



การใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm
ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน



วิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตาม
หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมชีวการแพทย์
วิทยาลัยวิศวกรรมชีวการแพทย์

บัณฑิตวิทยาลัย มหาวิทยาลัยรังสิต

ปีการศึกษา 2567



**USING ZIEGLER-NICHOLS-ASSISTED GENETIC ALGORITHM FOR
TUNING THE CONTROL SYSTEM OF EXOSKELETON ROBOT**

BY

SIWAKORN PANKRACHANG

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR

THE DEGREE OF MASTER OF ENGINEERING IN

BIOMEDICAL ENGINEERING

COLLEGE OF BIOMEDICAL ENGINEERING

GRADUATE SCHOOL, RANGSIT UNIVERSITY

ACADEMIC YEAR 2024

วิทยานิพนธ์เรื่อง

การใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการ
ปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน

โดย

ศิวกร ปั่นกระจำง

ได้รับการพิจารณาให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมชีวการแพทย์

มหาวิทยาลัยรังสิต

ปีการศึกษา 2567

ศ. ดร. ชูชาติ ปิ่นทิวรัตน์
ประธานกรรมการสอบ

ผศ. ว่าที่ ร.ต. ดร. พิชิตพล โชติกุลนันท์
กรรมการ

ผศ. ดร. ยุทธนา ปิติธีรภาพ
กรรมการและอาจารย์ที่ปรึกษา

บัณฑิตวิทยาลัยรับรองแล้ว

(ศ. ดร. สৌจิตต์ เพ็ชรประสาน)

คณบดีบัณฑิตวิทยาลัย

27 สิงหาคม 2567

Thesis entitled

**USING ZIEGLER-NICHOLS-ASSISTED GENETIC ALGORITHM FOR TUNING
THE CONTROL SYSTEM OF EXOSKELETON ROBOT**

by

SIWAKORN PANKRACHANG

was submitted in partial fulfillment of the requirements
for the degree of Master of Engineering in Biomedical Engineering

Rangsit University
Academic Year 2024

Prof. Chuchart Pintavirooj, Ph.D.
Examination Committee Chairperson

Asst. Prof. Acting Sub LT.
Phichitphon Chotikunnan, D.Eng.
Member

Asst.Prof. Yutthana Pititheeraphab, Ph.D.
Member and Advisor

Approved by Graduate School

(Prof. Suejit Pechprasarn, Ph.D.)

Dean of Graduate School

August 27, 2024

กิตติกรรมประกาศ

วิทยานิพนธ์เรื่องการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดินสำเร็จตามวัตถุประสงค์ได้เนื่องมาจากได้รับความอนุเคราะห์จากผู้ช่วยศาสตราจารย์ ดร. ยุทธนา ปิติธีรภาพ อาจารย์ที่ปรึกษา ที่ได้ให้คำแนะนำและคำปรึกษาในการทำวิทยานิพนธ์มาตลอด

ผู้จัดทำวิทยานิพนธ์ขอขอบพระคุณ รองศาสตราจารย์ ดร. มนัส สังวรศิลป์ และผู้ช่วยศาสตราจารย์ ว่าที่ ร.ต. ดร. พิชิตพล โชติกุลนันท์ ที่ให้คำปรึกษาและคำแนะนำในการทำวิทยานิพนธ์จนสำเร็จลุล่วงไปได้ด้วยดี

ผู้จัดทำขอขอบพระคุณบิดาและมารดา ที่คอยให้กำลังใจ การสนับสนุน และมอบโอกาสให้ผู้จัดทำมีโอกาสในการศึกษาในระดับวิศวกรรมศาสตรมหาบัณฑิต วิทยาลัยวิศวกรรมชีวการแพทย์ จนสำเร็จลุล่วงได้ด้วยดี

สุดท้ายนี้ ความรู้และประโยชน์ที่ได้รับจากการทำวิทยานิพนธ์ฉบับนี้ ผู้จัดทำขอมอบความดีที่ได้ให้แก่ผู้มีพระคุณทุกคน

ศิวกร ปั่นกระจำง

ผู้วิจัย

6609892 : ศิวกร ปั่นกระจำง
 ชื่อวิทยานิพนธ์ : การใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm
 ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน
 หลักสูตร : วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมชีวการแพทย์
 อาจารย์ที่ปรึกษา : ผศ. ดร. ยุทธนา ปิติธีรภาพ

บทคัดย่อ

ระบบควบคุมแบบ PID เป็นระบบควบคุมที่นิยมใช้ในการควบคุมการเคลื่อนที่ของหุ่นยนต์ เนื่องจากเป็นระบบควบคุมที่มีความน่าเชื่อถือและเรียบง่ายในการใช้งาน การปรับตั้งค่าระบบควบคุม PID ในหุ่นยนต์ที่มีโครงสร้างซับซ้อน เช่น หุ่นยนต์สำหรับร่างกายที่อ่อนแอให้ทำงานได้เหมาะสมด้วยวิธีการทั่วไปนั้นมีความยุ่งยากและขาดความแม่นยำ จึงมีการใช้ Algorithm เช่น Genetic Algorithm เข้ามาช่วยในการปรับตั้งค่าระบบควบคุม การใช้ Genetic Algorithm นั้นสามารถช่วยในการปรับค่าของระบบควบคุมได้ แต่ใช้เวลานานเนื่องจากจำเป็นต้องสุ่มค่าในช่วงที่กว้างเพื่อให้ครอบคลุมค่าที่อาจจะเป็นไปได้ทั้งหมด เพื่อแก้ปัญหานี้ ผู้จัดทำจึงใช้การปรับค่าด้วยวิธีของ Ziegler-Nichols ในการกำหนดช่วงข้อมูลสำหรับการทำ Genetic Algorithm ในการปรับค่าของระบบควบคุม PID สำหรับควบคุมการเดินของหุ่นยนต์ ในงานวิจัยนี้ได้ทำการสร้างแบบจำลองทางคณิตศาสตร์และแบบจำลองของ Exoskeleton สำหรับร่างกายที่อ่อนแอที่ควบคุมการเคลื่อนไหวที่ส่วนเข่าและส่วนสะโพก ระบบควบคุม PID ที่ใช้ในการควบคุมการเคลื่อนที่ของหุ่นยนต์ทำการปรับค่าด้วยการใช้ Ziegler-Nichols ร่วมกับการทำ Genetic Algorithm โดยทดสอบการทำงานผ่าน MATLAB Simulink แล้วเปรียบเทียบกับการใช้วิธีการของ Ziegler-Nichols และการใช้ Genetic Algorithm อย่างใดอย่างหนึ่ง โดยผลการทดสอบพบว่าการใช้ Ziegler-Nichols ร่วมกับ Genetic Algorithm นั้นใช้เวลาในการปรับค่าน้อยกว่าการใช้การปรับการควบคุมด้วยวิธีการ Genetic Algorithm อย่างเดียว และยังให้ระบบควบคุมที่แม่นยำกว่าการใช้ Ziegler-Nichols เพียงอย่างเดียว

(วิทยานิพนธ์มีจำนวนทั้งสิ้น 46 หน้า)

คำสำคัญ : หุ่นยนต์ช่วยเดิน, ระบบควบคุม PID, Genetic Algorithm, Ziegler-Nichols

ลายมือชื่อนักศึกษา ลายมือชื่ออาจารย์ที่ปรึกษา

6609892 : Siwakorn Pankrachang
 Thesis Title : Using Ziegler-Nichols-Assisted Genetic Algorithm for Tuning the Control System of Exoskeleton Robot
 Program : Master of Engineering in Biomedical Engineering
 Thesis Advisor : Asst.Prof. Yutthana Pititheeraphab, Ph.D.

The PID control system is widely used in controlling the movement of robots due to its reliability and simplicity in operation. Adjusting the PID control settings in robots with complex structures, such as exoskeletons for lower limbs, to work appropriately using conventional methods is complicated and lacks precision. Therefore, algorithms such as Genetic Algorithm are employed to assist in tuning the control settings. Using Genetic Algorithm can help in tuning the control system settings, but it takes a long time because it needs to randomly generate values over a wide range to cover all possible values. To solve this problem, the researchers used the Ziegler-Nichols method to determine the data range for the Genetic Algorithm to tune the PID control system for controlling the walking of the exoskeleton. In this research, mathematical and exoskeleton models for the lower limbs, which control the movement at the knee and hip joints, were created. The PID control system used to control the movement of the robot was tuned using Ziegler-Nichols in conjunction with Genetic Algorithm. The system was tested through MATLAB Simulink and compared with the use of only the Ziegler-Nichols method and only the Genetic Algorithm. The test results showed that using Ziegler-Nichols with Genetic Algorithm took less time to tune the values than using the Genetic Algorithm alone and provided more accurate control than using only the Ziegler-Nichols method.

(Total 46 pages)

Keywords: Walking Assist Robots, PID Control System, Genetic Algorithm, Ziegler-Nichols

Student's Signature Thesis Advisor's Signature

สารบัญ

	หน้า
กิตติกรรมประกาศ	ก
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูป	ช
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	3
1.3 สมมติฐานของการทำงานวิจัย	3
1.4 กรอบแนวความคิดการทำโครงการ	3
1.5 ขั้นตอนการศึกษา	5
บทที่ 2 ทฤษฎีที่เกี่ยวข้องและทบทวนวรรณกรรม	6
2.1 Exoskeleton ในทางการแพทย์	6
2.2 แบบจำลองทางคณิตศาสตร์ของ Exoskeleton	8
2.3 PID controller	11
2.4 Genetic Algorithm	13
2.5 การเดิน	16
บทที่ 3 ระเบียบวิธีการวิจัย	18
3.1 อุปกรณ์ในการดำเนินการ	18
3.2 การออกแบบการวิจัย	19
3.3 การทดสอบการทำงานของระบบควบคุม	27

สารบัญ (ต่อ)

	หน้า
บทที่ 4	ผลการวิจัย
	31
4.1	การจำลองด้วย MATLAB Simulink
	31
4.2	การทดลองกับแบบจำลองขาหุ่นยนต์ช่วยเดิน
	36
บทที่ 5	สรุปผลและข้อเสนอแนะ
	42
5.1	สรุปผลการวิจัย
	42
5.2	ข้อเสนอแนะ
	42
บรรณานุกรม	44
ประวัติผู้วิจัย	46



สารบัญตาราง

ตารางที่		หน้า
1.1	ขั้นตอนการศึกษา	5
3.1	ลักษณะของขาแต่ละท่อน	23
4.1	ค่า Gain ของระบบควบคุม PID	31
4.2	ระบบควบคุมส่วนสะโพกเมื่อ Input เป็น step response	35
4.3	ระบบควบคุมส่วนเข้าเมื่อ Input เป็น step response	35
4.4	ผลการทดสอบระบบควบคุมของแบบจำลองขาหุ่นยนต์เมื่อใช้ Input เป็น Step input	40
4.5	ผลการทดสอบระบบควบคุมของแบบจำลองขาหุ่นยนต์เมื่อใช้ Input เป็น ข้อมูลการเดิน	40



สารบัญรูป

รูปที่		หน้า
1.1	หุ่นยนต์ช่วยเหลือการเดิน HAL-5	1
1.2	กรอบแนวคิดการทำงานวิจัย	4
2.1	ภาพเปรียบเทียบลักษณะของ End-Effector Robot และ Exoskeleton	6
2.2	ลักษณะของ Exoskeleton	7
2.3	Free Body Diagram ของหุ่นยนต์ช่วยเดิน	9
2.4	Free Body Diagram ของ Pendulum Model	9
2.5	กระบวนการของ Genetic Algorithm	14
2.6	กระบวนการสร้างความหลากหลายด้วยการทำ Crossover	14
2.7	กระบวนการสร้างความหลากหลายด้วยการทำ Mutation	15
2.8	การเคลื่อนที่ของขาระหว่างวงจรการเดิน	16
2.9	ค่าเฉลี่ยของมุมของข้อต่างๆ ของขาในระหว่างการเดินบนสายพานที่ความเร็วต่างๆ เปรียบเทียบระหว่างผู้ใหญ่ตอนต้น (สี่เทา) และผู้ใหญ่อ่อนปลาย (สีน้ำเงิน)	17
3.1	ชุดโครงหุ่นยนต์ที่ใช้เป็นต้นแบบในงานวิจัย	19
3.2	แบบจำลองขาหุ่นยนต์ที่ใช้ในงานวิจัย	20
3.3	แผนผังของระบบควบคุมขาหุ่นยนต์ที่ใช้ในงานวิจัย	21
3.4	ระบบควบคุมขาหุ่นยนต์ด้วย MATLAB Simulink ที่ใช้ในงานวิจัย	21
3.5	ระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน	22
3.6	Free Body Diagram ของขาแต่ละข้าง	22
3.7	Pendulum model ของขาแต่ละท่อน	23
3.8	การสั้นของ Output ของระบบควบคุมในการปรับค่าด้วยวิธีของ Ziegler-Nichols	24
3.9	flowchart ของการปรับค่า Gain ด้วยวิธีการของ Zeigler-Nichols	25
3.10	การทำ Objective function	26

สารบัญรูป (ต่อ)

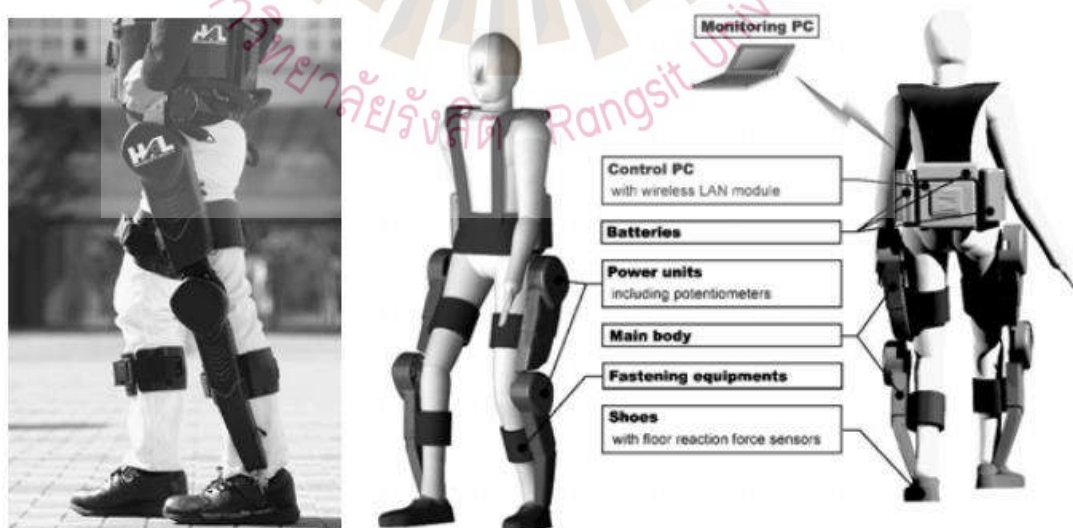
รูปที่	หน้า
3.11	กระบวนการใช้ Genetic Algorithm ในการหา Objective Function จากระบบควบคุมแบบ PID
3.12	flowchart ของการทำงานของ Genetic Algorithm
3.13	flowchart ของการทำงานของ Genetic Algorithm โดยใช้การปรับค่าด้วย วิธีของ Ziegler Nichols
3.14	Block Diagram การทดสอบระบบควบคุม
3.15	ข้อมูลมุมของข้อต่อของขามนุษย์ระหว่างการเดินที่ใช้ในการทดสอบ ระบบควบคุม
3.16	แบบจำลองขาหุ่นยนต์ช่วยเดิน
3.17	แผนผังของระบบควบคุมที่ใช้ในการทำงานของหุ่นยนต์
3.18	ระบบควบคุมภายใน Simulink
4.1	การทดสอบแบบจำลองระบบควบคุมใน MATLAB Simulink
4.2	Output ของระบบควบคุมส่วนสะโพกเมื่อ Input คือ step response
4.3	Output ของระบบควบคุมส่วนเข้าเมื่อ Input คือ step response
4.4	Output ของระบบควบคุมส่วนสะโพกเมื่อ Input เป็นข้อมูลการเดิน
4.5	Output ของระบบควบคุมส่วนเข้าเมื่อ Input เป็นข้อมูลการเดิน
4.6	ผลการทดสอบการทำงานของระบบควบคุมส่วนสะโพกเมื่อ Input เป็น Step input
4.7	ผลการทดสอบการทำงานของระบบควบคุมส่วนเข้าเมื่อ Input เป็น Step input
4.8	ผลการทดสอบการทำงานของระบบควบคุมส่วนสะโพกเมื่อ Input เป็น ข้อมูลการเดิน
4.9	ผลการทดสอบการทำงานของระบบควบคุมส่วนเข้าเมื่อ Input เป็นข้อมูล การเดิน

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ชุดโครงหุ่นยนต์สำหรับรองรับร่างกายท่อนล่าง หรือ Lower Extremities Exoskeleton กล่าวถึงหุ่นยนต์ที่ผู้ใช้สามารถสวมเข้ากับร่างกายของผู้ใช้ เพื่อช่วยในการเคลื่อนไหวของร่างกาย ในช่วงหลายสิบปีที่ผ่านมา มีการพัฒนาชุดโครงหุ่นยนต์สำหรับร่างกายท่อนล่างจำนวนมาก โดยสามารถแบ่งออกตามการใช้งานได้เป็นสามกลุ่ม ได้แก่ หุ่นยนต์สำหรับช่วยเหลือในการใช้แรงงาน หุ่นยนต์สำหรับการช่วยเหลือการเดินสำหรับผู้พิการหรือผู้ที่มีความบกพร่องในการเดิน และหุ่นยนต์สำหรับช่วยเหลือในการทำกายภาพบำบัดร่างกายท่อนล่าง (Chen et al., 2016) (Hussain, Goecke, & Mohammadian, 2021, pp. 1-2) เพื่อการรองรับปริมาณผู้ป่วยที่มีอาการบกพร่องทางการเดินที่เพิ่มขึ้น ในปัจจุบัน จึงมีการพัฒนาหุ่นยนต์ Exoskeleton สำหรับช่วยเหลือในการฟื้นฟูการเดินหรือการทำกายภาพบำบัดเพิ่มมากขึ้น ดังเช่น HAL-5 ดังแสดงในรูปที่ 1.1



รูปที่ 1.1 หุ่นยนต์ช่วยเหลือการเดิน HAL-5

ที่มา: Sankai, 2010

เนื่องจากการเดินเป็นการเคลื่อนที่ที่มีการเคลื่อนไหวของขาที่มีลักษณะเฉพาะเพื่อทำการพยุ่งและขับเคลื่อนร่างกายไปได้ (Simonsen, 2014) เพื่อให้หุ่นยนต์ช่วยเหลือในการเดินสามารถทำการเดินได้อย่างเหมาะสม ระบบควบคุมการเดินของหุ่นยนต์ต้องสามารถควบคุมให้ข้อต่อต่างๆ ของหุ่นยนต์เคลื่อนที่ไปยังตำแหน่งของขาระหว่างการเดินได้ โดยระบบควบคุมแบบ PID เป็นระบบควบคุมที่นิยมใช้ในการควบคุมการเคลื่อนที่ของหุ่นยนต์ เนื่องจากความน่าเชื่อถือของระบบและความเรียบง่ายในการใช้งาน ระบบควบคุม PID ประกอบด้วยระบบควบคุม 3 ส่วนที่ทำงานร่วมกันได้แก่ ระบบควบคุมแบบสัดส่วน (Proportional control) ระบบควบคุมแบบบูรณาการรวมหน่วย (Integral control) และระบบควบคุมแบบอนุพันธ์ (Derivative control) (Amiri et al., 2020) อย่างไรก็ตาม ประสิทธิภาพในการทำงานของระบบควบคุมแบบ PID นั้นขึ้นอยู่กับค่า Gain ของระบบควบคุมทั้ง 3 ส่วน โดยมีวิธีการปรับค่า Gain ระบบควบคุมหลายวิธี เช่น วิธีการของ Ziegler-Nichols, วิธีการของ Chien-Hrones-Reswick, วิธีการใช้ Fuzzy method เป็นต้น วิธีการปรับค่าเหล่านี้ผู้ที่ทำการปรับค่าจำเป็นต้องมีความรู้เกี่ยวกับการทำงานของระบบควบคุม และอาศัยการปรับค่ากับระบบควบคุมและวัดการตอบสนองโดยตรงเพื่อคำนวณค่า Gain ที่เหมาะสม ในการออกแบบระบบควบคุมของระบบที่มีความซับซ้อนเช่นหุ่นยนต์ช่วยเหลือการเดิน การปรับค่าด้วยวิธีการทั่วไปนั้นมีความยุ่งยากและขาดความแม่นยำ จึงมีการศึกษาวิธีการปรับค่า Gain แบบอื่นที่ทำการปรับค่า Gain ของระบบควบคุมโดยอัตโนมัติ โดยอาศัยการทำงานของ Algorithm มาใช้ช่วยในการปรับค่า เช่น Particle Swarm Optimization, Differential Evolution, และ Genetic Algorithm โดย Genetic Algorithm ซึ่งเป็น Algorithm ที่อาศัยการเลียนแบบกระบวนการคัดเลือกทางธรรมชาติของวิวัฒนาการของสิ่งมีชีวิตในการทำการปรับค่า Gain ของระบบควบคุม ซึ่งพบว่ามีประสิทธิภาพที่สูงขึ้นกว่าการปรับค่าของระบบควบคุม PID ด้วยวิธีการอื่น ๆ (Suseno and Ma'arif, 2021)

การใช้ Genetic Algorithm ในการปรับค่าของระบบควบคุมนั้น ในการหาค่า Gain ที่เหมาะสม จำเป็นต้องทำการสุ่มค่าในช่วงที่กว้างเพื่อให้ครอบคลุมทุกค่าของระบบควบคุมที่เป็นไปได้ ทำให้ใช้เวลานานในการหาค่าของระบบควบคุมที่เหมาะสม การหาช่วงของข้อมูลสำหรับการสุ่มค่าสำหรับ Genetic Algorithm ที่เหมาะสมจะช่วยให้ใช้เวลาในการทำงานน้อยลง เพื่อแก้ปัญหานี้ผู้จัดทำมีความประสงค์ที่จะใช้วิธีการปรับค่าของ Ziegler Nichols ในการช่วยหาช่วงข้อมูลที่เหมาะสมในการทำ Genetic Algorithm เพื่อทำการปรับค่าของระบบควบคุม PID สำหรับควบคุมการเดินของชุดโครงหุ่นยนต์สำหรับช่วยเหลือการเดิน โดยในงานวิจัยนี้ จะทำการสร้างแบบจำลองของชุดโครงหุ่นยนต์ที่ควบคุมการเคลื่อนที่ด้วยระบบควบคุมแบบ PID แล้วทำการจำลองการทำงานของระบบควบคุมโดยใช้โปรแกรม MATLAB Simulink โดยเปรียบเทียบเวลาที่ใช้ในการปรับค่า

ระบบควบคุม และการทำงานของระบบควบคุมระหว่างระบบควบคุมที่ใช้การ Tuning ด้วย Genetic Algorithm อย่างเดียว และระบบควบคุมที่ใช้การ Tuning โดยใช้การปรับค่าด้วยวิธีการของ Ziegler-Nichole ในการหาช่วงของข้อมูลที่เหมาะสม

1.2 วัตถุประสงค์

1.2.1 สร้างแบบจำลองทางคณิตศาสตร์ของขาของหุ่นยนต์ช่วยเหลือนการเดิน โดยใช้ระบบควบคุมแบบ PID สำหรับควบคุมการเดิน

1.2.2 ใช้ Genetic Algorithm ปรับค่าของระบบควบคุมแบบ PID โดยใช้การปรับค่าด้วย Ziegler-Nichols ในการหาช่วงสำหรับข้อมูลเริ่มต้นของ Genetic Algorithm

1.2.3 เปรียบเทียบกับระบบควบคุม PID ที่ปรับค่าด้วยวิธีการของ Ziegler-Nichols และ การใช้ Genetic Algorithm อย่างเดียว

1.2.4 ทดสอบการทำงานด้วย MATLAB Simulink และแบบจำลองหุ่นยนต์

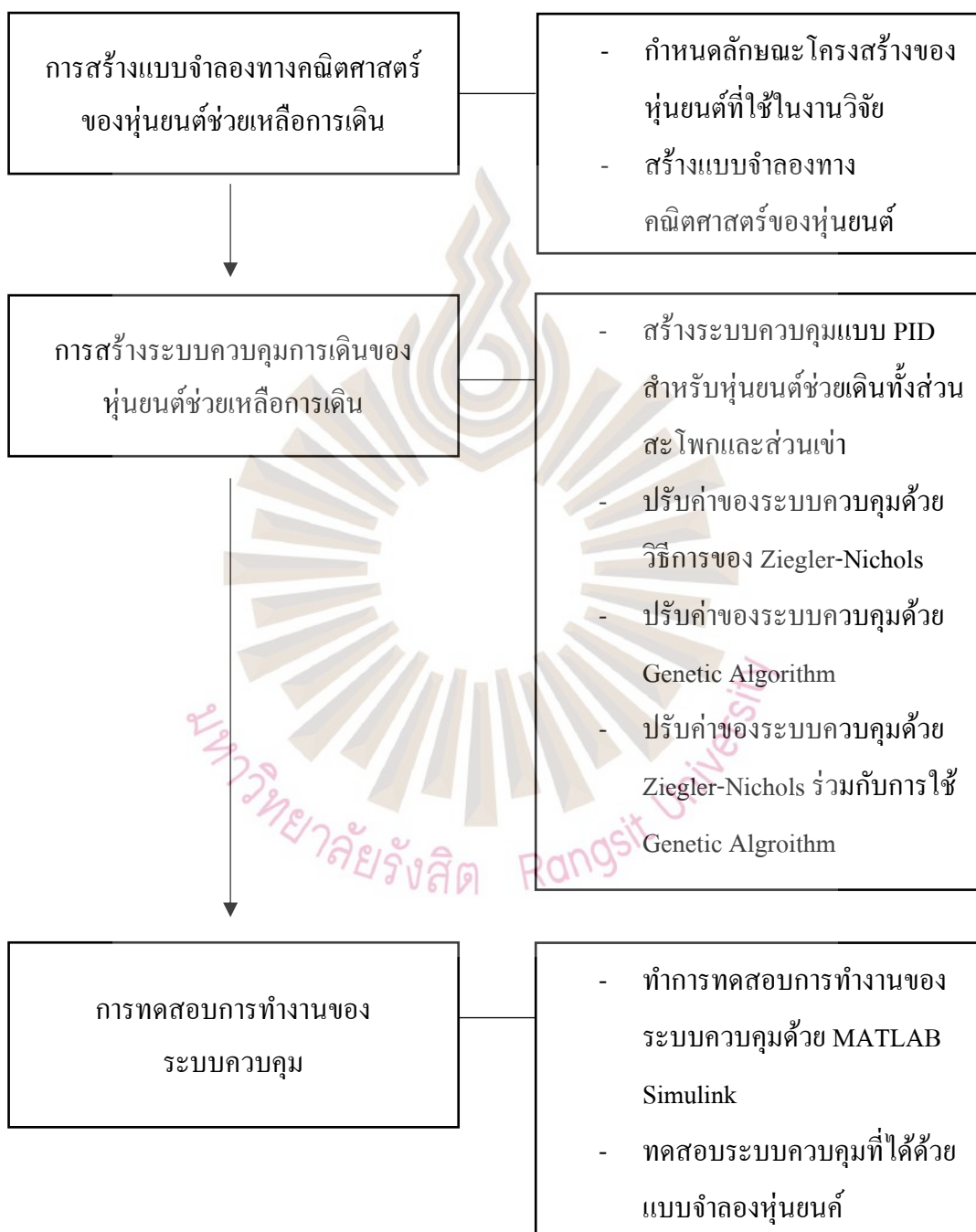
1.3 สมมติฐานของการทำงานวิจัย

การใช้การปรับค่าด้วยวิธีการของ Ziegler Nichols ร่วมกับ Genetic Algorithm ในการปรับค่าของระบบควบคุมการเดินของหุ่นยนต์ทำให้ระบบควบคุมการเดินมีความผิดพลาดน้อยกว่าการปรับค่าด้วยวิธีการของ Ziegler-Nichols อย่างเดียว และใช้เวลาในการค่าที่น้อยกว่าการใช้วิธี Genetic Algorithm เพียงอย่างเดียว

1.4 กรอบแนวคิดการทำงานวิจัย

งานวิจัยนี้มีการดำเนินการแบ่งเป็น 3 ส่วน โดยส่วนแรก เป็นการสร้างแบบสำหรับหุ่นยนต์ช่วยเดินแบบจำลองทางคณิตศาสตร์ของหุ่นยนต์ดังกล่าวสำหรับใช้ในการศึกษาในขั้นตอนต่อไป ในส่วนที่สองเป็นการสร้างระบบควบคุม PID ของหุ่นยนต์ช่วยเดินนี้ โดยในงานวิจัยนี้ศึกษาการปรับค่าด้วยวิธีการปรับค่าของระบบควบคุมโดยใช้วิธีการของ Ziegler Nichols ร่วมกับ Genetic Algorithm โดยมีการสร้างระบบควบคุมด้วยวิธีการของ Ziegler Nichols และ การใช้ Genetic

Algorithm เพียงอย่างเดียวในการเปรียบเทียบกับระบบควบคุมที่ต้องการศึกษา ในส่วนที่สามเป็นการเปรียบเทียบการทำงานของระบบควบคุมทั้ง 3 แบบ โดยทำการเปรียบเทียบและทดสอบการทำงานด้วยการจำลองโดยใช้ MATLAB Simulink และการทดลองกับแบบจำลองหุ่นยนต์ช่วยเดิน



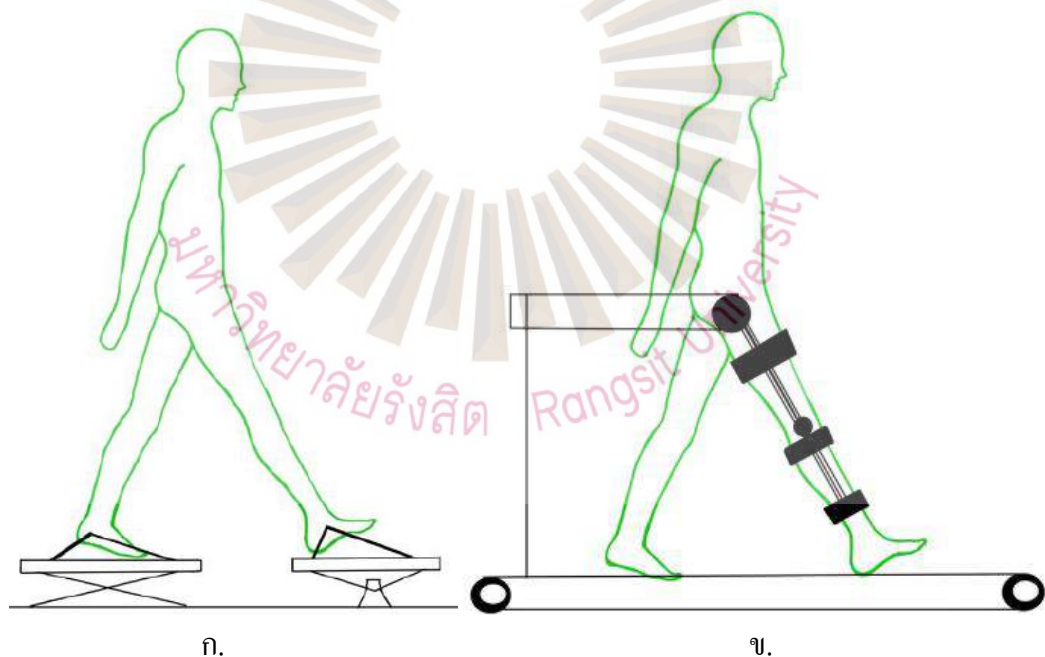
รูปที่ 1.2 กรอบแนวคิดการทำงานวิจัย

บทที่ 2

ทฤษฎีที่เกี่ยวข้องและทบทวนวรรณกรรม

งานวิจัยการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน มีทฤษฎีที่เกี่ยวข้องกับงานวิจัยทั้งหมด 5 ส่วน ได้แก่ Exoskeleton ทางกายภาพ การสร้างแบบจำลองทางคณิตศาสตร์ของ Exoskeleton ระบบควบคุมแบบ PID กระบวนการ Genetic Algorithm และการเดินของมนุษย์

2.1 Exoskeleton ในทางการแพทย์



รูปที่ 2.1 ภาพเปรียบเทียบลักษณะของ End-Effector Robot และ Exoskeleton

(ก.) End-Effector Robot (ข.) Exoskeleton

ที่มา: Molteni, Gasperini, Cannaviello และ Guanziroli, 2018

หุ่นยนต์สำหรับการทำกายภาพบำบัดนั้นสามารถแบ่งออกได้เป็น 2 กลุ่มใหญ่ๆ ได้แก่ End-effector type และ Exoskeleton ดังรูปที่ 2.1 โดยที่หุ่นยนต์แบบ End-effector นั้น จะควบคุมการเคลื่อนไหวที่ปลายสุดของรยางค์เท่านั้น ในขณะที่หุ่นยนต์แบบ Exoskeleton นั้น จะถูกออกแบบให้มีการเคลื่อนไหวในลักษณะเดียวกับรยางค์ของมนุษย์ โดยจะมีจุดเชื่อมต่อกับรยางค์ของผู้ใช้ในหลายจุด ทำให้สามารถควบคุมการเคลื่อนไหวในแต่ละส่วนของข้อต่อของมนุษย์ได้ และสามารถเคลื่อนไหวได้คล้ายคลึงกับการเคลื่อนที่ของรยางค์มนุษย์ (Molteni et al., 2018)

Exoskeleton สำหรับการช่วยเหลือในการฟื้นฟูการเดินนั้นจะต้องมีความสามารถในการออกแรงช่วยเหลือการเคลื่อนไหวสำหรับรยางค์ที่อ่อนแอ โดยสามารถแบ่งหุ่นยนต์ช่วยเหลือการเดินออกได้เป็น 2 ประเภท ได้แก่ Treadmill-Base Exoskeleton และ Overground Exoskeleton ดังรูปที่ 2.2



รูปที่ 2.2 ลักษณะของ Exoskeleton

ก. Treadmill-based, ข. Overground

ที่มา: Molteni et al., 2018

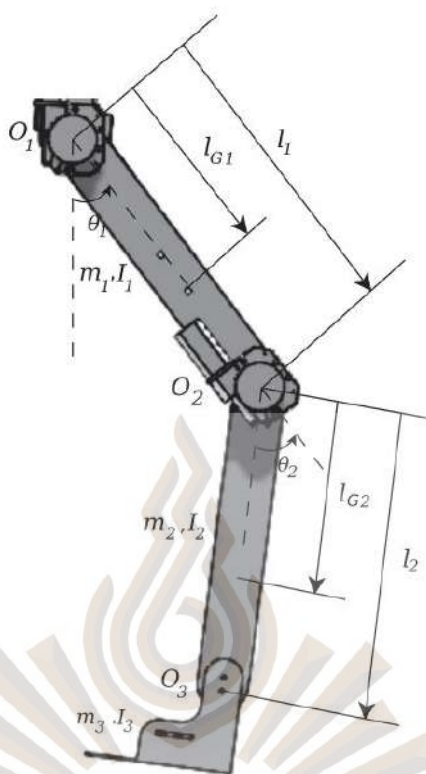
Treadmill-based exoskeleton คือชุดโครงหุ่นยนต์ที่ประกอบด้วยหุ่นยนต์ส่วนขา โครงสร้างรับน้ำหนัก และลู่วิ่ง ทำให้หุ่นยนต์ประเภทนี้จะยึดตรึงอยู่กับที่ โดยขาของผู้ใช้จะเดินอยู่ด้านบนสายพาน ทำให้ผู้ใช้สามารถใช้งานได้โดยไม่ต้องแบกรับน้ำหนักของผู้ใช้และระบบ อย่างไรก็ตาม

หุ่นยนต์แบบ Treadmill-based นั้นมีข้อจำกัดหลายประการ เช่น ผู้ป่วยไม่สามารถทำกายภาพบำบัดบนพื้นจริงได้ ทำให้การประสานงานระหว่างประสาทสัมผัสส่วนต่างๆ เกิดขึ้นน้อยกว่าการเดินปกติ และการรับน้ำหนักด้วยโครงสร้าง ประกอบกับการทำงานของสายพานนั้น ทำให้ผู้ป่วยไม่ได้มีส่วนในการตัดสินใจก้าวเดินมากนัก (Molteni et al., 2018; Goffredo et al., 2019)

ในหุ่นยนต์ Over-ground exoskeleton นั้น หุ่นยนต์จะไม่ได้ตรึงอยู่กับที่เหนือสายพาน แต่จะออกแบบให้ผู้ป่วยสวมขาของหุ่นยนต์เดินไปบนพื้นราบได้ หุ่นยนต์ลักษณะนี้ทำให้ผู้ป่วยสามารถเดินบนพื้นจริงได้ โดยการเคลื่อนไหวมาจากการเดิน โดยไม่มีการเคลื่อนที่ของสายพานเข้ามาช่วยให้เกิดการเดิน การออกแบบชุดโครงหุ่นยนต์แบบ Over-ground นั้น จึงต้องคำนึงถึงข้อจำกัดหลายประการด้วย เช่น ชุดโครงหุ่นยนต์นั้นต้องมีน้ำหนักที่ไม่สูงเกินไป และต้องมีขนาดที่ไม่ใหญ่เกินไป เพื่อไม่ให้เกิดภาระต่อผู้สวมใส่จนเป็นอุปสรรคต่อการใช้งาน จึงทำให้ Overground Exoskeleton จำนวนมากมีโครงสร้างที่ครอบคลุมการเคลื่อนไหวเพียง 3-4 Degree of Motion ต่อขาหนึ่งข้าง เพื่อให้โครงสร้างไม่ซับซ้อน แต่เนื่องจากคุณสมบัติของ Over-ground Exoskeleton ที่มีขนาดเล็กกว่า จึงสามารถให้อิสระในการเดินกับผู้ใช้งานได้มากกว่า ทั้งในเรื่องของสถานที่ในการใช้หุ่นยนต์ ซึ่งไม่จำเป็นต้องอยู่เหนือสายพานตลอดเวลา และในท่าทางการใช้งาน ซึ่งสามารถจำลองลักษณะการเคลื่อนไหวได้หลายหลายคล้ายคลึงกับการเคลื่อนที่ในชีวิตประจำวันมาก ทำให้ผู้ป่วยมีการตัดสินใจก้าวเดินมากกว่า และมีการใช้ประสาทสัมผัสมากกว่าด้วยเช่นกัน (Molteni et al., 2018; Chen et al., 2013)

2.2 แบบจำลองทางคณิตศาสตร์ของ Exoskeleton

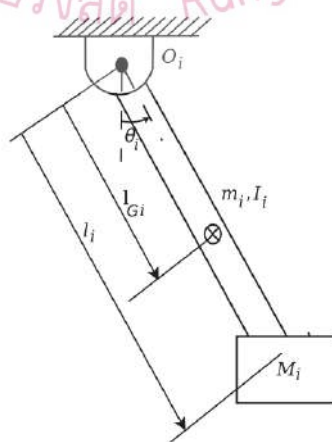
Amirit, Ramli และ Ubrahim (2019) ได้ศึกษาเกี่ยวกับการสร้างแบบจำลองทางคณิตศาสตร์ของหุ่นยนต์ Exoskeleton สำหรับช่วยเหลือนการเดิน 4 Degree of Freedom เพื่อนำไปใช้ในการจำลองการทำงาน โดยใช้ Pendulum model ในการคำนวณหา Transfer Function แทนขาแต่ละท่อน ขาของหุ่นยนต์แต่ละข้างสามารถแสดงอยู่ในรูปของ Free Body Diagram ได้ดังรูปที่ 2.3 โดยที่จุด O_1 และ O_2 แทนตำแหน่งข้อสะโพกและข้อเข่าตามลำดับ l_1 และ l_2 แทนความยาวของต้นขาและปลายขาตามลำดับ และ l_{G1} และ l_{G2} แทนระยะทางของต้นขาและปลายขาระหว่างข้อต่อและจุดศูนย์กลางมวล ตามลำดับ



รูปที่ 2.3 Free Body Diagram ของหุ่นยนต์ช่วยเดิน

ที่มา: Amiri et al., 2019

การเคลื่อนที่ของขาหุ่นยนต์แต่ละส่วนสามารถจำลองออกมาในรูปแบบ Pendulum Model ได้ดังรูปที่ 2.4 เพื่อหาสมการทางคณิตศาสตร์สำหรับการจำลองการเคลื่อนที่ของชุดโครงหุ่นยนต์ กระทำโดยการสร้างสมการการเคลื่อนที่ของขาโดยผลรวมของแรงบิดรอบจุด O_i ได้ดังนี้



รูปที่ 2.4 Free Body Diagram ของ Pendulum Model

ที่มา: Amiri et al., 2019

$$\tau_i = (M_i + m_i)gl_{Gi} \sin \theta_i + \tau_{fi} + I_i \ddot{\theta}_i \quad (2-1)$$

เมื่อ	m_i	แทนมวลของ Link ท่อนที่กำลังพิจารณา
	M_i	แทนมวลของ Link ที่อยู่ถัดจากท่อนที่กำลังพิจารณา
	g	แทนค่าคงที่ความโน้มถ่วง
	l_{Gi}	แทนความยาวจากจุด ไปยังศูนย์กลางมวลของ link
	τ_i	แทนแรงบิดที่กระทำต่อ link
	τ_{fi}	แทนแรงบิดจากความเสียดทาน
	I_i	แทนโมเมนต์ความเฉื่อยของ link
	$\ddot{\theta}_i$	แทนความเร่งเชิงมุม

จากสมการที่ 1 เพื่อหาสมการสุดท้ายออกมา ทำการหาค่าของ l_{Gi} และ τ_{fi} ได้ดังนี้

$$l_{Gi} = \frac{M_i I_i + 0.5 m_i I_i}{M_i + m_i} \quad (2-2)$$

$$\tau_{fi} = b_i \dot{\theta}_i \quad (2-3)$$

เมื่อ b_i แทนสัมประสิทธิ์ความเสียดทานจลน์ของข้อต่อ
นำค่าของสมการที่ 2 และ 3 ไปแทนค่าในสมการที่ 1 ได้

$$\tau_i = \ddot{\theta}_i + \frac{b_i}{I_i} \dot{\theta}_i + \frac{(M_i + m_i)gl_{Gi}}{I_i} \theta_i \quad (2-4)$$

นำสมการที่ 4 ไปผ่าน Laplace Transform ได้ดังนี้

$$\tau_i(s) = \theta_i(s)s^2 + \frac{b_i}{I_i} \theta_i(s)s + \frac{(M_i + m_i)gl_{Gi}}{I_i} \theta_i(s) \quad (2-5)$$

ได้ Transfer Function ของการเคลื่อนที่ของขาแต่ละท่อน คือ

$$G_i(s) = \frac{\theta_i(s)}{\tau_i(s)} = \frac{1}{s^2 + \frac{b_i}{I_i}s + \frac{(M_i + m_i)gl_{Gi}}{I_i}} \quad (2-5)$$

2.3 PID controller

ระบบควบคุมแบบสัดส่วนรวมกับแบบบูรณาการรวมหน่วยและอนุพันธ์ (Proportional Integral Derivative) หรือที่เรียกกันทั่วไปว่าระบบควบคุมแบบ PID นั้น เป็นระบบควบคุมแบบปิดที่อาศัยคุณสมบัติของระบบควบคุม 3 แบบเข้าด้วยกัน คือ ระบบควบคุมแบบสัดส่วน ระบบควบคุมแบบบูรณาการรวมหน่วย และระบบควบคุมแบบอนุพันธ์เข้าด้วยกัน ซึ่งได้รับความนิยมมาก เนื่องจากการปรับค่า Gain อาศัยหลักการที่ไม่ซับซ้อน โดยระบบควบคุมย่อยทั้ง 3 แบบ คือ

2.3.1 ระบบควบคุมแบบสัดส่วน

ระบบควบคุมแบบสัดส่วน (Proportional Control หรือ P control) จะให้สัญญาณ Output เป็นสัดส่วนกับสัญญาณ Input โดยตรงโดยไม่มีการหน่วง โดยการควบคุมส่วนนี้จะมีลักษณะคือการตอบสนองจะเพิ่มขึ้นเป็นสัดส่วนกับความผิดพลาดที่เกิดขึ้นโดยตรง เมื่อปรับให้ค่า Gain สูงขึ้น ระบบจะมีการตอบสนองที่เร็วขึ้นตามไปด้วย อย่างไรก็ตาม ระบบควบคุมแบบสัดส่วนไม่สามารถขจัดความผิดพลาดในสภาวะคงตัวได้ โดยสามารถเขียนออกมาได้เป็นสมการว่า

$$U_{out}(t) = K_p U_{in}(t) \quad (2-7)$$

เมื่อ U_{out} แทนสัญญาณ Output

K_p แทนอัตราขยายของระบบควบคุม

U_{in} แทนสัญญาณ Input หรือสัญญาณข้อผิดพลาดที่เกิดขึ้น

2.3.2 ระบบควบคุมแบบบูรณาการรวมหน่วย

ระบบควบคุมแบบบูรณาการรวมหน่วย (Integral Control หรือ I control) จะให้สัญญาณ Output เป็นสัดส่วนกับผลรวมของสัญญาณ input ในช่วงเวลาหนึ่งๆ การควบคุมในส่วนนี้การตอบสนองจะมีขนาดที่มากตามขนาดของความผิดพลาดที่เกิดขึ้นและระยะเวลา ทำให้สามารถเพิ่ม

สัญญาณ Output ได้เมื่อระยะเวลาผ่านไประยะหนึ่ง ถึงแม้สัญญาณที่เข้ามาในส่วน Input จะมีขนาดเล็กก็ตาม ทำให้ยังสามารถสร้าง Output ออกมาได้ถึงแม้ระบบจะเข้าสู่สภาวะคงตัว โดยสามารถเขียนออกมาได้เป็นสมการว่า

$$U_{out}(t) = K_i \int U_{in}(t) dt \quad (2-8)$$

เมื่อ U_{out} แทนสัญญาณ Output
 K_i แทนอัตราขยายของระบบควบคุม
 U_{in} แทนสัญญาณ Input หรือสัญญาณข้อผิดพลาดที่เกิดขึ้น

2.3.3 ระบบควบคุมแบบอนุพันธ์

ระบบควบคุมแบบอนุพันธ์ (Derivative Control หรือ D control) จะให้สัญญาณ Output เป็นสัดส่วนกับอัตราการเปลี่ยนแปลงความผิดพลาดที่เกิดขึ้น การตอบสนองของระบบจะมีขนาดที่ขึ้นกับอัตราการเปลี่ยนแปลง input ซึ่งช่วยในส่วนของความเร็วในการตอบสนอง โดยสามารถใช้ในการลดผลของการพุ่งเกินของผลตอบสนองได้ โดยสามารถเขียนออกมาเป็นสมการได้ว่า

$$U_{out}(t) = K_d \frac{dU_{in}(t)}{dt} \quad (2-9)$$

เมื่อ U_{out} แทนสัญญาณ Output
 K_d แทนอัตราขยายของระบบควบคุม
 U_{in} แทนสัญญาณ Input หรือสัญญาณข้อผิดพลาดที่เกิดขึ้น

ในระหว่างการปรับค่าของระบบควบคุมแบบ PID นั้น เกณฑ์ในการประเมินประสิทธิภาพของระบบควบคุมนั้นสามารถกระทำได้หลายวิธี โดย Tan, Marquez และ Chen (2004) ระบุในงานวิจัยว่า ระบบควบคุมที่ใช้ Integral Time-Absolute Error (ITAE) และ Integral Square Error (ISE) เป็นเกณฑ์ในการปรับค่าของระบบควบคุม และระบบควบคุมที่ใช้การปรับค่าด้วยวิธีการของ Ziegler-Nichols ให้ค่าของระบบควบคุมที่เหมาะสมที่สุด โดยการหาค่าของ ITAE สามารถหาได้โดยสมการดังต่อไปนี้

$$ITAE = \int_0^{\infty} t|E(t)|dt \quad (2-10)$$

2.4 Genetic Algorithm

Genetic Algorithm เป็นหลักการ Optimization ที่อาศัยการเลียนแบบการคัดเลือกทางธรรมชาติในวิวัฒนาการของสิ่งมีชีวิต ในสิ่งมีชีวิตชนิดหนึ่ง ในสภาวะแวดล้อมในธรรมชาติ ตัวของสิ่งมีชีวิตในแต่ละรุ่นที่มีความเหมาะสมต่อสภาพแวดล้อมมากที่สุดจะมีโอกาสที่จะอยู่รอดมากกว่า และสิ่งมีชีวิตตัวที่มีความเหมาะสมต่อสภาพแวดล้อมน้อยกว่าจะลดจำนวนลงหรือสูญพันธุ์ไป สิ่งมีชีวิตที่รอดชีวิตมาได้มากกว่าจะมีโอกาสที่จะสืบพันธุ์ให้รุ่นลูกออกมา ซึ่งสิ่งมีชีวิตรุ่นลูกนี้จะได้รับสารพันธุกรรมมาจากสิ่งมีชีวิตรุ่นพ่อแม่ การผสมยีนของพ่อแม่ออกมามีโอกาสที่จะให้สิ่งมีชีวิตรุ่นลูกที่มีคุณสมบัติที่เหมาะสมต่อสิ่งแวดล้อมมากกว่ารุ่นพ่อแม่ ส่งผลให้สามารถอยู่รอดและขยายพันธุ์ได้มากขึ้น ซึ่ง Genetic Algorithm นั้นอาศัยหลักการเดียวกันนี้ในการหาค่าที่เหมาะสมต่อปัญหาที่สุด โดยทำการกำหนด Fitness Function ขึ้นมาเพื่อใช้ประเมินความเหมาะสมของข้อมูลต่อปัญหาที่กำหนดกระบวนการทำ Genetic Algorithm นั้นแบ่งออกได้เป็น 5 ขั้นตอน สามารถสรุปออกมาได้ดังรูปที่ 2.5 ดังนี้ (Ma'arif, Nabila, Iswanto, & Wahyunggoro, 2019)

2.4.1 สร้างกลุ่มของข้อมูลเพื่อเป็นประชากรรุ่นแรก

2.4.2 ประเมินความเหมาะสมของข้อมูลแต่ละตัวในประชากร (Fitness) โดยใช้

Objective Function

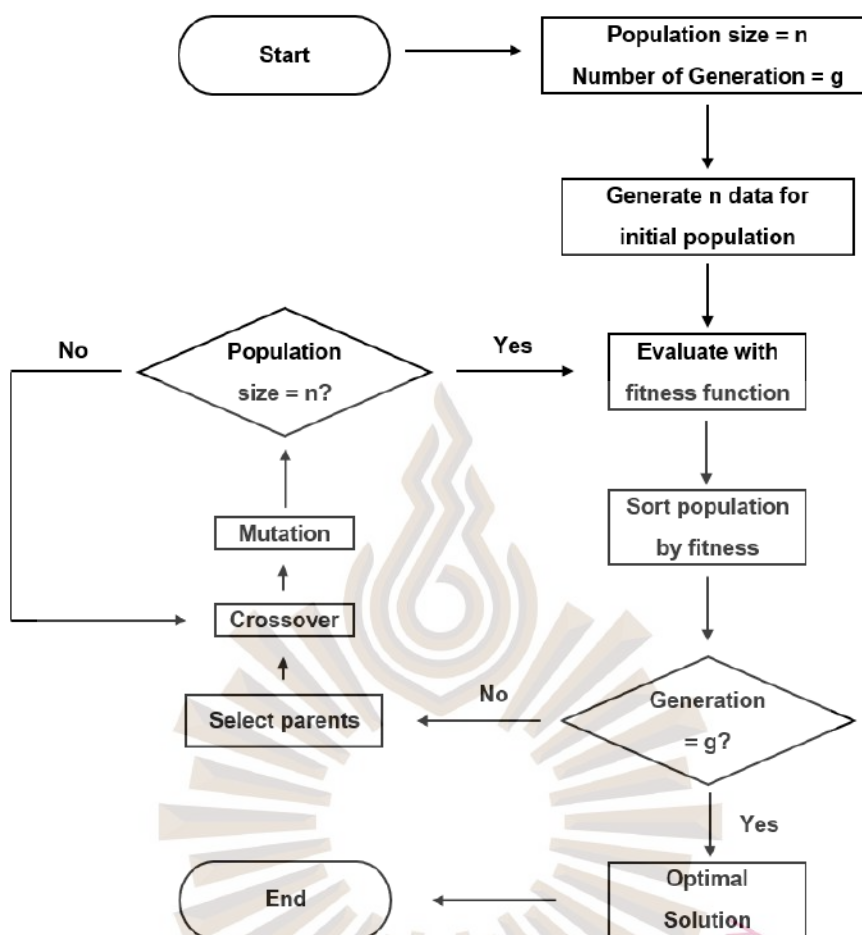
2.4.3 ทำการเลือก (Selection) ข้อมูลที่จะใช้สร้างประชากรรุ่นต่อไปตาม Fitness ในข้อ 2

2.4.4 ทำการ “ขยายพันธุ์” ข้อมูลที่ได้จากข้อ 3 เพื่อสร้างประชากรรุ่นลูก โดยใช้

กระบวนการ Crossover และ Mutation

2.4.5 ประเมิน Fitness ของประชากรรุ่นลูก

การ Crossover เป็นการแลกเปลี่ยนชิ้นส่วนของยีนรุ่นพ่อแม่จำนวน 2 ยีน เพื่อสร้างยีนรุ่นลูกออกมา ดังรูปที่ 2.6 ซึ่งสามารถทำได้หลายวิธี เช่น 1-point Crossover, 2-point Crossover และ Uniform Crossover ใน 1-point Crossover และ 2-point Crossover นั้น ข้อมูลรุ่นพ่อแม่จะถูกตัดในตำแหน่งที่กำหนดโดยผู้ที่ทำ Genetic Algorithm ข้อมูลของรุ่นลูกนั้นส่วนหนึ่งจะมาจากของข้อมูล



รูปที่ 2.5 กระบวนการของ Genetic Algorithm

Parent 1

1	1	0	0	1	0	0	1
---	---	---	---	---	---	---	---

Parent 2

1	0	1	0	0	0	1	1
---	---	---	---	---	---	---	---

Offspring 1

1	0	1	0	1	0	0	1
---	---	---	---	---	---	---	---

Offspring 2

1	1	0	0	0	0	1	1
---	---	---	---	---	---	---	---

รูปที่ 2.6 กระบวนการสร้างความหลากหลายด้วยการทำ Crossover

แรกของรุ่นพ่อแม่ และส่วนที่เหลือมาจากข้อมูลที่สอง โดย 1-point Crossover จะทำการตัดข้อมูลจำนวน 1 ตำแหน่ง และ 2-point Crossover ทำการตัดข้อมูล 2 ตำแหน่ง ในขณะที่ Uniform Crossover จะทำการตัดแบบสุ่มตำแหน่ง โดยให้ยีนรุ่นลูกจะได้รับการถ่ายทอดยีนจากรุ่นพ่อแม่ยีนละ 50%

Mutation เป็นการเพิ่มความหลากหลายในข้อมูลด้วยการเปลี่ยน bit ของข้อมูลอย่างสุ่ม โดยไม่มีการเปลี่ยนส่วนอื่นของข้อมูล ดังแสดงในรูปที่ 2.7 ซึ่งการสุ่มการเปลี่ยนข้อมูลนี้ช่วยให้เกิดข้อมูลใหม่ที่ส่งผลต่อความหลากหลายของข้อมูล ทำให้มีโอกาสที่จะได้ข้อมูลที่ดีขึ้นเพิ่มขึ้น

Offspring 1	1	1	0	0	1	0	0	1
New Offspring 1	1	1	0	0	0	0	0	1

รูปที่ 2.7 กระบวนการสร้างความหลากหลายด้วยการทำ Mutation

กระบวนการทั้งหมดนี้จะเกิดวนซ้ำไปเรื่อยๆ ในแต่ละรอบ ข้อมูลของประชากรจะถูกคัดเลือกโดยการประเมิน Fitness ประชากรที่เหลืออยู่จะให้ค่าที่มีความเหมาะสมต่อปัญหามากขึ้นเรื่อยๆ จนกระทั่งสิ้นสุดการทำ Genetic Algorithm ซึ่งอาจสิ้นสุดได้จากการวนรอบครบตามจำนวนที่กำหนดไว้ หรือได้คำตอบที่มีความเหมาะสมต่อปัญหาที่มากถึงระดับที่ผู้ใช้กำหนด (Chamnkar, 2017)

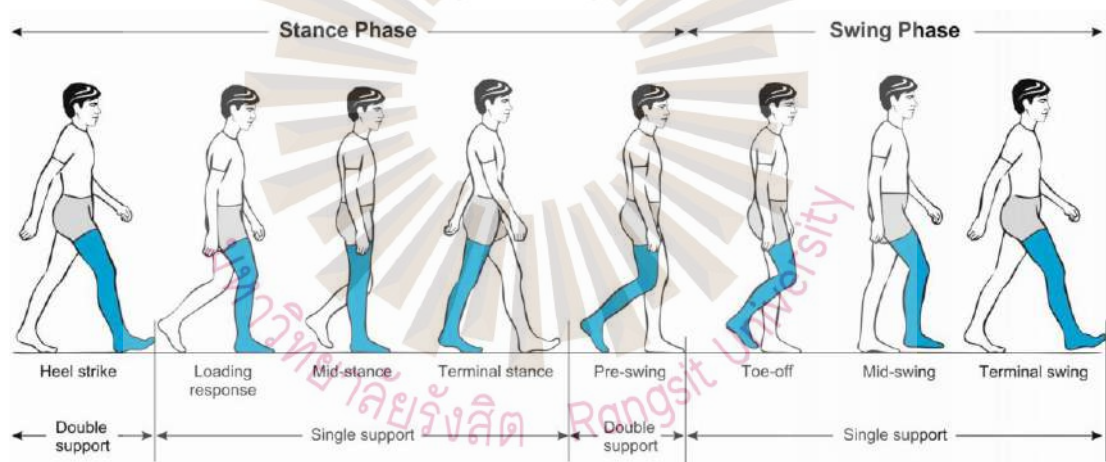
Suseno และ Ma'arif (2021) ได้ทำงานวิจัยเกี่ยวกับการใช้ Genetic Algorithm ในการปรับค่า Gain ของระบบควบคุม PID สำหรับควบคุมการทำงานของมอเตอร์ไฟฟ้ากระแสตรงผ่านบอร์ดพัฒนา Arduino Uno โดยสร้างแบบจำลองทางคณิตศาสตร์ของมอเตอร์ไฟฟ้า และสร้างระบบควบคุม PID ที่ปรับค่าของระบบควบคุมด้วยการใช้ Genetic Algorithm เปรียบเทียบกับการปรับค่าด้วยการลองผิดลองถูก แล้วทำการทดสอบทั้งจากการ Simulation และจากการทดลองกับมอเตอร์ไฟฟ้า ได้ผลสรุปว่าการใช้ Genetic Algorithm นั้นให้ระบบควบคุมที่มีผลลัพธ์ดีกว่าการปรับค่าด้วยการทำการปรับค่าด้วยการลองผิดลองถูก

Elbayomy, Zongxia และ Huaqing (2008) ได้ทำงานวิจัยเกี่ยวกับการใช้ Genetic Algorithm ในการปรับค่าของระบบควบคุม PID สำหรับควบคุมการทำงานของ Electro-hydraulic Servo

Actuation System (EHSAS) โดยใช้ค่า Gain เดิมของระบบควบคุม PID ที่ได้จากการลองผิดลองถูก มาใช้ในการกำหนดช่วงข้อมูลสำหรับการสร้างประชากรรุ่นแรกในการทำ Genetic Algorithm และ ใช้การหา Integral Absolute Error เป็น Fitness Function แล้วทำการทดสอบผ่าน MATLAB Simulink และแบบจำลองระบบ EHSAS ผลการวิจัยพบว่า การใช้ Genetic Algorithm ในการปรับค่า ของระบบควบคุมแบบ PID ทำให้ระบบควบคุมมีความแม่นยำสูงกว่าการใช้ PID แบบทั่วไป และสามารถนำผลลัพธ์จากการจำลองในคอมพิวเตอร์ไปใช้ในทางปฏิบัติได้

2.5 การเดิน

วงจรการเดิน คือการเคลื่อนไหวของขาในระหว่างการก้าวเดินไปข้างหน้า โดยนิยามแล้ว วงจรการเดินของร่างกายส่วนล่างแต่ละข้างจะเริ่มต้นขึ้นเมื่อส้นเท้าเริ่มแตะลงที่พื้น และสิ้นสุดเมื่อ ส้นเท้าข้างเดิมกลับมาแตะลงที่พื้นอีกครั้ง ดังแสดงในรูปที่ 2.8

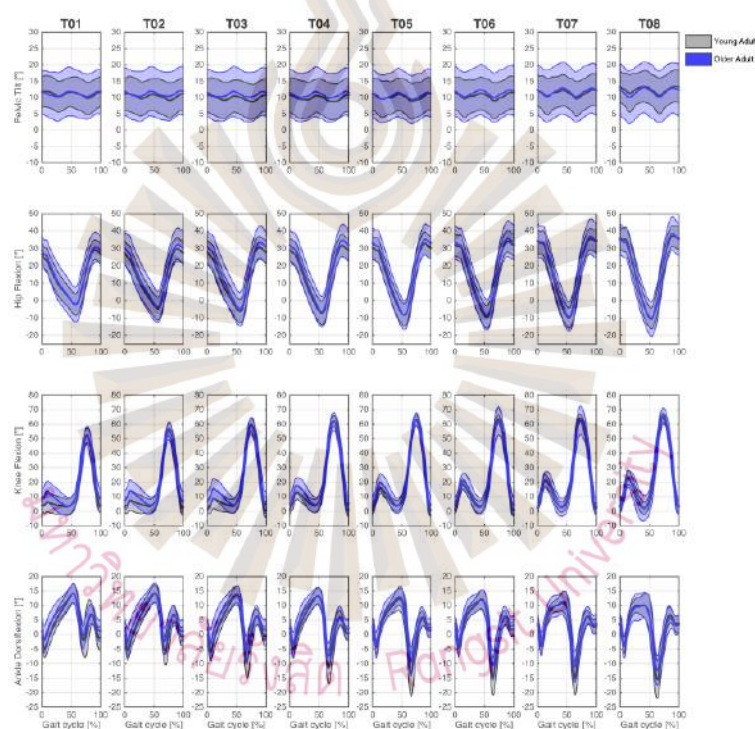


รูปที่ 2.8 การเคลื่อนที่ของขาระหว่างวงจรการเดิน

ที่มา: Pirker and Katzenschlager, 2016

ข้อมูลการเคลื่อนที่ของขาในระหว่างการเดินในงานวิจัยนี้อ้างอิงจากชุดข้อมูลการเดินจากการศึกษาการเดินของมนุษย์โดย Fukuchi, Fukuchi และ Duarte (2018) ในการศึกษาดังกล่าวนี้ ผู้จัดทำเก็บข้อมูลการเดินจากอาสาสมัคร 42 คน ซึ่งไม่มีประวัติโรคทางระบบประสาทหรือระบบกระดูก หรืออาการบาดเจ็บต่อร่างกายที่รุนแรงในช่วง 6 เดือนก่อนการเก็บข้อมูล โดยเก็บข้อมูลโดยใช้ระบบ Motion-capture ซึ่งประกอบด้วยกล้อง 12 ตัว เซนเซอร์วัดแรงจำนวน 5 แผ่นบนพื้น และลู่วิ่งสายพาน โดยอาสาสมัครทำการเดินด้วยเท้าเปล่าและสวมชุดที่อาสาสมัครรู้สึกสบาย และใช้แผ่น

สะท้อนแสงเป็นจุดอ้างอิงสำหรับการทำ Motion Capture จำนวน 26 ตำแหน่ง ผู้จัดทำให้อาสาสมัครทำการทดลองเดิน 2 ขั้นตอน ในขั้นตอนแรกอาสาสมัครกระทำการเดินบนพื้นราบ โดยกระทำการเดินบนทางเดินที่จัดเตรียมไว้ความยาว 10 เมตรที่ความเร็วที่อาสาสมัครเดินตามปกติ แล้วให้อาสาสมัครทำการเดินอีกสองครั้งที่ความเร็วที่เร็วกว่าการเดินปกติ 30% และที่ความเร็วช้ากว่าการเดินปกติ 30% ตามลำดับ ในขั้นตอนที่สอง อาสาสมัครทำการเดินบนลู่วิ่งสายพานเป็นเวลา 90 วินาทีจำนวน 8 ครั้ง ที่ระดับความเร็ว 40%, 55%, 70%, 85%, 100%, 115%, 130%, และ 145% ของความเร็วที่อาสาสมัครเดินตามปกติ แล้วทำการบันทึกข้อมูลการเดินในช่วง 30 วินาทีสุดท้ายของการทดลองแต่ละครั้ง ข้อมูลที่ได้แต่ละชุดข้อมูลจะถูกนำมาประมวลผลด้วยโปรแกรม Cortex 6.0



รูปที่ 2.9 ค่าเฉลี่ยของมุมของข้อต่างๆ ของขาในระหว่างการเดินบนสายพานที่ความเร็วต่างๆ

เปรียบเทียบระหว่างผู้ใหญ่ตอนต้น (สีเทา) และผู้ใหญ่ตอนปลาย (สีน้ำเงิน)

ที่มา: Fukuchi et al., 2018

จากข้อมูลการเคลื่อนที่ของข้อต่างๆ ที่ได้จากการศึกษา สามารถสรุปมุมของข้อสะโพก ข้อเข่า และข้อเท้าในแนว Sagittal Plane ได้ดังแสดงในรูปที่ 2.9 ซึ่งจากข้อมูลนี้ สามารถกำหนดช่วงของมุมของการเคลื่อนที่ของข้อต่างๆ ได้ว่า ช่วงของการเคลื่อนที่ของข้อสะโพกเคลื่อนที่จาก 50° ถึง -20° ช่วงการเคลื่อนที่ของเข่าจะเคลื่อนที่จาก 70° ถึง -10° และช่วงการเคลื่อนที่ของข้อเท้าเคลื่อนที่จาก 15° ถึง -20°

บทที่ 3

ระเบียบวิธีการวิจัย

งานวิจัยการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน มีระเบียบวิธีวิจัยแบ่งออกเป็น 3 ส่วน ได้แก่ อุปกรณ์ที่ใช้ในงานวิจัยนี้ ซึ่งรวมถึงส่วนซอฟต์แวร์ที่ใช้ในการสร้างระบบควบคุมและการจำลองการทำงานของหุ่นยนต์ และส่วนฮาร์ดแวร์ซึ่งกล่าวถึงอุปกรณ์ที่ใช้ในการสร้างแบบจำลองหุ่นยนต์ช่วยเดิน ในส่วนที่สองนั้นกล่าวถึงการออกแบบกระบวนการวิจัย ซึ่งกล่าวถึงการออกแบบแบบจำลองของหุ่นยนต์ช่วยเดินและการสร้างแบบจำลองทางคณิตศาสตร์ของหุ่นยนต์ การออกแบบระบบควบคุมหุ่นยนต์ และการปรับค่าของระบบควบคุมทั้ง 3 วิธี ได้แก่ การใช้วิธีการของ Ziegler-Nichols (ZN) การใช้ Genetic Algorithm (GA) และการใช้วิธีการของ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm (ZN-GA) และส่วนที่สามกล่าวถึงการทดสอบการทำงานของระบบควบคุม ซึ่งแบ่งออกเป็นสองส่วน ได้แก่ การทดสอบกับแบบจำลอง

3.1 อุปกรณ์ในการดำเนินการ

3.1.1 การทดสอบด้วยแบบจำลองทางคณิตศาสตร์

3.1.1.1 โปรแกรม MATLAB R2020a สำหรับดำเนินการ Genetic Algorithm

3.1.1.2 โปรแกรม Simulink R2020a สำหรับจำลองการทำงานของหุ่นยนต์

3.1.1.3 โปรแกรม CATIA สำหรับการออกแบบแบบจำลองของหุ่นยนต์

3.1.2 การทดสอบกับแบบจำลองหุ่นยนต์ช่วยเดิน

3.1.2.1 Microcontroller Arduino Nano จำนวน 1 ตัว

3.1.2.2 มอเตอร์ไฟฟ้ากระแสตรง JGB37-520 พร้อมเกียร์ทดรอบอัตราส่วน 1:270

จำนวน 2 ตัว

3.1.2.3 แหล่งจ่ายไฟฟ้ากระแสตรง 6V 30A จำนวน 1 ตัว

3.1.2.4 พลาสติก PLA

3.1.2.5 เครื่องพิมพ์ 3 มิติ Flashforge Guider 2

3.2 การออกแบบการวิจัย

การออกแบบงานวิจัยการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน แบ่งออกเป็น 5 ส่วน ได้แก่ การออกแบบแบบจำลองหุ่นยนต์ช่วยเดินต้นแบบ การออกแบบระบบควบคุมการเดินของหุ่นยนต์ การสร้างแบบจำลองทางคณิตศาสตร์ของชุดโครงหุ่นยนต์ การปรับค่า Gain ของระบบควบคุมด้วยวิธีการของ Ziegler-Nichols และการปรับค่า Gain ของระบบควบคุมด้วย Genetic Algorithm

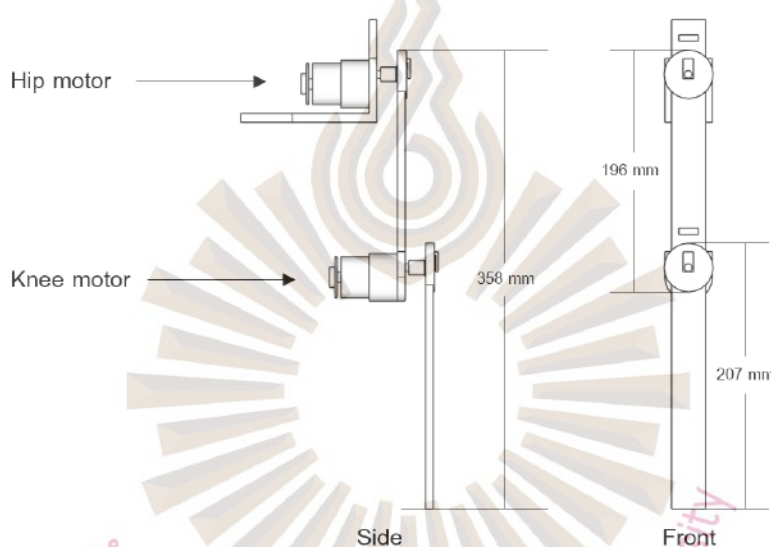
3.2.1 แบบจำลองหุ่นยนต์ช่วยเดินต้นแบบ

ชุดโครงหุ่นยนต์ที่ใช้เป็นต้นแบบสำหรับงานวิจัยนี้ เป็นโครงหุ่นยนต์ลักษณะ Overground Exoskeleton ที่มี 4 Degree of freedom ควบคุมการเคลื่อนไหวที่ข้อสะโพกและข้อเข่าในแนว Sagittal ดังรูปที่ 3.1 ขาแต่ละข้างออกแบบให้มีข้อต่อ 2 ข้อ ที่สะโพกและข้อเข่า โดยที่ข้อต่อแต่ละส่วนจะยึดกับ Brushless motor ข้อต่อละตัว รวมเป็นทั้งหมด 4 ตัว โดยมอเตอร์นี้จะเป็นตัวส่งกำลังไปยังส่วนขาที่ยึดติดกับข้อต่อนั้น โดยที่ตัวมอเตอร์จะมี Absolute Encoder อยู่ที่มอเตอร์ เพื่อใช้บอกตำแหน่งของข้อต่อในการควบคุมการเคลื่อนไหว โครงสร้างประกอบด้วย 3 ส่วน ส่วนแรกคือแกนอลูมิเนียมด้านใน เลือกใช้เป็นอลูมิเนียมทำเป็นโครงชุดซึ่งมีคุณสมบัติแข็งแรงและน้ำหนักเบาโดยมีน้ำหนักรวมประมาณ 15 กิโลกรัม ส่วนที่สองคือมอเตอร์สำหรับขับเคลื่อน ส่วนเปลือกนอกที่มาจากคาร์บอนไฟเบอร์ 3 มิติ มีไว้เพื่อปกปิดส่วนที่เป็นโครงชุด วงจรไฟฟ้า และสายสัญญาณต่างๆ ทำให้ดูเป็นระเบียบเรียบร้อย



รูปที่ 3.1 ชุดโครงหุ่นยนต์ที่ใช้เป็นต้นแบบในงานวิจัย

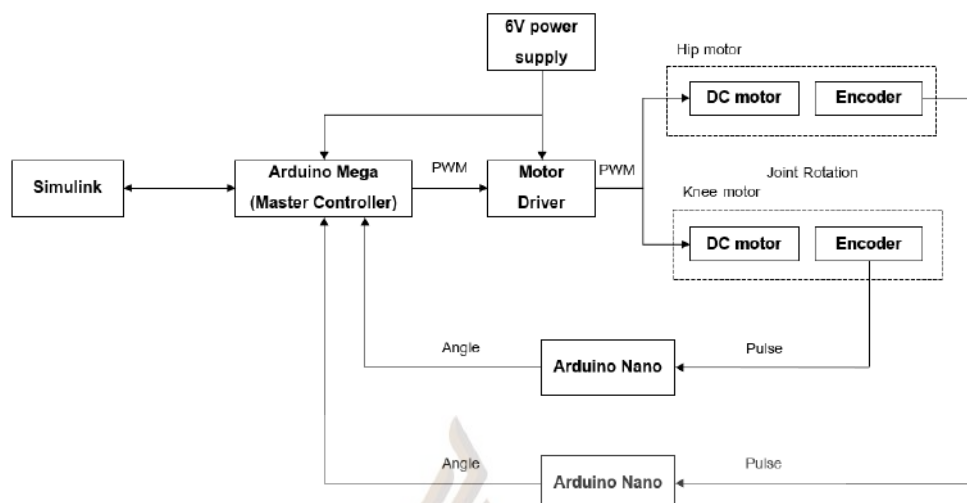
ผู้จัดทำใช้ลักษณะของขาแต่ละข้างของหุ่นยนต์นี้ในการสร้างแบบจำลองของขาหุ่นยนต์จำนวน 1 ข้าง ซึ่งจะสามารถเคลื่อนที่ได้อย่างอิสระใน 2 แกน (2 Degree of Freedom) ที่ข้อเข่าและข้อสะโพกในแนว Sagittal Plane ข้อต่อทั้งสองประกอบด้วยมอเตอร์ไฟฟ้ากระแสตรงขนาด 6V 20RPM ซึ่งแกนของมอเตอร์ต่อเข้ากับชุดเกียร์ทดรอบขนาด 1:270 จำนวน 2 ตัว สำหรับการควบคุมการเคลื่อนที่ของข้อสะโพกและข้อเข่า โครงสร้างของหุ่นยนต์ผลิตจากพลาสติก PLA ที่ได้จากการพิมพ์ด้วยเครื่องพิมพ์ 3 มิติ ซึ่งมีความยาว 196 และ 207 มิลลิเมตรตามลำดับ โดยลักษณะของหุ่นยนต์แสดงในรูปที่ 3.2 ดังนี้



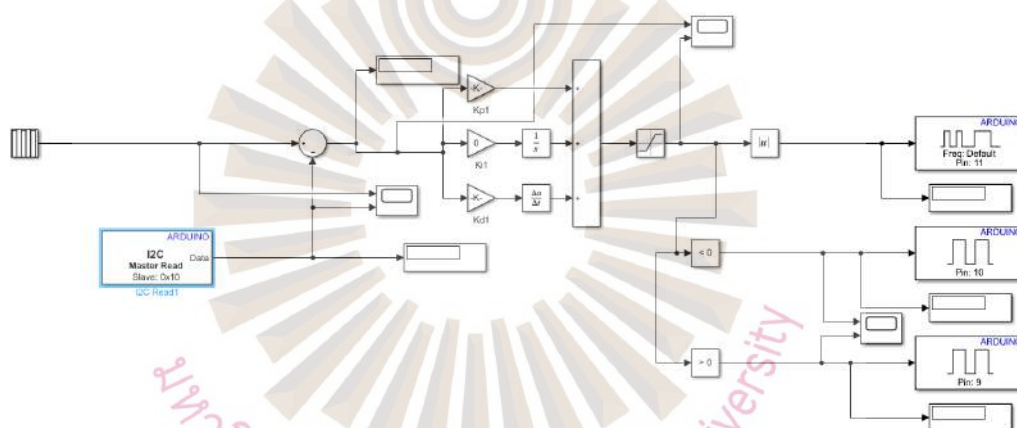
รูปที่ 3.2 แบบจำลองขาหุ่นยนต์ที่ใช้ในงานวิจัย

การทำงานของแบบจำลองหุ่นยนต์นี้ จะใช้บอร์ดพัฒนา Arduino Mega ในการควบคุมการทำงานของมอเตอร์ไฟฟ้าทั้งหมด โดยแสดงแผนผังของ Hardware ของระบบควบคุมของหุ่นยนต์ในรูปที่ 3.3

รูปที่ 3.3 แสดง Block Diagram ของระบบควบคุมของแบบจำลองหุ่นยนต์ โดยวงจรทั้งหมดได้รับไฟฟ้าจากแหล่งจ่ายไฟขนาด 6 โวลต์ ระบบทั้งหมดควบคุมและติดตามผลผ่านระบบที่ออกแบบผ่าน MATLAB Simulink ดังแสดงในรูปที่ 3.4 โดยใช้ Arduino Mega เป็นหน่วยควบคุมกลางของระบบควบคุม โดยควบคุมให้มอเตอร์ไฟฟ้ากระแสตรงในส่วนสะโพกและเข่าเคลื่อนที่ไปยังตำแหน่งที่กำหนด การเคลื่อนที่ของข้อต่อจะถูกบันทึกโดย Encoder ที่อยู่ที่แกนหมุน โดยข้อมูลจาก Encoder จะถูกแปลงเป็นมุมของข้อต่อด้วย Arduino Nano ซึ่งจะส่งข้อมูลตำแหน่งของข้อต่อไปยังหน่วยประมวลผลกลางผ่านระบบ I2C เพื่อใช้ในการควบคุมการเคลื่อนที่ต่อไป



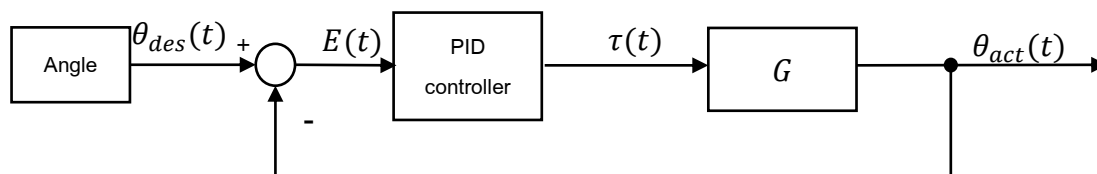
รูปที่ 3.3 แผนผังของระบบควบคุมขาหุ่นยนต์ที่ใช้ในงานวิจัย



รูปที่ 3.4 ระบบควบคุมขาหุ่นยนต์ด้วย MATLAB Simulink ที่ใช้ในงานวิจัย

3.2.2 ระบบควบคุมการเดินของหุ่นยนต์

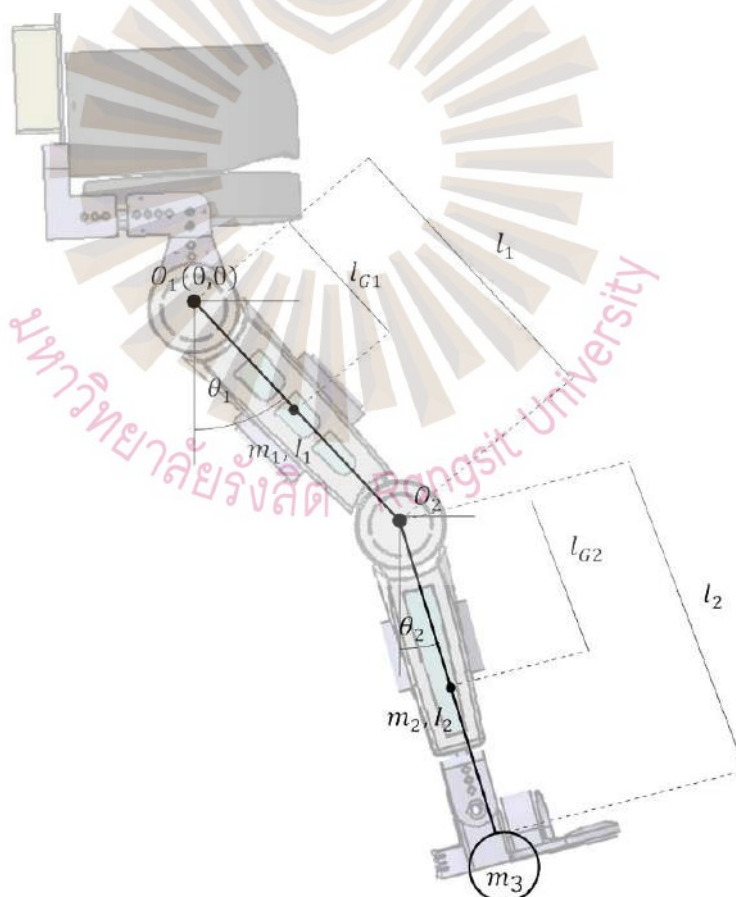
รูปที่ 3.5 แสดงแผนผังของระบบควบคุมการเคลื่อนที่ของหุ่นยนต์ช่วยเหลือการเดิน ซึ่งเป็นระบบควบคุมแบบ PID ระบบปิด โดย Input ของระบบคือมุมของข้อต่อที่ระบบต้องการ ($\theta_{des}(t)$) และ output คือมุมของข้อต่อในขณะนั้น ($\theta_{act}(t)$) โดยระบบควบคุม PID จะมี Input เป็นผลต่างระหว่างมุมของข้อต่อในขณะนั้นและมุมของข้อต่อที่ระบบต้องการ ($E(t)$) และ Output ออกมาเป็นแรงบิด ($\tau(t)$) ให้มอเตอร์กระทำต่อข้อต่อนั้นๆ



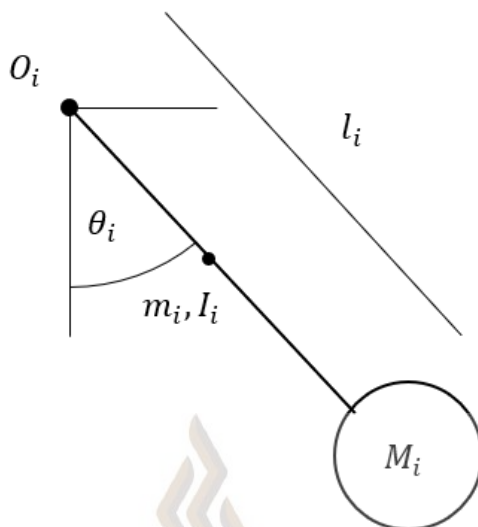
รูปที่ 3.5 ระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน

3.2.3 แบบจำลองทางคณิตศาสตร์ของชุดโครงหุ่นยนต์

จากชุดโครงหุ่นยนต์ที่ใช้ในงานวิจัย ขาของหุ่นยนต์แต่ละข้างสามารถแสดงอยู่ในรูปของ Free Body Diagram ได้ดังรูปที่ 3.6 โดยที่จุด O_1 และ O_2 แทนตำแหน่งข้อสะโพกและข้อเข่า ตามลำดับ l_1 และ l_2 แทนความยาวของต้นขาและปลายขาตามลำดับ และ l_{G1} และ l_{G2} แทนระยะทางของต้นขาและปลายขาระหว่างข้อต่อและจุดศูนย์กลางมวล ตามลำดับ



รูปที่ 3.6 Free Body Diagram ของขาแต่ละข้าง



รูปที่ 3.7 Pendulum model ของขาแต่ละท่อน

การเคลื่อนที่ของขาหุ่นยนต์แต่ละส่วนสามารถจำลองออกมาในรูปแบบ Pendulum Model ได้ดังรูปที่ 3.7 เพื่อหาสมการทางคณิตศาสตร์สำหรับการจำลองการเคลื่อนที่ของชุดโครงหุ่นยนต์ กระทำโดยการสร้างสมการการเคลื่อนที่ของขาโดยผลรวมของแรงบิดรอบจุด O_i ได้ดังสมการ 3-1 (Amiri et al., 2019)

$$G_i(s) = \frac{\theta_i(s)}{\tau_i(s)} = \frac{1}{s^2 + \frac{b_i}{I_i}s + \frac{(M_i + m_i)gl_{Gi}}{I_i}} \quad (3-1)$$

ผู้จัดทำได้ใช้โปรแกรม CATIA V5R21 ในการวัดขนาดและคำนวณมวลและโมเมนต์ความเฉื่อยของขาแต่ละท่อน โดยได้ค่าดังตารางที่ 1 ต่อไปนี้

ตารางที่ 3.1 ลักษณะของขาแต่ละท่อน

	ต้นขา	ปลายขา	เท้า
มวล (kg)	2.263	1.468	0.750
ความยาว (m)	0.412	0.512	
Moment of inertia (kgm ²)	0.053	0.018	

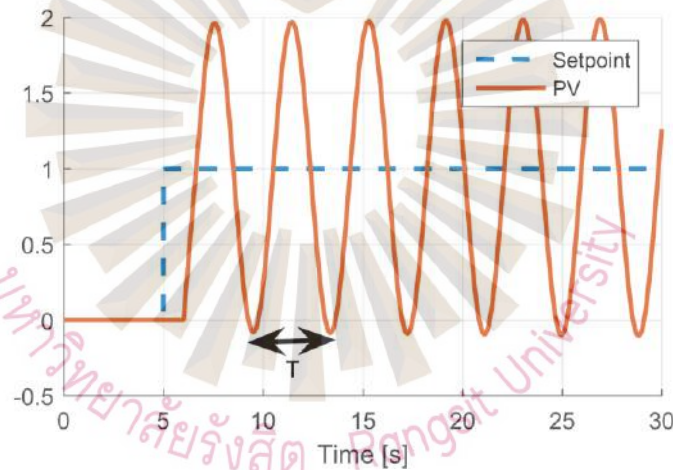
เมื่อนำข้อมูลจากตามตารางที่ 1 แทนลงในสมการที่ 3-1 ได้ Transfer Function ของต้นขา (G_1) และปลายขา (G_2) ดังนี้

$$G_1(s) = \frac{\theta_1(s)}{\tau_1(s)} = \frac{1}{s^2 + 7.55s + 677.48} \quad (3-2)$$

$$G_2(s) = \frac{\theta_2(s)}{\tau_2(s)} = \frac{1}{s^2 + 22.22s + 414.10} \quad (3-3)$$

3.2.4 การปรับค่า Gain ของระบบควบคุมด้วยวิธีการของ Zeigler-Nichols

เพื่อหาค่า Gain ของระบบควบคุมแบบ PID ที่จะใช้ในการกำหนดช่วงของการทำ Genetic Algorithm จึงทำการปรับค่า Gain ด้วยวิธีการของ Zeigler-Nichols โดยเริ่มต้นด้วยการกำหนด K_i และ K_d เป็น 0 และแทนค่า K_p ด้วย K_c โดยเพิ่มค่าของ K_c ขึ้นจนเกิดการสั่นของ output ของระบบ ควบคุมในลักษณะดังแสดงในรูปที่ 3.8 ดำเนินการวัดคาบ (T) ของการสั่นภายใน output ที่ได้นี้ ออกมา ค่า K_p , K_i , และ K_d ของระบบควบคุม PID นั้น สามารถหาได้จาก K_c และคาบ (T) ด้วย สมการที่ 3-4, 3-5, และ 3-6 ดังต่อไปนี้ (Amiri et al, 2020)



รูปที่ 3.8 การสั่นของ Output ของระบบควบคุมในการปรับค่าด้วยวิธีของ Ziegler-Nichols

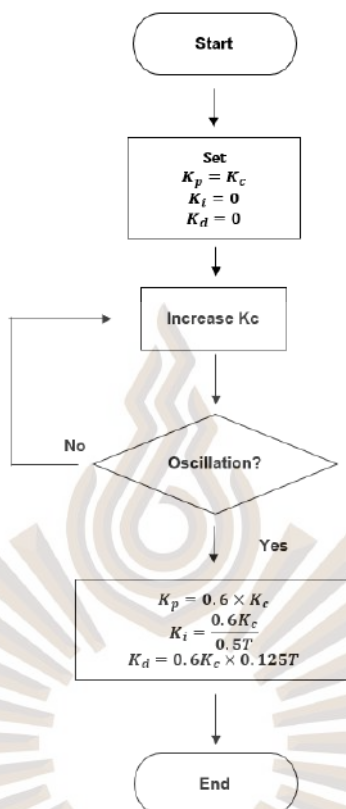
ที่มา: Kalsbeek, 2017

$$K_p = 0.6 \times K_c \quad (3-4)$$

$$K_i = \frac{0.6K_c}{0.5T} \quad (3-5)$$

$$K_d = 0.6K_c \times 0.125T \quad (3-6)$$

รูปที่ 3.9 แสดง flowchart ของการปรับค่าของระบบควบคุม PID ด้วยวิธีของ Zeigler-Nichols ในงานวิจัยนี้

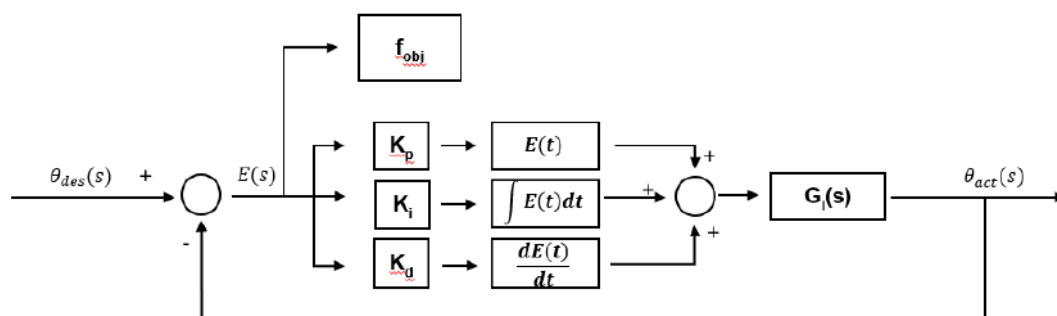


รูปที่ 3.9 flowchart ของการปรับค่า Gain ด้วยวิธีการของ Zeigler-Nichols

3.2.5 การปรับค่า Gain ของระบบควบคุมด้วย Genetic Algorithm

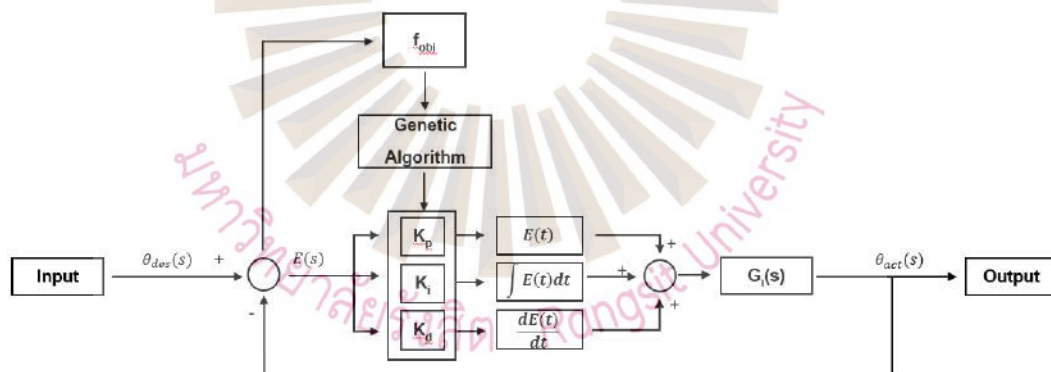
Genetic Algorithm เป็นหลักการ Optimization ที่อาศัยการเลียนแบบการคัดเลือกทางธรรมชาติในวิวัฒนาการของสิ่งมีชีวิต โดยในงานวิจัยนี้ Objective function (f_{obj}) หรือเป้าหมายของการทำ Optimization คือการปรับความผิดพลาดของระบบควบคุม PID ให้เข้าใกล้ 0 ที่สุด โดยในงานวิจัยนี้จะใช้วิธีการประเมินความผิดพลาดด้วยการใช้ Integral Time-Absolute Error (ITAE) เป็นเกณฑ์ในการปรับค่าของระบบควบคุม (Tan Marquez and Chen, 2004) โดยแสดงการประเมินค่าความผิดพลาดด้วยรูปที่ 3.10 และสมการที่ 3-7

$$f_{obj} = \int_0^{\infty} t|E(t)|dt \quad (3-7)$$



รูปที่ 3.10 การหา Objective function

ในขั้นตอนแรกของ Genetic Algorithm นั้น ระบบจะสร้างชุดข้อมูล หรือ “ยีน” ที่ประกอบด้วยค่า Gain ทั้ง 3 ค่าขึ้นมาด้วยการสุ่ม จำนวน 50 ชุด ซึ่งยีนเหล่านี้เปรียบเสมือนประชากรรุ่นพ่อแม่ ค่าของยีนจากประชากรเริ่มต้นนี้จะถูกนำไปจำลองผ่านระบบควบคุมที่ได้ออกแบบไว้เพื่อทำการประเมิน Objective Function ของระบบควบคุมจากการใช้ค่ายีนของระบบควบคุม PID จากยีนต่างๆ ดังรูปที่ 3.11 โดยนำค่าที่ได้มาใช้จัดเรียงเรียงยีนในประชากรตามขนาดของ Objective Function เพื่อนำมาใช้ในการสร้างประชากรรุ่นลูกต่อไป



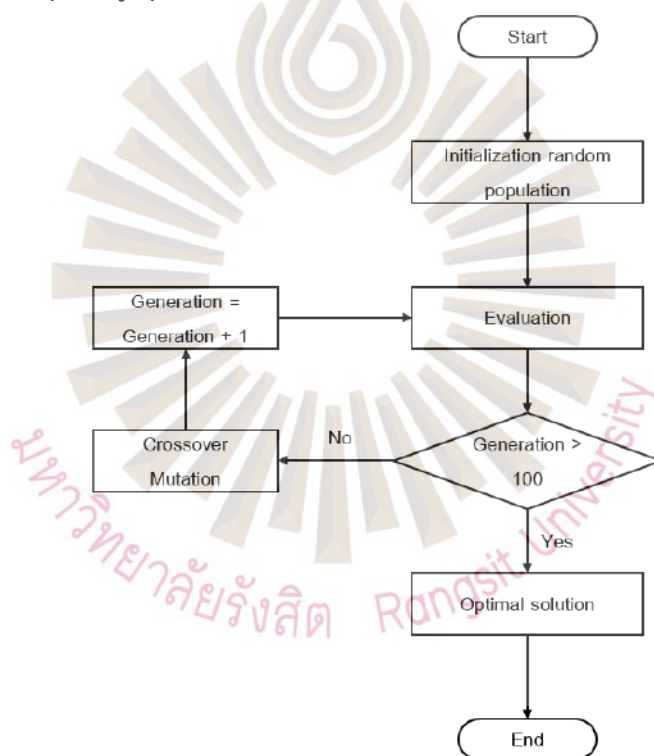
รูปที่ 3.11 กระบวนการใช้ Genetic Algorithm ในการหา Objective Function

จากระบบควบคุมแบบ PID

การสร้างประชากรรุ่นลูกนั้น จะอาศัยกระบวนการ Crossover และ Mutation โดยประชากรที่ได้ค่า ITAE ต่ำที่สุด 5% ของประชากรทั้งหมดจะถูกจัดว่าเป็นยีน Elite เพื่อรักษาชุดข้อมูลที่ได้ผลลัพธ์ดีที่สุดไว้ สำหรับชุดข้อมูลที่เหลืออื่น จะทำการผ่าน Crossover และ Mutation โดยกำหนดให้อัตราการ Crossover และ Mutation อยู่ที่ 0.8 และ 0.01 ของประชากรที่ไม่ใช่ Elite Genes ตามลำดับ ประชากรรุ่นลูกที่ได้จะใช้เป็นประชากรรุ่นพ่อแม่ชุดใหม่ที่จะผ่านกระบวนการนี้ซ้ำ ๆ

จนกระทั่งครบจำนวนรอบที่กำหนดไว้ โดยในงานวิจัยนี้ กำหนดให้จำนวนรอบของการทำ Genetic Algorithm มีจำนวน 100 รอบ

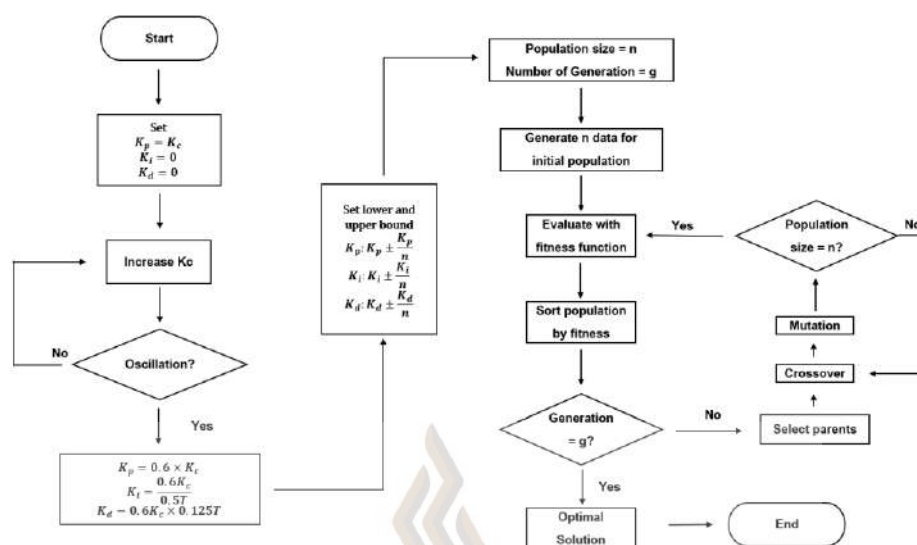
ในงานวิจัยนี้ มีการทำ Genetic Algorithm 2 วิธี โดยในตอนแรกจะกำหนดช่วงของการสุ่มค่า Gain มาเป็นชุดแรกตั้งแต่ 0 ถึง 3000 และในตอนที่สองจะกำหนดช่วงของการสุ่มข้อมูลรุ่นแรกตามค่า Gain ที่ได้จากการปรับค่าด้วยวิธีของ Ziegler-Nichols โดยการปรับค่าในตอนที 2 นั้น กำหนดให้หยุดการทำงานเมื่อได้ค่า Gain ที่มีความแม่นยำเท่ากับหรือดีกว่าการปรับด้วย Genetic Algorithm อย่างเดียว รูปที่ 3.12 แสดง flowchart ของการทำงานของ Genetic Algorithm และรูปที่ 3.23 แสดงการทำงานของ Genetic Algorithm โดยใช้การปรับค่าด้วยวิธีของ Ziegler Nichols ในการกำหนดช่วงของการสุ่มข้อมูลรุ่นแรก



รูปที่ 3.12 flowchart ของการทำงานของ Genetic Algorithm

3.3 การทดสอบการทำงานของระบบควบคุม

การทดสอบการทำงานของระบบควบคุม แบ่งออกเป็น 2 ส่วน ได้แก่ การทดสอบการทำงานด้วย MATLAB Simulink และการทดสอบการทำงานกับแบบจำลองหุ่นยนต์ช่วยเดิน

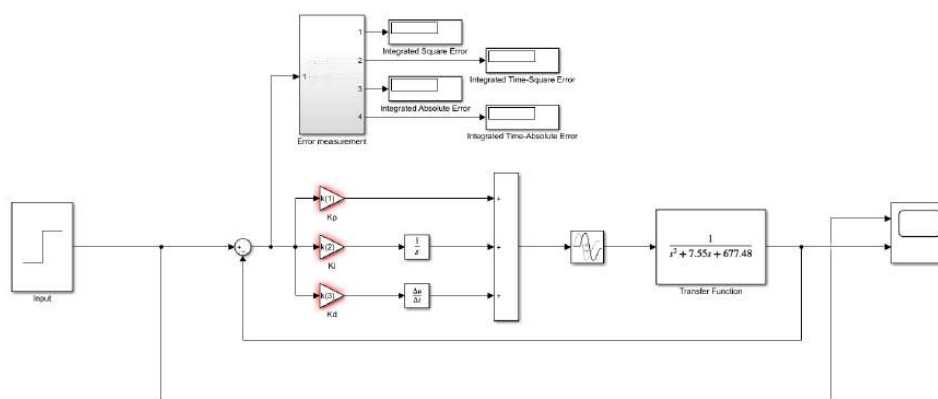


รูปที่ 3.13 flowchart ของการทำงานของ Genetic Algorithm

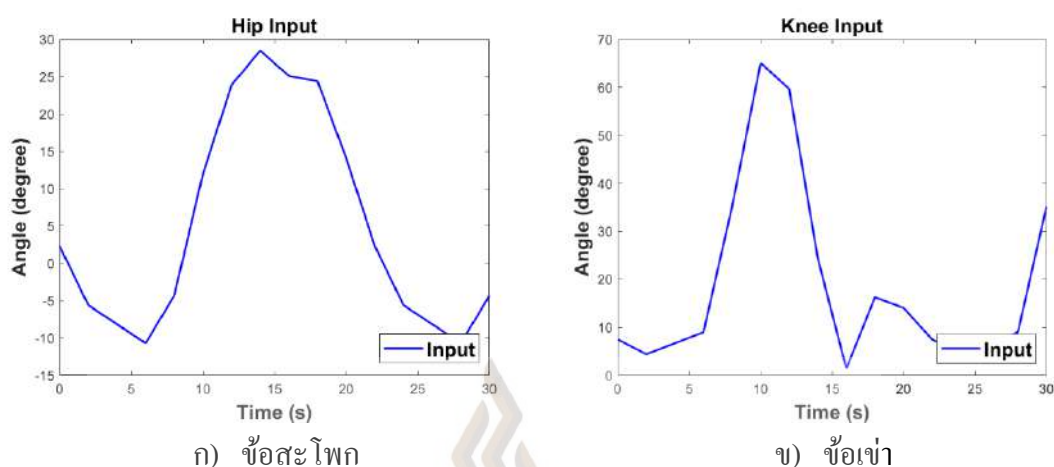
โดยใช้การปรับค่าด้วยวิธีของ Ziegler Nichols

3.3.1 การทดสอบการทำงานด้วย Simulink

การทดสอบการทำงานของระบบควบคุมที่ได้มาแต่ละวิธีจะถูกทดสอบโดยใช้ MATLAB Simulink โดยจัดแบบจำลองระบบควบคุมเพื่อหาความผิดพลาดดังรูปที่ 3.14 โดยค่า K_p , K_i , K_d ของระบบควบคุมนั้นใช้ค่าที่ได้จากการปรับค่าแต่ละวิธี การทดสอบในขั้นตอนแรกจะใช้ input เป็น step response และประเมินประสิทธิภาพของระบบควบคุมแต่ละวิธีด้วยการวัด Settling Time, Rise time, Overshoot, และ Steady-state error ของระบบควบคุมที่ได้จากการปรับค่าแต่ละวิธี และในตอนที่สองจะใช้ Input เป็นข้อมูลการเดินจากการศึกษาการเดินของมนุษย์ (Fukuchi et al., 2018) ซึ่งแสดงในรูปที่ 3.15 เพื่อเปรียบเทียบความแม่นยำของระบบควบคุมที่ได้จากทั้งสามวิธี



รูปที่ 3.14 แบบจำลองระบบควบคุมใน MATLAB Simulink



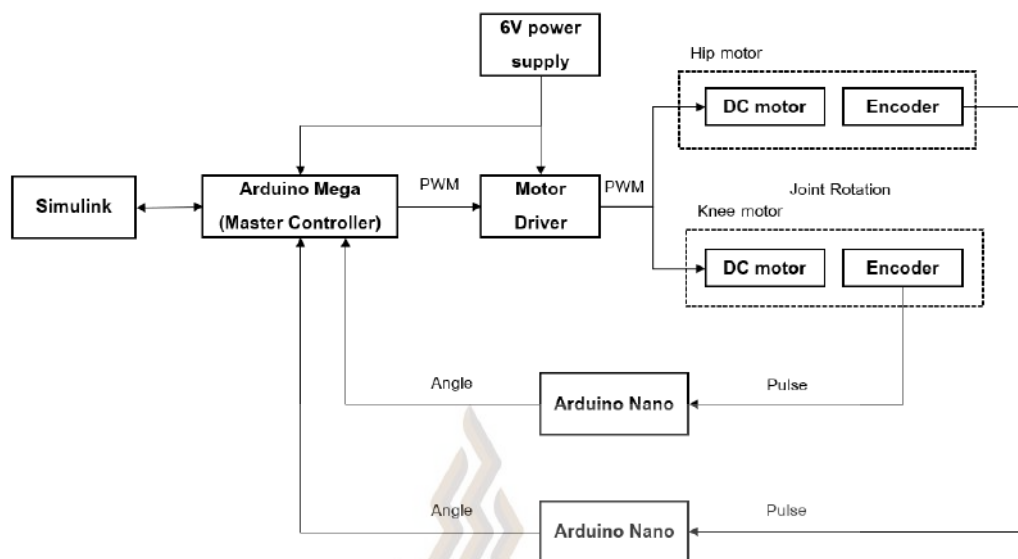
รูปที่ 3.15 ข้อมูลมุมของข้อต่อของขามนุษย์ระหว่างการเดินที่ใช้ในการทดสอบระบบควบคุม

3.3.2 การทดสอบการทำงานกับแบบจำลองหุ่นยนต์ช่วยเดิน

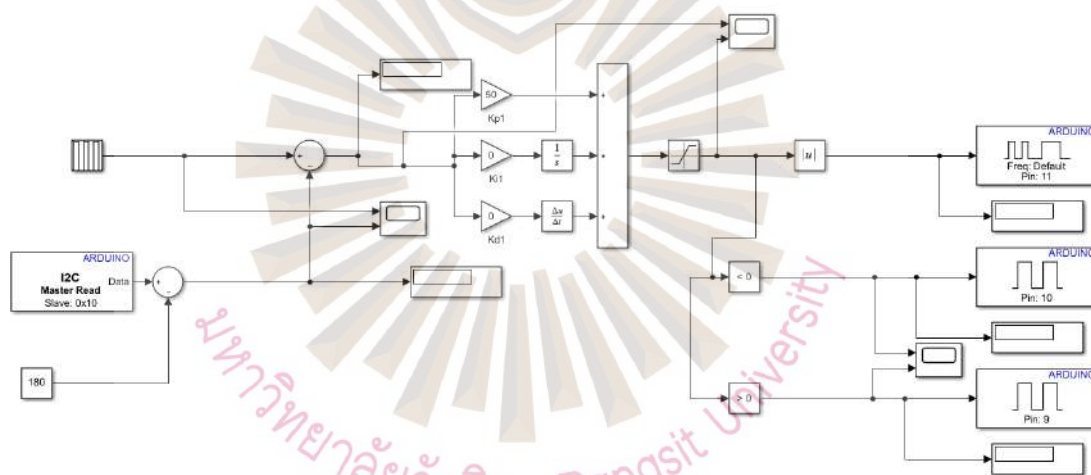
การทดลองในตอนนี้กระทำกับแบบจำลองขาหุ่นยนต์ช่วยเดินดังที่ออกแบบในตอน 3.3.1 ดังแสดงในรูปที่ 3.16 โดยปรับค่าของระบบควบคุม PID ด้วยวิธีการใช้ Genetic Algorithm และใช้ Simulink ในการเขียนโปรแกรมควบคุมการทำงานของหุ่นยนต์ โดยใช้ Simulink ในการเก็บผลการทำงานของระบบควบคุม และใช้ MATLAB ในการประเมินผล โดยรูปที่ 3.17 แสดงแผนผังของระบบควบคุมที่ใช้ในการทำงานของหุ่นยนต์ และรูปที่ 3.18 แสดงโปรแกรม Simulink ที่ใช้ควบคุมการทำงานของหุ่นยนต์



รูปที่ 3.16 แบบจำลองขาหุ่นยนต์ช่วยเดิน



รูปที่ 3.17 แผนผังของระบบควบคุมที่ใช้ในการทำงานของหุ่นยนต์



รูปที่ 3.18 ระบบควบคุมภายใน Simulink

บทที่ 4

ผลการวิจัย

ผลการทดลองของงานวิจัยการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือนการเดิน แบ่งออกได้เป็นสองส่วน ได้แก่ ผลการทดลองจากการจำลองด้วย MATLAB Simulink และผลการทดลองจากแบบจำลองขาหุ่นยนต์ช่วยเดิน

4.1 การจำลองด้วย MATLAB Simulink

ผู้จัดทำทำการหาค่า Gain สำหรับระบบควบคุม PID ของหุ่นยนต์ช่วยเหลือนการเดินทั้งส่วน สะโพกและขาโดยใช้วิธีของ Zeigler-Nichols และ Genetic Algorithm โดยตารางที่ 4.1 สรุปค่า Gain ของระบบควบคุม PID ที่ได้จากทั้งสามวิธีออกมา พร้อมกับเวลาที่ใช้ในการประมวลผลในการทำ Genetic Algorithm แต่ละขั้นตอน

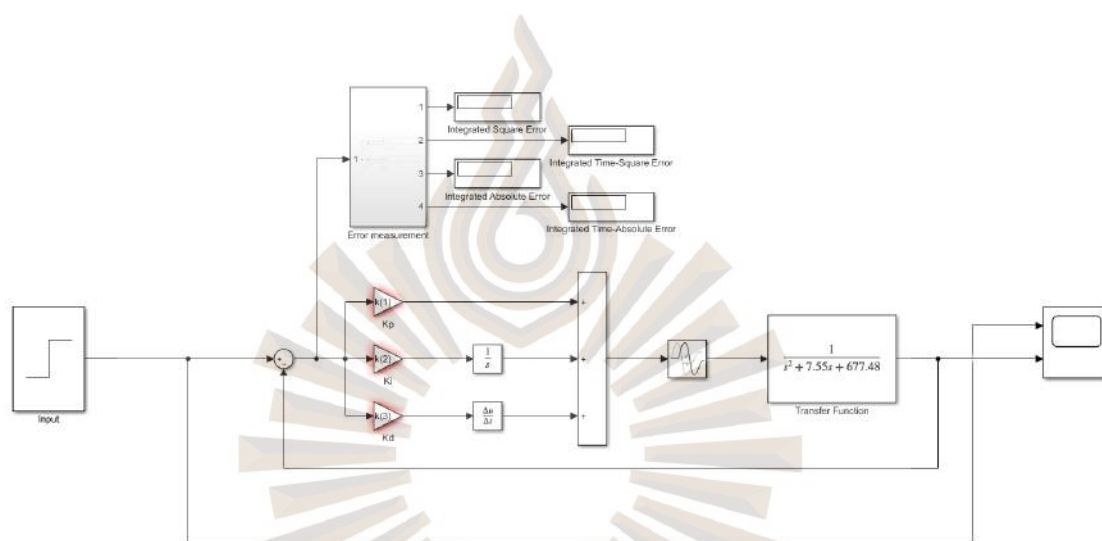
ระบบควบคุมที่ออกแบบมาถูกจำลองและทดสอบโดยใช้ MATLAB Simulink โดยจัดแบบจำลองระบบควบคุมเพื่อหาความผิดพลาดดังรูปที่ 4.1 โดยค่า K_p , K_i , K_d ของระบบควบคุมนั้นใช้ค่าที่คำนวณมาได้ตามตารางที่ 4.1 และประเมินประสิทธิภาพของระบบควบคุมแต่ละวิธีด้วยการวัด Settling Time, Rise time, Overshoot, และ Steady-state error ของระบบควบคุมที่ได้จากการปรับค่าแต่ละวิธีโดยการทดสอบในขั้นตอนแรกจะใช้ input เป็น step response ในขั้นตอนที่สอง จะใช้ input เป็นข้อมูลการเดินจากการศึกษาการเดินของมนุษย์ (Fukuchi et al., 2018) ซึ่งแสดงในรูปที่ 4.2 ผลการทดสอบเมื่อ Input เป็น Step Input แสดงในรูปที่ 4.3 และ 4.4 และตารางที่ 4.2 และ 4.3 และผลการทดสอบเมื่อ Input เป็นข้อมูลการเดินแสดงในรูปที่ 4.5 และ 4.6

ตารางที่ 4.1 ค่า Gain ของระบบควบคุม PID

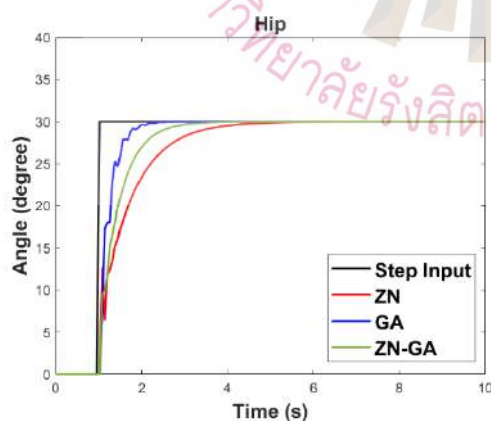
	Kp	Ki	Kd	Time (s)
สะโพก				
ZN	147	1075.6071	5.0225	-
GA	8.1563	2782.2	0.0572	325.8
ZN-GA	73.5	1613.4	2.5158	295.1

ตารางที่ 4.1 ค่า Gain ของระบบควบคุม PID (ต่อ)

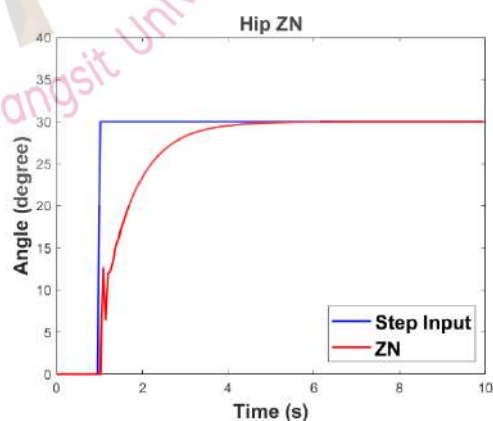
	Kp	Ki	Kd	Time (s)
เข้า				
ZN	318	1848.837	13.674	-
GA	152.42	2042.6	4.1054	320.8
ZN-GA	183.96	2519.9	10.031	5.9175



รูปที่ 4.1 การทดสอบแบบจำลองระบบควบคุมใน MATLAB Simulink

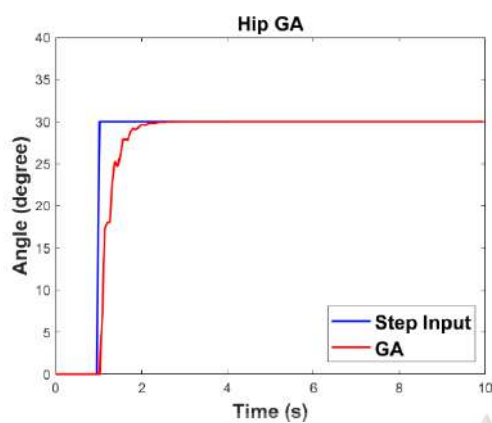


ก) ผลลัพธ์ของระบบควบคุมทั้งหมด

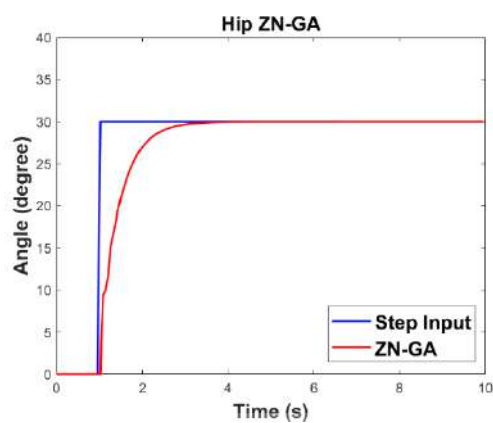


ข) Ziegler-Nichols

รูปที่ 4.2 Output ของระบบควบคุมส่วนสะโพกเมื่อ Input คือ step response

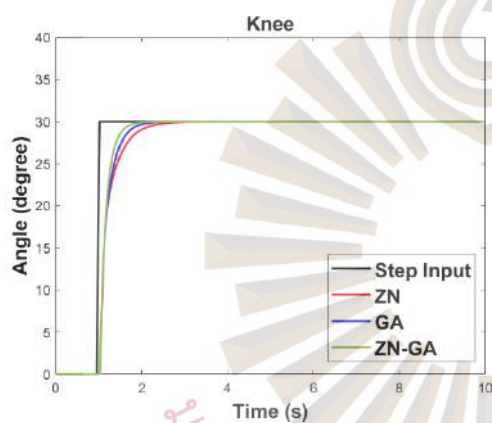


ค) GA

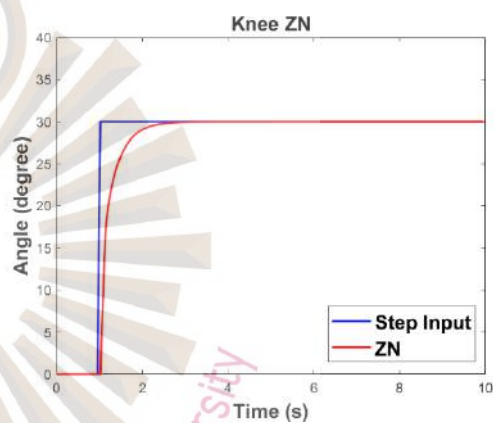


ง) ZN-GA

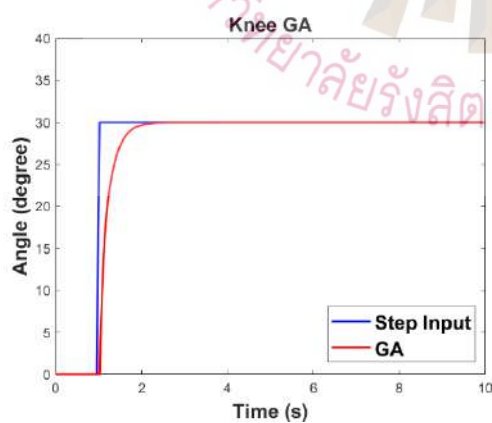
รูปที่ 4.2 Output ของระบบควบคุมส่วนสะโพกเมื่อ Input คือ step response (ต่อ)



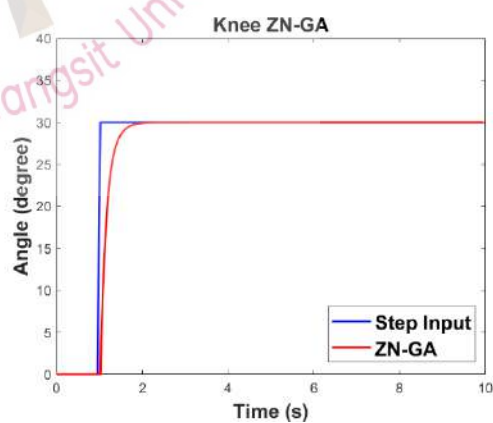
ก) ผลลัพธ์ของระบบควบคุมทั้งหมด



ข) Ziegler-Nichols

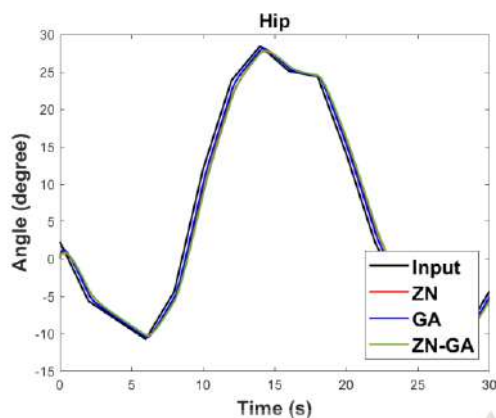


ค) GA

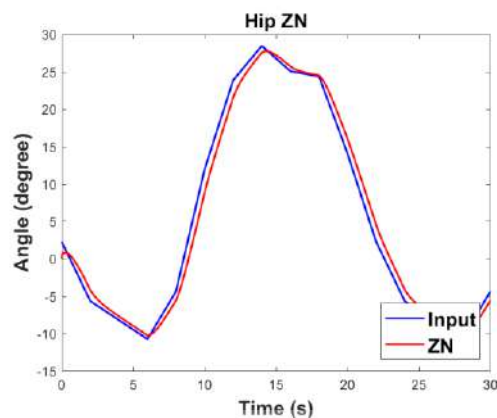


ง) ZN-GA

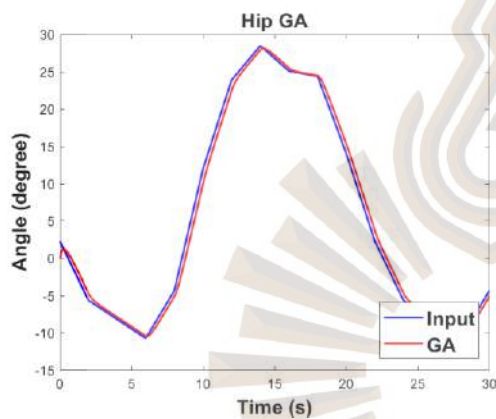
รูปที่ 4.3 Output ของระบบควบคุมส่วนเข่าเมื่อ Input คือ step response



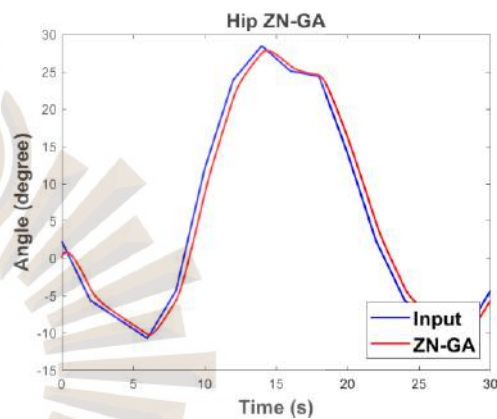
ก) ผลลัพธ์ของระบบควบคุมทั้งหมด



ข) Ziegler-Nichols

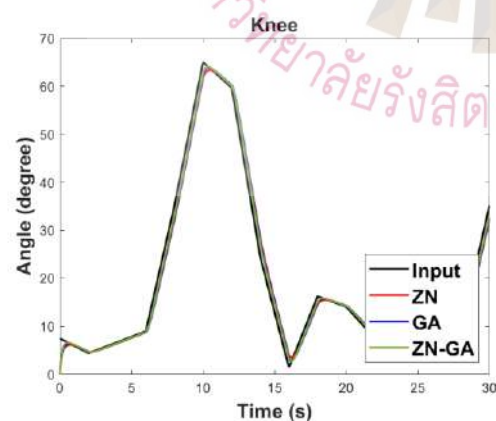


ค) GA

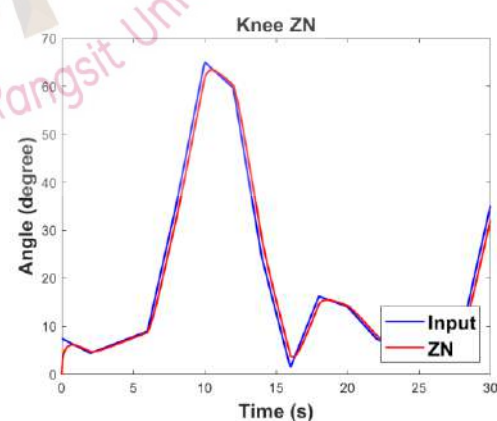


ง) ZN-GA

รูปที่ 4.4 Output ของระบบควบคุมส่วนสะโพกเมื่อ Input เป็นข้อมูลการเดิน

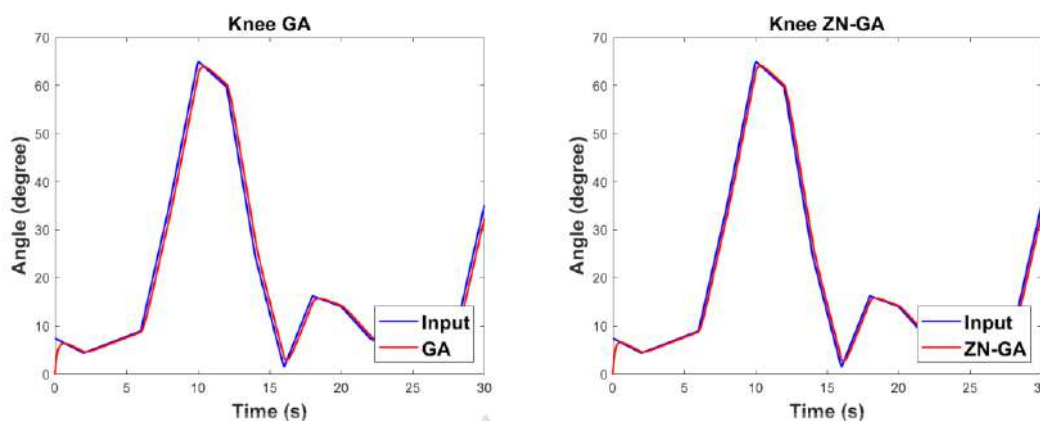


ก) ผลลัพธ์ของระบบควบคุมทั้งหมด



ข) Ziegler-Nichols

รูปที่ 4.5 Output ของระบบควบคุมส่วนเข่าเมื่อ Input เป็นข้อมูลการเดิน



ค) GA

ง) ZN-GA

รูปที่ 4.5 Output ของระบบควบคุมส่วนเข้าเมื่อ Input เป็นข้อมูลการเดิน (ต่อ)

ตารางที่ 4.2 ระบบควบคุมส่วนสะโพกเมื่อ Input เป็น Step Response

	Rise time (s)	Settling time (s)	Overshoot	Steady-state error
ZN	1.5589	3.5041	0.0000	0.0001
GA	0.4876	1.9407	0.0000	0.0000
ZN-GA	0.9640	2.7348	0.0000	0.0000

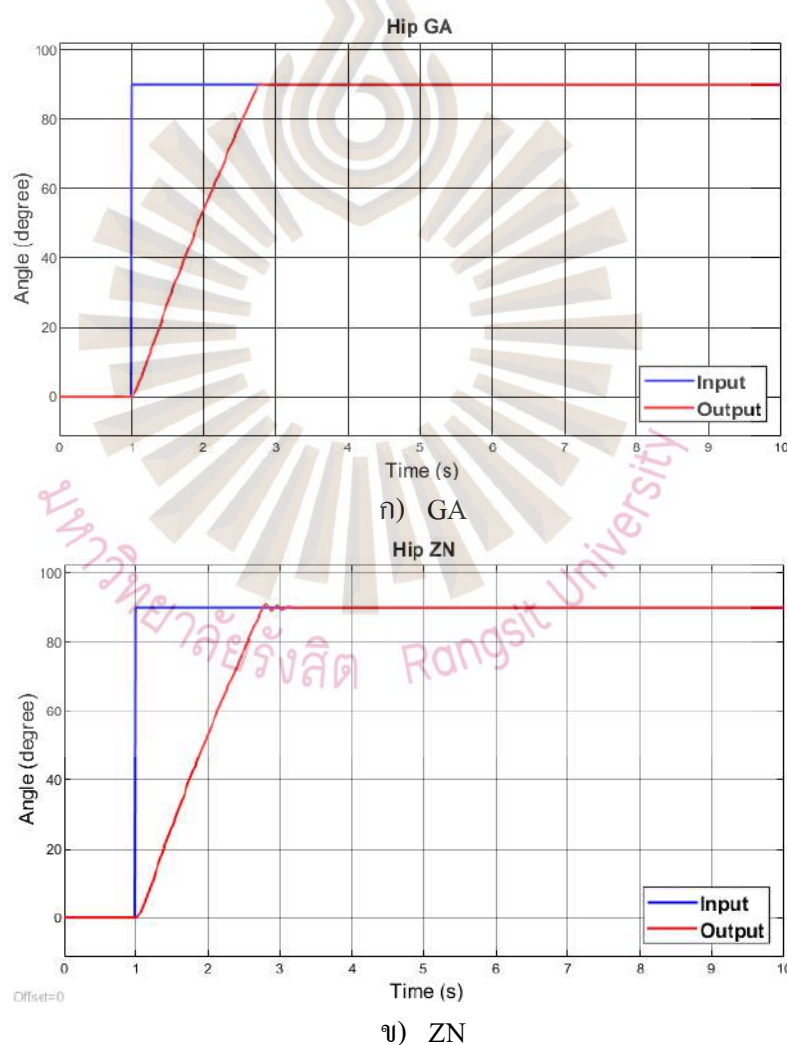
ตารางที่ 4.3 ระบบควบคุมส่วนเข้าเมื่อ Input เป็น Step Response

	Rise time (s)	Settling time (s)	Overshoot	Steady-state error
ZN	0.5656	2.155	0.0000	0.0000
GA	0.4462	1.8515	0.0000	0.0000
ZN-GA	0.3372	1.6543	0.0000	0.0000

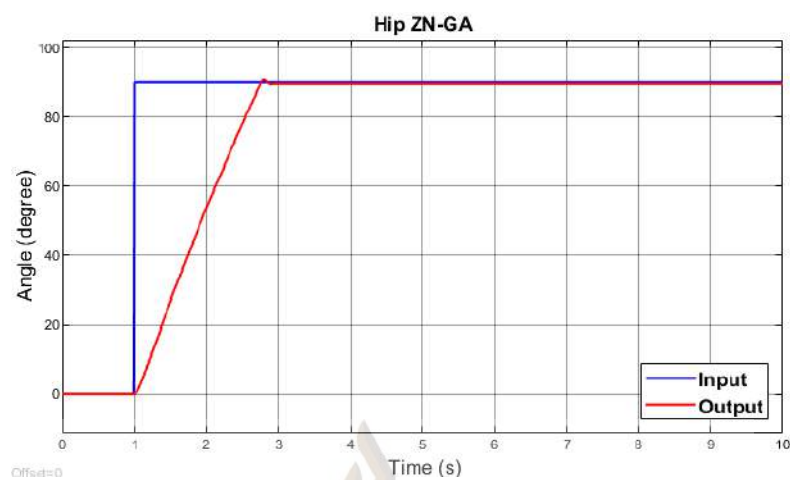
จากข้อมูลในตารางที่ 4.1 พบว่าการปรับค่าด้วย ZN-GA ใช้เวลาในการทำงานน้อยกว่าการปรับค่าด้วย GA เพียงอย่างเดียว ตารางที่ 4.2 และ 4.3 แสดงผลการทดสอบการทำงานของระบบควบคุมทั้ง 3 แบบของทั้งส่วนสะโพกและส่วนเข้า ซึ่งแสดงให้เห็นว่าการใช้ GA เพียงอย่างเดียว ให้ระบบควบคุมที่ใช้เวลา Settling Time ที่เร็วกว่าการปรับค่าด้วย ZN แต่การปรับค่าโดยใช้ ZN นั้นให้ระบบควบคุมที่มี Rise Time เร็วกว่าระบบควบคุมที่ปรับค่าด้วย GA โดยระบบควบคุมที่ปรับค่าโดยใช้ ZN-GA นั้นให้ระบบควบคุมที่มีลักษณะอยู่ระหว่างระบบควบคุมอีกสองวิธี โดยมี Rise Time ที่เร็วกว่าระบบควบคุมที่ปรับค่าด้วย GA และใช้เวลาเข้าสู่ Settling Time เร็วกว่าระบบควบคุมที่ปรับค่าด้วย ZN

4.2 การทดลองกับแบบจำลองขาหุ่นยนต์ช่วยเดิน

การทดลองในตอนนี้กระทำกับแบบจำลองขาหุ่นยนต์ช่วยเดินดังที่ออกแบบในตอน 3.2.6 โดยปรับค่าของระบบควบคุม PID ด้วยวิธีการใช้ Genetic Algorithm และใช้ Simulink ในการเขียนโปรแกรมควบคุมการทำงานของหุ่นยนต์ โดยใช้ Simulink ในการเก็บผลการทำงานของระบบควบคุม และประเมินผลโดยใช้ MATLAB รูปที่ 4.6, 4.7, และตารางที่ 4.4 แสดงผลการทดสอบการทำงานของระบบควบคุมเมื่อให้ Step input เป็น Input และรูปที่ 4.8, 4.9, และตารางที่ 4.5 แสดงผลการทดสอบการทำงานของระบบควบคุมเมื่อให้ข้อมูลการเดินเป็น Input ของระบบควบคุม

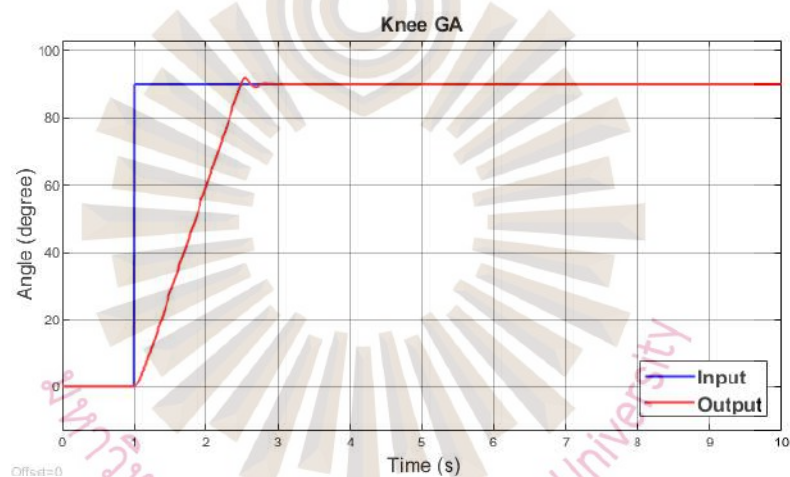


รูปที่ 4.6 ผลการทดสอบการทำงานของระบบควบคุมส่วนสะโพกเมื่อ Input เป็น Step Input

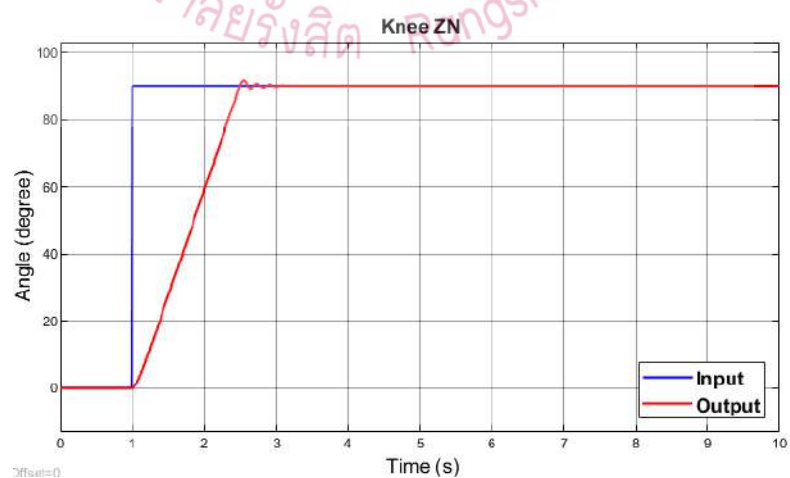


ค) ZN-GA

รูปที่ 4.6 ผลการทดสอบการทำงานของระบบควบคุมส่วนสะโพกเมื่อ Input เป็น Step Input (ต่อ)

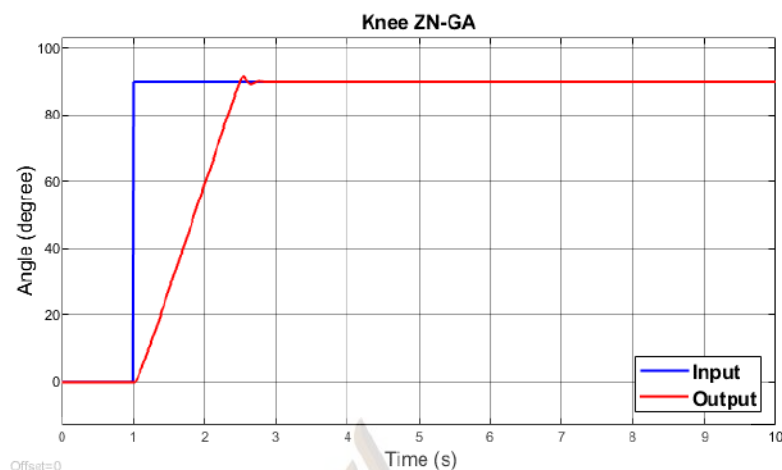


ก) GA



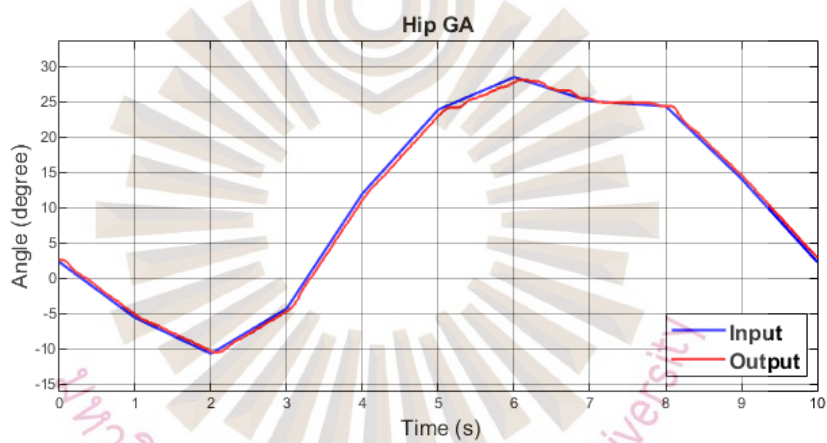
ข) ZN

รูปที่ 4.7 ผลการทดสอบการทำงานของระบบควบคุมส่วนเข่าเมื่อ Input เป็น Step Input



ค) ZN-GA

รูปที่ 4.7 ผลการทดสอบการทำงานของระบบควบคุมส่วนเข้าเมื่อ Input เป็น Step Input (ต่อ)

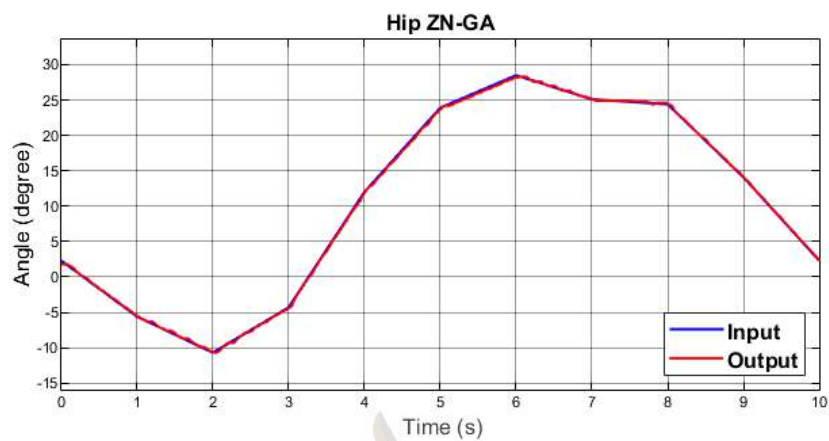


ก) GA



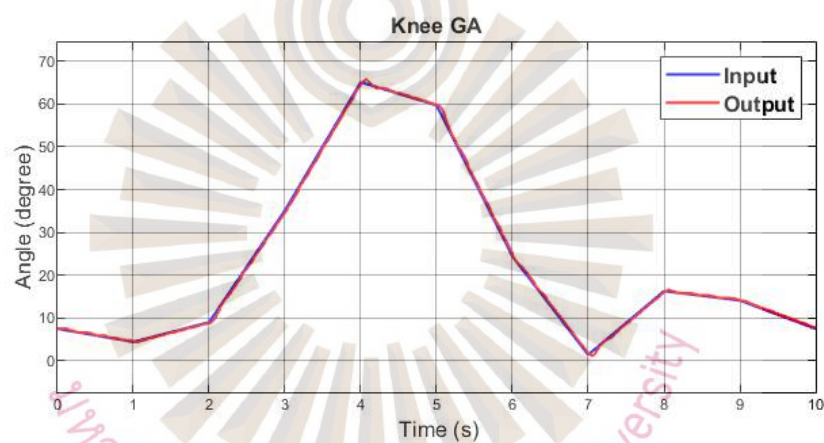
ข) ZN

รูปที่ 4.8 ผลการทดสอบการทำงานของระบบควบคุมส่วนสะโพกเมื่อ Input เป็นข้อมูลการเดิน



ค) ZN-GA

รูปที่ 4.8 ผลการทดสอบการทำงานของระบบควบคุมส่วนสะโพกเมื่อ Input เป็นข้อมูลการเดิน (ต่อ)

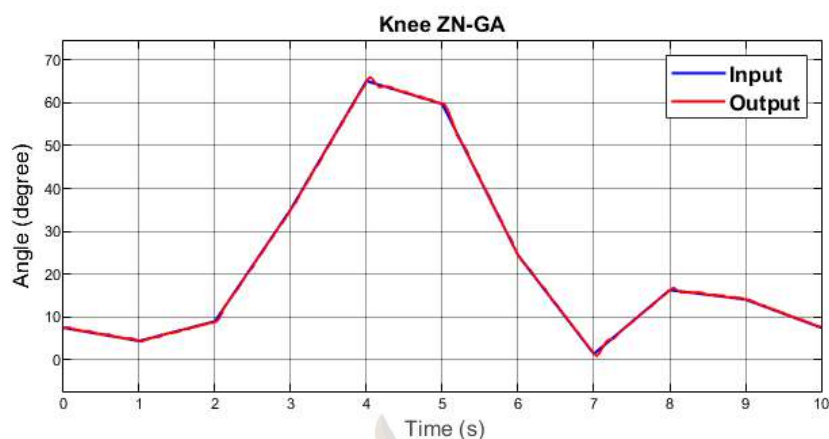


ก) GA



ข) ZN

รูปที่ 4.9 ผลการทดสอบการทำงานของระบบควบคุมส่วนเข่าเมื่อ Input เป็นข้อมูลการเดิน



ก) ZN-GA

รูปที่ 4.9 ผลการทดสอบการทำงานของระบบควบคุมส่วนเข้าเมื่อ Input เป็นข้อมูลการเดิน (ต่อ)

ตารางที่ 4.4 ผลการทดสอบระบบควบคุมของแบบจำลองขาหุ่นยนต์เมื่อใช้ Input เป็น Step Input

	Integral time-absolute error	
	Hip	Knee
ZN	125.9	106.2
GA	124.1	105.0
ZN-GA	120.7	105.4

ตารางที่ 4.5 ผลการทดสอบระบบควบคุมของแบบจำลองขาหุ่นยนต์เมื่อใช้ Input เป็นข้อมูลการเดิน

	Integral time-absolute error	
	Hip	Knee
ZN	6.96	8.38
GA	25.10	17.85
ZN-GA	6.73	8.57

จากผลการทดลองที่แสดงในรูปที่ 34 ถึง 37 และตารางที่ 4.4 และ 4.5 พบว่าระบบควบคุมแบบ ZN-GA สามารถควบคุมให้แบบจำลองขาหุ่นยนต์ช่วยเดินสามารถทำการเดินได้ โดยพบว่าในระบบควบคุมส่วนเข้า ZN-GA ให้ระบบควบคุมที่มี ITAE น้อยที่สุด แต่ในส่วนสะโพก ZN-GA ให้ระบบควบคุมที่มี ITAE น้อยที่สุดเฉพาะในการทดสอบที่ Input เป็น Step Input ในขณะที่ ZN ให้

ระบบควบคุมที่ ITAE ต่ำกว่าเล็กน้อยเมื่อใช้ ZN ปรับค่าระบบควบคุม ทั้งนี้ พบว่าเมื่อ Input เป็น ข้อมูลการเดิน การใช้ GA ในการปรับค่าให้ระบบควบคุมที่ ITAE มากที่สุด



บทที่ 5

สรุปผลและข้อเสนอแนะ

งานวิจัยการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน มีข้อสรุปของงานวิจัยและข้อเสนอแนะสำหรับการพัฒนาและศึกษาต่อในอนาคต ดังนี้

5.1 สรุปผลการวิจัย

ในงานวิจัยนี้ ได้ทำการปรับค่า Gain ของระบบควบคุมแบบ PID สำหรับหุ่นยนต์ช่วยเหลือการเดิน 4 DoF ด้วยการปรับค่าด้วยวิธีการของ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการลดความผิดพลาดของระบบควบคุม ค่า Gain ของระบบ PID ที่ได้ (K_p, K_i, K_d) ถูกนำไปทดสอบด้วยแบบจำลองภายใน MATLAB Simulink โดยมี Input เป็น Step Input และตำแหน่งของข้อต่อของขามนุษย์ในระหว่างการเดิน ผลการทดสอบพบว่าการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับค่าใช้เวลาในการทำงานน้อยกว่าการใช้ Genetic Algorithm เพียงอย่างเดียว และให้ระบบควบคุมทำให้ได้ระบบควบคุมที่มี Settling time และ Overshoot ต่ำกว่าการใช้วิธีการ Ziegler-Nichols ในการปรับค่า โดยผลการจำลองด้วย MATLAB Simulink ถูกยืนยันโดยการทดสอบด้วยแบบจำลองขาของหุ่นยนต์ช่วยเหลือเดิน

5.2 ข้อเสนอแนะ

งานวิจัยการใช้ Ziegler-Nichols ร่วมกับการใช้ Genetic Algorithm ในการปรับระบบควบคุมของหุ่นยนต์ช่วยเหลือการเดิน มีข้อเสนอแนะในการพัฒนางานวิจัยต่อไปในอนาคต 2 ประเด็น ได้แก่ การทดสอบการทำงานของระบบควบคุมกับแบบจำลองหุ่นยนต์ช่วยเหลือเดิน และการตั้งค่าการทำงานของ Genetic Algorithm ในการปรับค่าของระบบควบคุม

5.2.1 การทดสอบกับหุ่นยนต์ช่วยเดิน

ในการทดสอบการทำงานของระบบควบคุมของหุ่นยนต์ช่วยเดินส่วนขา นั้น ในการออกแบบแบบจำลองทางคณิตศาสตร์ใช้ต้นแบบมาจากหุ่นยนต์ที่มีลักษณะโครงสร้างเพื่อการใช้กับมนุษย์ แต่เนื่องด้วยข้อจำกัดด้านเวลา และเนื่องจากจุดประสงค์ของงานวิจัยนี้เป็นการทดสอบการสร้างระบบควบคุม จึงทำการทดสอบระบบควบคุมกับแบบจำลองขาหุ่นยนต์ช่วยเดินที่มีขนาดเล็กกว่าความเป็นจริง การสร้างหุ่นยนต์ช่วยเดินต้นแบบตามที่ออกแบบไว้จึงเป็นสิ่งที่จำเป็นในการศึกษาในอนาคต

5.2.2 การปรับค่าด้วย Genetic Algorithm

ในงานวิจัยนี้ การทำ Genetic Algorithm ใช้ฟังก์ชัน ga ของ MATLAB โดยมีการตั้งค่าให้จำนวนประชากรต่อรอบมีทั้งหมด 50 ชุด และจำนวน Generation ที่ 100 รอบ ในส่วนของอัตราการเกิด Crossover, Mutation, และการเก็บ Elite Gene นั้นใช้ค่าเริ่มต้นภายใน MATLAB ซึ่งกำหนดให้อัตราการเกิด Crossover ที่ 80%, อัตราการเกิด Mutation ที่ 1% และเก็บ Elite Gene 5% ของประชากรทั้งหมด ซึ่งการปรับการตั้งค่า Genetic Algorithm นี้ส่งผลต่อผลลัพธ์ที่ได้ออกมา ซึ่งมีผลต่อการทำงานของระบบควบคุม จึงเป็นประเด็นที่ต้องมีการศึกษาต่อในอนาคตว่าการตั้งคุณสมบัติต่างๆ มีผลต่อระบบควบคุมได้อย่างไร

บรรณานุกรม

- Amiri, M. S., Ramli, R., & Ibrahim, M. F. (2019). Hybrid design of PID controller for four DoF lower limb exoskeleton. *Applied Mathematical Modelling*, 72, 17-27.
<https://doi.org/https://doi.org/10.1016/j.apm.2019.03.002>
- Chen, B., Ma, H., Qin, L. Y., Gao, F., Chan, K. M., Law, S. W., Qin, L., & Liao, W. H. (2016). Recent developments and challenges of lower extremity exoskeletons. *J Orthop Translat*, 5, 26-37. <https://doi.org/10.1016/j.jot.2015.09.007>
- Chen, G., Chan, C. K., Guo, Z., & Yu, H. (2013). A review of lower extremity assistive robotic exoskeletons in rehabilitation therapy. *Crit Rev Biomed Eng*, 41(4-5), 343-363.
<https://doi.org/10.1615/critrevbiomedeng.2014010453>
- Elbayomy, K. M., Zongxia, J., & Huaqing, Z. (2008). PID Controller Optimization by GA and Its Performances on the Electro-hydraulic Servo Control System. *Chinese Journal of Aeronautics*, 21(4), 378-384. [https://doi.org/https://doi.org/10.1016/S1000-9361\(08\)60049-7](https://doi.org/https://doi.org/10.1016/S1000-9361(08)60049-7)
- Fukuchi, C. A., Fukuchi, R. K., & Duarte, M. (2018). A public dataset of overground and treadmill walking kinematics and kinetics in healthy individuals. *PeerJ*, 6, e4640.
<https://doi.org/10.7717/peerj.4640>
- Hussain, F., Goecke, R., & Mohammadian, M. (2021). Exoskeleton robots for lower limb assistance: A review of materials, actuation, and manufacturing methods. *Proc Inst Mech Eng H*, 235(12), 1375-1385. <https://doi.org/10.1177/09544119211032010>
- Ma'arif, A., Nabila, H., Iswanto, & Wahyunggoro, O. (2019). Application of Intelligent Search Algorithms in Proportional-Integral-Derivative Control of Direct-Current Motor System. *Journal of Physics: Conference Series*, 1373(1), 012039. <https://doi.org/10.1088/1742-6596/1373/1/012039>
- Molteni, F., Gasperini, G., Cannaviello, G., & Guanziroli, E. (2018). Exoskeleton and End-Effector Robots for Upper and Lower Limbs Rehabilitation: Narrative Review. *PM R*, 10(9 Suppl 2), S174-S188. <https://doi.org/10.1016/j.pmrj.2018.06.005>

บรรณานุกรม (ต่อ)

- Simonsen, E. B. (2014). Contributions to the understanding of gait control. *Dan Med J*, 61(4), B4823. <https://www.ncbi.nlm.nih.gov/pubmed/24814597>
- Soleimani Amiri, M., Ramli, R., Ibrahim, M. F., Abd Wahab, D., & Aliman, N. (2020). Adaptive Particle Swarm Optimization of PID Gain Tuning for Lower-Limb Human Exoskeleton in Virtual Environment. *Mathematics*, 8(11). <https://doi.org/10.3390/math8112040>
- Suseno, E., & Ma'arif, A. (2021). Tuning of PID Controller Parameters with Genetic Algorithm Method on DC Motor. *International Journal of Robotics and Control Systems*, 1, 41-53. <https://doi.org/10.31763/ijrcs.v1i1.249>
- Tan, W., Marquez, H. J., & Chen, T. (2004). Performance Assessment of PID Controllers. *Control and Intelligent Systems*, 32. <https://doi.org/10.2316/Journal.201.2004.3.201-1309>



ประวัติผู้วิจัย

ชื่อ	ศิวกร ปั่นกระจำง
วัน เดือน ปีเกิด	28 มีนาคม 2539
สถานที่เกิด	กรุงเทพมหานคร ประเทศไทย
ประวัติการศึกษา	มหาวิทยาลัยรังสิต ปริญญาวิศวกรรมศาสตรบัณฑิต วิศวกรรมชีวการแพทย์, 2566
ทุนงานวิจัย ที่อยู่ปัจจุบัน	ทุนสนับสนุนการทำวิจัยบัณฑิตศึกษา มหาวิทยาลัยรังสิต 529 ซอย 29 หมู่บ้านสีวลีรังสิต 2 ถนนรังสิต-นครนายก ตำบลประชาธิปัตย์ อำเภอัญมรี จังหวัดปทุมธานี 12130

