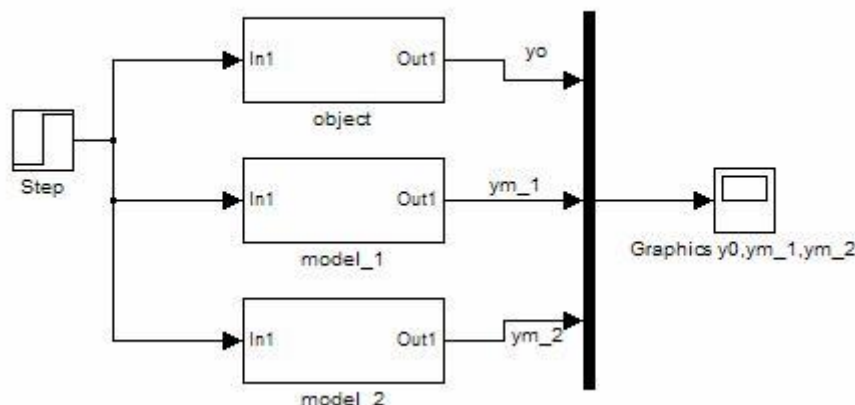


Рисунок 4. 1 - Блок-діаграма порівняння виходів об'єкта і моделей

**Варіанти завдань:** Вибираються з папки *Work/Objects* пакета Simulink.

#### Контрольні питання

1. Що таке ідентифікація?
2. Які методи називаються прямими методами ідентифікації?
3. Яка функція використовується для графічної ідентифікації лінійного об'єкта?
4. Який сигнал на вході об'єкта використовувався для проведення процедури ідентифікації?
5. Як провели дотичну до графіка перехідного процесу в оглядовому вікні Score?
6. Який блок використовується для того, щоб відобразити всі графіки в одному оглядовому вікні?
7. Який із способів ідентифікації дає більш точний результат? Чому?

### Практичне заняття № 5. Визначення параметрів узагальненої динамічної моделі одним з методів статичного оцінювання

**Мета роботи:** Оцінити параметри динамічної моделі за вхідними та вихідними даними методом найменшого квадрата, порівняти оцінки параметрів моделі.

#### Хід виконання практичного заняття

Проілюструємо застосування методу для вирішення задачі ідентифікації в разі апроксимації дослідних даних квадратичним поліномом  $y_m = a_0 + a_1x + a_2x^2$  при заданих коефіцієнтах регресії  $a_0=2,5$ ,  $a_1=-1,75$ ,  $a_2=5,06$

Таблиця 5.1

Параметр	Варіант (дорівнює номеру аспіранта по порядку в списку групи)									
	1	2	3	4	5	6	7	8	9	10
$a_0$	2,50	2,60	2,70	2,80	2,40	2,20	2,30	2,50	2,60	2,50
$a_1$	-1,20	-1,30	-1,40	-1,50	-1,60	-1,70	-1,80	-1,90	-2,00	-2,20
$a_2$	5,10	5,20	5,30	5,40	5,50	5,60	5,70	5,80	5,90	6,00
$a_3$	0,10	0,20	0,10	0,15	0,10	0,20	0,20	0,10	0,15	0,20

Критерій мінімуму середньоквадратичної помилки в цьому випадку визначається функціоналом

$$\min_{a_i} J(a_i) = \min_{a_0, a_1, a_2} \sum_{j=1}^N \left[ y_j - (a_0 + a_1 x_j + a_2 x_j^2) \right]^2$$

Система рівнянь для знаходження коефіцієнтів  $a_i$  набуває вигляду:



$$\begin{cases} a_0 N + a_1 \sum_{j=1}^N x_j + a_2 \sum_{j=1}^N x_j^2 = \sum_{j=1}^N y_j; \\ a_0 \sum_{j=1}^N x_j + a_1 \sum_{j=1}^N x_j^2 + a_2 \sum_{j=1}^N x_j^3 = \sum_{j=1}^N y_j x_j; \\ a_0 \sum_{j=1}^N x_j^2 + a_1 \sum_{j=1}^N x_j^3 + a_2 \sum_{j=1}^N x_j^4 = \sum_{j=1}^N y_j x_j^2. \end{cases}$$

Уявімо систему в матричному вигляді:

$$\begin{bmatrix} N & \sum_{j=1}^N x_j & \sum_{j=1}^N x_j^2 \\ \sum_{j=1}^N x_j & \sum_{j=1}^N x_j^2 & \sum_{j=1}^N x_j^3 \\ \sum_{j=1}^N x_j^2 & \sum_{j=1}^N x_j^3 & \sum_{j=1}^N x_j^4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^N y_j \\ \sum_{j=1}^N y_j x_j \\ \sum_{j=1}^N y_j x_j^2 \end{bmatrix}.$$

Рішенням системи є шукані вирази для коефіцієнтів рівняння регресії  $a_i$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} N & \sum_{j=1}^N x_j & \sum_{j=1}^N x_j^2 \\ \sum_{j=1}^N x_j & \sum_{j=1}^N x_j^2 & \sum_{j=1}^N x_j^3 \\ \sum_{j=1}^N x_j^2 & \sum_{j=1}^N x_j^3 & \sum_{j=1}^N x_j^4 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{j=1}^N y_j \\ \sum_{j=1}^N y_j x_j \\ \sum_{j=1}^N y_j x_j^2 \end{bmatrix}$$

Надалі, для зручності використання приймемо такі позначення

$$\begin{cases} S_1 = \sum_{j=1}^N x_j; & S_2 = \sum_{j=1}^N x_j^2; & S_3 = \sum_{j=1}^N x_j^3; & S_4 = \sum_{j=1}^N x_j^4; \\ S_5 = \sum_{j=1}^N y_j; & S_6 = \sum_{j=1}^N y_j x_j; & S_7 = \sum_{j=1}^N y_j x_j^2. \end{cases}$$

Відповідно до прийнятих позначеннями, вектор оцінок коефіцієнтів регресії  $a_x$  визначається як рішення наступної системи

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} n & S_1 & S_2 \\ S_1 & S_2 & S_3 \\ S_2 & S_3 & S_4 \end{bmatrix}^{-1} \begin{bmatrix} S_5 \\ S_6 \\ S_7 \end{bmatrix}$$

Наведемо програмну реалізацію розглянутого методу.

```
a0=2.5; % точні коефіцієнти регресії
```

```
a1=-1.75;
```

```
a2=5.06;
```

```
N=40;% розмір вибірки
```

```
x=10*normrnd(8, 2, [N 1]); % моделювання вхідного впливу
```

```
v=0.1*randn(N,1);% моделювання перешкоди у вигляді білого шуму
```

```
y=[a0+a1*x(1:N)+a2*x(1:N).^2+v(1:N)]; % моделювання вихідного сигналу з урахуванням
```

```
% перешкоди формування за вихідними даними підсумкових коефіцієнтів
```

```
s1=sum(x(1:N));
```

```

s2=sum(x(1:N).^2);
s3=sum(x(1:N).^3);
s4=sum(x(1:N).^4);
s5=sum(y(1:N));
s6=sum(y(1:N).*x(1:N));
s7=sum(y(1:N).*x(1:N).^2);
R=[N s1 s2; s1 s2 s3; s2 s3 s4]; % формування квадратної матриці даних
Y=[s5; s6; s7]; % формування вектора даних
betta=inv(R)*Y; % розрахунок оцінок по МНК
betta =% розраховані оцінки параметрів
2.5237
-1.7510
5.0600.

```

За результатами розрахунків видно, що оцінки параметрів, отримані в умовах зашумленості вихідних даних, мають досить задовільною точністю. Похибка отриманих оцінок може бути зменшена шляхом збільшення розміру вибірки, розширення діапазону вхідного сигналу і застосуванням згладжуючих процедур.

### Контрольні питання

1. Особливості різних видів математичного моделювання.
2. Ідентифікація моделі.
3. Імітаційне моделювання.
4. параметрична ідентифікація.
5. Методи ідентифікації.
6. Лінеаризація рівнянь.
7. Ідентифікація статичного об'єкта регресійний методом найменших квадратів.
8. Апроксимація експериментальних даних.
9. Рівняння регресії. Поліном.
10. Метод найменших квадратів.

## Практичне заняття № 6 Визначення параметрів регресійної моделі з використанням рекурентних методів оцінювання

**Мета роботи:** Оцінити параметри дискретної і безперервної моделі за вхідними та вихідними даними, ідентифікувати динамічний об'єкт явним і рекурентним методом найменшого квадрата, порівняти оцінки параметрів моделі.

### Хід виконання практичного заняття

Наведемо програмну реалізацію регресійної процедури оцінювання параметрів дискретної і безперервної моделей по вхідних і вихідних (незашумлених та зашумленими) даними для об'єкта другого порядку з коефіцієнтами  $k_0 = 25$ ;  $T_1 = 36$ ;  $T_2 = 15$ .

Таблиця 6.1

Параметр	Варіант (дорівнює номеру аспіранта по порядку в списку групи)									
	1	2	3	4	5	6	7	8	9	10
$k_0$	25	26	27	28	29	30	31	32	33	34
$T_1$	37	38	39	40	41	42	43	44	45	46
$T_2$	10	11	12	13	14	15	16	17	18	19

```

s1=tf([25],[36 15 1])% передавальна функція безперервної моделі об'єкта
T_end=60; % інтервал вимірювань
dt=0.2; % крок дискретизації
t=0:dt:T_end; % масив дискретного часу
N=length(t); % розмір вибірки

```

```

u=ones(N,1); % одиничне вхідний вплив
v=0.1*randn(N,1); % моделювання перешкоди (при обліку) у вигляді білого шуму
y=lsim(s1,u,t) %+v; % вихідна величина
% формування за вихідними даними підсумкових коефіцієнтів
S1=sum(y(1:N).^2);
S2=sum(y(2:N).*y(1:N-1));
S3=sum(y(1:N-1).^2);
S4=sum(y(3:N).*y(1:N-2));
S5=sum(y(2:N).*u(1:N-1));
S6=sum(y(2:N-1).*y(1:N-2));
S7=sum(y(1:N-1).*u(1:N-1));
S8=sum(y(1:N-2).^2);
S9=sum(y(1:N-2).*u(2:N-1));
S10=sum(u(1:N-1).^2);
A=[S3 S6 S7; S6 S8 S9; S7 S9 S10]; % формування квадратної матриці даних
B=[S2 S4 S5]'; % формування вектора даних
betta=inv(A)*B;% оцінки параметрів дискретної моделі
a1= betta(1);
a2= betta(2);
b= betta(3);
T1=dt^2/(1-betta(1)-betta(2)); % розрахунок параметрів безперервної моделі
T2=(betta(2)*T1+T1+dt^2)/dt;
K=betta(3)*T1/dt^2;
s2=tf([K],[T1 T2 1]);
y2=lsim(s2,u,t);
plot(t,y,t,y2,':'); % порівняння перехідних характеристик об'єкта і моделі
grid;
Розраховані оцінки параметрів дискретної моделі без урахування перешкоди:
      a1 = 1.9250;    a2 = -0.9260;    b = 0.0247;
Розраховані оцінки параметрів безперервної моделі без урахування перешкоди
      T1 = 40.3510;  T2 =15.1222;    K =24.9271;

```

Порівняння отриманих оцінок параметрів моделі з їх істинними значеннями при відсутності випадкових збурень показує високу точність оцінювання. Графічне зіставлення (рисунк 6.1) ідентифікованих і дійсних характеристик об'єкта показує практично точний збіг результатів при відсутності перешкод і задовільний відповідність вихідних сигналів об'єкта і моделі при зашумлених вхідних даних.

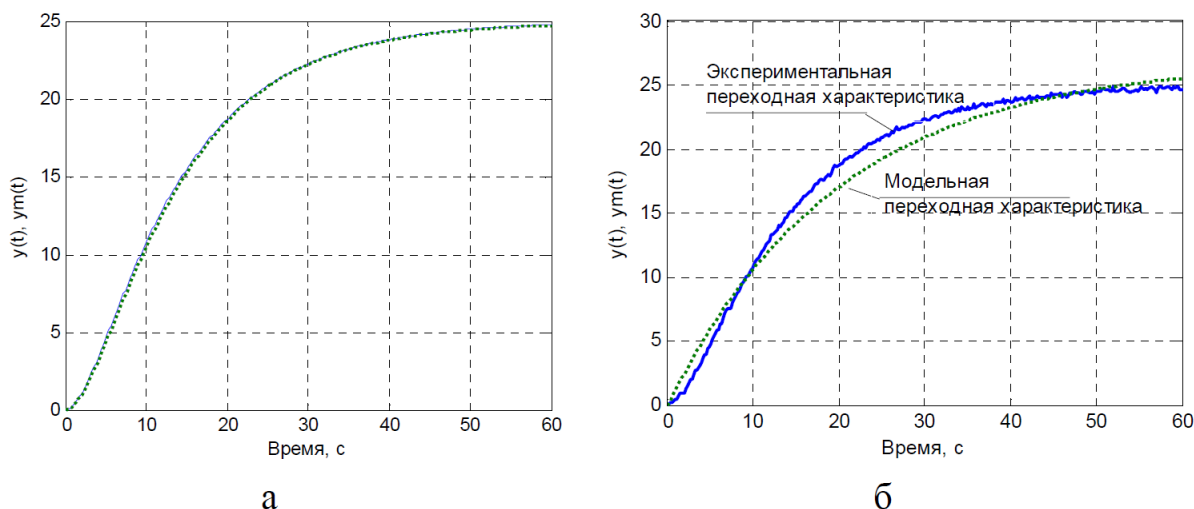


Рисунок 6.1 - Порівняння перехідних характеристик об'єкта і моделі, (а) в разі відсутності перешкоди; (Б) при обліку адитивної перешкоди на виході типу білий шум  $I = 0.1$

Що впливають на об'єкт перешкоди істотно впливають на точність оцінювання, і їх доцільно попередньо фільтрувати. Крім того, точність отриманих оцінок залежить від кроку дискретизації і обраного інтервалу вимірювань. При використанні МНК одержувані оцінки обчислюються з деякими помилками, які називаються зміщенням оцінок.

#### **Ідентифікації динамічного об'єкта явним МНК**

Розглянемо застосування явної форми МНК для параметричної ідентифікації тієї ж авторегресійної моделі зі змінним середнім (АРСС) об'єкта другого порядку, з урахуванням заданих порядків  $n=2$ ,  $m=1$ .

Наведемо програмну реалізацію явного МНК об'єкта другого порядку з передавальної функцією

$$W_o(p) = \frac{k_0}{T_1 p^2 + T_2 p + 1}$$

і коефіцієнтами  $k_0=25$ ;  $T_1=36$ ;  $T_2=15$ .

```
s1=tf([25],[36 15 1])% безперервна передавальна функція об'єкта
T_end=60;% інтервал вимірювань
dt=0.2;% крок дискретизації
t=0:dt:T_end;% масив дискретного часу
N=length(t);% розмір вибірки
u=ones(N,1);% моделювання одиничного вхідного впливу
y=lsim(s1,u,t);% моделювання вихідного впливу
n=2;% порядок об'єкта
R=[y(n:N-1) y(n-1:N-2) u(n:N-1)]; % формування розширеної матриці даних
Y=y(n+1:N); % формування вектора вихідних даних
betta=inv(R'*R)*R'*Y;
% розрахунок параметрів безперервної моделі
T1=dt^2/(1-betta(1)-betta(2))
T2=(betta(2)*T1+T1+dt^2)/dt
K=betta(3)*T1/dt^2
```

Розраховані оцінки параметрів дискретної моделі

$a_1 = 1.9190$ ;  $a_2 = -0.9200$ ;  $b = 0.0266$

Розраховані оцінки параметрів безперервної моделі

$T_1 = 37.5243$ ;  $T_2 = 15.2014$ ;  $K = 25.0000$

З отриманих результатів видно задовільна точність оцінювання параметрів. При цьому розрахункова практика показує, що метод чутливий до перешкод, їх доцільно фільтрувати.

#### **Ідентифікація динамічного об'єкта рекурентним МНК**

Наведемо програмну реалізацію оцінювання параметрів  $k_0$ ,  $T_1$ ,  $T_2$  об'єкта з попереднього прикладу за допомогою рекурентного МНК при використанні АРСС - моделі об'єкта другого порядку.

```
s1=tf([25],[36 15 1]) % безперервна передавальна функція об'єкта
T_end=60; % інтервал вимірювань
dt=0.2; % крок дискретизації
t=0:dt:T_end; % масив дискретного часу
N=length(t); % розмір вибірки
u=ones(N,1); % масив значень одиничного вхідного впливу
y=lsim(s1,u,t); % масив значень вихідного впливу
n=2; % порядок об'єкта
I=diag([1 1 1]);
i=1; % початковий крок
P=1000*I; % початкове наближення
betta=[0;0;0];
```

```

bet(i,:)=betta; % масив оцінок параметрів
% черговий крок обчислень
for i=n:N-1
R=[y(i+n-2:-1:i-1);u(i+n-2:-1:i)]'; % формування розширеної матриці даних
gamma=P*R'/(R*P*R'+1);
betta=betta+gamma*(y(i+1)-R*betta);
P=(I-gamma*R)*P;
bet(i,:)=betta;
end;
plot(bet,'+');
T1=dt^2/(1-betta(1)-betta(2)) % розрахунок параметрів безперервної моделі
T2=(betta(2)*T1+T1+dt^2)/dt
K=betta(3)*T1/dt^2

```

Оцінки параметрів безперервної моделі

$T1 = 35.6366$ ;  $T2 = 15.4333$ ;  $K = 25.0975$

З отриманих розрахункових результатів видно висока точність оцінювання всіх параметрів моделі. Розрахункова практика показує, що рекурентний МНК в порівнянні з його явною формою має кращу збіжність, і вимагає для досягнення тієї ж точності виконання меншої кількості кроків, і відповідно, обчислень. На рисунку 6.2 графічно представлені процеси збіжності оцінок параметрів для даної моделі.

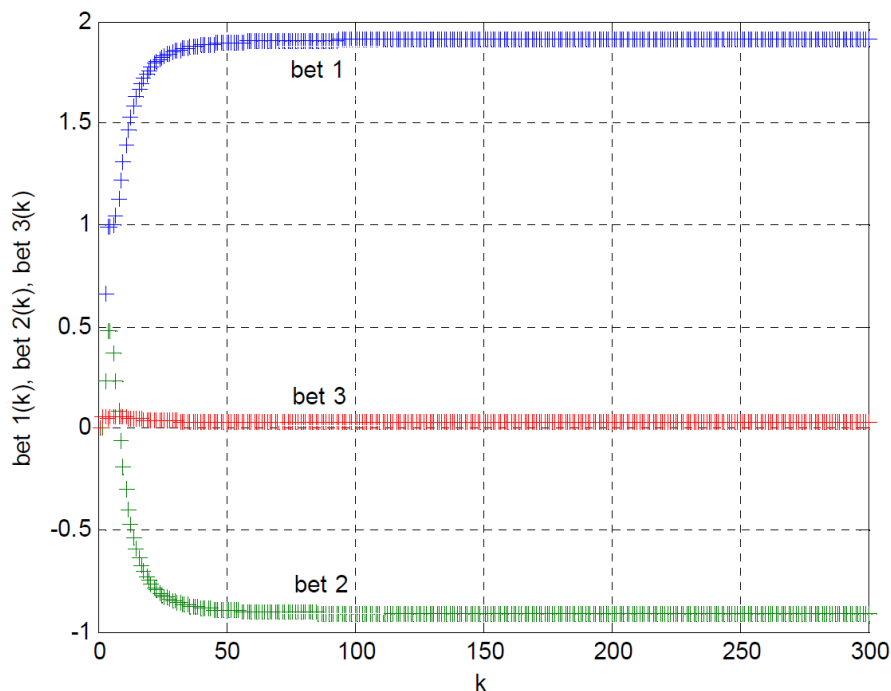


Рисунок 6.2 - Відповідність оцінок параметрів дискретної моделі

### Контрольні питання

1. Параметри дискретної і безперервної моделі.
2. Ідентифікація динамічного об'єкта.
3. Моделювання перешкоди.
4. Білий шум.
5. Види перешкод.
6. Перехідні характеристики об'єкта і моделі.
7. Рекурентний метод найменших квадратів.
8. Кореляція.
9. Передавальна функція об'єкта.

## Практичне заняття № 7. Параметрична ідентифікація лінійних систем

**Мета роботи:** освоїти методи параметричної ідентифікації систем, застосувати для процедури ідентифікації можливості командного вікна і графічного інструментарію ідентифікації системи *MatLab*.

### 7.1 Постановка завдання

В параметричному випадку модель визначається набором параметрів, які необхідно оцінити в процесі ідентифікації. Завдання параметричної ідентифікації можна сформулювати так: на основі експериментальних даних вказати значення вектора параметрів, при яких модель найкращим чином (або досить точно) в певному сенсі апроксимує оператор об'єкта.

Побудова моделі починається з вибору структури моделі, яка визначає взаємозв'язок спостережуваних даних через сукупність параметрів. Після цього формуються вхідні впливи і подаються на об'єкт, і вимірюються відгуки на ці дії (вихідні сигнали). Потім вхідні і вихідні сигнали і обрана структура використовуються для оцінки значень параметрів відповідно до прийнятого критерієм якості. Критерій якості ідентифікації характеризує ступінь адекватності моделі об'єкту в рамках узгоджених припущень і обмежень. Дуже часто використовується середньоквадратичний критерій, відповідно до якого шукаються такі оцінки параметрів, які забезпечують мінімальний середній квадрат різниці вихідних сигналів моделі і об'єкта при одному і тому ж вхідній дії. Оцінювання параметрів виконується на основі алгоритму ідентифікації, що визначає правила пошуку оцінок.

Нарешті, для того, щоб перевірити, наскільки точно побудована модель імітує або пророкує дані спостережень, необхідно порівняти їх при однакових впливах. Ця процедура називається верифікацією моделі.

### 7.2 Ідентифікація параметрів

У лабораторній роботі для ідентифікації лінійної системи, використовується графічна інтерактивна програма (*GUI - graphic user interface*) *ident*, що входить до складу інструментарію ідентифікації пакету *Matlab*.

Завантажте графічну програму ідентифікації, для чого в командному вікні необхідно набрати *ident*. В результаті завантаження на екрані з'явиться панель (головне вікно) програми *ident*.

Ідентифікація об'єкта проводиться наступним чином:

а) для завантаження даних експерименту виберіть спадне меню *Data-Import ...*; в діалоговому вікні введіть імена вхідних і вихідних змінних моделі; також можна ввести інтервал часу, текст коментаря. Потім натисніть кнопку *Import*, і дані відобразяться в іконці графічної оболонки. Закрийте вікно імпорту даних. Зверніть увагу, що дані також з'являться в полях *Working Data* та *Validation Data*;

б) прапорець *Time Plot* дозволяє відобразити вікно з графіками входів і виходів. Це вікно має меню для управління. Ви можете виділити мишею прямокутну область на графіку для збільшення фрагмента графіка, клацанням лівої кнопки миші можна збільшити / зменшити масштаб, подвійне клацання повертає вихідний масштаб. Щоб очистити вікно графіків клацніть на іконці з даними. Надалі, щоб працювати з новими даними, треба перемістити їх мишею в область *Working Data*.

Подвійне клацання на іконці даних викликає ще одне вікно, де можна змінити назву масиву даних;

в) команда *Select Range* меню, що розкривається *Preprocess* дозволяє вибрати підмножину з масиву даних. Це можна зробити через редаговані текстові поля вікна, що з'явилося або, виділяючи мишею прямокутну область на графіках сигналів. Натискання на *Insert* розміщує вибрані дані в новій іконці. Для верифікації моделі зазвичай використовуються дані, які не використовувалися для ідентифікації параметрів. Можна розділити вихідні масиви значень сигналів на дві половини, одна з яких використовується для ідентифікації, а друга - для підтвердження моделі. У реальній роботі, особливо якщо доводиться мати справу зі складними об'єктами, для верифікації використовуються кілька наборів різних даних.

Для ідентифікації вибрані дані переносяться в область *Working Data*, для верифікації моделі відповідні дані переносяться в область *Validation Data* (перетягуванням за допомогою миші);

г) для оцінювання параметричної моделі в спадному меню *Estimate* вибирається рядок *Parametric model*; задайте структуру моделі у вікні (поле *Structure*); відкрийте вікно для редагування порядку моделі (кнопка *Order Editor*), встановіть значення констант, потім щелкніть на кнопку *Estimate*. Результатом виконання останнього пункту є оцінки відповідних параметрів моделі. Для того, щоб ці оцінки відобразити в командному вікні Matlab, треба двічі натиснути правою клавішею миші по піктограмі моделі і у вікні, натиснути кнопку "*Present*".

### 7.3 Верифікація моделі

Для того, щоб переконатися, наскільки точно побудована модель описує об'єкт, необхідно проаналізувати її поведінку. Ця процедура називається підтвердженням або верифікацією моделі.

Для аналізу і підтвердження моделі програма *ident* надає користувачеві широкий набір засобів. Ці процедури виконуються за допомогою установки прапорців у вікнах, розташованих в лівій нижній частині панелі.

Для порівняння спостережуваних і модельованих вихідних сигналів встановіть прапорець "*Model output*". У вікні, представлені графіки виміряного (лінія чорного кольору) і розрахованого за моделлю вихідних сигналів. Для розгляду окремих фрагментів необхідно, утримуючи в натиснутому положенні ліву кнопку миші, виділити на графіку прямокутну область, яка буде відображена на все графічне вікно.

### Хід виконання практичного заняття

Завантажити в Simulink блок-діаграму «об'єкта» з папки *Work/Param* (за варіантом).

Визначити вхідні сигнали і подати їх на вхід об'єкта; розрахувати реакцію (вихідні сигнали).

Результати експерименту, тобто виміряні значення вхідних і вихідних сигналів передати в робочу область програми; для цього вивести сигнали в блоки *To Workspace*, в яких задати імена відповідних змінних.

Провести параметричну ідентифікацію об'єкта, тобто, оцінити невідомі параметри об'єкта за наявними реалізацій вхідних і вихідних сигналів. Оцінка проводиться по середньоквадратичного критерію, тобто в списку можливих моделей треба вибрати *Parametric Model*.

Для виконання процедури ідентифікації:

- набрати в командному вікні команду *ident*;
- у вікні командою *Data-Import* завантажити дані експерименту з робочою областю;
- виділити деякий діапазон експериментальних даних для ідентифікації об'єкта;
- виберіть структуру моделі - ARX, порядок моделі - перший, отримаєте оцінки параметрів моделі;
- проведіть верифікацію моделі, використовуючи інший діапазон експериментальних даних. Відобразіть параметри відновленої моделі в командному вікні програми.

За розрахованим параметрам зберіть блок-діаграму моделі.

Подайте на вхід моделі ті ж самі сигнали, що і на «об'єкті» і порівняйте відгуки об'єкта і моделі (в одному оглядовому вікні).

Проведіть аналіз отриманих результатів.

**Варіанти завдань:** Варіанти завдань вибираються з папки *Work/Param* пакета Simulink.

### Контрольні питання

- 1 Що таке параметричну ідентифікацію?
- 2 Для чого призначена графічна оболонка *ident*?
- 3 Що таке оцінка параметра?
- 4 Який критерій використовується для оцінювання параметрів лінійної моделі?
- 5 Що таке верифікація моделі?
- 6 Як можна в пакеті *ident* візуально оцінити правильність відновленої моделі?

## Практичне заняття № 8. Непараметрична ідентифікація динамічних об'єктів

**Мета роботи:** освоїти методи непараметричної ідентифікації систем, і пакет підгонки кривих *Curve Fitting Toolbox* системи *MatLab*.

### 8.1 Постановка завдання

Непараметрична модель визначається в загальному випадку безперервною функцією, але вона може бути задана точками або у вигляді розкладання в ряд по будь-якій системі функцій. Специфіка лінійного динамічного об'єкта однозначно визначається його реакцією на одиничне імпульсний вплив. Ця обставина і лежить в основі визначення непараметричної моделі, яка характеризується імпульсною (ваговий) перехідною функцією.

При ідентифікації лінійного стаціонарного динамічного об'єкта статистичними методами вагова функція визначається з рівняння Вінера-Хопфа

$$R_{yx}(\tau) = \int_0^T g(t) R_{xx}(\tau - t) dt \quad (8.1)$$

Один із способів його вирішення - чисельний. Суть чисельного методу полягає в можливості подання цього рівняння системою лінійних алгебраїчних рівнянь. Для цього замінимо інтеграл кінцевою сумою

$$R_{yx}(\tau) = \sum_{n=0}^N g(n\Delta) R_{xx}(\tau - n\Delta) \cdot \Delta$$

тут  $\Delta$  - інтервал дискретизації;

$n\Delta$  - дискретний час;

$N$  - кількість інтервалів.

Це рівняння справедливо для  $\tau = \Delta, \dots, N\Delta$ .

Підставляючи в це рівняння  $N$  значень  $\tau$ , отримаємо систему з  $N$  рівнянь. Це система для визначення значень імпульсної перехідної функції в дискретні моменти  $\Delta, \dots, N\Delta$ .

Отже, отримуємо систему рівнянь

$$Ag = b \quad (8.2)$$

де  $A$  – квадратна матриця;

$$A = \begin{bmatrix} R_x(1) & R_x(2) & R_x(3) & \dots & R_x(N) \\ R_x(0) & R_x(1) & R_x(2) & \dots & R_x(N-1) \\ R_x(1) & R_x(0) & R_x(1) & \dots & R_x(N-2) \\ \dots & \dots & \dots & \dots & \dots \\ R_x(N) & R_x(N-1) & R_x(N-2) & \dots & R_x(0) \end{bmatrix}$$

$b = [b_1, b_2, \dots, b_m]^T$  - вектор-стовпець з компонентами  $b_i = R_{yx}(\tau)/\Delta$ ,  $i = 1, \dots, m$ ;

$g = [g_1, g_2, \dots, g_m]^T$  – вектор-стовпець шуканих значень імпульсної перехідної функції.

Вирішення цієї системи дозволяє визначити дискретні значення ординат ваговій функції  $g(\tau)$  в точках  $t, 2t, \dots, mt$ . Ці рішення виходять з великими похибками, так як замість істинних значень кореляційних функцій використовуються їх оцінки, а сама система погано обумовлена. Хоча отримані значення імпульсних перехідних функцій мають малу середньоквадратичну помилку, близьку до мінімуму, цінність їх невелика, так як ці функції не відповідають фізичним змістом процесів в об'єкті. Фізичний сенс мають гладкі рішення. Тому використовуються різні процедури згладжування імпульсної перехідної функції.

## 8.2 Апроксимація дискретної імпульсної перехідної функції

Для отримання згладжених імпульсних перехідних функцій, використовується процедура наближення значень цієї функції за допомогою будь-якого апроксимуючого полінома. Найчастіше використовується розкладання по будь-якій системі ортогональних поліномів

$$g(t) = \sum_{k=0}^N a_k \varphi_k(t)$$

де  $\varphi_k(t)$  – задана система функцій;

$N$  - порядок розкладання;

$a_k$  - коефіцієнти ряду, які обчислюються за формулою

$$a_k = \sum_{t=0}^m g(t) \varphi_k(t)$$



Функції розкладання вибираються таким чином, щоб досягти гарного наближення при невеликому їх числі. Доцільно вибирати ортогональні функції, що перетворюються по Лапласа (поліноми Чебишева, Лагерра, Лежандра).

### 8.3 Пакет підгонки кривих *Curve Fitting Toolbox*

Пакет підгонки кривих *Curve Fitting Toolbox* забезпечує підгонку кривих з використанням методу найменших квадратів. Роботу з пакетом треба починати з завантаження даних. Дані можуть бути сформовані в командному вікні системи у відповідних векторах.

Основне вікно пакета *Curve Fitting Tool* відкривається командою *cftool* (в командному вікні MatLab). Спочатку воно має порожнє вікно графіки і 4 кнопки над ним: *Data ...* - завантаження даних; *Fitting ...* - виконання підгонки; *Plotting ...* - графічна візуалізація підгонки; *Analysis ...* - аналіз результатів підгонки.

*Import даних.* Зазвичай робота з вікном *Curve Fitting Tool* починається з уточнення типу даних при завантаженні, для чого необхідно активізувати кнопку *Data ....* Це веде до появи вікна імпорту даних *Data*. У цьому вікні за допомогою відповідних вкладок можна встановити тип множин даних, переглянути їх, скорегувати і згладити. Натискання кнопки *Apply* і закриття вікна призводить до завантаження скоригованих даних в основне вікно *Curve Fitting Tool*. В результаті точки даних з'являються в поле графіки цього вікна.

Виконання підгонки заданого типу. Вибір кнопок *Fitting/New Fit* дозволяє почати процедуру підгонки.

Список, що *Type of fit* дозволяє установити будь-який з наступних типів підгонки (регресії):

- Custom equations — функція регресії, що задається користувачем;
- Exponential — експоненціальна регресія;
- Fourier — наближення поруч Фур'є;
- Gaussian — наближення кривої гауссіана;
- interpolant — інтерполяція з вибором методу;
- Polynomial — поліноміальна регресія;
- Power — статечна регресія;
- Rational — регресія раціональної функцією з вибором її чисельника і знаменника;
- Smoothing Spline — регресія згладжуючими сплайнами;
- Sum of Sin Func — регресія сумою синусоїдальних функцій;
- Weibull — регресія кривої Вейбулла.

Як видно з цього перерахування, представлений значний набір можливих видів наближень. Особливо важливо, що серед них є можливість завдання користувачем функції регресії будь-якого виду, зрозуміло, на основі вбудованих функцій MatLab.

Якщо натиснути кнопку *Fit Options...* вікна підгонки *Fitting*, можна задати ряду опцій підгонки. У вікні опцій підгонки можлива установка різних алгоритмів підгонки, меж зміни похідних, максимального значення функції і початкових значень параметрів регресії. Всі ці величини суттєво впливають на збіжність і швидкість підгонки.

*Графічна візуалізація регресії.* Кнопка *Plotting ...* вікна *Curve Fitting Tool* відкриває вікно установок графіки.

*Аналіз результатів регресії.* Важливим етапом наближення є аналіз результатів. Він проводиться при активізації кнопки *Analysis ...* вікна *Curve Fitting Tool*. При цьому відкривається вікно аналізу *Analysis*. У лівій частині вікна є ряд опцій аналізу: виконання обчислень, виведення графіків похідних і інтеграла для функції регресії і ін. Потрібні опції задаються установкою знака пташки у їх назви. Після цього натискання кнопки *Apply* формує таблицю результатів аналізу зверху вікна і виводить вікно з графічним представленням результатів аналізу.

Основним критерієм вибору кривої регресії є мінімум похибки *Residuals* в заданому діапазоні зміни аргументу, відсутність різких викидів кривої похибки на краях цього діапазону або в окремих його областях і т. д. Нерідко важливим є відповідність кривої регресії можливого закону розташування точок вихідних даних. Рекомендується випробувати кілька кривих регресії для заданого набору точок.

### Хід виконання практичного заняття

Зареєстровані вхідні і вихідні змінні досліджуваного об'єкта протягом певного інтервалу часу. За результатами цих вимірів обчислені автокореляційна і взаємнокореляційна функції (див. Таблицю варіантів). Потрібно визначити чисельним методом з рівняння Вінера-Хопфа імпульсну перехідну функцію. Потім отримані дискретні значення цієї функції слід апроксимувати поліномами різних порядків і вибрати найкращу ступінь апроксимації.

Записати систему лінійних алгебраїчних рівнянь (8.2) для свого варіанту.

Вирішити систему (8.2), використовуючи командне вікно системи MatLab:

- введіть компоненти матриці  $A$  і вектора  $b$  в командне вікно системи;
- перевірте наявність рішення системи (8.2), набравши в командному рядку  $\det(A)$ ; як відомо, для існування рішення необхідно, щоб  $\det(A) \neq 0$ ;
- отримаєте рішення системи, застосувавши процедуру  $g = b \cdot \text{inv}(A)$ .

Використовуючи пакет *Curve Fitting Tool*, виконайте апроксимацію отриманих дискретних значень поліномами різних порядків. Для завдання свого рівняння регресії в списку видів регресії треба вибрати позицію *Custom equations*. Це призводить до появи вікна завдання регресії *Create Custom Equations*. Це вікно має дві вкладки. На першій вкладці *Linear Equations* можна задати параметри рівняння, лінійного щодо коефіцієнтів регресії. Зауважимо, що сама залежність при цьому може бути нелінійною. На іншій вкладці *General Equations* можна встановити довільне нелінійне рівняння регресії, т. е. Здійснити нелінійну регресію.

Завершивши підгонку, виведіть графіки вихідних точок, кривих наближення і похибок.

Виберіть поліном, який найкращим чином апроксимує імпульсну вагову функцію.

#### Варіанти завдань:

Варіант 1	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,37	0,16	0,05	0,02	0,01	0,005	0
	Rxy(t)	0,1788	0,4729	0,3866	0,2565	0,1454	0,06	0,03	0
Варіант 2	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,47	0,2	0,1	0,05	0,01	0,005	0
	Rxy(t)	0,2376	0,5128	0,1343	0,2848	0,155	0,003	0,003	0
Варіант 3	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,53	0,26	0,15	0,1	0,02	0,003	0
	Rxy(t)	0,3049	0,5792	0,5575	0,4152	0,2176	0,002	0,002	0
Варіант 4	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,5	0,3	0,2	0,09	0,03	0,003	0
	Rxy(t)	0,3978	0,7428	0,7030	0,5960	0,3200	0,001	0,001	0
Варіант 5	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,52	0,28	0,18	0,1	0,02	0,005	0
	Rxy(t)	0,4872	0,9122	0,8998	0,7320	0,4088	0,002	0,002	0
Варіант 6	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,46	0,22	0,12	0,1	0,03	0,004	0
	Rxy(t)	0,3825	0,8008	0,7846	0,6214	0,3794	0,001	0,002	0
Варіант 7	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,38	0,19	0,04	0,02	0,01	0,005	0
	Rxy(t)	0,1735	0,4398	0,3637	0,2415	0,1026	0,003	0,002	0
Варіант 8	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,45	0,2	0,1	0,05	0,01	0,004	0
	Rxy(t)	0,2531	0,5655	0,5020	0,3365	0,1445	0,003	0,002	0
Варіант 9	t, мин	0	1	2	3	4	5	6	$\infty$
	Rx(t)	1	0,39	0,19	0,06	0,01	0,001	0,0001	0
	Rxy(t)	0,2428	0,6008	0,5659	0,4181	0,1166	0,0005	0,003	0

Варіант 10	t, мин	0	1	2	3	4	5	6	$\infty$
	R <sub>x</sub> (t)	1	0,35	0,18	0,05	0,02	0,01	0,005	0
	R <sub>xy</sub> (t)	0.2495	0.6569	0.5835	0.3389	0.1058	0,002	0,003	0

### Контрольні питання

- 1 У чому суть непараметричної ідентифікації?
- 2 Які об'єкти описує рівняння Вінера-Хопфа,
- 3 Чому можливо уявити рівняння Вінера-Хопфа системою алгебраїчних рівнянь?
- 4 Що таке апроксимація функції?
- 5 Поясніть призначення пакета Curve Fitting Tool.
- 6 Яка з функцій дає кращі результати наближення?

### Практичне заняття № 9. Методи ідентифікації із застосуванням настроювання адаптивних моделей

**Мета роботи:** Створити модель фільтра Калмана в середовищі Simulink, вивчити властивості його роботи при згладжуванні шумів і при відстежуванні траєкторій неспостережуваних сигналів, дати оцінку апостеріорної коваріації і матричного коефіцієнта посилення фільтра.

#### Хід виконання практичного заняття

Дана безперервна система з функцією передачі

$$W(p) = \frac{y(p)}{u(p)} = \frac{5p + 50}{3p^2 + 3.2p + 1}$$

і заданими характеристиками випадкових процесів типу білого шуму  $V = 1$  (для вхідних перешкоди) і  $R = 0,01$  (для вихідний). Відома апріорна інформація про сигнал:  $x_0 = 0$ ;  $P_0 = 10000$ .

Потрібно побудувати спостерігач стану для оцінки невідомого вектора стану  $x$ , коли критерієм оптимальності є мінімум середньоквадратичного відхилення побудованої оцінки від самого сигналу.

Таблиця 9.1

Параметр	Варіант (дорівнює номеру аспіранта по порядку в списку групи)									
	1	2	3	4	5	6	7	8	9	10
$k_0$	5	6	7	8	9	10	11	12	13	14
$k_1$	50	55	56	54	53	52	55	56	57	53
$T_1$	3	3	4	3	4	5	4	3	5	3
$T_2$	3,1	3,2	4,3	3,4	4,5	5,6	4,7	3,8	5,9	3,2

```

sys1=tf([5 50],[3 3.2 1])% завдання передавальної функції системи
sys=ss(sys1)% завдання системи в просторі станів
[A,B,C,D]=ssdata(sys); % формування матриць системи
n=length(A); % визначення порядку системи
t=0:0.001:2; % завдання масиву значень часу
x=zeros(n,1); % початкове значення математичного очікування сигналу
p=10000*diag(ones(n,1)); % початкове значення коваріаційної матриці сигналу
V=1000*diag(ones(n,1)); % коваріаційна матриця вхідних перешкоди
R=10; % коваріаційна матриця вихідний перешкоди
eps=.001;% задана похибка збіжності Калмановського коефіцієнта
Lk(:,1)=eye(n,1); % початкові наближення
Lk(:,2)=ones(n,1);
i=2;
pk(1,i)=p(1,1); pk(2,i)=p(2,2);
while not (abs(Lk(1,i)-Lk(1,i-1))<eps & abs(Lk(2,i)-Lk(2,i-1))<eps)
i=i+1;

```

```

q=A*p*A'+V;
p=q-q*C'*inv(C*q*C'+R)*C*q;
L=p*C'*R^-1;
pk(1,i)=p(1,1); pk(2,i)=p(2,2);
Lk(:,i)=L;
end
k=3:i;
figure(1);
plot(k, pk(1, 3:i),'--o', k, pk(2, 3:i),':s');
figure(2);
plot(k, Lk(1,3:i), '--o', k, Lk(2,3:i), ':s');

```

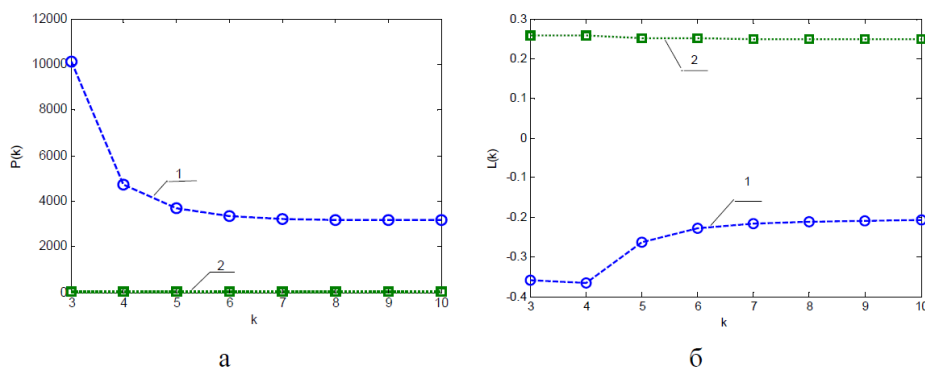


Рисунок 9.1 - Апостеріорна ковариація оцінок, (а) матричний коефіцієнт, (б) посилення фільтра

На рисунку 9.1 (а) показана покрокова апостеріорна ковариація, (невизначеність) оцінок  $\hat{x}_1$  (крива 1) і  $\hat{x}_2$  (крива 2). Видно, що починаючи з заданого апріорного значення  $P(0)$  величина ковариації оцінки кожного компонента вектора  $\hat{x}$  швидко наближається до свого асимптотичному значенням. Відповідно, на рисунку 9.1 (б) показана асимптотична збіжність матриці коефіцієнтів посилення фільтра  $L=[l_1 l_2]^T$  де зміни коефіцієнтів  $l_1$  і  $l_2$  відповідають криві 1 і 2.

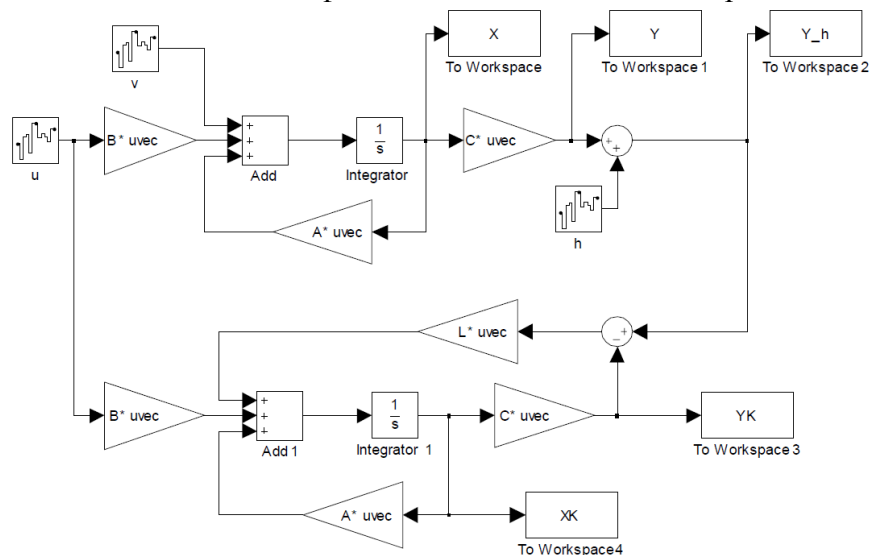


Рисунок 9.2 - Імітаційна модель фільтра Калмана

Структура даного об'єкту і фільтра Калмана, реалізована в середовищі Simulink системи MatLab, представлена на рисунку 9.2. У властивостях *To Workspace Save format* встановити *Array*.

На даній моделі прийняті позначення:  $u$ ,  $v$ ,  $h$  - моделюються сигнали вхідного впливу, перешкоди об'єкта і шуму вимірювань;  $X$ ,  $Y$ ,  $Y_h$  - вектор стану об'єкта, вектор виміру без урахування вихідний перешкоди і з її урахуванням відповідно;  $XK$ ,  $YK$  - вектор стану і вектор виміру фільтра Калмана (тобто оцінки вектора стану і вектора виміру об'єкта).

Запис  $A * uvec$ ,  $B * uvec$ ,  $C * uvec$  і  $L * uvec$  позначає векторне множення матриць  $A$ ,  $B$ ,  $C$  або  $L$  на сумісний зовнішній вхід. У властивостях *Gain Multiplication* встановити *Matrix* ( $K * u$ ) ( $u$  vector).

Результати комп'ютерного моделювання представлені на малюнках 9.3 - 9.5. Для відображення графіків в *Floating Scope - Signal Selector* вибрати *Gain3* і *Gain8*, а для *Floating Scope 1* - *Integrator1* і *Integrator2*.

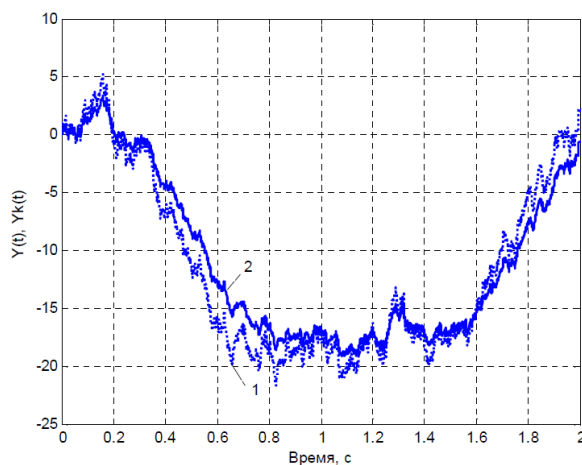


Рисунок 9.3 - Порівняння точного, без урахування перешкоди (крива 1) і оціненого  $\hat{y} = C\hat{x}$  (крива 2) значень вихідних сигналів

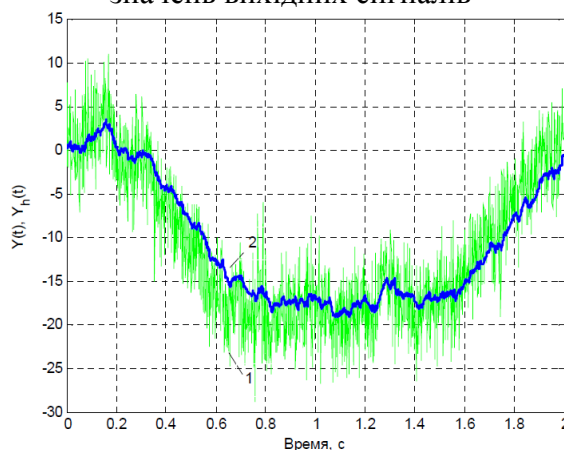


Рисунок 9.4 - Порівняння виміряного зашумленого  $y$  (крива 1) і оціненого  $\hat{y} = C\hat{x}$  (крива 2) значень вихідних сигналів

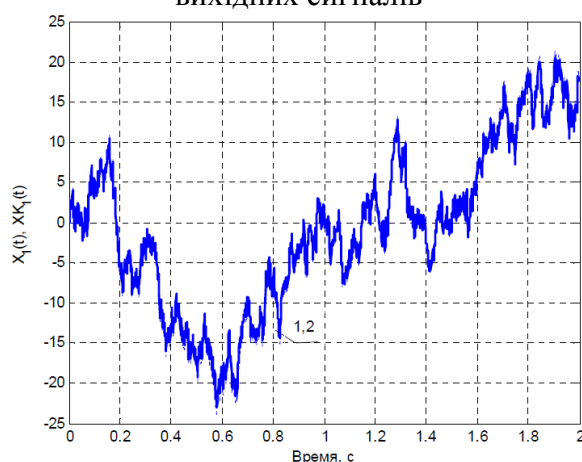


Рисунок 9.5 - Порівняння істинного (крива 1) значення компоненти  $x_1$  вектора стану системи і його оцінки  $\hat{x}_1$  (крива 2)

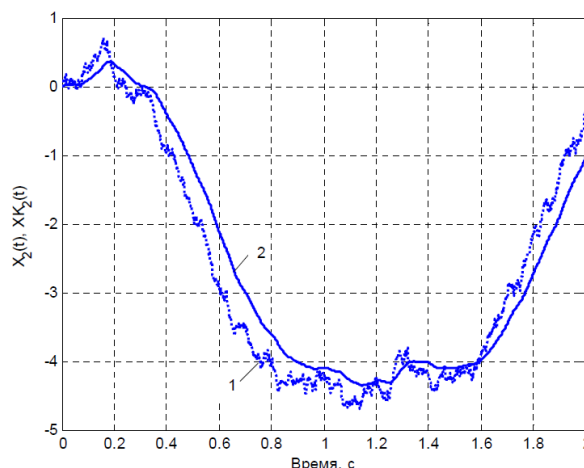


Рисунок 9.6 Порівняння істинного (крива 1) значення компоненти  $x_2$  вектора стану системи і його оцінки  $\hat{x}_2$  (крива 2)

Результати комп'ютерного моделювання показують цілком задовільну якість роботи фільтра Калмана як при згладжуванні шумів (рис. 9.4), так і при відстежуванні траєкторій неспостережуваних сигналів  $x$  (рис. 9.5, 9.6). Оцінка практично повністю збігається зі своїм істинним значенням. Для сигналу  $x_2$  оцінюється траєкторія відстає від реальної, але при цьому поліпшуються згладжують властивості фільтра.

#### Контрольні питання

1. Адаптивні моделі
2. Модель фільтра Калмана;
3. Алгоритм фільтра Калмана;
4. Імітаційна модель фільтра Калмана;
5. Апостеріорна коваріація,
6. Ідея  $n$ -мірного спостерігача;
7. Комп'ютерне моделювання.

### Практичне заняття № 10. Ідентифікація із застосуванням методу вагових функцій

**Мета роботи:** Дати порівняння аналітичної та експериментальної імпульсних вагових характеристик на основі ідентифікаційного експерименту в залежності від інтервалу вимірювань і кроку дискретизації.

#### Хід виконання практичного заняття

Наведемо програмну реалізацію процедури ідентифікації імпульсної перехідної характеристики для того ж базового об'єкта з передатною функцією

$$W_o(p) = \frac{k_0}{T_1 p^2 + T_2 p + 1}$$

і коефіцієнтами  $k_0=25$ ;  $T_1=36$ ;  $T_2=15$ .

```
s1=tf([25],[36 15 1])% безперервна передавальна функція об'єкта
T_end=45;% інтервал вимірювань
dt=1.5;% крок дискретизації
t=0:dt:T_end;% масив дискретного часу
N=length(t);% розмір вибірки
u=sign(normrnd(0, 2, [N 1]));% моделювання вхідного впливу
y=lsim(s1,u,t) ;% вихідний вплив
% заповнення матриці U вхідних значень
for i=1:N
```

```

for j=1:N
    if(i>=j) U(i,j)=u(i+1-j);
    else U(i,j)=0;
    end;
end
end
w=1/dt*(inv(U'*U))*(U'*y);
w0= impulse(s1,t);
plot (t, w0, t, w,':');
grid;

```

Результати ідентифікації ПФ даного об'єкту представлені на рисунку 10.1.

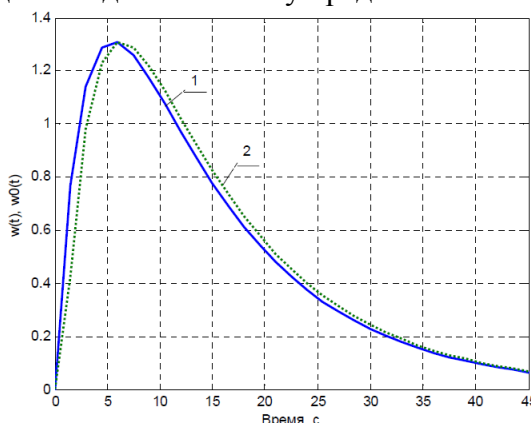


Рисунок 10.1 - Імпульсні характеристики об'єкта, аналітична (1) і експериментальна (2) імпульсні характеристики об'єкта

Ідентифікаційний експеримент проводився при подачі на об'єкт випадкової послідовності сигналів  $\pm 1$ . Порівняння аналітичної та експериментально отриманої імпульсних вагових характеристик (рисунок 10.1) показує високу точність даного методу, істотно залежить від інтервалу вимірювань і кроку дискретизації. Розрахунки показують, що потрібно уважно вибирати інтервал вимірювань  $T_m$  і час встановлення реакції  $T_s = N_s \Delta t$ , тому що при проведенні процедури оцінювання на ділянці, де ПФ прагне до нуля, матриця  $[U^T U]^{-1}$  стає близькою до виродження, наслідком чого стане розбіжність рішення.

#### Контрольні питання

1. Ідентифікація із застосуванням методу вагових функцій.
2. Аналітична імпульсна вагова характеристика.
3. Експериментальна імпульсна вагова характеристика.
4. Імпульсна перехідна функція.
5. Передавальна функція.

### Практичне заняття № 11. Кореляційний метод ідентифікації

**Мета роботи:** Обчислити кореляційний функцію, дати наближену оцінку ваговій функції об'єкта, побудувати графіки істинної імпульсної перехідної функції об'єкта і взаємну кореляційна функція вхідного і вихідного сигналів.

#### Хід виконання практичного заняття

Проілюструємо визначення імпульсної вагової функції аперіодиченого об'єкта першого порядку з запізненням, передавальна функція якого дорівнює

$$W(p) = \frac{10}{3p + 1} e^{-10p}$$

за результатами проведення експерименту при вхідному сигналі типу білого шуму.

```
s1=tf([10],[3 1], 'td', 10) % безперервна передавальна функція об'єкта
```

```

T_izm=60; % інтервал вимірювань
T_end=30; % інтервал оцінювання
dt=0.05; % крок дискретизації
t_izm=0:dt:T_izm; % масив дискретного часу вимірювань
t=0:dt:T_end; % масив дискретного часу оцінювання
N_izm=length(t_izm); % розмір вибірки інтервалу вимірювань
N=length(t); % розмір вибірки інтервалу оцінювання
u=randn(N_izm,1); % масив значень вхідного впливу
y=lsim(s1,u,t_izm); % масив значень вихідного впливу
Ruu=xcorr(u,u,'biased'); % обчислення кореляційної функції на інтервалі вимірювань по одній
реалізації
Ruy=xcorr(y,u,'biased'); % обчислення взаємної кореляційної функції по одній реалізації
% обчислення усереднених авто- і взаємної кореляційних функцій по 10 реалізацій
Ruu_mid=0; % початкове значення автокореляційної функції
Ruy_mid=0; % початкове значення взаємної кореляційної функції
for k=1:10
    u=randn(N_izm,1); % масив значень вхідного впливу
    y=lsim(s1,u,t_izm); % масив значень вихідного впливу
    Ruu_mid= Ruu_mid+xcorr(u,u,'biased');
    Ruy_mid= Ruy_mid+xcorr(y,u,'biased');
end;
Ruu_mid= Ruu_mid/10; % усереднене значення автокоррел. функції
Ruy_mid= Ruy_mid/10; % усереднене значення взаємної коррел. функції
tau=- T_end:dt: T_end;
figure (1);
plot (tau, Ruu(N_izm-N+1: N_izm +N-1));
figure (2);
plot (tau, Ruu_mid(N_izm-N+1: N_izm +N-1));
w= impulse(s1,t);
figure (3);
plot (t, w, t, 1/dt* Ruy (N_izm: N_izm +N-1));
figure (4);
plot (t, w, t, 1/dt* Ruy_mid (N_izm: N_izm +N-1));

```

На рисунках 11.1, 11.2 графічно представлені результати розрахунку

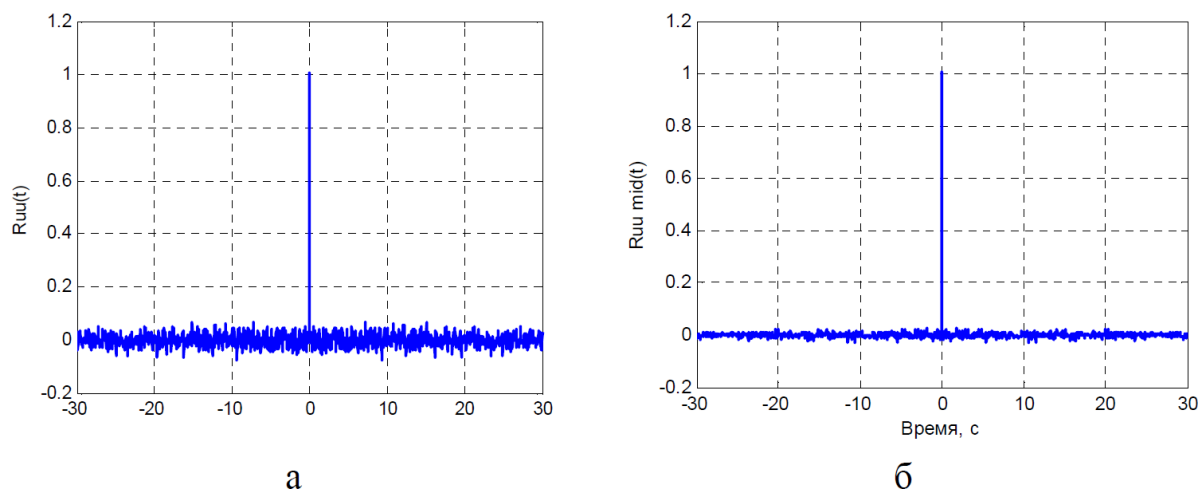


Рисунок 11.1 - Кореляційна функція вхідного сигналу типу білий шум  
Кореляційна функція отримана по одній реалізації (а) і усереднена по десяти реалізацій (б)



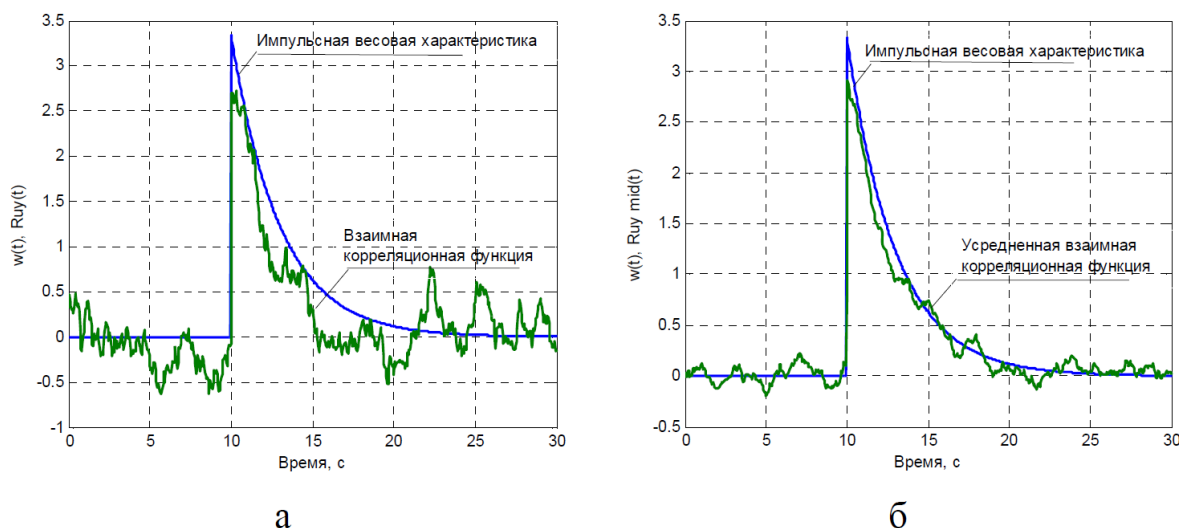


Рисунок 11.2 - Справжня (аналітична) імпульсна перехідна функція об'єкта і взаємна кореляційна функція вхідного і вихідного сигналів. Справжня функція отримана по одній реалізації (а) і усереднена по десяти реалізацій (б)

Існування певних похибок між отриманими взаємними кореляційними функціями і аналітичної вагової характеристикою пояснюється, по-перше, неможливістю сформувати на вході об'єкта ідеальний білий шум, а по-друге, використанням не істини кореляційних функцій процесу, а їх оцінок, отриманих на обмежених часових інтервалах. Крім того, можуть мати місце недостатня точність вимірювань, можлива нестационарність процесу. Всі ці чинники значно обмежують можливість використання кореляційних методів. Результати обчислень показують доцільність збільшення інтервалу вимірювань кореляційних функцій, зменшення кроку дискретизації, адекватного вибору інтервалу оцінювання та застосування згладжуючих алгоритмів.

#### Контрольні питання

1. Кореляційний метод ідентифікації.
2. Імпульсна перехідна функція об'єкта.
3. Взаємна кореляційна функція.
4. Кореляційні методи.
5. Рівняння Вінера-Хопфа.
6. Детерміновані моделі.
7. Функція Дірака.
8. Інтенсивність шуму.
9. Автокорреляційна функція.