

Київський національний університет імені Тараса Шевченка
Факультет комп'ютерних наук та кібернетики
Кафедра моделювання складних систем

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня магістра
за спеціальністю 113 "Прикладна математика"
на тему:

МЕТОДИ СТАБІЛІЗАЦІЇ ПОСТАВИ ЛЮДИНИ

студента 2 курсу
Пономаренка Євгена Костянтиновича

Науковий керівник:
доктор фізико-математичних наук, професор
Пічкур В. В.

Робота заслухана на засіданні кафедри моделювання складних систем та рекомендована до захисту, протокол № – від – травня 2021 р.

Завідувач кафедри МСС
канд. фіз.-мат. наук, доцент

Дмитро ЧЕРНІЙ

Київ - 2021

Зміст

Вступ	4
1 Математичні моделі	7
1.1 Математична модель руху нижніх кінцівок з реакцією ЦНС	7
1.2 Комплексна математична модель руху нижніх кінцівок людини	10
2 Методи побудови керувань для розв'язання задачі стабілізації	14
2.1 Задача оптимальної стабілізації для стаціонарної системи .	14
2.1.1 Постановка задачі оптимальної стабілізації	14
2.1.2 Постановка задачі оптимальної стабілізації для стаціонарної лінійної системи керування	16
2.2 Метод функцій Ляпунова	18
2.3 Метод демпфування	19
3 Алгоритми розв'язання задачі керування поставою людини для математичної моделі руху нижніх кінцівок з реакцією ЦНС	21
3.1 Отримання експериментальних даних	21
3.2 Алгоритм оптимальної стабілізації	28
3.3 Алгоритм методу функцій Ляпунова	33
3.4 Алгоритм на основі методу демпфування	37
3.4.1 Обмеження на координати вектора керування . . .	38
3.4.2 Обмеження на норму вектора керування	42
4 Розв'язання задачі про перехід із фіксованої точки на наперед задану поверхню для комплексної математичної моделі руху нижніх кінцівок людини	45
4.1 Постановка задачі про перехід із фіксованої точки на наперед задану поверхню	45
4.2 Лінеаризація	45
4.3 Використання методу демпфування	48

4.3.1	Обмеження на координати вектора керування . . .	49
4.3.2	Обмеження на норму вектора керування	54
4.4	Опис програмної реалізації	59
	Висновки	61
	Література	62

Вступ

Останнім часом моделювання, аналіз і контроль біомеханічними системами є одними з найбільш складних проблем. У цьому контексті теорія систем керування, заснована на реальних, експериментальних даних, є ключовим інструментом, який може бути застосований для вирішення сучасних біомеханічних і механічних проблем, таких як стабілізація штучних систем. Біомеханічні моделі різної складності широко використовувалися при вивченні координації рухів, напр. [8]. Однак більша частина існуючих методів має обмежену придатність до реальних потреб. Система контролю поставою людини вміщує широку різноманітність вимог. ЦНС повинна вибрати відповідні структури активації м'язів для асортименту збурень різних типів і величин, відповідно до біомеханічних обмежень.

Цікаво дослідити, як ці моделі можуть бути адаптовані для вирішення актуальних проблем біомеханіки і для вирішення конкретних проблем в нейрофізіології людини. Стабільність постави у людини в прямому і тихому стані добре підходить для моделювання, вона є ідеальною стартовою експериментально-дослідною платформою для застосування моделювання до конкретних проблем управління ЦНС. Метою даного дослідження є розуміння функціонування ЦНС, а також розробка систем контролю нейро-м'язової стимуляції з метою відновлення функції спинного мозку у поранених. У цьому контексті необхідно приймати рішення щодо того, які м'язи стимулювати і запропонувати відповідну систему контролю для носних роботів (наприклад, екзоскелетів). Вимоги - це не тільки здатність м'язів протидіяти силі тяжіння, але і їх здатність прискорювати тіло. Оскільки сила, що виробляється м'язом, викликає прискорення в багатьох суглобах, верхня межа кутових прискорень будь-якого одного суглоба залежить від силових здібностей всіх м'язів. Виконання навіть простих завдань, таких як стабілізація постави, часто передбачає загальну координацію більш ніж одного суглобу. Тому доцільно вивчати не тільки одиночні суглобові кутові прискорення, а й вектори, що описують одночасні кутові прискорення декількох суглобів

[9].

Також, дуже важливим зараз є те, що активний розвиток високотехнологічного протезобудування в світі призвело до появи все більш досконаліх конструкцій протезів і протезованих систем, проектування і виготовлення яких здійснюється з використанням жорстких, а також на базі пружних елементів (тросові і ремінні передачі та ін.). На початковому етапі проектування подібних пристроїв особлива увага приділяється математичному моделюванню їх руху, яке повинно бути проведено відповідно до законів і принципами руху людини. Динаміку переміщення нижніх кінцівок людини доцільно розглядати в комплексі, тобто поряд з методами, що дозволяють вивчити та дослідити динаміку детермінованих елементів його будови (наприклад, кістки і суглоби), доцільно застосовувати методи, що дозволяють досліджувати закони руху його пружних елементів (м'язи та сухожилля).

В цій роботі одним із завдань буде розв'язок задачі стабілізації для системи, що описує реакцію ЦНС, трьома різними способами, побудувати керування та дослідити їхні відмінності.

Також в цій роботі буде висвітлена комплексна математична модель руху нижніх кінцівок людини. Модель, що описує рух детермінованих елементів ноги людини, представлена в формі звичайної динамічної системи. Отримана математична модель дозволяє досліджувати фізичні процеси, що виникають під час руху людини, пов'язані з визначенням значень кутів відхилення ланок.

Робота має наступний вигляд: в першому розділі розглядається побудова математичної моделі, яка буде основою для розв'язку задачі стабілізації постави людини, також описана комплексна математична модель руху нижніх кінцівок людини, пов'язана з визначенням значень кутів відхилення ланок; в другому розділі описуються теоретичні відомості про усі методи, які будуть використані для побудови керування. Перший метод - розв'язок задачі оптимальної стабілізації за допомогою методу функцій Ляпунову, в якому будується керування оберненого типу, за допомогою розв'язку алгебраїчного рівняння Ріккати. Два інші методи – розв'язки задачі стабілізації, де в першому знов використовується метод

функцій Ляпунова, а в другому – побудова керування, на основі методу демпфування, де на керування накладаються обмеження двох різних типів. В третьому розділі описується використання теоретичних даних для розв’язання задачі стабілізації системи керування, що описує поставу людини, показані результати використання програми для знаходження відповіді. В четвертому розділі описується алгоритм побудови розв’язку задачі про перехід із фіксованої точки на наперед задану поверхню для комплексної математичної моделі руху нижніх кінцівок людини. Також в цьому розділі висвітлено детальні відомості про роботу програми та інструменти, які були використані для її реалізації.

1 Математичні моделі

1.1 Математична модель руху нижніх кінцівок з реакцією ЦНС

Розглянута модель була запропонована [8], має три стани: суглобові кути, швидкості і прискорення (вектори x , \dot{x} , \ddot{x} відповідно), як показано на рис.1.1 (а). Вони описують головну динаміку і найбільш важливі властивості системи. Ці стани зібрані в $x(t)$. Крутні моменти, що виробляються м'язами і зовнішніми силами, що виробляються збуреннями, діють на тіло, в результаті чого рух описується компонентами вектора стану x , \dot{x} , \ddot{x} . Оскільки ці кінематичні характеристики x , \dot{x} , \ddot{x} та моменти T сили є єдиними змінними, які можна достовірно виміряти (або оцінити) для порівняння з моделлю, вони будуть ідеальними входами та виходами для моделі керування центральною нервовою системою. Для відображення вхідних кутових прискорень, що досягаються у суглобах в області щиколотки та ідентифікації системи контролю стану та простору, ми використовуємо «можливі набори прискорень» (МНП), розроблені [8]. Це набір всіх кутових прискорень суглоба (зібраних у вектор \ddot{x}), які можуть бути отримані будь-якою комбінацією можливих активацій м'язів (тобто з нормалізованими рівнями активації в межах діапазону $0 \leq a_i \leq 1, i = 1, 2, \dots, m$ суглобів). МНП може бути знайдений з відображення, отриманого з рівнянь руху, геометрії опорно-рухового апарату і м'язових властивостей, припускаючи, що м'язи скорочуються повільно, а динаміка збудження-скорочення є швидким відношенням до руху. Для моделі сагітальної площини, яка дозволяє рухатися до гомілковостопного, колінного і кульшового суглобів, цей набір є багатогранником в просторі кутового прискорення, представлений на рис. 1.1. Таким чином, МНП може використовуватися як міра прискорення, яка може бути досягнута в будь-якому напрямку в області прискорення гомілковостопного суглоба для заданого обсягу нервових зусиль. Враховуючи припущення вище, нервові зусилля еквівалентні кількості м'язової сили, нормалізованої до максимально можливої сили. Форма МНП корисна для визначення

стратегій щиколотки та стегна, що базуються виключно на кінематиці. На основі аналізу МНП на рис. 1.1 (с) отримуємо модель управління

$$\begin{pmatrix} \ddot{x}_{ank} \\ \ddot{x}_{knee} \\ \ddot{x}_{hip} \end{pmatrix} = b \cdot \begin{pmatrix} u_{ank} \\ u_{hip} \end{pmatrix} = \begin{pmatrix} 1 & -7.7 \\ 0 & 0 \\ 0 & 24.5 \end{pmatrix} \begin{pmatrix} u_{ank} \\ u_{hip} \end{pmatrix}$$

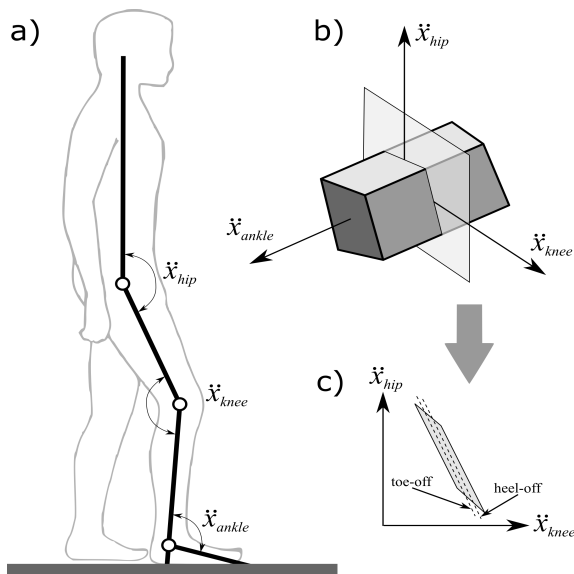


Рис. 1.1: а) Прискорення суглобів; б) Стани системи; с) МНП

де u_{ank} та u_{hip} є управління ЦНС, пов'язане зі стратегіями щиколотки та стегна (стовпці б), відповідно у випадку, коли стратегія колінного суглоба не розглядається.

Визначаючи одиничну стратегію управління як гомілковостопної та стегнової стратегії, основу б використовують для пошуку кутових прискорень суглоба. Перший стовпець б (стратегія щиколотки) знайдений шляхом вимірювання ширини багатокутника МНП (див. рис. 1.1 (с)) вздовж осі щиколотки, тоді як другий стовпчик (стратегія стегна) знайдений шляхом вимірювання МНП вздовж довгої осі [8]. Ці дві стратегії, описані векторами в області щиколотки, можуть служити основою для всіх рухів щиколотки. Будь-яке переміщення (в якому коліна стоять прямими) можуть бути виражені як комбінація стратегій стегна і гомілковостопного суглоба. Визначаючи величину кожного базисного вектора, щоб відповідати максимально можливого прискорення в цьому

напрямку, основа може також враховувати нейронні зусилля, пов'язані зі стратегіями.

Модель має наступні обмеження, що мають відношення до людської пози. Ці обмеження включають [9]: а) Коліна залишаються прямими (як експериментально спостерігали в людських реакціях на зворотні збурення опорної поверхні, яка висуває тіло вперед); б) Ноги тримаються на землі; в) Обмеження на максимальні сили м'язів. Звідси отримуємо обмеження:

$$\ddot{x}_{knee} = 0; \quad c_{cm0} \leq c_{cm}^T x \leq c_{cm1}, \quad (1)$$

$$c_{heel}^T \ddot{x} \leq c_{heel0}; \quad c_{toe}^T \ddot{x} \leq c_{toe0}, \quad (2)$$

де x - вектор стану.

У результаті, стратегія стегна може бути визначена як рух уздовж обмежень (1) і (2), в той час як стратегія щиколотки заснована на прискореннях вздовж осі щиколотки. Крім обмежень моделювання та нейронних зусиль, пов'язаних з рухом, кількість ступенів свободи в системі може бути зменшена. Обмеження положення тіла характеризують горизонтальне розташування центру тіла тіла, який повинен залишатися над базою опори для стійкої позиції. Тому система зменшеного порядку має вигляд [8]).

$$\dot{x} = Ax + Bu,$$

де вектор стану та керування

$$x \equiv \begin{pmatrix} x_{ank} \\ x_{hip} \\ \dot{x}_{ank} \\ \dot{x}_{hip} \end{pmatrix}, \quad u \equiv \begin{pmatrix} u_{ank} \\ u_{hip} \end{pmatrix}.$$

Системні матриці A , B включають як механіку тіла, так і динаміку

датчиків, залежно від бажаної складності моделі, вони мають форму

$$A \equiv \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, B \equiv \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & -7.7 \\ 0 & 24.5 \end{pmatrix}.$$

Розглянута модель включає мінімальні набори станів. Для того, щоб розрізнити збурення, які достатньо малі, щоб протиставити їх стратегії щиколотки, і ті, які достатньо великі, щоб вимагати стратегії стегна, ЦНС повинна мати інформацію як від нижнього, так і від верхнього датчиків для оцінки руху центру мас. Експериментальні дані свідчать, що інформація про повноцінний стан також є доступною і використовується при виборі стратегії управління. Тому передбачається, що всі механічні стани, у довільній реалізації, доступні або оцінені ЦНС для використання при формуванні управління. Основна вимога моделі полягає в тому, що вона адекватно представляє поведінку вводу-виводу людської системи, що складається з датчиків, контролерів і приводів. Будемо вважати, що ЦНС, що складається з безлічі нелінійних нейронних елементів, буде вести себе лінійно [9].

1.2 Комплексна математична модель руху нижніх кінцівок людини

У процесі руху людини беруть участь наступні групи м'язів [3]:

1. Для руху щодо тазостегнового суглоба: черевна - клубова, велика і мала сідничні м'язи.
2. Для руху щодо колінного суглоба: двоголовий і чотириголового м'яза стегна.
3. Для руху щодо гомілковостопного суглоба: литковий м'яз, зовнішні і передні м'язи гомілки.

Зазвичай м'язи, які здійснюють згинання (флексори), знаходяться попереду, а що виробляють розгинання (екстензори) - ззаду від суглоба. Тільки в колінному і гомілковостопному суглобах передні м'язи виробляють розгинання, а задні - згинання [3].

На рис. 1.2 представлена спрощена схема будови нижніх кінцівок людини [3].

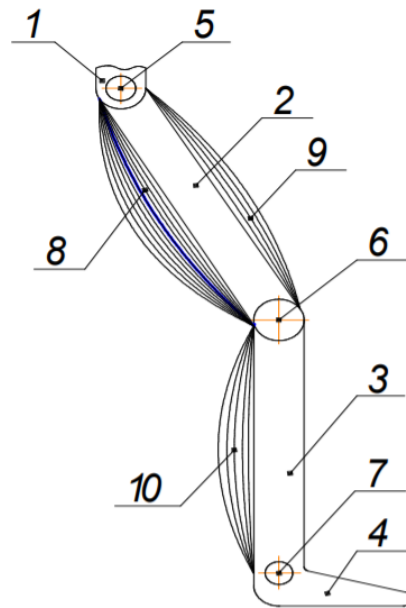


Рис. 1.2: Структурна схема будови нижньої кінцівки людини: 1 - тазостегновий суглоб; 2 - колінний суглоб; 3 - гомілковостопний суглоб; 4 - тазова кістка; 5 - стегнова кістка; 6 - кістка гомілки; 7 - стопа; 8 - двоголовий м'яз стегна; 9 - чотириглавий м'яз стегна; 10 - литковий м'яз.

Кістки стегна, гомілки, стопи і їх суглоби нами прийняті за абсолютно детерміновані (жорсткі) елементи, тому динаміку їх переміщення будемо розглядати в рамках звичайної динамічної системи (ОДС). Динаміку м'язової системи нижніх кінцівок людини можна досліджувати методом кінцевих елементів, що застосовуються до пристроїв з пружними елементами [3]. Мета дослідження - створення математичного комплексу опису переміщення людини, який дозволить вивчити динаміку руху нижніх кінцівок людини і використовувати отримані результати моделювання в процесі проектування протезованих пристроїв.

Досягнення поставленої мети пов'язано з наступними задачами:

- Здійснити математичне моделювання руху людини, яке описується

системою звичайних диференціальних рівнянь.

- Побудувати керування для розв'язку задачі переходу системи із фіксованої точки на наперед задану поверхню.
- Реалізувати програму, за допомогою якої можна буде побудувати траєкторії для керування та фазових координат для розв'язку вказаної задачі.

Схема, що дозволяє провести математичне моделювання руху нижніх кінцівок людини в рамках ОДС, представлена на рис. 1.3 [3]. Найбільш ефективним методом дослідження динаміки руху детермінованих елементів нижньої кінцівки людини на початковому етапі їх моделювання є метод, запропонований І. Като [7]. Переваги цього методу - простота системи диференціальних рівнянь, які використовуються для опису руху людини, і відсутність необхідності використовувати спеціалізоване обладнання для вимірювання фізичних параметрів руху людини.

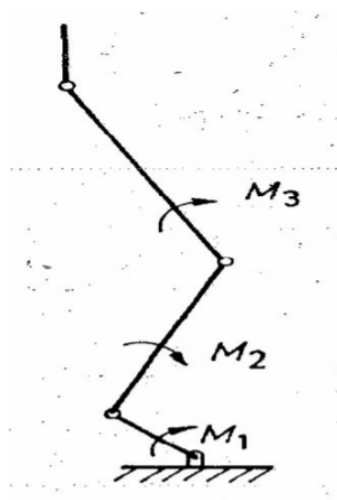


Рис. 1.3

Залежності суставних моментів $M_i = f(\theta_i)$ мають наступний вигляд [7]:

$$\begin{aligned}
& \{I_1 + m_1 a_1^2 + (m_2 + m_3) l_1^2\} \ddot{\theta}_1 + \\
& + \{m_2 l_2 l_1 \cos(\theta_1 - \theta_2) + m_3 l_1 l_2 \cos(\theta_1 - \theta_2)\} \ddot{\theta}_2 + \\
& + \{m_3 l_1 a_3 \cos(\theta_1 - \theta_3)\} \ddot{\theta}_3 - \\
& - (G_1 a_1 + G_2 l_1 + G_3 l_1) \sin \theta_1 = M_1 - M_2 - M_3, \\
& \{m_2 l_1 a_2 \cos(\theta_1 - \theta_2) + m_3 l_1 l_2 \cos(\theta_1 - \theta_2)\} \ddot{\theta}_1 + \\
& + \{I_2 + m_2 a_2^2 + m_3 l_2^2\} \ddot{\theta}_2 + \\
& + \{m_2 l_2 a_2 \cos(\theta_2 - \theta_3)\} \ddot{\theta}_3 - (G_2 a_2 + G_3 l_2) \sin \theta_2 = M_2 - M_3, \\
& \{m_3 l_1 a_3 \cos(\theta_1 - \theta_3)\} \ddot{\theta}_1 + \{m_3 l_2 a_3 \cos(\theta_2 - \theta_3)\} \ddot{\theta}_2 + (I_3 + m_3 a_3^2) \ddot{\theta}_3 - \\
& - G_3 a_3 \sin \theta_3 = M_3,
\end{aligned}$$

де m_1, m_2, m_3 - відповідні маси ланок; l_1, l_2, l_3 - довжини між суглобами; a_1, a_2, a_3 - довжини між центрами тяжіння ланки та відповідного суглоба; b_1, b_2, b_3 - довжини між ецентрами тяжіння ланки та відповідної точки опори; G_1, G_2, G_3 - вага ланки; M_1, M_2, M_3 - суглобні моменти; $\theta_1, \theta_2, \theta_3$ - кутові координати, обчислені від вертикалі.

З систем рівняння складемо системи диференціальних рівнянь для визначення кутів для визначення кутів обертання відносно тазостегнового і колінного суглобів.

$$\begin{cases}
\ddot{\theta}_1 = \frac{M_1 - M_2 - M_3 - a \cos(\theta_1 - \theta_2) \ddot{\theta}_2 - b \cos(\theta_1 - \theta_3) \ddot{\theta}_3 + c \sin \theta_1}{d}, \\
\ddot{\theta}_2 = \frac{M_2 - M_3 - a \cos(\theta_1 - \theta_2) \ddot{\theta}_1 - f \cos(\theta_2 - \theta_3) \ddot{\theta}_2 + g \sin \theta_2}{n}, \\
\ddot{\theta}_3 = \frac{M_3 - b \cos(\theta_1 - \theta_3) \ddot{\theta}_1 - m \cos(\theta_2 - \theta_3) \ddot{\theta}_2 + G_3 a_3 \sin \theta_3}{s},
\end{cases} \quad (3)$$

де $a = (m_2 l_2 l_1 + m_3 l_1 l_2)$, $b = m_3 l_1 a_3$, $c = (G_1 a_1 + G_2 l_1 + G_3 l_1)$, $d = I_1 + m_1 a_1^2 + (m_2 + m_3) l_1^2$, $e = (m_2 a_2 + m_3 l_2) l_1$, $f = m_3 l_2 a_3$, $g = (G_2 a_2 + G_3 l_2)$, $n = I_2 + m_2 a_2^2 + m_3 l_2^2$, $m = m_3 l_2 a_3$, $s = I_3 + m_3 a_3^2$;

2 Методи побудови керувань для розв'язання задачі стабілізації

2.1 Задача оптимальної стабілізації для стаціонарної системи

2.1.1 Постановка задачі оптимальної стабілізації

Нехай керований процес описується рівнянням

$$\dot{x} = f(t, x, u), \quad t > 0, \quad (4)$$

у якому $x \in R^n$, $u \in R^r$, вектор-функція $f(t, x, u)$ визначена в області $\|x\| \in \Omega \subset E^n$, де H - задана постійна, а Ω - відкрита або замкнена область в R^n [4]. Крім того, нехай у цій області функція $f(t, x, u)$ неперервна по t та неперервно диференційовна за сукупністю інших змінних, а

$$f_i(t, \theta^n, \theta^r) \equiv 0, \quad t = 1, 2, \dots, n$$

где θ^q - нульовий вектор порядку q . Допустимим керування вважаються кусково неперервні вектор-функції $u = u(t)$ зі значеннями в Ω . Керування типу $u = u(t, x)$ також вважається допустимим, якщо воно кусково неперервне по t та неперервно диференційовне по x . Нехай також, що на розв'язку задачі Коші

$$\dot{x} = f(t, x, u(t, x)), \quad t > 0, \quad x(0) = x^0, \quad \|x^0\| \leq H,$$

керування $u(t, x)$ приймає значення з області Ω принаймні за достатньо малих значень $t, t > 0$.

Нехай далі $\omega(t, x, u)$ - невід'ємна скалярна функція, визначена в указаній вище області зміни t, x та u . Задача оптимальної стабілізації має такий вигляд: потрібно знайти таке допустиме керування $u = u(t, x)$, щоб [2]:

- тривіальний розв'язок рівняння

$$\dot{x} = f(t, x, u(t, x)) \quad (5)$$

був асимптотично стійким;

- функціонал

$$I[u] = \int_0^{\infty} \omega(t, x, u) dt$$

приймав найменше значення при цьому керванні та відповідному розв'язку (5) за будь-якої початкової умови $x(0) = x^0$, $\|x^0\| \leq Y$.

Для розв'язку задачі оптимальної стабілізації скористуємось ідеями другого методу Ляпунова [5]. Позначимо через $V(t, x_1, \dots, x_n)$ функцію Ляпунова, визначену в області

$$t \geq 0, \quad \|x\| \leq H. \quad (6)$$

Введемо далі позначення

$$B[V, t, x, u] = \frac{\partial V(t, x)}{\partial t} + \sum_{i=1}^n \frac{\partial V(t, x)}{\partial x_i} f_i(t, x, u) + \omega(t, x, u). \quad (7)$$

Очевидно, якщо при якомусь виборі функцій $V(t, x)$ та $u(t, x)$ в області (6) виконується рівність

$$B[V(t, x), t, x, u(t, x)] = 0,$$

тоді в силу формули (7), повна похідна $V(t, x)$, в силу рівняння (5), в області (6) задовільняє умову

$$\frac{dV(t, x(t))}{dt} = \omega(t, x(t), u(t, x(t))).$$

Теорема. *Якщо для диференціального рівняння (4) можна знайти таку додатно визначену функцію $V(t, x)$, що має нескінченно малу вищу границю та разом з функцією $u(t, x)$ задовільняє в області (6) умовам [4]:*

- функція $\omega(t, x) = \omega(t, x, u(t, x))$ є додатно визначеною;
- має місце нерівність

$$B[V(t, x), t, x, u(t, x)] = 0;$$

- яким би не був вектор u , має місце нерівність

$$B[V(t, x), t, x, u] > 0,$$

то керування $u = u(t, x)$ є оптимальним. При цьому виконується рівність

$$\int_0^{\infty} \omega(t, x(t), u(t, x(t))) dt = \min_u \int_0^{\infty} \omega(t, x(t), u) dt = V(0, x(0)).$$

2.1.2 Постановка задачі оптимальної стабілізації для стаціонарної лінійної системи керування

Тепер розглянемо стаціонарну лінійну систему керування

$$\dot{x} = Ax + Bu,$$

де A , B – постійні матриці розмірності $n \times n$ та $n \times r$ відповідно, а критерій оптимальності має вигляд [4]

$$I[u] = \int_0^{\infty} [x^* Q x + \beta u^* R u] dt,$$

де матриця Q – невід’ємна постійна, R – додатня постійна. В цьому випадку функцію V будемо шукати у вигляді

$$V = \sum_{i,k=1}^n c_{ik} x_i x_k = x^* C x,$$

де $c_{ik}, i, k = 1, \dots, n$ – постійні. Тоді формула (7), що визначає функцію B , можна записати так

$$B[V, t, x, u] = \frac{\partial V}{\partial t} + \left(\frac{\partial V}{\partial x} \right)^* (Ax + Bu) + x^* Qx + \beta u^* Ru. \quad (8)$$

На оптимальному керуванні $u = u(t, x)$ функція (8) повинна досягати свого мінімуму та при цьому повинна виконуватись умова [4]

$$B[V, t, x, u(t, x)] = 0. \quad (9)$$

З умови мінімуму знаходимо

$$u(t, x) = -\frac{1}{2\beta} R^{-1} B^* \frac{\partial V}{\partial x},$$

та з визначення функції V , отримуємо

$$u(t, x) = -\frac{1}{\beta} R^{-1} B^* Cx. \quad (10)$$

Підставляємо знайдене керування у (8) та використовуємо умову (9), виходить

$$\frac{\partial V}{\partial t} + \left(\frac{\partial V}{\partial x} \right)^* Ax - \frac{1}{4\beta} \left(\frac{\partial V}{\partial x} \right)^* B R^{-1} B^* \frac{\partial V}{\partial x} + x^* Qx = 0,$$

Підставивши похідну V в це рівняння, отримуємо

$$CA + A^* C - \frac{1}{\beta} C B R^{-1} B^* C + Q = 0,$$

де 0 - квадратна матриця з нульовими елементами. Отримане рівняння - алгебраїчне рівняння Ріккати, в якому потрібно знайти постійну симетричну матрицю C , що задовільняє цьому рівнянню.

При цьому оптимальне керування знаходиться за формулою (10).

2.2 Метод функцій Ляпунова

Маємо нелінійну автономну систему керування, яка у загальному випадку має вигляд:

$$\frac{dx(t)}{dt} = F(x, t) + G(x, t)u(x)$$

Тут $x \in R^n$ - вектор фазових координат, $F(x, t)$ - n - вимірний вектор-функція, неперервна за t і локально ліпшицева за x , $G(x, t)$ - $n \times m$ - матриця з компонентами, неперервними за t і локально ліпшицевими за x , u - m - вимірний вектор керування. Керування u вибирається у класі з оберненим зв'язком, $u = u(x, t)$, $u(0, t) = 0$, $F(0, t) = 0$, $t \geq 0$ [2].

Згідно з прямим методом Ляпунова, будь-яке керування u з оберненим зв'язком, яке задовільняє умові [1]

$$\frac{\partial V(x, t)}{\partial t} + \langle \text{grad}_x V(x, t), F(x, t) + G(x, t)u(x) \rangle = -W(x, t), \quad (11)$$

є стабілізуючим законом керування. Тут $V(x, t)$, $W(x, t)$ - додатно визначені функції, при цьому $V(x, t)$ допускає нескінченно малу вищу границю. Отже, в рамках методу функцій Ляпунова розв'язок задачі стабілізації зводиться до побудови множини керувань з оберненим зв'язком, що забезпечують виконання співвідношення (11). Тоді

$$\langle G^*(x, t) \text{grad}_x V(x, t), u(x) \rangle = -W(x, t) - \frac{\partial V(x, t)}{\partial t} - \langle \text{grad}_x V(x, t), F(x, t) \rangle.$$

Подамо функцію керування у вигляді [6]

$$u(x, t) = \begin{cases} u_1(x), & \|x\| > \epsilon, \\ 0, & \|x\| < \epsilon. \end{cases} \quad (13)$$

Керування $u_1(x, t)$ вибирається у вигляді [1]

$$u_1(x, t) = -\lambda(x, t)G^*(x, t)\text{grad}_x V(x, t), \quad (14)$$

де $\lambda(x, t)$ - скалярна функція, яку потрібно визначити.

2.3 Метод демпфування

Розглянемо систему керування

$$\frac{dx(t)}{dt} = f(x(t), u(t), t), \quad (15)$$

де x — n -вимірний вектор фазових координат, $f(x, u, t)$ — n -вимірна вектор-функція, неперервна за сукупністю змінних, $u = u(t)$ — r -вимірний вектор керування, що належить класу кусково неперервних функцій, $u(t) \in G$, G — компакт в R^r , $x(t_0) = x_0, t \geq t_0$ [1]. Розв'язок системи (15), що відповідає керуванню u та початковій умові $x(t_0) = x_0$ будемо позначати $x(t, u, x_0, t_0)$. Нехай

$$S = \{(x, t) : \psi(x, t) = 0\},$$

де $\psi \in C(R^{n+1})$ і задача керування системою (15) полягає у знаходженні допустимого керування $u(t) \in G$ такого, що інтегральна крива системи (15) в деякий момент часу попадає з фіксованого положення на поверхню S . При цьому можуть бути задані обмеження на фазову змінну та керування [2].

Розглянемо функцію $V(x, t)$, яка характеризує відстань від точки x в момент t до поверхні S . Тоді сформульована задача керування може бути зведена до мінімізації $V(x, t)$ на розв'язках системи (15). У цьому полягає зміст оптимального керування по відношенню до демпфування функції.

Означення. Керування u_* називається оптимальним по відношенню до демпфування функції V , якщо функція V максимально спадає вздовж траєкторії $x_*(t) = x(t, u_*)$ [1].

Припустимо, що функція $V(x, t)$ є неперервно диференційованою. Знайдемо повну похідну функції V на розв'язках системи (15)

$$\frac{dV}{dt} = \frac{\partial V}{\partial t} + \langle \text{grad}_x V(x, t), f(x, u, t) \rangle = W(x, t).$$

згідно означення, оптимальне керування по відношенню до демпфування функції V доставляє функції $W(x, u, t)$ найменше від'ємне значення.

Теорема. Нехай виконуються такі умови [1]:

- керування u_* є оптимальним по відношенню до демпфування функції $V(x, t)$;
- функція $V(x, t) > 0$ коли $\psi(x, t) \neq 0$ і $V(x, t) = 0$ при $\psi(x, t) = 0$;
- повна похідна функції $V(x, t)$ на розв'язках системи (15) при керуванні $u = u_*$ рівна -1 .

Тоді керування u_* є оптимальним за швидкістю при переводі системи (15) зі стану $x(t_0) = x_0$ на поверхню $S = \{(x, t) : \psi(x, t) = 0\}$.

3 Алгоритми розв'язання задачі керування поставою людини для математичної моделі руху нижніх кінцівок з реакцією ЦНС

3.1 Отримання експериментальних даних

Біомеханічні експериментальні дослідження були проведені вченими лабораторії біомеханіки Технологічного університету міста Лодзь [9].



Рис. 3.1: Місце знаходження маркерів

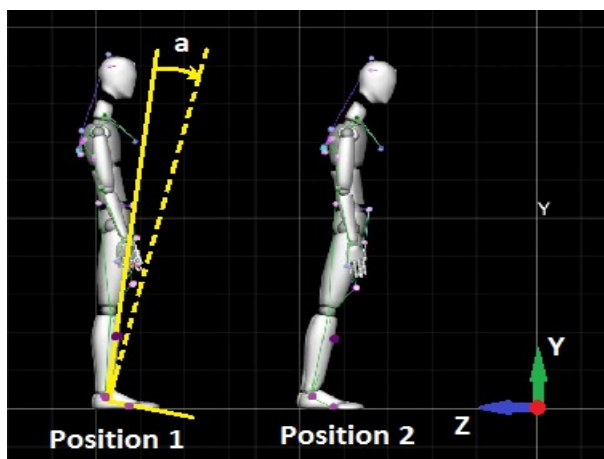


Рис. 3.2: Експерименти: моделювання стоячих відхилень

Для отримання кінематичних даних для дослідження руху використовувалася система захоплення руху OptiTrack, обладнана спеціальним програмним забезпеченням. Система має точність 0,5 мм, роздільну здатність 1,3 млн. пікселів, частоту дискретизації 120 FPS. Для спостереження за рухом (рис. 3.1) використовувався маркер всього тіла, заснований

на протоколі Ріццолі. 42 маркери захоплення руху були розміщені на тілі моделі для надійного відстеження маркерів в режимі реального часу для забезпечення подальшого аналізу в середовищі MatLab.

Для того, щоб провести аналіз і отримати реальну кінематику дані і характеристики траєкторій як опорних значень стоячих стійок, експериментальні випробування проводилися на здорових чоловіків-добровольцях, без будь-яких нервових і м'язових порушень (маса тіла 70 кг і висота тіла 183 см), що відхилялись різним чином. В ході експерименту було застосовано кілька відхилень [9].

Експерименти для моделювання вертикально стоячих відхилень (кут α – кут згинання щиколотки), а точки активації контролера - на рис. 3.2. Опорні значення для робочих точок були отримані з цих експериментів і були використані для валідації розробленої стратегії управління. Для одержання робочих точок для контролера ми використовуємо вбудований кінематичний двигок реального часу, який використовує чітку інформацію про унікальну структуру скелета для забезпечення високо-точних даних. Точна функціональна оцінка вимагає ретельної ідентифікації та відстеження даних. Дані тестування були зібрані з платформи і використані для збору та аналізу даних. Було проведено серію експериментів [9].

Використовуючи доступні інструменти MatLab, ми розраховуємо опорні значення кутів і швидкостей кутових траєкторій за даними експериментами (1) - (5), які проводилися з 2 добровольцями (70 кг маси тіла / 183 см. кг маси тіла / 179 см, відповідно). Кадр даних [350-500] відповідає часовому інтервалу $t(sec) = [4; 8]$. На рис. 3.3, 3.5, 3.7, 3.9, 3.11 наведено графік наборів даних (1) - (5).

Відхилення максимального кута дорівнює $\sin(\alpha) = 0,44$ в експерименті 5 (див. рис. 3.11 і 3.12). Кадри даних експериментів (1) - (5) тривають від точок A (початкові відхилення) до точок C (кінцеві значення). Точки B є контрольними точками активації (див. рис. 3.3, 3.5, 3.7, 3.9, 3.11). Для обчислення розрахункових швидкостей. Ми підготуємо криву завантажених даних, використовуючи відповідну панель інструментів Curve Fitting Toolbox. Ми використовуємо поліноми для інтерполяції

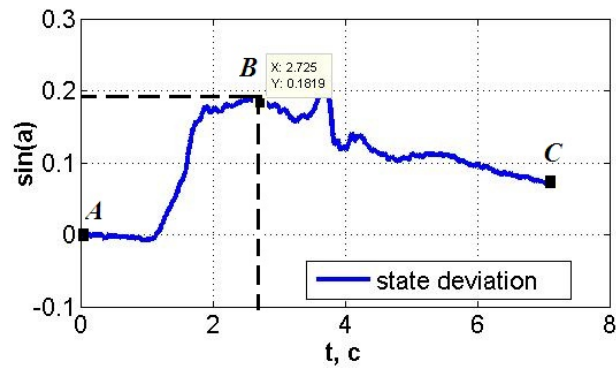


Рис. 3.3: Моделювання збурень - Експеримент (1): А - початок експерименту; В - максимальне відхилення; С - кінцеве положення (цільове)

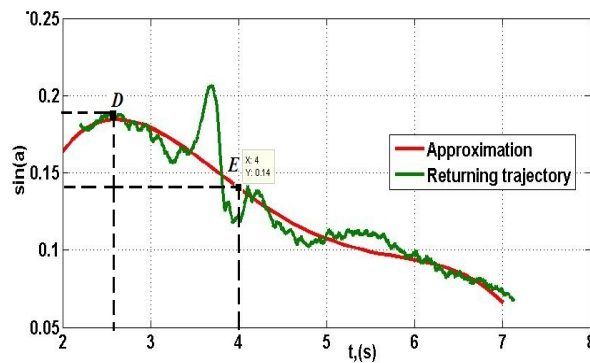


Рис. 3.4: Експеримент (1) - зворотня траєкторія: D - точка активації керування; E - обчислення похідної;

та характеризуємо дані, використовуючи глобальну підгонку для того, щоб отримати прості емпіричні моделі кожного експерименту. Основними перевагами відібраного поліноміального підходу є розумна гнучкість для даних, які не є занадто складними. На рис. 3.4, 3.6, 3.8, 3.10, 3.12 представлені траєкторії повернення, а також їх сплайн-наближення. Тоді еталонні швидкості обчислюються як похідні поліноміальних функцій зворотних доріжок, представлених на рис. 3.4, 3.6, 3.8, 3.10, 3.12 у відповідних точках E , припускаючи, що максимальне значення знаходиться в середині траєкторія.

Аналізуючи набори даних експериментів (1) - (5), отримаємо нелінійні моделі, які є поліномами ступенів 4-6 відповідно. Коефіцієнти розраховуються з 95 відсотками довірчого інтервалу. Точність підходів визначається сумою квадратів відхилень (SSE), R-квадратом (R-square), скоригованим R-квадратом (adjusted R-square), середньо квадратичним відхиленням (RMSE). Похідні обчислювали, як описано нижче, в точках

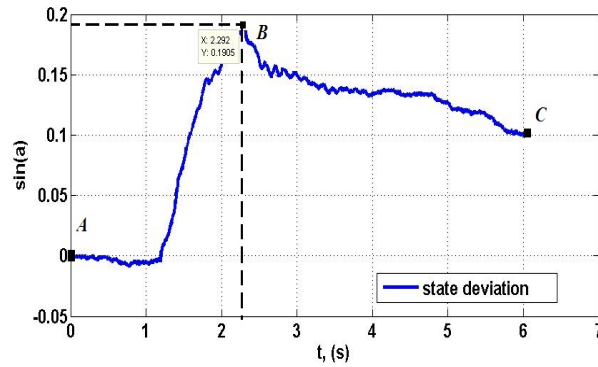


Рис. 3.5: Моделювання збурень - Експеримент (2): А - початок експерименту; В - максимальне відхилення; С - кінцеве положення (цільове)

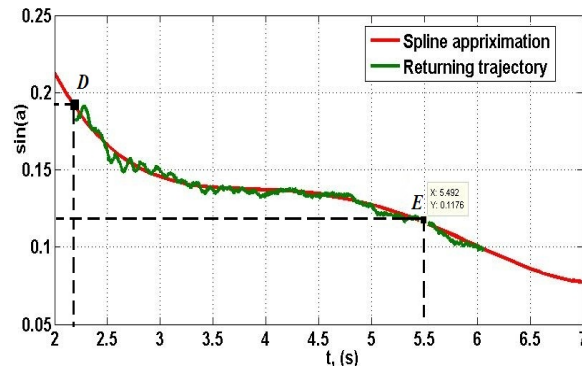


Рис. 3.6: Експеримент (2) - зворотня траєкторія: D - точка активації керування; E - обчислення похідної;

E , які відповідають середині зворотних шляхів і мають плавну деградацію [9].

Експеримент (1). З даних, отриманих в експерименті (1), отримуємо поліном ступеня 4 у наступному вигляді

$$f(x) = p_1x^4 + p_2x^3 + p_3x^2 + p_4x + p_5.$$

Отримуємо коефіцієнти:

$$p_1 = -0.001955, (-0.002281, -0.001629);$$

$$p_2 = 0.0374, (0.03131, 0.0435);$$

$$p_3 = -0.2561, (-0.2973, -0.2148);$$

$$p_4 = 0.712, (0.5929, 0.8311);$$

$$p_5 = -0.5035, (-0.6269, -0.38),$$

Результати SSE: 0.07621, R-square: 0.9141, Adjusted R-square: 0.9135, RMSE: 0.01139, (рис. 3.4).

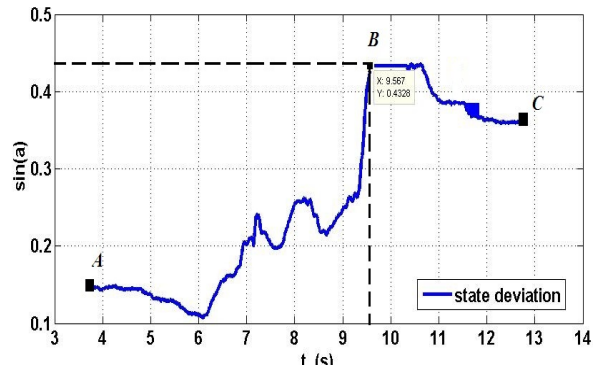


Рис. 3.7: Моделювання збурень - Експеримент (3): А - початок експерименту; В - максимальне відхилення; С - кінцеве положення (цільове)

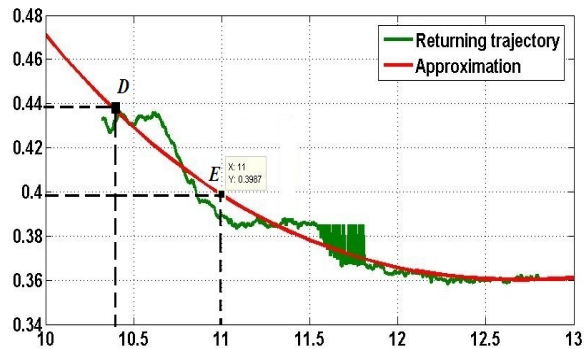


Рис. 3.8: Експеримент (3) - зворотня траєкторія: D - точка активації керування; E - обчислення похідної;

Експеримент (2). З даних, отриманих в експерименті (2), отримуємо поліном ступеня 5 у наступному вигляді

$$f(x) = p_1x^5 + p_2x^4 + p_3x^3 + p_4x^2 + p_5x + p_6.$$

Отримуємо коефіцієнти:

$$p_1 = -5.557e - 05, (-0.0002942, 0.0001831);$$

$$p_2 = 0.002943, (-0.00199, 0.007875);$$

$$p_3 = -0.04429, (-0.08418, -0.004408);$$

$$p_4 = 0.2821, (0.1246, 0.4395);$$

$$p_5 = -0.8159, (-1.119, -0.5129);$$

$$p_6 = 1.025, (0.798, 1.252).$$

SSE: 0.003351, R-square: 0.9803, Adjusted R-square: 0.9801, RMSE: 0.002705.

(рис. 3.6).

Експеримент (3). Для 3-го набору даних будуємо поліном 3-го сту-

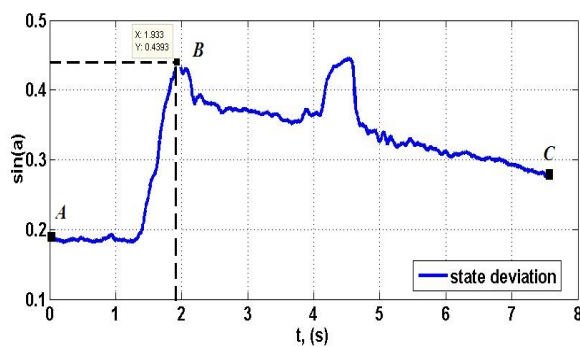


Рис. 3.9: Моделювання збурень - Експеримент (4): А - початок експерименту; В - максимальне відхилення; С - кінцеве положення (цільове)

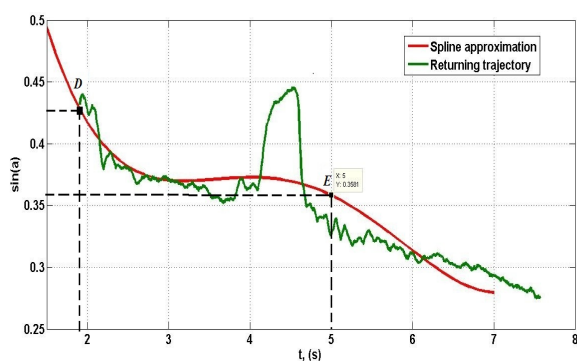


Рис. 3.10: Експеримент (4) - зворотня траєкторія: D - точка активації керування; E - обчислення похідної;

пеня у такій формі

$$f(x) = p_1x^3 + p_2x^2 + p_3x + p_4.$$

Отримуємо коефіцієнти:

$$p_1 = -0.001909, (-0.004265, 0.0004461);$$

$$p_2 = 0.08274, (0.00106, 0.1644);$$

$$p_3 = -1.178, (-2.12, -0.2357);$$

$$p_4 = 5.886, (2.269, 9.503).$$

SSE: 0.0108, R-square: 0.9431, Adjusted R-square: 0.9425, RMSE: 0.006049,

(рис. 3.8).

Експеримент (4). Для 4-го набору даних будуємо поліном 4-го ступеня

$$f(x) = p_1x^4 + p_2x^3 + p_3x^2 + p_4x + p_5.$$

Отримуємо коефіцієнти:

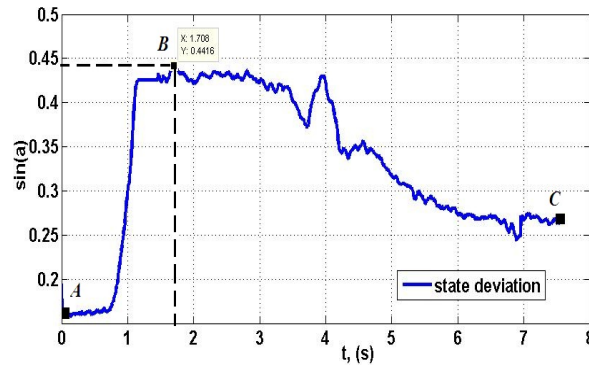


Рис. 3.11: Моделювання збурень - Експеримент (5): А - початок експерименту; В - максимальне відхилення; С - кінцеве положення (цільове)

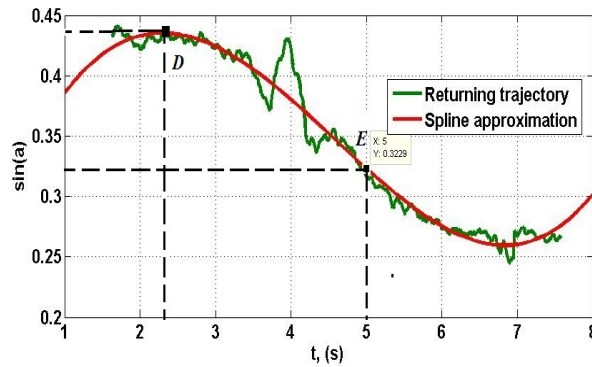


Рис. 3.12: Експеримент (5) - зворотня траєкторія: D - точка активації керування; E - обчислення похідної;

$$p_1 = 0.002227, (0.001896, 0.002558);$$

$$p_2 = -0.04223, (-0.04852, -0.03593);$$

$$p_3 = 0.2822, (0.2393, 0.3251);$$

$$p_4 = -0.8002, (-0.9235, -0.6769);$$

$$p_5 = 1.191, (1.066, 1.317).$$

$$\text{SSE: } 0.3193, \text{ R-square: } 0.7619, \text{ Adjusted R-square: } 0.7605, \text{ RMSE: } 0.02173$$

(рис. 3.10).

Експеримент (5). Для 5-го набору даних будуємо поліном 4-го ступеня

$$f(x) = p_1x^4 + p_2x^3 + p_3x^2 + p_4x + p_5.$$

Отримуємо коефіцієнти:

$$p_1 = 3.098e - 05(-0.0001003, 0.0001623);$$

$$p_2 = 0.003211(0.0007842, 0.005638);$$

$$p_3 = -0.04814(-0.06408, -0.03219);$$

$$p_4 = 0.1688(0.1251, 0.2126);$$

$$p_4 = 0.2617(0.2199, 0.3035).$$

SSE: 0.08068, R-square: 0.9755, Adjusted R-square: 0.9754, RMSE: 0.01067, (рис. 3.12) .

Аналізуючи кожен набір даних (1) - (5), до таблиці 1 зібрані наступні дані: x_{ref} – цільові значення відхилень кутів кожного експерименту; x_{ref} – кутові швидкості, розраховані як похідні сплайнів в певних точках E , як описано вище; початкові значення x_{init} (координати точок D) [9].

3.2 Алгоритм оптимальної стабілізації

Для того, щоб обрати правильну відповідь, центр вибору управління обчислює механічні стани для вибору кількості відповідей, необхідних для протидії порушенням. Цей центр оцінює різницю між оціненим станом і бажаним станом і обирає відповідну траєкторію, а також вигоди від зворотного зв'язку. Контрольний відбірний центр ефективно вибирає відповідну матрицю посилення. Модель системи керування (19) передбачає отримання необхідного вектору стану $\bar{x}(t)$ з найвищих рівнів ЦНС, порівняння його з вимірним станом і генерування команди u в просторі u_{ank} і u_{hip} . Регулювання станів $x(t)$ до заданого стану $\bar{x}(t)$, визначеного з експериментальних даних, було отримано з біомеханічних випробувань.

Будемо обчислювати лінійно-квадратичний регулятор для визначення траєкторій, що мінімізують цільову функцію. Розроблене керування переводить стани системи до заданих точок, визначених за допомогою статичних експериментальних даних, які вимірюються. [9]

Зробивши заміну $x(t) = \bar{x} + y$, $u = \bar{u} + v$, де \bar{u} визначена з рівняння

$$A\bar{x} + B\bar{u} = 0.$$

Тоді отримуємо цільове керування, за якого система $\dot{y} = Ay + Bv$ має нульову позицію рівноваги:

$$\bar{u} = -B^{-1}A\bar{x}.$$

Шукатимемо u у вигляді

$$u = \bar{u} - K(x - \bar{x}),$$

причому u повинна мати такі властивості:

- $u(x)$ є асимптотично стікою,
- $u(x)$ мінімізує функціонал.

$$J = \int_0^{\infty} [(x - \bar{x})^T Q (x - \bar{x}) + (u - \bar{u})^T R (u - \bar{u})] dt, \quad (16)$$

де Q та R є ваговими матрицями.

Оптимальна траєкторія $x_{opt}(t)$, що залежить також і від $u(t)$, мінімізує отриманий функціонал (16). Обидві матриці Q та R обираються довільно, але повинні залишатись сталими, симетричними та додатньо визначеними.

Визначемо квадратичну форму

$$V(x) = (x - \bar{x})^T P (x - \bar{x}),$$

де P – наперед задана симетрична, стала, додатньо визначена матриця. Якщо можна буде показати, що $V < 0$ на усіх траєкторіях (окрім $x = \bar{x}$, де $V = 0$), тоді усі траєкторії повинні наближатись до \bar{x} з визначеним раніше керуванням. Отримується такий результат, оскільки V має єдиний глобальний мінімум на \bar{x} . Якщо кожна траєкторія зменшує V , тоді будь-яка траєкторія наближається до \bar{x} [5].

Тоді використовуючи теорію з підрозділу 2.1, та розв'язавши алгебраїчне рівняння Рікати (невідомо матриця P)

$$PA + A^T P - PBR^{-1}B^T P = -Q, \quad (17)$$

отримуємо $K = R^{-1}B^T P$. За наперед заданими матрицями Q та R , розв'язавши рівняння (17), отримуємо матрицю P , а знаючи її, можна отримати K . Тоді потрібне керування знайдене.

Результати експерименту

Оберемо вагові матриці

$$Q = \begin{pmatrix} 0.9 & 0 & 0 & 0 \\ 0 & 2.1 & 0 & 0 \\ 0 & 0 & 2.2 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}, R = \begin{pmatrix} 1.3 & 0 \\ 0 & 1 \end{pmatrix}.$$

Тоді матриця P набуває вигляду

$$P = \begin{pmatrix} 1.9637 & 0.2048 & 1.0523 & 0.3218 \\ 0.2048 & 2.636 & 0.3823 & 0.1777 \\ 1.0523 & 0.3823 & 2.3333 & 0.7194 \\ 0.3218 & 0.1777 & 0.7194 & 0.2963 \end{pmatrix},$$

а матриця K :

$$K = R^{-1}B^T P = \begin{pmatrix} 0.8095 & 0.2941 & 1.7949 & 0.5534 \\ -0.2195 & 1.4098 & -0.3414 & 1.7197 \end{pmatrix}.$$

З вигляду системи стає очевидним, що для будь-яких $x_0, \bar{x}, \bar{u} = (0, 0)^T$. Нехай $x_0 = (2, 4, 0.8, 1)^T, \bar{x} = (3, 9, 0, 0)^T$. Проінтегрувавши систему (19) чисельним методом Рунге-Кутти четвертого порядку, отримуємо такі результати на проміжку $0 \leq t \leq 40$ (рис. 3.13).

Далі змінимо x_0 та \bar{x} . Нехай $x_0 = (3, 14, 23, 27.4)^T, \bar{x} = (100, 9, 0, 0)^T$. Проінтегрувавши систему (19) чисельним методом Рунге-Кутти четвертого порядку, отримуємо такі результати на проміжку $0 \leq t \leq 40$ (рис. 3.14).

Нехай тепер $x_0 = (5, 7, 12, 5.4)^T, \bar{x} = (13, 2, 0, 0.03)^T$. Проінтегрувавши систему (19) чисельним методом Рунге-Кутти четвертого порядку, отримуємо такі результати на проміжку $0 \leq t \leq 40$ (рис. 3.15).

Тепер змінимо вагову матрицю Q , залишивши при цьому початкову,

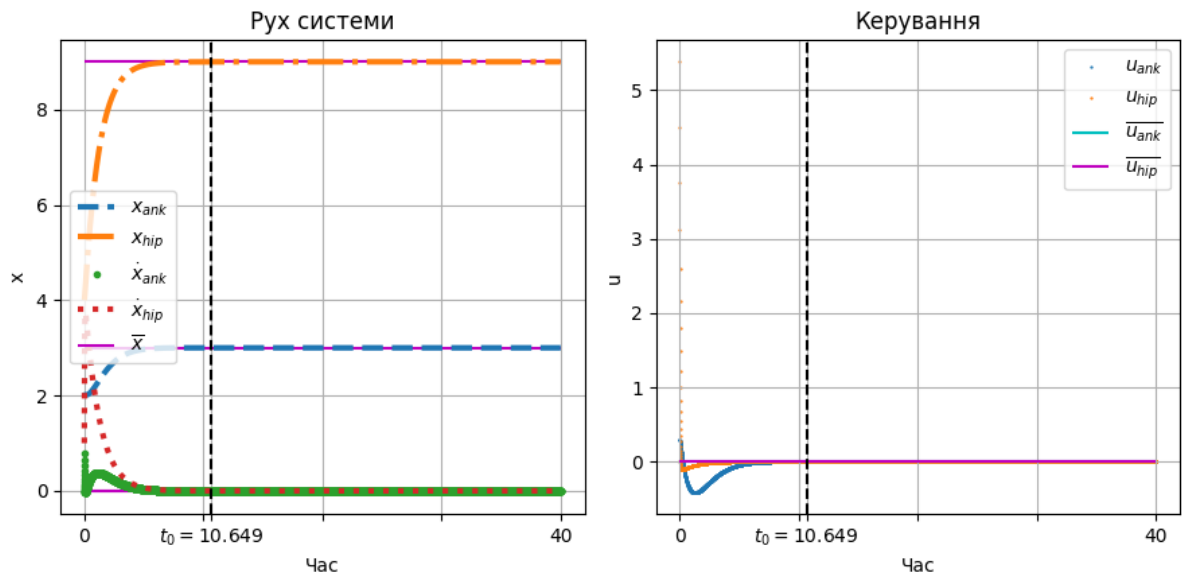


Рис. 3.13: $x_0 = (2, 4, 0.8, 1)^T, \bar{x} = (3, 9, 0, 0)^T$.

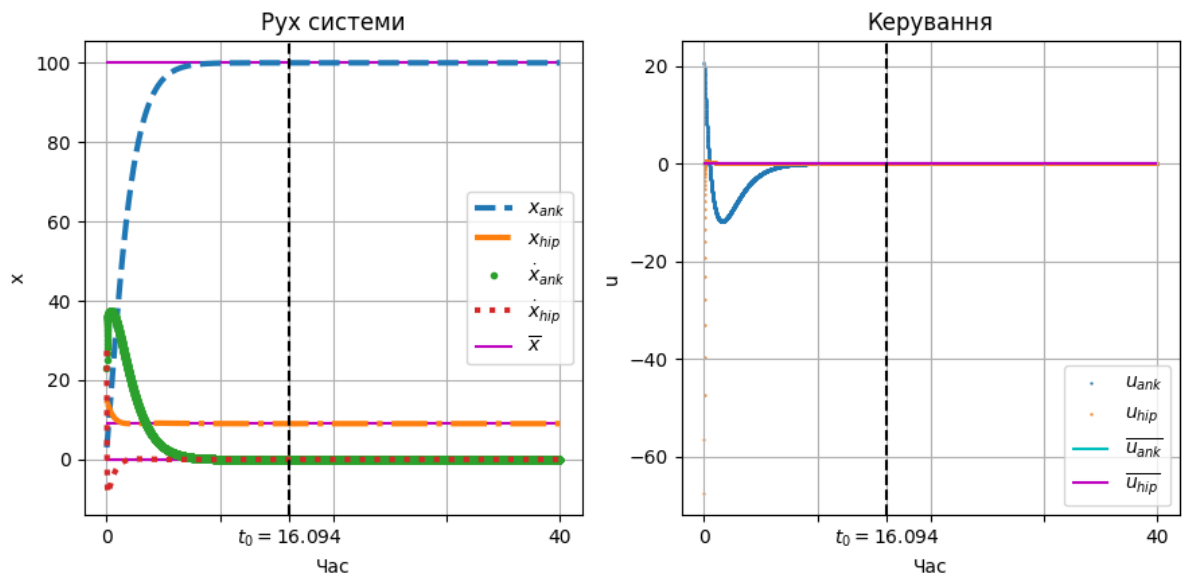


Рис. 3.14: $x_0 = (3, 14, 23, 27.4)^T, \bar{x} = (100, 9, 0, 0)^T$.

цільову точку та матрицю R . Нехай

$$Q = \begin{pmatrix} 0.2 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 8 \end{pmatrix}.$$

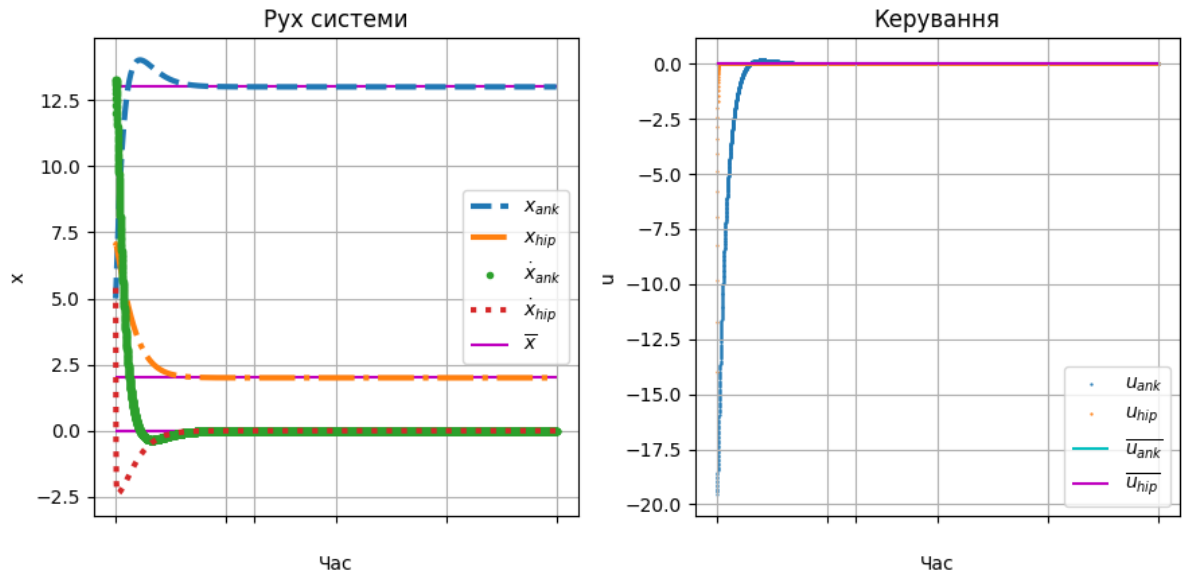


Рис. 3.15: $x_0 = (5, 7, 12, 5.4)^T, \bar{x} = (13, 2, 0, 0.03)^T$.

Тоді матриця P набуває вигляду

$$P = \begin{pmatrix} 1.0954 & 0.0853 & 0.5004 & 0.1538 \\ 0.0853 & 5.7961 & 0.4371 & 0.2175 \\ 0.5004 & 0.4371 & 2.7502 & 0.8469 \\ 0.1538 & 0.2175 & 0.8469 & 0.3808 \end{pmatrix},$$

а матриця K :

$$K = R^{-1}B^T P = \begin{pmatrix} 0.385 & 0.3362 & 2.1155 & 0.6514 \\ -0.0857 & 1.9629 & -0.4276 & 2.8077 \end{pmatrix}.$$

Тоді результат виконання програми (рис. 3.16). Змінемо початкову і

кінцеву точки:

$$x_0 = (3, 9, 7, 2)^T, \bar{x} = (10, 2, 0, 0)^T.$$

Оберемо вагові матриці

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 8 \end{pmatrix}, R = \begin{pmatrix} 4 & 0 \\ 0 & 5 \end{pmatrix}.$$

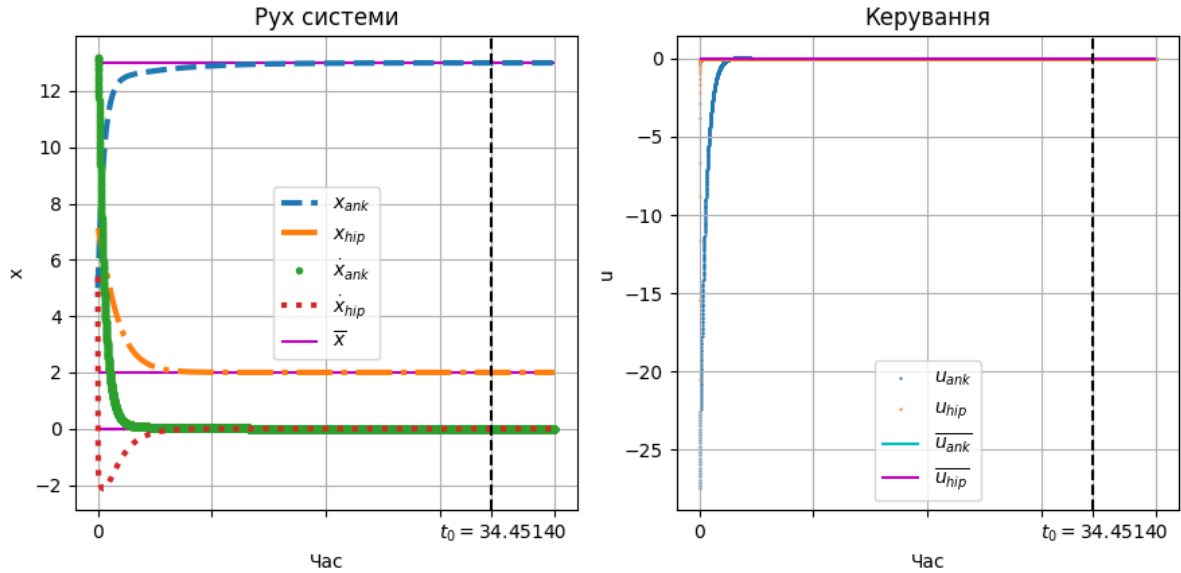


Рис. 3.16: $x_0 = (5, 7, 12, 5.4)^T$, $\bar{x} = (13, 2, 0, 0)^T$.

Тоді матриця P набуває вигляду

$$P = \begin{pmatrix} 2.8933 & 0.2805 & 1.6954 & 0.5161 \\ 0.2805 & 5.8946 & 0.7088 & 0.3826 \\ 1.6954 & 0.7088 & 4.955 & 1.5202 \\ 0.5161 & 0.3826 & 1.5202 & 0.7086 \end{pmatrix},$$

а матриця K :

$$K = R^{-1}B^T P = \begin{pmatrix} 0.5651 & 0.2363 & 1.6517 & 0.5067 \\ -0.1023 & 0.9788 & -0.2274 & 1.4138 \end{pmatrix}.$$

Тоді отримуємо такі результати (рис. 3.17).

3.3 Алгоритм методу функцій Ляпунова

Підставимо (13) - (14) у (12). Отримуємо

$$\lambda(x, t) = \frac{W(x, t) + \frac{\partial V(x, t)}{\partial t} + \langle \text{grad}_x V(x, t), F(x, t) \rangle}{\|G^*(x, t) \text{grad}_x V(x, t)\|^2}. \quad (18)$$

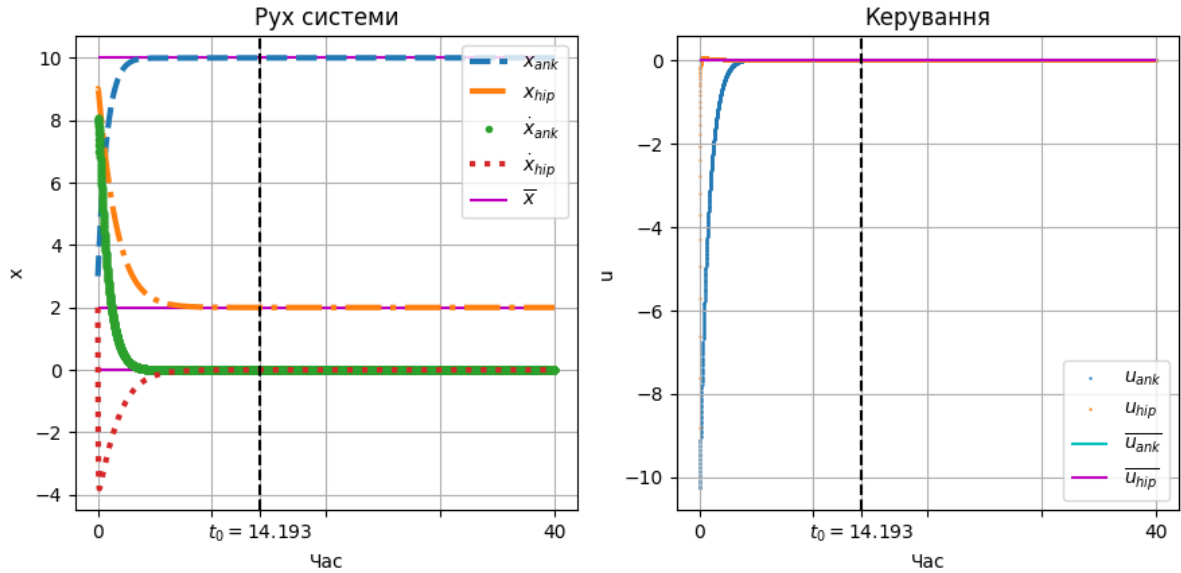


Рис. 3.17: $x_0 = (3, 9, 7, 2)^T$, $\bar{x} = (10, 2, 0, 0)^T$.

Підставивши (18) у (14), отримуємо стабілізуюче керування.
Будемо обирати функції $V(x)$, $W(x)$ у вигляді

$$V(x) = (x - \bar{x})^T P (x - \bar{x}), \quad W(x) = \frac{V(x)}{2}.$$

Тоді керування має вигляд

$$u(x, t) = \frac{\frac{(x - \bar{x})^T P (x - \bar{x})}{2} + \langle \text{grad}_x((x - \bar{x})^T P (x - \bar{x})), F(x, t) \rangle}{\|G^*(x, t) \text{grad}_x((x - \bar{x})^T P (x - \bar{x}))\|^2}.$$

В позначеннях нашої системи: $G(x, t) = B$, $F(x, t) = A$.

Результати експерименту

Оберемо функцію

$$V(x) = (x - \bar{x})^T P (x - \bar{x}),$$

де

$$P = \begin{pmatrix} 1.9637 & 0.2048 & 1.0523 & 0.3218 \\ 0.2048 & 2.636 & 0.3823 & 0.1777 \\ 1.0523 & 0.3823 & 2.3333 & 0.7194 \\ 0.3218 & 0.1777 & 0.7194 & 0.2963 \end{pmatrix}.$$

Тоді на проміжку $0 \leq t \leq 30$ отримуємо такий результат (рис. 3.18). Для інших початкових та цільових точок при тієї ж матриці P , результати

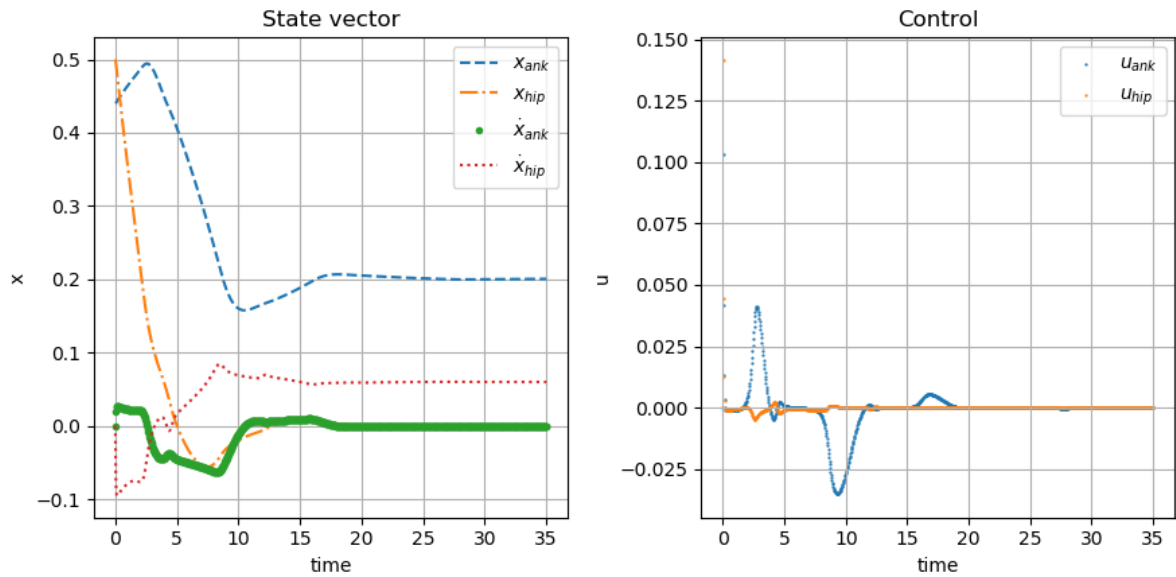


Рис. 3.18: $x_0 = (0.44, 0.5, 0, 0)^T, \bar{x} = (0.2, 0, 0, 0.06)^T$.

можна побачити на рис. 3.19 та рис. 3.20. Змінимо тепер матрицю P :

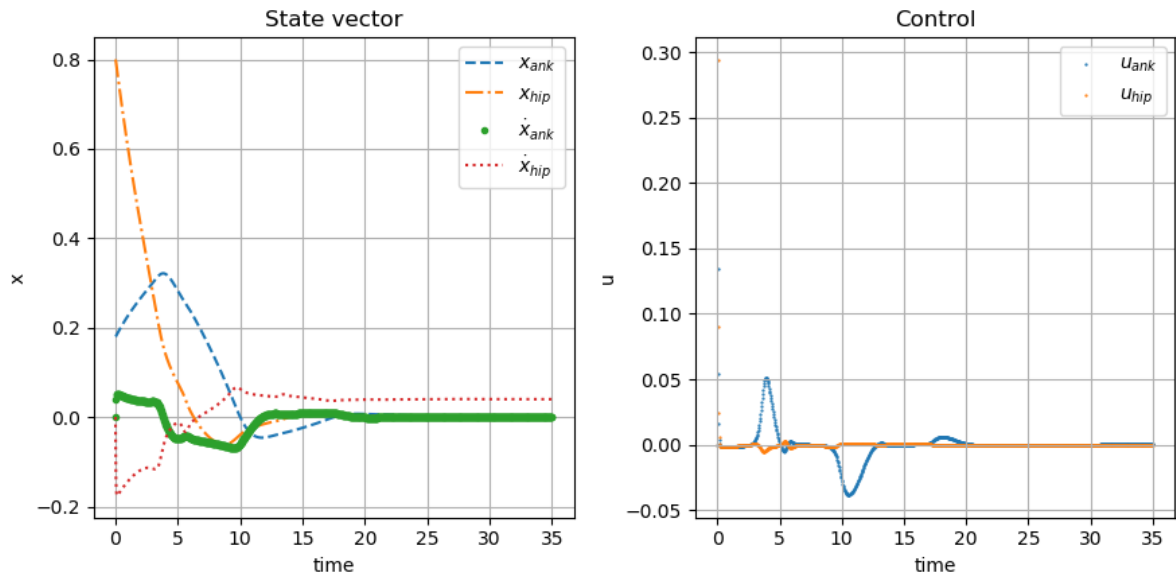


Рис. 3.19: $x_0 = (0.18, 0.8, 0, 0)^T, \bar{x} = (0, 0, 0, 0.04)^T$.

$$P = \begin{pmatrix} 2.8933 & 0.2805 & 1.6954 & 0.5161 \\ 0.2805 & 5.8946 & 0.7088 & 0.3826 \\ 1.6954 & 0.7088 & 4.955 & 1.5202 \\ 0.5161 & 0.3826 & 1.5202 & 0.7086 \end{pmatrix}.$$

Тоді отримуємо результат на рис. 3.21. Змінимо початкову і кінцеву то-

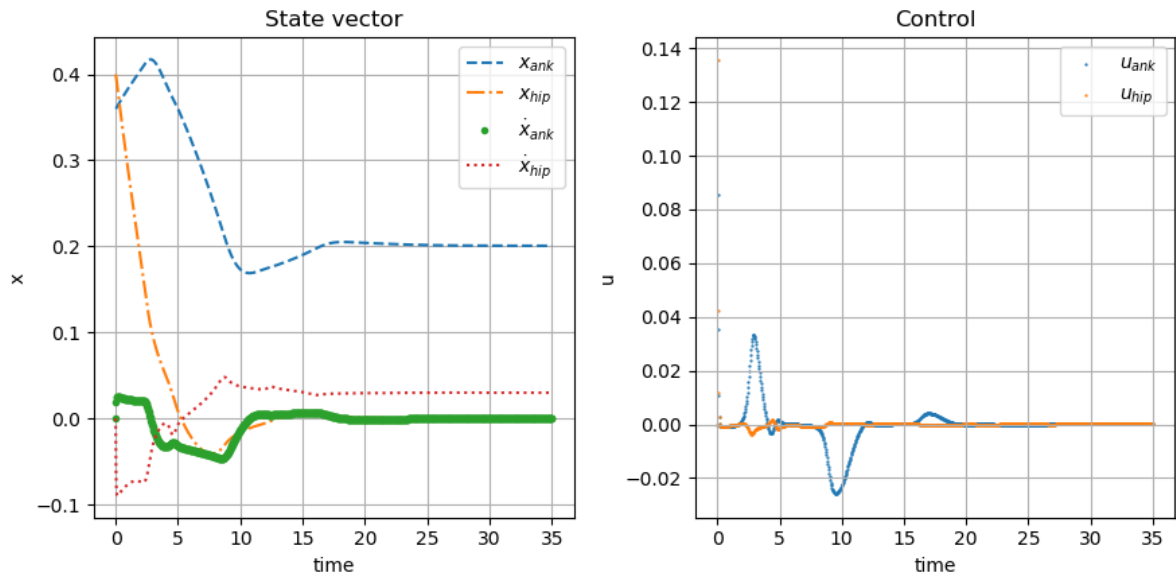


Рис. 3.20: $x_0 = (0.36, 0.4, 0, 0)^T, \bar{x} = (0.2, 0, 0, 0.03)^T$.

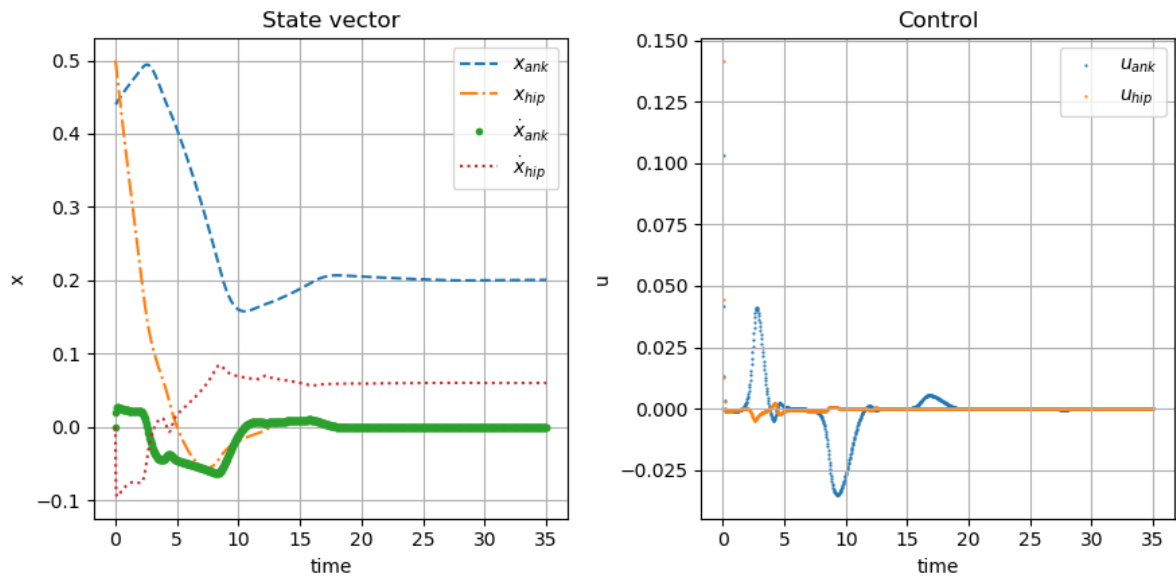


Рис. 3.21: $x_0 = (0.18, 0.8, 0, 0)^T, \bar{x} = (0, 0, 0, 0.04)^T$.

чки:

$$x_0 = (3, 9, 7, 2)^T, \bar{x} = (10, 2, 0, 0)^T.$$

Оберемо вагові матриці

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 8 \end{pmatrix}, R = \begin{pmatrix} 4 & 0 \\ 0 & 5 \end{pmatrix}.$$

Тоді матриця P набуває вигляду

$$P = \begin{pmatrix} 2.8933 & 0.2805 & 1.6954 & 0.5161 \\ 0.2805 & 5.8946 & 0.7088 & 0.3826 \\ 1.6954 & 0.7088 & 4.955 & 1.5202 \\ 0.5161 & 0.3826 & 1.5202 & 0.7086 \end{pmatrix},$$

а матриця K :

$$K = R^{-1}B^T P = \begin{pmatrix} 0.5651 & 0.2363 & 1.6517 & 0.5067 \\ -0.1023 & 0.9788 & -0.2274 & 1.4138 \end{pmatrix}.$$

Таким чином отримуємо результат на рис. 3.22.

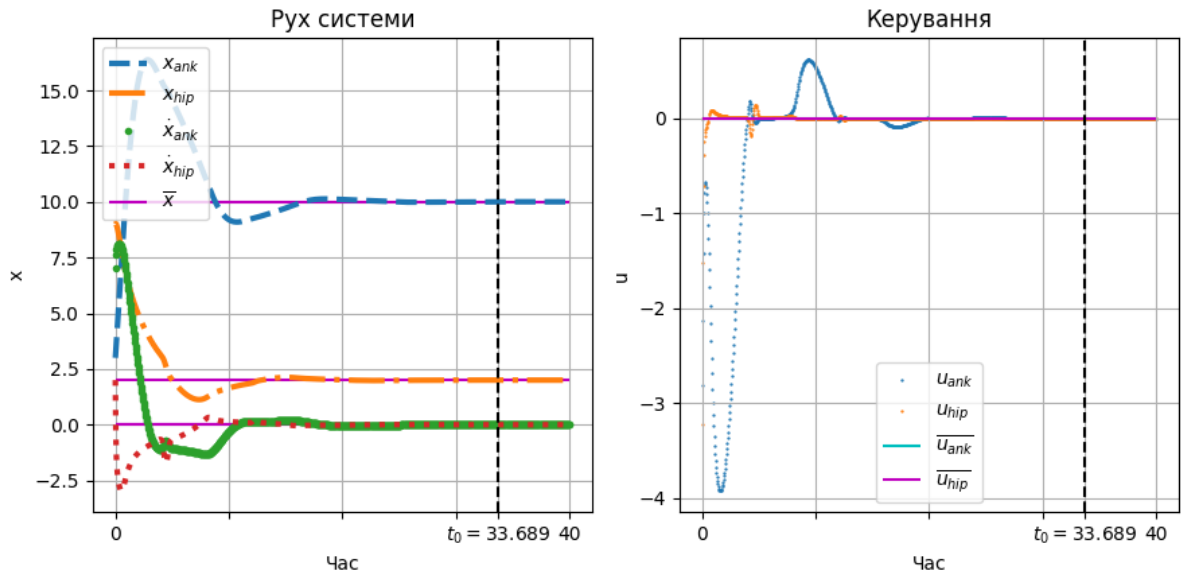


Рис. 3.22: $x_0 = (3, 9, 7, 2)^T$, $\bar{x} = (10, 2, 0, 0)^T$.

3.4 Алгоритм на основі методу демпфування

У нашому випадку маємо лінійну систему звичайних диференціальних рівнянь

$$\dot{x} = Ax + Bu. \quad (19)$$

Для нашого випадку оберемо квадратичну форму

$$V(x) = (x - \bar{x})^T P (x - \bar{x}),$$

де P – наперед задана симетрична, стала, додатньо визначена матриця. Функція $V(x)$ визначає відстань від точки x до \bar{x} . Побудуємо керування u так, щоб функція V спадала найбільшим чином вздовж траєкторії системи (19). Побудуємо повну похідну функції V на розв'язках системи (19). Отримуємо

$$W(x) = \langle P(Ax + Bu) - \bar{x}, x - \bar{x} \rangle + \langle P(x - \bar{x}), Ax + Bu - \bar{x} \rangle \rightarrow \min_u.$$

Переписавши вираз (20) у трохи іншому вигляді, отримаємо

$$\langle P(Ax - \bar{x}), x - \bar{x} \rangle + \langle P(x - \bar{x}), Ax - \bar{x} \rangle + \langle PBu, x - \bar{x} \rangle + \langle P(x - \bar{x}), Bu \rangle \rightarrow \min_u.$$

Очевидно, що u входить у цей вираз тільки лінійно, тому зробивши декілька перетворень, стає зрозумілим, що, для того, щоб мінімізувати (20), потрібно розв'язати таку оптимізаційну задачу:

$$\langle u, B^T P^T(x - \bar{x}) + B^T P(x - \bar{x}) \rangle \rightarrow \min_u. \quad (22)$$

Далі будемо розглядати:

$$\begin{cases} \psi(x) = 0, \|x\| \leq \epsilon, \\ \psi(x) = 1, \text{ інакше.} \end{cases}$$

3.4.1 Обмеження на координати вектора керування

Нехай на функцію керування $u = (u_1, u_2)^T$ маємо обмеження вигляду

$$\begin{cases} |u_1| \leq r_1, \\ |u_2| \leq r_2. \end{cases} \quad (23)$$

Перепишемо (22) у вигляді

$$u_1(b_1^T P(x - \bar{x}) + b_1^T P^T(x - \bar{x})) + u_2(b_2^T P(x - \bar{x}) + b_2^T P^T(x - \bar{x})),$$

де b_1 та b_2 – стовпчики матриці B . Оскільки (23), то в (22) впливає, що функція W отримує найменше значення при

$$\begin{cases} u_1 = -\text{sign}(b_1^T P(x - \bar{x}) + b_1^T P^T(x - \bar{x})), \\ u_2 = -\text{sign}(b_2^T P(x - \bar{x}) + b_2^T P^T(x - \bar{x})). \end{cases} \quad (24)$$

Підставляючи (24) у (19), отримуємо оптимальну систему керування по відношенню до демпфування функції $V(x, t)$, в якій права частина є розривною.

Результати експерименту

Оберемо функцію $V(x, t)$ у вигляді

$$V(x) = (x - \bar{x})^T P(x - \bar{x}), P = \begin{pmatrix} 1.9637 & 0.2048 & 1.0523 & 0.3218 \\ 0.2048 & 2.636 & 0.3823 & 0.1777 \\ 1.0523 & 0.3823 & 2.3333 & 0.7194 \\ 0.3218 & 0.1777 & 0.7194 & 0.2963 \end{pmatrix},$$

Для параметрів

$$\begin{aligned} r_1 &= 0.2, \quad r_2 = 0.1, \\ x_0 &= (0.44, 0.5, 0, 0)^T, \quad \bar{x} = (0.2, 0, 0, 0.06)^T, \end{aligned}$$

отримуємо результат на рис. 3.23.

Змінивши початкові і кінцеві точки, отримуємо результати на рис. 3.24.

Змінимо початкову і кінцеву точки:

$$x_0 = (3, 9, 7, 2)^T, \quad \bar{x} = (10, 2, 0, 0)^T$$

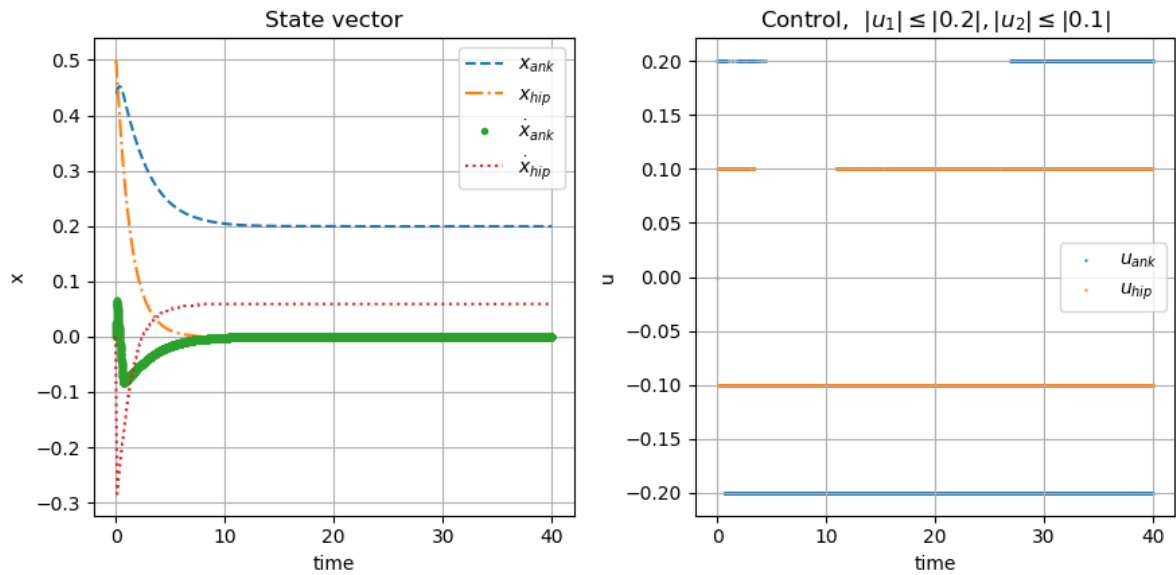


Рис. 3.23: $r_1 = 0.2, r_2 = 0.1$, та $x_0 = (0.44, 0.5, 0, 0)^T, \bar{x} = (0.2, 0, 0, 0.06)^T$.

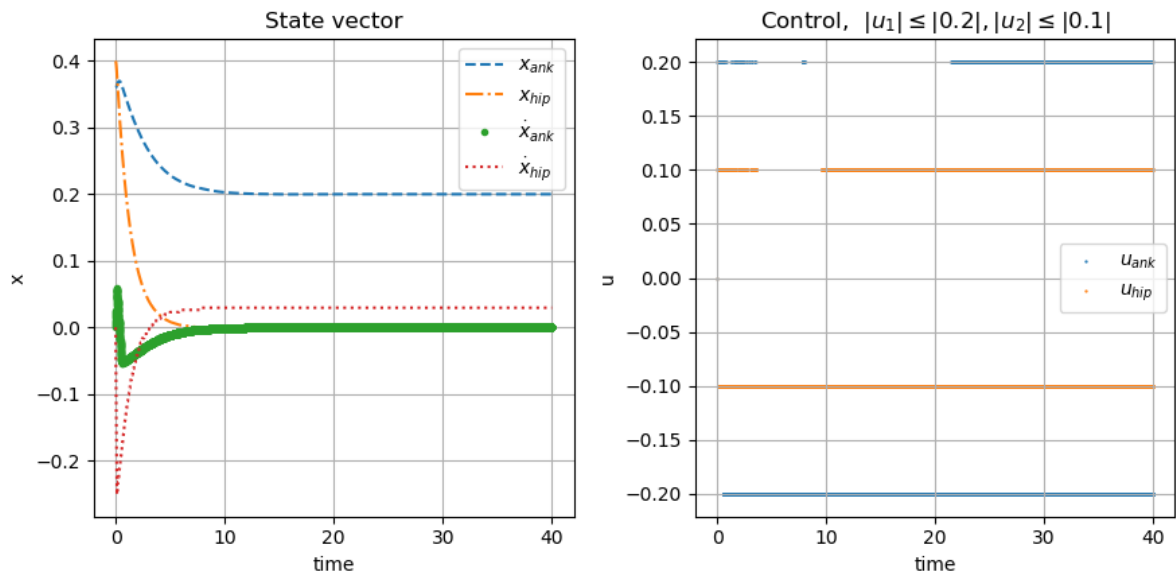


Рис. 3.24: $r_1 = 0.2, r_2 = 0.1$, та $x_0 = (0.36, 0.4, 0, 0)^T, \bar{x} = (0.2, 0, 0, 0.03)^T$.

Оберемо вагові матриці

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 8 \end{pmatrix}, R = \begin{pmatrix} 4 & 0 \\ 0 & 5 \end{pmatrix}.$$

Тоді матриця P набуває вигляду

$$P = \begin{pmatrix} 2.8933 & 0.2805 & 1.6954 & 0.5161 \\ 0.2805 & 5.8946 & 0.7088 & 0.3826 \\ 1.6954 & 0.7088 & 4.955 & 1.5202 \\ 0.5161 & 0.3826 & 1.5202 & 0.7086 \end{pmatrix},$$

а матриця K :

$$K = R^{-1}B^T P = \begin{pmatrix} 0.5651 & 0.2363 & 1.6517 & 0.5067 \\ -0.1023 & 0.9788 & -0.2274 & 1.4138 \end{pmatrix}.$$

Також мають місце значення для обмежень на керування:

$$r_1 = 3, \quad r_2 = 2.$$

Результат для такого набору параметрів можна побачити на рис. 3.25.

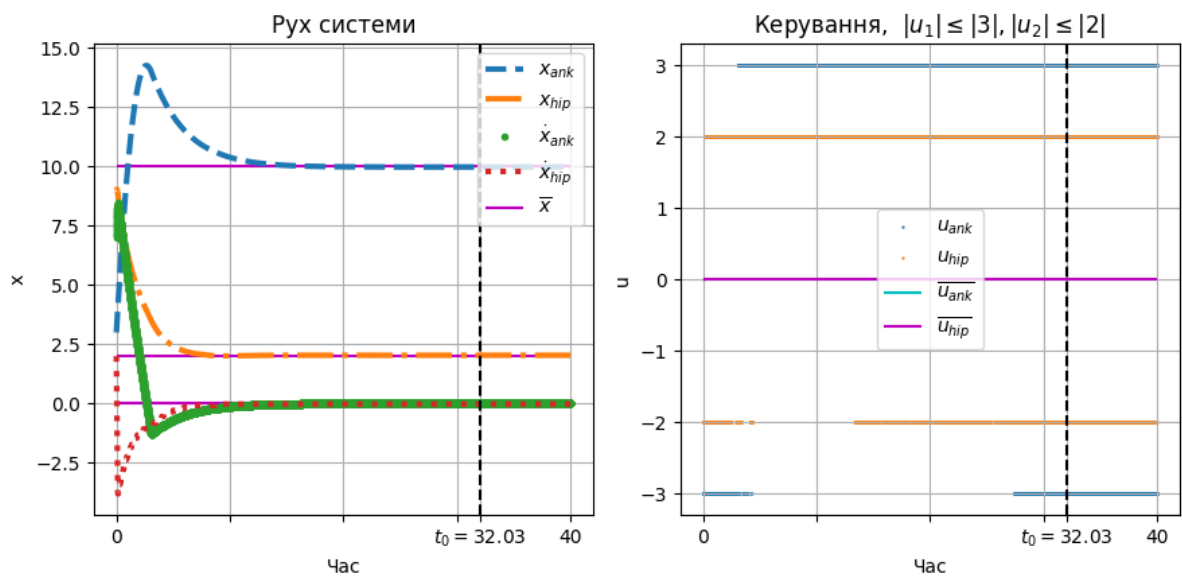


Рис. 3.25: $r_1 = 3, r_2 = 2, x_0 = (3, 9, 7, 2)^T, \bar{x} = (10, 2, 0, 0)^T$.

3.4.2 Обмеження на норму вектора керування

Нехай маємо обмеження на $u = (u_1, u_2)^T$ вигляду

$$\|u\| \leq r, \quad r \geq 0. \quad (25)$$

Розв'язавши (22), методом множників Лагранжа з обмеженнями (25), отримуємо

$$u_{min}(x) = -r \frac{B^T P^T (x - \bar{x}) + B^T P (x - \bar{x})}{\|B^T P^T (x - \bar{x}) + B^T P (x - \bar{x})\|}.$$

Результати експерименту

Наведемо декілька прикладів використання цього методу. Нехай

$$V(x) = (x - \bar{x})^T P (x - \bar{x}), \quad P = \begin{pmatrix} 1.9637 & 0.2048 & 1.0523 & 0.3218 \\ 0.2048 & 2.636 & 0.3823 & 0.1777 \\ 1.0523 & 0.3823 & 2.3333 & 0.7194 \\ 0.3218 & 0.1777 & 0.7194 & 0.2963 \end{pmatrix},$$

а $r = 0.25$. Тоді результат виконання програми на рис. 3.26

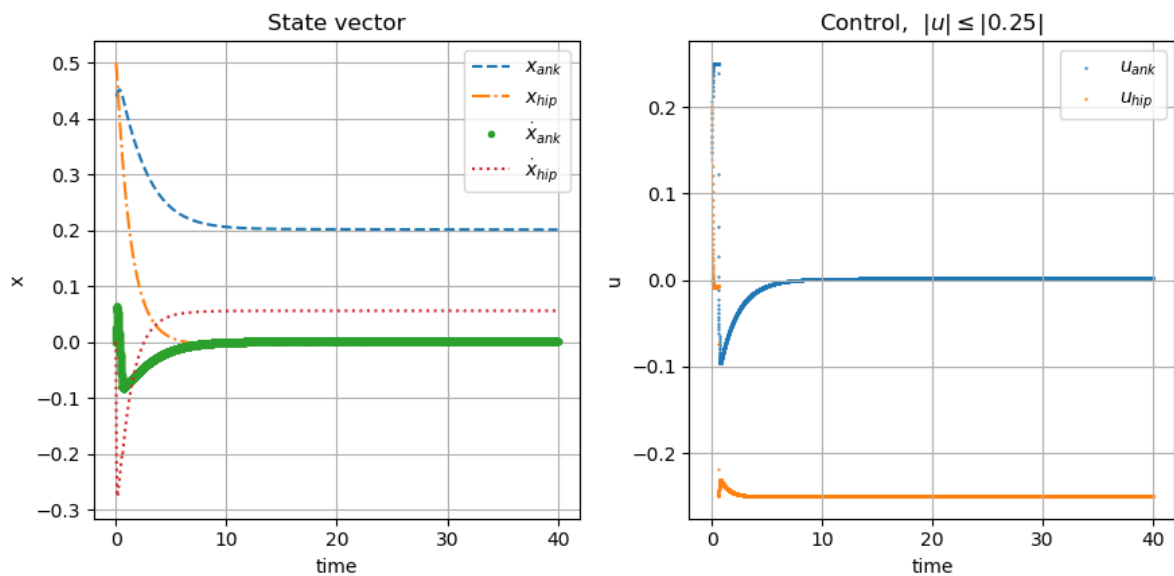


Рис. 3.26: $x_0 = (0.44, 0.5, 0, 0)^T, \bar{x} = (0.2, 0, 0, 0.06)^T$.

Далі для інших точок (рис. 3.27, 3.28):

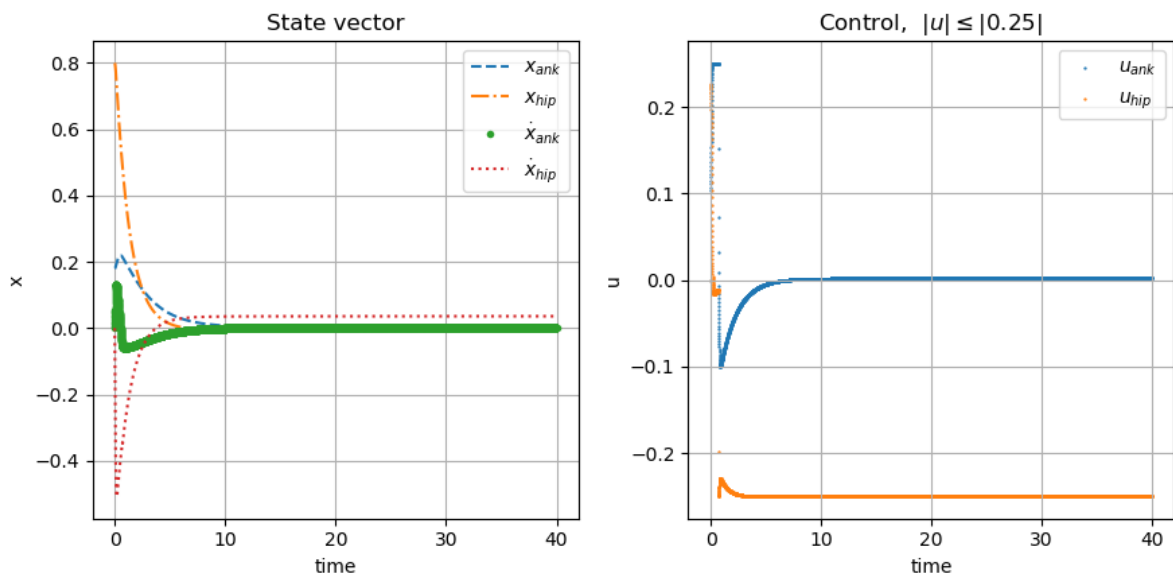


Рис. 3.27: $x_0 = (0.18, 0.8, 0, 0)^T$, $\bar{x} = (0, 0, 0, 0.04)^T$.

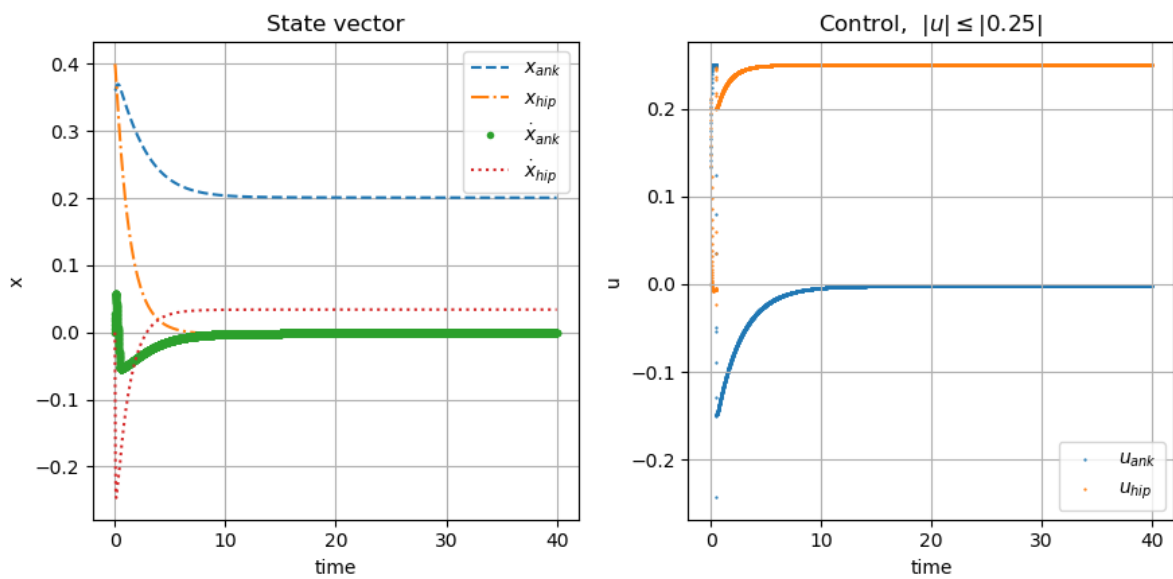


Рис. 3.28: $x_0 = (0.36, 0.4, 0, 0)^T$, $\bar{x} = (0.2, 0, 0, 0.03)^T$.

Змінимо початкову і кінцеву точки:

$$x_0 = (3, 9, 7, 2)^T, \bar{x} = (10, 2, 0, 0)^T$$

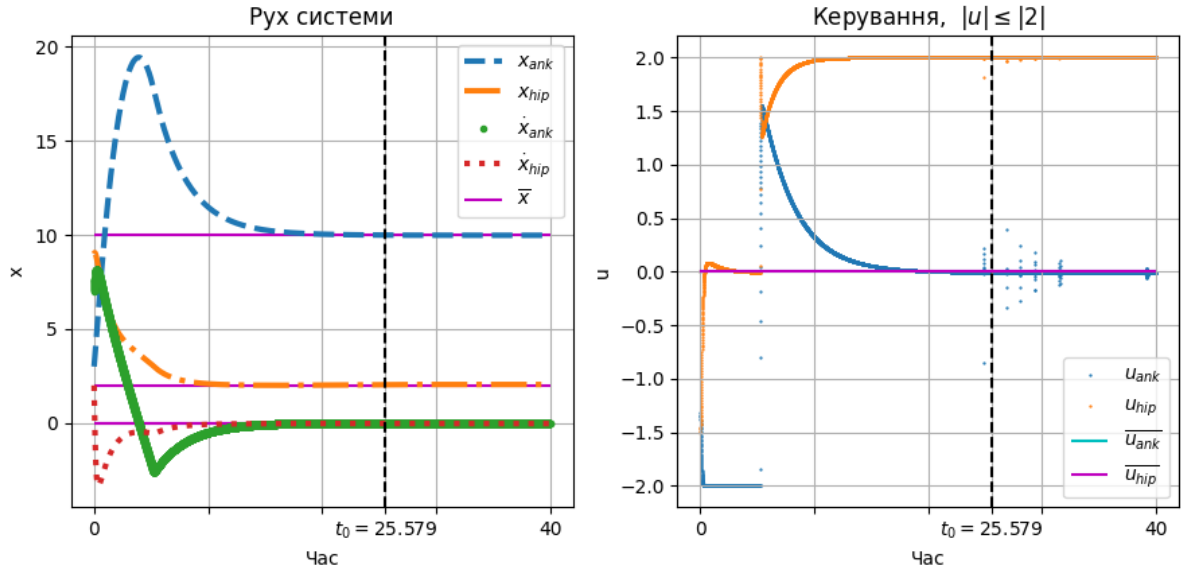


Рис. 3.29: $r_1 = 3, r_2 = 2, x_0 = (3, 9, 7, 2)^T, \bar{x} = (10, 2, 0, 0)^T$.

Оберемо вагові матриці

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 8 \end{pmatrix}, R = \begin{pmatrix} 4 & 0 \\ 0 & 5 \end{pmatrix}.$$

Тоді матриця P набуває вигляду

$$P = \begin{pmatrix} 2.8933 & 0.2805 & 1.6954 & 0.5161 \\ 0.2805 & 5.8946 & 0.7088 & 0.3826 \\ 1.6954 & 0.7088 & 4.955 & 1.5202 \\ 0.5161 & 0.3826 & 1.5202 & 0.7086 \end{pmatrix},$$

Також має місце значення для обмеження на керування:

$$r = 2.$$

Для такого набору параметрів, отримуємо результат на рис. 3.29 а матриця K :

$$K = R^{-1}B^T P = \begin{pmatrix} 0.5651 & 0.2363 & 1.6517 & 0.5067 \\ -0.1023 & 0.9788 & -0.2274 & 1.4138 \end{pmatrix}.$$

4 Розв'язання задачі про перехід із фіксованої точки на наперед задану поверхню для комплексної математичної моделі руху нижніх кінцівок людини

4.1 Постановка задачі про перехід із фіксованої точки на наперед задану поверхню

Розглянемо систему керування

$$\frac{dx(t)}{dt} = f(x(t), u(t), t), \quad (26)$$

де x — n -вимірний вектор фазових координат, $f(x, u, t)$ — n -вимірна вектор-функція, неперервна за сукупністю змінних, $u = u(t)$ — r -вимірний вектор керування, що належить класу кусково неперервних функцій, $u(t) \in G$, G — компакт в R^r , $x(t_0) = x_0, t \geq t_0$ [1]. Розв'язок системи (26), що відповідає керуванню u та початковій умові $x(t_0) = x_0$ будемо позначати $x(t, u, x_0, t_0)$.

Нехай S - деяка поверхня, яка має вигляд:

$$S = \{(x, t) : \psi(x, t) = 0\}, \quad (27)$$

де $\psi \in C(R^{n+1})$ і задача керування системою (15) полягає у знаходженні допустимого керування $u(t) \in G$ такого, що інтегральна крива системи (15) в деякий момент часу попадає з фіксованого положення на поверхню S [1].

4.2 Лінеаризація

Наступним кроком для розв'язку задачі використаємо лінеаризацію системи. Для цього будемо вважати, що

$$\theta_1, \theta_2, \theta_3 \rightarrow 0.$$

Розкладемо, наприклад, функцію $\cos(\theta_1 - \theta_2)$ в ряд Тейлора. Оскільки $\theta_1, \theta_2 \rightarrow 0$, то $\theta_1 - \theta_2 \rightarrow 0$. Тоді

$$\cos(\theta_1 - \theta_2) = 1 - \frac{(\theta_1 - \theta_2)^2}{2} + o((\theta_1 - \theta_2)^2).$$

Аналогічним чином отримуємо розклад для $\cos(\theta_1 - \theta_3)$, $\cos(\theta_2 - \theta_3)$, $\cos(\theta_2 - \theta_3)$.

Тепер розкладемо $\sin(\theta_1)$, $\sin(\theta_2)$, $\sin(\theta_3)$. Виписавши для $\sin(\theta_1)$ відповідним чином отримаємо і для інших.

$$\sin(\theta_i) = \theta_i + o(\theta_i), i = 1..3.$$

Далі, використавши ці результати, можемо переписати (3) у такому вигляді:

$$\begin{cases} \ddot{\theta}_1 = \frac{M_1 - M_2 - M_3 - a\ddot{\theta}_2 - b\ddot{\theta}_3 + c\theta_1}{d}, \\ \ddot{\theta}_2 = \frac{M_2 - M_3 - a\ddot{\theta}_1 - f\ddot{\theta}_2 + g\theta_2}{n}, \\ \ddot{\theta}_3 = \frac{M_3 - b\ddot{\theta}_1 - m\ddot{\theta}_2 + G_3 a_3 \theta_3}{s}. \end{cases} \quad (28)$$

Цю систему можна представити у матричному вигляді

$$C\ddot{\Theta} = D\Theta + EM, \quad (29)$$

де

$$C = \begin{pmatrix} 1 & \frac{a}{d} & \frac{b}{d} \\ \frac{e}{n} & 1 & \frac{f}{n} \\ \frac{b}{s} & \frac{m}{s} & 1 \end{pmatrix}, D = \begin{pmatrix} \frac{c}{d} & 0 & 0 \\ 0 & \frac{g}{n} & 0 \\ 0 & 0 & \frac{G_3 a_3}{s} \end{pmatrix}, E = \begin{pmatrix} 1 & 1 & -1 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \quad (30)$$

Для того, щоб прибрати другу похідну Θ , введемо додаткову змінну Ω :

$$\dot{\Theta} = \Omega.$$

Таким чином, можемо переписати систему у вигляді:

$$\begin{cases} \dot{\theta}_1 = \omega_1, \\ \dot{\theta}_2 = \omega_2, \\ \dot{\theta}_3 = \omega_3, \\ C\dot{\Omega} = D\dot{\Theta} + EM. \end{cases}$$

З вигляду матриці C у (30) можна допустити, що матриця C не є виродженою. Тоді, можна домножити у (29) на C^{-1} , і тоді систему можна переписати у вигляді:

$$\begin{cases} \dot{\theta}_1 = \omega_1, \\ \dot{\theta}_2 = \omega_2, \\ \dot{\theta}_3 = \omega_3, \\ \dot{\Omega} = C^{-1}D\dot{\Theta} + C^{-1}EM. \end{cases}$$

Тоді цю систему можна виписати у такому вигляді:

$$\dot{x} = Ax + Bu, \tag{31}$$

де

$$x = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{pmatrix}, u = \begin{pmatrix} M_1 \\ M_2 \\ M_3 \end{pmatrix}, A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & & & \\ 0 & 0 & 0 & & C^{-1}D & \\ 0 & 0 & 0 & & & \end{pmatrix},$$

$$B = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ & & & C^{-1}E \end{pmatrix}.$$

Тепер, обравши довільну точку \bar{x}_1 , визначимо S з (27), як окіл точки \bar{x}_1 , тобто

$$S = O_\epsilon(\bar{x}_1),$$

де $O_\epsilon(\bar{x}_1)$ – ϵ -окіл точки \bar{x}_1 .

4.3 Використання методу демпфування

З алгоритму, наведеного в цій роботі у розділі (3.4), оскільки побудова, для нашого випадку оберемо квадратичну форму у вигляді:

$$V(x) = (x - \bar{x})^T P(x - \bar{x}),$$

де P – наперед задана симетрична, стала, додатньо визначена матриця. Функція $V(x)$ визначає відстань від точки x до \bar{x} . Побудуємо керування u так, щоб функція V спадала найбільше вздовж траєкторії системи (19). Побудуємо повну похідну функції V на розв'язках системи (19). Отримуємо

$$W(x) = \langle P(Ax + Bu) - \bar{x}, x - \bar{x} \rangle + \langle P(x - \bar{x}), Ax + Bu - \bar{x} \rangle \rightarrow \min_u.$$

Переписавши вираз (32) у трохи іншому вигляді, отримаємо

$$\langle P(Ax - \bar{x}), x - \bar{x} \rangle + \langle P(x - \bar{x}), Ax - \bar{x} \rangle + \langle PBu, x - \bar{x} \rangle + \langle P(x - \bar{x}), Bu \rangle \rightarrow \min_u.$$

Очевидно, що u входить у цей вираз тільки лінійно, тому зробивши декілька перетворень, стає зрозумілим, що, для того, щоб мінімізувати (32), потрібно розв'язати оптимізаційну задачу:

$$\langle u, B^T P^T(x - \bar{x}) + B^T P(x - \bar{x}) \rangle \rightarrow \min_u. \quad (34)$$

Далі будемо розглядати

$$\begin{cases} \psi(x) = 0, \|x\| \leq \epsilon, \\ \psi(x) = 1, \text{ інакше.} \end{cases}$$

В цьому випадку система $\dot{y} = Ay + Bv$ має нульову позицію рівноваги, якщо \bar{u} дорівнює:

$$\bar{u} = -B^{-1}A\bar{x},$$

де $x(t) = \bar{x} + y$, $u = \bar{u} + v$.

4.3.1 Обмеження на координати вектора керування

Нехай маємо обмеження на $u = (u_1, u_2, u_3)^T$ вигляду

$$\begin{cases} |u_1| \leq r_1, \\ |u_2| \leq r_2, \\ |u_3| \leq r_3. \end{cases} \quad (35)$$

Перепишемо (34) у вигляді

$$u_1(b_1^T P(x - \bar{x}) + b_1^T P^T(x - \bar{x})) + u_2(b_2^T P(x - \bar{x}) + b_2^T P^T(x - \bar{x})) + u_3(b_3^T P(x - \bar{x}) + b_3^T P^T(x - \bar{x})),$$

де b_1, b_2 та b_3 - стовпчики матриці B .

Оскільки (35), то в (34) впливає, що функція W отримує найменше значення при

$$\begin{cases} u_1 = -\text{sign}(b_1^T P(x - \bar{x}) + b_1^T P^T(x - \bar{x})), \\ u_2 = -\text{sign}(b_2^T P(x - \bar{x}) + b_2^T P^T(x - \bar{x})), \\ u_3 = -\text{sign}(b_3^T P(x - \bar{x}) + b_3^T P^T(x - \bar{x})). \end{cases} \quad (36)$$

Підставляючи (36) у (31), отримуємо оптимальну систему керування по відношенню до демпфування функції $V(x, t)$.

Результати експерименту

В усіх експериментах оберемо $\epsilon = 10^{-3}$, а матрицю P у вигляді:

$$P = \begin{pmatrix} 1.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.5 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.5 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 637.32 & 23.89 & 29.2 \\ 0.0 & 0.0 & 0.0 & 23.89 & 3.61 & 2.89 \\ 0.0 & 0.0 & 0.0 & 29.2 & 2.89 & 19.1 \end{pmatrix}$$

Експеримент 1.

Нехай параметри системи в (31)

$$\begin{aligned} m_1 &= 1.0, m_2 = 2.0, m_3 = 3.0, l_1 = 2.0, l_2 = 4.0, l_3 = 4.0, \\ a_1 &= 1.9, a_2 = 1.2, a_3 = 1.5, b_1 = 0.0, b_2 = 0.0, b_3 = 0.0, \\ G_1 &= 0.2, G_2 = 0.9, G_3 = 0.4, I_1 = 0.0, I_2 = 0.0, I_3 = 0.0, \\ a &= 40.0, b = 240.0, c = 2.98, d = 23.61, e = 28.8, f = 18.0, \\ g &= 2.68, n = 50.88, m = 18.0, s = 6.75. \end{aligned}$$

За таких параметрів матриця A набуває вигляду:

$$A = \begin{pmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & -0.004 & 0.003 \\ 0.0 & 0.0 & 0.0 & -0.005 & 0.056 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.013 & -0.009 & 0.0 \end{pmatrix},$$

а матриця B :

$$B = \begin{pmatrix} 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & -0.075 & 0.103 \\ -0.037 & 1.102 & -1.028 \\ 0.105 & -0.275 & 0.063 \end{pmatrix}.$$

Оберемо початкову точку x_0 , та кінцеву точку \bar{x} таким чином:

$$x_0 = (0, 0, 0, 0.8, 0.1, 0.4)^T,$$

$$\bar{x} = (0, 0, 0, 0.4, 0.3, 0.2)^T.$$

Для цієї точки, $\bar{u} = (-0.1018, -0.0336, -0.0178)^T$. Також оберемо значення r_1, r_2, r_3 :

$$r_1 = 0.8, \quad r_2 = 0.3, \quad r_3 = 0.4.$$

Проінтегрувавши систему (31) чисельним методом Рунге-Кутти четвертого порядку, отримуємо такі результати (рис. 4.30). Розв'язок був отриман в точці $T = 72.442$:

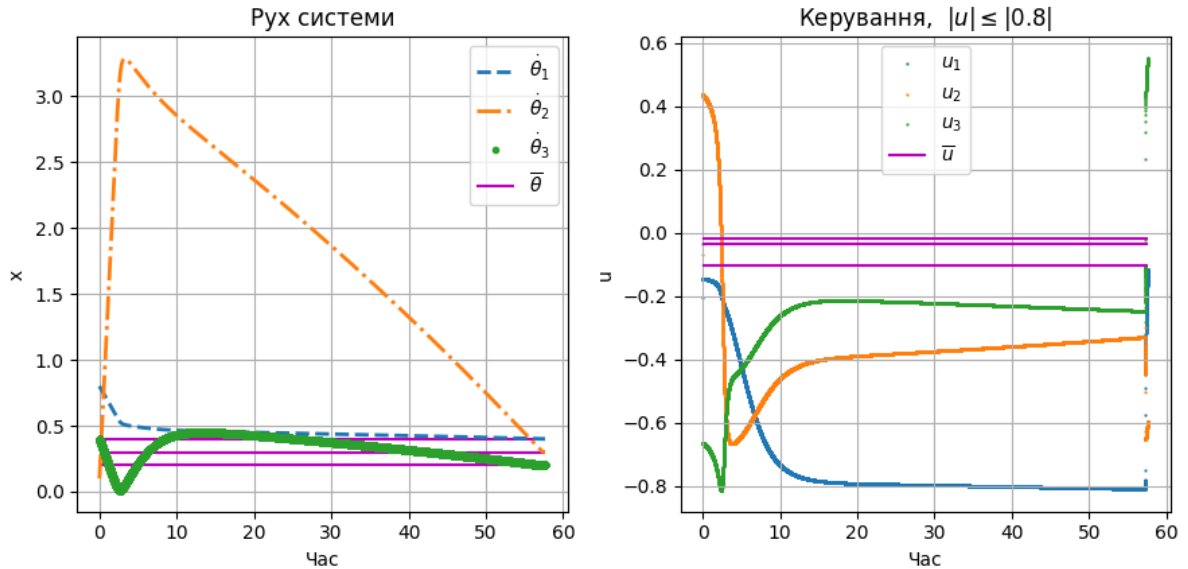


Рис. 4.30: $x_0 = (0, 0, 0, 0.8, 0.1, 0.4)^T, \bar{x} = (0, 0, 0, 0.4, 0.3, 0.2)^T$.

Експеримент 2.

Залишимо всі параметри такими ж, але змінимо обмеження на керування:

$$r_1 = 0.5, r_2 = 0.2, r_3 = 0.35.$$

Проінтегрувавши систему (31) чисельним методом Рунге-Кутти четвертого порядку, отримуємо результати, зображені на рисунку 4.31. Розв'язок був отриман в точці $T = 129.912$:

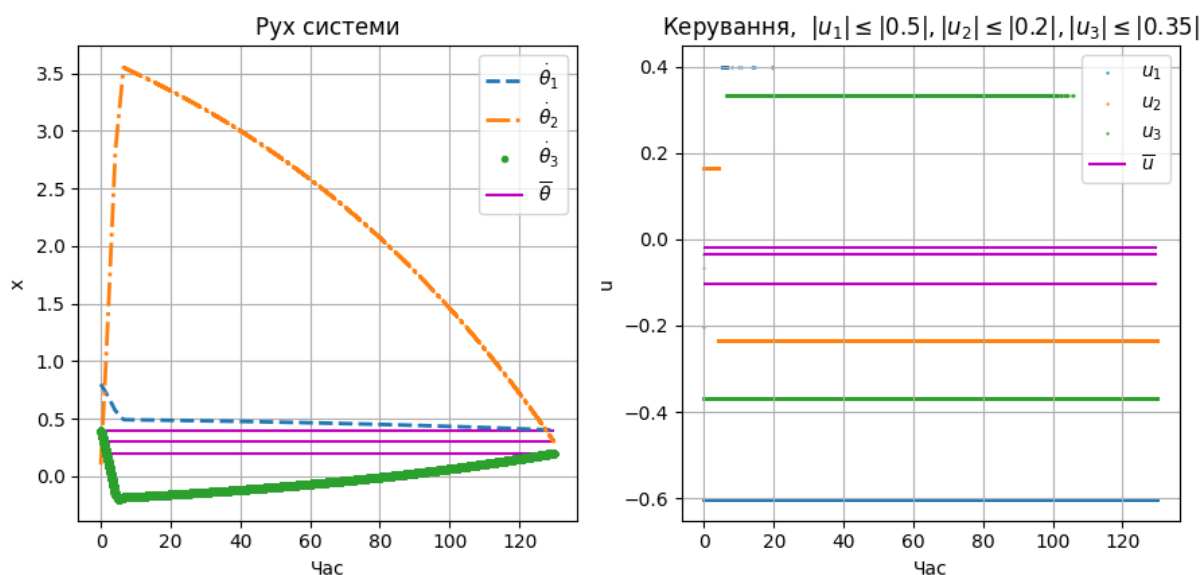


Рис. 4.31: $x_0 = (0, 0, 0, 0.8, 0.1, 0.4)^T, \bar{x} = (0, 0, 0, 0.4, 0.3, 0.2)^T$.

Експеримент 3.

Залишимо всі параметри такими ж, як в Експерименті 1, але змінимо початкову і кінцеву точки:

$$x_0 = (0, 0, 0, 0.2, 0.6, 0.5)^T,$$

$$\bar{x} = (0, 0, 0, 0.7, -0.4, 0.6)^T.$$

Проінтегрувавши систему (31) чисельним методом Рунге-Кутти четвертого порядку, отримуємо результати, зображені на рисунку 4.32. Розв'язок був отриман в точці $T = 79.75$:

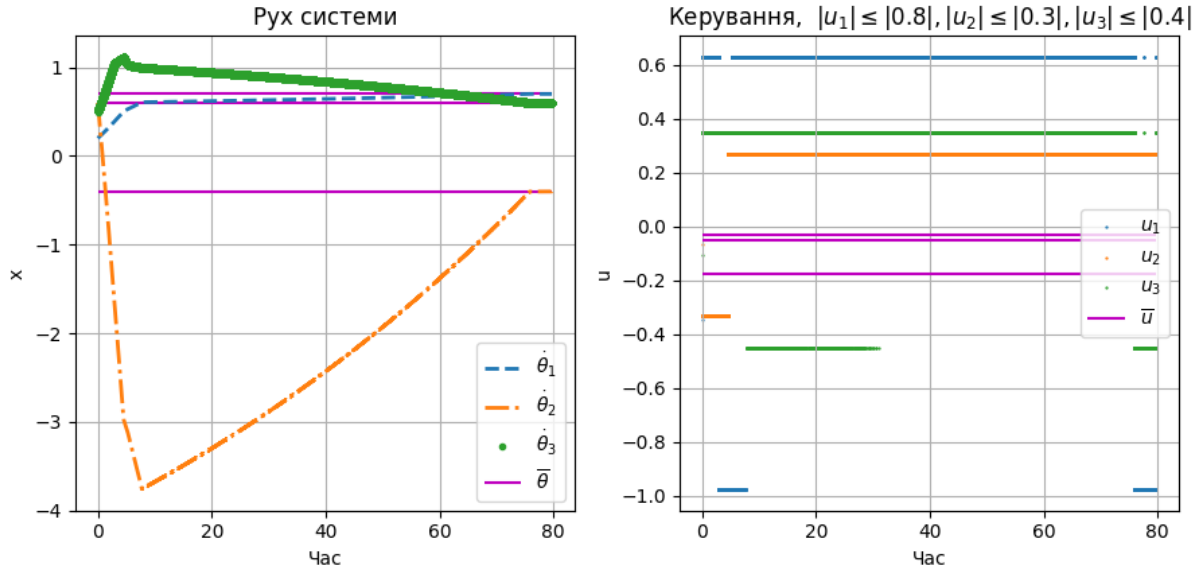


Рис. 4.32: $x_0 = (0, 0, 0, 0.2, 0.6, 0.5)^T$, $\bar{x} = (0, 0, 0, 0.7, -0.4, 0.6)^T$.

Експеримент 4.

Залишимо всі параметри такими ж, як в **Експерименті 1**, але змінимо обмеження:

$$r_1 = 0.8, \quad r_2 = 0.6, \quad r_3 = 0.9.$$

Але цього разу, розв'яжемо одразу три задачі про перехід із точки в точку, тобто спочатку потрібно перевести систему з x_0 в \bar{x}^1 , потім з \bar{x}^1 в \bar{x}^2 , і завершити в точці \bar{x}^3 . Таким чином можна побудувати будь-яку послідовність руху системи, яка потрібна у дослідженні.

Оберемо послідовність точок таким чином:

$$\begin{aligned} x_0 &= (0, 0, 0, 0.2, 0.6, 0.5)^T, \\ \bar{x}^1 &= (0, 0, 0, 0.7, -0.4, 0.6)^T, \\ \bar{x}^2 &= (0, 0, 0, 0.2, 0.1, 0.8)^T, \\ \bar{x}^3 &= (0, 0, 0, 0.6, 0.2, 0.5)^T. \end{aligned}$$

Таким чином, послідовність \bar{u}^i , $i = 1..3$:

$$\begin{aligned} \bar{u}^1 &= (-0.1739, -0.0323, -0.0533)^T, \\ \bar{u}^2 &= (-0.1727, -0.0764, -0.0711)^T, \\ \bar{u}^3 &= (-0.1752, -0.055, -0.0444)^T. \end{aligned}$$

Результат виконання програми можна побачити на рис. 4.33

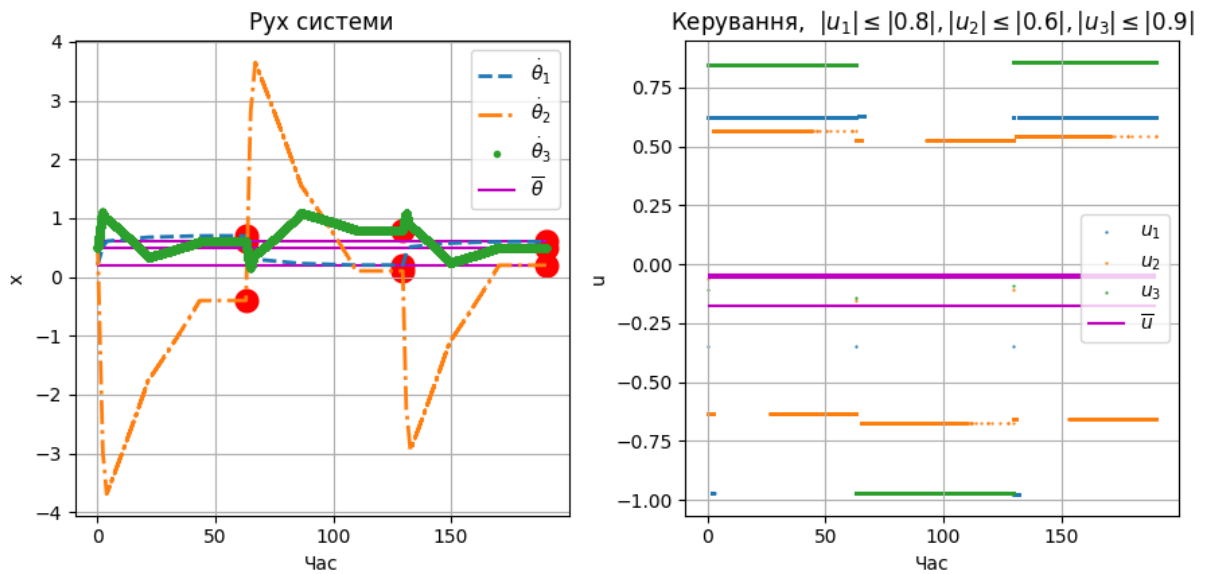


Рис. 4.33: Послідовність точок.

4.3.2 Обмеження на норму вектора керування

Нехай маємо обмеження на $u = (u_1, u_2, u_3)^T$ вигляду

$$\|u\| \leq r, \quad r \geq 0. \quad (37)$$

Розв'язавши (34), методом множників Лагранжа з обмеженнями (37), отримуємо

$$u_{min}(x) = -r \frac{B^T P^T (x - \bar{x}) + B^T P (x - \bar{x})}{\|B^T P^T (x - \bar{x}) + B^T P (x - \bar{x})\|}.$$

Результати експерименту

В усіх експериментах оберемо $\epsilon = 10^{-3}$, а матрицю P у вигляді:

$$P = \begin{pmatrix} 1.5 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.5 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.5 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 637.32 & 23.89 & 29.2 \\ 0.0 & 0.0 & 0.0 & 23.89 & 3.61 & 2.89 \\ 0.0 & 0.0 & 0.0 & 29.2 & 2.89 & 19.1 \end{pmatrix}$$

Експеримент 1.

Нехай параметри системи в (31)

$$\begin{aligned} m_1 &= 1.0, m_2 = 2.0, m_3 = 3.0, l_1 = 2.0, l_2 = 4.0, l_3 = 4.0, \\ a_1 &= 1.9, a_2 = 1.2, a_3 = 1.5, b_1 = 0.0, b_2 = 0.0, b_3 = 0.0, \\ G_1 &= 0.2, G_2 = 0.9, G_3 = 0.4, I_1 = 0.0, I_2 = 0.0, I_3 = 0.0, \\ a &= 40.0, b = 240.0, c = 2.98, d = 23.61, e = 28.8, f = 18.0, \\ g &= 2.68, n = 50.88, m = 18.0, s = 6.75 \end{aligned}$$

За таких параметрів матриця A набуває вигляду:

$$A = \begin{pmatrix} 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & -0.004 & 0.003 \\ 0.0 & 0.0 & 0.0 & -0.005 & 0.056 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.013 & -0.009 & 0.0 \end{pmatrix},$$

а матриця B :

$$B = \begin{pmatrix} 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & -0.075 & 0.103 \\ -0.037 & 1.102 & -1.028 \\ 0.105 & -0.275 & 0.063 \end{pmatrix}.$$

Оберемо початкову точку x_0 , та кінцеву точку \bar{x} таким чином:

$$\begin{aligned} x_0 &= (0, 0, 0, 0.8, 0.1, 0.4)^T, \\ \bar{x} &= (0, 0, 0, 0.4, 0.3, 0.2)^T. \end{aligned}$$

Також оберемо значення r :

$$r = 0.8.$$

Проінтегрувавши систему (31) чисельним методом Рунге-Кутти четвертого порядку, отримуємо результати на рис. 4.34. Розв'язок був отриман в точці $T = 57.658$.

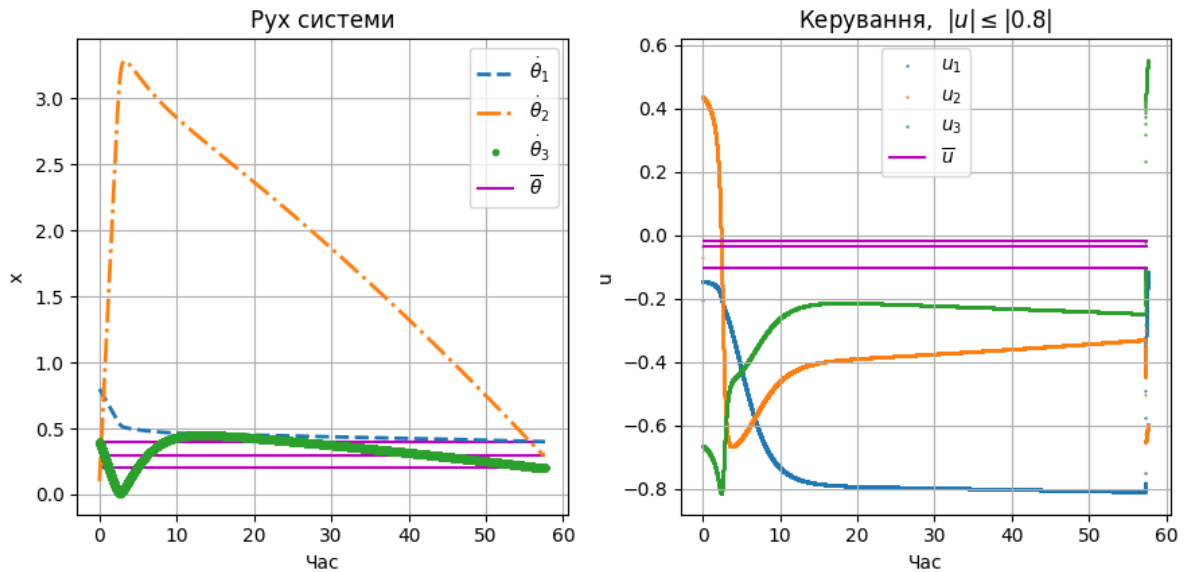


Рис. 4.34: $x_0 = (0, 0, 0, 0.8, 0.1, 0.4)^T$, $\bar{x} = (0, 0, 0, 0.4, 0.3, 0.2)^T$.

Експеримент 2.

Залишимо всі параметри такими ж, але змінимо обмеження на керування:

$$r = 0.6.$$

Проінтегрувавши систему (31) чисельним методом Рунге-Кутти четвертого порядку, отримуємо результати на рис. 4.35. Розв'язок був отриман в точці $T = 83.14$.

Експеримент 3.

Залишимо всі параметри такими ж, як в Експерименті 1, але змінимо початкову і кінцеву точки:

$$x_0 = (0, 0, 0, 0.2, 0.6, 0.5)^T,$$

$$\bar{x} = (0, 0, 0, 0.7, -0.4, 0.6)^T.$$

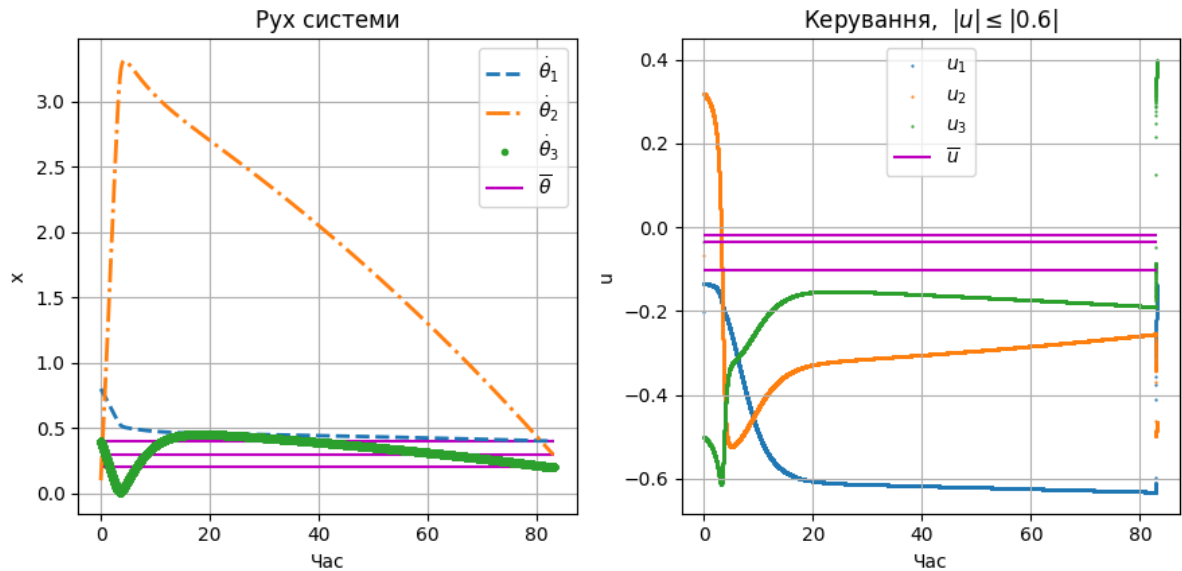


Рис. 4.35: $x_0 = (0, 0, 0, 0.8, 0.1, 0.4)^T$, $\bar{x} = (0, 0, 0, 0.4, 0.3, 0.2)^T$.

Проінтегрувавши систему (31) чисельним методом Рунге-Кутти четвертого порядку, отримуємо результати на рис. 4.36. Розв'язок був отриман в точці $T = 59.196$.

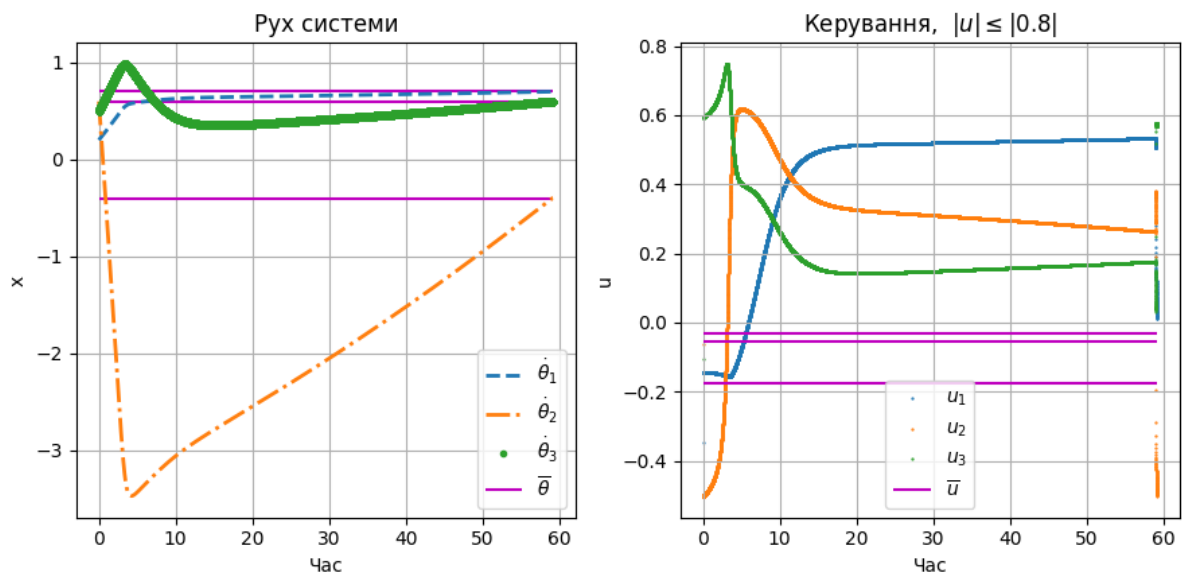


Рис. 4.36: $x_0 = (0, 0, 0, 0.2, 0.6, 0.5)^T$, $\bar{x} = (0, 0, 0, 0.7, -0.4, 0.6)^T$.

Експеримент 4.

Залишимо всі параметри такими ж, як в **Експерименті 1**, але змінимо обмеження:

$$r = 0.9.$$

Але цього разу, розв'яжемо одразу три задачі про перехід із фіксованої точки на наперед задану поверхню, тобто спочатку потрібно перевести систему з x_0 в \bar{x}^1 , потім з \bar{x}^1 в \bar{x}^2 , і завершити в точці \bar{x}^3 . Таким чином можна побудувати будь-яку послідовність руху системи, яка потрібна у дослідженні.

Оберемо послідовність точок таким чином:

$$\begin{aligned} x_0 &= (0, 0, 0, 0.2, 0.6, 0.5)^T, \\ \bar{x}^1 &= (0, 0, 0, 0.7, -0.4, 0.6)^T, \\ \bar{x}^2 &= (0, 0, 0, 0.2, 0.1, 0.8)^T, \\ \bar{x}^3 &= (0, 0, 0, 0.6, 0.2, 0.5)^T. \end{aligned}$$

Таким чином, послідовність \bar{u}^i , $i = 1..3$:

$$\begin{aligned} \bar{u}^1 &= (-0.1739, -0.0323, -0.0533)^T, \\ \bar{u}^2 &= (-0.1727, -0.0764, -0.0711)^T, \\ \bar{u}^3 &= (-0.1752, -0.055, -0.0444)^T. \end{aligned}$$

Результат виконання програми можна побачити на рис. 4.37.

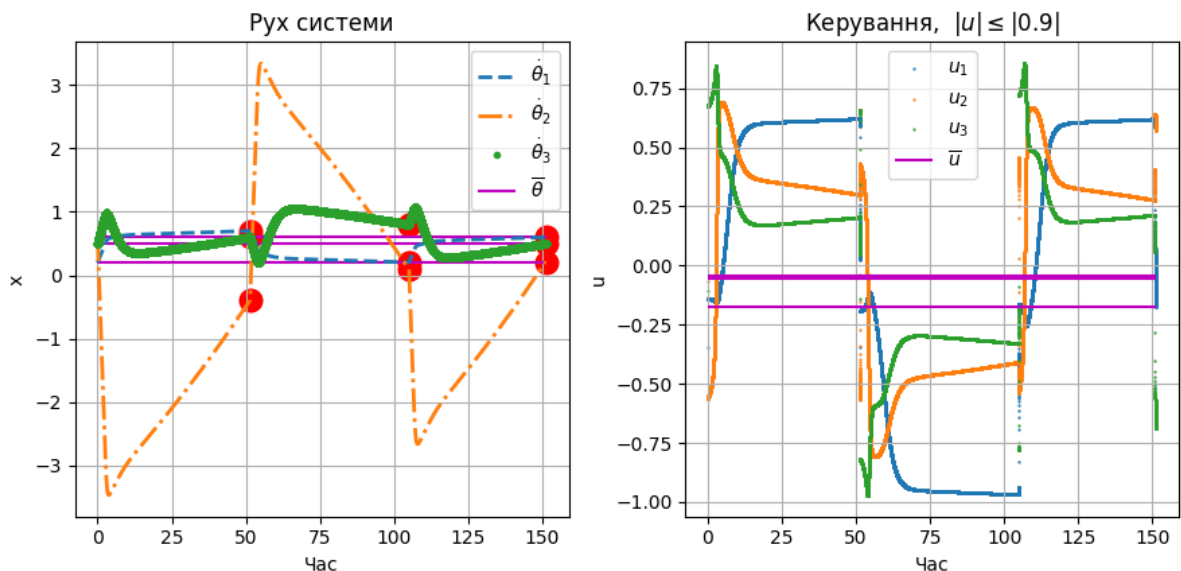


Рис. 4.37: Послідовність точок.

4.4 Опис програмної реалізації

Програму було реалізовано мовою *Python* версії 3.7.2, також було використано сторонні бібліотеки *NumPy* (для обчислення норми векторів, матричних представлень, обчислення добутку та обернених матриць), *SciPy* [10] (для обчислення розв'язку алгебраїчного рівняння Ріккати та частинних похідних) та *Matplotlib* [11] (для представлення графіків результатів виконання). Також використовується пакет *virtualenv* для того, щоб пакети та бібліотеки, використані в програмі, не були пов'язані зі встановленими в системі. В файлі *requirements.txt* можна знайти залежності, які використані в повній мірі.

Також було реалізовано метод Рунге-Кутти четвертого порядку для знаходження наближеного розв'язку задачі Коші.

Вхідною точкою для використання програми є файл *src/main_mag.py*.

Програма розділена на класи, кожен клас відповідає за свою частину реалізації.

Наприклад, існують два інтерфейси (або абстрактні класи):

- *Plotter*, який використовується для виводу на екран розв'язку. В цьому інтерфейсі є тільки один метод - *plot()*, завдяки якому можна якимось чином вивести розв'язок. Реалізована реалізація цього інтерфейсу *PlotterMag*, яка будує графік за допомогою бібліотеки *Matplotlib* [11]. Ця функція будує графік траєкторії системи та графік траєкторії керувань.
- *Solver*, який відповідає за різні типи побудови керувань. Усі реалізації цього класу є залежними від нащадків *Plotter*. Для нащадків цього абстрактного класу потрібно реалізувати метод *solve()*, завдяки якому отримується наступна точка траєкторії системи. В самому класі *Solver* реалізовано метод Рунге-Кутти четвертого порядку для знаходження наближеного розв'язку задачі Коші, а також метод *solve_and_plot()*, який є центральним і визивається в файлі *src/main_mag.py*. В цьому методі будується повна траєкторія системи та визивається метод *plot()* у відповідного *Plotter*'а. Нащадками цього класу є класи *solvers.Demp1rSolver* (*src/solvers.Demp1rSolver*),

та `solvers.DempRVecSolver` (`src/solvers.DempRVecSolver`), які є відповідальними за реалізацію алгоритмів на основі методу демпфування.

Також є допоміжні модулі (`helper.py`, в якому описані допоміжні функції, такі як:

- `gradF` - пошук градієнта функції.
- `part_derivative()` - пошук частинної похідної. [10].

Висновки

В роботі розглянуто методіку конструювання керування для координації багаточарових з'єднань і балансу постави людини у відповідь на невеликі збурення. Стратегія оптимального за критерієм керування була розроблена на основі моделі опорно-рухового апарату нижньої кінцівки людини, запропонованої в [8], з метою стабілізації постави людини.

Основні результати такі:

1. Для розв'язання задачі стабілізації системи, що описує модель руху нижніх кінцівок з реакцією ЦНС, побудовано керування, яке дозволяє розв'язати задачу оптимальної стабілізації для лінійної стаціонарної динамічної системи. Розроблено алгоритм розв'язання задачі.
2. Для системи, що описує модель руху нижніх кінцівок з реакцією ЦНС побудовано керування за допомогою функції Ляпунова та керування, основане на методі демпфування. Для останнього методу було розглянуто два види обмежень на керування. Розроблено алгоритм розв'язання задачі.
3. Для розв'язання задачі про перехід із фіксованої точки на наперед задану поверхню комплексної моделі руху нижніх кінцівок людини було проведено лінеаризацію, побудовано керування на основі методу демпфування. Розглянуто два види обмежень на керування.
4. Розроблено пакет програм на мові Python для комп'ютерного моделювання розроблених алгоритмів. За його допомогою проведено серію обчислювальних експериментів.

Результати роботи можуть бути застосовані для аналізу постави людини, для дослідження руху нижніх кінцівок людини, що може бути корисним для побудови більш досконалих конструкцій протезів і протезованих систем. Розроблені алгоритми можуть бути застосованими для створення автоматизованих систем реабілітації пацієнтів, які мають проблеми ЦНС, зі зниженою м'язовою активністю, для осіб, які піддаються ризику падіння.

Література

- [1] Башняков О. М., Пічкур В. В. *Задача синтезу в теорії керування* : Навчальний посібник. - К.: Вид-во “Сталь”, 2012. - 116 с.
- [2] Гаращенко Ф. Г., Пічкур В. В. *Прикладні задачі теорії стійкості*. – К.: Київський університет, 2014. – 120 с.
- [3] Глазков В., Кулик А., Большаков А. *Построение математической модели движения нижних конечностей человека для исследования протезируемых систем // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика* . 2014. №. 2. С. 117-127.
- [4] Егоров А. И. *Основы теории управления*. – М.: Физматлит, 2004. – 504 с.
- [5] Зубов В. И. *Лекции по теории управления*. - М.: Наука, 1975. - 496 с.
- [6] Кунцевич В. М., Лычак М. М. *Синтез систем автоматического управления с помощью функций Ляпунова*. - М.: Наука, 1977. - 400 с.
- [7] Kato I. *Modeling and Control of the Biped Gait* / I. Kato. Waseda Univ., Токуо, 1970.
- [8] Куо, А. *An optimal control model for analyzing human postural balance*, IEEE Transactions on biomedical engineering, Vol.42, No 1.
- [9] Kuzmich, O., Awrejcewicz, J., Mekush, O., Pekh, P., Bortnik, K., Potomkina, E., Shubala, I. (2018). *Development of control for the ankle joint simulator applied to the problem on vertical posture balance of a human*. Eastern-European Journal of Enterprise Technologies, Vol. 6, No 7 (96), 49–57.
- [10] <https://docs.scipy.org/doc/scipy-0.19.0/reference/generated/>
- [11] <https://matplotlib.org/3.1.0/resources/index.html>