

Міністерство освіти і науки України  
Дніпровський національний університет імені Олеся Гончара

---

Кафедра систем автоматизованого управління

Методичні вказівки  
до виконання лабораторних робіт  
з курсу  
КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ ДИНАМІЧНИХ СИСТЕМ

Для студентів  
Фізико-технічного факультету  
Дніпровського національного університету

Розробив: доцент кафедри САК Мороз Ю.І.

## ЗМІСТ

|  |    |
|--|----|
| Лабораторна робота № 1 Розв’язання диференційних рівнянь динамічної системи за допомогою систем візуального моделювання або на аналогових обчислювальних машинах ..... | 2  |
| Лабораторна робота № 2 Візуальне моделювання систем автоматичного керування.....   | 11 |
| Лабораторна робота № 3 Дослідження впливу збурюючого сигналу на реакцію системи і величину помилки керування .....   | 19 |
| Лабораторна робота № 4 Симуляція кулі, що підстрибує .....   | 24 |
| Лабораторна робота № 5 Особливості дослідження системи з ковзним процесом.....   | 32 |
| СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ .....   | 38 |

## Лабораторна робота № 1

**Розв’язання диференційних рівнянь динамічної системи за допомогою систем візуального моделювання або на аналогових обчислювальних машинах**

Ціль роботи: Опанувати навичками складання структурних схем систем диференціальних рівнянь і їхнього рішення за допомогою систем автоматизованого візуального моделювання.

Потрібне обладнання і програмне забезпечення: ПЕОМ класу Pentium, або вище; операційна система Windows XP або вище; математичний пакет Matlab 5.3 або вище.

У загальному випадку поведження системи автоматичного керування (САК) або іншої динамічної системи (ДС) в часі можна характеризувати не тільки вхідною й вихідною величинами, але й проміжними змінними в ланцюзі системи, число яких дорівнює порядку системи  $n$ . Ці проміжні змінні потрібно вибирати так, щоб, задавши початкові значення цим  $n$  змінним, ми могли б визначити значення змінної виходу в будь-який наступний момент часу, тобто визначити стан системи в будь-який момент часу. Тому цей  $n$ -мірний вектор змінних називається **вектором стану системи**, а множина можливих значень цього вектора утворює **простір станів системи**.

Обрати змінні стану можна різними способами. Після їхнього вибору властивості системи керування в просторі станів описуються не одним диференціальним рівнянням  $n$ -го порядку, а системою з  $n$  диференціальних рівнянь першого порядку в нормальній формі. У загальному випадку така система рівнянь має вигляд



$\mathbf{D} = \begin{bmatrix} d_{11} & \dots & d_{1m} \\ \dots & \dots & \dots \\ d_{r1} & \dots & d_{rm} \end{bmatrix}$  – прохідна матриця системи розміром  $(r \times m)$ . В системах

автоматичного керування ця матриця в більшості випадків дорівнює нулю, бо вхідні впливи не проходять на вихід системи без перетворення.

Системі рівнянь (1.1) і (1.2) може бути поставлена у відповідність структурна схема, що складається тільки з підсилювальних і інтегруючих ланок, а також суматорів. Пояснюється це тим, що вирази у правих частинах (1.1) являють собою похідні змінних стану й для одержання значень самих змінних залишається тільки інтегрувати дані вирази.

Можливість подання рівнянь (1.1) у вигляді блок-схеми, що містить найпростіші елементи, використовується при моделюванні процесів у САК на аналогових обчислювальних машинах (АОМ). Найпростіша АОМ являє собою набір підсилювачів зі змінним коефіцієнтом передачі й інтеграторів, виконаних на стандартних операційних підсилювачах. Вона має також спеціальне складальне поле, що дозволяє з'єднувати її елементи в потрібній послідовності (у відповідності зі структурною схемою системи рівнянь, що моделюється). Результати моделювання можуть бути виведені безпосередньо на екран осцилографа або оброблені на ЕОМ з використанням аналогово-цифрових перетворювачів. Достоїнством такого моделювання є можливість побудови аналогових моделей окремих складних і дорогих частин системи або об'єкта керування й включення цих моделей у ланцюг САК для проведення стендових випробувань і відпрацювання різних законів керування в реальному масштабі часу.

В останні роки широке поширення одержують так звані системи автоматизованого візуального моделювання систем керування на ЕОМ (ДІСПАС, СІАМ, Simulink і т.п.). Відмінною рисою цих програмних продуктів є те, що вони оперують моделями динамічних систем у вигляді структурних схем, які складаються зі стандартних ланок, наявних в арсеналі даних програм. Така програма сама формує загальну систему рівнянь ДС з моделей окремих ланок, що перебувають у неї в базі даних, і проводить її аналіз одним з наявних у неї в розпорядженні чисельних методів.

Використання систем автоматизованого моделювання є дуже зручним, тому що не вимагає знання спеціальних мов програмування. Аналіз ДС зводиться в остаточному підсумку до вміння формувати на підставі рівнянь ланок структурну схему системи й уведенню її в ЕОМ.

Процедуру складання структурної схеми розглянемо на прикладі САК четвертого порядку з одним входом  $u$  і одним виходом  $y$ , що описується наступною системою рівнянь у матричній формі

$$p \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & a_{13} & a_{14} \\ 0 & a_{22} & a_{23} & 0 \\ a_{31} & a_{32} & a_{33} & 0 \\ a_{41} & 0 & 0 & a_{44} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ b_4 \end{bmatrix} \cdot u, \quad (1.4)$$

$$y = \begin{bmatrix} 0 & c_2 & c_3 & 0 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = c_2 x_2 + c_3 x_3. \quad (1.5)$$

Даній системі рівнянь відповідає структурна схема, показана на рис. 1.1.

Як видно з рис. 1.1 кожному з рівнянь (1.4) відповідає інтегруючий блок із суматором на вході. На виході кожного інтегратора одержуємо одну зі змінних стану. Сигнали з виходів інтеграторів подаються на входи суматорів через підсилювальні блоки з коефіцієнтами передачі, рівними коефіцієнтам  $a_{ij}$  відповідних диференціальних рівнянь. Керуючий вплив  $u$  надходить на вхід суматора, що відповідає четвертому рівнянню з (1.4), через блок з коефіцієнтом підсилення  $b_4$ . І, нарешті, відповідно до рівняння (1.5) змінні  $x_2$  і  $x_3$  через блоки з коефіцієнтами передачі  $c_2$  і  $c_3$  подаються на ще один суматор (унизу схеми), утворюючи вихідний сигнал  $y$ .

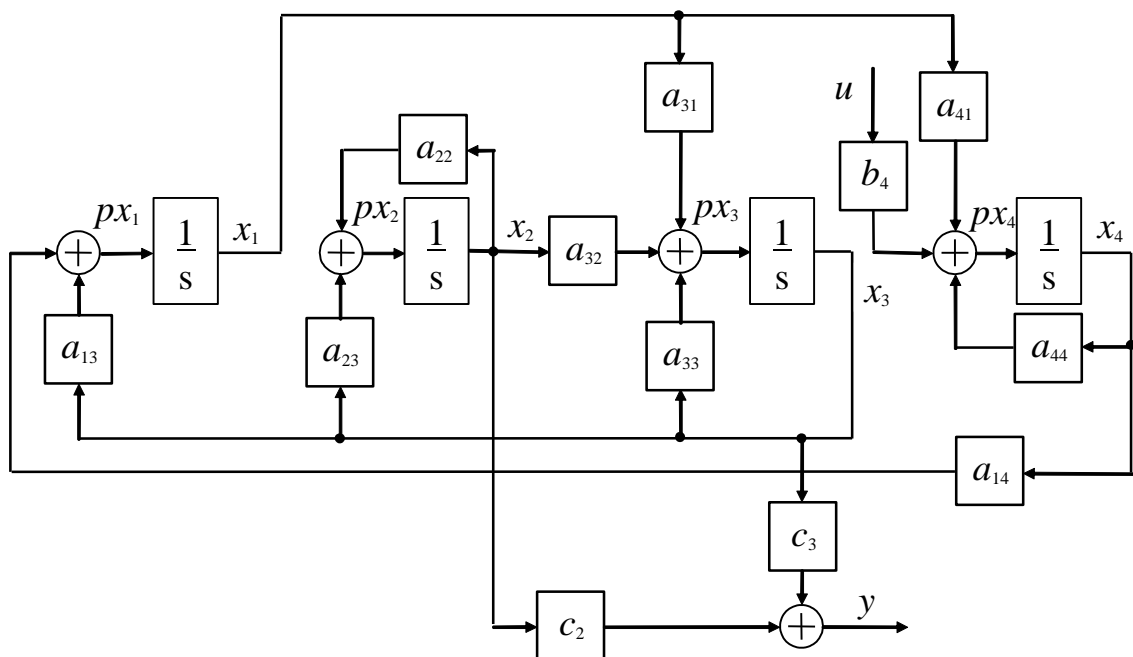


Рисунок 1.1 – Структурна схема системи рівнянь (1.4) і (1.5)

### Хід роботи

1. Скласти структурну схему для наступної системи диференціальних рівнянь

$$p\bar{x} = A\bar{x} + Bu; \quad y = C\bar{x},$$

де  $\bar{x}$  – вектор-стовпець змінних стану розміром  $(4 \times 1)$ ;  $A$  – власна параметрична матриця системи розміром  $(4 \times 4)$ ;  $B$  – вхідна матриця розміром  $(4 \times 1)$ ;  $C$  – матриця виходу розміром  $(1 \times 4)$ ;  $u$  і  $y$  – вхідний вплив і вихідна змінна відповідно. Значення коефіцієнтів матриць  $A$ ,  $B$  і  $C$  наведені в табл. 1.1 (номер варіанта задається викладачем).

2. Після перевірки викладачем схеми викликати систему автоматизованого моделювання Simulink, увести отриману структурну схему й провести розрахунок перехідного процесу в системі керування.

Для цього включити комп'ютер, після його завантаження запустити оболонку програми Matlab. Після завантаження оболонки Matlab запустити браузер бібліотеки Simulink (рис. 1.2а), для чого натиснути на панелі інструментів відповідну кнопку, або ввести команду Simulink у командному вікні Matlab.

Натиснувши кнопку «**Create a new model**» відкрити вікно нової моделі (рис. 1.2б). Увести структурну схему шляхом переносу потрібних блоків з вікна бібліотеки у вікно моделі і з'єднання їх у необхідному порядку. Задати параметри ланок, викликаючи їх вікна властивостей подвійним натиском лівої кнопки миші на піктограмі блоку. Вхідний вплив  $u$  прийняти у вигляді одиначної східчастої функції  $1(t)$ . Початкові умови  $x_1(0)$ ,  $x_2(0)$ ,  $x_3(0)$ ,  $x_4(0)$  прийняти нульовими.

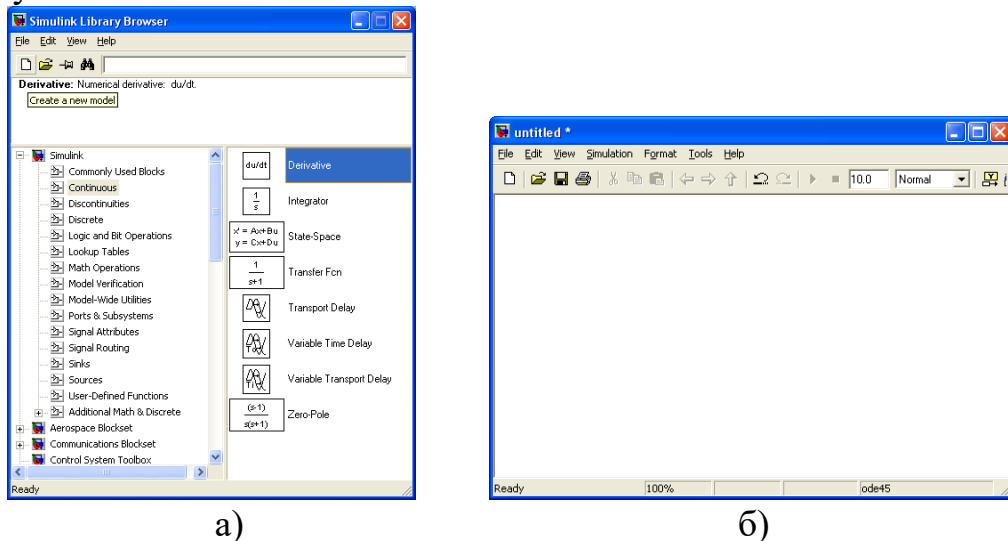


Рисунок 1.2 – Вікно бібліотеки Simulink і вікно пустої моделі

Таблиця 1.1 – Варіанти вхідних даних

| Варіант | Матриця A |    |    |     | B | C <sup>T</sup> |
|---------|-----------|----|----|-----|---|----------------|
| 1       | 0         | 0  | 0  | 10  | 0 | 1              |
|         | 0         | 2  | 20 | -20 | 0 | 0              |
|         | 0         | -4 | 0  | 0   | 4 | 0              |
|         | -2        | 2  | 0  | -8  | 0 | 4              |
|         | -20       | 0  | 0  | 10  | 0 | 2              |

|   |     |     |     |     |    |    |
|---|-----|-----|-----|-----|----|----|
| 2 | 0   | -25 | 50  | 25  | 0  | 1  |
|   | 0   | -20 | 0   | 0   | 20 | 0  |
|   | -20 | 10  | 0   | -10 | 0  | -1 |
| 3 | -10 | -8  | 2   | 0   | 8  | 1  |
|   | 10  | 0   | -10 | 0   | 0  | 0  |
|   | 2   | 8   | -4  | -8  | 0  | 0  |
|   | 0   | 0   | 10  | 0   | 0  | 0  |
| 4 | -16 | 8   | 0   | -8  | 0  | 2  |
|   | -5  | 0   | 5   | 0   | 0  | 0  |
|   | 0   | -4  | -12 | 0   | 4  | 0  |
|   | 20  | 0   | 0   | -5  | 0  | 1  |
| 5 | 0   | 0   | 10  | -10 | 0  | 0  |
|   | 0   | 0   | 0   | 20  | 0  | 0  |
|   | -4  | 0   | -5  | 1   | 4  | 1  |
|   | 8   | -8  | 2   | -4  | 0  | 0  |
| 6 | -4  | 4   | 0   | 20  | 0  | 1  |
|   | 2   | -2  | 10  | 0   | 0  | -1 |
|   | 0   | 5   | -50 | 0   | 0  | 10 |
|   | -2  | 0   | 0   | 0   | 2  | 0  |

Блок інтегратора знаходиться у розділі бібліотеки **Simulink**→**Continuous** і має вигляд, показаний на рис. 1.3. Там же показане вікно його властивостей. В даному випадку моделювання можна залишити параметри, задані за замовчуванням, незмінними. Зокрема, джерело завдання початкових умов в полі «**Initial condition source**» треба залишити внутрішнім («**internal**»), а також у полі «**Initial condition**» залишити нульове значення. Для кращої орієнтації у схемі рекомендується після вводу чергового інтегратора відредагувати його заголовок, додавши найменування вихідної змінної стану ( $x_1, x_2, x_3$  або  $x_4$ ).

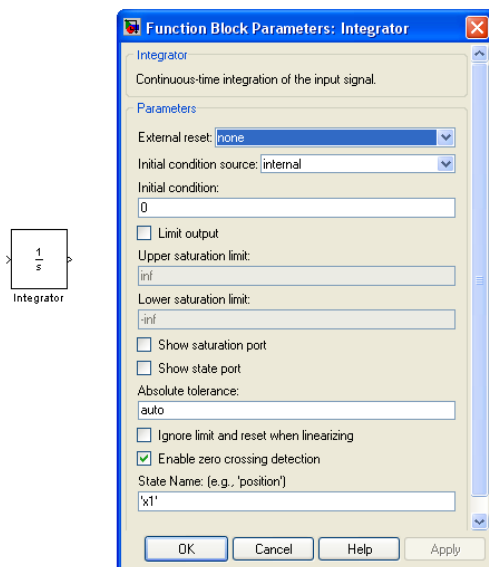


Рисунок 1.3 – Блок інтегратора

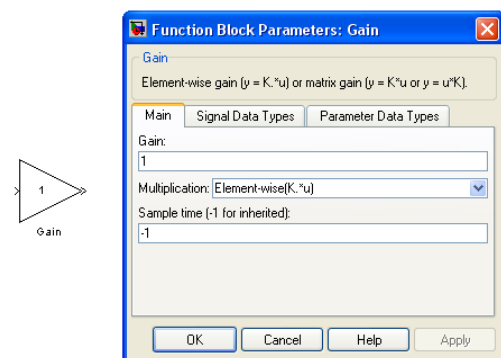


Рисунок 1.4 – Блок підсилювача

Блок підсилювача **Gain** знаходиться у розділі бібліотеки **Simulink**→**Math Operations** і має вигляд, показаний на рис. 1.4. Там же показане вікно його властивостей. Значення коефіцієнта підсилення задається у полі «**Gain**», інші параметри блоку можна залишити без змін. Для кращої орієнтації у схемі рекомендується після вводу чергового підсилювача відредагувати його заголовок, додавши найменування відповідного коефіцієнта рівнянь  $a_{ij}$ ,  $b_i$  або  $c_i$ .

Блок суматора також знаходиться у розділі бібліотеки **Simulink**→**Math Operations** і має вигляд, показаний на рис. 1.5. Вид піктограми блоку (круглий чи прямокутний) регулюється у полі «**Icon shape**» вибором пунктів «**round**» чи «**rectangular**». Кількість входів та їх знак регулюється вводом відповідної кількості знаків «+» чи «-» у полі «**List of signs**».

Вхідний вплив  $u$  у вигляді одиничної східчастої функції  $1(t)$  можна задати за допомогою джерела східчастого сигналу **Step** (рис. 1.6), блок якого знаходиться у розділі бібліотеки **Simulink**→**Sources**. У даному випадку треба тільки змінити час виникнення стрибка у полі «**Step time**» – встановити 0 замість значення 1, що встановлено за замовчуванням. Початкове (0) і кінцеве (1) значення сигналу у полях «**Initial value**» і «**Final value**» можна залишити без змін.

Для реєстрації результатів моделювання сигнал з виходу суматора, що формує вихідну змінну  $y$ , треба подати на осцилограф, який реалізовано блоком **Scope** (рис. 1.7a). Цей блок знаходиться у розділі бібліотеки **Simulink**→**Sinks**.

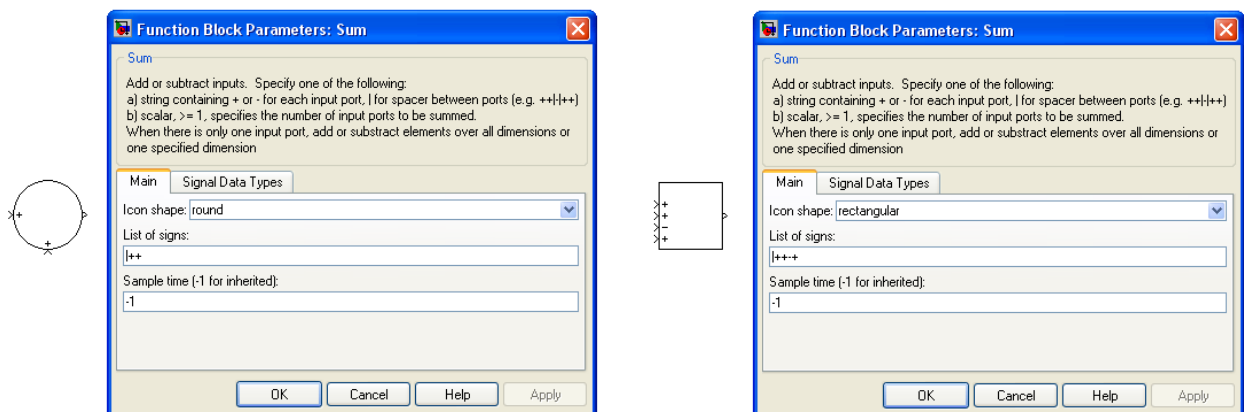


Рисунок 1.5 – Блок суматора у виді круга чи прямокутника з різною кількістю входів



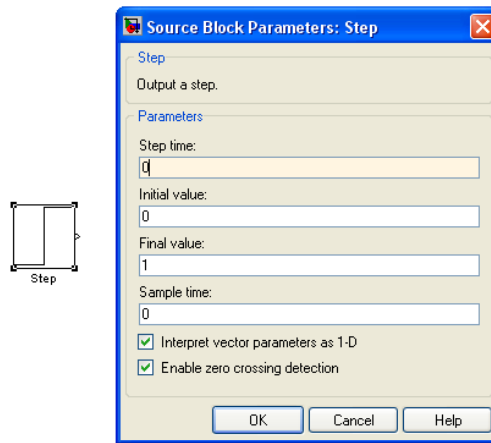
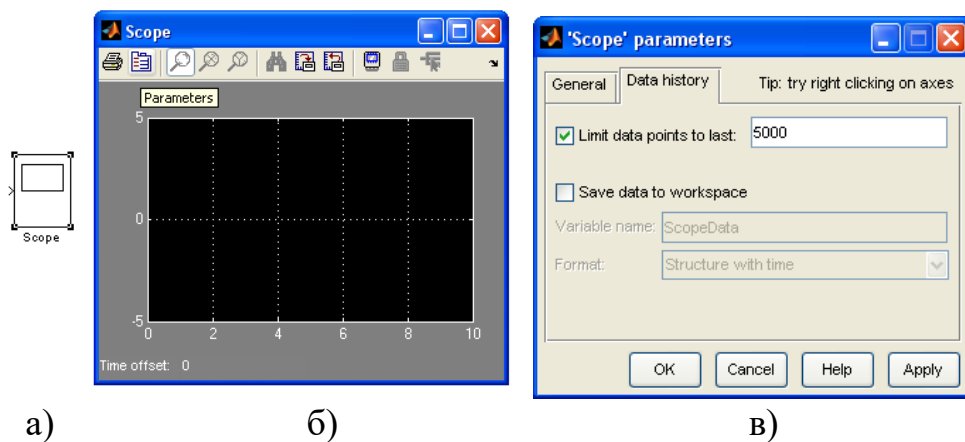


Рисунок 1.6 – Джерело східчастого сигналу Step



а)

б)

в)

Рисунок 1.7 – Блок осцилографа та його вікна

Подвійний натиск на піктограму осцилографа відкриває вікно графіку (рис. 1.7б), яке поки що зостається пустим. Перед початком моделювання є доцільним натиснути кнопку «**Parameters**» і відкрити вікно властивостей блоку (рис. 1.7в), перейти на вкладку «**Data history**» і зняти позначку з поля «**Limit data points to last**», що обмежує кількість точок, що виводяться на графік, до останніх 5000.

Користуючись меню вікна моделі виконати команду «**Simulation**→**Configuration parameters**» для відкриття відповідного вікна параметрів моделювання (рис. 1.8). На вкладці «**Solver**» можна обрати метод інтегрування, наприклад ode45, тип інтегрування з постійним або змінним шагом, задати максимальне та мінімальне значення шагу, допустимого погрішність, а також початковий і кінцевий моменти часу. У даному випадку треба задати початковий і кінцевий моменти часу  $t_0=0$  і  $t_k=4$  сек. в полях «**Start time**» і «**Stop time**». Інші параметри можна залишити заданими за замовчуванням.

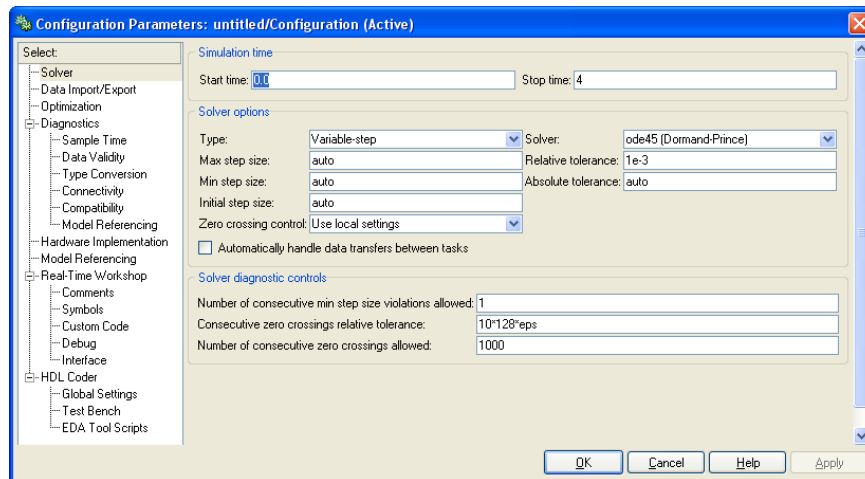



Рисунок 1.8 – Вікно параметрів конфігурації Simulink

Розрахувати перехідну характеристику  $h(t)=y(t)$  системи, виконавши команду меню «**Simulation**→**Start**» або натиснувши відповідну кнопку  на панелі інструментів вікна моделі. Хід інтегрування рівнянь відображається бігунком на панелі стану унизу вікна моделі. Після закінчення інтегрування там з'являється повідомлення «Ready».

Вивести результати розрахунку  $y(t)$  на екран у вигляді графіка, відкривши вікно осцилографа. У випадку, якщо перехідний процес  $y(t)$  ще не встановився, продовжити інтегрування рівнянь, збільшивши кінцевий момент часу  $t_k$ . Якщо ж перехідний процес є сталим вже значний час, повторити інтегрування рівнянь спочатку, зменшивши  $t_k$  до моменту встановлення  $y(t)$ . Оцінити загальний час, витрачений на виконання даної роботи (починаючи зі складання структурної схеми до одержання результатів).

3. У випадку лінійної системи диференційних рівнянь у просторі станів (1.3) в Simulink існує можливість спростити структурну схему моделі, за рахунок використання готового блоку **State-Space** (див. рис. 1.9). У вікні його властивостей задаються матриці коефіцієнтів рівнянь **A**, **B**, **C** і **D**: у квадратних дужках у одну строку записуються елементи першого рядка матриці через пробіл, ставиться точка з комою, записуються елементи другого рядка і т.д. поки не будуть записані усі рядки. Наприклад, одинична матриця третього порядку в Matlab буде мати вид [1 0 0;0 1 0;0 0 1].

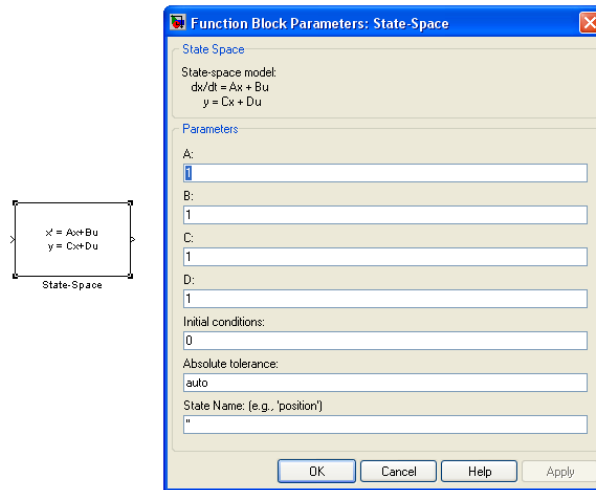


Рисунок 1.9 – Блок завдання моделі у просторі станів

Для перевірки результатів моделювання в пункті 2 з розділу бібліотеки **Simulink**→**Continuous** перенесіть блок **State-Space** у вікно вашої моделі. Відкрийте вікно його властивостей і задайте значення матриць ваших рівнянь, а також вектор нульових початкових умов в полі «**Initial conditions**». На вхід цього блоку подайте одиничний східчастий сигнал з блоку **Step**, а вихід блоку подайте на другий осцилограф. Проведіть моделювання і порівняйте графіки двох осцилографів. Якщо ваша перша модель правильна, то графіки будуть співпадати.

4. Оформіть звіт про виконання лабораторної роботи.

#### Контрольні питання

1. Що таке змінні стану й простір станів системи?
2. Яким чином описується САК в просторі станів?
3. Як складається структурна схема системи диференціальних рівнянь САК?
4. Як виконується уведення структурної схеми в систему SIMULINK?
5. Як виконується розрахунок перехідного процесу в системі SIMULINK (вибір методу інтегрування, рахунок, вивід результатів на екран)?

## Лабораторна робота № 2

### Візуальне моделювання систем автоматичного керування

Ціль роботи: Виробити навички дослідження характеристик САК за допомогою пакету візуального моделювання Simulink. Ознайомитися з

методами корекції САК. Дослідити залежність реакції системи та помилки управління від типу вхідного сигналу.

Потрібне обладнання і програмне забезпечення: ПЕОМ класу Pentium, або вище; операційна система Windows XP або вище; математичний пакет Matlab 5.3 або вище.

Порядок розрахунку САК розглянемо на прикладі.

Для зниження затрат на виробництво прийнято рішення про автоматизацію технологічного процесу, у якому використовуються газові сушильні печі. Проектується система управління температурним режимом (СУТР) сушильної печі.

Розробити функціональну схему СУТР згідно табл. 2.1. Скласти передавальні функції ланок та усієї СУТР в цілому. Вхідні дані для розрахунку наведені у табл. 2.2. При вибраних параметрах побудувати ЛАЧХ та ЛФЧХ розімкненої системи і визначити запаси стійкості. Побудувати перехідну характеристику замкненої системи і визначити величину перерегулювання та тривалість перехідного процесу. При необхідності вибрати корегувальний пристрій для забезпечення заданих параметрів якості.

Таблиця 2.1

| Ланка                                 | Тип елемента                          | Рівняння   |
|---------------------------------------|---------------------------------------|--|
| Піч сушильна                          | Газова                                | $T_{\text{п}}\dot{\theta} + \theta = k_{\text{п}}\alpha$ , де $\theta$ - температура;<br>$\alpha$ - кут повороту вентиля газопроводу |
| Підсилювач                            | Магнітний                             | $T_{\text{у}}\dot{u}_2 + u_2 = k_{\text{у}}u_1$ , де $u_1$ і $u_2$ - напруги на вході і виході підсилювача                           |
| Чутливий елемент – датчик температури | Резистивний міст з термометром опору  | $u_1 = k_{\text{т}}(\theta_3 - \theta)$ , де $\theta_3$ - задана температура; $u_1$ - напруга на виході мосту                        |
| Виконавчий орган                      | Двигун постійного струму з редуктором | $T_{\text{м}}\dot{\alpha} + \alpha = k_{\text{д}}u_2$  |

Таблиця 2.2

| Найменування величин, їх позначення і розмірність  | Значення |
|--|----------|
| Стала часу печі, $T_{\text{п}}$ , с                | 3.0      |
| Коефіцієнт передачі печі, $k_{\text{п}}$ , °С/град | 4.5      |

|   |       |
|---|-------|
| Коефіцієнт передачі резистивного мосту, $k_T$ , В/°С        | 0.018 |
| Електромеханічна стала часу двигуна з редуктором, $T_M$ , с | 0.14  |
| Коефіцієнт передачі двигуна з редуктором, $k_d$ , град/В·с  | 1.4   |
| Коефіцієнт передачі магнітного підсилювача, $k_y$           | 44.1  |
| Стала часу магнітного підсилювача, $T_y$ , с                | 0.002 |
| Допустима тривалість перехідного процесу, $t_{п}$ , с       | 6     |
| Допустиме перерегулювання, $\sigma$ , %                     | 15    |

Розробку функціональної схеми САК почнемо з визначення передавальних функцій окремих ланок, що є у системі.

Піч сушильна. Рівняння печі має вид  $T_{п}\dot{\theta} + \theta = k_{п}\alpha$ , де  $\theta$  - температура у печі є вихідною або регульованою змінною, а  $\alpha$  - кут повороту вентиля газопроводу є вхідною змінною для цієї ланки.

Для отримання передавальної функції спочатку запишемо рівняння у стандартній формі, використовуючи оператор диференціювання  $p$ :

$$(T_{п}p+1)\theta = k_{п}\alpha.$$

Передавальна функція ланки дорівнює відношенню зображень Лапласа вихідної і вхідної змінних при нульових початкових умовах. Формально її можна отримати із диференційного рівняння розділивши множник при вихідній змінній у правій частині рівняння на множник при вихідній змінній у лівій частині, замінивши  $p$  на  $s$ . Тому для даної ланки будемо мати

$$W_{п}(s) = \frac{\theta(s)}{\alpha(s)} = \frac{k_{п}}{T_{п}s+1}.$$

Магнітний підсилювач. Його рівняння у стандартній формі має вигляд

$$(T_y p+1)u_2 = k_y u_1,$$

а передавальна функція дорівнює

$$W_y(s) = \frac{U_2(s)}{U_1(s)} = \frac{k_y}{T_y s+1}.$$

Чутливий елемент. Рівняння датчика температури є алгебраїчним

$$u_1 = k_T \Delta\theta,$$

де  $\Delta\theta = (\theta_3 - \theta)$ . Тому передавальна функція чутливого елемента дорівнює  $k_T$ :

$$W_{чЕ}(s) = \frac{U_1(s)}{\Delta\theta(s)} = k_T.$$

Виконавчий орган. З рівняння двигуна постійного струму  $(T_M p^2 + p)\alpha = k_d u_2$  отримуємо його передавальну функцію

$$W_d(s) = \frac{\alpha(s)}{U_2(s)} = \frac{k_d}{s(T_M s+1)}.$$

Виходячи з огляду рівнянь ланок, можна описати послідовність їх роботи в САК. Чутливий елемент вимірює температуру в печі, порівнює її із заданою і перетворює сигнал різниці у відповідну напругу  $u_1$ . Ця напруга посилюється магнітним підсилювачем і подається на вхід двигуна постійного струму, який

через редуктор змінює положення газової заслінки, впливаючи тим самим на температуру в печі. Таким чином, функціональна схема САК буде мати вид, показаний на рис. 2.1.

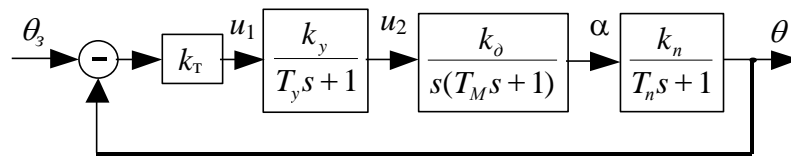




Рисунок. 2.1 – Функціональна схема САК

### Хід роботи

1. Викликати програму Matlab. Для запуску Simulink набрати відповідну команду в командному рядку Matlab або натиснути кнопку . Дочекається відкриття вікна Simulink Library Browser (вікно бібліотеки блоків) і натиснути кнопку  для відкриття нового (чистого) вікна моделі.

2. Ввести функціональну схему системи на рис. 2.2, яка відповідає САК зображеній на рис. 2.1 (параметри ланок задавати відповідно вказаного викладачем варіанта за табл. 2.3). Для цього у вікні бібліотеки відкрити розділ Simulink, потім відповідні підрозділи і перетягти необхідні блоки у вікно моделі за допомогою миші.

Ланки зі складними передавальними функціями задаються блоками **Transfer Fcn** (підрозділ **Continuous**). Підсилювач **Gain** і суматор **Sum** знаходяться в підрозділі **Math**. Блок **Step** формує ступінчатий вхідний вплив і знаходиться в підрозділі **Sources**. Блок **Scope** служить для відображення результатів моделювання на екрані у вигляді графіків (підрозділ **Sinks**).

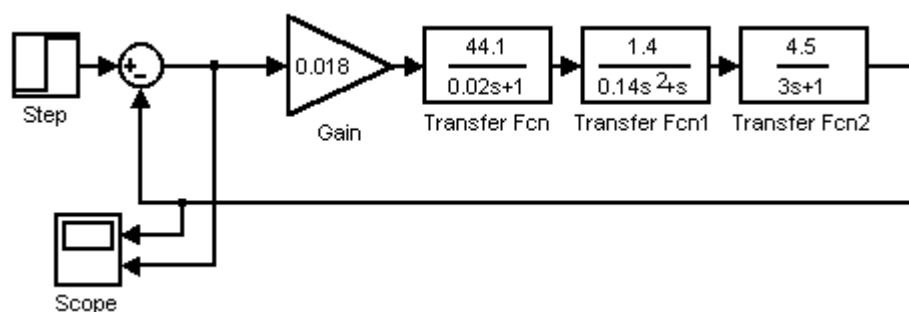


Рисунок 2.2 – Модель САК у Simulink


Таблиця 2.3

| Найменування величин,<br>їх позначення й розмірність                  | Варіант |       |       |       |       |       |       |       |       |       |
|---|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
|   | 1       | 2     | 3     | 4     | 5     | 6     | 7     | 8     | 9     | 10    |
| Стала часу печі, $T_{п}$ , с  | 4.0     | 2.0   | 3.5   | 2.8   | 3.4   | 3.2   | 4.0   | 2.8   | 2.8   | 4.0   |
| Коефіцієнт передачі печі, $k_{п}$ , °С/град                           | 6.0     | 5.0   | 6.0   | 7     | 4.5   | 5.0   | 6.0   | 6.0   | 5.5   | 5.2   |
| Коефіцієнт передачі резистивного мосту, $k_{т}$ , В/°С                | 0.015   | 0.01  | 0.02  | 0.012 | 0.017 | 0.016 | 0.014 | 0.02  | 0.015 | 0.014 |
| Електромеханічна стала часу двигуна з редуктором, $T_{м}$ , с         | 0.13    | 0.15  | 0.1   | 0.2   | 0.16  | 0.14  | 0.15  | 0.2   | 0.1   | 0.12  |
| Коефіцієнт передачі двигуна з редуктором, $k_{д}$ , град/В·с          | 2.0     | 1.6   | 2.1   | 1.7   | 1.3   | 1.4   | 1.8   | 1.5   | 2.0   | 1.6   |
| Коефіцієнт передачі магнітного підсилювача, $k_{γ}$                   | 45      | 50    | 43    | 40    | 54    | 46    | 48    | 52    | 42    | 48    |
| Стала часу магнітного підсилювача, $T_{γ}$ , с                        | 0.003   | 0.001 | 0.002 | 0.005 | 0.003 | 0.001 | 0.002 | 0.004 | 0.003 | 0.005 |
| Допустима тривалість перехідного процесу, $t_{п}$ , с                 | 6       | 6     | 6     | 6     | 6     | 6     | 6     | 6     | 6     | 6     |
| Допустиме перерегулювання, $σ$ , %                                    | 15      | 15    | 15    | 15    | 15    | 15    | 15    | 15    | 15    | 15    |
| Максимальна швидкість зміни температури, $\dot{\theta}_{\max}$ , °С/с | 8.0     | 7.0   | 9     | 6.5   | 8.5   | 6.0   | 7.0   | 9.0   | 9.0   | 7.0   |
| Допустима помилка управління, $\varepsilon_{\max}$ , °С               | 1.5     | 2.0   | 1.0   | 1.3   | 1.4   | 1.0   | 1.5   | 1.8   | 1.0   | 1.0   |

Для завдання або зміни параметрів ланки виділіть потрібний блок і натисніть клавішу **Enter** на клавіатурі (альтернатива – подвійне натискання лівої кнопки миші), у вікні властивостей, що з'явиться, введіть значення параметрів. Наприклад, вікно властивостей блока **Transfer Fcn** показано на рис. 2.3. Воно відповідає ланці з передавальною функцією виду

$$W(s) = \frac{1.4}{s(0.14s + 1)} = \frac{1.4}{0.14s^2 + s}$$

Чисельник і знаменник передавальної функції задаються відповідно у полях **Numerator** і **Denominator** у вигляді вектора-рядка, що записується у квадратних дужках і утримує коефіцієнти полінома у порядку зменшення ступеня  $s$ . Значення коефіцієнтів відділяються одне від одного пробілом.

Параметри суматора задаються шляхом введення необхідної кількості знаків «+» і «-» в полі **List of signs** вікна його властивостей. Подвійне натискання лівої кнопки миші на блоці **Scope** відкриває вікно графіка (осцилограми), вид якого показано на рис. 2.4. Для установки двох входів блока натисніть кнопку  і у вікні, що з'явиться, введіть кількість осей 2 у полі **Number of axes** вкладки **General**. На вкладці **Data history** зніміть обмеження **Limit data**.

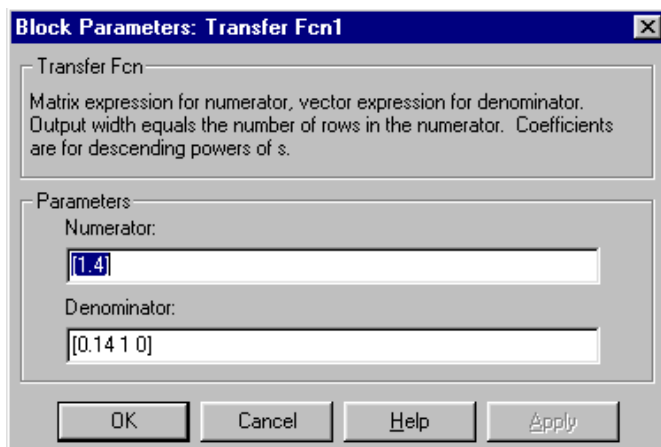


Рисунок 2.3

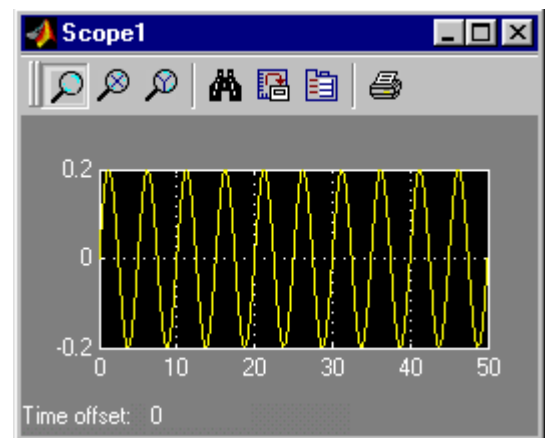



Рисунок 2.4

Для блока **Step** задайте параметр **Step time** рівним нулю.

3. Задайте час моделювання 50 с, для чого оберіть пункт меню **Simulation**→**Configuration parameters**, у вікні, що з'явиться, введіть цифру 50 в поле **Stop time** вкладки **Solver**, закрийте вікно. Проведіть моделювання, виконавши пункт меню **Simulation**→**Start** або натиснувши кнопку  на панелі інструментів. Скопіюйте графіки перехідного процесу в звіт.

4. Побудуйте частотні характеристики розімкненої системи. Для чого розімкніть головний зворотній зв'язок – захватіть мишею стрілку біля знака мінус суматора і відтягніть її трохи вниз. В Matlab версії нижче 7-ї виконайте пункт меню **Tools**→**Linear analysis...**. З'явиться два нових вікна: **LTI Viewer** і **Model Inputs and Outputs**. Перше є вікном перегляду характеристик досліджуваної системи, зверніть його поки що. Друге вікно утримує два блоки:



**Input Point** і **Output Point**, що вказують пакету лінійного аналізу вхід і вихід системи. Захватіть мишею блок **Input Point** і опустіть його на лінію, що виходить з суматора, а блок **Output Point** опустіть на лінію, що виходить з останнього блоку системи. Ви отримаєте схему, зображену на рис. 2.5.

В Matlab версії 7.x і вище **Input Point** і **Output Point** вставляються за допомогою контекстного меню, що виникає при натисканні правої кнопки миші на необхідній з'єднувальній лінії. Треба обрати пункт цього меню «**Linearization Points**» або «**Linear Analysis Points**» і в додатковому меню обрати потрібний порт. Аналог **LTI Viewer** тут викликається через пункт меню «**Tools→Control Design→Linear analysis**» або «**Analysis→Control Design→Linear analysis**».

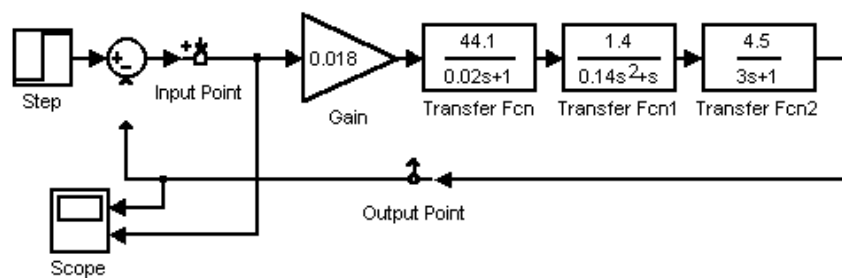


Рисунок 2.5

В Matlab версії нижче 7-ї розверніть вікно **LTI Viewer** і виконайте в ньому пункт меню **Simulink→Get Linearised Model**. З'явиться графік однієї з характеристик розімкненої системи. Викличте правою кнопкою миші контекстне меню і виберіть **Plot Type→Bode** для виводу логарифмічних характеристик. Пункт контекстного меню **Characteristics→Stability Margins** виводить на графік маркери, що відповідають запасу стійкості по фазі і амплітуді. Підведіть курсор до кожного з маркерів і переписіть значення запасів, що при цьому з'являться на екрані. Зверніть, але не закривайте, вікно **LTI Viewer**.

В Matlab версії 7.x і вище вид вікна **LTI Viewer** змінюється від версії до версії. Тому аналог команди **Get Linearised Model** треба шукати, ретельно переглядаючи меню і вкладки вікна **LTI Viewer**.

5. Розірвіть зв'язок між суматором і підсилювачем, а на його місце вставте коригувальний пристрій (КП), схема якого показана на рис. 2.6. Замкніть головний зворотній зв'язок.

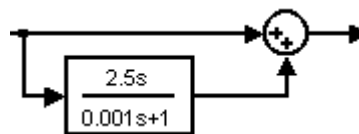


Рисунок 2.6

Сталу часу в знаменнику задайте меншою від найменшої сталої часу вашої системи, наприклад, 0.001 с. Інтенсивність введення похідної (коефіцієнт підсилення КП) підберіть так, щоб забезпечити задані в табл. 2.3 показники якості перехідного процесу  $t_n$  і  $\sigma$ . Для цього кожного разу після зміни параметра КП проведіть розрахунок перехідного процесу в системі.

Скопіюйте графіки зміни вихідного сигналу і помилки керування скоригованої системи в звіт, визначте сталі значення помилки керування.

6. Замість блока **Step** вставте в систему блок **Ramp** (підрозділ **Sources** бібліотеки **Simulink**), який формує лінійний сигнал, що зростає з постійною швидкістю. Проведіть моделювання системи при трьох значеннях швидкості зростання температури в печі  $\dot{\theta}_{\max}$ ,  $0.5\dot{\theta}_{\max}$  і  $2\dot{\theta}_{\max}$ . Скопіюйте графіки перехідних процесів і визначте сталі значення помилки керування.

7. Замість блока **Ramp** вставте блок **Sine Wave**, що формує гармонічний сигнал. Проведіть моделювання системи при двох значеннях частоти вхідного впливу: 1 рад/с і 0.05 рад/с. Зарисуйте графіки перехідних процесів і визначте значення амплітуди сталої помилки керування. При необхідності збільште час моделювання до 150 – 200 с.

8. Побудуйте логарифмічні характеристики розімкненого кола скоригованої системи, для чого розімкніть головний зворотній зв'язок, перейдіть у вікно **LTI Viewer** і виконайте пункт меню **Simulink→Get Linearised Model**. Порівняйте характеристики скоригованої і не скоригованої систем. Скопіюйте їх. Випишіть нові значення запасів стійкості.

9. Збережіть введену модель на магнітному диску. Закрийте Matlab. Оформіть звіт.

### Контрольні запитання

1. Як ввести функціональну схему САК в систему Simulink?
2. Як задати або змінити параметри моделювання і блоків системи?
3. Як проводиться контроль результатів моделювання?
4. Як побудувати частотні характеристики системи?
5. Як впливає вид і швидкість зміни вхідного сигналу на величину сталої помилки керування?

### Лабораторна робота № 3

#### Дослідження впливу збурюючого сигналу на реакцію системи і величину помилки керування

Ціль роботи: Дослідити залежність реакції системи та помилки управління від характеру збурюючого впливу.

Потрібне обладнання і програмне забезпечення: ПЕОМ класу Pentium, або вище; операційна система Windows XP або вище; математичний пакет Matlab 5.3 або вище.

#### Теоретичні відомості

Окрім вхідного задавального впливу, який система повинна відтворити на виході, на систему можуть діяти різні сили, що перешкоджають точній роботі системи. Ці сили називаються збурюючими впливами. Реакція системи на збурюючий вплив повністю йде в похибку керування, тому задача системи – не пропустити збурюючий сигнал на вихід, придумати його якомога сильніше.

В системі управління температурним режимом сушильної газової печі, що розглядалася вище, в якості збурюючих впливів можуть виступати, наприклад, зміни температури зовнішнього середовища, або коливання тиску в газопроводі. Причому коливання тиску газу мають більший вплив на систему.

У першому наближенні зміну тиску в газопроводі можна врахувати, переписавши рівняння печі (див. табл. 1) у вигляді

$$T_{\Pi} \dot{\theta} + \theta = k_{\Pi} \alpha + k_P \Delta P,$$

де  $\Delta P$  – відхилення тиску від номінального значення, при якому було визначено параметр  $k_{\Pi}$ ;  $k_P$  – коефіцієнт передачі по тиску, що залежить від конструкції печі.

Тоді піч можна представити як ланку з двома входами і, відповідно, з двома передавальними функціями (рис. 3.1,а), або як комбінацію двох ланок з тими ж передавальними функціями (рис. 3.1,б).

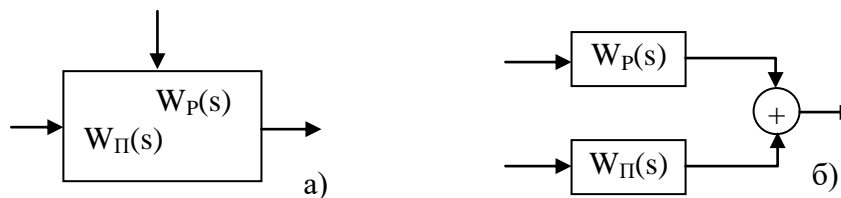


Рисунок 3.1

Передавальна функція  $W_{\Pi}(s)$  по куту повороту вентиля газопроводу була визначена у роботі 2. Не важко бачити, що передавальна функція  $W_P(s)$  по тиску буде мати аналогічний вигляд:

$$W_P(s) = \frac{\theta(s)}{\Delta P(s)} = \frac{k_P}{T_{\Pi}s + 1}.$$

Проведемо дослідження впливу характеру зміни тиску  $\Delta P$  на помилку керування. Припустимо, що тиск у газопроводі підтримується з точністю  $\pm 0.2$  атмосфери. Причому тиск може коригуватися скачками, змінюватися по лінійному закону, або по синусоїдальному. Нехай коефіцієнт передачі по тиску дорівнює  $k_p = 100$  °C/Ат.

Схема Simulink-моделі СУТР, побудована з урахуванням вищесказаного, показана на рис. 3.2. Тут для моделювання коливань тиску використано блок **Signal Generator**, який генерує коливання прямокутної (рис. 3.3,а), трикутної (рис. 3.3,б) і синусоїдальної форми (рис. 3.3,в) різної частоти. Блок **Transport Delay** використано для затримки збурюючого впливу у часі до завершення перехідного процесу від задавального впливу.

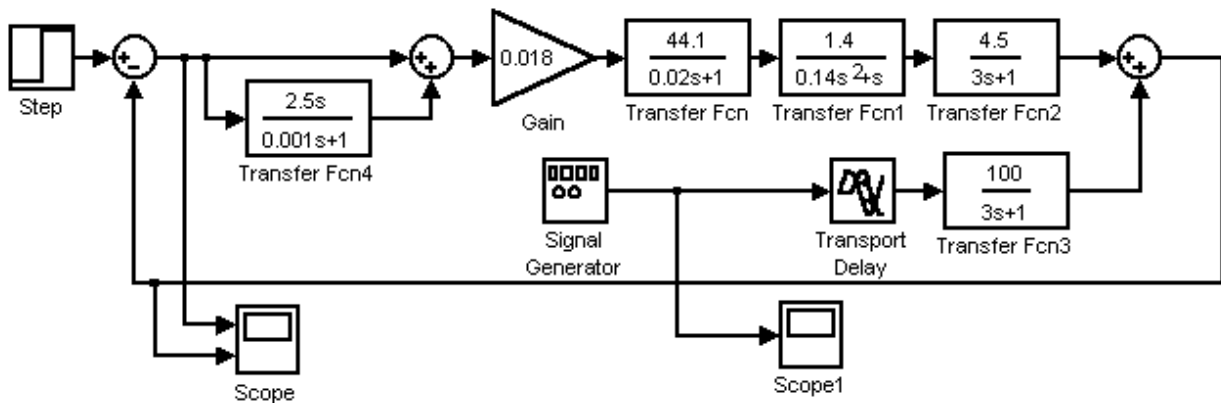


Рисунок 3.2

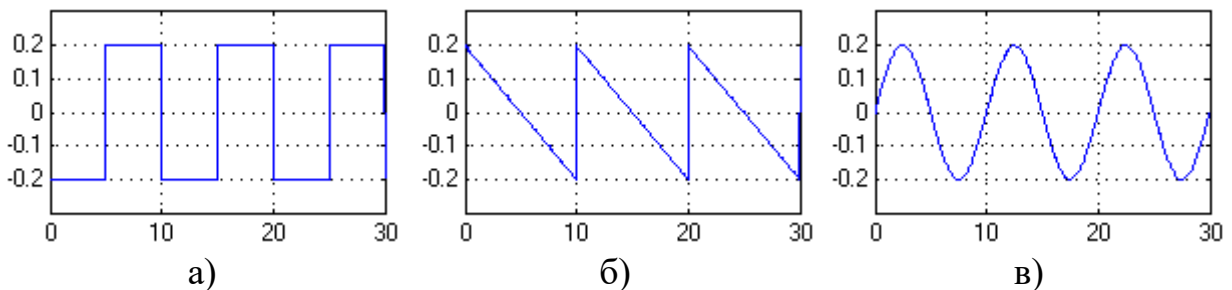


Рисунок 3.3

На рис. 3.4, 3.5 і 3.5 показана зміна у часі помилки керування і реакції системи при синусоїдальному, прямокутному і трикутному характері збурюючого впливу з амплітудою  $0.2$  Ат і двох різних значеннях частоти. Аналіз цих рисунків показує, що найгірші результати в роботі системи отримуються при стрибкоподібній зміні тиску. При цьому амплітуда помилки залишається практично незмінною з ростом частоти, бо величина стрибка тиску не міняється. При зменшенні частоти система встигає відпрацювати зміну тиску приблизно за  $10$  с. При прямокутній формі впливу помилка зменшується до нуля, а при трикутному впливі – до деякого постійного значення, що залежить від швидкості зниження тиску. Але кожен новий стрибок тиску призводить до нового стрибка помилки. Такий режим роботи печі в деяких технологічних процесах не є прийнятним.

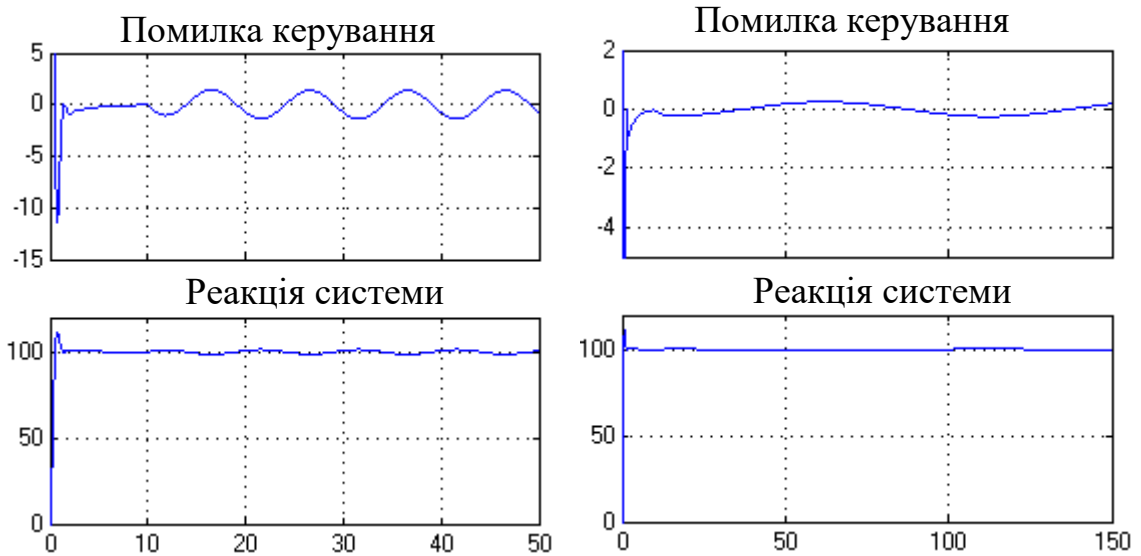


Рисунок 3.4 – Синусоїдальна зміна тиску

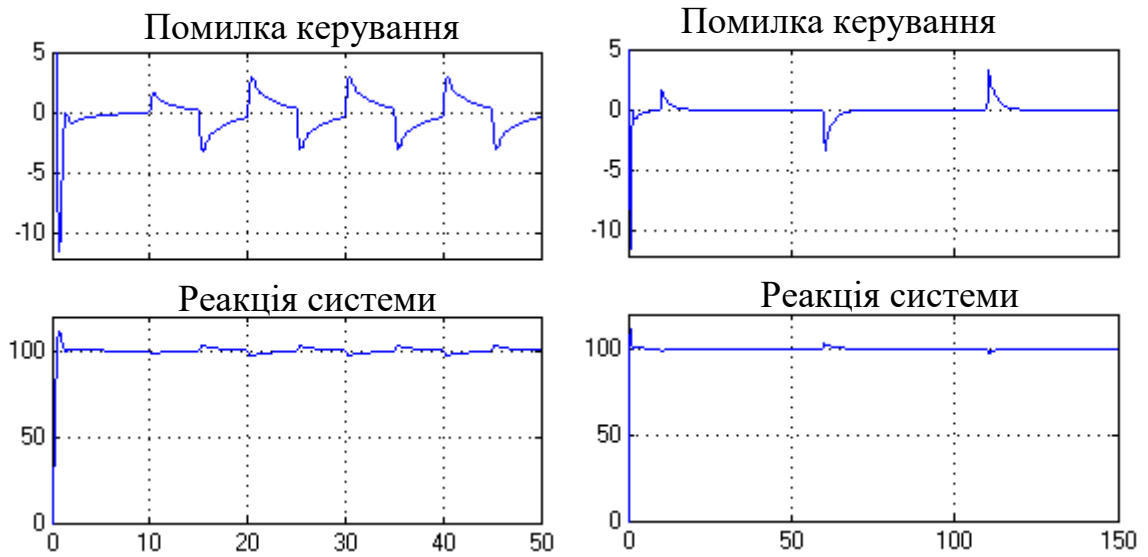


Рисунок 3.5 – Прямокутна зміна тиску

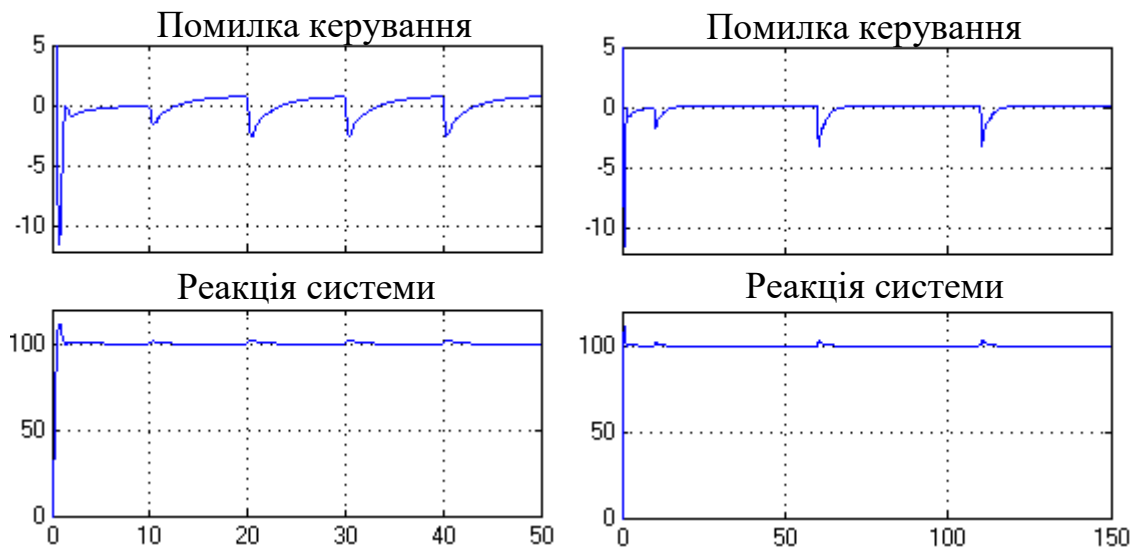




Рисунок 3.6 – Трикутна зміна тиску

Коли тиск змінюється по синусоїді, то і помилка керування теж змінюється по синусоїді, і зі зниженням частоти амплітуда її зменшується, починаючи з деякого значення частоти помилка стає менше допустимого значення. Таким чином, при регулюванні тиску в газопроводі треба надавати перевагу плавним законам його зміни, без стрибків.

### Хід роботи

1. Викликати програму Matlab. Для запуску Simulink набрати відповідну команду в командному рядку Matlab або натиснути кнопку . Дочекайтесь відкриття вікна Simulink Library Browser (вікно бібліотеки блоків) і натиснути кнопку  для відкриття нового (чистого) вікна моделі. Ввести функціональну схему системи на рис. 3.2 (якщо ви зберегли модель, розроблену у другій лабораторній роботі, відкрийте її і додайте необхідні блоки). Параметри блоків задавати згідно з вказаним варіантом за табл. 2.3. Коефіцієнт передачі по тиску встановити рівним  $k_p=100$  °C/Ат.

2. У вікні властивостей блоку **Step** задайте **Step time** рівним нулю, **Final value** рівним 100, що відповідає кінцевій температурі 100 °C. У вікні властивостей блоку **Signal Generator** у полі **Wave form** виберіть **square** (прямокутний), амплітуду задайте 0.2, а частоту (Frequency) 0.1 Гц. Для блока **Transport Delay** параметр **Time delay** задайте 15÷20 с, в залежності від часу затухання перехідного процесу в вашій системі. Промодельуйте систему на інтервалі часу 50÷60 с. Скопіюйте графіки помилки керування та реакції (вихідної змінної) системи. Змінюючи масштаб графіка, визначте амплітуду помилки й час її затухання до нуля. Зменшіть частоту збурюючого впливу до 0.01 Гц, а час моделювання збільшіть до 150 с. Проведіть повторне дослідження системи. Скопіюйте графіки, визначте амплітуду помилки.

3. У вікні властивостей блоку **Signal Generator** у полі **Wave form** виберіть **sawtooth** (пилкоподібний), амплітуду задайте 0.2, а частоту (Frequency) 0.1 Гц. Промодельуйте систему на інтервалі часу 50÷60 с. Зарисуйте графіки помилки керування та реакції системи. Визначте амплітуду і стале значення помилки між імпульсами, а також час її затухання. Зменшіть частоту збурюючого впливу до 0.02 Гц, а час моделювання збільшіть до 150 с. Проведіть повторне дослідження системи. Скопіюйте графіки, визначте амплітуду і стале значення помилки.

4. У вікні властивостей блоку **Signal Generator** у полі **Wave form** виберіть **sine** (синусоїдальний), амплітуду задайте 0.2, а частоту (Frequency) 0.1 Гц. Промодельуйте систему на інтервалі часу 50÷60 с. Скопіюйте графіки помилки керування та реакції системи. Визначте амплітуду помилки. Зменшіть частоту збурюючого впливу до 0.01 Гц, а час моделювання збільшіть до 150 с. Проведіть повторне дослідження системи. Скопіюйте графіки, визначте амплітуду помилки.

Проведіть моделювання декілька разів на різних частотах. Визначте граничну частоту коливань тиску, при якій помилка керування буде дорівнювати заданій в табл. 2.3

5. Закрийте Matlab. Оформіть звіт.

#### Контрольні запитання

1. Які впливи на систему називаються збурюючими? Наведіть приклади.
2. Як система повинна реагувати на збурюючі впливи?
3. Який режим зміни тиску газу найбільш безпечний з точки зору підтримання постійної температури в печі?

## Лабораторна робота № 4 Симуляція кулі, що підстрибує

Ціль роботи: Навчитися використовувати різні підходи, щоб моделювати кулю, що підстрибує, з використанням Simulink.

Потрібне обладнання і програмне забезпечення: ПЕОМ класу Pentium, або вище; операційна система Windows XP або вище; математичний пакет Matlab 5.3 або вище.

### Теоретичні відомості

Розглянемо приклад з довідкової системи SIMULINK: Модель кулі, що підстрибує, є класичним прикладом гібридної динамічної системи. Гібридна динамічна система – це система, що включає як безперервну динаміку, так і дискретні переходи, де динаміка системи може змінюватися, а значення змінних стану можуть стрибати.

Задача формулюється наступним чином. Куля, що знаходиться на висоті  $x_0 = 10$  м, підкидають уверх з початковою швидкістю  $v_0 = 15$  м/с (див. рис. 4.1). Опір повітря не враховується. Особливістю задачі є те, що у момент стикання кулі з землею відбувається зміна напрямку руху – куля відскакує.

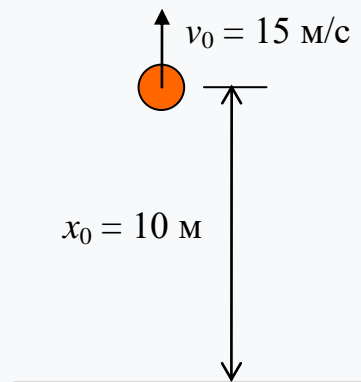


Рисунок 4.1

Безперервна динаміка стрибаючої кулі задається просто:

$$\frac{dv}{dt} = -g \quad \text{і} \quad \frac{dx}{dt} = v. \quad (4.1)$$

де  $g$  – прискорення, спричинене силою тяжіння,  $x$  і  $v$  – положення кулі та швидкість. Отже, система має дві безперервних змінні стану: положення та швидкість. Гібридний системний аспект моделі бере початок із моделювання зіткнення кулі з землею. Якщо припустити частково еластичне зіткнення із землею, то швидкість до зіткнення  $v^-$  та швидкість після зіткнення  $v^+$  можуть бути пов'язані коефіцієнтом  $k$  відновлення кулі наступним чином:



$$v^+ = -kv^- \text{ при } x=0. \quad (4.2)$$

Отже, підстрибуюча куля відображає стрибок у безперервній змінній стану (швидкості) за умови переходу через  $x=0$ .

Підстрибуюча куля – одна з найпростіших моделей, яка показує явище Зено. Поведінка Зено неофіційно характеризується нескінченною кількістю подій, що відбуваються в кінцевому інтервалі часу для певних гібридних систем. Оскільки куля, що підстрибує, втрачає енергію в моделі велика кількість зіткнень із землею починається через послідовно менші проміжки часу. Звідси модель переживає поведінку Зено. Моделі з поведінкою Зено за своєю суттю важко моделювати на комп'ютері, але вони зустрічаються у багатьох загальних та важливих інженерних додатках.

Ви можете використовувати два блоки інтегратора для моделювання стрибаючої кулі (див. рис. 4.2). Інтегратор ліворуч – це інтегратор швидкості, що моделює перше рівняння в (4.1), а інтегратор праворуч – інтегратор положення (друге рівняння в (4.1)). Зауважте, що діалогове вікно блоку інтегратора позицій повинно мати нижню межу нуль. Ця умова є обмеженням того, що м'яч не може опускатися нижче землі.

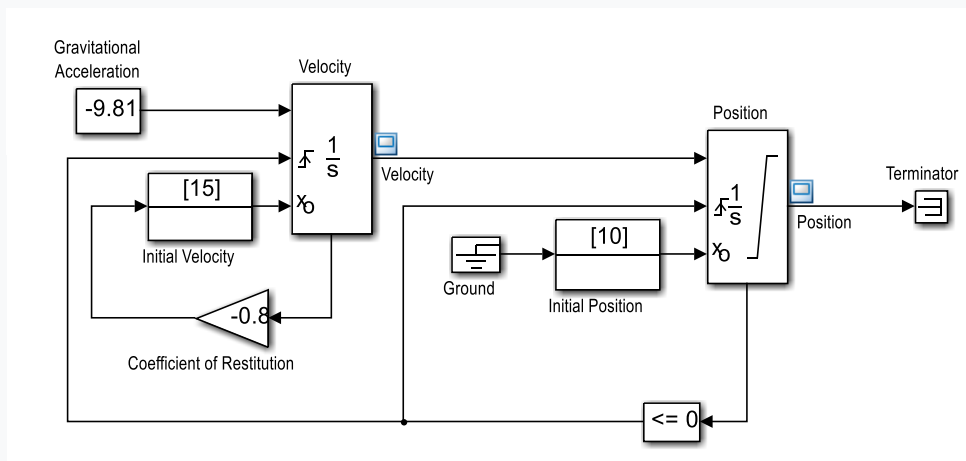


Рисунок 4.2

Порт стану інтегратора позицій та відповідний результат порівняння використовуються для виявлення, коли кулька потрапляє на землю, і для скидання обох інтеграторів. Порт стану інтегратора швидкості використовується для обчислення  $v^+$ .

Результати моделювання стрибків кулі на протязі 20 секунд показані на рис. 4.3.

Щоб спостерігати за поведінкою системи Зено у системі, треба перейти до панелі **Solver** діалогового вікна **Configuration Parameters**. У розділі **Zero-crossing options** («Параметри нульового перетину») у полі «**Algorithm**» встановити значення «Nonadaptive» («Неадаптивний»), а для моделювання час зупинки встановити 25 секунд (див. рис. 4.4). Потім запуснути моделювання.

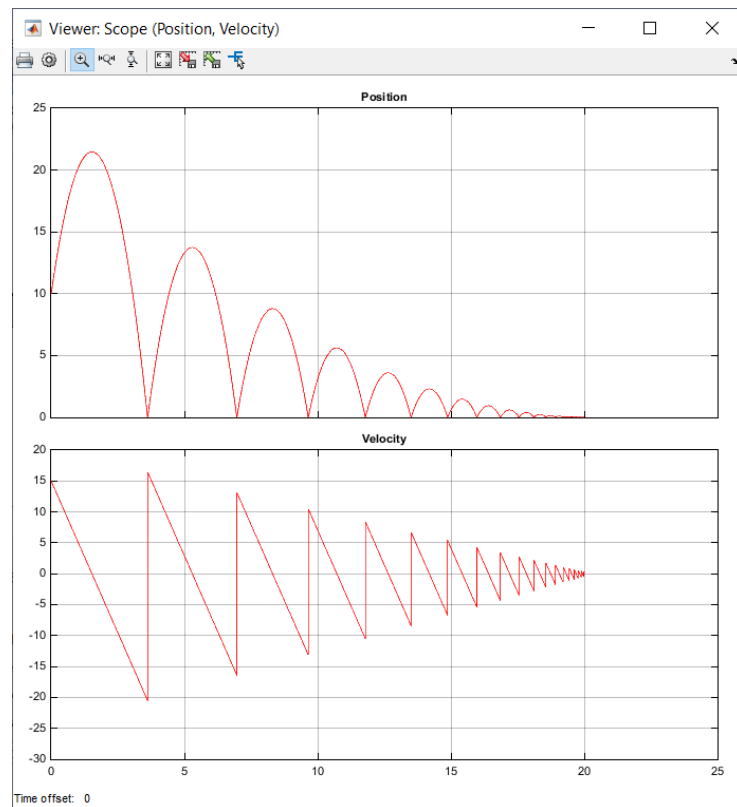


Рисунок 4.3

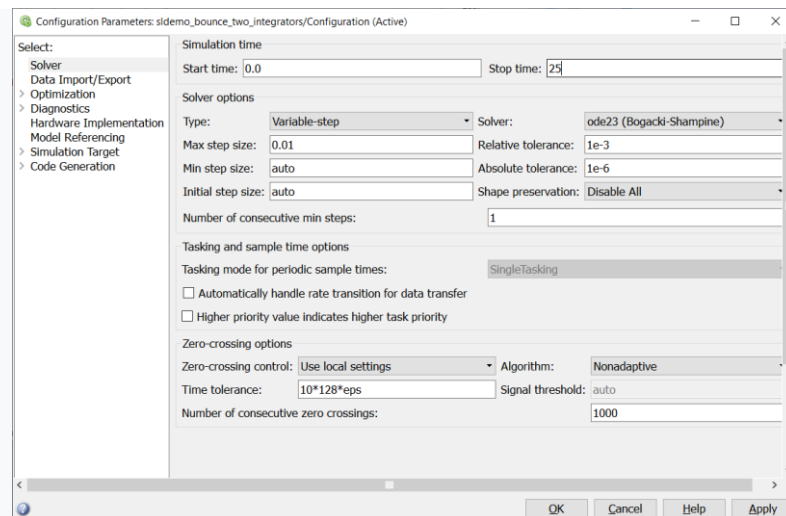


Рисунок 4.4

Зверніть увагу, що коли м'яч все частіше і частіше падає на землю, він втрачає енергію. І коли моделювання перевищує типову межу в 1000 для дозволеної кількості послідовних нульових перетинів, Matlab завершує симуляцію і видає повідомлення про помилку (див. рис. 4.5).

Цю ситуацію можна перебороти, якщо перейти до діалогового вікна **Configuration Parameters** і у розділі **Zero-crossing options** встановити у полі «**Algorithm**» значення «Adaptive» («Адаптивний»). Цей алгоритм запроваджує складне трактування такої бовтанки. Отже, тепер ви можете імітувати систему довше 20 секунд. Однак зверніть увагу на бовтанку змінних стану між 20 і 25

секундами (див. рис. 4.6, де кінець процесу показано у збільшеному масштабі) та попередження від Simulink про сильну бовтанку в моделі близько 20 секунд.

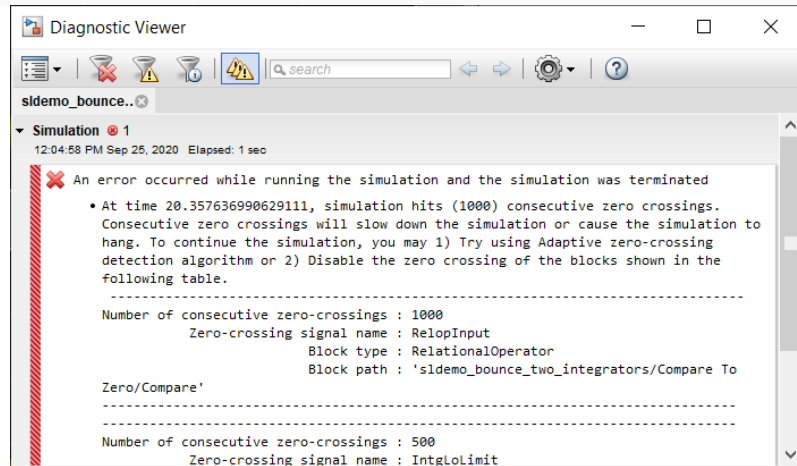


Рисунок 4.5

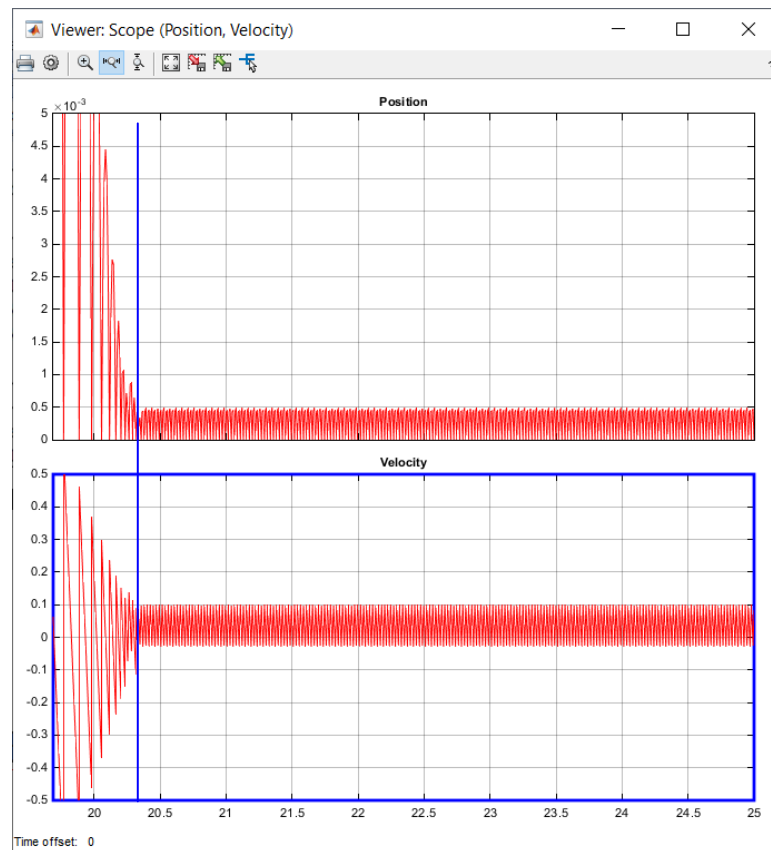


Рисунок 4.6

Починаючи з Matlab R2014, ви можете використовувати один блок інтегратора другого порядку (**Second-Order Integrator**) для моделювання цієї системи (див. рис. 4.7). У цьому випадку друге рівняння є внутрішнім для блоку інтегратора другого порядку. Перейдіть до діалогового вікна блоку **Second-Order Integrator** і встановіть, як і раніше, нижню межу рівною нулю. Перейдіть на вкладку «**Attributes**» у діалоговому вікні блоку та зауважте, що встановлено прапорець «**Reinitialize  $dx/dt$  when  $x$  reaches saturation**» («Повторно

ініціалізувати  $dx/dt$ , коли  $x$  досягає насичення»). Цей параметр дозволяє нам повторно ініціалізувати ( $v$  у моделі підстрибуючої кулі) нове значення в той момент, коли  $x$  досягає межі насиченості. Отже, для моделі стрибаючої кулі цей варіант передбачає, що коли куля вдаряється об землю, його швидкість може бути встановлена на інше значення, тобто на швидкість після удару. Зверніть увагу на петлю для розрахунку швидкості після зіткнення з землею. Для фіксації швидкості кульки безпосередньо перед зіткненням використовуються вихідний порт блоку інтегратора другого порядку та блок пам'яті. Потім використовується для обчислення швидкості відскоку.

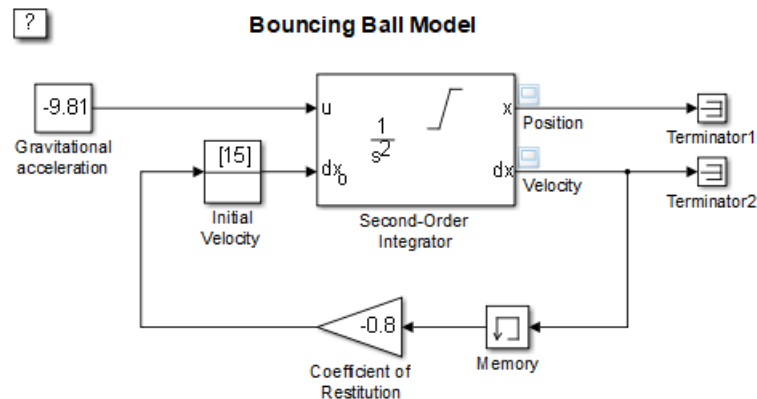


Рисунок 4.7

Перейдіть до панелі **Solver** діалогового вікна **Configuration Parameters**. Переконайтеся, що для параметра «**Algorithm**» встановлено значення «Nonadaptive» у розділі **Zero-crossing options**, а для моделювання час зупинки встановлено 25 секунд. Імітуйте модель. Зверніть увагу, що моделювання не зазнало жодних проблем. Ви змогли змодельовати модель, не відчувачи надмірної бовтанки через  $t = 20$  секунд і не встановивши для «**Algorithm**» значення «Adaptive».

Можна аналітично розрахувати точний час, коли кулька осідає на землі з нульовою швидкістю, підсумовуючи час, необхідний для кожного відскоку. Цей час є сумою нескінченного геометричного ряду, що дорівнює

$$t^* = \frac{1}{g} \left( v_0 + v_1 \frac{1+k}{1-k} \right), \quad \text{де } v_1 = \sqrt{v_0^2 + 2gx_0}.$$

Тут  $x_0$  і  $v_0$  – початкові умови положення і швидкості відповідно. Швидкість і положення кулі повинні бути однаково нульовими для  $t > t^*$ .

На рис. 4.8 показано кінець процесу в моделі рис. 4.7 у збільшеному масштабі для порівняння з процесом на рис. 4.6, отриманим в моделі рис. 4.2. Вертикальна синя лінія на графіках рис. 4.6 і 4.8 відповідає значенню  $t^*$  при заданих параметрах моделі. Для  $t < t^*$  і далеко від  $t^*$  обидві моделі дають точні і однакові результати. Однак результати моделювання з першої моделі є неточними після  $t > t^*$ ; вона продовжує відображати надмірну бовтанку для  $t > t^*$ . На відміну від неї, друга модель, що використовує блок інтегратора другого порядку, осідає рівно до нуля при  $t > t^*$ .

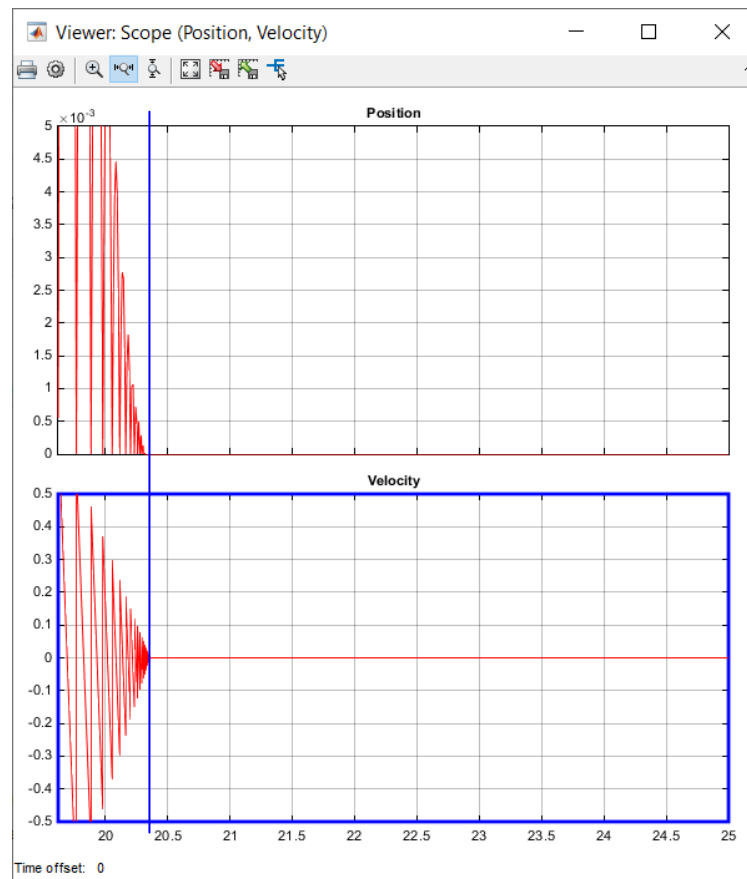




Рисунок 4.8

Рис. 4.6 та 4.8 остаточно показують, що друга модель має кращі числові характеристики в порівнянні з першою моделлю. Причина вищої точності, пов'язаної з моделлю інтегратора другого порядку, полягає в наступному. Друге диференціальне рівняння є внутрішнім для блоку інтегратора другого порядку. Отже, блокові алгоритми можуть використовувати цей відомий зв'язок між двома станами та застосовувати евристику, щоб припинити небажану поведінку бовтанки для певних умов. Ці евристики стають активними, коли два стани перестають взаємно узгоджуватися між собою через помилки інтегрування та бовтанку. Таким чином, ви можете використовувати фізичні знання системи, щоб полегшити проблему симуляції, яка застряє в стані Зено для певних класів моделей Зено.

### Хід роботи

1. Викликати програму Matlab. Для запуску Simulink набрати відповідну команду в командному рядку Matlab або натиснути кнопку . Дочекатися відкриття вікна Simulink Library Browser (вікно бібліотеки блоків) і натиснути кнопку  для відкриття нового (чистого) вікна моделі. Ввести функціональну схему системи на рис. 4.2.

Для цього з розділу бібліотеки **Simulink**→**Continuous** скопіюйте у вікно моделі два блока інтеграторів (**Integrator**). Змініть ім'я першого інтегратора на «Velocity» (швидкість), а другого – на «Position». Відкрийте вікна властивостей

інтеграторів і встановіть поля параметрів блоків, як показано на рис. 4.9 (а і б). У полі «**External reset**» («Зовнішнє скидання») оберіть значення «rising» («при зростанні»). У полі «**Initial condition source**» («Джерело початкових умов») оберіть значення «external» («зовнішній»). При цьому під основним входом інтеграторів з'являться додаткові входи для сигналів скидання значень і встановлення початкових умов змінної стану. Також у обох інтеграторів встановіть чекер «**Show state port**» для відкриття додаткового виходу змінної стану, який використовується для запобігання створення алгебраїчної петлі при введенні зворотного зв'язку для скидання значень інтеграторів. У інтегратора «Position» встановіть чекер «**Limit output**» і у полі «**Lower saturation limit**» встановіть нульове значення для обмеження сигналу положення тільки додатними значеннями.

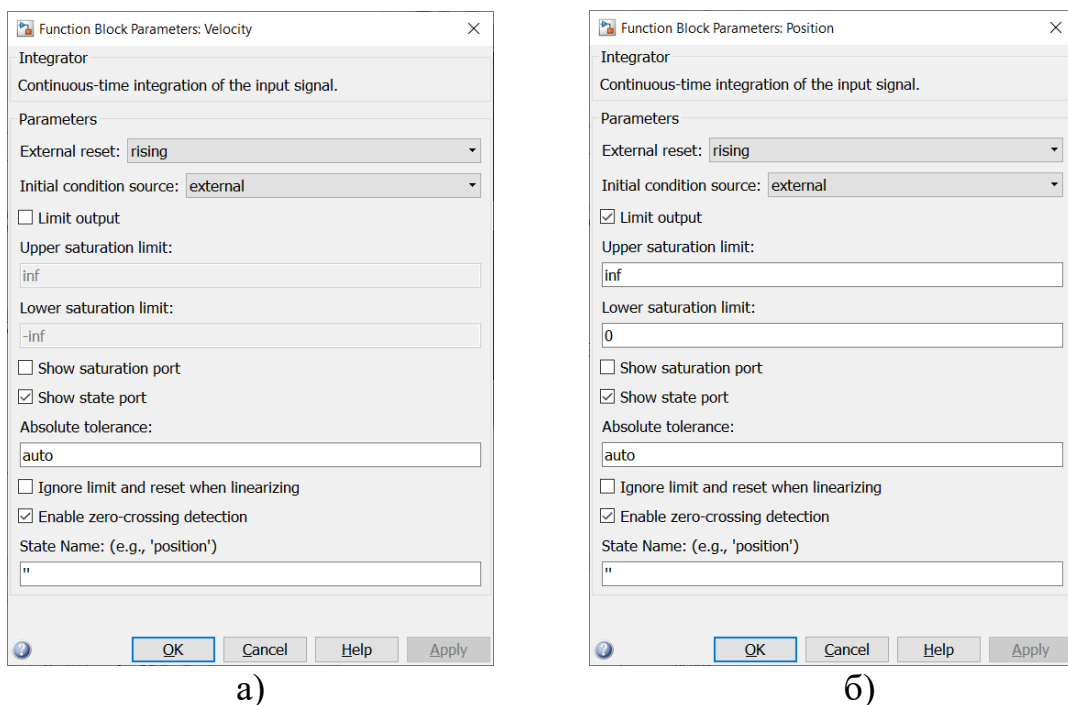


Рисунок 4.9

На сигнальний вхід першого інтегратора подайте сигнал з виходу джерела постійного сигналу **Constant** (розділ бібліотеки **Simulink**→**Sources**). Значення сигналу встановіть рівним прискоренню сили тяжіння ( $9.81 \text{ м/с}^2$ ) зі знаком мінус. Для завдання початкових умов використовуйте блоки **IC** з розділу бібліотеки **Simulink**→**Signal Attributes**.

Для завдання коефіцієнту відновлення швидкості використайте блок **Gain** з розділу бібліотеки **Simulink**→**Math Operations**. Для переустановки початкових умов блоку позиції використайте блок **Ground** (розділ бібліотеки **Simulink**→**Sources**).

Для формування сигналу скидання інтеграторів при стиканні кулі із землею використайте блок **Compare To Zero** з розділу бібліотеки **Simulink**→**Logic and Bit Operations** і встановіть його параметри, як показано

на рис. 4.10. З'єднайте блоки, як показано на схемі рис. 4.2. Вихід другого інтегратора подайте на блок **Terminator** з розділу бібліотеки **Simulink**→**Sinks**.

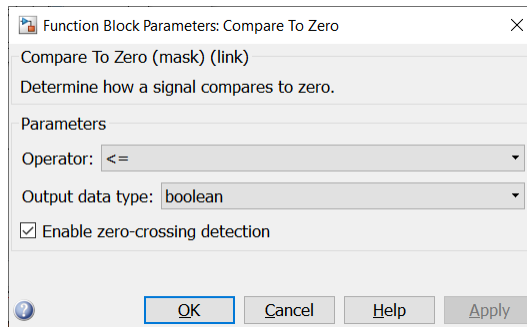


Рисунок 4.10

Для реєстрації результатів моделювання з'єднайте виходи інтеграторів з входами осцилографу з двома осями, або додайте блок **Floating Scope**, встановіть кількість осей 2, зніміть обмеження на кількість точок графіку, укажіть на кожне з координатних вікон мишею і правою кнопкою миші викликайте контекстне меню. Оберіть пункт **Signal selection**, у з'явившомуся вікні оберіть потрібний сигнал для відображення і нажміть кнопку **Close**.

Починаючи з Matlab R2014 для цього є інша можливість. Виділіть мишею лінію потрібного сигналу і правою кнопкою миші викликайте контекстне меню. Оберіть пункт меню **Create & Connect Viewer**→**Simulink**→**Scope**. Над лінією сигналу з'явиться значок осцилографу. Подвійний клік на цьому значку відкриє вікно відповідного осцилографа. У ньому також можна встановити потрібну кількість осей і додати потрібний сигнал на потрібну вісь. Є і інші шляхи додавання сигналів до цього осцилографу.

2. Проведіть моделювання системи, як описано починаючи від рис. 4.2 і по рис. 4.6. Скопіюйте схему моделі і графіки процесу у звіт.

3. Відкрийте вікно нової моделі і побудуйте схему з подвійним інтегратором відповідно рис. 4.7. Проведіть моделювання з цією схемою, як описано після рис. 4.6 і до кінця теоретичної частини. Зкопіюйте графіки і схему у звіт. Порівняйте результати моделювання (їх точність) за допомогою двох схем.

### Контрольні запитання

1. Яка поведінка системи називається поведінкою Зено?
2. Як можна задати початкові умови інтегратора?
3. Як обмежити вихідний сигнал інтегратора?
4. Як можна заново ініціювати інтегратор під час моделювання?

## Лабораторна робота № 5

### Особливості дослідження системи з ковзним процесом

Ціль роботи: Навчитися моделювати нелінійну систему з ковзним процесом з використанням Simulink.

Потрібне обладнання і програмне забезпечення: ПЕОМ класу Pentium, або вище; операційна система Windows XP або вище; математичний пакет Matlab 5.3 або вище.

#### Теоретичні відомості

Нехай задана система автоматичного регулювання (рис. 5.1а) з нелінійністю релейного типу (рис. 5.1б). Рівняння динаміки її мають вигляд

$$\begin{cases} p^2 x = k_1 x_2, & x_2 = F(x_1) = c \cdot \text{sign } x_1 \\ x_1 = -x - x_{33} = -(1 + k_{33} p)x \end{cases}$$

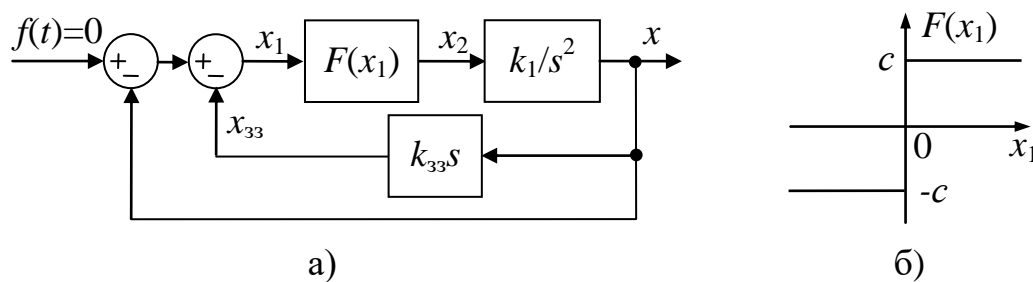


Рисунок 5.1 – Система з ковзним процесом

Ці рівняння можна представити у вигляді

$$\frac{dx}{dt} = y, \quad \frac{dy}{dt} = -ck_1 \text{sign}(x + k_{33}y). \quad (5.1)$$

Розділивши друге рівняння (5.1) на перше, отримаємо диференціальне рівняння фазових траєкторій

$$\frac{dy}{dx} = -\frac{k_1 c}{y} \text{sign}(x + k_{33}y). \quad (5.2)$$

Функція  $\text{sign}(x + k_{33}y) = 1$  при  $x + k_{33}y > 0$  й  $\text{sign}(x + k_{33}y) = -1$  при  $x + k_{33}y < 0$ . При переході  $x + k_{33}y$  через нуль похідна  $dy/dx$  міняє знак, тому лінія, що відповідає рівнянню

$$x + k_{33}y = 0 \quad \text{або} \quad x = -k_{33}y \quad (5.3)$$

називається лінією перемикування. У даному випадку (5.3) це пряма лінія, що проходить через початок координат (рис. 5.2). Праворуч від цієї лінії  $x + k_{33}y > 0$ , тому рівняння фазової траєкторії має вигляд



$$\frac{dy}{dx} = -\frac{k_1 c}{y} \quad \text{або} \quad y dy = -k_1 c dx.$$

Після інтегрування одержуємо:  $y^2 = -2k_1 c x + C_1$ . Тобто це параболи, гілки яких спрямовані в негативну сторону осі  $x$  (див. рис. 5.2). Положення вершини параболи визначається довільною постійною інтегрування  $C_1$ , тобто початковими умовами.

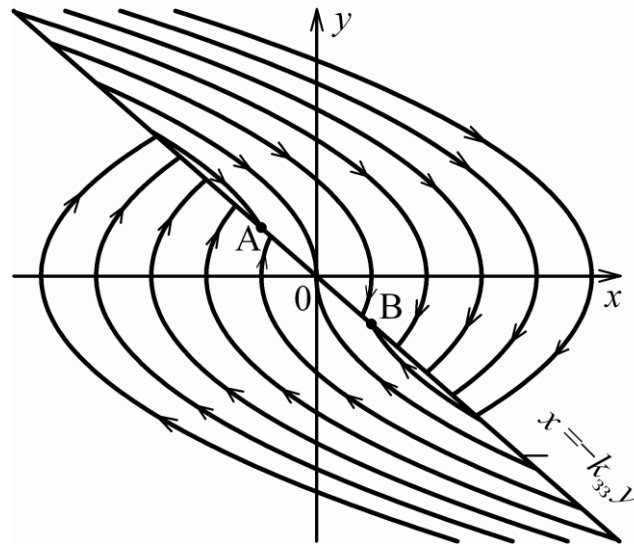


Рисунок 5.2 – Фазовий портрет системи з ковзним процесом

Ліворуч від лінії перемикання  $x + k_{33} y < 0$ , тому рівняння фазових траєкторій буде мати вид

$$y dy = k_1 c dx \quad \text{або} \quad y^2 = 2k_1 c x + C_2.$$

Це параболи, гілки яких спрямовані у бік збільшення  $x$  (див. рис. 5.2). Напрямок руху по фазових траєкторіях визначається знаком похідної  $y = dx/dt$ . У верхній півплощині ( $y > 0$ ) рух убік збільшення  $x$ , а в нижній ( $y < 0$ ) – убік зменшення  $x$ .

Всі ці параболи двічі перетинаються лінією перемикання (це властивість будь-якої параболи). Таким чином, кожна з фазових траєкторій починається й закінчується на лінії перемикання. На лінії перемикання можна виділити деякий відрізок АВ навколо початку координат, усередині якого не починається жодна фазова траєкторія. До цього відрізка фазові траєкторії тільки підходять із обох сторін і як би впираються одна в одну на цьому відрізку.

Розглянемо процеси на цьому відрізку більш докладно (рис. 5.3). Нехай процес іде по фазовій траєкторії 1. Як тільки фазова траєкторія перетне лінію перемикання ОА, вступає у свої права фазова траєкторія 2, що поверне процес до відрізка ОА, але тут виникне фазова траєкторія 3 і т.д. У результаті точка, що зображує, шляхом вібрацій біля лінії перемикання переміститься в початок

координат. Такий хід процесу відповідає перемиканням релейного елемента з великою частотою, теоретично частота перемикання нескінченна, а амплітуда вібрації на рис. 5.3 прагне до нуля, отже теоретично зображувальна точка ковзає по лінії перемикання до початку координат – до рівноважного стану. Процес такого роду називається ковзним процесом.

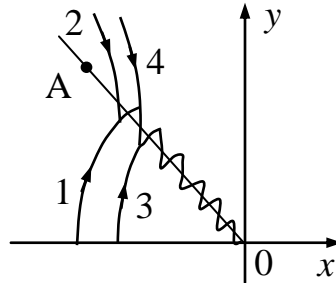


Рисунок 5.3 – Ковзний процес

Знайдемо закон руху в ковзному процесі. На лінії перемикання, згідно (5.3) і першого з рівнянь (5.1) має місце рівняння

$$\frac{dx}{dt} + \frac{1}{k_{33}} x = 0. \quad (5.4)$$

Розв'язком цього рівняння є  $x = x_0 e^{-\frac{t}{k_{33}}}$ , де значення  $t = 0$  і  $x = x_0$  відраховуються в момент влучення точки, що зображує, на відрізок АВ. Отже ковзний процес відбувається за експонентним законом. Нелінійна система другого порядку (5.1) вироджується в лінійну систему першого порядку (5.4). При цьому закон руху в ковзному процесі не залежить від параметрів прямого ланцюга системи й визначається тільки коефіцієнтом зворотного зв'язку.

Визначимо положення кінців відрізка ковзного процесу А і В на фазовій площині. Очевидно, що в цих точках дотичні до парабол, які з них виходять, збігаються з лінією перемикання. Дійсно, якщо в точці А дотична до параболи, що з неї виходить, буде збігатися з лінією перемикання, то вся парабола буде лежати нижче лінії перемикання, що суперечить рівнянню (5.2). Отже, така траєкторія існувати не може. Те ж і для точки В: вихідна траєкторія лежить не у своїй півплощині й тому не може існувати.

Для всіх точок лінії перемикання між точками А і В нахил дотичних до парабол, що з них виходять, виявляється більше нахилу ліній перемикання й парабола попадає не у свою півплощину фазової площини, і така траєкторія існувати не може.

Згідно з (5.3) нахил лінії перемикання визначається виразом

$$dy/dx = -1/k_{33}.$$



З рівнянь фазових траєкторій одержуємо нахили дотичної в точках А и В:

$$\left. \frac{dy}{dx} \right|_{y=y_A} = -\frac{k_1 c}{y_A} = -\frac{1}{k_{33}}, \quad \left. \frac{dy}{dx} \right|_{y=y_B} = \frac{k_1 c}{y_B} = -\frac{1}{k_{33}}.$$

Звідси одержуємо, що  $y_A = k_1 c k_{33}$  і  $y_B = -k_1 c k_{33}$ . Отже відрізок ковзного процесу АВ тим більше, чим більше коефіцієнти підсилення прямого ланцюга і зворотного зв'язку.

Не зважаючи на те, що даний приклад має аналітичний розв'язок, його моделювання в системі Simulink може викликати деякі труднощі. Ці труднощі пов'язані з моделюванням ковзного режиму, який відповідає перемиканням релейного елемента з великою (нескінченною) частотою. Погане розуміння процесів, що протікають в подібних системах, може призвести до помилкових висновків про правильність моделі, що використовується. Це, у свою чергу, може призвести до непродуктивного збільшення часу дослідження системи за рахунок пошуку неіснуючої помилки. Ціль даної роботи показати шлях подолання цієї проблеми засобами Simulink.

### Хід роботи

1. Викликати програму Matlab. Для запуску Simulink набрати відповідну команду в командному рядку Matlab або натиснути кнопку . Дочекатися відкриття вікна Simulink Library Browser (вікно бібліотеки блоків) і натиснути кнопку  для відкриття нового (чистого) вікна моделі. Ввести функціональну схему на рис. 5.4, що відповідає системі на рис. 5.1.

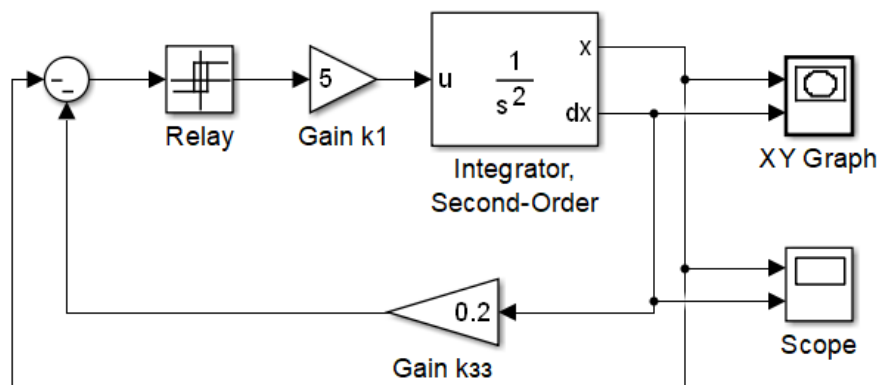


Рисунок 5.4

Тут для визначення похідної  $dx/dt$  використовується нижній вихід подвійного інтегратора, що дає можливість не використовувати диференційний блок, який сам по собі може вносити в систему помилки чисельного диференціювання.

Задайте значення коефіцієнтів  $k_1 = 5$  і  $k_{33} = 0.2$ , параметри релейного блоку задайте згідно з рис. 5.5. Поля **Switch on point** і **Switch off point** задають значення вхідного сигналу, при яких елемент перемикається уверх і униз. За замовченням там стоїть значення eps, що відповідає машинному нулю. Нас це

значення в даному випадку влаштовує. Поля **Output when on** і **Output when off** задають значення виходу при перемиканнях реле уверх і униз. Задаємо тут значення 100 і -100. У вікні блоку подвійного інтегратора задайте початкові умови для змінної  $x$  і її похідної рівними -50 і 800 відповідно.

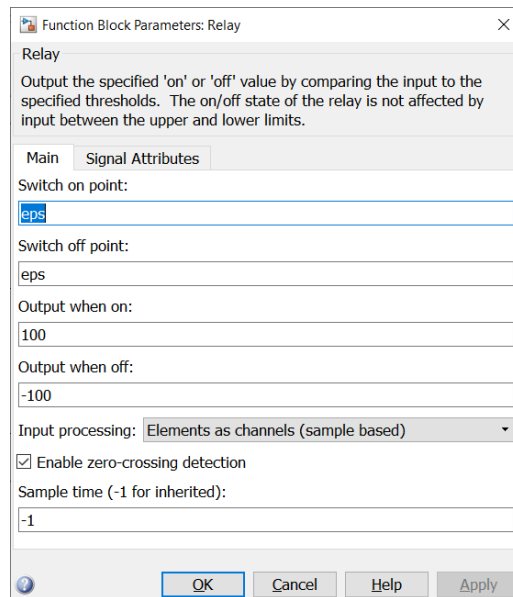


Рисунок 5.5

Перейдіть до панелі **Solver** діалогового вікна **Configuration Parameters**. Переконайтеся, що для параметра «**Algorithm**» встановлено значення «Nonadaptive» у розділі **Zero-crossing options**, а для моделювання час зупинки встановлено 9 секунд. Максимальний крок інтегрування встановіть рівним 0.01 секунди. Відкрийте вікно властивостей блоку **XY Graph** і встановіть для мінімальних і максимальних значень змінних  $x$  і  $y$  значення -400 і 600 та -800 і 900 відповідно. Імітуйте модель.

Зверніть увагу, що моделювання не викликає труднощів до моменту попадання фазової траєкторії на лінію перемикання в межах відрізка ковзного процесу. Це можна спостерігати за процесом малювання фазової траєкторії на графіку блоку **XY Graph**, а також по ходу бігунка на панелі стану вікна моделі. При попаданні траєкторії на лінію ковзного процесу швидкість інтегрування різко знижується. Це пояснюється швидким перемиканням реле і відповідно автоматичним зменшенням кроку інтегрування. Коли кількість перемикань перевищить 1000, Matlab припиняє моделювання і видає повідомлення про помилку, подібне до показаного на рис. 4.5. Як уже вказувалось вище, така ситуація при не повному розумінні процесів в системі може привести до висновку, що в моделі є помилка. Але Simulink дає можливість перевірити, чи бовтанка у процесі є притаманною природі системи, чи викликана нестійкістю процесу інтегрування, що часто є результатом помилок у моделі.

Перейдіть до панелі **Solver** діалогового вікна **Configuration Parameters**. Для параметра «**Algorithm**» встановіть значення «Adaptive» у розділі **Zero-crossing options**, і повторіть моделювання. Ви побачите, що не зважаючи на

зменшення швидкості інтегрування після попадання фазової траєкторії на лінію ковзного процесу, моделювання не зупиняється і іде до заданого моменту часу.

На рис. 5.6 показана фазова траєкторія системи в координатах  $x$  і  $y$ . На ній чітко видно точки переключення реле, через які легко можна провести пряму лінію (червона точкова пряма на рис. 5.6), що відповідає теоретичній лінії переключення. Видно також, що остання ділянка процесу іде по лінії переключення. На рис. 5.7а показані графіки змінної  $x$  і її похідної  $y$  часі. Вертикальна синя лінія показує момент переходу системи у ковзний режим.

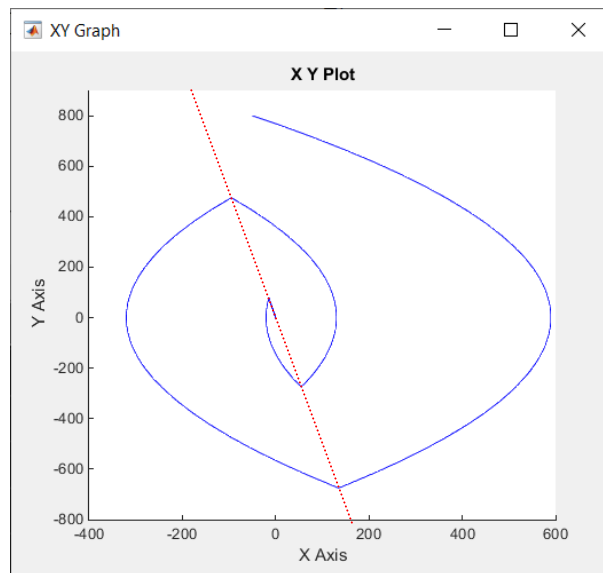


Рисунок 5.6

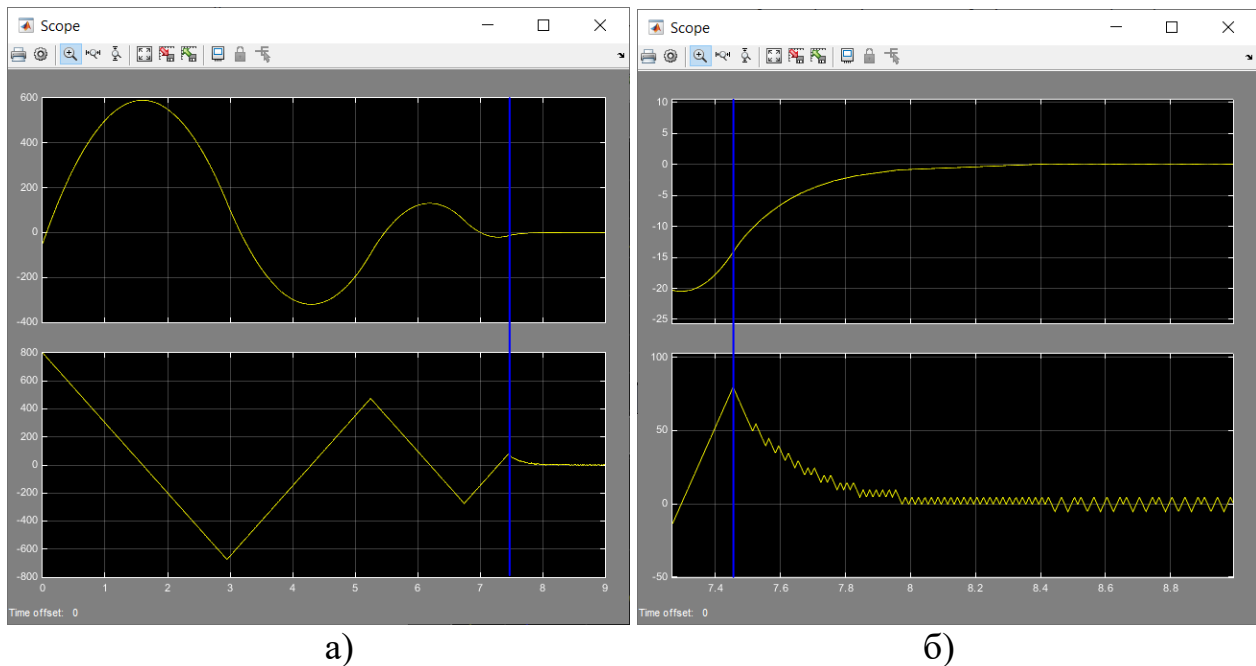


Рисунок 5.7

На рис. 5.7б показано у збільшеному масштабі кінцеву стадію перехідного процесу. Видно, що коливальний процес змінної  $x$  переходить у

експоненціальний. Але коливання на графіку швидкості говорять, що рішення в ковзному режимі не є зовсім точним. Тим не менш, знайдене рішення якісно правильно відображає теоретичний хід процесу і є близьким до нього кількісно.

2. Оформіть звіт про виконання лабораторної роботи. Скопіюйте схему моделі, графіки фазової траєкторії і перехідних процесів, у тому числі в збільшеному масштабі.

### Контрольні питання

1. Який процес називається ковзним і в яких системах він може виникнути?
2. Як задаються параметри реле?
3. Як задати початкові умови подвійного інтегратора?
4. Як продовжити моделювання при виникненні ковзного режиму?

### СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Дьяконов В.П. Simulink 5/6/7: Самоучитель. – М: ДМК-Пресс, 2008. – 784 с.
2. Бессекерский В.А., Попов Е.П. Теория систем автоматического регулирования. - М.: Наука, 1975. - 776 с.
3. Попов Е.П. Теория линейных систем автоматического регулирования и управления. - М.: Наука, 1989. - 301 с.