

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»**

ННК“Інститут прикладного системного аналізу”

(повна назва інституту/факультету)

Кафедра Системного проектування

(повна назва кафедри)

«До захисту допущено»

Завідувач кафедри

А.І.Петренко

(підпис)

(ініціали, прізвище)

“ ” \_\_\_\_\_ 2017 р.

**Дипломна робота**

першого (бакалаврського) рівня вищої освіти

(першого (бакалаврського), другого (магістерського))

з напрямку підготовки 6.050101 Комп’ютерні науки

(код і назва)

на тему: Розробка системи керування рухом крокуючого робота засобами  
SimMechanics та Unity

Виконав (-ла): студент (-ка) 4 курсу, групи ДА-31

(шифр групи)

Видолоб Антон Валерійович

(прізвище, ім’я, по батькові)

(підпис)

Керівник

доцент, к.т.н., Чкалов О. В.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант

Економічний доцент, к.е.н., Рощина Н.В.

(назва розділу) (посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Рецендент

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Нормоконтроль

ст. викладач, к.т.н., Бритов О.А.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Засвідчую, що у цій дипломній роботі немає  
запозичень з праць інших авторів без  
відповідних посилань.

Студент \_\_\_\_\_

(підпис)

Київ – 2017 року

**Національний технічний університет України  
«Київський політехнічний інститут  
імені Ігоря Сікорського»**

Факультет (інститут) ННК «Інститут прикладного системного аналізу»  
(повна назва)

Кафедра Системного проектування  
(повна назва)

Рівень вищої освіти Перший (Бакалаврський)  
(перший (бакалаврський), другий (магістерський) або спеціаліста)

Напрямок підготовки 6.050101 Комп'ютерні науки  
(код і назва)

ЗАТВЕРДЖУ

Ю Завідувач

кафедри

А.І.Петренко  
(підпис) (ініціали, прізвище)

«\_\_» \_\_\_\_\_ 2017 р.

**ЗАВДАННЯ  
на дипломний проект студенту**

Видолобу Антону Валерійовичу  
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка системи керування рухом крокуючого робота засобами SimMechanics та Unity

керівник проекту (роботи) Чкалов Олексій Валерійович, к.т.н., доцент,  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «10» травня \_\_\_\_\_ 2017 р. № 1477-с

2. Строк подання студентом проекту (роботи) \_\_\_\_\_

3. Вихідні дані до проекту (роботи)

Крокуючий апарат антропоморфної структури із 7 рухомими ланками пересувається по рівній горизонтальній поверхні. Відносний рух окремих ланок задається програмно.

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити)

1. Дослідження сучасного стану проблеми дотримання рівноваги антропоморфних крокуючих апаратів.
  2. Дослідження механіко-математичної моделі опорно-рухового апарата людин.
  3. Дослідження динаміки крокуючого апарата в середовищі MATLAB - Simulink.
  4. Моделювання руху крокуючого апарата в середовищі Unity.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслеників, плакатів тощо)
- Математична модель крокуючого апарата в середовищі MATLAB Simulink – плакат.
  - Моделювання руху крокуючого апарата в середовищі Unity – плакат.
  - Порівняння ходи крокуючого апарата в середовищах MATLAB Simulink та Unity – плакат.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічна частина	доцент, к.е.н., Рощина Н.В.		

\* Консультантом не може бути зазначено керівника дипломного проекту (роботи).

7. Дата видачі завдання 01.02.2017.

## Календарний план

№ з/п	Назва етапів виконання дипломного проекту (роботи)	Строк виконання	Примітка
1	Отримання завдання	01.02.2017	
2	Планування дипломного проекту	15.02.2017	
3	Дослідження існуючих рішень	28.02.2017	
4	Дослідження механіко-математичної моделі опорно-рухового апарата людин	15.03.2017	
5	Дослідження динаміки крокуючого апарата в середовищі MATLAB - Simulink	15.04.2017	
6	Моделювання руху крокуючого апарата в середовищі Unity	30.04.2017	
7	Аналіз результатів роботи	15.05.2017	
8	Оформлення роботи	30.05.2017	
9	Проходження нормо-контролю	10.06.2017	
10	Підготовка презентації дипломного проекту	15.06.2017	

Студент

\_\_\_\_\_  
(підпис)Видолоб А.В.

(ініціали, прізвище)

Керівник роботи

\_\_\_\_\_  
(підпис)Чкалов О. В.

(ініціали, прізвище)

## АНОТАЦІЯ

бакалаврської дипломної роботи Видолоба Антона Валерійовича на тему  
«Розробка системи керування рухом крокуючого робота засобами  
SimMechanics та Unity»

Дипломна робота присвячена задачі моделювання руху антропоморфного крокуючого апарата (АКА) по рівній горизонтальній поверхні на основі експериментальних даних про ходу людини. В роботі описано динаміку руху АКА за допомогою диференціальних рівнянь Лагранжа другого роду, на підставі експериментальних біомеханічних даних сформулювати ритмічні, кінематичні та динамічні обмеження антропоморфного руху АКА. Розроблено методику обробки експериментальних біомеханічних даних і алгоритми обчислення кінематичних та динамічних характеристик ходи АКА. Розроблено програмне забезпечення для розв'язання сформульованих задач, візуалізації та аналізу руху АКА.

Загальний обсяг роботи: 98 сторінок, 30 ілюстрації, 6 таблиць та 81 джерел.

Ключові слова: двонога хода, антропоморфний крокуючий апарат, робототехнічна система, математична модель, числовий алгоритм.

## АННОТАЦИЯ

бакалаврской бакалаврской дипломной работы Выдолоба Антона Валерьевича на тему «Разработка системы управления движением шагающего робота средствами SimMechanics и Unity»

Дипломная работа посвящена задаче моделирования движения антропоморфного шагающего аппарата (АША) по ровной горизонтальной поверхности на основе экспериментальных данных о походке человека. В работе описано динамику движения АША с помощью дифференциальных уравнений Лагранжа второго рода, на основании экспериментальных биомеханических данных сформулировать ритмичные, кинематические и динамические ограничения антропоморфного движения АША. Разработана методика обработки экспериментальных биомеханических данных и алгоритмы вычисления кинематических и динамических характеристик походки АША. Разработано программное обеспечение для решения сформулированных задач, визуализации и анализа движения АША.

Общий объем работы: 98 страниц, 30 иллюстрации, 6 таблиц и 81 источников.

Ключевые слова: двуногая походка, антропоморфный шагающий аппарат, робототехническая система, математическая модель, числовой алгоритм.

## ABSTRACT

to the bachelor thesis by Vydolob Anton Valeriaiovich on “Development of Walking Robot Motion Control System Using SimMechanics and Unity environments”

The thesis is devoted to the problem of modeling of the anthropomorphic walking machine (AWM) on a flat horizontal surface on the basis of human’s walk experimental data. The work describes the dynamics of AWM using Lagrange differential equations of the second type, based on experimental biomechanical data to formulate a rhythmic, dynamic and kinematic constraints anthropomorphic AWM movement. The method of experimental biomechanical data and algorithms for calculating the kinematic and dynamic characteristics AWM movement is represented. The software for solving of described problem, visualize and analyze the movement of AWM is developed.

Includes 99 pages, 30 figures, 6 tables and 81 sources.

Keywords: bipedal gait, anthropomorphic walking machine, robotic system, mathematical model, numerical algorithm.

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ .....	10
ВСТУП.....	11
1 АНАЛІЗ СУЧАСНОГО СТАНУ ПРОБЛЕМИ ДОТРИМАННЯ РІВНОВАГИ АНТРОПОМОРФНИХ КРОКУЮЧИХ АПАРАТІВ.....	16
1.1 Історичні та соціальні аспекти формування антропоморфної робототехніків.....	16
1.2 Основні напрями розвитку РТС.....	18
1.3 Аналіз сучасного стану антропоморфної робототехніки для соціальної сфери господарювання.....	19
1.4 Аналіз методів та принципів моделювання і побудови антропоморфних апаратів .....	20
1.5 Висновок .....	31
2 МЕХАНІКО-МАТЕМАТИЧНА МОДЕЛЬ ОПОРНО-РУХОВОГО АПАРАТУ ЛЮДИН .....	33
2.1 Рівняння руху.....	33
2.2 Ритмічні, кінематичні та динамічні обмеження на ходу людини ..	37
2.3 Методика обробки експериментальних біомеханічних даних ходи людини.....	42
2.4 Висновок .....	45
3 ДОСЛІДЖЕННЯ ДИНАМІКИ КРОКУЮЧОГО АПАРАТА В СЕРЕДОВИЩІ MATLAB SIMULINK.....	46
3.1 Висновок .....	54
4 МОДЕЛЮВАННЯ РУХУ КРОКУЮЧОГО АПАРАТА В СЕРЕДОВИЩІ UNITY .....	55
4.1 Середовище розробки Unity.....	55



4.2	Створення моделі крокуючого апарата в середовищі розробки Unity.....	56
4.3	Створення опорної поверхні в середовищі розробки Unity .....	61
4.4	Створення законів руху частин крокуючого апарата в середовищі розробки Unity .....	61
4.5	Створення елемента контролю тестування ходи крокуючого апарата в середовищі розробки Unity .....	64
4.6	Порівняння ходи моделей крокуючого апарата виконаних в середовищі розробки MATLAB Simulink та Unity .....	67
4.7	Висновок .....	68
5	ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ .....	70
5.1	Вступ .....	70
5.2	Постановка задачі техніко-економічного аналізу .....	71
5.3	Обґрунтування системи параметрів ПП.....	75
5.4	Аналіз рівня якості варіантів реалізації функцій .....	82
5.5	Економічний аналіз варіантів розробки ПП.....	83
5.6	Вибір кращого варіанта ПП техніко-економічного рівня .....	87
5.7	Висновок .....	88
	ВИСНОВКИ .....	89
	ПЕРЕЛІК ПОСИЛАНЬ .....	91

## ПЕРЕЛІК СКОРОЧЕНЬ

АКА – антропоморфний крокуючий апарат;

АКС – антропоморфна крокуюча система;

ГШ – гомілково-ступневий шарнір;

КШ – колінний шарнір;

ПШ – плесно-фаланговим шарнір;

ОП – опорна поверхня;

ОРА – опорно-руховий апарат ;

РТС – робототехнічні системи;

СУ – система управління;

ТПЦМ – точка проекції центру мас;

ТЦМ – точка центру мас;

ТШ – тазостегновий шарнір.

## ВСТУП

В останні роки велика увага дослідників в області біомеханіки привернута до процесу ходи та його математичного моделювання. Це зумовлено суттєвими перевагами у здійсненні переміщення за допомогою ходи, що виявляється у високій адаптивності до умов навколишнього середовища і меншим негативним впливом на нього, значно вищій прохідності по пересіченій місцевості у порівнянні з колісними чи гусеничними транспортними засобами завдяки дискретній слідовій доріжці. Під ходою розуміємо таке локомоційне переміщення крокуючої системи, при якому принаймні одна з ніг контактує з поверхнею опори. На відміну від багатогоної ходи, яку можна реалізувати за допомогою послідовності статично стійких конфігурацій, двонога хода вимагає значно складніших алгоритмів для реалізації переміщення і забезпечення динамічної рівноваги системи. Особлива увага вчених приділяється дослідженням двонової ходи в рамках моделей антропоморфних крокуючих систем (АКС) — нелінійних механічних систем, якими моделюють опорно-руховий апарат (ОРА) людини, біотехнічні системи “людина-протез”, двоногі людиноподібні роботи.

Хо́да для людини є звичайним, повсякденним видом діяльності. Розум людини у поєднанні з нейром’язовою та скелетною системами утворює одну з найбільш досконалих і високо адаптивних керованих систем. Тому природними є спроби пізнання дослідниками принципів координації цілеспрямованих рухів ОРА людини. Ці знання можуть бути застосовані в медицині, спортивній біомеханіці, робототехніці. Зокрема, для медицини важливими проблемами є розробка протезно-ортопедичних пристроїв для відновлення втрачених рухових здатностей опорно-рухового апарату людини. Для робототехніки — це створення двоногого автономного робота антропоморфного типу для роботи в обмежених середовищах, небезпечних для перебування людини, екзоскелетонів, які розширюють можливості рухової діяльності людини.

Перші спроби дослідників спрямовувались на одержання та узагальнення даних про кінематику руху ОРА людини [1]. Сучасні засоби експериментальних досліджень дозволяють отримати значний об'єм експериментальних даних про ходу людини. Зокрема, можна виміряти лінійні і кутові переміщення ланок, сили взаємодії опорно-рухового апарату людини з поверхнею крокування та електроміографічну інформацію про активність м'язів [2]. Але, даючи відповідь на запитання «як відбуваються цілеспрямовані рухи людини, зокрема хода», експериментальні засоби не дають відповіді на запитання «які моменти сил генеруються м'язами». Відповідь на останнє запитання можна одержати засобами механіко-математичного моделювання.

Опорно-руховий апарат людини з погляду механіки є надзвичайно складною системою, яка має багато ступенів вільності, а переміщення в суглобах мають складні траєкторії рухів. Згідно з медичними даними [5], рух верхніх кінцівок здійснюється 52 парами м'язів, нижніх кінцівок — 62 парами, спини — 112 парами, грудей — 52, тазу — 8, шиї — 16 і голови — 25 парами м'язів. Очевидно, що керувати такою механічною системою, що містить понад 300 ступенів вільності, є досить складною математичною задачею. Тому ОРА людини зображають механічною системою меншої кількості твердих тіл, з'єднаних між собою циліндричними та сферичними шарнірами, а дію м'язів замінюють відповідними керуючими моментами у цих шарнірах.

Висока мобільність, здатність гнучко враховувати непередбачувані зміни умов навколишнього середовища в АКС забезпечується наявністю кінематичної надлишковості. Це означає, що кількість ступеней вільності в АКС є більша, ніж це необхідно для реалізації заданих рухових задач у заданому робочому середовищі. Тому виконання заданої мети переміщення може бути реалізовано різними конфігураціями рухів. Проте кінематична надлишковість приводить до суттєвого ускладнення механічних моделей АКС. Отже, моделювання процесу ходи людини залишається однією з найбільш складних математичних задач.

Моделювання ходи АКС пов'язане з розв'язанням задач динаміки, керуванням рухом та оптимізації для багатовимірних нестационарних нелінійних систем з деревовидною структурою, фазовий стан яких на різних проміжках часу описується різними диференціальними рівняннями. Крім того, на рух таких систем накладається низка фазових і динамічних обмежень, які суттєво ускладнюють розв'язування оптимізаційних задач. Можливість розв'язувати складні задачі динаміки та керування, які виникають під час моделюванні руху АКС з'явилися завдяки швидкому розвитку засобів обчислювальної техніки.

Значний внесок у вивчення проблем кінематики, динаміки та керування рухом антропоморфних крокуючих систем належить, зокрема Альошинському С. Ю., Бернштейну Н. А., Білецькому В. В., Бербюку В. Є., Болотіну Ю. В., Бордюгу Б. А., Лавровському Е. К., Ларіну В. Б., Морейнісу І. Ш., Новожилову І. В., Формальському А. М., Hatze H, McGeer R., Morecky A. Pedotti A., Schiehlen W., Wucobratovich M., Waldron K., Winter D. A., Goswami A., Chessé S, Bessonnet G., Chevallereau C. та багатьом іншим вченим.

Метою роботи є:

- розгляд існуючих та побудова власної математичної моделі антропоморфного крокуючого апарата (АКА);
- розробка методики і числових алгоритмів для розв'язання задачі керування рухом АКА на основі експериментальних біомеханічних досліджень;
- створення програмного забезпечення для проведення числових досліджень та аналізу результатів математичного моделювання і оптимізації законів руху АКА.

Для досягнення поставленої мети потрібно було розв'язати такі задачі:

- за допомогою диференціальних рівнянь Лагранжа другого роду описати динаміку руху АКА, на підставі експериментальних біомеханічних даних сформулювати ритмічні, кінематичні та динамічні обмеження антропоморфного руху АКА;

- розробити методику обробки експериментальних біомеханічних даних (подографії і гоніометрії) і на їх основі алгоритми обчислення кінематичних та динамічних характеристик ходи АКА;
- розробити програмне забезпечення для розв'язання сформульованих задач, візуалізації та аналізу ритмічних, кінематичних, динамічних, енергетичних характеристик результатів математичного моделювання руху АКА.

*Об'єктом* досліджень є антропоморфні крокуючі системи — нелінійні механічні системи, якими моделюють опорно-руховий апарат людини, біо-технічні системи “людина-протез”, двоногі крокуючі роботи з активними та пасивними приводами керування.

*Предметом* досліджень є задачі моделювання та оптимізації процесу ходи і конструктивних параметрів антропоморфних крокуючих систем.

*Методи дослідження.* Математичним апаратом для дослідження руху антропоморфних крокуючих систем вибрано процедуру рівнянь Лагранжа другого роду, методи оптимального керування нелійними механічними системами, концепція обернених задач механіки, методи параметризації з використанням кубічних згладжувальних сплайнів та числові методи нелінійного математичного програмування.

У першому розділі наведено огляд математичних моделей, задач оптимізації руху, числових методів розв'язування задач оптимального керування АКА.

В другому розділі описується плоска нелінійна дев'ятиланкова механічна модель антропоморфної крокуючої системи. Математичним апаратом для дослідження її динаміки вибрано рівняння Лагранжа 2-го роду.

У третьому розділі описано методику автоматизованої побудови та чисельного дослідження моделі АКА в середовищі MATLAB, Simulink із використанням компонентів з бібліотеки SimMechanics.

Вчетвертому розділі представлено методику візуалізації та аналізу ритмічних, кінематичних, динамічних, енергетичних характеристик руху АКА в середовищі Unity.

Отже, задачі моделювання двоногій ходи, з огляду на їх складність і значне практичне значення, залишаються актуальними і на сьогодні. Тема роботи спрямована на формулювання задач моделювання та оптимізації керованих рухів АКА, розробку числово-аналітичних методик розв'язання відповідних задач керування та розробку відповідного програмного забезпечення.

# **1 Аналіз сучасного стану проблеми дотримання рівноваги антропоморфних крокуючих апаратів**

У розділі проаналізовано та показано етапи розвитку робототехніки соціального спрямування. Також висвітлено особливу роль антропоморфної робототехніки для вирішення низки задач господарювання людини. Значну частину розділу присвячено аналізу базових та сучасних підходів до моделювання антропоморфних крокуючих апаратів, вирішенню задачі управління усталеним рухом апаратів у їх статичних фазах на опорній поверхні.

## **1.1 Історичні та соціальні аспекти формування антропоморфної робототехніки**

Упродовж розвитку цивілізації людина прагне до нагромадження достатку та підвищення показників якості життя. У результаті, з зростанням соціальних норм, невпинно ростуть потреби підвищення кількісних та якісних характеристик товарів споживання, що, в свою чергу, потребує покращення інструментів видобутку сировини та виготовлення продукції [72]. Одним із методів підвищення обсягів нагромадження достатку залишається впровадження експлуатаційної і найманої людської праці [54, 72]. Так, базовими властивостями робітників, у цьому випадку, була здатність їх виконувати певну роботу під чинниками управління, що могли надходити до окремих працівників та їхніх груп. Такі властивості централізованого управління дали змогу виконувати складні задачі. Можливість централізовано управляти робітниками за допомогою доступних засобів впливу встановила явну перевагу застосування людської праці у порівнянні з будь-якими іншими її видами.



Використання методів підвищення особистого достатку, що базувалися на експлуатації людської праці, залишалися актуальними до часу перевищення граничної межі їхньої рентабельності [47, 54, 72].

Для вирішення проблеми нерентабельності людської праці в умовах примусової чи найманої форми впливу було застосовано методи підвищення якісних характеристик виробів та зменшення затратності виробництва. В основу цих методів було покладено постійну модернізацію інструментального забезпечення робітників.

Заміна людської праці засобами автоматичного управління уможливила мінімізацію затрат на видобування ресурсів та виробництво споживчих товарів. Проте автоматизація певних операцій потребувала застосування складних математичних обчислень і призвела до підвищення вартості відповідних інструментальних засобів, що виконували ці обчислення. Окрім цього, механічні автоматизовані інструменти замінювали лише заздалегідь задані технологічні операції. Для виконання не заданих, а лише подібних операції, стало необхідним перелагодження або заміна інструмента. Для вирішення цих проблем залучалися комп'ютеризовані системи та гнучкі методи автоматизації [11, 18, 19, 22, 32, 33, 35, 75].

Упровадження комп'ютеризованих систем та гнучких методів автоматизації дало змогу застосовувати засоби цифрової схемотехніки та концепцію програмного управління для вирішення широкого спектру задач господарювання людини. Зокрема, у разі автоматичного виконання технічних операцій, які передбачають функціональну взаємодію інструмента з об'єктом в умовах впливу навколишнього середовища (може включати вплив людини), впроваджувалися робототехнічні системи (РТС) [15, 18, 19, 22, 31]. Важливою відмінністю РТС, у порівнянні з іншими інструментальними засобами, є застосування математично-алгоритмічного програмного забезпечення, що за допомогою електронно-обчислювальних машин може взаємодіяти з об'єктом, середовищем експлуатації, людиною та класифікувати власний технічний стан

[40]. Тож, з названих причин, РТС набули широкого спектру застосування практично у всіх областях господарювання людини.

## 1.2 Основні напрями розвитку РТС

Економічна доцільність впровадження та простота розробки дали поштовх для значного розповсюдження промислових РТС. Такі РТС поділяються на спеціалізовані та універсальні, мобільні та стаціонарні. У залежності від поставлених задач, названі системи додатково характеризуються рівнем гнучкості конструкції і системи управління (СУ), що обумовлює придатність до переналаштування, підвищення адаптивності тощо.

Основний науковий внесок у розвиток сучасної промислової робототехніки зробили такі вчені, як: А. Ф. Верещагін, М. Вукобратович, В. Я. Копп, К. С. Лі, Г. В. Орловський, М. М. Поліщук, Р. Пол, О. А. Стенін М. М. Ткач, В. Н. Томашевський, М. Б. Ігнат'єв, К. С. Фу, Р. С. Хантенберг, Л. С. Ямпольський.

Паралельно з розвитком промислової робототехніки, широкого розповсюдження набула тематика розробки і впровадження автономної та автоматизованої транспортної робототехніки [39, 49]. Окрім промислового або транспортного використання, існує інша низка областей застосування РТС. Кожна з них має свій рівень розвитку та впровадження робототехніки. Проте на сьогодні існують галузі господарювання, що лише починають провадження таких засобів, зокрема: медицина, сервіс, торгівля, вантажні перевезення тощо [61, 65]. Впровадження РТС у ці галузі дасть змогу істотно підвищити рівень соціального розвитку суспільства. Значну кількість задач, які визначають рівень соціального розвитку, складають задачі, що пов'язані з медичними проблемами та сервісним обслуговуванням людини [29]. Однак, впровадження засобів РТС для автоматизації вирішення проблем соціального спрямування характеризується низькою економічною доцільністю, у порівнянні з промисловою чи транспортною робототехнікою [49]. Проте, їх застосування

спроможне вивільнити великий людський ресурс, який здатний виконувати більш затратну інтелектуальну роботу.

Відзначимо, що за останні десятиліття розвиток математичних і апаратно-програмних засобів розпізнавання та синтезу мови досяг якісно нового рівня [44]. Паралельно з цим розвинулися і методи обробки зображення [1, 23] та відеосигналу [34], тактильної інформації [21] та інших сенсорних засобів [21, 58]. Такий розвиток стимулював прискорення інтеграції РТС у соціальну сферу. Також позитивно на розвиток РТС цього напрямку вплинуло якісне зростання продуктивності, енергоефективності та мініатюрності мікро-ЕОМ [10, 16, 38] та іншої електроніки, що застосовується для управління технічними об'єктами.

### **1.3 Аналіз сучасного стану антропоморфної робототехніки для соціальної сфери господарювання**

Перспективною галуззю РТС для застосування у соціальній сфері господарювання залишається мобільна робототехніка [4, 6, 9, 14, 24]. Проте інтенсивність її впровадження знаходиться на низькому рівні [55]. Існуючі розробки в цьому напрямку базуються на принципах гусеничного або колісного переміщення [55]. Антропоморфні ж системи розвинені значно гірше [5, 6, 17, 21, 26, 52, 55]. Так, найуспішнішими системами соціального спрямування, що рухаються за колісним або гусеничним принципами, є такі: *Rollin' Justin* (рис. 1.1 а) створена під керівництвом К. Бурста (C. Borst) і К. Уотта (C. Ott) [69, 70], *KATE rev.1* (рис. 1.1 б) створена лабораторією *FutureBots Labs'* [76], *Meet Pepper* (Рис. 1.1 с) створена фірмами *Softbank i Aldebaran Robotics* [37]. Основний науково-експериментальний внесок у сучасні системи соціальної робототехніки на колісному чи гусеничному принципах руху зробили такі автори, як: Р. Волпе (*Richard Volpe*) [77, 78], А. Ансаром (*Adnan Ansar*) [51], К. Ассадом (*Christopher Assad*) [48], М. Беналеже (*Mehdi Benallegue*) [43]. Зокрема, робота “*Contribution of Actuated Head and Trunk to Passive Walkers Stabilization*” [43], яка подана групою авторів: М.Беналеже, Ж.П. Ламондом (*Jean-Paul Laumond*) і А.Бертозом

(*Alain Berthoz*), орієнтована на побудову руху робота-рикші. У роботі А. Бустоса (A. Bustos) [46] визначена актуальність інтеграції засобів навігації, маніпуляції, динаміки, антропоморфності, ергономіки, енергоефективності тощо в єдину систему для забезпечення високого рівня їх взаємодії.

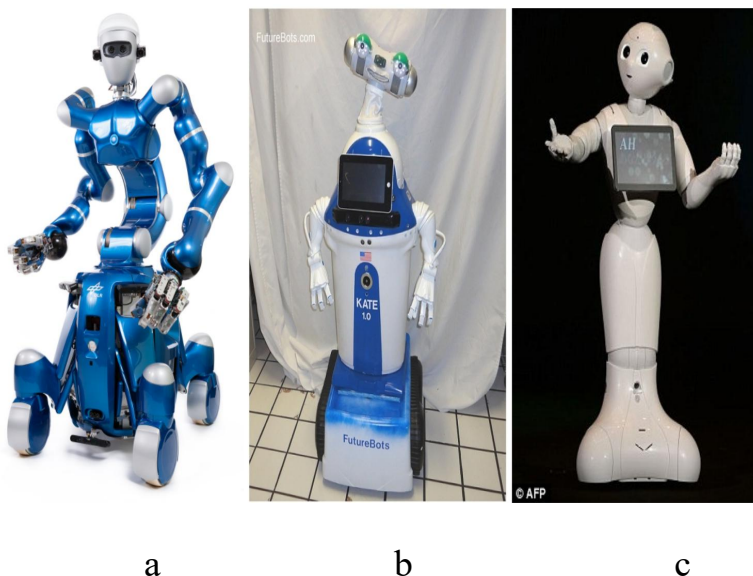


Рисунок 1.1 – Гусеничні та колісні систем соціальної робототехніки: а – *Rollin' Justin*; б – *KATE rev.1*; в – *Meet Pepper*

#### 1.4 Аналіз методів та принципів моделювання і побудови антропоморфних апаратів

Починаючи з публікації Е. Мейбриджа (*Eadweard Muybridge*) у 1901 році статті “*Human Figure in Motion*” [66] не припинялися спроби побудови крокуючих систем. Серед найуспішніших розробок слід відмітити такі проекти: *DLR BiPed* (рис. 1.2 а), який побудований під керівництвом К. Уотта (*Christian Ott*), М.А. Роа (*M.A. Roa*) та Г. Гірзінгера (*Gerd Hirzinger*); *Atlas* (рис. 1.2 б), який розроблений компанією *Boston Dynamics*; *ASIMO* (*Advanced Step in Innovative Mobility*) (рис. 1.2 в) – авторство компанії *Honda Motor Company*; *REEM-C* (Рис. 1.2 д), який побудований компанією *PAL robotics*; *HRP-4* (“*Promet*”) (Рис. 1.2 е), який розроблений спільними силами фінансово-

економічної палати Японії (*METI*), організацією з розвитку нових енергетичних і промислових технологій (*NEDO*) та організацією *Kawada Industries, Inc; Surena* (Рис. 1.2 f), який розроблений *Iran's Tehran University*.

Проте, незважаючи на значний прогрес у побудові АКА, не існує завершеного виробу, здатного забезпечити сучасні потреби [68, 69, 71, 81].

Варто відзначити, що в результаті робіт, проведених такими авторами, як: Б. А. Бордюг [8], В. Б. Ларін [9], А. Г. Тімошенко [8], В. В. Величенко [12], Д. Є. Охоцимский [28], Н. А. Бернштейн [15], В. В. Белецький [3, 4], Д. Вольгер (*D. Wollher*) [80], Копман (*Koopman*) [141], А. Франк (*A. Frank*) [56], Д. Гілл (*J. Hill*) [64], Р. Мк-Гі (*R. McGee*) [62, 63], Дж. Фурушо (*J. Furesho*) [125], М. Вукобратович (*M. Vucobratovic*) [14], К. Уотт (*Ch. Ott*) [69, 70], М.А. Роа (*M. A. Roa*), Г. Гірзінгер *G. Hirzinger* [42], загальну задачу побудови роботизованого крокуючого апарата з антропоморфним принципом руху було розмежовано на ряд підзадач, вирішення яких дало змогу, в перспективі, успішно розв'язати початкову задачу з побудови АКА з динамікою, властивою людському організму.

Так, у публікації [69] основну увагу приділено побудові динамічної моделі антропоморфного крокуючого апарата. Зокрема, автором виділено три основні етапи:

- побудова статичної та динамічної моделей крокуючого апарата;
- побудова математичних принципів дотримання усталеного положення та алгоритмів руху апарата;
- енергетичний аналіз і реалізація штучної ходи.

Натомість, у публікаціях [2, 8] вже розмежовано етапи моделювання самого крокуючого апарата та управління його усталеним рухом:

- моделювання апарата і середовища;
- управління дотриманням усталеного положення апарата на опорній поверхні (ОП);

- управління усталеним рухом апарата на ОП.

Під час виконання етапу “моделювання” робиться опис АКА, зовнішнього середовища та ОП. Також на цьому етапі задаються характеристики інших зовнішніх впливів на апарат. Модель АКА характеризується типом і методом його подання [2, 8].



a



b



c



d



e



f

Рисунок 1.2 – Найуспішніші крокуючі РТС соціальної сфери:  
 a – DLR BiPed; b – Atlas; c – ASIMO; d – REEM-C; e – HRP-4  
 ; f – Surena

Виконання етапу “управління усталеним рухом АКА на ОП” передбачає побудову та опис методів управління усталеним рухом АКА, що базуються на критеріях дотримання стійкості та обраній стратегії управління апаратом [13, 14, 35, 45].

Крім того, на етапі “управління усталеним рухом апарата по ОП” необхідно формалізувати метод управління рухом АКА, який залежить від його динамічної моделі, моделі середовища, структури сенсорної системи та можливостей виконавчої системи, що у комплексі з траєкторією переміщення точки центру мас АКА формують закон руху [7, 20, 25, 36, 59, 79]. Реалізація цього етапу, в свою чергу потребує попереднього вирішення таких задач:

1. Моніторинг і аналіз інформації про взаємодію апарата з навколишнім середовищем;
2. Моніторинг поточного стану та просторового положення апарата;
3. Зчитування траєкторії руху ТЦМ апарата;
4. Розв'язання задачі побудови руху як циклу кроків АКА;
5. Розв'язання задачі класифікації циклічної зміни фаз руху апарата;
6. Визначення задач управління усталеним рухом АКА та зміною фаз його руху;
7. Вирішення зворотної кінематичної задачі управління ланками апарата в межах задачі зміни фази руху.

В результаті вирішення наведених вище задач, стає можливим вирішення і базової задачі управління рухом АКА.

#### **1.4.1 Аналіз базових підходів до моделювання кінематичних зв'язків і динаміки антропоморфних крокуючих апаратів**

Основний внесок в моделювання АКА внесли такі вчені: Н. А. Бернштейн [6], В. В. Веліченко [12], К. Чов (*C. Chow*), Е. Чеморі (*A. Chemori*), Ю. Фуджімото (*Y. Fujimoto*) [57], Е. Френк (*A. Frank*) [56], А. Шеморі (*A. Chemori*) [50], Д. Якобсон (*D. Jacobson*) [53], С. Крут (*S. Krut*) [50], Е. Мейбрідж (*E.*



*Muybridge*) [66], Т. Мак-Гір (Т. McGeer) [62, 63], Р. Томовіч (*R. Tomovic*) [74], М. Вукобратович (*M. Vukobratovic*) [14], К. Нішівакі (К. Nishiwaki) [67] та інші.

Так, у роботах [2, 66] авторами виконані перші наукові спостереження за рухом тварин та людини. Зроблено кінограми та ескізні роботи щодо моторики і ритміки руху, визначено та класифіковано фази людської ходи (рис. 1.3) і вперше застосовано метод кінограми руху. На основі даних, отриманих за допомогою кінограми (рис. 1.3), стало можливим визначити пропорції і положення ланок, діапазони кутів у шарнірах, кінематику (рис. 1.4) та ритміку руху АКА.

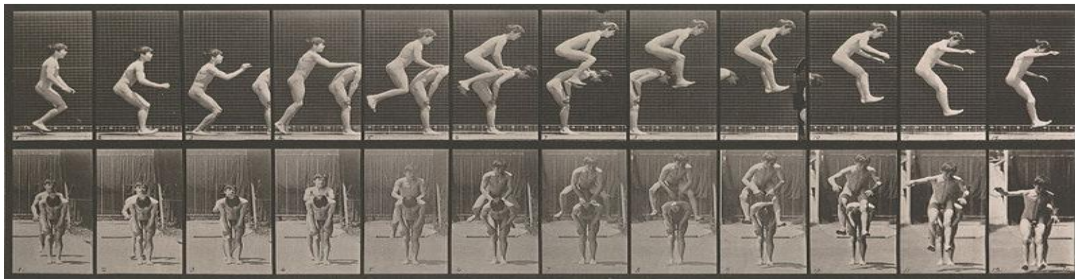


Рисунок 1.3 – Циклічність фаз переміщення срибком

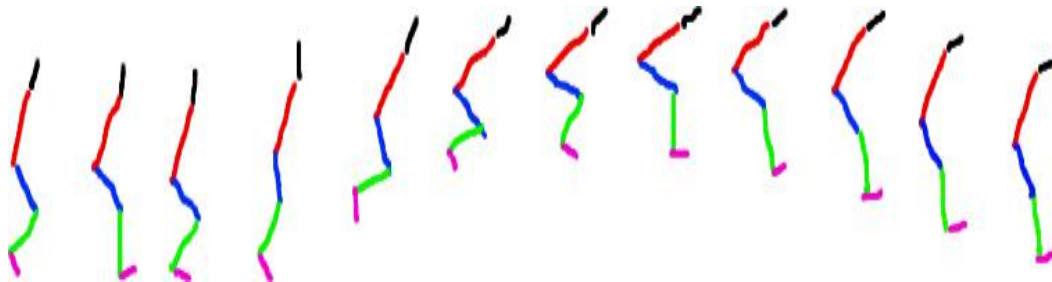


Рисунок 1.4 – Двовірна кінематична модель АКА, отримана за допомогою кінограми Е Muybridge

У роботах Т. Мак-Гіра [62, 63] виконано побудову моделі двовимірного крокуючого апарата з використанням математичних засобів його синтезу. Зокрема, для опису кінематичної схеми і динаміки руху автор застосував методи теоретичної механіки та елементи теорії стійкості, а для формування рівняння руху – теорію про кінцеві стани [2, 62]. Кінематична структура моделі крокуючого апарата (рис. 1.5) налічує три ланки, які формують дві одноланкові ноги та корпус. Взаємний рух ланок даної моделі забезпечується одноосевим шарніром. Кутове положення ланок визначається параметрами  $-\Delta\Theta_f$  і  $-\Delta\Theta_c$ . Маса ланок задаються за допомогою параметрів  $m_h, m_c, m_f$  – відповідно для корпусу, випереджаючої та відстаючої ноги. Довжини ланок задані параметрами  $l, c$ , де  $c$  – задає відстань від точки дотику ОП до ТЦМ ланки. Особливістю розглянутої моделі є будова стопи апарата. Так, область дотичних точок ніг апарата до ОП сформована сектором кола з радіусом  $R$ . Таке подання дає змогу описувати кутове положення апарата методом напрямних косинусів.

У роботі автор також демонструє динаміку моделі апарата. Так, рух апарата відбувається за принципом почергового переносу однієї з ніг у нове положення, подібно до оберненого маятника. Врахування моментів при зіткненні ніг з ОП описується так:

$$H^b = (\cos 2\alpha_0 + r_{\text{гир}}^2) ml^2 \Omega,$$

$$H^a = (1 + r_{\text{гир}}^2) ml^2 \Omega,$$

де  $H^b, H^a$  – моменти до зіткнення та після зіткнення відповідно;

$\alpha_0$  – кут який описується як  $\Theta = -\alpha_0$  у разі зупинки апарата і змінюється в діапазоні  $[-\Delta\Theta() \dots \Delta\Theta(c)]$ .

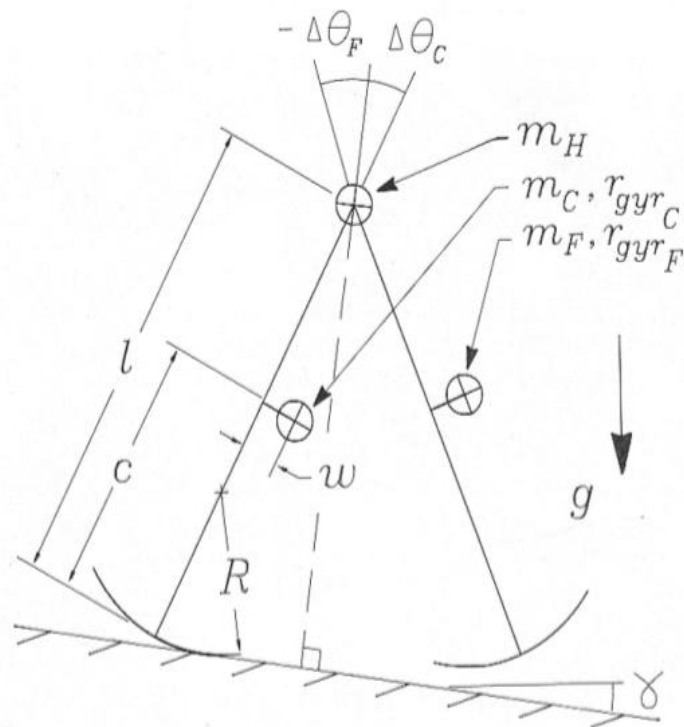


Рисунок 1.5 – Кінематична схема крокуючого апарата з радіальними опорами, задана в сагітальній площині:  $-\Delta\theta_f, \Delta\theta_c$  – діапазон відхилення ніг апарата;  $\omega$  – відхилення проекції ТПЦМ;  $\varphi$  – максимальний кут нахилу опорної поверхні;  $g$  – спрямування вектора сили тяжіння;  $c, l$  – довжини ланок апарата;  $m_H, m_c, m_f$  – маси апарата і його ланок;  $r_{gyr\ c}, r_{gyr\ f}$  – сили, які виникають у наслідок взаємодії апарата з ОП;  $\gamma$  – кут нахилу ОП

Під час переносу ноги точка проекції центра мас апарата переміщується від дотичних точок задньої ноги до дотичних точок передньої ноги. Таким чином, після виходу ТПЦМ за межі стопи апарата, він починає завалюватися. Фізична реалізація апарата зображена на рис. 1.6.

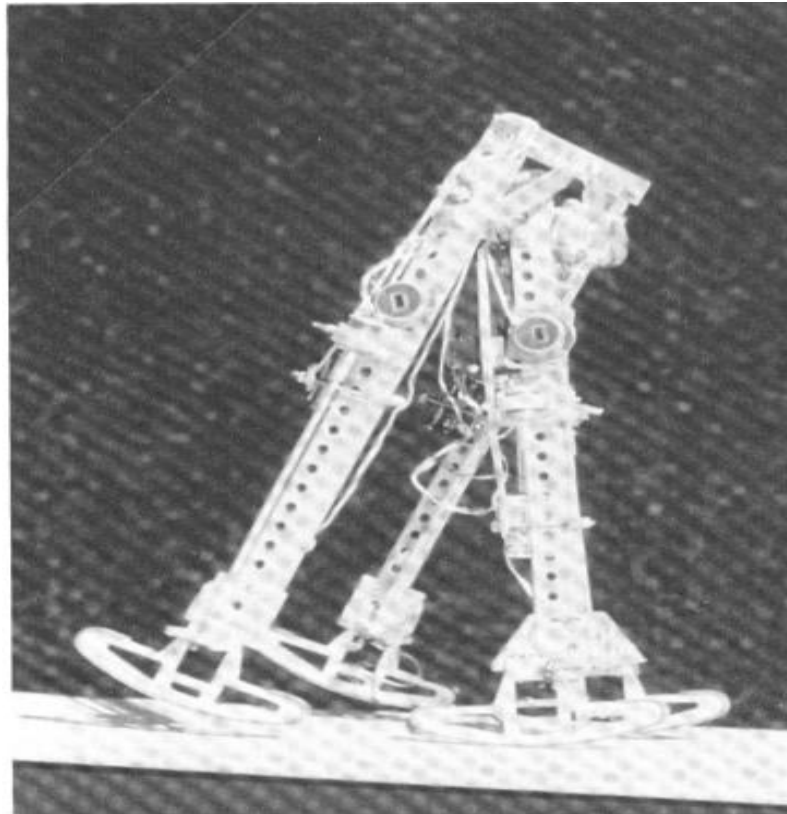


Рисунок 1.6 – Крокуючий апарат з перекочувальним принципом руху

У публікації “*Contribution of Actuated Head and Trunk to Passive Walkers Stabilization*” [43] авторів М. Беналеге (*M. Benallegue*), Дж.-П. Ламонда (*J.P. Laumond*) і Е. Бертоза (*A. Berthoz*) показано розробку двовимірної моделі антропоморфного крокуючого апарата для задач пасивної ходи. Модель налічує дві одноланкові ноги та дволанковий корпус. Кожна з ланок моделі характеризується довжиною  $l$  та масою  $m$ , відповідно.

Динаміка моделі описується вектором  $x$ , який має вигляд

$$x = [\Theta, \varphi, \alpha, \beta, \dot{\theta}, \dot{\varphi}, \dot{\alpha}, \dot{\beta}]^t,$$

де параметр  $t$  визначається відносно моменту зіткнення ланки апарата з опорною поверхнею.

Характер зіткнення не є пружним, тому управління моделлю може виконуватися методом кінцевих станів. Динаміка моделі простежується в інтервалі між зіткненнями ніг з ОП і формалізується у вигляді рівняння

$$\tau_k = [\tau_t, k, \tau_k]^t.$$

Зміна параметрів вектора стану описується так:

$$x_{k+1} = f(x_k, \tau_k),$$

де  $f$  — функція, що набуває приросту параметрів вектора  $x$ , у проміжку часу між  $\tau_{k-1}$  і  $\tau_k$ .

Для моделювання динамічних процесів управління рівновагою АКА авторами застосовано принцип теореми Ляпунова, що в значній мірі обмежує структуру кінематичної схеми апарата.

Натомість, у публікації “*A Control Architecture With Stabilizer For 3D Stable Dynamic Walking of SHERPA Biped Robot on Compliant Ground*” [50] авторами Е. Чеморі (A. Chemori), С.Ле-Флоком (S. Le Flock), С. Крутом (S. Krut) і Е. Дембре (E. Dombre) показано побудову і застосування кінематичної моделі антропоморфного крокуючого апарата, що налічує 12 ланок. До того ж, всі ланки переміщуються з врахуванням наявної опорної поверхні і мають 18 ступенів свободи. Для управління моделлю значення кутів задаються у вигляді вектора управління  $X \in R^{18}$ . Тоді, як сама модель описується вектором  $X_{tags} = f(q); q \in R^{18}; X_{tags} \in R^{45}$ , а її зворотня модель (простір станів) подається так:  $q = f^{-1}(X_{tags}); q \in R^{18}; X_{tags} \in R^{45}$ . Графічна інтерпретація кінематичних зв'язків запропонованої моделі АКА зображена на рис. 1.7.

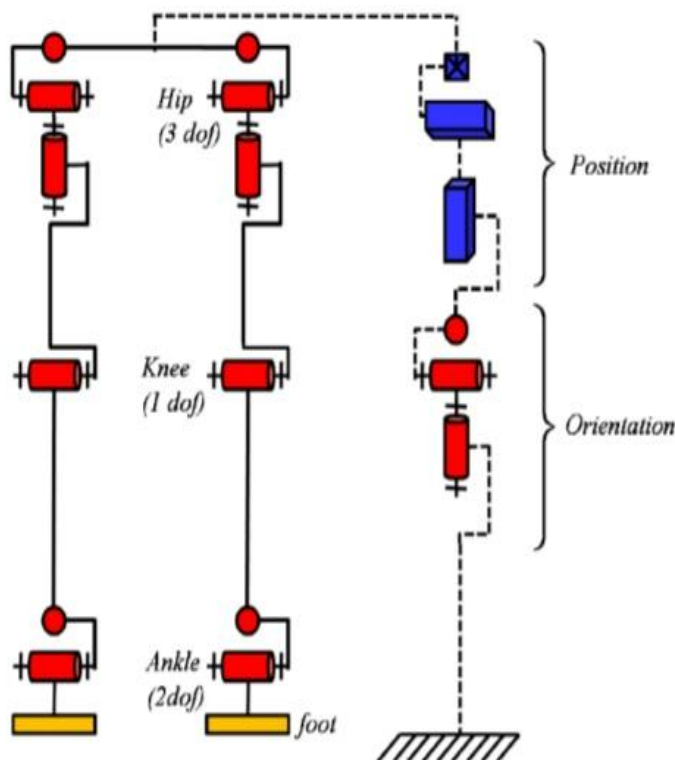


Рисунок 1.7 – Кінематична схема тривимірного АКА з 18 ступенями свободи: *Hip*, *Knee*, *Ankle* – шарніри апарата; *dof* – кількість ступенів свободи у шарнірі; *Position* – блок схеми, що відповідає за переміщення апарата; *Orientation* – блок схеми, що відповідає за розворот апарата

Зокрема, з рис. 1.7 видно, що кутове переміщення у тазобедерному шарнірі задається трьома ступенями свободи, в гомілкобедровому шарнірі – одним ступенем свободи, тоді як у гомілкоstopовому шарнірі – двома ступенями свободи.

Особливістю розглянутої моделі є наявна група шарнірів *Position*, *Orientation* (рис. 1.7), які задають кутове та лінійне положення моделі АКА відносно опорної поверхні. На основі запропонованої кінематичної моделі авторами побудовано динамічну модель апарата, що налічує 18 ступенів свободи. Динаміка моделі задана рівняннями Лагранжа [27, 30, 41, 60, 73] у такому вигляді:

$$M(q)q'' + C(q, q')q' + G(q) = Su$$

де  $q = [q_1, \dots, q_{12}, \omega_x, \omega_y, \omega_z, T_x, T_y, T_z]^T \in R^{18}$  – вектор узагальнених координат;

$M(q) \in R^{18 \times 18}$  – матриця інерції ланок;  $C(q, \dot{q}) \in R^{18 \times 18}$  – матриця Коріолісових та доцентрових прискорень;

$G(q) \in R^{18}$  – вектор сили тяжіння;  $u \in R^{12}$  – вектор вхідних управляючих впливів;

$S \in R^{18 \times 12}$  – матриця розподілу обертових моментів шарнірів. У випадку дотику ланки (ланок) до ОП, рівняння динаміки моделі АКА задається рівнянням:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = Su + J(q)^T \lambda,$$

де  $\lambda$  і  $J(q)$  – відповідно, Лагранжеві множники та матриця Якобі.

## 1.5 Висновок

Підводячи підсумки можна визначити, що наведені в роботах методи подання моделей АКА мають наступні особливості:

1. Поліморфність. Модель придатна для зображення поведінки апарата при одноопорній та двохопорній фазах руху;
2. Спостережуваність. Модель одночасно може бути застосована для моніторингу і управління положенням крокуючого апарата.

У всіх розглянутих вище моделях крокуючі апарати подано з врахуванням чотирьох основних складових:

- структури кінематичних ланцюгів;
- динамічної поведінки апарата;
- взаємодії апарата з опорною поверхнею;
- взаємодії з сенсорною системою та системою управління.

Їх кінематичні моделі поділяються за структурою на:

- оглядові та концептуальні моделі апаратів;
- двохопорні з птахоподібним типом згину кінематичних пар;
- моделі антропоморфних апаратів;
- моделі апаратів, що налічують більше двох опор.

На відміну від кінематичних моделей, динамічні моделі враховують сили, що спричиняють переміщення апарата і його рух відносно нерухомої точки простору, та рух ланок апарата одна відносно одної. Проаналізовані динамічні моделі крокуючих апаратів базуються на застосуванні динаміки твердих тіл або комп'ютерній комбінаторній динаміці.



## 2 Механіко-математична модель опорно-рухового апарату людин

### 2.1 Рівняння руху

Опорно-руховий апарат людини моделюється плоскою системою з дев'яти твердих тіл, з'єднаних ідеальними циліндричними шарнірами (рис. 2.1). Система складається із вагомego інерційного корпусу (тіло  $NG$ ) і двох чотирьохланкових ніг. Тіла, які моделюють стегна ( $NK_i$ ) і гомілки ( $K_iA_i$ ) є вагомими та інерційними. Безінерційні тіла  $H_iA_iM_i$ ,  $M_iS_i$  моделюють стопу  $i$ -ої ноги,  $i = 1, 2$ . Шарнір між корпусом і стегном будемо називати тазостегновим шарніром (ТШ), стегном і гомілкою — колінним (КШ), гомілкою і ступнею — гомілково-ступневим (ГШ), ступнею і фалангами — плесно-фаланговим шарніром (ПШ).

Обмежимося моделюванням динаміки руху людини в площині  $OXY$  інерційної системи координат  $OXYZ$ . Вважаємо, що керований рух системи відбувається внаслідок взаємодії сили тяжіння, керуючих моментів сил  $q_i$ ,  $u_i$ ,  $p_i$ ,  $w_i$ ,  $i = 1, 2$ , що діють відповідно у ТШ, КШ, ГШ, ПШ ніг, та сил реакції опорної поверхні.

Плоску систему сил реакції опори, що діють на ступню  $i$ -ої ноги замінимо рівнодієюною силою  $R_i$ ,  $i = 1, 2$ .

Конфігурацію ОРА людини в площині  $OXY$  визначаємо вектором узагальнених координат  $\xi = (x, y, \psi, \alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \varepsilon_1, \varepsilon_2)$ , де  $x, y$  — декартові координати ТШ,  $\psi$ ,  $\alpha_i$ ,  $\beta_i$ ,  $\gamma_i$ ,  $\varepsilon_i$  — кути відхилення корпусу, стегна, гомілки, передньої частини ступні і фаланг  $i$ -ї ноги від вертикалі (рис. 2.1).

Введемо позначення:

$m$  — маса корпусу;

$r$  — віддаль від ТШ до центру мас корпусу;

$J$  — момент інерції корпусу відносно осі  $OZ$  у точці ТШ;

$m_{ai}$  – маса  $i$ -го стегна;

$a_i$  – довжина  $i$ -го стегна;

$r_{ai}$  – віддаль від ТШ до центру мас  $i$ -го стегна;

$J_{ai}$  – момент інерції  $i$ -го стегна відносно осі  $OZ$  у точці ТШ;

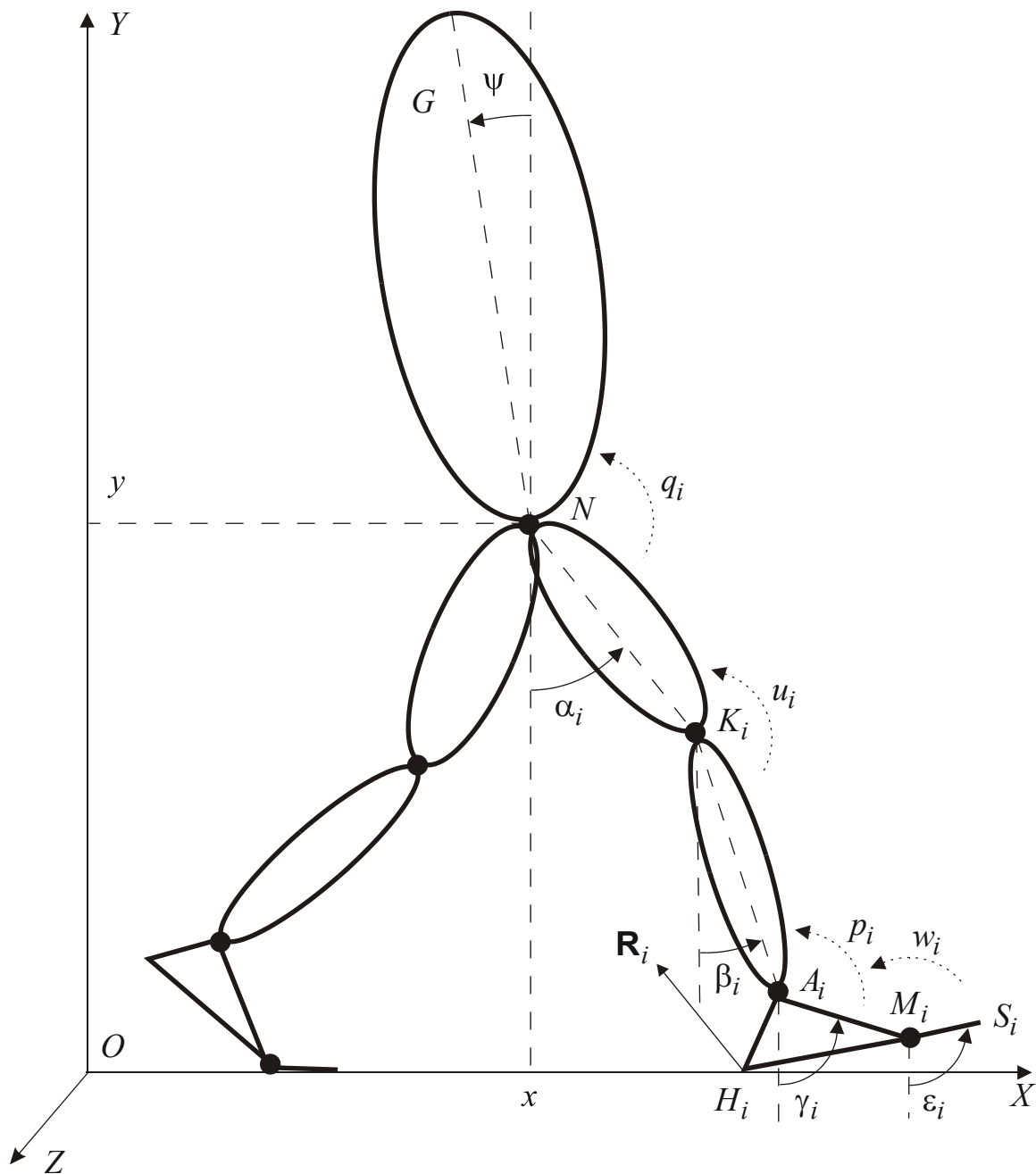


Рис. 2.1 - Механічна модель опорно-рухового апарату людини.

$m_{bi}$  – маса  $i$ -ої гомілки;

$b_i$  – довжина  $i$ -ої гомілки;

$r_{bi}$  – віддаль від КШ до центру мас  $i$ -ої гомілки;

$J_{bi}$  – момент інерції  $i$ -ої гомілки відносно осі  $OZ$  у точці КШ;

$m_{fi}$  – маса  $i$ -ої ступні, яка зосереджена в точці  $A_i$ ;

$g$  – прискорення сили тяжіння;

$l_{1i} = |A_i H_i|$ ;  $l_{2i} = |A_i M_i|$ ;  $l_{3i} = |M_i S_i|$ ;  $l_{-i} = |H_i M_i|$ ;  $\varphi_{Ai} = \angle H_i A_i M_i$ ;  
 $\varphi_{Mi} = \angle A_i M_i H_i$ ;  $(x_i, y_i), (x_{hi}, y_{hi}), (x_{mi}, y_{mi}), (x_{si}, y_{si})$ , – відповідно декартові координати точок  $A_i, H_i, M_i, S_i$  ступні  $i$ -ї ноги,  $i = 1, 2$ , в інерційній системі відліку  $OXY$  (рис 2.1).

Математичним апаратом для дослідження ходи виберемо рівняння Лагранжа другого роду [17]:

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{\xi}_j} \right) - \frac{\partial T}{\partial \xi_j} = - \frac{\partial \Pi}{\partial \xi_j} + Q_j, j = 1, 2, \dots, 11,$$

де

$T, \Pi$  – кінетична та потенціальна енергії системи;

$Q_j$  – узагальнені сили.

Записавши вирази для кінетичної та потенціальної енергії у випадку досліджуваної 9-ти ланкової системи після відповідного диференціювання отримуємо рівняння руху:

$$f_1(t) = R_{1x}(t) + R_{2x}(t), \quad (2.1)$$

$$f_2(t) = R_{1y}(t) + R_{2y}(t), \quad (2.2)$$

$$f_{3i}(t) = q_i - u_i + a_i (R_{1x} \cos \alpha_i + R_{1y} \sin \alpha_i), \quad i = 1, 2; \quad (2.3)$$

$$f_{4i}(t) = u_i - p_i + b_i (R_{1x} \cos \beta_i + R_{1y} \sin \beta_i), \quad i = 1, 2; \quad (2.4)$$

$$f_5(t) = -q_1 - q_2; \quad (2.5)$$

$$p_i = (x_i - x_{Ri})R_{iy} - y_i R_{ix}; \quad \omega_i = 0, \quad \text{при } x_{Ri} \in [x_{hi}^0, x_{mi}], \quad i = 1, 2, \quad (2.6)$$

$$p_i = w_i + (x_i - x_{mi})R_{iy} + (y_{mi} - y_i)R_{ix},$$

$$w_i = (x_{mi} - x_{Ri})R_{iy} - y_{mi} R_{ix} \quad \text{при } x_{Ri} \in [x_{mi}, x_{si}^0], \quad i=1, 2. \quad (2.7)$$

де позначено

$$f_1(t) = M\ddot{x} + \sum_{i=1}^2 [K_{ai}(\dot{\alpha}_i \cos \alpha_i) + K_{bi}(\dot{\beta}_i \cos \beta_i)] - K_r(\dot{\psi} \cos \psi)$$

$$f_2(t) = M(\ddot{y} + g) + \sum_{i=1}^2 [K_{ai}(\dot{\alpha}_i \sin \alpha_i) + K_{bi}(\dot{\beta}_i \sin \beta_i)] - K_r(\dot{\psi} \sin \psi)$$

$$f_{3i}(t) = J_i \ddot{\alpha}_i + K_{ai}[\ddot{x} \cos \alpha_i + \ddot{y} \sin \alpha_i + g \sin \alpha_i] + a_i K_{bi}[\ddot{\beta}_i \cos(\alpha_i - \beta_i) + \dot{\beta}_i^2 \sin(\alpha_i - \beta_i)], \quad i = 1, 2,$$

$$f_{4i}(t) = J_{ci} \ddot{\beta}_i + K_{bi}[\ddot{x} \cos \beta_i + \ddot{y} \sin \beta_i + g \sin \beta_i] + a_i K_{bi}[\ddot{\alpha}_i \cos(\alpha_i - \beta_i) - \dot{\alpha}_i^2 \sin(\alpha_i - \beta_i)], \quad i = 1, 2,$$

$$f_5(t) = J\ddot{\psi} - gK_r \sin \psi - K_r(\ddot{x} \cos \psi + \ddot{y} \sin \psi)$$

$$M = m + m_{a1} + m_{b1} + m_{f1} + m_{a2} + m_{b2} + m_{f2}, \quad K_r = rm,$$

$$K_{ai} = m_{ai} r_{ai} + a_i(m_{bi} + m_{fi}), \quad K_{bi} = m_{bi} r_{bi} + b_i m_{fi},$$

$$J_i = J_{ai} + a_i^2(m_{bi} + m_{fi}), \quad J_{ci} = J_{bi} + b_i^2 m_{fi}, \quad i = 1, 2.$$

У співвідношеннях (2.1) – (2.7) позначено:  $R_{ix}(t)$ ,  $R_{iy}(t)$  горизонтальна і вертикальна складові рівнодійної  $\mathbf{R}_i$  сил взаємодії  $i$ -ї ноги з поверхнею опори;  $x_{Ri}$  — абсциса точки прикладання сили  $\mathbf{R}_i$ ; « $\dot{\phantom{x}}$ » — диференціювання за часом  $t$ , символом « $^\circ$ » — значення величини (абсциса точки) за умови, що відповідна точка лежить на прямій  $OX, i=1, 2$ .

Співвідношення (2.1)–(2.5) є рівняннями Лагранжа другого роду, що описують динаміку ОРА людини без урахування ступень. Рівняння (2.6)–(2.7) — це умови кінетостатичної рівноваги невагомих ступень.

Дослідженню системи диференціальних рівнянь (2.1) – (2.7), а також різноманітних її узагальнень та часткових випадків, які описують керований рух двоногих керуючих апаратів, присвячено низку робіт, зокрема [22,35].

## 2.2 Ритмічні, кінематичні та динамічні обмеження на ходу людини

Хо́да людини в усталеному режимі є періодичним процесом з періодом, рівним тривалості подвійного кроку. Позначимо відповідно через  $T_i$ ,  $L_i$  тривалість та довжину  $i$ -го одинарного кроку,  $T=T_1+T_2$ ,  $L=L_1+L_2$  — відповідно тривалість та довжина подвійного кроку. Для визначеності прийемо, що нога з індексом  $i=1$  на початку подвійного кроку починає фазу опори. Прийемо таку послідовність фаз руху ступень: переكات через п'ятку (рис. 2.2), опора на всю ступню (рис. 2.3), плесно-фаланговий переكات (рис. 2.4), переكات через носок, (рис. 2.5) перенесення ступні над поверхнею опори (рис. 2.6) [45].

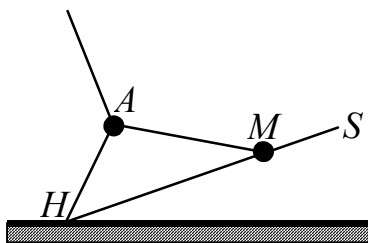


Рисунок. 2.2

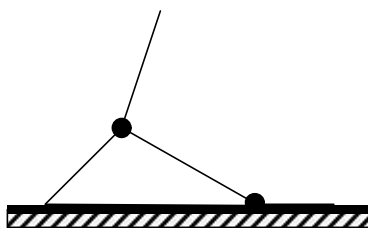


Рисунок. 2.3

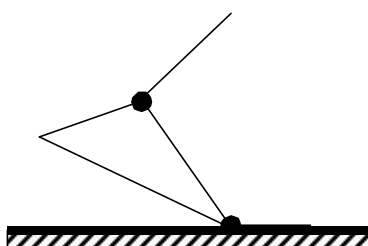


Рисунок. 2.4

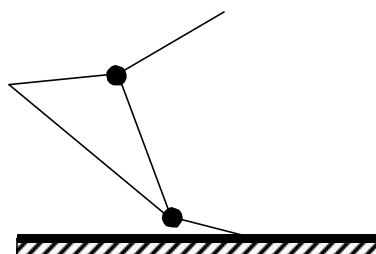


Рисунок 2.5

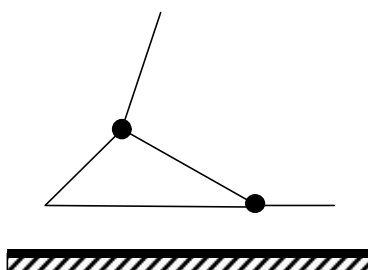


Рисунок 2.6

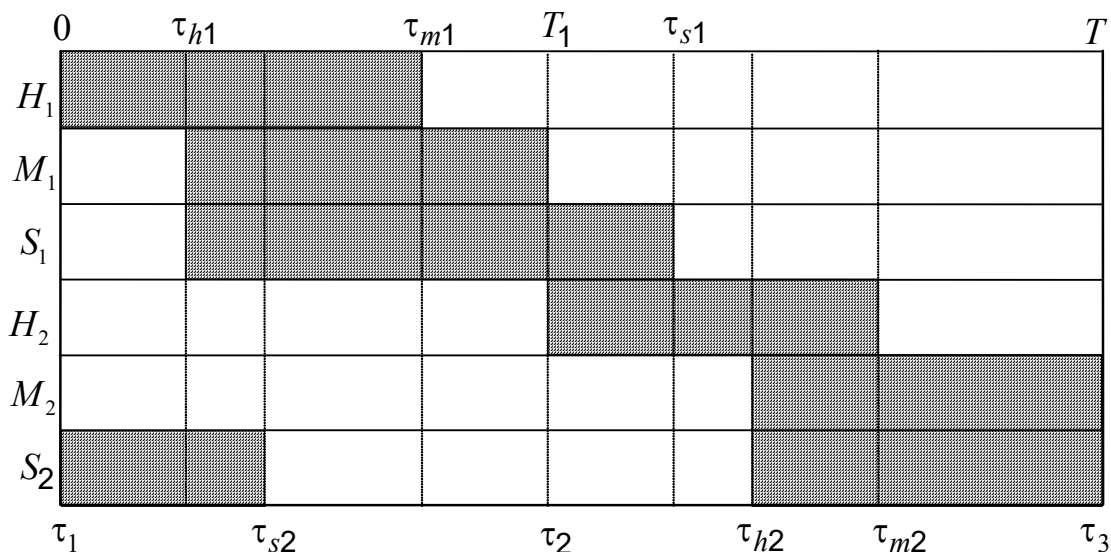


Рисунок 2.7 – Подограма ходи людини

Задамо часові параметри тривалості фаз руху ступні 1-ї ноги:  $t \in [0, \tau_{h1})$  — переكات через п'ятку;  $t \in [\tau_{h1}, \tau_{m1})$  — опора на всю ступню;  $t \in [\tau_{m1}, T_1]$  — плесно-фаланговий переكات;  $t \in (T_1, \tau_{s1})$  — переكات через носок;  $t \in [\tau_{s1}, T)$  — перенесення ступні. Аналогічно визначимо фази руху ступні 2-ї ноги:  $t \in (0, \tau_{s2})$  — переكات через носок;  $t \in [\tau_{s2}, T_1)$  — перенесення ступні;  $t \in [T_1, \tau_{h2})$  — переكات через п'ятку;  $t \in [\tau_{h2}, \tau_{m2})$  — опора на всю ступню;  $t \in [\tau_{m2}, T]$  — плесно-фаланговий переكات. Вибрану ритміку ходи людини можна зобразити за допомогою подограми(рис. 2.7), де заштрихованими областями позначено проміжки часу, на яких точки  $H_i, M_i, S_i, i = 1, 2$  контактують з поверхнею опори. Не заштрихованими — коли ці точки ступні знаходяться вище поверхні опори. На введені часові параметри накладемо “природні” ритмічні умови

$$0 < \tau_{s2} < \tau_{m1} < T_1 < \tau_{s1} < \tau_{m2} < T \quad (2.8)$$

Вибрана послідовність ритмічних фаз характеризується наступними кінематичними умовами антропоморфного руху ніг:

$$x_{hi}(t) \equiv x_{hi}^0, y_{hi}(t) \equiv 0, y_{si} > y_{mi} > 0, \eta_{Mi}(t) = 0, t \in [\tau_i, \tau_{hi}), \quad (2.9)$$

$$x_{hi}(t) \equiv x_{hi}^0, y_{hi}(t) \equiv y_{mi}(t) \equiv y_{si}(t) \equiv 0, t \in [\tau_{hi}, \tau_{mi}), \quad (2.10)$$

$$x_{mi}(t) \equiv x_{mi}^0, y_{hi} > 0, y_{mi}(t) \equiv y_{si}(t) \equiv 0, t \in [\tau_{mi}, \tau_{1+i}], \quad (2.11)$$

$$x_{si}(t) \equiv x_{si}^0, y_{hi} > y_{mi} > 0, y_{si}(t) \equiv 0, \eta_{Mi}(t) \geq 0, t \in (\tau_{3-i}, \tau_{si}), \quad (2.12)$$

$$y_{hi} > 0, y_{mi} > 0, y_{si} > 0, \eta_{Mi}(t) \geq 0, t \in [\tau_{si}, \tau_{4-i}), i = 1, 2; \quad (2.13)$$

де  $\tau_1 = 0, \tau_2 = T_1, \tau_3 = T, \eta_{Mi}(t) = \varepsilon_i(t) - \gamma_i(t) - \varphi_{Mi}$  — кут між прямими  $H_i M_i$  і  $M_i S_i$ . Обмеження (2.13) визначають умову "вільності" ходи [45]. На закон руху досліджуваної механічної системи накладаються ряд обмежень динамічного характеру, які повинні враховуватись під час побудови антропоморфної ходи. Умови, що визначають односторонність взаємодії ступень ніг з поверхнею опори, задаються у вигляді:

$$R_{iy} > 0, t \in [\tau_i, \tau_{i+1}) \cup [\tau_{3-i}, \tau_{si}),$$

$$R_{iy}(t) \equiv R_{ix}(t) \equiv 0, t \in [\tau_{si}, \tau_{4-i}). \quad (2.14)$$

Дане обмеження визначає, що сили, які діють зі сторони поверхні опори, спрямовані вгору.

Для реалізації ходи необхідно забезпечити непроковзування ступні:

$$|R_{ix}(t)| \leq \mu |R_{iy}(t)|, \quad t \in [0, T], \quad i = 1, 2, \quad (2.15)$$

де  $\mu$  — коефіцієнт тертя між ступнею і опорною поверхнею.



Обмеження на керуючий момент у ступні, що забезпечує динамічну рівновагу системи, визначає положення точки прикладення рівнодійної сил реакції опори [24]:

$$\begin{aligned} x_{hi}^0 \leq x_{Ri} \leq x_{si}^0, \quad t \in [\tau_i, \tau_{mi}), \\ x_{mi}^0 \leq x_{Ri} \leq x_{si}^0, \quad t \in [\tau_{mi}, \tau_{1+i}) \cup [\tau_{3-i}, \tau_{si}), \quad i = 1, 2. \end{aligned} \quad (2.16)$$

Дане обмеження вимагає, щоб абсциса рівнодійної сил реакції опорної поверхні знаходилась в межах опорного відрізка ступні. Зокрема, в межах всієї ступні на проміжку часу від початку фази опори до початку фази плеснофалангового перекату, і в межах фаланг на проміжку часу до кінця фази опори.

Як відомо з експериментальних досліджень [53], під час ходи в установленому темпі абсциса точки прикладання рівнодійної сил реакції опорної поверхні протягом фази опори рухається монотонно від п'ятки до носка. Тому вводиться обмеження, яке описує дану властивість:

$$\begin{aligned} \dot{x}_{Ri}(t) \geq \frac{2\varpi(l_i + l_{3i})}{T}, \\ t \in [\tau_i, \tau_{i+1}) \cup [\tau_{3-i}, \tau_{si}), \quad i = 1, 2, \end{aligned} \quad (2.17)$$

де  $\varpi \geq 0$  — заданий параметер.

Задамо умови періодичності руху механічної системи, а також кінематичні умови контактування ступень з поверхнею опори в початковий  $t = 0$  і кінцевий  $t = T$  моменти часу:

$$\xi(T) = \xi(0) + \mathbf{L}, \quad \dot{\xi}(T) = \dot{\xi}(0), \quad (2.18)$$

$$x_{m2}(0) = x_{h1}^0 + l_1 - L_1, \quad y_{m2}(0) = 0, \quad x_{h1}(T) = x_{h1}^0 + L, \quad y_{h1}(T) = 0, \quad (2.19)$$

де  $\mathbf{L} = (L, 0, \dots, 0)$  — вектор розмірності 11;

$$x_{mi}^0 = x_{hi}^0 + l_i, \quad x_{si}^0 = x_{mi}^0 + l_{3i}, \quad x_{h2}^0 = x_{h1}^0 + L_2, \quad i = 1, 2;$$

$x_{h1}^0$  — параметр, який характеризує абсцису п'ятки ноги на початку подвійного кроку,  $x_{h1}^0 = 0$ .

У формулах (2.9)–(2.19) декартові координати точок  $A_i, H_i, M_i, S_i$  ступні  $i$ -ої ноги визначаються співвідношеннями:

$$\begin{aligned} x_i(t) &= x(t) + a_i \sin \alpha_i(t) + b_i \sin \beta_i(t), \\ y_i(t) &= y(t) - a_i \cos \alpha_i(t) - b_i \cos \beta_i(t), \end{aligned} \quad (2.20)$$

$$\begin{aligned} x_{hi}(t) &= x_i(t) + l_{1i} \sin(\gamma_i(t) - \varphi_{Ai}), \\ y_{hi}(t) &= y_i(t) - l_{1i} \cos(\gamma_i(t) - \varphi_{Ai}) \end{aligned} \quad (2.21)$$

$$x_{mi}(t) = x_i(t) + l_{2i} \sin \gamma_i(t), \quad y_{mi}(t) = y_i(t) - l_{2i} \cos \gamma_i(t), \quad (2.22)$$

$$x_{si}(t) = x_{mi}(t) + l_{3i} \sin \varepsilon_i(t), \quad y_{si}(t) = y_{mi}(t) - l_{3i} \cos \varepsilon_i(t), \quad i=1, 2. \quad (2.23)$$

Таким чином, описані обмеження, притаманні ході людини в нормі.

### 2.3 Методика обробки експериментальних біомеханічних даних ходи людини

В даному підрозділі описано процедуру обробки експериментальних подографічних і гоніометричних даних біомеханічних досліджень ходи людини [17] і побудову за її допомогою усереднених функцій, які описують кутові переміщення у 6-ти основних суглобах опорно-рухового апарату людини на проміжку подвійного кроку.

Дані експериментальних досліджень зображуються масивами розмірності  $N$ , у яких зберігається подографічна інформація, а також значення кутів, виміряних гоніометрами. Збір інформації здійснюється з частотою квантування

$h$  ( $j$ -му елементу масиву експериментальних даних відповідає час  $t_j=j/h$ , в який знято значення величин). Елемент подографічної інформації описує умову “є контакт” чи “відсутній контакт” з поверхнею опори п’ятки, 5-ї плюсневої кістки, 1-ї плюсневої кістки і носка для лівої та правої ноги. У вибраній моделі ступні і ритміці руху проміжок  $[\tau_{is}, \tau_{ih})$  відповідає часу, коли контактує з поверхнею опори лише п’ятка; момент  $\tau_{im}$  відповідає часу відриву п’ятки від поверхні опори;  $\tau_{is}$ – час відриву всієї ступні від поверхні опори. Гоніометрична інформація для кожного кута зберігається у вигляді

$$\tilde{\eta}_j = C(\hat{\eta}_j - \eta_0), \quad j=1, \dots, N,$$

де  $C$ – коефіцієнт множення для гоніометра;  $\hat{\eta}_j$ — значення елемента масиву даних гоніометра;  $\eta_0$  — нульове значення гоніометра, яке вимірюється, коли людина стоїть прямо.

Відзначимо, що одержана за допомогою комплексу [20] експериментальна гоніометрична інформація (тонкі криві на рис. 3.1) містить ряд збурень. Для вилучення цих збурень перед подальшою обробкою застосовується метод медіанної цифрової фільтрації [61]. Згідно з даним методом, нелінійна фільтрація масиву експериментальних даних відбувається наступним чином. Задається “кадр” в околі  $j$ -го елемента масиву радіусом  $s$  (елементи масиву  $\tilde{\eta}_k$ ,  $k=j-s, \dots, j+s$ ) і формується масив впорядкованих за зростанням значень  $\tilde{\eta}_{k_i}$ ,  $i=0, \dots, 2s$ . Тоді покладемо, що відфільтроване значення  $\eta_j = \tilde{\eta}_{k_s}$ , де  $\tilde{\eta}_{k_s}$  — середнє за порядком значення серед елементів кадру, впорядкованих за зростанням. Далі, кадр зміщується на один елемент (поклавши  $j=j+1$ ) і процедура повторюється. Для фільтрації використовується кадр радіусом  $s=5$  (здійснювалось сортування масиву розмірності 11). Відфільтровані таким чином дані зображені на рис. 2.8 жирною лінією.

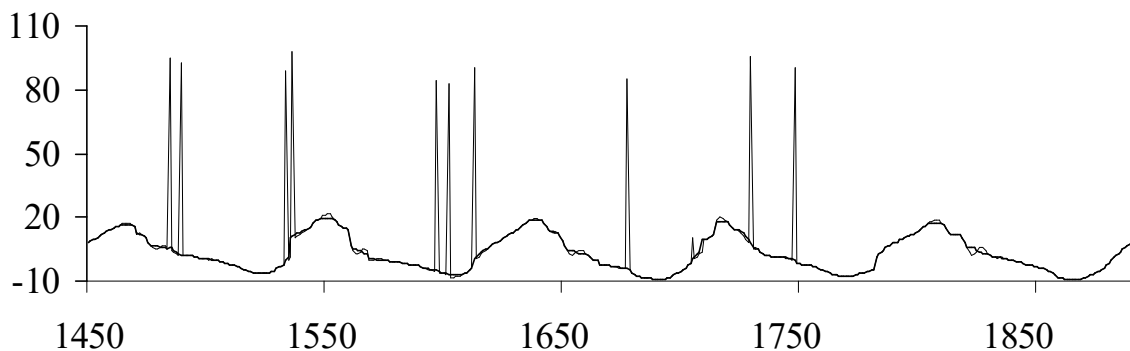


Рисунок 2.8 – Фрагмент експериментальних даних гоніометричних досліджень.  
Кут в стегні збереженої ноги, град. Час в мс.

Аналізуючи значення подографії знаходимо часи початку одинарних кроків. Вхідний масив експериментальних даних розділяється за подвійними кроками. Перший і останній подвійний кроки, на яких відбувається початок і завершення руху, для подальшої обробки не використовується. Далі будуються функції усередненого значення і середньоквадратичного відхилення для оброблюваних одинарних кроків. На кожному подвійному кроці здійснюється періодизація експериментальних даних і нормалізація часу. Покладається  $\tilde{\eta}_{K_k, k} = \tilde{\eta}_{0, k}$ ,  $t_{j, k} = j/N_k$ ,  $k=1, \dots, n$ ,  $j=0, \dots, N_k$ ,  $n$  — кількість усереднених подвійних кроків,  $N_k$  — кількість квантів часу, що припадає на  $k$ -й подвійний крок. Будується сітка за часом для всіх подвійних кроків  $\sigma_l = l/N_c$ ,  $l=0, \dots, N_c$ , де  $N_c$  — кількість проміжків спільного розбиття. За допомогою лінійної сплайн-інтерполяції знаходиться значення кутів у суглобах  $\eta_{lk}$ ,  $l=0, \dots, N_c$  у моменти часу  $\sigma_l$  на  $k$ -му подвійному кроці,  $k=1, \dots, n$ .

Визначається середнє значення кута  $\bar{\eta}_l$  для моментів часу  $\sigma_l$  за усіма подвійними кроками:

$$\bar{\eta}_l = \frac{1}{n} \left( \sum_{k=1}^n \eta_{lk} \right), \quad l=0, \dots, N_c,$$

Аналогічно, обчислюється середнє відхилення значення кута

$$\Delta\eta_l = \left( \frac{1}{n} \sum_{k=1}^n |\bar{\eta}_l - \eta_{lk}|^2 \right)^{\frac{1}{2}}, \quad l=0, \dots, N_c.$$

Таким чином, описано процедуру обробки експериментальних даних біомеханічних досліджень ходи людини, одержаних за допомогою комплексу [20]. Використовуючи задані на послідовності подвійних кроків дані експериментальних досліджень, побудовані функції, які описують середнє значення кутів і їх середньо квадратичне відхилення на одному подвійному кроці.

## 2.4 Висновок

У даному розбілі було досліджено модель опорно-рухового апарата людини. Виведені рівняння руху (2.1) – (2.7) . Описані обмеження які притаманні ході людини в нормі. Описано процедуру обробки експериментальних подографічних і гоніометричних даних біомеханічних досліджень ходи людини [17] і побудовано за її допомогою усереднених функцій, які описують кутові переміщення у 6-ти основних суглобах опорно-рухового апарату людини на проміжку подвійного кроку.

### 3 Дослідження динаміки крокуючого апарата в середовищі MATLAB Simulink

Математична модель крокуючого апарата, що представлена в розділі 2 характеризується підвищеною складністю і мало придатна для дослідження аналітичними методами. Основний об'єм дослідження виконується методами комп'ютерного моделювання із використанням відповідного програмного забезпечення. Програмне забезпечення, що використовується для моделювання, повинно задовольняти наступним вимогам:

- Максимальна автоматизація дослідження моделі на етапах її побудови, чисельного розв'язування та обробки результатів.
- Можливість задавання довільних функціональних залежностей для опису законів руху елементів системи, сил взаємодії апарата із зовнішнім середовищем, тощо.
- Представлення математичних моделей крокуючого апарата як механічної системи та системи керування ним в рамках єдиної моделі на основі єдиного формалізму.
- Наявність засобів наочного та інформативного представлення результатів дослідження.

Виходячи з вказаних вимог, в якості засобу дослідження динаміки крокуючого апарата в даній роботі було обрано середовище MATLAB, Simulink.

Для моделювання механічної частини крокуючого апарату використано пакет SimMechanics із стандартної бібліотеки моделей Simulink. В рамках вказаного пакету багатомасова механічна система (зокрема, крокуючий апарат) розглядається як сукупність абсолютно твердих тіл, що з'єднані шарнірами. Для твердих тіл, шарнірів, різноманітних сенсорів та засобів докладання зовнішніх зусиль розроблені відповідні модельні компоненти. Побудова моделі системи зводиться до об'єднання необхідних компонент. Таким чином,

користувач позбавляється необхідності ручного виконання громіздких формальних математичних перетворень, процес побудови моделі і внесення до неї модифікацій відбувається максимально швидко і достатньо захищений від внесення користувачем формальних помилок.

Схема верхнього рівня моделі крокуючого апарата представлена на рис. 3.1. Основними елементами на цьому рівні є корпус апарата, що моделюється блоком `Body`, ліва і права кінцівки (блоки `Left`, `Right`) та їх стопи (блоки `Left_foot`, `Right_foot`). Окрім цього, до моделі включено блоки `Machine Environment` та `Ground`, що задають базові параметри моделі, такі як орієнтація базової системи координат, значення прискорення сили тяжіння та ряд технічних параметрів, що пов'язані із специфікою чисельного дослідження моделі. Блок `Plane` фіктивним шарніром, що вводить обмеження, характерні для плоского руху апарата. В рамках основної площини рух апарата є вільним, однак, для полегшення відлагодження в моделі передбачено можливість часткової або повної фіксації корпусу відносно базової системи відліку. Для цього використовуються блоки `Fixation_X(Y, Psi)` та `Joint_Actuator_X(Y, Psi)`. В нормальному режимі, як можна бачити, ці блоки відімкнені від основної моделі. Блок `Body_deflection` є сенсором, що контролює відхилення корпусу від вертикалі в процесі руху. Поточне значення відхилення передається в блоки моделей кінцівок, де ця інформація використовується для роботи системи керування рухом апарата.

Окрім цього, модель включає ще одне тіло, що виконує роль поверхні руху. Це тіло є нерухомим відносно базової системи координат і на схемі рис. 3.1 для простоти не показане.

Перейдемо до розгляду моделі кінцівки. Як було зазначено в розділі 2, власне кінцівка без урахування стопи складається з двох ланок: стегна та гомілки. Схема моделі кінцівки представлена на рис. 3.2. Модель включає блоки для моделювання стегна (`Hip`), гомілки (`Leg`) і трьох шарнірів-суглобів, що пов'язують ланки між собою і кінцівку з корпусом і стопою. Окрім цього

модель включає блоки для задавання еталонного закону руху кінцівки та сервоприводів, що керують рухом в кожному з трьох суглобів.



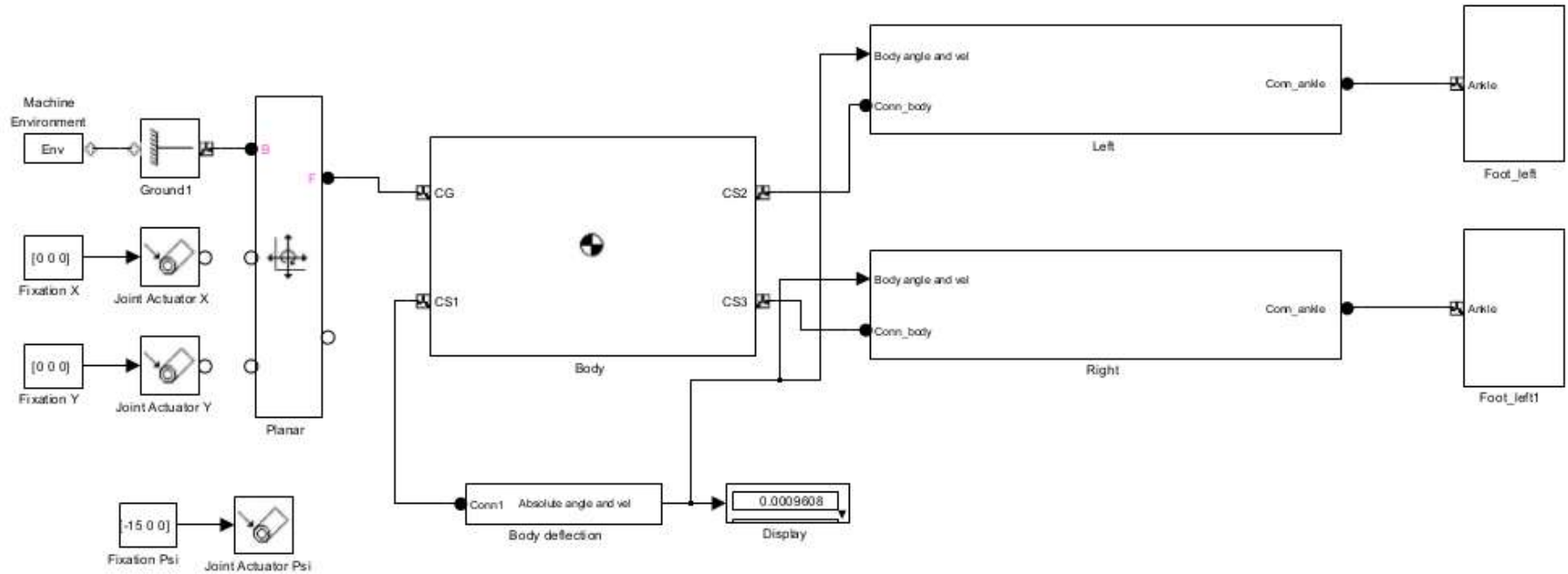


Рисунок 3.1 – Схема верхнього рівня моделі крокуючого апарата в середовищі MATLAB, Simulink з використанням елементів бібліотеки SimMechanics

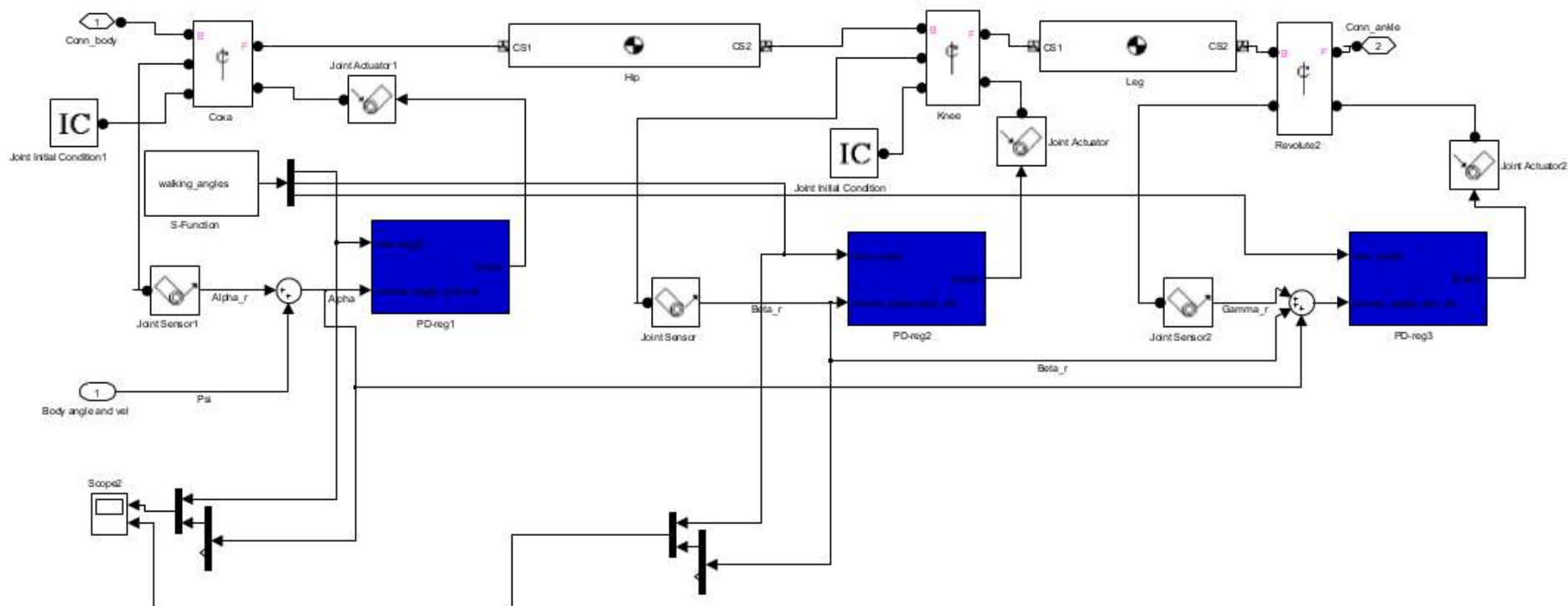


Рисунок 3.2 – Схема моделі дволанкової кінцівки крокуючого апарата

Моделі тіл – ланок кінцівки передбачають задавання геометричних та кінематичних параметрів, таких як розташування точок під'єднання шарнірів в системі координат, що пов'язана з тілом, маса тіла та його момент інерції відносно центра мас.

Моделі шарнірів включають додаткові елементи-блоки, призначені для задавання обмежень по кутах повороту, початкових умов, а також докладання до шарніру моментів керування від сервоприводів та зняття інформації про поточні значення кута повороту та кутової швидкості, що потрібна для керування сервоприводами.

Керування рухом кінцівки здійснюється за принципом слідкуючої системи. В якості еталонного руху використовуються кінематичні параметри руху, отримані шляхом відео фіксації руху реальної людини. Ці кінематичні параметри передаються на входи сервоприводів, а останні намагаються відтворити еталонний рух шляхом формування відповідних керуючих моментів. Сервопривід побудовано за схемою ПД-регулятора, тобто закон керування має вигляд:

$$T_c = -k_a(\varphi - \varphi^*) - k_v(\omega - \omega^*),$$

де  $T_c$ – керуючий момент, що формується сервоприводом,  $(\varphi^*, \omega^*)$ – програмні, а  $(\varphi, \omega)$ – фактичні значення кута повороту в шарнірі та кутової швидкості;  $(k_a, k_v)$ – коефіцієнти – параметри закону керування.

Модель стопи, окрім опису її властивостей як твердого тіла описує також взаємодію з опорною поверхнею. Ми вважаємо, що сили реакції поверхні зведені в загальному випадку до двох точок опори: передньої та задньої. Сила, що докладається в кожній з цих точок, має дві складових: нормальну та дотичну. Перша з них є пружною реакцією опорної поверхні, а друга

обумовлена дією сил кулонового тертя. Модель сили тертя, що використана при побудові моделі передбачає адекватне відтворення як зчепленого стану стопи з поверхнею, так і проковзування стопи відносно поверхні.

Початкові умови для корпусу апарата та між ланкових шарнірів встановлюються таким чином, щоб в початковий момент обидві стопи, точніше, контактні точки обох стоп знаходились в декількох сантиметрах над опорною поверхнею. Це дозволяє апарату достатньо швидко та «м'яко» (без значних ударних навантажень) приземлитись на опорну поверхню і встановити фізично обумовлену взаємодію з останнею. Після встановлення стійкої взаємодії рух відбувається циклічно, і апарат крок за кроком пересувається вздовж опорної поверхні.

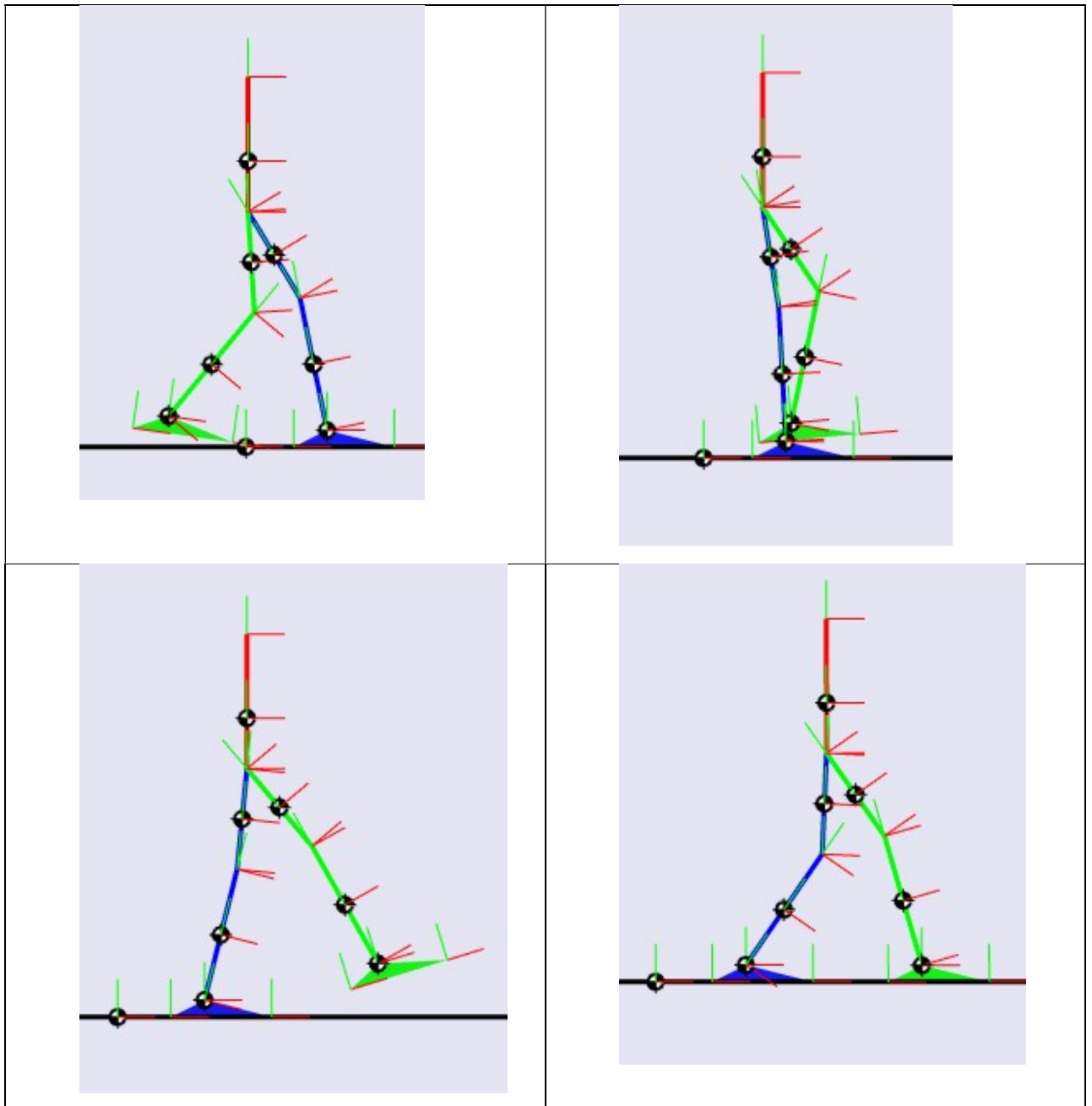


Рисунок 3.3 – Вибрані фази ходи крокуючого апарата за результатами моделювання

На рис. 3.3 представлено декілька фаз кроку апарата. Можна бачити, як відбувається відрив задньої (в даний момент) кінцівки від опорної поверхні, її перенос в передню позицію і так званий «передній поштовх», в результаті якого передня кінцівка входить в щільний, без проковзування, контакт з

опорною поверхнею, після чого стає можливим перенос на неї основної ваги тіла, тобто щойно перенесена кінцівка перебирає на себе функції опори тіла.

Наведена на рис. 3.3 візуалізація використовується для контролю правильності моделювання, та стійкості режиму ходи. Для подальшого ж моделювання руху апарата в середовищі Unity кінематичні параметри руху, такі як кути повороту та швидкості в кожному з суглобів, а також поточні координати центра мас корпуса і кут нахилу корпуса відносно вертикалі фіксуються через рівні проміжки часу і записуються до файлу. В даному моделюванні період кроку становив 1.4 с, інтервал фіксування параметрів 0.01 с. Рух апарата вважається строго періодичним, тобто результати моделювання фіксуються протягом одного кроку, а в подальшому використовуються циклічно.

### **3.1 Висновок**

В даному розділі було обрано для моделювання крокуючого апарата середовище MATLAB, Simulink. Для моделювання механічної частини крокуючого апарату використано пакет SimMechanics із стандартної бібліотеки моделей Simulink.

Розроблена модель крокуючого апарата. Вибрані фази ходи крокуючого апарата за результатами моделювання які використовуються для контролю правильності моделювання, та стійкості режиму ходи. Для подальшого ж моделювання руху апарата в середовищі Unity кінематичні параметри руху, такі як кути повороту та швидкості в кожному з суглобів, а також поточні координати центра мас корпуса і кут нахилу корпуса відносно вертикалі зафіксовано через рівні проміжки часу і записані до файлу.

## 4 Моделювання руху крокуючого апарата в середовищі Unity

### 4.1 Середовище розробки Unity

Unity — кросплатформенний інструмент для розробки дво- та тривимірних додатків та ігор, що працює на операційних системах Windows і OS X. Створені за допомогою Unity додатки працюють під системами Windows, OS X, Android, Apple iOS, Linux, а також на гральних консолях Wii, PlayStation 3 і Xbox 360. Також є можливість створювати інтернет-додатки за допомогою спеціального під'єднуваного модуля для браузера Unity, а також за допомогою експериментальної реалізації в межах модуля Adobe Flash Player. Додатки, створені за допомогою Unity, підтримують DirectX та OpenGL.

Редактор Unity має простий Drag & Drop інтерфейс, який легко налаштовувати, що складається з різних вікон, завдяки чому можна проводити налагодження проекту прямо в редакторі. Рушій підтримує дві скриптові мови програмування: C# і JavaScript. Проект в Unity ділиться на сцени (рівні) - окремі файли, що містять свої ігрові світи зі своїм набором об'єктів, скриптів, і налаштувань. Сцени можуть містити в собі як, об'єкти (моделі), так і порожні ігрові об'єкти – тобто ті які не мають моделі. Об'єкти, в свою чергу містять набори компонентів, з якими і взаємодіють скрипти. Також у них є назва (в Unity допускається наявність двох і більше об'єктів з однаковими назвами), може бути тег (мітка) і шар, на якому він повинен відображатися. Так, у будь-якого предмета на сцені обов'язково присутній компонент Transform - він зберігає в собі координати місця розташування, повороту і розмірів по всіх трьох осях. У об'єктів з видимою геометрією також за замовчуванням присутній компонент Mesh Renderer, що робить модель видимою.

Також Unity підтримує фізику твердих тіл і тканини, фізику типу Ragdoll (ганчіркова лялька). У редакторі є система успадкування об'єктів; дочірні об'єкти будуть повторювати всі зміни позиції, повороту і масштабу батьківського об'єкта. Скрипти в редакторі прикріплюються до об'єктів у вигляді окремих компонентів.

При імпорті текстури в рушій можна згенерувати alpha-канал, mip-рівні, normal-map, light-map, карту відображень, проте безпосередньо на модель текстуру прикріпити не можна - буде створено матеріал, з яким буде призначений шейдер, і потім матеріал прикріпиться до моделі. Редактор Unity підтримує написання і редагування шейдерів. Крім того він містить компонент для створення анімації, яку також можна створити попередньо в 3D-редакторі та імпортувати разом з моделлю, а потім розбити на файли.

Unity ідеально підходить для двох цілей одночасно: розрахунок фізики та візуальне відображення. Можливі й інші варіанти, такі як CryEngine, або UnrealEngine, проте Unity показав себе як найпретіший, найшвидший і найзрозуміліший інструмент для розробки системи, тому було вибрано саме його.

З двох варіантів скриптової мови (C#; JavaScript) було вибрано C# через її більшу зрозумілість і кращу документованість.

## **4.2 Створення моделі крокуючого апарата в середовищі розробки Unity**

Модель крокуючого апарата (Рис. 4.1) було створено за допомогою стандартних об'єктів Unity під назвою GameObject. GameObject є основними об'єктами в Unity, які представляють собою символи, реквізит і декорації. Вони



не виконують багато ролей, але виступають в якості контейнерів для компонентів, які реалізують реальну функціональність.

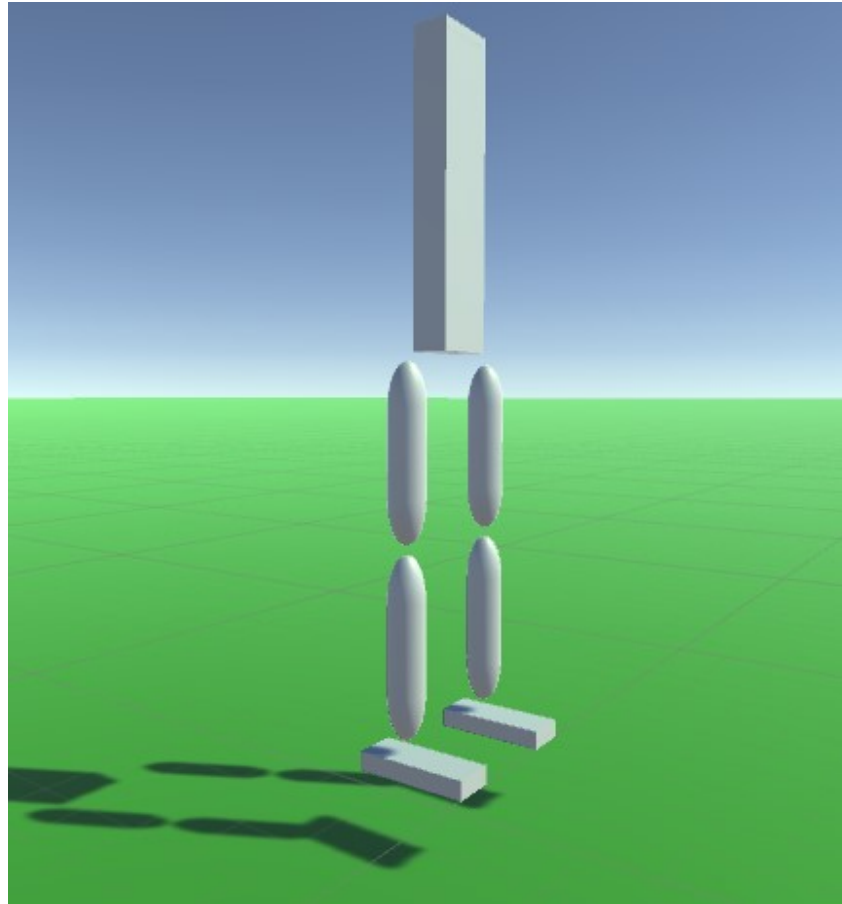


Рисунок 4.1 – Модель крокуючого апарата у середовищі Unity

Апарат (Apparat.w3) складається із 3-ох основних частин (Рис. 4.2):

- тіло (Body);
- ліва нога (Leg l);
- права нога (Leg r).



Рисунок 4.2 – Логічна структура моделі крокуючого апарата у середовищі Unity

Ці об'єкти в середовищі Unity являють собою пустий GameObject і призначені для структуризації апарату задля кращого його подання при подальшій роботі з ним.

Компоненти “Leg l” та “Leg r” кожен складається з таких частин:

- Верхня частина ноги (Leg Up);
- Нижня частина ноги (Leg Down);
- Стопа(Stopa).

Компонент “Body” має лише одну частину – таз (Taz).

Всі ці компоненти (Leg Up, Leg Down, Stopa, Taz ) виконані на основі стандартних GameObject (Рис. 4.3) в середовищі Unity: “Leg Up”, “Leg Down” на основі 3D об'єкта Capsule, а “Stopa”, “Taz” на основі 3D об'єкта Cube.

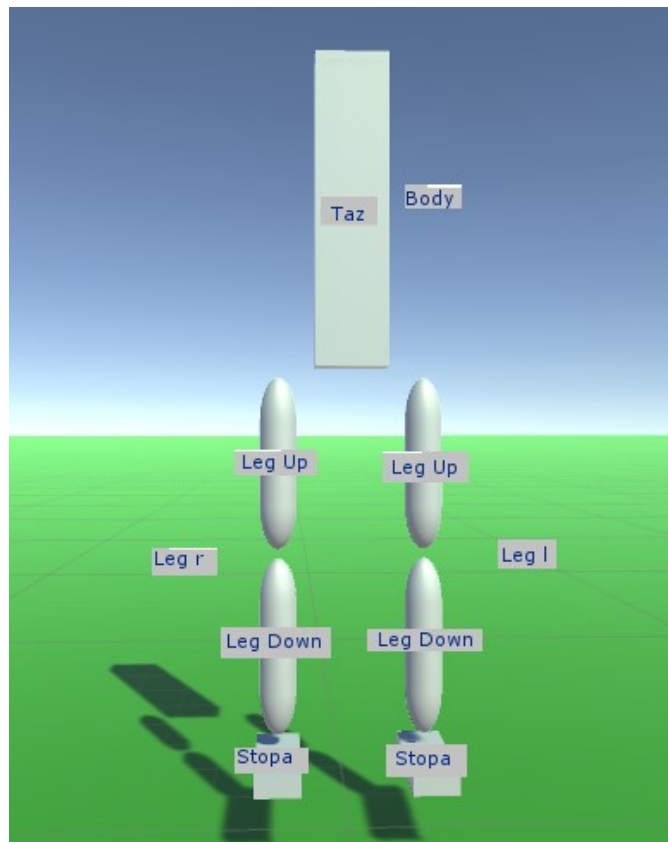


Рисунок 4.3 – Відповідність між візуальною і логічною моделлю апарата у середовищі Unity

Кожен GameObject у середовищі Unity має компонент Transform, де можна задати початкову позицію, кут та масштаб об'єкта. Саме завдяки цьому можна налаштувати пропорції апарата відповідно до тих що використовувалися при роботі в Matlab. Компоненти моделі апарата в Unity мають такі пропорції по висоті(Рис. 4.4):

- Leg Up – 47 відносних одиниць;
- Leg Down – 48 відносних одиниць;
- Stopa – 5 відносних одиниць;
- Taz – 85 відносних одиниць.

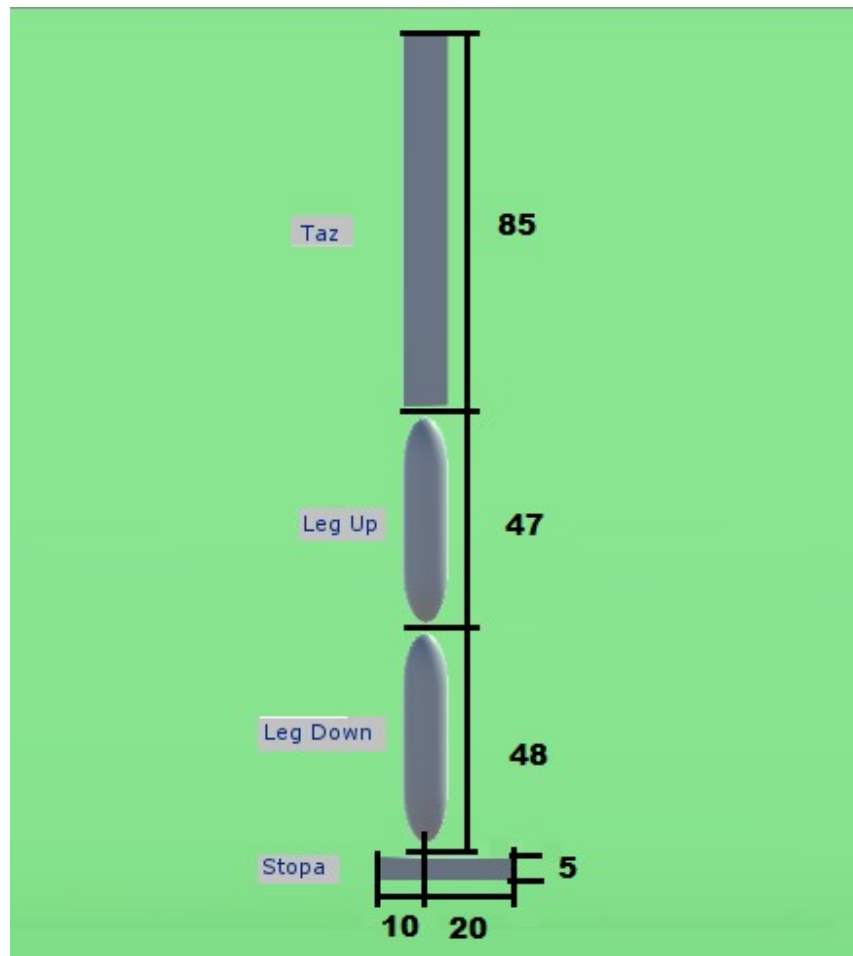


Рисунок 4.4 – Пропорції моделі апарата у середовищі Unity

До кожного GameObject Leg Up, Leg Down, Stopa, Taz додано компонент Rigidbody та Collider. Rigidbody дозволяє GameObject діяти під контролем фізики. Rigidbody може отримати сили і крутний момент, щоб змусити об'єкти рухатися реалістично. Будь-який GameObject повинен містити Rigidbody щоб потрапити під вплив сили тяжіння або взаємодіяти з іншими об'єктами через фізичний движок NVIDIA PhysX. Компонент Collider визначають форму об'єкта для цілей фізичних зіткнень.

GameObject Leg Up, Leg Down, Stopa, Taz з'єднані між собою за допомогою компонента HingeJoint. Hinge Joint або шарнірна петля це група з двох Rigidbodies, що змушує їх переміщатися спільно, немов вони з'єднані петлею.

Частини кожної ноги апарата з'єднані у такому порядку: Stopa приєднана до нижнього кінця Leg Down, Leg Down приєднана до нижнього кінця Leg Up, Leg Up приєднана до нижнього кінця до спільної для обох ніг частини Taz, але з протилежних боків.

### **4.3 Створення опорної поверхні в середовищі розробки Unity**

Для створення опорної поверхні було використано стандартний GameObject під назвою Plane, який являє собою плоский квадрат. Plane було надано плоский Collider та розтягнуто по осям Oz та Ox у 1000 разів та жорстко зафіксовано у просторі. Також, компоненту Plane було надано зеленого кольору для кращої спостережуваності апарата на фоні опорної поверхні.

### **4.4 Створення законів руху частин крокуючого апарата в середовищі розробки Unity**

До кожного GameObject Leg Up, Leg Down, Stopa апарата додано компонент script на мові C#, а саме benderfallower. Unity дозволяє створювати свої власні компоненти за допомогою Script. Це дозволяє викликати події, змінювати властивості компонентів з плином часу і реагувати на введення даних користувачем. Script являє собою програмний код на мові C#. При створенні нового компонента script середовище Unity створює клас який назва якого є і назвою script, в якому є дві функції void Start()(тут задаються початкові значення, виконується один раз одразу після створення свого GameObject) та void Update()(тут задаються закони зміни параметрів зі зміною кожного кадру, виконується кожний раз при зміні кадру на кожному кадрі).

Слід зазначити, що деякі змінні класів в script подані режимі доступу public це дозволяє спостерігати та змінювати значення цих змінних у процесі виконання програми в редакторі Unity.

Основною задачею script `benderfollower` є контроль зміни кута між частиною до якої його додано та частиною з якою частина зі script'ом з'єднана. `Benderfollower` приймає такі значення:

- `public float TargetVelocity` (максимальна кутова швидкість між з'єднаними частинами);
- `public float TargetForce` (сила повороту між з'єднаними частинами);
- `public float PolosaAngle` (максимальний градус похибки);
- `public float StartAngle` (початковий кут між з'єднаними частинами);
- `public float FallowAngle` (кут відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана якого буде намагатися досягти частина з `benderfollower`);
- `public float MinAngle`(мінімальний кут відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана);
- `public float MaxAngle` (максимальний кут відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана);
- `public bool Block = false` (змінна створена задля функції, яка фіксує частину з `benderfollower` відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана);
- `public int WaitFrame = 100` (номер кадру на якому буде почато рух апарата);
- `public string NameOfPart = "Name"` (назва та номер даної частини апарата);
- `public float TimeStep` (крок часу між зміною значень кутів слідування);
- `public float[] Angels` (кути відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана яких буде намагатися досягти частина з `benderfollower` по порядку їх зміни);

У `benderfollower` у функції `void Start()` встановлюються початкові параметри максимального та мінімального кута відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана, та сила повороту між з'єднаними частинами.

У `benderfollower` у функції `void Update()` на встановленому у `WaitFrame` номері кадру виконується функція пересування(`void Moving()`) та функція блокування(`void Blocking()`) зміни кута відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана.

Функція `void Moving()` виконує зміну кутової швидкості між з'єднаними частинами залежно від поточного кута та кута слідування (`FallowAngle`). Функція з кожним наступним кроком зміни часу (`TimeStep`) встановлює наступний кут слідування з `Angels[]`. А потім (Рис. 4.5) , якщо поточний кут відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана доданий до кута похибки (`PolosaAngle`) менший кута слідування (`FallowAngle`), то встановлюється кутова швидкість рівна заданій (`TargetVelocity`). Якщо поточний кут відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана мінус кут похибки (`PolosaAngle`) більший кута слідування (`FallowAngle`), то встановлюється кутова швидкість рівна відємній заданій (`-TargetVelocity`). Якщо дві попередні умови не виконуються, то встановлюється кутова швидкість рівна нулю.

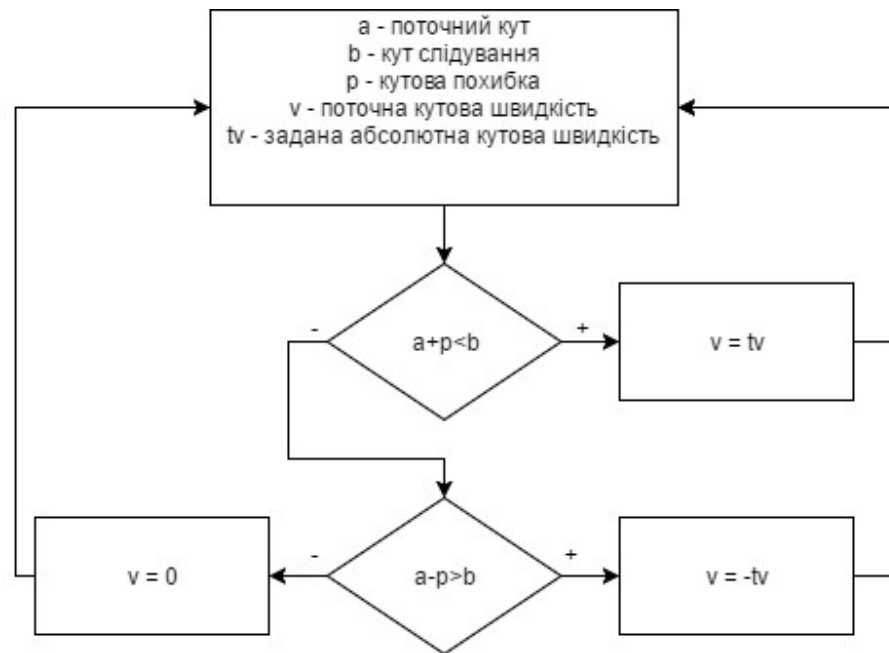


Рисунок 4.5 – Алгоритм роботи частини відповідальної за зміну кутової швидкості функції void Moving().

Функція void Blocking() блокує частину з benderfallower відносно продовжної вісі частини до якої частина з benderfallower з'єднана, якщо змінна Block = true, і розблоковує якщо Block = false.

#### 4.5 Створення елемента контролю тестування ходи крокуючого апарата в середовищі розробки Unity

Для створення елемента контролю тестування ходи крокуючого апарата було використано пустий GameObject та доданого до нього script TestControllerFollower. Головною задачею TestControllerFollower є:

- зчитування з файлу дані про значення кутів між всіма з'єднаними шарнірними зв'язками (HingeJoint) частинами апарата, отриманими при моделюванні руху крокуючого апарата в середовищі Matlab – Simulink, в певний момент циклу ходи;



- створення копії моделі крокуючого апарата;
- передача частинам апарата, що містять компонент `benderfollower` їх початкових даних(`TargetVelocity`, `TargetForce`, `PolosaAngle`, `StartAngle`, `MinAngle`, `MaxAngle`, `WaitFrame`, `Block`, `NameOfPart`, `TimeStep` ) та значення кута(`Angels[]`) відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана, отриманими при моделюванні руху крокуючого апарата в Matlab – Simulink, в певний момент циклу ходи;
- запуск моделювання ходи крокуючого апарата.

Компонент `TestControllerFollower` приймає такі значення:

- `public GameObject apparat` (модель апарата, який буде використано для моделювання ходи);
- `public Vector3 spawnValues` (Позиція на якій з'явиться апарат);
- `public float TargetVelocity` (максимальна кутова швидкість між з'єднаними частинами);
- `public float Force` (сила повороту між з'єднаними частинами);
- `public float Polosa` (максимальній градус похибки);
- `public int WaitF` (номер кадру на якому буде почато рух апарата);
- `public float[] LeglUAngles = { -90.0f, 90.0f }` (мінімальний та максимальний кут відносно продовжної вісі частини `Taz` до продовжної вісі частини `Leg Up` лівої ноги(`Leg 1`));
- `public float[] LeglDAngles = { -90.0f, 90.0f }` (мінімальний та максимальний кут відносно продовжної вісі частини `Leg Up` лівої ноги(`Leg 1`) до продовжної вісі частини `Leg Down` лівої ноги(`Leg 1`));
- `public float[] LeglSAngles = { -90.0f, 90.0f }` (мінімальний та максимальний кут відносно продовжної вісі частини `Leg Down` лівої ноги(`Leg 1`) до продовжної вісі частини `Stora` лівої ноги(`Leg 1`));

- `public float[] LegrUAngles = { -90.0f, 90.0f }` (мінімальний та максимальний кут відносно продовжної вісі частини Taz до продовжної вісі частини Leg Up правої ноги(Leg r));
- `public float[] LegrDAngles = { -90.0f, 90.0f }` (мінімальний та максимальний кут відносно продовжної вісі частини Leg Up правої ноги(Leg r) до продовжної вісі частини Leg Down правої ноги(Leg l));
- `public float[] LegrSAngles = { -90.0f, 90.0f }` (мінімальний та максимальний кут відносно продовжної вісі частини Leg Down правої ноги(Leg l) до продовжної вісі частини Stora правої ноги(Leg l));
- `public float TimeStep = 1.0f` (крок часу між зміною значень кутів слідування);
- `public bool start = false;` (перемикач для зручного ручного початку моделювання ходи крокуючого апарата).

У `TestControllerFollower` у функції `void Start()` виконується функція (`void ParametrReader()`) зчитування з файлу даних про значення кутів між всіма з'єднаними шарнірними зв'язками (`HingeJoint`) частинами апарата, отриманими при моделюванні руху крокуючого апарата в середовищі Matlab – Simulink, в певний момент циклу ходи. Також, запускається функція(`private IEnumerator FirstStart(float waitTime)`), яка виконується незалежно від зміни кадрів, створення копії моделі крокуючого апарата та передає частинам апарата, що містять компонент `benderfollower` їх початкових даних(`TargetVelocity`, `TargetForce`, `PolosaAngle`, `StartAngle`, `MinAngle`, `MaxAngle`, `WaitFrame`, `Block`, `NameOfPart`, `TimeStep` ) та значення кута(`Angles[]`) відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана, отриманими при моделюванні руху крокуючого апарата в Matlab – Simulink, в певний момент циклу ходи.

У `TestControllerFollower` у функції `void Update()` запускається процес симуляції ходи крокуючого апарата, якщо змінна `start = true`.

Функція `void ParametrReader()` зчитує з файлу дані про значення кутів між всіма з'єднаними шарнірними зв'язками (`HingeJoint`) частинами апарата, отриманими при моделюванні руху крокуючого апарата в середовищі `Matlab – Simulink`, в певний момент циклу ходи.

Функція `private IEnumerator FirstStart(float waitTime)` виконується незалежно від зміни кадрів, створює копії моделі крокуючого апарата та передає частинам апарата, що містять компонент `benderfollower` їх початкових даних (`TargetVelocity`, `TargetForce`, `PolosaAngle`, `StartAngle`, `MinAngle`, `MaxAngle`, `WaitFrame`, `Block`, `NameOfPart`, `TimeStep`) та значення кута (`Angels[]`) відносно продовжної вісі частини до якої частина з `benderfollower` з'єднана, отриманими при моделюванні руху крокуючого апарата в `Matlab – Simulink`, в певний момент циклу ходи.

#### **4.6 Порівняння ходи моделей крокуючого апарата виконаних в середовищі розробки `MATLAB Simulink` та `Unity`**

Вибрані фази ходи моделі крокуючого апарата за результатами моделювання в середовищі `MATLAB Simulink` було порівняно з тими ж самими фазами ходи крокуючого апарата вибраними за результатом моделювання в середовищі `Unity` (рис 4.6).

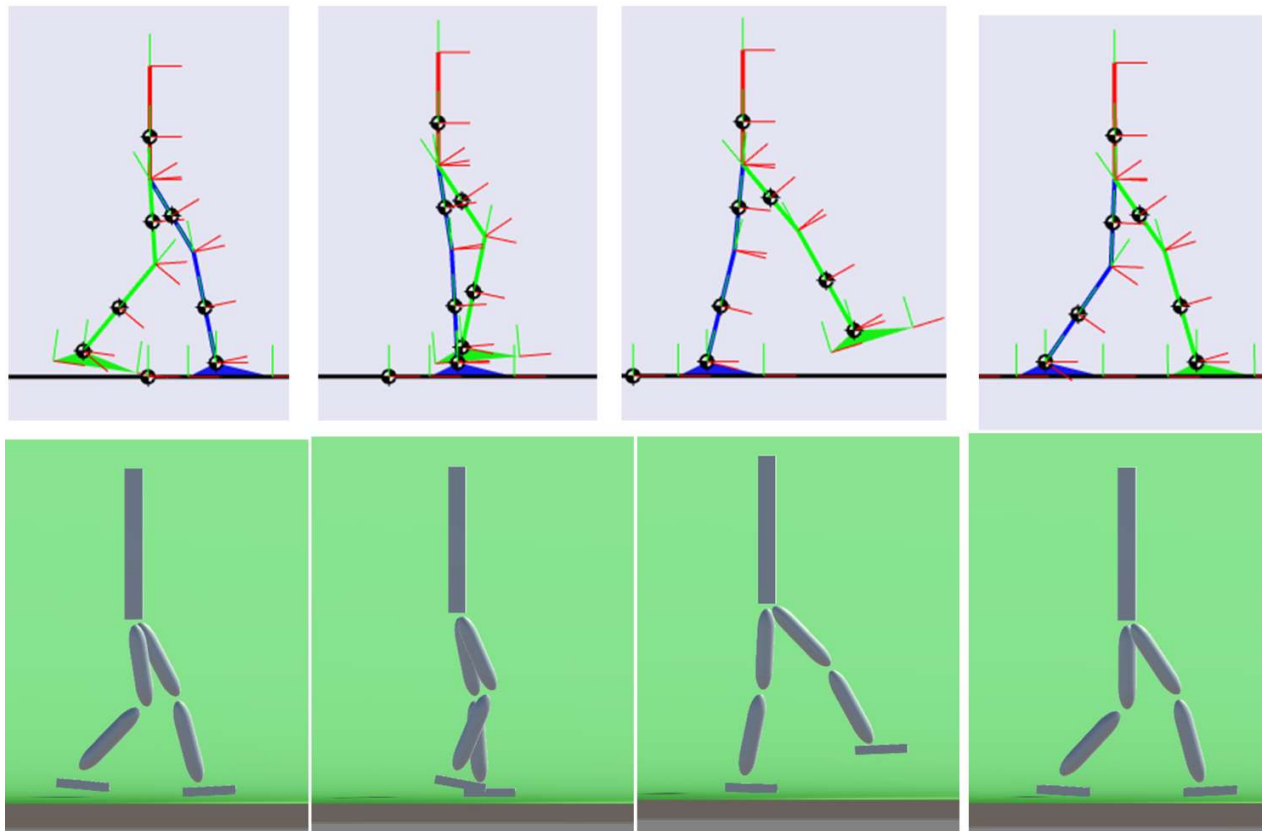


Рисунок 4.6 – Порівняння ходи моделей крокуючого апарата виконаних в середовищах MATLAB Simulink та Unity

Як видно з рисунку 4.6 всі представлені фази ходи моделі крокуючого апарата виконані в середовищі MATLAB Simulink були повторені моделлю крокуючого апарата в середовищі Unity.

#### 4.7 Висновок

У даному розділі побудовано модель антропоморфного крокуючого апарата в середовищі Unity, яка відповідає моделі крокуючого апарата створеного раніше в середовищі MATLAB Simulink.

Створено ОП, закони руху частин крокуючого апарата, елемент контролю тестування ходи крокуючого апарата в середовищі розробки Unity.

Порівняно фази ходи моделей крокуючого апарата виконані в середовищі MATLAB Simulink з виконаними в середовищі Unity. Всі представлені фази ходи моделі крокуючого апарата виконані в середовищі MATLAB Simulink повторені моделлю крокуючого апарата в середовищі Unity.

## **5 Функціонально-вартісний аналіз програмного продукту**

### **5.1 Вступ**

У даному розділі проводиться оцінка основних характеристик програмного продукту, призначеного для керування рухом крокуючого робота. Програмний продукт був розроблений за допомогою мови програмування C# у середовищі розробки Unity.

Програмний продукт призначено для використання на персональних комп'ютерах під управлінням операційної системи Windows.

Нижче наведено аналіз різних варіантів реалізації модулю з метою вибору оптимальної, з огляду при цьому як на економічні фактори, так і на характеристики продукту, що впливають на продуктивність роботи і на його сумісність з апаратним забезпеченням. Для цього було використано апарат функціонально-вартісного аналізу.

Функціонально-вартісний аналіз (ФВА) – це технологія, яка дозволяє оцінити реальну вартість продукту або послуги незалежно від організаційної структури компанії. Як прямі, так і побічні витрати розподіляються по продуктам та послугам у залежності від потрібних на кожному етапі виробництва обсягів ресурсів. Виконані на цих етапах дії у контексті метода ФВА називаються функціями.

Мета ФВА полягає у забезпеченні правильного розподілу ресурсів, виділених на виробництво продукції або надання послуг, на прямі та непрямі витрати. У даному випадку – аналізу функцій програмного продукту й виявлення усіх витрат на реалізацію цих функцій.

Фактично цей метод працює за таким алгоритмом:

- Визначається послідовність функцій, необхідних для виробництва продукту. Спочатку – всі можливі, потім вони розподіляються по двом групам: ті, що впливають на вартість продукту і ті, що не впливають. На цьому ж етапі оптимізується сама послідовність скороченням кроків, що не впливають на цінність і відповідно витрат.
- для кожної функції визначаються повні річні витрати й кількість робочих часів.
- для кожної функції на основі оцінок попереднього пункту визначається кількісна характеристика джерел витрат.
- після того, як для кожної функції будуть визначені їх джерела витрат, проводиться кінцевий розрахунок витрат на виробництво продукту.

## **5.2 Постановка задачі техніко-економічного аналізу**

У роботі застосовується метод ФВА для проведення техніко-економічного аналізу розробки системи аналізу нелінійних нестационарних процесів. Оскільки основні проектні рішення стосуються всієї системи, кожна окрема підсистема має їм задовольняти. Тому фактичний аналіз представляє собою аналіз функцій програмного продукту, призначеного для збору, обробки та проведення аналізу гетероскедастичних процесів в економіці та фінансах.

Відповідно цьому варто обирати і систему показників якості програмного продукту.

Технічні вимоги до продукту наступні:

- програмний продукт повинен функціонувати на персональних комп'ютерах із стандартним набором компонент;

- забезпечувати високу швидкість обробки великих об'ємів даних у реальному часі;
- забезпечувати зручність і простоту взаємодії з користувачем або з розробником програмного забезпечення у випадку використання його як модуля;
- передбачати мінімальні витрати на впровадження програмного продукту.

### 5.2.1 Обґрунтування функцій програмного продукту

Головна функція F0 – розробка програмного продукту, керує ходом роботи.

Виходячи з конкретної мети, можна виділити наступні основні функції ПП:

F1 – вибір мови програмування;

F2 – створення моделі;

F3 – вибір кількості ланок кожної ноги.

Кожна з основних функцій може мати декілька варіантів реалізації.

Функція F1:

а) мова програмування C#;

б) мова програмування JavaScript.

Функція F2:

а) створення 3D моделі з фіксуванням моделі по осі Oz;

б) створення 2D моделі.

Функція F3:

а) 3 ланки кожної ноги;

б) 2 ланки кожної ноги.

### 5.2.2 Варіанти реалізації основних функцій

Варіанти реалізації основних функцій наведені у морфологічній карті системи (рис. 5.1). На основі цієї карти побудовано позитивно-негативну матрицю варіантів основних функцій (таблиця 5.1).



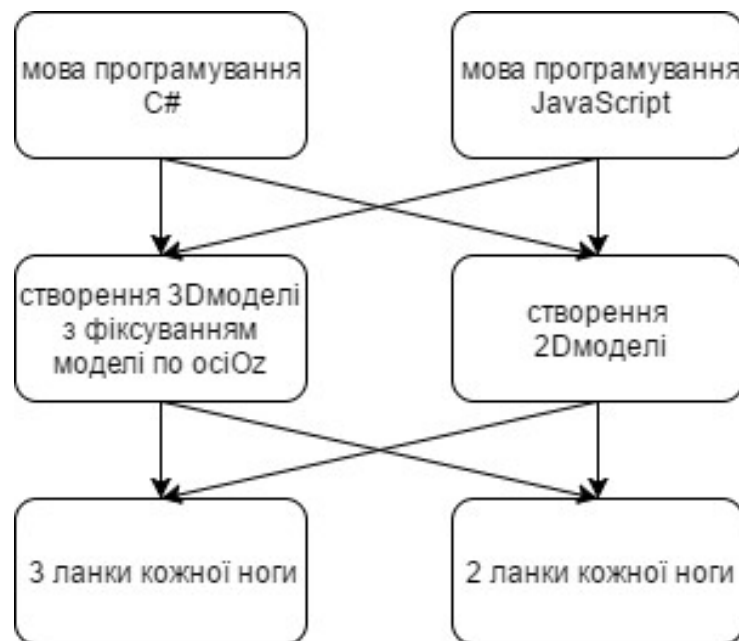


Рисунок 5.1 – Морфологічна карта

Морфологічна карта відображує всі можливі комбінації варіантів реалізації функцій, які складають повну множину варіантів ПП.

Таблиця 5.1 – Позитивно-негативна матриця

Основні функції	Варіанти реалізації	Переваги	Недоліки
<i>F1</i>	<i>A</i>	Гарно документована, легка у застосуванні, відома, краще підходить для великих проектів	–
	<i>B</i>	Краще підходить для малих проектів	Необхідність вивчення
<i>F2</i>	<i>A</i>	Можливість перегляду з більшої кількості ракурсів та простота реалізації	Потрібно фіксувати змінні, які змінюються вздовж вісі Oz.
	<i>B</i>	Немає зайвої вісі Oz	Більш складна реалізація та застосування
<i>F3</i>	<i>A</i>	Більша налагодженість ходи	Більше змінних параметрів
	<i>B</i>	Менше змінних параметрів	Менша налагодженість ходи

На основі аналізу позитивно-негативної матриці робимо висновок, що при розробці програмного продукту деякі варіанти реалізації функцій варто відкинути, тому, що вони не відповідають поставленим перед програмним продуктом задачам. Ці варіанти відзначені у морфологічній карті.

Функція *F1*:

Оскільки проект є великим доцільніше використовувати варіант а), до того ж недоцільно спеціально вивчати іншу мову програмування, тому варіант б) має бути відкинутий.

Функція *F2*:

Оскільки метою даної роботи є розробка системи керування рухом крокуючого робота, буде доцільна спостерігати ходу з багатьох ракурсів, а основні недоліки варіанта а) легко контролюються та усуваються, що в свою

чергу робить перевагу варіанту б) менш вагомою ніж варіанта а) , тому варіант б) має бути відкинутий

#### Функція F3:

Обидві варіанти кількості ланок є доцільними для застосування, тому вважаємо варіанти а) та б) гідними розгляду.

Таким чином, будемо розглядати такі варіанти реалізації ПП:

1. F1a – F2a – F3a
2. F1a – F2a – F3б

Для оцінювання якості розглянутих функцій обрана система параметрів, описана нижче.

### **5.3 Обґрунтування системи параметрів ПП**

#### **5.3.1 Опис параметрів**

На підставі даних про основні функції, що повинен реалізувати програмний продукт, вимог до нього, визначаються основні параметри виробу, що будуть використані для розрахунку коефіцієнта технічного рівня.

Для того, щоб охарактеризувати програмний продукт, будемо використовувати наступні параметри:

- X1 – швидкодія програми;
- X2 – об'єм пам'яті для збереження даних;
- X3 – точність результату;
- X4 – потенційний об'єм програмного коду;
- X5 – час розрахунку значення функції.

X1: Відображає швидкодію операцій залежно від обраної евристики.

X2: Відображає об'єм пам'яті в оперативній пам'яті персонального комп'ютера, необхідний для збереження та обробки даних під час виконання програми.

X3: Відображає на скільки велике відхилення ходи від ідеальної.

X4: Показує розмір програмного коду який необхідно створити безпосередньо розробнику.

X5: Відображає час, який витрачається на розрахунок значення функцій.

### 5.3.2 Кількісна оцінка параметрів

Гірші, середні і кращі значення параметрів вибираються на основі вимог замовника й умов, що характеризують експлуатацію ПП як показано у табл. 5.2.

Таблиця 5.2 – Основні параметри ПП

Назва Параметра	Умовні позначення	Одиниці виміру	Значення параметра		
			гірші	середні	кращі
Швидкодія програми	X1	мс	6000	4200	800
Об'єм пам'яті для збереження даних	X2	Мб	32	16	8
Точність результату	X3	%	10	5	0
Потенційний об'єм програмного коду	X4	кількість строк коду	2000	1000	500
Час розрахунку значення функції	X5	мкс	200	70	20

За даними таблиці 5.2 будуються графічні характеристики параметрів – рис. 5.2 – рис. 5.6.

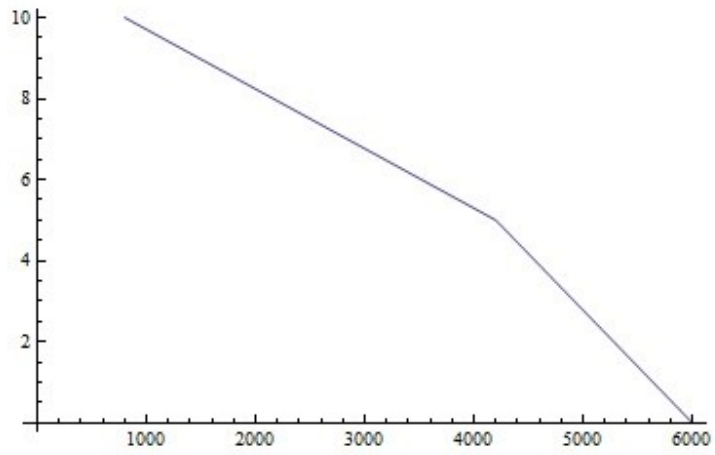


Рисунок 5.2 – X1, швидкодія програми

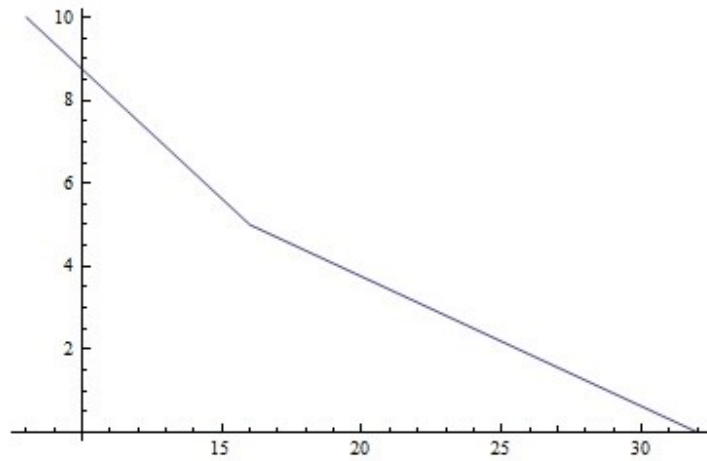


Рисунок 5.3 – X2, об'єм пам'яті для збереження даних

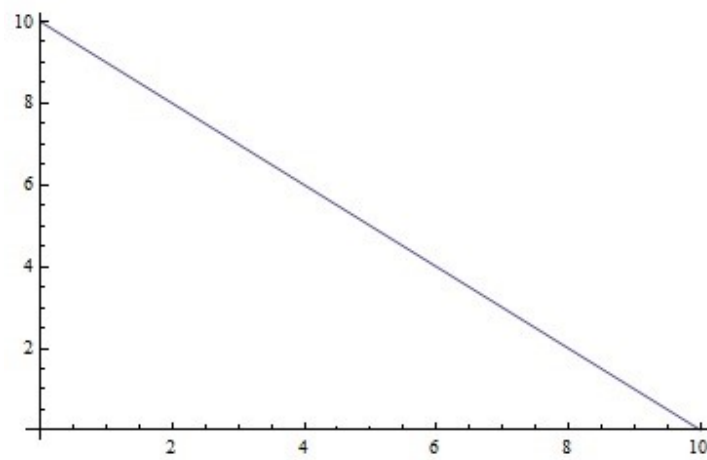


Рисунок 5.4 – X3, точність результату

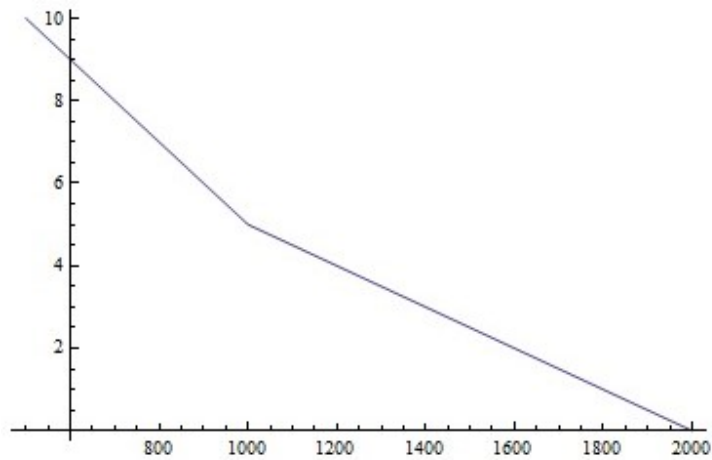


Рисунок 5.5 – X4, потенційний об'єм програмного коду

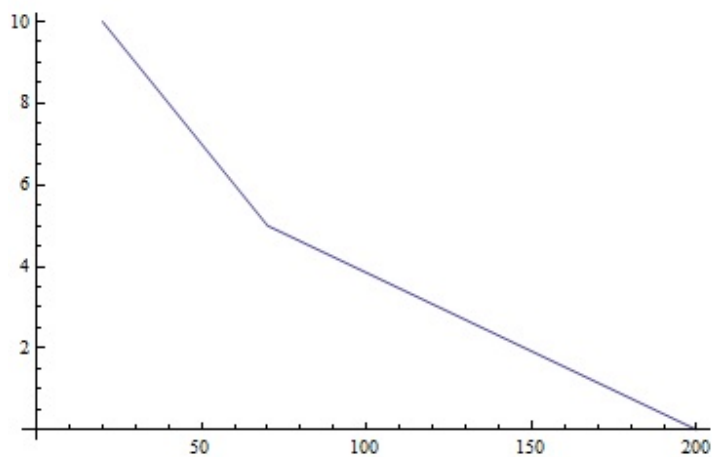


Рисунок 5.6 – X5, час розрахунку значення функції

### 5.3.3 Аналіз експертного оцінювання параметрів

Після детального обговорення й аналізу кожний експерт оцінює ступінь важливості кожного параметру для конкретно поставленої цілі – розробка програмного продукту, який дає найбільш точні результати при знаходженні параметрів моделей адаптивного прогнозування і обчислення прогнозних значень.

Значимість кожного параметра визначається методом попарного порівняння. Оцінку проводить експертна комісія із 7 людей. Визначення коефіцієнтів значимості передбачає:

- Визначення рівня значимості параметра шляхом присвоєння різних рангів;
- Перевірку придатності експертних оцінок для подальшого використання;
- Визначення оцінки попарного пріоритету параметрів;
- Обробку результатів та визначення коефіцієнту значимості.
- Результати експертного ранжування наведені у таблиці 5.3.

Таблиця 5.3 – Результати ранжування параметрів

Позначення параметра	Назва параметра	Одиниці виміру	Ранг параметра за оцінкою експерта							Сума рангів в $R_i$	Відхилення $\Delta_i$	$\Delta_i^2$
			1	2	3	4	5	6	7			
X1	Швидкодія програми	мс	5	6	5	6	5	5	6	38	17	289
X2	Об'єм пам'яті для збереження даних	Мб	3	2	3	3	3	4	2	20	-1	1
X3	Точність результату	%	2	2	3	2	2	2	3	16	-5	25
X4	Потенційний об'єм програмного коду	кількість строк коду	3	2	2	2	3	2	1	15	-6	36
X5	Час розрахунку значення функції	мкс	2	3	2	2	2	2	3	16	-5	25
	Разом		15	15	15	15	15	15	15	105	0	376

Для перевірки степені достовірності експертних оцінок, визначимо наступні параметри:

- а) сума рангів кожного з параметрів і загальна сума рангів:

$$R_i = \sum_{j=1}^N r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 105,$$

де  $N=7$  – число експертів,  $n=5$  – кількість параметрів;

б) середня сума рангів:

$$T = \frac{1}{n} R_{ij} = 21.$$

в) відхилення суми рангів кожного параметра від середньої суми рангів:

$$\Delta_i = R_i - T$$

Сума відхилень по всіх параметрам повинна дорівнювати 0;

г) загальна сума квадратів відхилення:

$$S = \sum_{i=1}^N \Delta_i^2 = 376.$$

Порахуємо коефіцієнт узгодженості:

$$W = \frac{12S}{N^2(n^3 - n)} = \frac{12 \cdot 376}{7^2(5^3 - 5)} = 0,767 > W_k = 0,67$$

Ранжування можна вважати достовірним, тому що знайдений коефіцієнт узгодженості перевищує нормативний, котрий дорівнює 0,67.

Скориставшись результатами ранжирування, проведемо попарне порівняння всіх параметрів і результати занесемо у таблицю 5.4.



Таблиця 5.4 – Попарне порівняння параметрів

Параметри	Експерти							Кінцева оцінка	Числове значення
	1	2	3	4	5	6	7		
X1 і X2	>	>	>	>	>	>	>	>	1,5
X1 і X3	>	>	>	>	>	>	>	>	1,5
X1 і X4	>	>	>	>	>	>	>	>	1,5
X1 і X5	>	>	>	>	>	>	>	>	1,5
X2 і X3	>	=	=	>	>	>	<	>	1,5
X2 і X4	=	=	>	=	=	>	>	=	1
X2 і X5	>	<	>	>	>	>	<	>	1,5
X3 і X4	<	=	>	=	<	=	>	=	1
X3 і X5	=	<	>	=	=	=	=	=	1
X4 і X5	>	<	=	=	>	=	<	=	1

Числове значення, що визначає ступінь переваги  $i$ -го параметра над  $j$ -тим,  $a_{ij}$  визначається за формулою:

$$a_{ij} = \begin{cases} 1,5 & \text{при } X_i > X_j \\ 1,0 & \text{при } X_i = X_j \\ 0,5 & \text{при } X_i < X_j \end{cases}$$

З отриманих числових оцінок переваги складемо матрицю  $A = \| a_{ij} \|$ .

Для кожного параметра зробимо розрахунок вагомості  $K_{ei}$  за наступними формулами:

$$K_{ei} = \frac{b_i}{\sum_{i=1}^n b_i}, \text{ де } b_i = \sum_{j=1}^N a_{ij}.$$

Відносні оцінки розраховуються декілька разів доти, поки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%). На другому і наступних кроках відносні оцінки розраховуються за наступними формулами:

$$K_{vi} = \frac{b'_i}{\sum_{i=1}^n b'_i}, \text{ де } b'_i = \sum_{j=1}^N a_{ij} b_j.$$

Як видно з таблиці 5.5, різниця значень коефіцієнтів вагомості не перевищує 2%, тому більшої кількості ітерацій не потрібно.

Таблиця 5.5 – Розрахунок вагомості параметрів

Параметрих <sub>i</sub>	Параметрих <sub>j</sub>					Перша ітер.		Друга ітер.		Третя ітер	
	X1	X2	X3	X4	X5	$b_i$	$K_{Bi}$	$b_i^1$	$K_{Bi}^1$	$b_i^2$	$K_{Bi}^2$
X1	1,0	1,5	1,5	1,5	1,5	7,0	0,28	34	0,287	160,75	0,287
X2	0,5	1,0	1,5	1,0	1,5	5,5	0,22	25,5	0,215	120,25	0,215
X3	0,5	0,5	1,0	1,0	1,0	4,0	0,16	18,75	0,158	88,75	0,158
X4	0,5	1,0	1,0	1,0	1,0	4,5	0,18	21,5	0,181	101,5	0,181
X5	0,5	0,5	1,0	1,0	1,0	4,0	0,16	18,75	0,158	88,75	0,158
Всього:						25	1	118,5	1	560	1

#### 5.4 Аналіз рівня якості варіантів реалізації функцій

Визначаємо рівень якості кожного варіанту виконання основних функцій окремо.

Абсолютні значення параметрів X2(об'єм пам'яті для збереження даних) та X1 (швидкодія програми) відповідають технічним вимогам умов функціонування даного ПП.

Абсолютне значення параметра X3 (точність результату) обрано не найгіршим (не максимальним), тобто це значення відповідає або варіанту а) 800 мс або варіанту б) 80мс.

Коефіцієнт технічного рівня для кожного варіанта реалізації ПП розраховується так (таблиця 5.6):

$$K_K(j) = \sum_{i=1}^n K_{ei,j} B_{i,j},$$

де  $n$  – кількість параметрів;  $K_{ei}$  – коефіцієнт вагомості  $i$ -го параметра;  $B_i$  – оцінка  $i$ -го параметра в балах.

Таблиця 5.6 – Розрахунок показників рівня якості варіантів реалізації основних функцій ПП

Основні функції	Варіант реалізації функції	Абсолютне значення параметра	Бальна оцінка параметра	Коефіцієнт вагомості параметра	Коефіцієнт рівня якості
F1(X4)	А	500	6,8	0,181	1,231
F2(X3)	А	3	7,2	0,158	1,138
F3(X2,X1)	А	16	5	0,215	1,075
	Б	800	10	0,287	2,870

За даними з таблиці 5.6 за формулою

$$K_K = K_{TY}[F_{1k}] + K_{TY}[F_{2k}] + \dots + K_{TY}[F_{zk}],$$

визначаємо рівень якості кожного з варіантів:

$$K_{K1} = 1,231 + 1,138 + 1,075 = 3,221;$$

$$K_{K2} = 1,231 + 1,138 + 2,870 = 5,432.$$

Як видно з розрахунків, кращим є другий варіант, для якого коефіцієнт технічного рівня має найбільше значення.

## 5.5 Економічний аналіз варіантів розробки ПП

Для визначення вартості розробки ПП спочатку проведемо розрахунок трудомісткості.

Всі варіанти включають в себе два окремих завдання:

1. Розробка логічної частини програмного продукту;
2. Розробка візуальної частини програмного продукту.

Завдання 1 за ступенем новизни відноситься до групи А, завдання 2 – до групи Б. За складністю алгоритми, які використовуються в завданні 1 належать до групи 1; а в завданні 2 – до групи 3.

Для реалізації завдання 1 використовується довідкова інформація, а завдання 2 використовує інформацію у вигляді даних.

Проведемо розрахунок норм часу на розробку та програмування для кожного з завдань.

Проведемо розрахунок норм часу на розробку та програмування для кожного з завдань. Загальна трудомісткість обчислюється як

$$T_0 = T_P \cdot K_{\Pi} \cdot K_{СК} \cdot K_M \cdot K_{СТ} \cdot K_{СТ,М}, \quad (5.1)$$

де  $T_P$  – трудомісткість розробки ПП;  $K_{\Pi}$  – поправочний коефіцієнт;  $K_{СК}$  – коефіцієнт на складність вхідної інформації;  $K_M$  – коефіцієнт рівня мови програмування;  $K_{СТ}$  – коефіцієнт використання стандартних модулів і прикладних програм;  $K_{СТ,М}$  – коефіцієнт стандартного математичного забезпечення

Для першого завдання, виходячи із норм часу для завдань розрахункового характеру степеню новизни А та групи складності алгоритму 1, трудомісткість дорівнює:  $T_P = 90$  людино-днів. Поправочний коефіцієнт, який враховує вид нормативно-довідкової інформації для першого завдання:  $K_{\Pi} = 1.7$ . Поправочний коефіцієнт, який враховує складність контролю вхідної та вихідної інформації для всіх семи завдань рівний 1:  $K_{СК} = 1$ . Оскільки при розробці першого завдання використовуються стандартні модулі, врахуємо це за допомогою коефіцієнта  $K_{СТ} = 0.8$ . Тоді, за формулою 5.1, загальна трудомісткість програмування першого завдання дорівнює:

$$T_1 = 90 \cdot 1.7 \cdot 0.8 = 122.4 \text{ людино-днів.}$$

Проведемо аналогічні розрахунки для подальших завдань.

Для другого завдання (використовується алгоритм третьої групи складності, степінь новизни Б), тобто  $T_P = 27$  людино-днів,  $K_{\Pi} = 0.9$ ,  $K_{СК} = 1$ ,  $K_{СТ} = 0.8$ :

$$T_2 = 27 \cdot 0.9 \cdot 0.8 = 19.44 \text{ людино-днів.}$$

Складаємо трудомісткість відповідних завдань для кожного з обраних варіантів реалізації програми, щоб отримати їх трудомісткість:

$$T_I = (122.4 + 19.44 + 4.8 + 19.44) \cdot 8 = 1328,64 \text{ людино-годин};$$

$$T_{II} = (122.4 + 19.44 + 6.91 + 19.44) \cdot 8 = 1345.52 \text{ людино-годин};$$

Найбільш високу трудомісткість має варіант II.

В розробці беруть участь два програмісти з окладом 20000 грн., один фінансовий аналітик з окладом 25000 грн. Визначимо зарплату за годину за формулою:

$$C_{\text{ч}} = \frac{M}{T_m \cdot t} \text{ грн.},$$

де  $M$  – місячний оклад працівників;  $T_m$  – кількість робочих днів тиждень;  $t$  – кількість робочих годин в день.

$$C_{\text{ч}} = \frac{20000 + 20000 + 25000}{3 \cdot 21 \cdot 8} = 128,97 \text{ грн.}$$

Тоді, розрахуємо заробітну плату за формулою

$$C_{\text{зп}} = C_{\text{ч}} \cdot T_i \cdot K_{\text{д}},$$

де  $C_{\text{ч}}$  – величина погодинної оплати праці програміста;  $T_i$  – трудомісткість відповідного завдання;  $K_{\text{д}}$  – норматив, який враховує додаткову заробітну плату.

Зарплата розробників за варіантами становить:

$$I. \quad C_{\text{зп}} = 128,97 \cdot 1328,64 \cdot 1.2 = 205622,86 \text{ грн.}$$

$$II. \quad C_{\text{зп}} = 128,97 \cdot 1345.52 \cdot 1.2 = 208238,06 \text{ грн.}$$

Відрахування на єдиний соціальний внесок в залежності від групи професійного ризику (II клас) становить 22%:

$$I. \quad C_{\text{вд}} = C_{\text{зп}} \cdot 0,22 = 205622,86 \cdot 0,22 = 45237,03 \text{ грн.}$$

$$II. \quad C_{\text{вд}} = C_{\text{зп}} \cdot 0,22 = 208238,06 \cdot 0,22 = 45812,37 \text{ грн.}$$

Тепер визначимо витрати на оплату однієї машино-години. ( $C_M$ )

Так як одна ЕОМ обслуговує одного програміста з окладом 20000 грн., з коефіцієнтом зайнятості 0,2, то для однієї машини отримаємо:

$$C_{\Gamma} = 12 \cdot M \cdot K_3 = 12 \cdot 20000 \cdot 0,2 = 48000 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$C_{3П} = C_{Г} \cdot (1 + K_3) = 48000 \cdot (1 + 0.2) = 57600 \text{ грн.}$$

Відрахування на єдиний соціальний внесок:

$$C_{ВІД} = C_{3П} \cdot 0,22 = 57600 \cdot 0,22 = 12672 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 30000 грн.

$$C_A = K_{ТМ} \cdot K_A \cdot Ц_{ПР} = 1.15 \cdot 0.25 \cdot 30000 = 8625 \text{ грн.,}$$

де  $K_{ТМ}$  – коефіцієнт, який враховує витрати на транспортування та монтаж приладу у користувача;  $K_A$  – річна норма амортизації;  $Ц_{ПР}$  – договірна ціна приладу.

Витрати на ремонт та профілактику розраховуємо як:

$$C_P = K_{ТМ} \cdot Ц_{ПР} \cdot K_P = 1.15 \cdot 30000 \cdot 0.05 = 1725 \text{ грн.,}$$

де  $K_P$  – відсоток витрат на поточні ремонти.

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$$T_{ЕФ} = (D_K - D_B - D_C - D_P) \cdot t_3 \cdot K_B = (365 - 104 - 8 - 16) \cdot 8 \cdot 0.9 = 1706.4 \text{ годин,}$$

де  $D_K$  – календарна кількість днів у році;  $D_B$ ,  $D_C$  – відповідно кількість вихідних та святкових днів;  $D_P$  – кількість днів планових ремонтів устаткування;  $t$  – кількість робочих годин в день;  $K_B$  – коефіцієнт використання приладу у часі протягом зміни.

Витрати на оплату електроенергії розраховуємо за формулою:

$$C_{ЕЛ} = T_{ЕФ} \cdot N_C \cdot K_3 \cdot Ц_{ЕН} = 1706,4 \cdot 1,94 = 3310,42 \text{ грн.,}$$

Де  $N_C$  – середньо-споживча потужність приладу;  $K_3$  – коефіцієнт зайнятості приладу;  $Ц_{ЕН}$  – тариф за 1 КВт-годин електроенергії.

Накладні витрати розраховуємо за формулою:

$$C_H = Ц_{ПР} \cdot 0.67 = 30000 \cdot 0,67 = 20100 \text{ грн.}$$

Тоді, річні експлуатаційні витрати будуть:

$$C_{ЕКС} = C_{3П} + C_{ВІД} + C_A + C_P + C_{ЕЛ} + C_H$$

$$C_{ЕКС} = 57600 + 12672 + 8625 + 1725 + 3310,42 + 20100 = 104032,42 \text{ грн.}$$

Собівартість однієї машино-години ЕОМ дорівнюватиме:

$$C_{M-Г} = C_{EКС} / T_{EФ} = 104032,42 / 1706,4 = 60,97 \text{ грн/час.}$$

Оскільки в даному випадку всі роботи, які пов'язані з розробкою програмного продукту ведуться на ЕОМ, витрати на оплату машинного часу, в залежності від обраного варіанта реалізації, складає:

$$C_M = C_{M-Г} \cdot T$$

$$I. \quad C_M = 60,97 \cdot 1328,64 = 81007,18 \text{ грн.};$$

$$II. \quad C_M = 60,97 \cdot 1345,52 = 82036,35 \text{ грн.}$$

Накладні витрати складають 67% від заробітної плати:

$$C_H = C_{ЗП} \cdot 0,67$$

$$I. \quad C_H = 205622,86 \cdot 0,67 = 137767,32 \text{ грн.};$$

$$II. \quad C_H = 208238,06 \cdot 0,67 = 139519,50 \text{ грн.};$$

Отже, вартість розробки ПП за варіантами становить:

$$C_{ПП} = C_{ЗП} + C_{Вид} + C_M + C_H$$

$$I. \quad C_{ПП} = 205622,86 + 45237,03 + 81007,18 + 137767,32 = 469634,39 \text{ грн.}$$

$$II. \quad C_{ПП} = 208238,06 + 45812,37 + 82036,35 + 139519,50 = 475606,28 \text{ грн.}$$

## 5.6 Вибір кращого варіанта ПП техніко-економічного рівня

Розрахуємо коефіцієнт техніко-економічного рівня за формулою:

$$K_{TEPj} = K_{Кj} / C_{Фj},$$

$$K_{TEP1} = 5,432 / 469634,39 = 1,15 \cdot 10^{-5};$$

$$K_{TEP2} = 3,221 / 475606,28 = 0,68 \cdot 10^{-5}.$$

Як бачимо, найбільш ефективним є перший варіант реалізації програми з коефіцієнтом техніко-економічного рівня  $K_{TEP1} = 1,15 \cdot 10^{-5}$ .

## 5.7 Висновок

В даному розділі проведено повний функціонально-вартісний аналіз ПП, який було розроблено в рамках дипломного проекту. Процес аналізу можна умовно розділити на дві частини.

В першій з них проведено дослідження ПП з технічної точки зору: було визначено основні функції ПП та сформовано множину варіантів їх реалізації; на основі обчислених значень параметрів, а також експертних оцінок їх важливості було обчислено коефіцієнт технічного рівня, який і дав змогу визначити оптимальну з технічної точки зору альтернативу реалізації функцій ПП.

Другу частину ФВА присвячено вибору із альтернативних варіантів реалізації найбільш економічно обґрунтованого. Порівняння запропонованих варіантів реалізації в рамках даної частини виконувалось за коефіцієнтом ефективності, для обчислення якого були обчислені такі допоміжні параметри, як трудомісткість, витрати на заробітну плату, накладні витрати.

Після виконання функціонально-вартісного аналізу програмного комплексу що розроблюється, можна зробити висновок, що з альтернатив, що залишились після першого відбору двох варіантів виконання програмного комплексу оптимальним є перший варіант реалізації програмного продукту. У нього виявився найкращий показник техніко-економічного рівня якості  $K_{\text{TEP1}} = 1,15 \cdot 10^{-5}$ .

Цей варіант реалізації програмного продукту має такі параметри:

Мова програмування – C#;

створення 3Dмоделі з фіксуванням моделі по осі Oz;

3 ланки кожної ноги.

Даний варіант виконання програмного комплексу дає користувачу необхідну швидкодію без втрат точності та зміни кількості коду.



## ВИСНОВКИ

Бакалаврська робота присвячена розгляду математичних моделей, розробці методик, алгоритмів, засобів комп'ютерного моделювання ходи АКА з активними та пасивними приводами, що ґрунтуються на даних експериментальних досліджень ходи людини.

У ході виконання роботи розглянуто існуючі та побудовані власні математичної моделі АКА. Розроблено методики і числові алгоритми для розв'язання задачі керування рухом АКА на основі експериментальних біомеханічних досліджень. Створено програмне забезпечення для проведення числових досліджень та аналізу результатів математичного моделювання і оптимізації законів руху АКА.

Також, порівняно фази ходи моделей крокуючого апарата виконані в середовищі MATLAB Simulink з виконаними в середовищі Unity. Всі представлені фази ходи моделі крокуючого апарата виконані в середовищі MATLAB Simulink повторені моделлю крокуючого апарата в середовищі Unity.

У роботі отримано такі основні результати:

- Уточнено модель АКА, яка враховує природну ритміку руху ніг, кінематичні та динамічні обмеження, побудовані на підставі експериментальних біомеханічних досліджень ходи людини.
- Розроблено числово-аналітичну методику та алгоритми комп'ютерного моделювання та оптимізації ходи АКА в середовищі MATLAB, Simulink із використанням компонентів з бібліотеки SimMechanics.
- методику візуалізації та аналізу ритмічних, кінематичних, динамічних, енергетичних характеристик руху АКА в середовищі Unity.

Сукупність отриманих результатів дозволяє ставити та розв'язувати практично важливі задачі комп'ютерної імітації руху антропоморфних крокуючих систем (людини, як в нормі, так і з протезованою ногою, та двоногого крокуючого робота з пасивними приводами в ступнях ніг).

Розроблені числово-аналітичні методики, алгоритми та програми дозволяють розраховувати динамічні і енергетичні характеристики ходи людини на основі гоніометричних даних експериментальних досліджень і використовувати їх в реабілітаційних технологіях, наприклад, в протезуванні ніг людини.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Бармак О. В. Інформаційна технологія розпізнавання емоційної міміки на обличчі людини/ Бармак О. В., Ю. В. Крак, Г. М. Єфімов// Штучний інтелект. — 2008. — № 1. — С. 102-109.
2. Белецкий В. В. Двухногая ходьба: Модельные задачи динамики и управления. / Белецкий В. В.//— Москва : Наука. — 1994. — 286 с.
3. Белецкий В. В. Параметрическая оптимизация движений двухногого шагающего аппарата/ Белецкий В. В., Бербюк В. Е., Самсонов В. А // Изв. АН СССР, МТТ. — 1982. — №1. — С. 28–40.
4. Белецкий В. В. Двухногая ходьба: модельные задачи динамики и управления. / Белецкий В. В.// — Москва : Наука, 1984. — 228с.
5. Бербюк В. Е. Динамика и оптимизация робототехнических систем./ Бербюк В. Е.// — Киев : Наукова думка, 1989. — 187 с.
6. Бернштейн Н. А. О построении движений./ Бернштейн Н. А.// — Москва, 1967.
7. Болотин Ю. В. Управление походкой двухногого шагающего аппарата / Болотин Ю. В., Новожилов И. В.// Изв. АН СССР. МТТ. — 1977. — №3. — С. 47–52.
8. Бордюг Б. А. Задачи управления шагающими аппаратами./ Бордюг Б. А., Ларин В. Б., Тимошенко А.Г.// — Киев : Наукова думка, 1985.— 264 с.
9. Бордюг Б. А. Моделирование движения электромеханического шагающего аппарата/ Бордюг Б. А., Ларин В. Б. // Проблемы управления и информатики. — 1995. — № 4. — С. 115–119.
10. Брискин Е. С. О минимизации энергозатрат при движении шагающей машины с цикловыми движителями / Е. С. Брискин, Я.В. Калинин // 2009.

- 11.Блехерман М. Х. Оперативно-производственное планирование гибких производственных систем./ Блехерман М. Х. // - М.: Высш. шк., 1989.- 95 с.
- 12.Величенко В. В, Матрично-геометрические методы в механике с приложениями к задачам робототехники, / Величенко В. В //1988
- 13.Витензон А. С. Динамические фазы цикла ходьбы / Витензон А. С.// Биомеханика.– Рига, 1975. – С. 251–257.
- 14.Вукобратович М. Шагающие роботы и антропоморфные механизмы./ Вукобратович М. // – Москва : Мир, 1976. – 543 с.
- 15.Голубничий Н. И. Беседы по автоматике / Н. И. Голубничий, Г. Ф. Зайцев, М. А. Иващенко, П. И. Чумаков // Київ : “Техніка”, 1971. – 232 с.
- 16.Гриценко Г. П. Энергетическая оценка ходьбы человека в норме и на протезах бедра / Гриценко Г. П., Морейнис И. Ш // Биомеханика. Рига, 1975.– С. 228–232.
17. Девянин Е. А. Задача управления движением шагающего аппарата / Девянин Е. А., Ленский А. В., Самсонов В. А.// Биомеханика. Рига: 1975.– С. 640–643.
- 18.Довбня Н. М. Роботизированные технологические комплексы в ГПС./ Довбня Н. М., Кондратьев А. Н., Юревич Е. И.// - Л.: Машиностроениеб, 1990. – 303 с.
- 19.Иванов Г. Н. Применение ипи-технологий для проектирования многофункциональных гибких производственных систем (гпс)/ Иванов Г. Н. , Тимошенко Ю. Н. //, 2002.
- 20.Кириченко Н. Ф. Кинематические и динамические модели мехатронных систем / Кириченко Н. Ф., Сопронюк Ф. А. // Проблемы управления и информатики. – Киев. 1995, №6. – С. 116–127.
- 21.Кисленко Ю. И. Тактильное очувствление шагающего робота / Кисленко Ю. И. //Техническая кибернетика, 1991. – № 6. – С. 62–50.

- 22.Коровин Б. Г. Системы программного управления промышленными установками и робототехническими комплексами./ Коровин Б. Г., Прокофьев Г. И., Рассудов Л. Н. // - Л.: Энергоатомиздат, 1990. - 352 с.
- 23.Крак Ю. В. Координационный подход к организации движения манипуляционных роботов / Крак Ю. В. // Проблемы управления и информатики. – 1995. – № 4. – С. 120–128.
- 24.Красовский Н. Н. Теория управления движением./ Красовский Н. Н. // – Москва : Наука, 1968. – 478 с.
- 25.Лавровский Э. К. Динамика двуногой ходьбы при больших скоростях движения / Лавровский Э. К. // Изв. АН СССР. МТТ.– 1980.– №4.– С. 50–58.
- 26.Ларин В. Б. Управление шагающими аппаратами./ Ларин В. Б. // – Киев : Наукова думка, 1980.– 168с.
- 27.Маркеев А. П. Теоретическая механика./ Маркеев А. П. // – Москва : Наука, 1990. – 416 с.
- 28.Охоцимский Д. Е. Механика и управление движением автоматического шагающего аппарата./ Охоцимский Д. Е., Голубев Ю. Ф.//– Москва : Наука, 1984.– 312 с.
- 29.Скворцов Д. В. Клинический анализ движений. Анализ походки./ Скворцов Д. В. // – Иваново :Издательство НППЦ “Стимул”, 1996. – 344 с.
- 30.Скопеецкий В. В. Математичне моделювання прямих та обернених задач динаміки систем з розподіленими параметрами./ Скопеецкий В. В., Стоян В. А., Кривонос Ю. Г.// – Київ : Наукова думка, 2002.– 361 с.
- 31.Тимофеев А. В. Управление роботами./ Тимофеев А. В. // – Ленинград : Ленинградского университета, 1986. – 240 с.
- 32.Ткач М. М. Оптимізація управління запасами матеріальних ресурсів на підприємстві / М. М. Ткач, Д. О. Гуменний // , 2014.

33. Abba G. Robot dynamic modeling using a power flow approach with application to biped locomotion / G. Abba, N. Chaillet // Williams, 1999. – 660 p.
34. Adolfsson J. 3D passive walkers: finding periodic gaits in the presence of discontinuities / J. Adolfsson, H. Dankowicz, A. Nordmark // Springer, 2001. – 205– 229 p.
35. Aguiar A. P. Path-following for nonminimum phase systems removes performance limitations / Aguiar A. P. // IEEE Transaction and Automation vol.50 iss.2, 2005. – 234–239 p.
36. Ahmadi M. Stable control of a simulated one-legged running robot with hip and leg compliance / M. Ahmadi, M. Bhlér // IEEE Transaction and Robotics Vol.13 iss.1, 1997. – 96–104 p.
37. Aldebaran Robotics. Humanoid robotics & programmable robots [Электронный ресурс] . –Режим доступа: <http://www.aldebaran.com/en> – Дата доступа : 16.04.2017.
38. Ames A. D. Stably extending two-dimensional bipedal walking to three / Aaron D. Ames, Robert D. Gregg // IEEE American Control Conference, New York, NY, 2007. – 2848–2854 p.
39. Arvidson R. Results From The Mars Phoenix Lander Robotic Arm Experiment / R. Arvidson, R. Bonitz, M. Robinson, J. Carsten, R. Volpe, A. Trebi-Ollennu, et al. // Journal of Geophysical Research, vol. 114, 2009.
40. Bainov D. D. Systems with Impulse Effects : Stability, Theory and Applications / D. D. Bainov, P. S. Simeonov // Chichester, Ellis Horwood Limited / John Wiley & Sons, 1989. – 255 p.
41. Banks S. P. Control Systems Engineering / Banks S. P. // Pentoce Hall, Englewood Cliffs, 1986.
42. Butterfa. DLR-Hand II: Next generation of a dextrous robot hand. / Butterfa., Jorg, Markus Grebenstein, Hong Liu, and Gerd Hirzinger.// In Robotics and

- Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on, vol. 1, pp. 109-114. IEEE, 2001.
43. Benallegue M. Contribution of Actuated Head and Trunk to Passive Walkers Stabilization/ Benallegue M., Laumond J-P. and Berthoz A., //, 2010
  44. Berbyuk V. E. Dynamics and optimal control problems for biotechnical system “Man-Prosthesis” / Berbyuk V. E. // IUTAM Symposium of Interaction Between Dynamics Control in Advanced Mechanical Systems, D. H. van Kampen editor, The Netherlands, Kluwer Academic Publishers.– 1996.– P. 28–35.
  45. Berbyuk V., Optimization of Control Laws of the Bipedal Locomotion Systems / Berbyuk V., Bostrom A., Lytwyn B., Peterson Bo. // Advances in Computational Multibody Dynamics / J. Ambrosio, W. Schiehlen (eds.), IDMEC/IST. – Lisbon (Portugal). – 1999. – P. 713–728.
  46. Bhat S. P. Finite-time stability of continuous autonomous systems/ Bhat S. P. // – SIAM Journal on Control and Optimisation vol.38 no.3, 2006. – 751–766 p.
  47. Brenner R. The Origins of Capitalist Development: a Critique of Neo-Smithian Marxism / R. Brenner// – UK : 1977. – 204 p.
  48. Stoica, A. (2008, December). Developing humanoid robots for real-world environments. In Humanoid Robots/ Stoica, A., Kuhlman, M., Assad, C., & Keymeulen, D.//, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on (pp. 567-572). IEEE.
  49. Buher M. A family of robotucontrol strategies for intermittent dynamical environments / M. Buher, D. E. Koditschek, P. J. Kindlmann // IEEE Robotics and Automation, vol.3, 1989. – 1296–1301 p.
  50. Chemori A control architecture with stabilizer for 3D stable dynamic walking of SHERPA biped robot on compliant ground./ Chemori, Ahmed// Humanoids' 10: 10th International Conference on Humanoid Robots. IEEE-RAS, 2010.

51. Cheng Y. Landmark Based Position Estimation for Pinpoint Landing on Mars / Y. Cheng A. Ansar // Mobility Systems Concept Development Section Jet Propulsion Laboratory, 2005
52. Chesse S. Optimal dynamics of constrained multibody systems. Application to bipedal walking synthesis/ Chesse S., Bessonnet G. // Proc. of 2001 IEEE Int. conf on Robotics & Automation. – 2001. P. 2499–2505.
53. Chow C. K. Studies of human locomotion via optimal programming /Chow C. K., Jacobson D. H. // Mathem. Biosciences. – 1971. – Vol. 10, №.3/4.– P. 239–306.
54. Dubovik A. After Makhno: The Anarchist Underground in the Ukraine in the 1920s and 1930s: Outlines of History & The Story of a Leaflet and The Fate of the Anarchist Varshavskiy (From the History of Anarchist Resistance to Totalitarianism) / A. Dubovik, D. I. Rublyov // Kate Sharpley Library, 2009. – 288 p.
55. Espiau B. BIP: a joint project for the development of an anthropomorphic biped robot / Espiau B. //IEEE International Conference on Robotics & Automation, 2000. – P. 3996– 4001.
56. Frank L. Robot Manipulator Control. Theory and Practice / L. Frank, T. Abdallah, D. Dawson // USA, NY. : “Marcel Dekker Inc.”, 2006. – 614 p.
57. Fujimoto Y. Simulation of an autonomous bipedwalking robot including environmental force interaction / Y. Fujimoto and A. Kawamura // 1998.
58. Furusho J. Sensor-based control of a nine-link biped / Furusho J. , Sano A. // Int. J. Robot. Res.1990. Vol. 9, No 2. P. 83–98.
59. Hardt M. Optimal biped walking with a complete dynamic model / Hardt M., Kreuz-Delgado K., Helton J. W. // Proc. 38th conference on decision and control, Phoenix AZ, USA, December 7–10 (IEEE 1999). – 1999. – P. 2999–3004.



60. Hardt M. Dynamic modeling in the simulation, optimization, and control of bipedal and quadrupedal robots / Hardt M., von Stryk O. // ZAMM. Z. angew. Math. Mech. 2003. – Vol. 84, – No. 10. – P. 1–16.
61. Larin V. B. Problems of control of a hopping apparatus/ Larin V. B. // J. Franklin Inst. – 1998. – Vol. 335B, №3. – P. 579–593.
62. McGeer T. Passive dynamic walking/ McGeer T. // Int. J. Robot Res.–1990.– Vol. 9.– P. 62–82.
63. McGeer T. Passive walking with knees' / McGeer T. // Proc. of IEEE conf. on Robotics & Automation.–Cincinnati.–1990.– P. 1640–1645.
64. Hill, J. Active disturbance rejection for walking bipedal robots using the acceleration of the upper limbs./ Hill, J., & Fahimi, F. // Robotica, 33(02), 264-281.
65. Morecki A. Human and Machine Locomotion, / Morecki A. and Waldron // CISM Courses and Lectures No. 375, Springer-Verlag, Wien New York (1997).
66. Muybridge, Eadweard. Animal locomotion. New York: Da Capo Press, 1969.
67. Nishiwaki K. Toe joints that enhance bipedal and fullbody motion of humanoid robots / Nishiwaki K., Kagami S., Kuniyoshi Y., Inaba M., Inoue H. // Proc. of 2002 IEEE int. conf. on Robotics & Automation. 2002. P. 3105–3110. 210
68. Van Oort An energy efficient knee locking mechanism for a dynamically walking robot./ Van Oort, G., Carloni, R., Borgerink, D. J., & Stramigioli, S. // Robotics and Automation (ICRA), 2011 IEEE International Conference on (pp. 2003- 2008). IEEE.
69. Ott C. Prioritized Multi-Task Compliance Control of Redundant Manipulators / Christian Ott, Alexander Dietrich, and Alin Albu-Schaffer // Elsevire, Automatica, vol. 53, 2015. – P. 416–423.

70. Ott, C. (2011, September). Bipedal walking control based on capture point dynamics/ Ott, C., Roa, M. A., Albu-Schaffer, A., & Hirzinger, G.//. Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on (pp. 4420-4427). IEEE.
71. Sardain P. An anthropomorphic biped robot: dynamic concepts and technological design / Sardain P., Rostami M., Bessonnet G.// IEEE trans. on Systems, Man, and cybernetics - part A: systems and humans. 1998. Vol. 28, No. 6. P. 823–838.
72. Schuessler J. History Departments, It's Up With Capitalism / J. Schuessler// – USA, DC : 2013. – 102 p.
73. Shoemake K. Animating Rotation with Quaternion Curves Proc. / K. Shoemake // SIG-GRAPH, 1985, P. 245–254.
74. Tomović. Sensitivity analysis of dynamic systems/ Tomović, Rajko // McGraw-Hill, 1963.
75. Taga G. A model of the neuro-musculo-skeletal system for anticipatory adjustment of human locomotion during obstacle avoidance / Taga G. // Biological Cybernetics. – 1998.– Vol. 78, №1.– P. 9–17.
76. Travis Deyle. KATE: A New Humanoid Robot from FutureBots Labs [Электронный ресурс]. – Режим доступа: <http://www.hizook.com/blog/2012/04/16/kate-new-humanoid-robot-futurebots-labs> – Дата доступа : 29.03.2017.
77. Volpe R. 2014 Robotics Activities at JPL - International Symposium on Artificial Intelligence,/ Volpe R. // Robotics and Automation in Space (iSAIRAS), Montreal, Canada, 17 June 2014.
78. Volpe R. Rover Functional Autonomy Development for the Mars Mobile Science Laboratory / Volpe R. //IEEE Aerospace Conference, 2003. – P. 2643–2652.

79. Von Stryk O. Direct and indirect methods for trajectory optimization / Von Stryk O., Bulirsch R. // *Annals of operational research*. 1992. Vol. 37. P. 357–373.
80. Wollher D. Actuator selection and hardware realization of a small and fast-moving, autonomous humanoid robot / Wollher D., Hardt M., Buss M., von Stryk O. // *Proc. of the 2002 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Lausanne, Switzerland. September 30–October 4, 2002. P. 2491–2496.
81. Zielińska T. Coupled oscillators utilised as gait rhythm generators of a two-legged walking machine / Zielińska T. // *Biological Cybernetics*. – 1996. – Vol. 74, No3. – P. 263–273.