Навчально-методична література

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ
УНІВЕРСИТЕТ ІМ. ІВАНА ПУЛЮЯ

І.Р. Козбур, Г.В. Козбур, Р.І. Михайлишин

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторної роботи по дисципліні

«КОМП'ЮТЕРНІ МЕТОДИ ДОСЛІДЖЕННЯ СИСТЕМ АВТОМАТИЧНОГО УПРАВЛІННЯ»

«Моделювання нелінійних систем керування у пакеті MATLAB SIMULINK»

для студентів 4 курсу спеціальності

6.050201 «Системна інженерія»

Тернопіль 2019

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМ. ІВАНА ПУЛЮЯ

Кафедра
Автоматизації технологічних процесів і виробництв

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторної роботи по дисципліні
«КОМП'ЮТЕРНІ МЕТОДИ ДОСЛІДЖЕННЯ СИСТЕМ АВТОМАТИЧНОГО УПРАВЛІННЯ»
«Моделювання нелінійних систем керування у пакеті MATLAB SIMULINK»

для студентів 4 курсу спеціальності
6.050201 «Системна інженерія»

Тернопіль - ТНТУ – 2019

Методичні вказівки до виконання лабораторної роботи «Моделювання нелінійних систем керування у пакеті MATLAB SIMULINK», по курсу «Комп'ютерні методи дослідження систем автоматичного управління», для студентів 4 курсу спеціальності 6.050201 «Системна інженерія» / Авт.: Козбур І.Р., Козбур Г.В. Михайлишин Р.І.,– Тернопіль: ТНТУ, каф. АВ, 2019. - с. 19

Укладачі: ст. викл. каф. АВ Козбур І.Р., ст. викл. каф. КН Козбур Г.В., ст. викл. каф. АВ, к.т.н. Михайлишин Р.І.

Рекомендовано кафедрою «Автоматизації технологічних процесів і виробництв», протокол № 10 від «5» лютого 2019 р.

Рекомендовано науково-методичною радою ФПТ, протокол № 6 від «26» березня 2019 р.

ЗМІСТ

ЗМІСТ 4

Моделювання систем керування в пакеті MATLAB SIMULINK**Ошибка! Закладка не определена.**

Мета роботи ..... **Ошибка! Закладка не определена.**

Завдання роботи ..... **Ошибка! Закладка не определена.**

Моделювання систем керування в пакеті SIMULINK (короткі теоретичні відомості) ..... **Ошибка! Закладка не определена.**

Створення моделей в SIMULINK ..... **Ошибка! Закладка не определена.**

Основні джерела сигналів (Sources) ..... **Ошибка! Закладка не определена.**

Основні пристрої виводу (Sinks) ..... **Ошибка! Закладка не определена.**

Лінійні системи (Continuous) ..... **Ошибка! Закладка не определена.**

Інші часто використовувані блоки ..... **Ошибка! Закладка не определена.**

Блок Scope ..... **Ошибка! Закладка не определена.**

Оформлення графіків ..... **Ошибка! Закладка не определена.**

Компенсація постійних збурювань ..... **Ошибка! Закладка не определена.**

Оформлення звіту ..... **Ошибка! Закладка не определена.**

Опис системи ..... **Ошибка! Закладка не определена.**

Інструкція з виконання роботи ..... **Ошибка! Закладка не определена.**

Таблиця коефіцієнтів (варіанти завдань) ..... **Ошибка! Закладка не определена.**

Контрольні питання до захисту ..... **Ошибка! Закладка не определена.**

Приклад виконання й звіту по лабораторній роботі – Моделювання систем керування в пакеті SIMULINK ..... **Ошибка! Закладка не определена.**

Опис системи ..... **Ошибка! Закладка не определена.**

Дослідження системи із ПД-Регулятором ..... **Ошибка! Закладка не определена.**

Дослідження системи з ПІД-Регулятором ..... **Ошибка! Закладка не определена.**

Лабораторна робота

Моделювання нелінійних систем керування у пакеті MATLAB SIMULINK

Мета роботи

- освоєння методів моделювання нелінійних систем у пакеті MATLAB SIMULINK

Завдання роботи

- навчитися будувати й редагувати моделі з підсистемами
- навчитися використовувати нелінійні ланки типу «насичення»
- навчитися будувати трохи графіків одночасно на одному осцилографі
- навчитися становити, редагувати й налагоджувати скріпти
- навчитися змінювати властивості елементів графіка (шрифт, товщину лінії)

Моделювання нелінійних систем керування (короткі теоретичні відомості)

Моделі нелінійних ланок (*Discontinuities*)

Saturation – насичення, у параметрах задаються верхній і нижній межі (**Upper limit** і **Lower limit**).

Dead zone – нечутливість, «мертва зона». У параметрах задаються межі нечутливості (**Start of dead zone** і **End of dead zone**).

Rate Limiter – обмежувач швидкості зміни сигналу, у параметрах задаються межі на швидкість збільшення (**Rising slew rate**) і на швидкість зменшення (**Falling slew rate**).

Relay – реле, у параметрах задаються точки перемикання (**Switch on point** і **Switch off point**), у також величини сигналів у режимах «включене» (**Output when on**) і «виключене» (**Output when off**).

Backlash – люфт, «мертвий хід». У параметрах задаються величина мертвого ходу (**Deadband width**) і початкове значення виходу (**Initial output**).

Coulomb and Viscous Friction – кулонівське й в'язке тертя.

Підсистеми

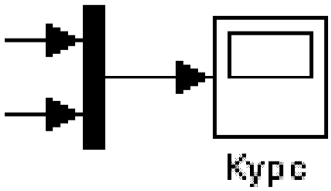
Якщо на схемі багато блоків, вона стає громіздкою й працювати з такою схемою незручно. Щоб не перевантажувати схеми, можна поєднувати блоки в *підсистеми*. Найпростіше виділити потрібні блоки мишкою («обвести» при натиснутої ЛКМ) і натиснути клавіші **Ctrl+G** (або вибрати пункт меню **Edit – Create subsystem**). На основній схемі підсистема зображується як блок типу **Subsystem** (із групи **Ports and Subsystems**). Цей блок можна додати й вручну, перетягнувши з вікна *Library Browser*.

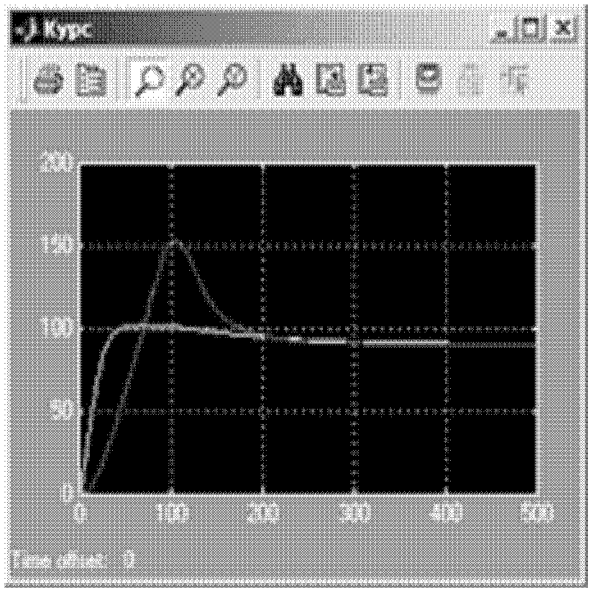
За допомогою подвійного клацання ЛКМ можна «відкрити» підсистему. Входи позначаються блоками **In**, а виходи – блоками **Out** (також із групи **Ports and Subsystems**). Можна додавати нові входи й виходи, видаляти непотрібні, міняти назви, працюючи з ними так само, як з іншими блоками.

Усередині підсистем можна створювати вкладені підсистеми. У підсистемі не може бути блоків з однаковими назвами, однак у різних підсистемах – можуть бути.

Зберегти модель можна з вікна будь-якої підсистеми, але закриття вікна підсистеми не приводить до закриття моделі.

Блок *Scope* (кілька сигналів)





До входу осцилографа (блоку **Scope**) можна підключати кілька сигналів одночасно. Для цього їх треба об'єднати в один векторний сигнал («палета») за допомогою блоку **Mux** (мультиплексор) із групи **Signal Routing**.

Якщо використовується два вхідні сигнали, перший зображується жовтою лінією, другий – фіолетової. При передачі даних у робочу область MATLAB у форматі **Array**, масив буде містити три стовпці – час, перший сигнал і другий сигнал. Якщо сигналів більше, відповідно збільшується кількість стовпців масиву.

Скріпти

При роботі в MATLAB часто для одержання потрібного результату треба ввести послідовно кілька команд. Якщо з'ясується, що в якийсь команді була зроблена помилка або потрібно змінити вихідні дані, усі команди прийде вводити знову. Щоб не набирати їх вручну, можна записати всю послідовність команд на диск у вигляді текстового файлу (М-Файлу з розширенням .m), а потім виконувати його, викликаючи по імені. Такий файл називається *скріптом.*

Скріпт – це програма, яка являє собою список команд мовою системи MATLAB. Скріпти можна створювати й редагувати в будь-якому найпростішому текстовому редакторі (наприклад, у *Блокноті*), однак зручніше за все використовувати вбудований редактор MATLAB, у якому є підсвічування синтаксису (команди, символьні рядки, коментарі й інші елементи програми виділяються різними квітами).

У М-Файлі перелічуються послідовно всі необхідні команди. Точка з коми наприкінці команди пригнічує вивід результату на екран. Можна розташовувати в одному рядку кілька команд, розділяючи їх комі (якщо потрібний вивід результату на екран) або точкою з коми. Якщо треба перенести довгу команду на наступний рядок, наприкінці рядка ставиться трикрапка.

Коментарем уважається все, що розташоване праворуч від знака % до кінця рядка. Його можна ставити в будь-якому місці рядка, наприклад, праворуч від команди MATLAB.

Для виклику скріпта треба набрати його ім'я в командному вікні MATLAB і натиснути клавішу **Enter**. Скріпт повинен перебувати в робочій папці (ім'я якої показано у вікні **Current directory** у верхній частині командного вікна) або в одній з папок, що входять у *шлях для пошуку*. У шлях для пошуку включені папки, у яких перебувають М-Файли для стандартних функцій системи MATLAB і додаткових пакетів (*toolbox*). Якщо треба, щоб скріпт запускався з будь-якої папки, треба включити папку, де він перебуває, у шлях для пошуку (команда **File – Set Path** верхнього меню).

Запустити скріпт можна безпосередньо з вікна редактора MATLAB, нажавши на клавішу **F5**. Можна виконати не весь скріпт, а тільки деякі рядки – їх потрібно виділити й натиснути клавішу **F9**. Можна розташувати два вікна (редактор і командне вікно MATLAB) поруч так, щоб вони не перекривали один одного. Тоді при виконанні скріпта (або окремих команд) відразу буде видний результат.

Якщо в командах скріпта є помилки (або вони виникли при виконанні) повідомлення, що відповідають, виводяться в командне вікно Matlab.

Форматування графіка

За замовчуванням команда `plot` малює на активному графіку нову криву, стираючи старі лінії. Для того, щоб існуючі криві збереглися, перед викликом `plot` треба виконати команду

```
>> hold on
```

Зворотна їй команда

```
>> hold off
```

відновлює стандартний режим.

Кожний об'єкт на графіку (осі координат, написи, лінії й т.п.) являють собою *об'єкти*, що мають властивості. Для того, щоб одержати значення властивості, використовують команду `get`, а для зміна властивості – команду `set`. Скорочення `gca` позначає поточні (активні) осі координат (*get current axes*). Команда

```
>> get(gca)
```

виводить на екран усі властивості осей і їх значення. Для керування розміром шрифту (він виміряється в пунктах) використовується властивість **Font Size**:

```
>> get(gca, 'Fontsize')    % визначити розмір шрифту
ans =
     10

>> set(gca, 'Fontsize', 16)      % змінити розмір шрифту
```

Для того, щоб настроїти властивості окремої лінії, треба спочатку одержати її *хендл* (*handle* – ручка, рукоятка, покажчик). Так називається унікальний числовий код об'єкта, через який до цього об'єкта можна звертатися. Команда `gca` насправді повертає хендл поточних координатних осей. Команда

```
>> h = get(gca, 'Children')
```

записує в змінну `h` масив хендлів – покажчиків на об'єкти лінії. Для кожної лінії можна встановити товщину (у пунктах, за замовчуванням – 0,5 пункту) .

```
>> set(h(1),'Linewidth',1.5)

>> set(h(2),'Linewidth',1.5)
```

Аналогічно можна управляти й іншими властивостями.

Оформлення звіту

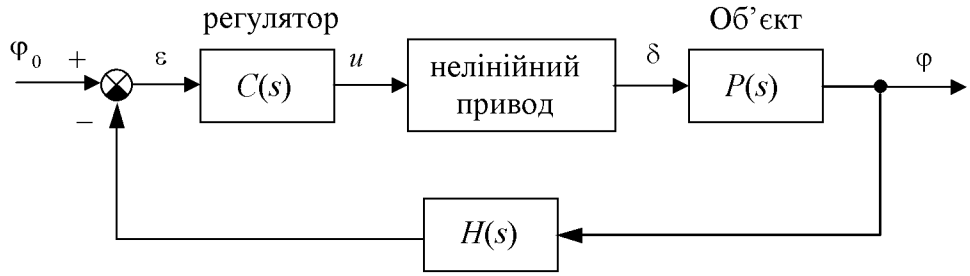
Звіт по лабораторній роботі виконується у вигляді тексту у файлі формату *Microsoft Word* (шрифт основного тексту **Times New Roman**, 12 пунктів, через 1,5 інтервалу, вирівнювання по ширині). Він повинен включати

- назва предмета, номер і назва лабораторної роботи
- прізвище й ініціали авторів, номер групи
- прізвище й ініціали викладача
- номер варіанта
- короткий опис досліджуваної системи
- результати виконання всіх пунктів інструкції, які виділені сірим тлом (див. нижче): результати обчислень, графіки, відповіді на запитання.

При складанні звіту рекомендується копіювати необхідну інформацію через буфер обміну з робочого вікна середовища MATLAB. Для цих даних використовуйте шрифт **Courier New**, у якому ширина всіх символів однакова.

Опис системи

У роботі розглядається система керування кута повороту за курсом. Її структурна схема показана на малюнку.



Вимірювальна система

Структурна схема системи стабілізації кута повороту на курсі

Лінійна математична модель, що описує нишпорення кута повороту, має вигляд

$$\dot{\varphi} = \omega_y$$

$$\dot{\omega}_y = -\frac{1}{T_s}\omega_y + \frac{K}{T_s}\delta$$

де φ – кут нишпорення (кут відхилення від заданого курсу), ω_y – кутова швидкість обертання навколо вертикальної осі, δ – кут повороту вертикального керма щодо положення рівноваги, T_s – постійна часу, K – постійний коефіцієнт, що має розмірність *рад/сек*. Передавальна функція від кута повороту керма до кута нишпорення запишеться у вигляді

$$P(s) = \frac{K}{s(T_s s + 1)}.$$

Лінійна модель привода (кермової машини) являє собою інтегруюча ланка з передавальною функцією

$$R_0(s) = \frac{1}{T_R s},$$

охоплена одиничним від'ємним зворотним зв'язком. На кут перекладки керма й швидкість перекладки накладаються нелінійні обмеження

$$\left|\dot{\delta}(t)\right| < 3\ ^\circ/сек, \qquad \left|\delta(t)\right| < 30^\circ.$$

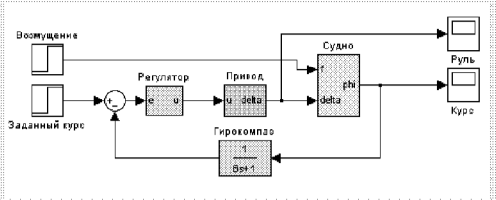
Для виміру кута нишпорення використовується гірокомпас, математична модель якого записується у вигляді аперіодичної ланки першого порядку з передавальною функцією[1]

$$H(s) = \frac{1}{T_{oc}s + 1},$$

У якості керуючого пристрою використовується ПІД-Регулятор з передавальною функцією[2]

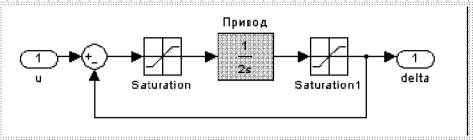
$$C(s) = K_c\left(1 + \frac{T_s s + 1}{T_v + 1}\right) + \frac{1}{T_I s}, \text{ де } T_v = 1 \text{ сек і } T_I = 200 \text{ сек.}$$

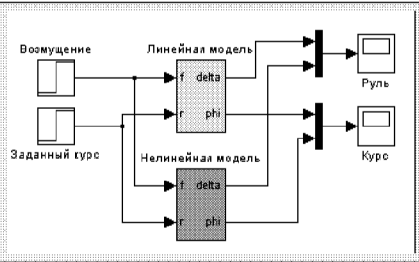
Інструкція з виконання роботи

Етап виконання завдання	Команди MATLAB
1. Зробіть свою папку робочою папкою MATLAB.	ЛКМ по кнопці … праворуч від поля Current Directory
2. Відкрийте вікно робочої папки.	View – Current directory
3. Відкрийте модель, побудовану в лабораторній роботі № 3.	подвійне клацання на lab3.mdl
4. Збережіть модель у своїй папці під іменем **lab4.mdl**.	lab3 File – Save as ...
5. Виділите мишею регулятор разом з інтегратором і перетворіть його в підсистему.	lab4 Edit – Create subsystem
6. Дайте підсистемі ім'я **Регулятор**, розташуєте назву зверху й виберіть фоновий колір на свій смак.	Подвійне клацання на імені ПКМ – Format – Flip name ПКМ – Background color
7. Визначите для входу й виходу цього блоку імена **e** и **u** відповідно.	Подвійне клацання на блоці ЛКМ на імені входу або виходу
8. Аналогічно побудуйте підсистему **ПРИВОД** із входом **u** і виходом **delta** і підсистему **Судно** із входами **f** і **delta** і виходом **phi**. Збережіть модель і скопіюйте її через буфер обміну у звіт.	

[1] Чисельні значення K, T_s, T_R і T_{oc} треба взяти з таблиці наприкінці файлу. Вони повинні збігатися з даними, які використовувалися Вами в лабораторних роботах № 2 і 3.

[2] Значення K_c було визначено в лабораторній роботі № 2.

9. Обведіть мишею (при натиснутої ЛКМ) усі блоки, крім джерел сигналів і осцилографів. Створіть підсистему **Лінійна система** із входами **r** (заданий курс) і **f** (збурювання) і виходами **phi** і **delta**.	lab4 Edit – Create subsystem
10. Скопіюйте блок **Лінійна система** й змініте його ім'я на **Нелінійна система.** Підключіть до входів нового блоку ті ж сигнали (заданий курс і збурювання), що й для першого блоку. Встановіть для блоку лінійної системи жовтий фоновий колір, а для нелінійної – фіолетовий.	Перетягнути за допомогою ПКМ Format – Background color
11. Відкрийте підсистему **Привод** у нелінійній системі. Ми побудуємо нелінійну модель привода, враховуючи обмеження на кут перекладки керма й швидкість його зміни.	Подвійне клацання на блоці
12. Вилучите сполучні лінії.	ЛКМ на елементі, натиснути Delete.
13. Змініте передавальну функцію на $R_0(s) = \frac{1}{T_R s}$.	Подвійне клацання на блоці • Denominator
14. Відкрийте вікно *Library Browser*.	lab4 View – Library Browser
15. Перетягнете у вікно моделі блок **Sum** із групи **Math Operations**. Змініте його так, щоб організувати негативний зворотний зв'язок.	Подвійне клацання на блоці |+- у поле List of signs
16. Перетягнете у вікно моделі два блоки **Saturation** (насичення) із групи **Discontinuities**. Розташуєте один блок перед інтегратором (обмеження на швидкість перекладки), другий – після (обмеження на кут перекладки).	
17. Уведіть потрібні межі припустимих значень, так щоб швидкість перекладки керма була не більш 3 градусів у секунду, а кут перекладки – не більш 30 градусів. У звіті вкажіть усі встановлені значення.	Подвійне клацання на блоці • Upper limit • Lower limit
18. З'єднаєте блоки потрібним способом. Збережіть модель. Скопіюйте схему нелінійної підсистеми **Привод** у звіт.	
19. Закрийте зайві вікна й перейдіть у головне вікно моделі. Звільните обоє осцилографа від зв'язків, перетягнувши їх вправо при натиснутій клавіші **Shift**.	
20. Перетягнете у вікно моделі два блоки **Mux** (мультиплексор) із групи **Signal Routing**. Ці блоки служать для об'єднання сигналів в «джгут» (багатожильний кабель).	
21. З'єднаєте входи першого блоку із сигналами керування (**delta**) лінійної й нелінійної систем, а вихід – із входом осцилографа **Кермо**.	
22. Аналогічно з'єднаєте входи другого мультиплексора із сигналами виходу (**phi**) лінійної й нелінійної систем, а вихід – із входом осцилографа **Курс.**	

23. Збережіть модель і скопіюйте її через буфер обміну у звіт.	
24. Встановіть заданий курс 10 градусів і збурювання 0. Виконаєте моделювання й подивіться результати. Жовтий графік показує зміну першого входу осцилографів – (лінійна система), фіолетовий – другого (нелінійна система).	
25. Поясніть розбіжність між результатами моделювання лінійної й нелінійної системи. Яка нелінійна ланка істотна впливає на результат?	
26. Створіть новий М-Файл.	`File - New - M-file`

27. У вікні редактора введіть команди для виводу графіків перехідних процесів за курсом. Тепер масив **`phi`** містить 3 стовпця: час і сигнали із двох входів осцилографа (виходи лінійної й нелінійної системи). Увесь текст праворуч від знака % уважається коментарем. Третій параметр команди **`plot`** означає колір: **`'b'`** – синій, **`'g'`** – зелений, **`'r'`** – червоний і т.д. (див. довідку по цій команді). Команда **`hold on`** означає, що не треба стирати старий графік, **`hold off`** – треба.	`figure(1); % відкрити мал. 1`
	`subplot(2,1,1);`
	`plot(phi(:,1),phi(:,2),'b');`
	`hold on;`
	`plot(phi(:,1),phi(:,3),'g');`
	`hold off;`
	`legend('Лінійна система', ...`
	`'Нелінійна система')`

28. Збережіть файл під іменем **`lab4graph.m`**.	`File - Save`
29. Запустіть файл (скріпт[3]) на виконання. Якщо графік не з'явився на екрані, дивитеся повідомлення про помилки в командному вікні MATLAB.	`клавіша F5`
30. Збільшіть розмір шрифту, вставивши цю команду відразу після виклику **`subplot`**. Тут **`gca`** означає поточні осі координат (*get current axis*). Ще раз запустіть скріпт.	`set(gca,'Fontsize',16);`

31. Додайте в скріпт назва графіка й осей координат, так само, як і в роботі № 3.	`title('Поворот на 10 градусів')`
	`xlabel('Час, сек');`
	`ylabel('\phi, гради');`

32. Збільшіть товщину ліній. За допомогою функції **`get`** ми спочатку одержуємо масив покажчиків (хендлів, *handle*) на всі об'єкти-лінії. Потім за допомогою функції set установлюємо для кожної лінії властивість **Linewidth** (товщина лінії), рівне 1,5 пункту. Збережіть файл і запустіть його на виконання.	`h = get(gca, 'Children')`
	`set(h(1),'Linewidth',1.5)`
	`set(h(2),'Linewidth',1.5)`

[3] *Скріптом* називається файл, що містить команди MATLAB. При його запуску команди виконуються послідовно одна за іншою.

33. Додайте в скріпт команди, за допомогою яких у нижній половині вікна на одному графіку будуються криві зміни сигналу керування в лінійній і нелінійної системах. Не додавайте заголовок для цього графіка (він буде заважати верхньому графіку).	
34. Добийтеся, щоб скріпт працював правильно. Скопіюйте текст скріпта у звіт.	
35. Запустіть скріпт на виконання. Скопіюйте отриманий графік у звіт.	
36. Змініте величину заданого курсу на 90 градусів і знову проведіть моделювання.	 Подвійне клацання на блоці Заданий курс увести 90 у поле Final value

37. Перейдіть у вікно редактора й змініте назву графіка на «Поворот на 90 градусів». Знову запустіть скріпт і скопіюйте побудований графік у звіт.	`title('Поворот на 90 градусів')`
	клавіша F5
	`print -dmeta`

38. Поясніть, чому розбіжність вийшла така істотна розбіжність між лінійної й нелінійної моделями? Як тепер нелінійності впливають на результат?	

Таблиця коефіцієнтів (варіанти завдань)

Варіант	T_s, сек	K, рад/сек	T_R, сек	T_{oc}, сек
1.	16.0	0.06	1	1
2.	16.2	0.07	2	2
3.	16.4	0.08	1	3
4.	16.6	0.07	2	4
5.	16.8	0.06	1	5
6.	17.0	0.07	2	6
7.	17.2	0.08	1	1
8.	17.4	0.07	2	2
9.	17.6	0.06	1	3
10.	17.8	0.07	2	4
11.	18.0	0.08	1	5
12.	18.2	0.09	2	6
13.	18.4	0.10	1	1
14.	18.6	0.09	2	2
15.	18.8	0.08	1	3
16.	19.0	0.07	2	4
17.	19.2	0.08	1	5
18.	19.4	0.09	2	6
19.	19.6	0.10	1	1
20.	18.2	0.0694	2	6

Контрольні питання до захисту

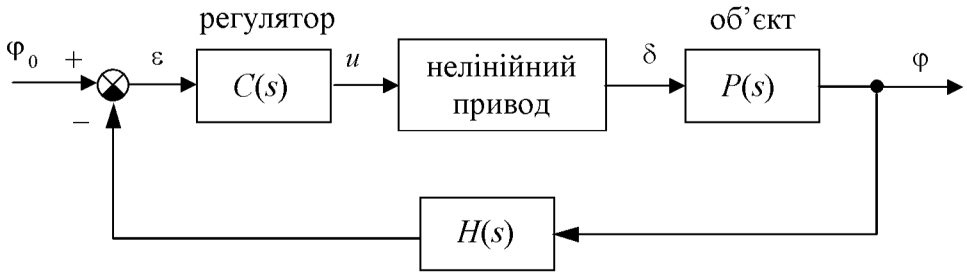
1. Див. усі питання до робіт № 1, № 2 і № 3.
2. Як побудувати підсистему з декількох існуючих блоків моделі?
3. Скільки входів і виходів може мати підсистема?
4. Як редагувати підсистему?
5. Як змінити назви входів і виходів підсистеми?
6. Як скопіювати існуючий блок або підсистему?
7. Як вилучити блок або сполучну лінію?
8. Поясніть структуру нелінійної моделі привода.
9. Чому в нелінійній моделі не можна використовувати загальну передавальну функцію лінійної моделі привода $R(s) = \dfrac{1}{T_R s + 1}$?
10. Як вибираються межі насичення для блоку, розташованого на вході інтегратора?
11. Як звільнити блок від зв'язків з іншими блоками?
12. Поясніть призначення блоку **Mux**.
13. Як розібратися, який сигнал надходить на перший вхід осцилографа (через мультиплексор), який – на другий?
14. Що таке скріпт у середовищі MATLAB?
15. Що означає знак % усередині скріпта?
16. Як уводити кілька команд в одному рядку?
17. Як правильно перенести довгу команду на наступний рядок?
18. Що означає третій параметр при виклику функції `plot`?
19. Що означають команди `hold on` і `hold off`?
20. Як запустити скріпт на виконання?
21. Як виконати тільки деякі команди зі скріпта?
22. Що означає `gca`?
23. Для чого служать функції `set` і `get`?
24. Як довідатися й змінити розмір шрифту на графіку?
25. Навіщо потрібний хендл (*handle*) графічного об'єкта?
26. Як змінити товщину лінії на графіку?
27. Де виводяться повідомлення про помилки в скріпте?

Приклад виконання звіту по лабораторній роботі

Моделювання нелінійних систем керування у пакеті MATLAB SIMULINK

1. Опис системи

Досліджується нелінійна система керування кута повороту за курсом, структурна схема якої показана на малюнку.



вимірювальна система

Рух кута повороту описується лінійною математичною моделлю у вигляді передавальної функції

$$P(s) = \frac{K}{s\,(T_s s + 1)}, \quad \text{де } K = 0.0694 \text{ рад/сек}, \ T_s = 18.2 \text{ сек},$$

Лінійна модель привода являє собою інтегруюче ланку з передавальною функцією

$$R_0(s) = \frac{1}{T_R\, s}, \qquad T_R = 2 \text{ сек},$$

охоплена одиничним від'ємним зворотним зв'язком. На кут перекладки керма й швидкість перекладки накладаються нелінійні обмеження

$$\left|\dot{\delta}(t)\right| < 3 \ ^\circ/\text{сек}, \qquad \left|\delta(t)\right| < 30^\circ.$$

Вимірювальний пристрій (гірокомпас) моделюється як аперіодична ланка з передавальною функцією

$$H(s) = \frac{1}{T_{oc} s + 1}, \quad T_{oc} = 6 \text{ сек},$$

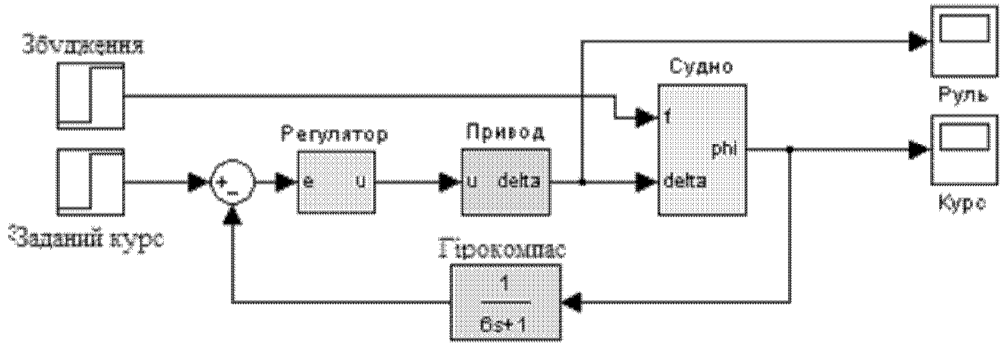
У якості керуючого пристрою використовується ПІД-Регулятор з передавальною функцією

$$C(s) = K_c\left(1 + \frac{T_s s + 1}{T_v + 1}\right) + \frac{1}{T_I s},$$

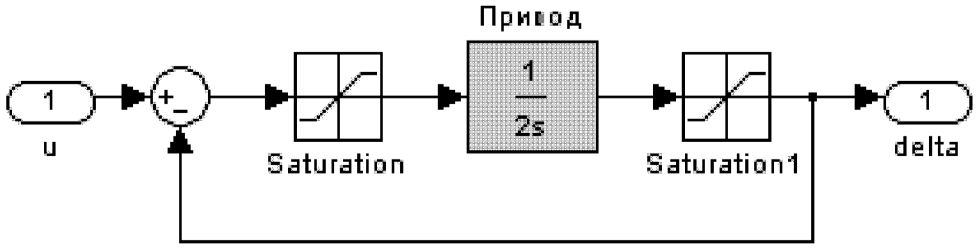
де $K_c = 0.7045$, $T_s = 18.2$ сек, $T_v = 1$ сек, $T_I = 200$ сек,

2. Побудова нелінійної моделі

- модель системи керування з виділеними підсистемами



- нелінійна модель привода



- нижній і верхній межі насичення

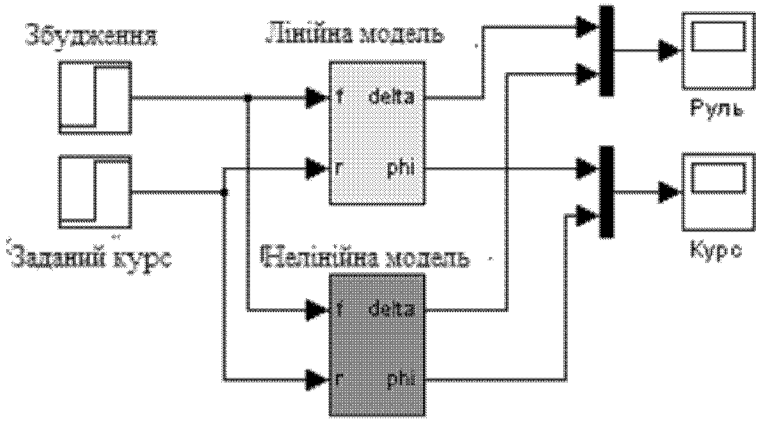
  блок **Saturation**: від -6 до 6

  блок **Saturation1**: від -30 до 30

  ці величини пояснюються тим, що …

3. Порівняння лінійної й нелінійної моделей

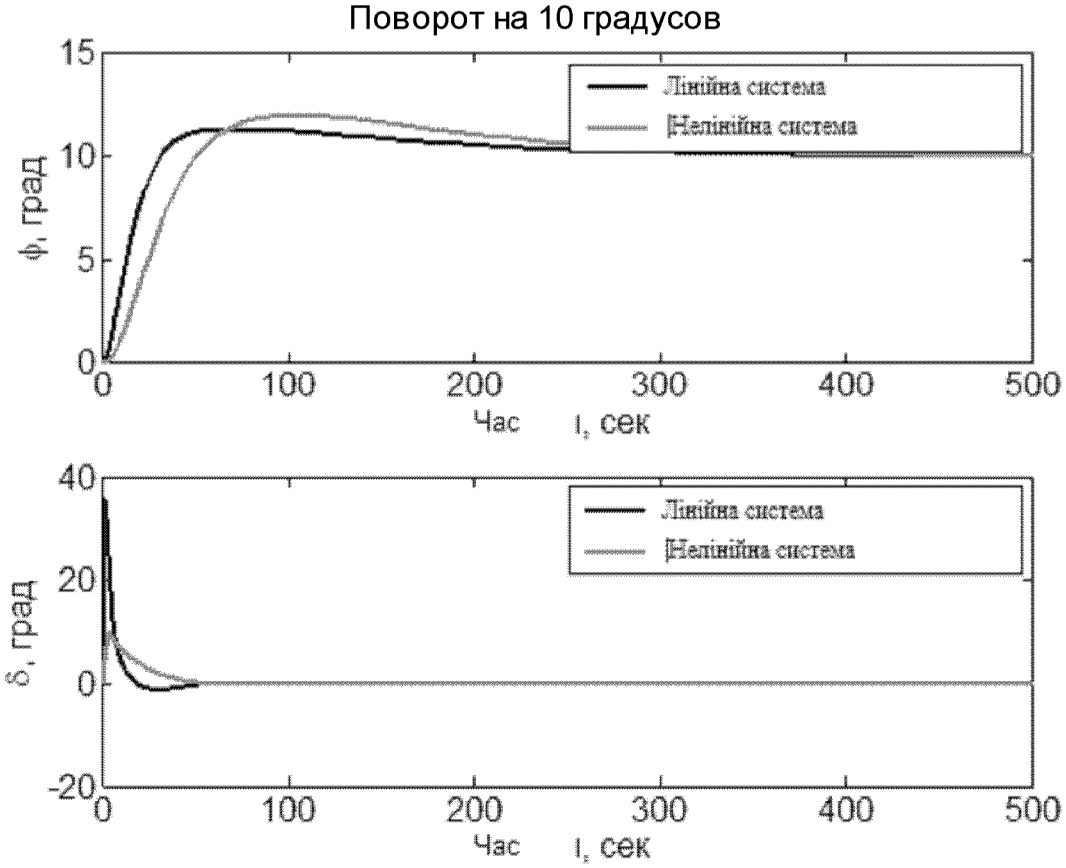
- структурна схема системи для порівняння лінійної й нелінійної моделей



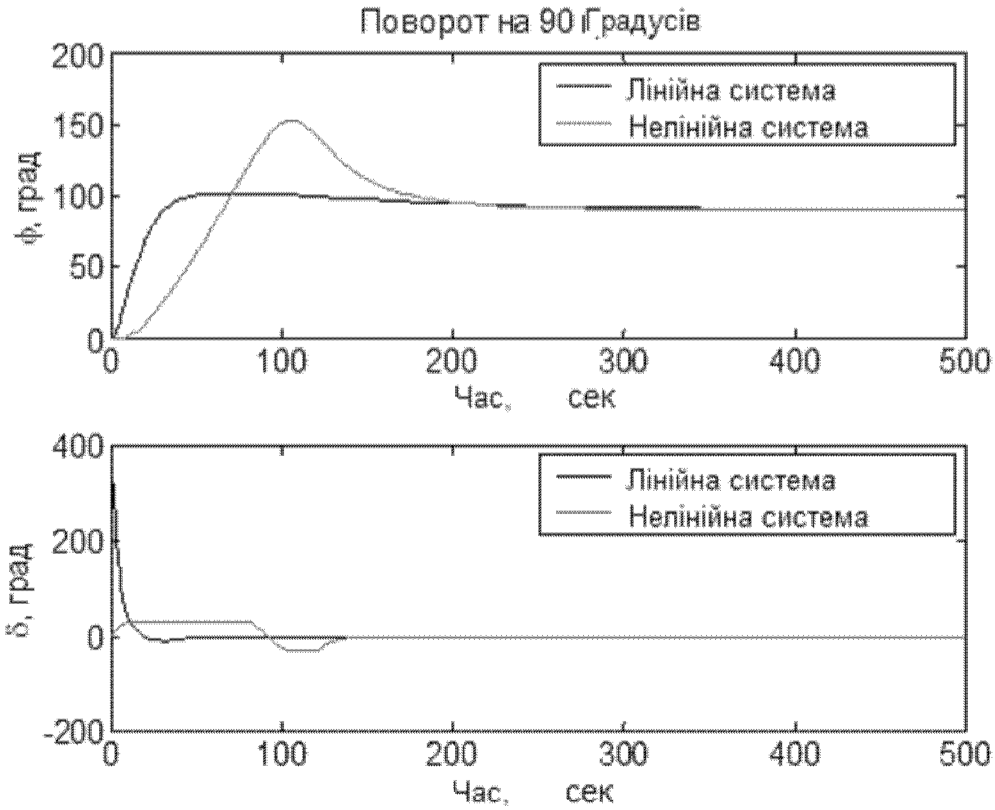
- скріпт для побудови й оформлення графіків

```
close(1);
figure(1);
subplot(2,1,1);
...
set(h(1),'Linewidth',1.5)
set(h(2),'Linewidth',1.5)
```

- перехідні процеси при зміні курсу на 10 градусів



- розбіжність між результатами моделювання лінійної й нелінійної системи пояснюється тим, що …
- найбільший вплив виявляє … (яка нелінійність?), тому що …
- у той же час … (що можна сказати про другу нелінійність?)
- перехідні процеси при зміні курсу на 90 градусів



- при більших кутах повороту спостерігається істотна розбіжність між процесами в лінійній і нелінійної системах, тому що …
- у цьому випадку … (як впливають нелінійності?)

ВИСНОВОК.

Використана література: