การสังเคราะห์โปรแกรมควบคุมหุ่นยนต์เดินสองขา แบบสมดุลสถิตโดยอัตโนมัติ ด้วยการคำนวณเชิงวิวัฒน์

นาย ชาญชัย ชัยสุขโกศล

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2544

ISBN 974-03-0207-6

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

AUTOMATIC SYNTHESIS OF ROBOT PROGRAMS FOR A BIPED STATIC WALKER BY EVOLUTIONARY COMPUTATION

Mr. Chanchai Chaisukkosol

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2001

ISBN 974-03-0207-6

หัวข้อวิทยานิพนธ์	การสังเคราะห์โปรแกรมควบคุมหุ่นยนต์เดินสองขา แบบสมดุลสถิตโดยอัตโนมัติ ด้วยการคำนวณเชิงวิวัฒน์
โดย	นาย ชาญชัย ชัยสุขโกศล
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(อาจารย์ ดร.ฐิต ศิริบูรณ์)

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

..... กรรมการ
(รองศาสตราจารย์ ดร.ชิต เหล่าวัฒนา)

..... กรรมการ
(อาจารย์ ดร.วีระ เหมืองสิน)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ชาญชัย ชัยสุขโกศล: การสังเคราะห์โปรแกรมควบคุมของหุ่นยนต์เดินสองขา แบบสมดุลสถิตโดยอัตโนมัติ ด้วยการคำนวณเชิงวิวัฒน์ (AUTOMATIC SYNTHESIS OF ROBOT PROGRAMS FOR A BIPED STATIC WALKER BY EVOLUTIONARY COMPUTATION) อาจารย์ที่ปรึกษา: ผศ.ดร.ประภาส จงสถิตย์วัฒนา; 57 หน้า. ISBN 974-03-0207-6

วิทยานิพนธ์ฉบับนี้เสนอวิธีการของการสังเคราะห์โปรแกรมควบคุมหุ่นยนต์ โดยอัตโนมัติ ด้วยวิธีการคำนวณเชิงวิวัฒน์ ปัญหาการเดินของหุ่นยนต์สองขาถูกเลือกใช้ในการทดสอบข้อเสนอนี้ ท่าทางการเดินถูกแบ่งออกเป็นขั้นตอนย่อย 6 ขั้นตอน การคำนวณเชิงวิวัฒน์ถูกประยุกต์ใช้ในการหาคำตอบย่อยของแต่ละขั้นตอน และนำคำตอบย่อยของแต่ละขั้นตอนมาประกอบรวมกันเป็นคำตอบสุดท้าย การแบ่งปัญหาออกเป็นการหาคำตอบย่อย ๆ นั้นทำให้หาคำตอบได้ง่ายขึ้น เนื่องจากสามารถกำหนดเกณฑ์การวัดค่าความเหมาะสมให้เหมาะสมและแตกต่างกันไปได้ในแต่ละขั้นตอน งานวิจัยนี้จำกัดขอบเขตของการเดินอยู่ที่การเดินไปข้างหน้าบนพื้นผิวที่ราบและเรียบด้วยสมดุลสถิต ระบบของการทดลองใช้การทดลองในสภาพแวดล้อมจำลองประกอบกับการทดลองกับหุ่นยนต์จริง เพื่อลดเวลาในการทดลองและทำให้คำตอบที่ได้สามารถใช้ได้กับหุ่นยนต์จริง จากผลการทดลองแสดงให้เห็นว่าสามารถได้ท่าทางการเดินที่หลากหลาย อีกทั้งหุ่นยนต์ยังรักษาสมดุลของการเดินได้อย่างต่อเนื่องอีกด้วย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่อนิสิต .....
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา	2544	ลายมือชื่ออาจารย์ที่ปรึกษาร่วม .....

CHANCHAI CHAISUKKOSOL: AUTOMATIC SYNTHESIS OF ROBOT PROGRAMS FOR A BIPED STATIC WALKER BY EVOLUTIONARY COMPUTATION. THESIS ADVISOR: PRABHAS CHONGSTITVATANA. 57 pp. ISBN 974-03-0207-6

The thesis proposes a method to automatically synthesise the robot control programs by Evolutionary Computation. The problem of biped robot walking is chosen to test the proposed method. Walking motion is divided into six stages. Evolutionary Computation is carried out stage-by-stage and sub-solutions of all stages are combined to produce a final solution. Dividing the problem into searching for sub-solutions makes it easier to find the solution, because the fitness function can be set appropriately and differently in each stage. The work restricts walking locomotion to the forward walking on the flat and smooth surface with static balance. The experimental system uses simulation and the experiment with a real robot to reduce the time and to produce the solutions that work with the real robot. The result of the experiment shows that various types of gaits are achievable and the stability of biped walking is maintained continuously.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

Department	Computer Engineering	Student ’s signature .....
Field of study	Computer Engineering	Advisor ’s signature .....
Academic year	2001	Co-advisor ’s signature .....

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลงได้ด้วยดีนั้น ผู้ที่มีส่วนช่วยเหลือมากที่สุดที่ผู้วิจัยอยากจะขอบพระคุณเป็นบุคคลแรกคือ อาจารย์ประภาส จงสถิตย์วัฒนา ซึ่งคอยดูแลให้คำชี้แนะหาทางแก้ไขปัญหาต่าง ๆ ทั้งปัญหาเชิงเทคนิค ปัญหาเชิงแนวคิด คอยให้คำปรึกษาเรื่องชีวิตการเรียน อีกทั้งยังคอยกระตุ้นให้ผู้วิจัยได้ตั้งหน้าตั้งตาทำงานอย่างมีสมาธิ ไม่วอกแวก ถึงแม้หลายครั้งอาจารย์จะไม่เห็นด้วยการแนวคิดเกี่ยวกับการใช้ชีวิตการเรียนในระดับปริญญาโทของผู้วิจัยก็ตาม แต่อาจารย์ท่านก็ยังยอมรับให้ผู้วิจัยเป็นผู้เลือกทางของตัวเอง ผู้วิจัยขอขอบพระคุณอาจารย์ฐิต ศิริบูรณ์, อาจารย์วีระ เหมืองสิน และอาจารย์ชิต เหล่าวัฒนา ที่ให้ความกรุณาเป็นกรรมการสอบวิทยานิพนธ์ให้แก่ผู้วิจัย ขอขอบคุณพี่ ๆ ที่ห้องปฏิบัติการทุกคนที่ให้คำแนะนำทั้งในรายละเอียดเล็ก ๆ น้อย ๆ ในการทำงานและข้อคิดต่าง ๆ ในการดำเนินชีวิตนักศึกษาที่นี่ ขอขอบคุณบุคคลอีกมากมายที่ยังไม่ได้กล่าวถึง ซึ่งล้วนมีส่วนช่วยให้ผู้วิจัยสามารถบรรลุงานวิจัยชิ้นนี้ได้

ท้ายที่สุด ขอขอบคุณนางสาวศจินทร์ ประชาสันติ์ ที่คอยเป็นกำลังใจให้แก่ผู้วิจัยเสมอมาตั้งแต่ช่วงแรก ๆ ของงานวิจัยที่ผู้วิจัยรู้สึกสับสน ท้อแท้ และเหน็ดเหนื่อย จนกระทั่งในงานวิจัยชิ้นนี้สำเร็จลุล่วงไปได้ ขอขอบคุณที่อยู่ข้าง ๆ ผู้วิจัยตลอดมา

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ง

บทคัดย่อภาษาอังกฤษ..... จ

กิตติกรรมประกาศ..... ฉ

สารบัญ..... ช

สารบัญตาราง..... ฌ

สารบัญภาพ..... ญ

บทที่

1. บทนำ..... 1

1.1. ความสำคัญและความเป็นมา..... 1

1.2. วัตถุประสงค์ของการวิจัย..... 2

1.3. ขอบเขตของงานวิจัย..... 2

1.4. ขั้นตอนการวิจัย..... 2

1.5. ประโยชน์ที่คาดว่าจะได้รับจากงานวิจัย..... 3

1.6. ผลงานตีพิมพ์จากงานวิจัย..... 3

2. ทฤษฎีที่เกี่ยวข้อง..... 4

2.1. แนวทางการทำงานของการคำนวณเชิงวิวัฒน์..... 4

2.2. การคัดเลือก..... 5

2.3. การให้กำเนิดกลุ่มคำตอบรุ่นต่อไป..... 7

ก. การไขว้เปลี่ยน 7

ข. การกลายพันธุ์..... 7

2.4. สมดุลของการเดินแบบสองขา..... 8

ก. การเดินแบบสมดุลสถิต (static balance walking) 8

ข. การเดินแบบสมดุลพลวัต (dynamic balance walking) 8

3. งานวิจัยที่เกี่ยวข้อง..... 9

3.1. หุ่นยนต์สองขา..... 9

3.2. ปัญญาประดิษฐ์กับหุ่นยนต์สองขา..... 10

3.3. สรุป..... 12

4. การทดลอง..... 13

4.1. ลักษณะของปัญหา..... 13

หน้า

4.2. หุ่นยนต์ 13
4.2.1 โครงสร้างของหุ่นยนต์สองขา 13
4.2.2 เซอร์โวมอเตอร์ 15
4.2.3 ส่วนการควบคุม 16
4.3. การออกแบบการทดลอง 17
4.3.1 ท่าทางการเดิน 17
4.3.2 โปรแกรมควบคุมการเดิน 18
4.3.3 ระบบของการทดลอง 20
4.3.4 ภาพรวมระบบการทำงาน 25
4.4. ขั้นตอนวิธีพันธุกรรม 26
4.4.1 การเข้ารหัสของคำตอบ (individual representation) 26
4.4.2 การประเมินค่าความเหมาะสม (fitness evaluation) 26
1) เกณฑ์การวัดค่าความเหมาะสมทั่วไป (general fitness function) 27
2) เกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอน (particular fitness function) 28
4.4.3 การคัดเลือก (selection) 33
4.4.4 การไขว้เปลี่ยน (crossover) และการกลายพันธุ์ (mutation) 35
4.5. สภาพแวดล้อมจำลอง (simulation) 37
5. ผลการทดลอง 40
5.1. การทดลองในช่วงต่าง ๆ 40
5.1.1 การทดลองในระยะที่หนึ่ง 40
5.1.2 การทดลองในระยะที่สอง 42
5.1.3 การทดลองในระยะที่สาม 44
5.1.4 การทดลองในระยะที่สี่ 46
5.1.5 สรุปการทดลองทั้ง 4 ระยะ 46
5.2. พฤติกรรมการเดิน 47
5.3. คุณภาพของการเดิน 51
6. บทสรุปและข้อเสนอแนะ 53
6.1. สรุปงานวิจัย 53
6.2. ข้อเสนอแนะ 53
7. รายการอ้างอิง 56
8. ประวัติผู้วิจัย 58

สารบัญตาราง

ตารางที่ หน้า

2.1. แสดงขั้นตอนวิธีของการคัดเลือกแบบ combined rank..... 6

4.1. แสดงลักษณะสายอักขระที่ส่งระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์..... 16

4.2. แสดงเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนของการทดลองในสภาพแวดล้อมจำลอง เบื้องต้น..... 28

4.3. แสดงเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนของการทดลองในสภาพแวดล้อมจำลอง ที่ใช้จริง..... 32

4.4. แสดงสรุปของเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนของการทดลองในสภาพแวดล้อมจำลอง..... 32

4.5. แสดงวิธีการประเมินคำตอบแต่ละตัวของแต่ละขั้นตอนในการทดลองกับหุ่นยนต์จริง... 33

4.6. แสดงขั้นตอนการคิดค่าความแตกต่าง..... 34

4.7. แสดงตัวอย่างการคิดค่าความแตกต่างในขั้นตอนที่สอง..... 35

4.8. แสดงพารามิเตอร์ของขั้นตอนวิธีพันธุกรรม..... 36

5.1. แสดงข้อมูลระยะและมุมเบี่ยงเบนของการเดิน 10 ก้าว 50

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

รูปที่		หน้า
2.1.	แสดงแผนผังการทำงานของขั้นตอนวิธีพันธุกรรม	4
2.2.	แสดงวิธีการให้กำเนิดกลุ่มคำตอบ	7
2.3.	แสดงลักษณะจุดศูนย์ถ่วงจากมุมมองเบื้องบนในขณะเดิน	8
4.1.	แสดงอุปกรณ์ต่างๆที่ใช้ในระบบการทดลอง	13
4.2.	แสดงหุ่นยนต์ที่ใช้ในการทดลอง	14
4.3.	แสดงขอบเขตการหมุนของข้อต่อสะโพก หัวเข่า ข้อเท้า และหาง	14
4.4.	แสดงการควบคุมตำแหน่งของเซอร์โวมอเตอร์ด้วยความกว้างพัลส์ (pulse width modulation)	15
4.5.	แสดงรายละเอียดเกี่ยวกับเซอร์โว	15
4.6.	ขั้นตอนการเดิน 4 ขั้นตอน	17
4.7.	ขั้นตอนการเดิน 6 ขั้นตอน	18
4.8.	แสดงหมายเลขกำกับมอเตอร์แต่ละตัวของหุ่นยนต์	19
4.9.	แสดงความสัมพันธ์ของการทดลองในสภาพแวดล้อมจำลองและสภาพแวดล้อมจริงของแต่ละขั้นตอนของการเดิน	20
4.10.	แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่หนึ่ง	22
4.11.	แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่สอง	22
4.12.	แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่สาม	23
4.13.	แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่หก	24
4.14.	แสดงภาพรวมระบบการทำงานต่าง ๆ	25
4.15.	แสดงลักษณะการเข้ารหัสของคำตอบ	26
4.16.	แสดงการตั้งแกนของหุ่นยนต์	27
4.17.	แสดงตัวอย่างประกอบการคิดค่าความแตกต่างของคำตอบ 2 ตัว	34
4.18.	แสดงตัวอย่างวิธีการไขว้เปลี่ยนในขั้นตอนการเดินที่ 2 - 6	36
4.19.	แสดงภาพตัวอย่างหน้าจอการทำงานของสภาพแวดล้อม	37
5.1.	แสดงลักษณะการถ่ายน้ำหนักในพื้นที่รับน้ำหนักของระยะการก้าวสองแบบ	41
5.2.	แสดงกราฟค่าเฉลี่ยของค่าความเหมาะสมทั่วไปและค่าความเหมาะสมเฉพาะของทั้ง 6 ขั้นตอน	43
5.3.	แสดงภาพตัวอย่างคำตอบการเดินของหุ่นยนต์จริง	48
5.4.	แสดงการเคลื่อนที่ของน้ำหนักหุ่นยนต์	51

บทที่ 1

บทนำ

1.1. ความสำคัญและความเป็นมา

การโปรแกรมโดยอัตโนมัติเพื่อควบคุมหุ่นยนต์ เป็นเป้าหมายใหญ่ของงานวิจัยทางด้านหุ่นยนต์มานานแล้ว ปัจจุบันการเขียนโปรแกรมควบคุมหุ่นยนต์โดยมนุษย์ เป็นเรื่องที่ทำได้ยากขึ้นมาก เนื่องจากกลไกการทำงานของหุ่นยนต์ได้ถูกพัฒนาให้มีความซับซ้อนสูงขึ้น หุ่นยนต์บางชนิดถูกพัฒนาให้มีความสามารถในการทำงานพื้นฐานแทนมนุษย์ได้ใกล้เคียงมาก เช่น หุ่นยนต์ P2 ซึ่งเป็นหุ่นยนต์มนุษย์ (humanoid robot) ของบริษัทฮอนด้า (Honda) (Hirai, 1998) เป็นต้น สามารถกล่าวได้ว่า ในปัจจุบันข้อจำกัดในการควบคุมหุ่นยนต์เหล่านี้ให้ทำงานตามที่ต้องการจึงอยู่ที่การโปรแกรมเพื่อควบคุมหุ่นยนต์

การคำนวณเชิงวิวัฒน์ (Evolutionary Computation) เป็นวิธีการหนึ่งในหลาย ๆ วิธีของปัญญาประดิษฐ์ (Artificial Intelligence) ซึ่งสามารถหาคำตอบของปัญหาหลาย ๆ ปัญหา รวมทั้งปัญหาทางด้านหุ่นยนต์ และให้คำตอบอยู่ในรูปของโปรแกรมได้ ในปัจจุบัน มีงานวิจัยที่เกี่ยวข้องกับการคำนวณเชิงวิวัฒน์อยู่มากมาย (Davidor, 1990; Koza, 1992; Polvichai, 1996) การคำนวณเชิงวิวัฒน์ไม่มีรูปแบบการแก้ปัญหาที่ตายตัว (weak method) สามารถประยุกต์ใช้ในการแก้ปัญหาต่าง ๆ ได้หลากหลาย เช่น ปัญหาการหาสมการถดถอย (symbolic regression), ปัญหาการจัดวางตารางงาน (job scheduling), ปัญหาการควบคุมหุ่นยนต์ (robot control) เป็นต้น

การคำนวณเชิงวิวัฒน์เป็นวิธีการหาคำตอบโดยอิงกับกลุ่มคำตอบ (population-based method) ซึ่งกลุ่มคำตอบเหล่านี้จะถูกวิวัฒนาการผ่านรุ่นต่อรุ่นเพื่อหาทางเข้าสู่แนวของคำตอบที่ดีที่สุด ในการหาคำตอบนี้ จะมีตัววัดค่าความดีของคำตอบแต่ละตัวเป็นตัวชักจูงให้กลุ่มพัฒนาไปในแนวทางที่ต้องการ เรียกตัวชักจูงนี้ว่า “เกณฑ์การประเมินค่าความเหมาะสม (fitness function)” เกณฑ์การประเมินค่าความเหมาะสมนี้จะถูกกำหนดหรือนิยามโดยขึ้นกับเงื่อนไขและสภาพแวดล้อมของปัญหาแต่ละปัญหา

งานวิจัยนี้เสนอวิธีการสังเคราะห์โปรแกรมควบคุมหุ่นยนต์เดินสองขาแบบสมดุลสถิต หุ่นยนต์เดินสองขาเป็นหุ่นยนต์แบบมีจำนวนข้อต่อมาก ซึ่งการควบคุมหุ่นยนต์ให้ทำงานตามที่ต้องการเป็นไปได้ยาก จากข้อจำกัดนี้จึงทำให้ปัญหานี้เป็นปัญหาที่น่าสนใจในการที่จะนำวิธีการสังเคราะห์

โปรแกรมควบคุมโดยอัตโนมัติมาใช้แก้ไข อีกทั้งหุ่นยนต์สองขาเป็นหุ่นยนต์ชนิดเคลื่อนที่ด้วยการเดิน (walking robot) และมีความสามารถมากกว่าหุ่นยนต์ชนิดมีล้อ (wheel-robot) เนื่องจากหุ่นยนต์ชนิดเคลื่อนที่ด้วยการเดินสามารถเคลื่อนที่ได้ในพื้นที่ต่าง ๆ โดยไม่จำกัดชนิดของพื้นผิว กล่าวคือ สามารถเคลื่อนที่ได้ทั้งในพื้นผิวเรียบ ขุรขระ หรือแม้แต่พื้นผิวที่ไม่ต่อเนื่องได้ นอกจากนี้ ถ้าเปรียบเทียบการทำงานของหุ่นยนต์สองขากับหุ่นยนต์แบบหลายขา (multilegged robot) ในพื้นที่ต่าง ๆ ที่สร้างขึ้นมาเพื่อมนุษย์ เช่น บนทางเท้าที่มีผู้คนขวักไขว่ ในอุโมงค์ ในรถยนต์ หรือในลิฟต์ เป็นต้น ในสถานที่ดังกล่าวนี้ หุ่นยนต์สองขายังคงมีความเหมาะสมในการทำงานมากกว่าหุ่นยนต์แบบหลายขา การเดินของหุ่นยนต์สองขาจัดได้ว่าควบคุมได้ยากกว่าการเดินของหุ่นยนต์แบบหลายขา เนื่องจากในระหว่างการเคลื่อนที่ หุ่นยนต์ทั้งสองชนิดต้องมีการรักษาสมดุล แต่หุ่นยนต์สองขารักษาสมดุลโดยใช้จำนวนข้อต่อน้อยกว่า ด้วยเหตุผลด้วยกล่าวข้างต้น จึงทำให้งานวิจัยนี้สนใจในการแก้ปัญหาการเดินของหุ่นยนต์สองขามากกว่าหุ่นยนต์ชนิดอื่น

1.2. วัตถุประสงค์ของการวิจัย

งานวิจัยนี้พยายามหาวิธีสังเคราะห์โปรแกรมเพื่อควบคุมหุ่นยนต์สองขาให้สามารถเดินได้ โดยเน้นการสังเคราะห์โปรแกรมให้เป็นแบบอัตโนมัติ

1.3. ขอบเขตของงานวิจัย

วิธีที่ใช้ในการสังเคราะห์โปรแกรมควบคุมหุ่นยนต์คือการคำนวณเชิงวิวัฒน์และโปรแกรมที่สังเคราะห์ขึ้นจะควบคุมให้หุ่นยนต์สามารถเดินหน้าได้เท่านั้น ไม่ถอยหลัง ไม่เดินเลี้ยวซ้ายหรือเลี้ยวขวา หุ่นยนต์จะเดินได้เฉพาะบนพื้นผิวราบ เรียบและแข็งเท่านั้น ไม่มุ่งเน้นให้หุ่นยนต์เดินขึ้นลงบันไดหรือวิ่งได้ สมดุลของการเดินของหุ่นยนต์เป็นแบบสมดุลสถิต

1.4. ขั้นตอนการวิจัย

งานวิจัยนี้สามารถแบ่งขั้นตอนการดำเนินงานโดยสังเขปได้ดังนี้

1) ศึกษาทฤษฎีการคำนวณเชิงวิวัฒน์ และขั้นตอนวิธีพันธุกรรม
2) ศึกษางานวิจัยที่เกี่ยวข้อง
3) ศึกษาและออกแบบโครงสร้างของหุ่นยนต์สองขา
4) จัดสร้างหุ่นสองขา
5) เขียนโปรแกรมควบคุมหุ่นยนต์

6) ออกแบบการเข้ารหัสของโปรแกรม และออกแบบการทดลอง
7) ทำการทดลองและเก็บผลการทดลอง
8) สรุปผลการทดลอง และทำรายงานวิทยานิพนธ์

1.5. ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย

เนื่องจากในส่วนการเรียนรู้ของหุ่นยนต์ของงานวิจัยนี้ได้กระทำภายใต้สภาพแวดล้อมจำลองและเรียนรู้กับหุ่นยนต์จริงประกอบกัน ทำให้สามารถเข้าใจความสัมพันธ์การเรียนรู้ของการคำนวณเชิงวิวัฒน์ที่ประยุกต์ใช้กับปัญหาการเดินของหุ่นยนต์สองขาระหว่างสภาพแวดล้อมจำลองและสภาพแวดล้อมจริงได้ เช่น ช่วงใดของการเรียนรู้ที่ควรจะกระทำในสภาพแวดล้อมจำลองและช่วงใดควรกระทำกับหุ่นยนต์จริง หรือ สัดส่วนเวลาที่ใช้ในการเรียนรู้ระหว่างสภาพแวดล้อมจริงกับสภาพแวดล้อมจำลองควรเป็นอย่างไร เป็นต้น ซึ่งถ้าสามารถปรับความสัมพันธ์เหล่านี้ให้เหมาะสมกับการเรียนรู้ได้ก็จะทำให้การเรียนรู้เกิดประสิทธิภาพมากยิ่งขึ้น กล่าวคือ สามารถเรียนรู้ได้รวดเร็วและได้คำตอบที่นำไปใช้จริงได้อีกด้วย

1.6. ผลงานที่ตีพิมพ์จากงานวิจัย

วิทยานิพนธ์นี้ได้รับการตีพิมพ์ในหัวข้อเรื่อง “Automatic Synthesis of Robot Programs for a Biped Static Walker by Evolutionary Computation” โดย Chanchai Chaisukkosol และ Prabhas Chongstitvatana ในงานประชุมวิชาการ “The Second Asian Symposium on Industrial Automation and Robotics (ASIAR)” ณ กรุงเทพฯ ประเทศไทย ในวันที่ 17-18 พฤษภาคม พ.ศ. 2544

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

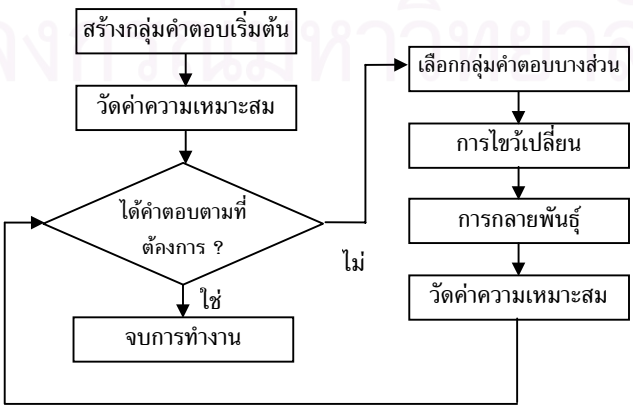
บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1. แนวทางการทำงานของการคำนวณเชิงวิวัฒน์

การคำนวณเชิงวิวัฒน์จะมีขั้นตอนการทำงาน (รูปที่ 2.1) โดยเริ่มจากการ**สร้างกลุ่มคำตอบ**ขึ้นมาก่อน (population initialization) โดยการสร้างกลุ่มคำตอบนี้จะสร้างด้วยวิธีการสุ่มเพื่อให้มีความหลากหลายของกลุ่ม จากนั้นจะทำ**การประเมินค่าความเหมาะสม** (fitness evaluation) ของคำตอบแต่ละคำตอบ เพื่อดูว่าแนวโน้มของกลุ่มว่ามีความสามารถตามที่ต้องการแล้วหรือไม่ คำตอบที่ได้สามารถแก้ปัญหาได้ลุล่วงหรือไม่ ถ้าตรงตามที่ต้องการแล้วก็จะสิ้นสุดการวิวัฒนาการ แต่ถ้ายังไม่ได้ตามที่ต้องการจะทำ**การคัดเลือก** (selection) คำตอบบางตัว เพื่อไปใช้ในการผลิตกลุ่มคำตอบรุ่นต่อไป กลุ่มคำตอบที่ถูกเลือกมานี้ จะถูกนำไปดำเนินการพันธุกรรม ซึ่งมี 2 วิธี ได้แก่ **การไขว้เปลี่ยน** (crossover) และ**การกลายพันธุ์** (mutation) เมื่อผ่านการดำเนินการทางพันธุกรรมแล้ว จะได้กลุ่มคำตอบสำหรับรุ่นใหม่ ซึ่งอาจมีหน้าตาเหมือนหรือแตกต่างจากรุ่นพ่อรุ่นแม่ไปบ้างเล็กน้อยก็ได้ กลุ่มคำตอบสำหรับรุ่นใหม่นี้จะถูกทำการประเมินค่าความเหมาะสมของคำตอบแต่ละตัวอีก เพื่อดูว่ารุ่นนี้มีความสามารถเหมาะสมกับปัญหาแล้วหรือไม่ ถ้าได้แล้วก็จบการวิวัฒนาการ แต่ถ้ายังไม่ได้ก็จะทำการคัดเลือก ให้กำเนิดกลุ่มคำตอบในรุ่นต่อไป แล้วก็ทำการประเมินค่าความเหมาะสมอีก เป็นเช่นนี้ไป รุ่นแล้วรุ่นเล่าจนกระทั่งได้คำตอบหรือครบกำหนดเวลาในการวิวัฒนาการ

ในงานวิจัยนี้ใช้วิธีการคำนวณเชิงวิวัฒน์ที่เรียกว่า “ขั้นตอนวิธีพันธุกรรม” (Genetic Algorithms) ซึ่งเป็นวิธีการหนึ่งในหลาย ๆ วิธีของการคำนวณเชิงวิวัฒน์ แนวคิดของวิธีนี้ถูกเสนอ



โดย Holland (Holland, 1975) โดยที่การเข้ารหัสของกลุ่มคำตอบของวิธีนี้จะอยู่ในรูปแบบของสายอักขระ (string) ซึ่งเป็นวิธีการที่เลียนแบบมาจากสายโครโมโซมของสิ่งมีชีวิต และสามารถอ่านรายละเอียดได้จากเอกสารของ Goldberg (Goldberg, 1989)

2.2. การคัดเลือก (selection)

ในการคัดเลือกคำตอบเพื่อนำไปผลิตกลุ่มคำตอบรุ่นถัดไป สามารถทำได้หลายวิธี โดยปกติจะใช้วิธีคัดเลือกตามสัดส่วนของค่าความเหมาะสม (roulette wheel selection) ด้วยวิธีการนี้ คำตอบที่มีค่าความเหมาะสมสูงจะมีโอกาสถูกเลือกได้สูงกว่าคำตอบที่มีค่าความเหมาะสมต่ำ โดยสมการความน่าจะเป็นของการถูกเลือกของคำตอบแต่ละตัวสามารถแสดงได้ดังนี้

$$P_n = \frac{f_n}{\sum_{i=0}^{n} f_i}$$

P_n : ความน่าจะเป็นที่คำตอบแต่ละตัวจะถูกเลือก

f_n : ค่าความเหมาะสมของคำตอบแต่ละตัว

อย่างไรก็ตาม วิธีการเลือกแบบนี้จะทำให้ความหลากหลายของคำตอบลดลงเมื่อมีวิวัฒนาการผ่านไปหลาย ๆ รุ่น กลุ่มคำตอบส่วนใหญ่จะมีแนวโน้มเหมือนหรือคล้ายกับคำตอบที่มีค่าความเหมาะสมที่สูง ๆ ซึ่งแม้ว่าจำนวนคำตอบที่มีค่าความเหมาะสมสูงเพิ่มขึ้น แต่การขาดความหลากหลายของกลุ่มคำตอบนี้มีผลเสียคือ ในกรณีที่คำตอบที่มีค่าความเหมาะสมสูงไม่ใช่คำตอบที่ถูกที่สุดก็จะทำให้กลุ่มคำตอบทั้งหมดวิวัฒนาการไปผิดทาง การแก้ไขปัญหานี้ คือ ในการวิวัฒนาการไปจะต้องทำให้เกิดความหลากหลายขึ้นในกลุ่มคำตอบ และกลุ่มคำตอบยังคงต้องวิวัฒนาการไปในแนวทางที่ทำให้ค่าเฉลี่ยของค่าความเหมาะสมเพิ่มขึ้นอีกด้วย

ในงานวิจัยนี้ใช้วิธีการคัดเลือกที่เรียกว่า combined rank selection (Winston, 1992) ซึ่งวิธีการนี้จะคัดเลือกคำตอบไปใช้ผลิตกลุ่มคำตอบรุ่นถัดไปโดยพิจารณาข้อมูล 2 ชนิด คือค่าความเหมาะสมและค่าความแตกต่างกันของคำตอบแต่ละตัว การพิจารณาข้อมูลทั้งสองตัวนี้ ไม่ได้พิจารณาในลักษณะเป็นค่าสัดส่วนเปรียบเทียบกับคำตอบอื่น แต่จะพิจารณาโดยการจัดอันดับ (ranking) ของค่าความเหมาะสมและค่าความแตกต่าง แล้วนำเลขอันดับของค่าความเหมาะสมและค่าความแตกต่างมารวมกัน จากนั้นจึงค่อยกำหนดสัดส่วนความน่าจะเป็นของการถูกเลือกตามเลขอันดับของผลรวมนี้ต่อไป กล่าวคือ คำตอบที่มีอันดับของผลรวมอยู่ในอันดับต้น ๆ จะมีความน่าจะเป็นของการถูกเลือกมากกว่าอันดับท้าย ๆ ขั้นตอนวิธีการของการคัดเลือกแบบ combined rank สามารถแสดงได้ดังตารางที่ 2.1

ตารางที่ 2.1: แสดงขั้นตอนวิธีของการคัดเลือกแบบ combined rank

data structure

parent[0..N]	refer to array of parent population selected to produce new generation.
record pop[0..k-1]	refer to record of population of k individuals, combined from 6 variables:
{individual	refer to variable of individuals' aspect.
fitness	refer to variable of fitness value.
diversity	refer to variable of diversity.
f_rank	refer to variable of fitness ranking number, ordering by fitness value.
d_rank	refer to variable of diversity ranking number, ordering by diversity.
c_rank };	refer to variable of combined ranking number, ordering by summation of f_rank and d_rank.

algorithm

```
Sort pop[0..k-1] by fitness.
Copy pop[0] to parent[0].
for(i=1; i<k; i++)
{   Sort pop[j] by fitness; j is range from i to k-1.*
    Update ranking number to pop[i..k].f_rank

    for(j=i; j<k; j++)
    {   Calculate cumulative diversity of pop[i-1] compared with pop[j] and assigned to pop
        [j].diversity .
    }

    Sort pop[j] by diversity; j is range from i to k-1.
    Assign ranking number to pop[i..k].d_rank

    for(j=i; j<k; j++)
    pop[j].c_rank = pop[j].f_rank + pop[j].d_rank;
```

* ไม่จำเป็นต้อง `sort pop[j] by fitness` ในทุก ๆ รอบ สามารถเอาออกจากลูป ได้ จะมีผลทำให้การคำนวณเร็วขึ้น อย่างไรก็ตาม ในงานนี้ ผู้วิจัยมิได้เอา statement นี้ออกนอกลูป จึงเสนอแนะไว้ในที่นี้

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

```
        find n that pop[n].c_rank is a minimum value.
        Swap(pop[i],pop[n]);
        Copy pop[i] to parent[i]
    }
```

ในการวัดค่าความแตกต่างกันของคำตอบละตัวสามารถวัดได้หลายวิธี หนึ่งในวิธีที่นิยมใช้กันคือ การเปรียบเทียบความแตกต่างกันของสายอักขระของคำตอบ 2 ตัว โดยเปรียบเทียบเป็นอักขระต่ออักขระ หรือบิตต่อบิต (Hamming distance) ตัวอย่างเช่น

คำตอบที่ 1: **1 1 1 0 1 0 1**
คำตอบที่ 2: **0 0 1 0 1 1 1**

ถ้าเรากำหนดให้คำตอบที่แตกต่างกัน 1 อักขระ มีค่าความแตกต่างกันเป็น 1 ในตัวอย่างนี้คำตอบทั้งสองจะมีค่าความแตกต่างกันเท่ากับ 3 เป็นต้น

2.3. การให้กำเนิดกลุ่มคำตอบรุ่นต่อไป

ในการให้กำเนิดกลุ่มคำตอบรุ่นใหม่นั้นจะมีวิธีการดำเนินการทางพันธุกรรมอยู่ 2 วิธี คือ การไขว้เปลี่ยน (crossover) และการกลายพันธุ์ (mutation)

ก. การไขว้เปลี่ยน

การไขว้เปลี่ยน เป็นการสลับค่าโครโมโซมบางช่วงของคำตอบสองคำตอบที่ถูกสุ่มเลือกขึ้นมา (รูปที่ 2.2ก) โดยปกติแล้ว คำตอบแต่ละคำตอบจะมีส่วนที่ดีและที่ไม่ดีอยู่ด้วยกัน ในการไขว้เปลี่ยนกันนี้ ได้คาดหวังไว้ว่าอาจเป็นทำให้ส่วนที่ดี ๆ ของคำตอบแต่ละคำตอบมาอยู่ด้วยกัน และส่วนที่ไม่ดีก็จะอยู่ด้วยกัน ซึ่งคำตอบที่มีแต่ส่วนที่ดีมาอยู่ด้วยกันก็จะมีโอกาสถูกเลือกไปผลิตรุ่นต่อไป

ข. การกลายพันธุ์

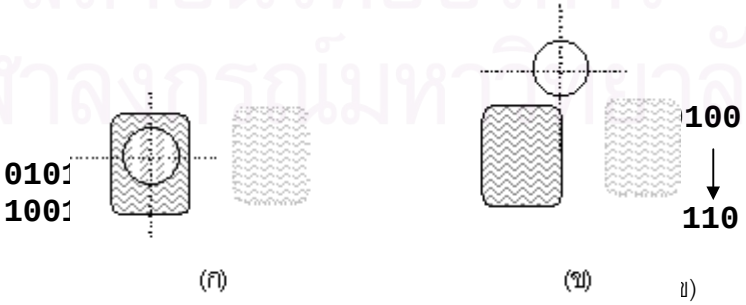
การกลายพันธุ์ (รูปที่ 2.2ข) เป็นการสุ่มเปลี่ยนค่าโครโมโซมบางตัวของคำตอบนั้น ๆ ตัวดำเนินการกลายพันธุ์นี้มีขึ้นเพื่อประโยชน์ในการเกิดมีคำตอบในลักษณะใหม่ ๆ ขึ้นมาบ้างโดยทั่วไป มักกำหนดให้ความน่าจะเป็นของการกลายพันธุ์มีค่าน้อยๆ เช่น 0.001 เป็นต้น เพราะค่าความน่าจะเป็นของการกลายพันธุ์ที่สูง จะส่งผลให้กลุ่มคำตอบสะเปะสะปะเกินไป คำตอบไม่อยู่ในแนวเดียวกัน และทำให้ไม่มีแนวทางในการวิวัฒนาการของคำตอบ

2.4. สมดุลของการเดินแบบสองขา

ในการเดินแบบสองขานี้จะมีการถ่ายน้ำหนักไปมาระหว่างเท้าซ้ายเท้าขวา มีบางช่วงที่น้ำหนักจะตกลงบนเท้าข้างใดข้างหนึ่งหรือทั้งสองข้าง การเดินให้ได้โดยไม่ล้มนั้นจำเป็นต้องรักษาสมดุลของการเดินให้ได้ตลอดช่วงของการเดิน ซึ่งสมดุลของการเดินแบบสองขาสามารถแบ่งได้เป็น 2 แบบ ได้แก่

ก. **การเดินแบบสมดุลสถิต (static balance walking)** (รูปที่ 2.3ก) การเดินแบบนี้น้ำหนักตัวจะไม่มีการเคลื่อนไหวออกนอกบริเวณฐานรับน้ำหนักตลอดช่วงการเดิน ไม่ว่าจะเป็นช่วงที่รับน้ำหนักด้วยเท้าข้างเดียวหรือสองข้างก็ตาม ในขณะที่การควบคุมเพื่อการเดินแบบสมดุลสถิตนี้จะทำได้ง่ายกว่า แต่หุ่นยนต์จะใช้เวลาในการก้าวเดินมากกว่า และหุ่นยนต์ที่มีฝ่าเท้าใหญ่จะทำให้ง่ายต่อการก้าวเดินมากขึ้น

ข. **การเดินแบบสมดุลพลวัต (dynamic balance walking)** (รูปที่ 2.3ข) คือการเดินที่น้ำหนักตัวสามารถเคลื่อนออกนอกฐานรับน้ำหนักได้ กล่าวในเชิงปฏิบัติการจะทำให้หุ่นยนต์เดินได้แบบสมดุลพลวัตนี้จะยากกว่าการเดินแบบสมดุลย์สถิต แต่เป็นการเดินแบบเดียวกับมนุษย์ การเดินจะราบรื่นกว่า และเดินได้เร็วกว่าแบบแรก พลังงานที่ใช้ในการเดินก็ใช้น้อยกว่าแบบแรกเนื่องจากมีการใช้ประโยชน์จากแรงโน้มถ่วงของโลกช่วยในการเดินด้วย



รูปที่ 2.3: แสดงลักษณะจุดศูนย์ถ่วงจากมุมมองเบื้องบนในขณะเดิน
(ก) ในขณะเดินแบบสมดุลสถิต
(ข) ในขณะเดินแบบสมดุลพลวัต

บทที่ 3

งานวิจัยที่เกี่ยวข้อง

3.1. หุ่นยนต์สองขา

งานวิจัยทางด้านหุ่นยนต์สองขาจากอดีตจนถึงปัจจุบันส่วนใหญ่จะเป็นการพัฒนาความสามารถของการเดินของหุ่นยนต์สองขา เช่น เริ่มต้นจากแรกสุดจะเป็นการพัฒนาให้หุ่นยนต์สองขาเดินหน้าได้ จากนั้นก็เพิ่มความสามารถให้หุ่นยนต์เดินบนพื้นเอียง พื้นขรุขระ เดินเลี้ยวซ้ายขวา เดินขึ้นลงบันได ฯลฯ เป็นต้น

นอกจากนี้ยังมีการพัฒนาปรับปรุงสมดุลของการเดินแบบสองขาอีกด้วย สมดุลของการเดินแบบสองขาสามารถแบ่งได้เป็น 2 แบบ ได้แก่ การเดินแบบสมดุลสถิต (static balance walking) และการเดินแบบสมดุลพลวัต (dynamic balance walking) งานในยุคแรก ๆ จะพัฒนาหุ่นยนต์ให้เดินได้โดยเป็นการเดินแบบสมดุลสถิต ต่อมาจึงพัฒนาหุ่นยนต์เดินสองแบบสมดุลกึ่งพลวัต (quasi-dynamic balance walking) และเป็นสมดุลพลวัตในที่สุด ซึ่งเป็นการเดินแบบที่เหมือนมนุษย์มากที่สุด งานวิจัยเกี่ยวกับหุ่นยนต์สองขาในยุคแรก ๆ เป็นการพัฒนาตัวควบคุมการเดินของหุ่นยนต์ ซึ่งใช้ความรู้ทางด้านกลศาสตร์ค่อนข้างมาก มีการพิสูจน์สมการต่าง ๆ ค่อนข้างซับซ้อน จะขอยกตัวอย่างบางงานที่เกี่ยวกับการพัฒนาตัวควบคุมการเดินหุ่นยนต์สองขา ดังนี้

Zheng และคณะ (1988) พัฒนาหุ่นยนต์สองขาที่สามารถเดินจากพื้นราบได้ ให้สามารถเดินต่อเนื่องไปบนพื้นเอียงได้ด้วย พื้นเอียงที่ใช้มีลักษณะเป็นพื้นเอียงขึ้น หุ่นยนต์ที่ใช้ในงานนี้มีข้อต่อสะโพก (hip), ข้อเท้า (ankle) และลำตัว (torso) มีหน่วยวัดแรงกด (force sensor) ติดตั้งอยู่ที่ปลายเท้าและส้นเท้าแต่ละข้างเพื่อใช้วัดการเคลื่อนในแนวหน้าหลังของน้ำหนักโดยรวม (center of gravity) ของหุ่นยนต์ การเดินของงานวิจัยนี้จะพิจารณาเฉพาะการเดินในแนวหน้าหลัง (sagittal plane) โดยมีหลักการคือ การเดินบนพื้นเอียงโดยที่หุ่นยนต์ยังเดินในท่าทางเหมือนกับตอนที่เดินบนพื้นราบจะทำให้น้ำหนักโดยรวมของหุ่นยนต์เลื่อนถอยหลังไป ดังนั้นการที่หุ่นยนต์ขยับลำตัวไปด้านหน้าจะทำให้น้ำหนักโดยรวมของหุ่นยนต์กลับมาอยู่ตรงกลางของพื้นที่รับน้ำหนักเหมือนเดิม ซึ่งจะทำให้หุ่นยนต์มีความสมดุลได้ ดังนั้นข้อมูลที่ได้จากหน่วยวัดแรงกดที่เท้าจะถูกนำมาคำนวณตลอดการเดินเพื่อใช้ในการปรับเปลี่ยนมุมการขยับของลำตัว การเดินบนพื้นราบเป็นแบบสมดุลสถิตและการเดินบนพื้นเอียงก็ยังคงเป็นแบบสมดุลสถิตเช่นกัน

Inaba และคณะ (1995) สร้างหุ่นยนต์เลียนแบบลิง (ape-like biped) ประกอบด้วยสองมือและสองขา มีการเดินแบบสมดุลสถิต งานวิจัยนี้มีความคิดว่านอกจากการทำให้หุ่นยนต์สองขาเดินได้โดยไม่ล้มแล้ว ควรจะทำหุ่นยนต์ที่สามารถลุกขึ้นเองได้หลังจากที่ล้มแล้วด้วย ดังนั้นในงานนี้ หุ่นยนต์ถูกพัฒนาให้สามารถเดิน เมื่อล้มแล้วก็สามารถพลิกตัวและลุกขึ้นมาเดินให้ได้

Kun และ Miller (1996) ได้นำ โครงข่ายประสาทเทียม (Artificial Neural Network) มาประยุกต์ใช้ในการปรับเปลี่ยนท่าทางการเดินโดยอัตโนมัติของหุ่นยนต์สองขา การที่หุ่นยนต์สามารถปรับเปลี่ยนท่าทางได้โดยอัตโนมัตินี้มีประโยชน์ทำให้หุ่นยนต์เดินได้บนพื้นผิวหลากหลายลักษณะมากขึ้น ในงานนี้พิจารณาทั้งสมดุลในแนวหน้าหลัง (sagittal plane) และแนวซ้ายขวา (frontal plane) และการเดินของหุ่นยนต์เป็นแบบสมดุลพลวัต หลักการทำงานประกอบด้วยหน่วยสร้างท่าทางการเดินหลักอยู่หนึ่งตัว และมีหน่วยปรับท่าทางการเดินทั้งแนวหน้าหลังและซ้ายขวาอีกจำนวนหนึ่ง ซึ่งคำนวณอัตราการปรับเปลี่ยนมาจากการประมวลข้อมูลต่าง ๆ เช่น แรงกดที่เท้า ความยาวการก้าวเท้า ความสูงของการยกเท้า เป็นต้น นอกจากนี้ ในปีถัดมาทั้งสองยังได้ใช้หลักการที่ใช้ในงานนี้ไปประยุกต์ใช้กับการเดินแบบสมดุลสถิตของหุ่นยนต์ (Kun and Miller, 1997)

Hirai และคณะ (1998) พัฒนาหุ่นยนต์มนุษย์ (humanoid robot) ซึ่งหุ่นยนต์มีส่วนประกอบต่าง ๆ คล้ายมนุษย์มาก สามารถเดินได้อย่างราบรื่นคล้ายมนุษย์มากที่สุด เช่น สามารถเดินได้ในพื้นผิวชนิดต่าง ๆ เดินได้บนพื้นเอียงขึ้นเอียงลง เดินขึ้นลงบันไดได้ เดินเข็นรถ เป็นต้น การเดินในทุกสถานการณ์เป็นการเดินแบบสมดุลพลวัตทั้งสิ้น หุ่นยนต์สามารถเดินได้ด้วยความเร็วสูงสุด 4.7 กิโลเมตรต่อชั่วโมง หุ่นยนต์ที่ใช้ในงานนี้ประกอบด้วยแขนข้างละ 9 แกน ขาข้างละ 6 แกน ที่บริเวณศีรษะมีกล้องติดตั้งอยู่ 4 ตัว ใช้สำหรับการประมวลเกี่ยวกับภาพ (vision processing), การกำหนดเป้าหมาย และการถ่ายทอดสัญญาณภาพ นอกจากนี้ยังมีอุปกรณ์ที่ใช้ในการรักษาสมดุลอื่น ๆ อีกได้แก่ หน่วยวัดมุมในแต่ละข้อต่อ (angular sensor), หน่วยวัดความเอียง (inclination sensor)ของลำตัว, หน่วยวัดแรงกด (force sensor) ติดตั้งที่เท้าทั้งสองข้าง

3.2. ปัญญาประดิษฐ์กับหุ่นยนต์สองขา

จากงานวิจัย 4 - 5 ชิ้นที่ยกตัวอย่างมานั้น เป็นงานเกี่ยวกับควบคุมหุ่นยนต์โดยการพัฒนาตัวควบคุม (controller) ขึ้นมาเท่านั้น หรือแม้ในหุ่นยนต์ที่เดินเหมือนคนมากที่สุดก็ยังใช้การควบคุมหุ่นยนต์ด้วยการโปรแกรมโดยมนุษย์ (manual programming) ซึ่งการควบคุมหุ่นยนต์โดยวิธีทั้งสองแบบนี้ทำได้ยาก และยากขึ้นเรื่อยๆ เมื่อเทียบกับความซับซ้อนของกลไกและความสามารถที่ต้องการให้มีมากขึ้นของหุ่นยนต์ในปัจจุบัน โดยเฉพาะหุ่นยนต์ที่สร้างขึ้นในงานวิจัยของ Hirai ซึ่งมีความซับซ้อนมากเนื่องจากต้องการให้หุ่นยนต์มีการเคลื่อนไหวคล้ายมนุษย์มากที่สุด ดังนั้นการโปรแกรมควบคุมหุ่นยนต์แบบอัตโนมัติจึงมีความจำเป็นในการแก้ไขจัดการกับความซับซ้อนของโปรแกรมควบคุมให้

ง่ายขึ้น ปัญญาประดิษฐ์ถูกนำมาประยุกต์ใช้ในการสังเคราะห์โปรแกรมควบคุมหุ่นยนต์โดยอัตโนมัติ หนึ่งในหลายวิธีของปัญญาประดิษฐ์คือการคำนวณเชิงวิวัฒน์ (Evolutionary Computation) ซึ่งเป็นวิธีการที่สามารถให้คำตอบออกมาในรูปของโปรแกรมได้ มีงานวิจัยหลายชิ้นที่นำวิธีนี้มาใช้ในการควบคุมหุ่นยนต์ ได้แก่

Polvichai (1996) แก้ปัญหาเกี่ยวกับการเคลื่อนที่เข้าหาเป้าหมายโดยหลบหลีกสิ่งกีดขวางของแขนหุ่นยนต์แบบ 3 ข้อต่อซึ่งมีการเคลื่อนที่แบบแบบสองมิติ โปรแกรมควบคุมหุ่นยนต์ถูกสังเคราะห์ขึ้นโดยใช้กำหนดการเชิงพันธุกรรม (Genetic Programming) ซึ่งเป็นวิธีการแบบหนึ่งของการคำนวณเชิงวิวัฒน์ การทดลองของงานวิจัยนี้จะทำในสภาพแวดล้อมจำลอง (simulation) และนำคำตอบที่ได้จากสภาพแวดล้อมจำลองมาทดสอบกับหุ่นยนต์จริงเพื่อยืนยันคุณภาพของคำตอบ

Cheng (1995) ใช้ขั้นตอนวิธีพันธุกรรม (Genetic Algorithms) ซึ่งเป็นอีกวิธีการหนึ่งที่ได้รับความนิยมของการคำนวณเชิงวิวัฒน์ในการหาสังเคราะห์โปรแกรมควบคุมหุ่นยนต์ ในงานวิจัยนี้หุ่นยนต์ถูกนำมาแก้ไขปัญหาภายใต้เงื่อนไขและสภาพแวดล้อมหลาย ๆ แบบ ได้แก่ การเดินบนพื้นราบ การเดินบนพื้นเอียงทั้งเอียงขึ้นและเอียงลง การเดินด้วยความเร็วที่แตกต่างกัน และการเดินโดยมีการกำหนดความยาวของระยะก้าวเท้า การเดินของหุ่นยนต์เป็นแบบสมดุลพลวัต และพิจารณาเฉพาะการเดินไปข้างหน้าเท่านั้น หุ่นยนต์ที่ใช้มี 5 ข้อต่อ คือ สะโพก และหัวเข่า ขั้นตอนวิธีพันธุกรรมถูกนำมาใช้หาในการหาค่าคงที่ที่เหมาะสมของสมการควบคุมแรงบิดในแต่ละข้อต่อของหุ่นยนต์ รวมทั้งปรับตำแหน่งมุมของแต่ละข้อต่อในแต่ละช่วงเวลาของการเดิน โปรแกรมควบคุมหุ่นยนต์ถูกเข้ารหัสให้อยู่ในรูปของเมทริกซ์ งานวิจัยนี้ทำการทดลองในสภาพแวดล้อมจำลองเช่นกัน

Rodrigues (1996) ประยุกต์ขั้นตอนวิธีพันธุกรรมในการหาแรงบิดน้อยที่สุดที่จำเป็นสำหรับใช้ในการเดินของหุ่นยนต์ มีเกณฑ์การวัดค่าความเหมาะสมโดยการเปรียบเทียบความเหมือนกันของท่าทางหุ่นยนต์ที่เกิดขึ้นจริง (actual posture) กับท่าทางในอุดมคติ (ideal posture) ขั้นตอนวิธีพันธุกรรมจะสังเคราะห์คำตอบซึ่งเป็นสายลำดับคำสั่งสำหรับควบคุมหุ่นยนต์โดยตรง ในการทดลองหุ่นยนต์จะถูกทำการทดลองในสภาพแวดล้อมจำลองเท่านั้น

Arakawa และ Fukuda (1996, 1997) พยายามหาท่าทางการเดินที่เป็นธรรมชาติของหุ่นยนต์และพยายามลดพลังงานที่ใช้ในการเดินให้เหลือน้อยที่สุด งานวิจัยนี้ทำการทดลองในสภาพแวดล้อมจำลองและทดสอบยืนยันผลการทดลองที่ได้กับหุ่นยนต์จริงอีกทอดหนึ่ง

3.3. สรุป

จะเห็นได้ว่างานวิจัยที่ประยุกต์ขั้นตอนวิธีพันธุกรรมมาใช้ในการหาคำตอบของหุ่นยนต์นั้น ส่วนใหญ่นิยมทำการทดลองในสภาพแวดล้อมจำลอง ทั้งนี้เนื่องจากหลักการทำงานของการคำนวณเชิงวิวัฒน์จำเป็นต้องมีการประเมินค่าความเหมาะสมของคำตอบแต่ละคำตอบเป็นจำนวนมาก ซึ่งถ้าทำการทดลองในสภาพแวดล้อมจริงจะต้องอาศัยเวลามาก อีกทั้งหุ่นยนต์จริงอาจชำรุดเสียหายได้เนื่องจากการประเมินคำตอบจำนวนหลาย ๆ ครั้ง อย่างไรก็ตาม เป็นที่ทราบกันดีว่าการนำคำตอบที่ได้จากการทดลองในสภาพแวดล้อมจำลองมาใช้กับหุ่นยนต์จริงมีอัตราการประสบความสำเร็จน้อยมาก ทั้งนี้เนื่องจากการทำงานกับหุ่นยนต์จริงมีเงื่อนไขและสภาวะการณ์ต่าง ๆ ที่เกิดขึ้นมากมายและไม่อาจทำนายหรือจำลองได้ทั้งหมด (Brooks, 1991a)

ดังนั้นในการที่จะทำการทดลองโดยพยายามใช้เวลาให้น้อย และหุ่นยนต์จริงไม่ชำรุดเสียหายไปก่อน อีกทั้งคำตอบที่ได้จากการทดลองยังต้องสามารถใช้กับหุ่นยนต์จริงได้อีกนั้น การทดลองของงานวิจัยนี้จึงอาศัยการทดลองทั้งในสภาพแวดล้อมจำลองและทดลองกับหุ่นยนต์จริงประกอบกันเพื่อแก้ไขปัญหาข้างต้น ซึ่งจะกล่าวถึงในบทต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

การทดลอง

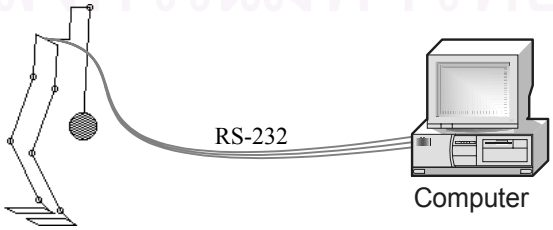
4.1. ลักษณะของปัญหา

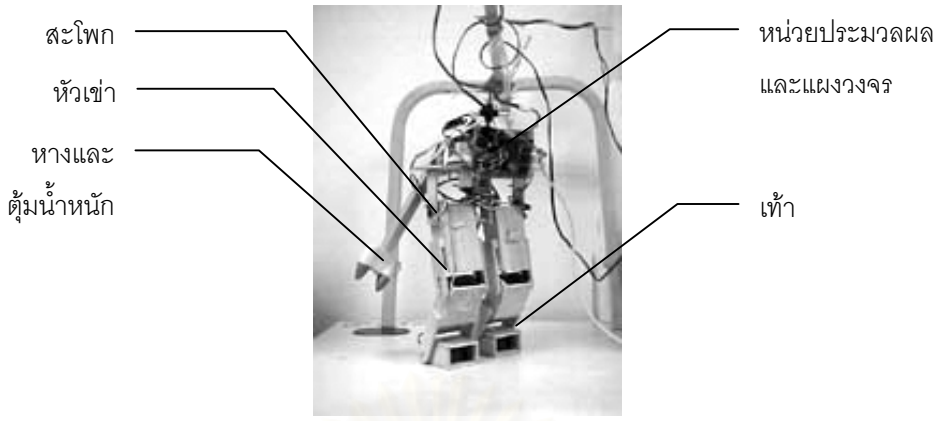
งานวิจัยครั้งนี้ศึกษาวิธีการสังเคราะห์โปรแกรมควบคุมหุ่นยนต์แบบเดินสองขาให้สามารถเดินไปข้างหน้าเพียงอย่างเดียว หุ่นยนต์ไม่เลี้ยวซ้ายขวา เดินเป็นทางโค้งหรือเดินถอยหลัง รวมทั้งไม่มุ่งเน้นให้หุ่นยนต์ขึ้นลงบันไดหรือวิ่งได้อีกด้วย ลักษณะพื้นผิวที่ใช้ในการทดลองจะเป็นพื้นผิวที่ราบเรียบและแข็ง ไม่ใช้พื้นผิวขรุขระหรือเอียงขึ้นลง อุปกรณ์ที่สำคัญสำหรับการทดลอง (รูปที่ 4.1) ประกอบด้วยตัวหุ่นยนต์สองขา หน่วยประมวลผลขนาดเล็กซึ่งติดอยู่บนตัวหุ่นยนต์ และเครื่องคอมพิวเตอร์ ซึ่งติดต่อกับหน่วยประมวลผลขนาดเล็กผ่านทางพอร์ตอนุกรมโดยใช้สาย RS-232 หน่วยประมวลผลขนาดเล็กจะมีหน้าที่ในการแปลงคำสั่งที่ได้รับจากเครื่องคอมพิวเตอร์ที่ส่งมาเพื่อใช้ในการควบคุมองศาการหมุนของแต่ละข้อต่อ ส่วนเครื่องคอมพิวเตอร์มีหน้าที่ในการอ่านโปรแกรมคำตอบที่สังเคราะห์ขึ้นเพื่อใช้สำหรับควบคุมหุ่นยนต์

4.2. หุ่นยนต์

4.2.1 โครงสร้างของหุ่นยนต์สองขา

หุ่นยนต์สองขาที่ใช้ในงานวิจัยนี้จำนวนข้อต่อทั้งหมด 7 ข้อต่อ ได้แก่ สะโพก (hip) 2 ข้อต่อ หัวเข่า (knee) 2 ข้อต่อ ข้อเท้า (ankle) 2 ข้อต่อ และหาง (tail) 1 ข้อต่อ ข้อต่อสะโพก หัวเข่า

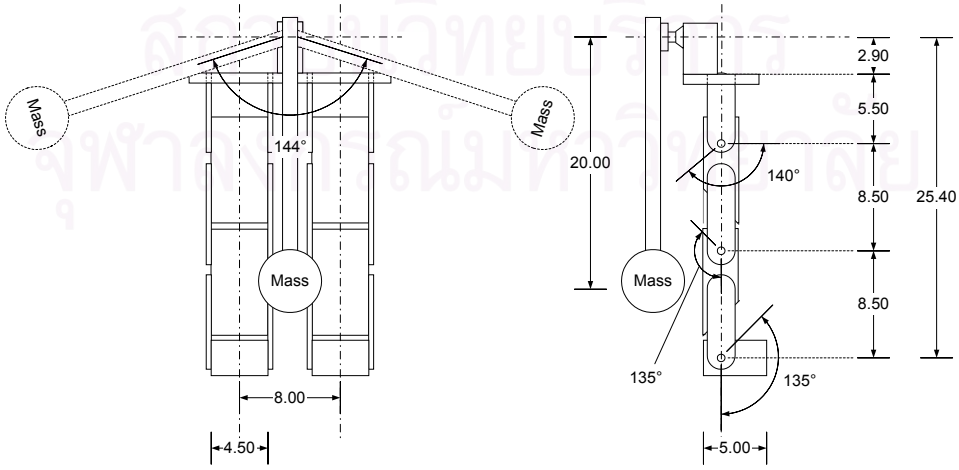




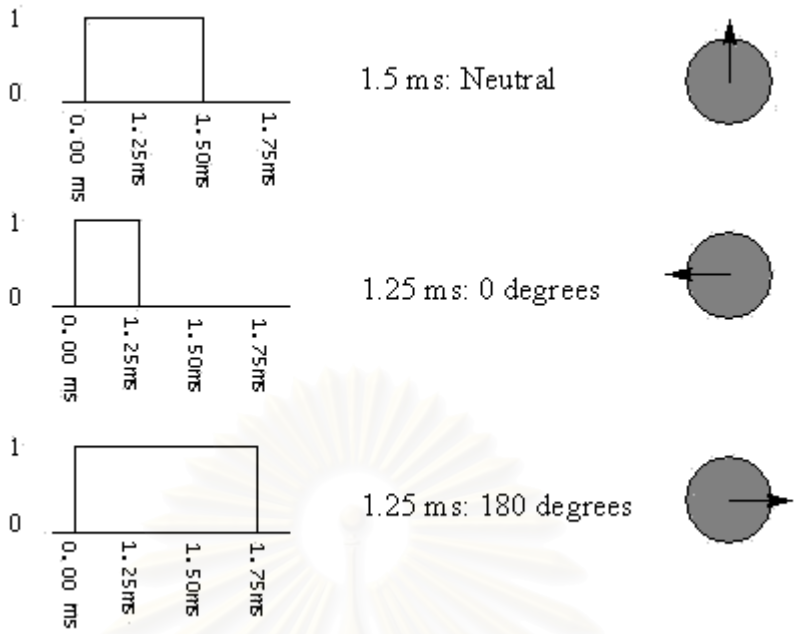
รูปที่ 4.2: แสดงหุ่นยนต์ที่ใช้ในการทดลอง

และข้อเท้ามีลักษณะการหมุนในระนาบหน้าหลัง (sagittal plane) ส่วนของหางมีลักษณะการหมุนในระนาบซ้ายขวา (frontal plane) ในส่วนของหางมีตุ้มน้ำหนักขนาด 150 กรัม ถ่วงอยู่ซึ่งส่วนหางนี้มีไว้สำหรับการโอนถ่ายน้ำหนักไปยังเท้าซ้ายหรือขวาของหุ่นยนต์ ใช้ทำหน้าที่เหมือนลำตัว (torso) ของหุ่นยนต์สองขาแบบทั่วไป การใช้หางแทนที่การใช้ลำตัวในการรักษาสมดุลนั้นมีข้อดีเนื่องจากตุ้มน้ำหนักในตำแหน่งที่ต่ำกว่าสะโพกลงมา ส่งผลให้จุดศูนย์ถ่วงของตัวหุ่นยนต์โดยรวมอยู่ในระดับต่ำกว่าจุดศูนย์ถ่วงของหุ่นยนต์แบบมีลำตัวนั่นเอง หุ่นยนต์ที่ใช้ในงานวิจัยนี้สามารถแสดงได้ดังรูปที่ 4.2

หุ่นยนต์ที่ใช้มีขนาดโดยประมาณ กว้าง 15 เซนติเมตร หนา 10 เซนติเมตร ขณะยืนยืดขาสุดมีความสูง 28 เซนติเมตร และมีน้ำหนัก 700 กรัม ขาท่อนบนและท่อนล่างยาว 8.5 เซนติเมตร ฝ่าเท้ามีขนาด 4.5 x 5 ตารางเซนติเมตร ส่วนหางมีความยาว 20 เซนติเมตร แผงวงจรและหน่วยประมวลผลขนาดเล็กถูกติดตั้งไว้บริเวณเอวของหุ่นยนต์ และสื่อสารกับเครื่องคอมพิวเตอร์ผ่านพอร์ตอนุกรม ในแต่ละข้อต่อจะมีขอบเขตการหมุนที่แตกต่างกันดังแสดงในรูปที่ 4.3



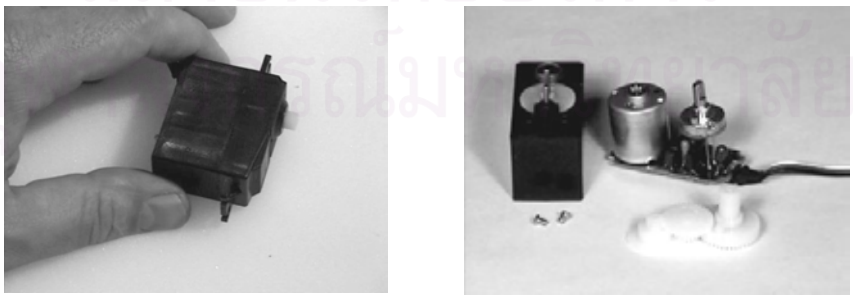
รูปที่ 4.3. แสดงขอบเขตการหมุนของข้อต่อสะโพก หัวเข่า ข้อเท้า และหาง



รูปที่ 4.4: แสดงการควบคุมตำแหน่งของเซอร์โวมอเตอร์ด้วยความกว้างพัลส์ (pulse width modulation)

4.2.2 เซอร์โวมอเตอร์

เซอร์โวมอเตอร์เป็นมอเตอร์ที่ใช้ในเครื่องเล่นประเภทเครื่องบิน เรือ หรือรถบังคับวิทยุ ซึ่งมีคุณสมบัติในการกำหนดตำแหน่งองศาของการหมุนได้โดยใช้ความกว้างของพัลส์เป็นตัวระบุตำแหน่ง (pulse width modulation) (รูปที่ 4.4) ข้อจำกัดของเซอร์โวคือมีขอบเขตการหมุนได้เพียง 180 องศาเท่านั้น โดยทั่วไปเซอร์โวมีลักษณะภายนอกเป็นกล่องสี่เหลี่ยมเล็กๆ (รูปที่ 4.5) ใช้แรงดันไฟฟ้า 4-6 โวลต์ มีสายส่งสัญญาณ 3 เส้นคือ สายดิน(ground) สายไฟเลี้ยง(V_{cc}) และสายสัญญาณ (signal)



(ก) *(ข)*

รูปที่ 4.5: แสดงรายละเอียดเกี่ยวกับเซอร์โว
(ก) แสดงตัวอย่างของเซอร์โวมอเตอร์
(ข) แสดงส่วนประกอบของเซอร์โวมอเตอร์

การทดลองครั้งนี้ใช้เซอร์โวของ Futaba 2 รุ่น คือรุ่น FP-S148 มีทอร์ก (torque) 3 กิโลกรัมเซนติเมตร จำนวน 5 ตัว ติดตั้งไว้ที่ข้อต่อสะโพก 2 ตัว หัวเข่า 2 ตัว และที่หางอีก 1 ตัว ในส่วนข้อต่อนั้น เดิมทีใช้เซอร์โวรุ่นเดียวกันนี้ติดตั้งไว้ แต่หลังจากทดลองควบคุมการเดินดูปรากฏว่าเซอร์โวรุ่นนี้ไม่มีแรงบิดพอที่จะใช้ขับเคลื่อนข้อต่อบริเวณข้อเท้าได้ ดังนั้นจึงเปลี่ยนรุ่นเซอร์โวเป็นอีกรุ่นคือ S9204 ซึ่งมีกำลังบิด 9.5 กิโลกรัมเซนติเมตร ติดตั้งไว้ที่ข้อเท้าทั้ง 2 ข้าง

4.2.3 ส่วนการควบคุม

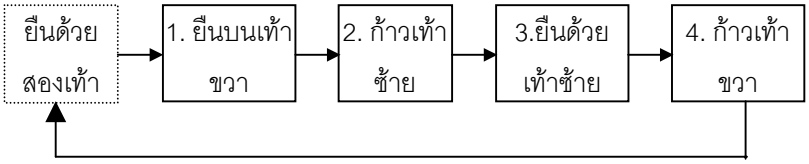
บอร์ดประมวลผลขนาดเล็กถูกติดตั้งอยู่บริเวณเหนือสะโพกของหุ่นยนต์ เพื่อใช้ในการควบคุมการทำงานของมอเตอร์ทั้ง 7 ตัว บอร์ดประมวลผลขนาดเล็กที่ใช้คือ CP-JR6811 ของบริษัท ETT. มีไมโครคอนโทรลเลอร์ MC68HC11E2 ทำหน้าที่ประมวลผล โดยส่งสัญญาณพัลส์ติด-ต่อกับเซอร์โวผ่านทางพอร์ตบี (port B) ซึ่งเป็นพอร์ตสัญญาณขาออกจำนวน 8 บิต และรับคำสั่งตำแหน่งการหมุนของเซอร์โวแต่ละตัวจากเครื่องคอมพิวเตอร์ผ่านพอร์ตอนุกรม (serial port) บนบอร์ดนี้จะมี *"โปรแกรมควบคุมมอเตอร์"*[1] ซึ่งควบคุมเซอร์โวได้ 8 ตัว รูปแบบการใช้งานคือการส่งสายอักขระ 5 ตัวอักษรผ่านพอร์ตอนุกรมจากเครื่องคอมพิวเตอร์ไปสู่ไมโครคอนโทรลเลอร์ โดยมีลักษณะดังตารางที่ 4.1

ตารางที่ 4.1: แสดงลักษณะสายอักขระที่ส่งระหว่างคอมพิวเตอร์กับไมโครคอนโทรลเลอร์

อักษรตัวแรก	ถ้าเป็น 'S' จะหมายถึงการสั่งทำงานเซอร์โว ถ้าเป็น 'R จะหมายถึงการตั้งค่าตำแหน่งมุมของเซอร์โวไปที่ 90^{O}
ตัวอักษรสองตัวถัดมา	เป็นตัวบอกหมายเลขของเซอร์โวที่ต้องการสั่งงาน ตั้งแต่ 00-07
ตัวอักษรสองตัวสุดท้าย	เป็นตัวบอกตำแหน่งมุมที่ต้องการ ในช่วง 180^{O} โดยตัวอักษรจะเป็นตัวเลขฐานสิบหกตั้งแต่ 00 (มุมน้อยที่สุด) ถึง FF (มุมมากที่สุด) ถ้าเป็น 80 หมายถึงตำแหน่งมุมที่ 90^{O}

ตัวอย่างเช่น "S02FF" หมายถึง หมุนเซอร์โวตัวที่สองไปที่มุม 180^{O} เป็นต้น
"S0300" หมายถึง หมุนเซอร์โวตัวที่สองไปที่มุม 0^{O} เป็นต้น
"S0090" หมายถึง หมุนเซอร์โวตัวที่สองไปที่มุม 90^{O} เป็นต้น

[1] โปรแกรมที่ใช้ทำหน้าที่ดังกล่าวนี้ดาวน์โหลดมาจากเวปไซต์
ftp://nyquist.ee.ualberta.ca/pub/motorola/68hc11/servo8.asm.txt



รูปที่ 4.6 ขั้นตอนการเดิน 4 ขั้นตอน

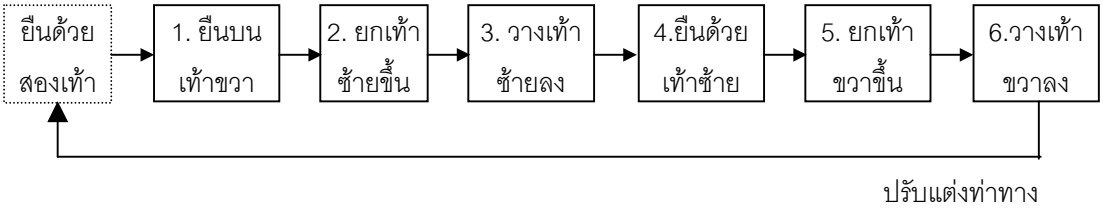
4.3. การออกแบบการทดลอง

4.3.1 ท่าทางการเดิน

เนื่องจากปัญหาการเดินสองขาเป็นปัญหาแบบมีลำดับเวลา (time sequence) กล่าวคือ ในช่วงเวลาแต่ละช่วง หุ่นยนต์จะมีพฤติกรรมแตกต่างกันไป การที่จะใช้ขั้นตอนวิธีพันธุกรรมในการสังเคราะห์หาโปรแกรมควบคุมการเดินทั้งหมดในคราวเดียวเลยจะเป็นเรื่องยาก ดังนั้นในงานวิจัยนี้จึงทำการแบ่งท่าทางการเดินออกเป็นขั้นตอนย่อย ๆ โดยกำหนดให้แต่ละขั้นตอนมีเป้าหมายย่อย (sub-goal) หรือปัญหาย่อย และใช้ขั้นตอนวิธีพันธุกรรมในการหาคำตอบย่อยของแต่ละขั้นตอน หลังจากได้คำตอบย่อยของแต่ละขั้นตอนแล้วจะนำคำตอบย่อยทั้งหมดมารวมกันเป็นคำตอบสุดท้ายซึ่งก็คือโปรแกรมควบคุมท่าทางการเดินของหุ่นยนต์สองขา วิธีการนี้เรียกว่า “การวิวัฒนาการแบบลำดับขั้น” (staged evolution) (Brook, 1991b) ด้วยวิธีการแบ่งปัญหาใหญ่ให้เป็นปัญหาย่อยนี้ทำให้สามารถกำหนดเกณฑ์การวัดค่าความเหมาะสมให้แตกต่างกันไปในแต่ละขั้นตอนได้ ซึ่งจะทำให้สามารถหาคำตอบสุดท้ายได้เร็วขึ้น

ในการแบ่งท่าทางการเดินออกเป็นขั้นตอนย่อย ๆ นี้ ถ้าแบ่งขั้นตอนละเอียดเกินไปจะทำให้มีขั้นตอนย่อยต่าง ๆ มากเกินความจำเป็น ดังนั้นจึงผู้วิจัยพยายามแบ่งท่าทางการเดินให้ขั้นตอนย่อยน้อยที่สุดเท่าที่ขั้นตอนวิธีพันธุกรรมจะสามารถหาคำตอบย่อยของแต่ละขั้นตอนได้ ซึ่งเดิมทีได้แบ่งท่าทางการเดินออกเป็น 4 ขั้นตอน (ดังรูปที่ 4.6)

ในตอนเริ่มต้นการทดลองนั้น หุ่นยนต์จะต้องอยู่ใน “ท่าทางเริ่มต้น” ที่เหมือน ๆ กันทุกครั้ง ซึ่งผู้วิจัยได้เลือกท่า “ยืนสองขา” ซึ่งเป็นท่าเตรียมพร้อมสำหรับการเดินเป็นท่าทางเริ่มต้นสำหรับหุ่นยนต์ จากนั้นจึงเริ่มขั้นตอน 4 ขั้นตอนของการเดินซึ่งได้แก่ 1) การยืนด้วยเท้าขวาเพียงเท้าเดียว ในที่นี้หมายถึงการย้ายจุดศูนย์ถ่วงของหุ่นยนต์ไปไว้เหนือเท้าขวาเท้าเดียว 2) การก้าวเท้าซ้ายไปสัมผัสพื้นด้านหน้า 3) การยืนด้วยเท้าซ้ายเท้าเดียวซึ่งหมายถึงการถ่ายนำหนักจากเท้าขวาไปไว้ที่เท้าซ้าย 4) การก้าวเท้าขวา (ซึ่งขณะนั้นจะอยู่หลังเท้าซ้าย) ขึ้นไปพื้นในตำแหน่งเดียวกับเท้าซ้าย (ตามแนวหน้าหลัง) มองโดยภาพรวมแล้ว การเดินของหุ่นยนต์จะเป็นลักษณะก้าวเดินแบบเท้าซ้ายนำแล้วเท้าขวาตามนั่นเอง



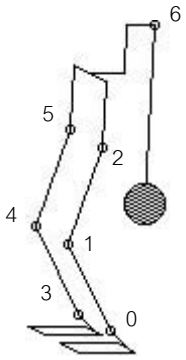
รูปที่ 4.7 ขั้นตอนการเดิน 6 ขั้นตอน

อย่างไรก็ตาม เมื่อได้ทำการทดลองในเบื้องต้นปรากฏว่า ขั้นตอนวิธีพันธุกรรมไม่สามารถหาคำตอบสำหรับขั้นตอนที่สองซึ่งมีเป้าหมายคือการก้าวเท้าซ้ายไปสัมผัสพื้นด้านหน้าได้ เนื่องจากขั้นตอนนี้ยังคงมีความซับซ้อนและประกอบด้วยพฤติกรรมที่ขึ้นกับลำดับเวลาอยู่อีก (ซึ่งจะแสดงรายละเอียดของการทดลองเบื้องต้นของขั้นตอนที่สองนี้ในส่วนต่อไปของบท) ดังนั้นผู้วิจัยจึงจำเป็นต้องแบ่งขั้นตอนที่สองออกเป็นอีกสองขั้นตอน กล่าวคือแบ่ง “การก้าวเท้าซ้ายไปสัมผัสพื้นด้านหน้า” ให้เป็น “การยกเท้าซ้ายไปข้างหน้า” และ “การวางเท้าซ้ายลง” กับพื้น ในทำนองเดียวกัน ขั้นตอนที่สี่ก็จำเป็นต้องแบ่งออกเป็นอีกสองขั้นตอนเช่นเดียวกัน กล่าวคือแบ่ง “การก้าวเท้าขวาไปชิดเท้าซ้าย” ให้เป็น “การยกเท้าขวาขึ้นไปอยู่ในระดับเดียวกับเท้าซ้าย (ในแนวหน้าหลัง)” และ “วางเท้าขวาลง”

โดยสรุปแล้วท่าทางการเดินถูกแบ่งออกเป็น 6 ขั้นตอนย่อย (ดังรูปที่ 4.7) ดังนี้ หุ่นยนต์จะถูกจัดให้อยู่ในท่าทางเริ่มต้นคือท่า “ยืนสองขา” ขั้นตอนทั้งหกได้แก่ 1) การยืนด้วยเท้าขวาเพียงเท้าเดียว 2) การยกขาซ้ายขึ้น 3) การวางเท้าซ้ายลงสัมผัสพื้น 4) การยืนด้วยเท้าซ้ายเท้าเดียว 5) การยกเท้าขวาขึ้น และ 6) การวางเท้าลงกับพื้น ในขั้นตอนที่ 2 และ 5 นั้นเน้นเพียงการยกเท้าไปข้างหน้า (เทียบกับตำแหน่งเดิม) แต่ไม่สนใจว่าเท้าจะวางอยู่กับพื้นหรือลอยเหนือพื้นก็ตาม

4.3.2 โปรแกรมควบคุมการเดิน

ในงานวิจัยนี้จะทำการสังเคราะห์โปรแกรมควบคุมการเดินของหุ่นยนต์สองขา โดยสังเคราะห์โปรแกรมควบคุมการเดินเพียง 1 ก้าวเท่านั้น และใช้โปรแกรมควบคุม 1 ก้าวนี้ซ้ำ ๆ เพื่อให้หุ่นยนต์สามารถเดินได้หลาย ๆ ก้าวอย่างต่อเนื่อง ระบบการควบคุมหุ่นยนต์เป็นระบบควบคุมแบบเปิด (open-loop control system) กล่าวคือ ในช่วงของการก้าวเดิน โปรแกรมจะสั่งให้หุ่นยนต์ขยับข้อต่อต่าง ๆ เพียงอย่างเดียว ข้อมูลสถานะต่าง ๆ ของหุ่นยนต์ เช่น ตำแหน่งของจุดศูนย์ถ่วงของหุ่นยนต์หรือสถานะการสัมผัสพื้นของฝ่าเท้า หรือตำแหน่งของเท้าแต่ละข้าง เป็นต้น จะไม่ถูกนำเข้ามาประมวลผลเลย



รูปที่ 4.8: แสดงหมายเลขกำกับมอเตอร์แต่ละตัวของหุ่นยนต์

ดังนั้นในการใช้โปรแกรมควบคุมตัวเดิมควบคุมให้หุ่นยนต์สามารถเดินก้าวถัดไปได้อย่างต่อเนื่องหลังจากเดินได้สำเร็จก้าวหนึ่งแล้วนั้น จำเป็นที่จะต้องทำให้หุ่นยนต์กลับมาอยู่ในท่าทางเริ่มต้นคือ ท่า “ยืนสองขา” เหมือนเดิมเสียก่อน ซึ่งเดิมทีขั้นตอนการปรับแต่งท่าทางนี้ถูกบรรจุอยู่ในเป้าหมายย่อยของขั้นตอนที่หกด้วย กล่าวคือ ขั้นตอนที่หกมีเป้าหมายให้หุ่นยนต์วางเท้าขวาลงกับพื้นโดยให้มุมของทุกข้อต่อเหมือนกับตอนท่าทางเริ่มต้น อย่างไรก็ตามในการผนวกการปรับแต่งท่าทางเข้าไปอยู่ในขั้นตอนที่หกด้วยนั้นทำให้เป้าหมายของขั้นตอนนี้มีขนาดใหญ่เกินกว่าที่ขั้นตอนวิธีพันธุกรรมจะหาคำตอบให้ได้ (ซึ่งจะได้แสดงรายละเอียดการหาคำตอบสำหรับขั้นตอนนี้ในส่วนต่อไปของ บท) ดังนั้นผู้วิจัยจึงได้แยกหน้าที่การปรับแต่งท่าทางนี้ออกมาจากขั้นตอนที่หก โดยหน้าที่การปรับแต่งท่าทางนี้จะเป็นหน้าที่ของ “*โปรแกรมอ่าน-แปลคำตอบ*” แทน

โปรแกรมคำตอบที่สังเคราะห์ได้ในงานวิจัยนี้เป็นโปรแกรมควบคุมหุ่นยนต์ที่ควบคุมในระดับการหมุนของมอเตอร์ รูปแบบของโปรแกรมควบคุมหุ่นยนต์จะอยู่ในลักษณะสายลำดับของคำสั่งควบคุมมอเตอร์ (sequence of motor commands) ในที่นี้จะขอแทนลักษณะนามของ “คำสั่งควบคุมมอเตอร์” เป็น “โนด” เพื่อป้องกันการสับสนเมื่อมีการกล่าวอ้างถึง คำสั่งควบคุมมอเตอร์ในแต่ละโนดจะมีลักษณะดังนี้

m : r

โดยที่ m คือ คำสั่งควบคุมมอเตอร์ {0+, 0-, 1+, 1-, 2+, 2-, 3+, 3-, 4+, 4-, 5+, 5-, 6+, 6-}

r คือ “จำนวนกระทำซ้ำ”ของคำสั่งควบคุมมอเตอร์

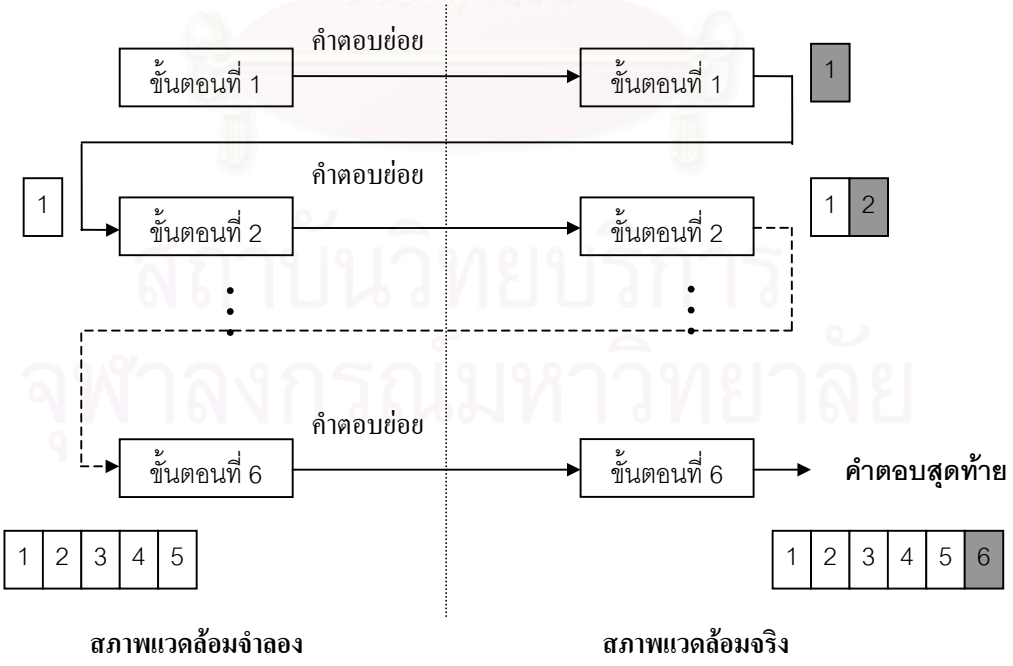
คำสั่งควบคุมมอเตอร์จะควบคุมให้มอเตอร์ขยับไปมา เป็นมุมครั้งละ 1.34° ตัวเลข 0 - 6 เป็นหมายเลขกำกับมอเตอร์แต่ละตัว (รูปที่ 4.8) และเครื่องหมาย ‘+’, ‘-’ เป็นตัวบ่งบอกว่าการขยับเพิ่มหรือลดมุมของมอเตอร์ตัวนั้น ๆ ยกตัวอย่างเช่น **3+:2, 6+:27, 0+:2, 4+:2, 1-:6, …, 3-:3,**

ในโปรแกรมควบคุมนี้ คำสั่งแรกคือ เพิ่มมุมของมอเตอร์หมายเลขสามไป 2 ครั้ง กล่าวคือขยับข้อเท้าขวานั่นเอง

การใช้รูปแบบของสายคำสั่งควบคุมหุ่นยนต์ในลักษณะนี้ทำให้ง่ายต่อการทำความเข้าใจและการอ่านคำสั่งเพื่อควบคุมหุ่นยนต์จะทำได้ง่าย แต่วิธีนี้มีข้อเสียอยู่หนึ่งอย่างคือสายคำสั่งสายเดียวกันเมื่อใช้ควบคุมหุ่นยนต์ที่มีท่าทางเริ่มต้นต่างกันก็จะทำให้พฤติกรรมที่แสดงออกมาแตกต่างกันไปด้วย ดังนั้นในการใช้รูปแบบสายคำสั่งแบบนี้จำเป็นต้องระวังในเรื่องท่าทางเริ่มต้นของหุ่นยนต์ด้วย

4.1.3 ระบบของการทดลอง

ข้อจำกัดของงานวิจัยที่เกี่ยวกับการสังเคราะห์โปรแกรมควบคุมหุ่นยนต์โดยใช้ขั้นตอนวิธีพันธุกรรมคือ ไม่สามารถใช้วิธีนี้กับหุ่นยนต์ที่มีจำนวนข้อต่อมากนัก และอีกข้อจำกัดหนึ่งซึ่งงานวิจัยส่วนใหญ่จะต้องพบคือไม่สามารถทำการทดลองกับหุ่นยนต์จริง เนื่องจากการทดลองกับหุ่นยนต์จริงใช้เวลามาก และเนื่องจากวิธีการคำนวณเชิงวิวัฒน์นี้จำเป็นต้องมีการประเมินค่าความเหมาะสมของคำตอบจำนวนมหาศาล ซึ่งการประเมินกับหุ่นยนต์จริงอาจทำให้เกิดความเสียหายกับหุ่นยนต์ได้ ดังนั้นงานวิจัยส่วนใหญ่จะเลี่ยงข้อจำกัดนี้โดยใช้การทดลองในสภาพแวดล้อมจำลอง (Simulation) แทน อย่างไรก็ตาม เป็นที่รู้กันดีว่า การนำคำตอบที่ได้จากการเรียนรู้ในสภาพแวดล้อมจำลองไปใช้กับหุ่นยนต์จริงมักไม่ประสบความสำเร็จ (Brooks, 1991a) เนื่องจากการทำงานกับหุ่นยนต์จริงใน



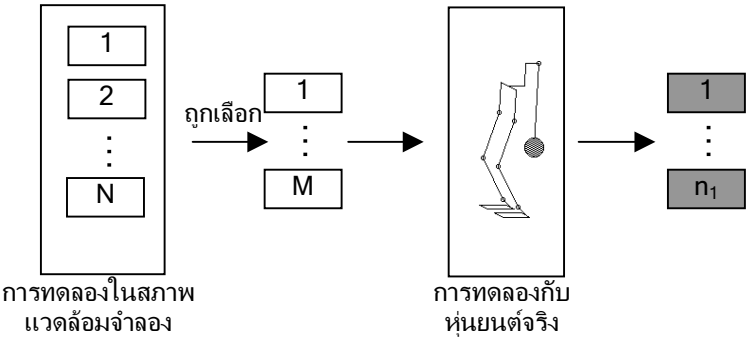
รูปที่ 4.9: แสดงความสัมพันธ์ของการทดลองในสภาพแวดล้อมจำลองและสภาพแวดล้อมจริงของแต่ละขั้นตอนของการเดิน

สภาพแวดล้อมจริงนั้นมีรายละเอียดมาก มีเงื่อนไขหรือสถานการณ์ที่ไม่คาดคิดเกิดขึ้นมากมาย ซึ่งไม่สามารถจำลองสิ่งเหล่านี้ได้หมด ดังนั้น ในการเพิ่มอัตราความสำเร็จของการทำงานกับหุ่นยนต์จริง จำเป็นต้องทำให้ระบบของการทดลองมีการเชื่อมโยงกับสภาพแวดล้อมจริงมากขึ้น

วิธีการหนึ่งที่ใช้ในระบบการทดลองของงานวิจัยนี้คือ การนำการทดลองในสภาพแวดล้อมจำลอง (simulation) และการทดลองในสภาพแวดล้อมจริง (real experiment) ประกอบร่วมกัน (รูปที่ 4.9) กล่าวคือ คำตอบย่อยที่สังเคราะห์ได้จากการทดลองในสภาพแวดล้อมจำลองของขั้นตอนแต่ละขั้นตอนจะถูกคัดเลือกและนำมาประเมินพฤติกรรม โดยทดลองกับหุ่นยนต์จริงในขั้นตอนเดียวกัน คำตอบย่อยที่ใช้ได้กับหุ่นยนต์จริง (กล่องสี่เหลี่ยมแรเงาในรูป 4.9) จะถูกนำไปใช้เป็นท่าทางเริ่มต้นของขั้นตอนถัดไปของการทดลองในสภาพแวดล้อมจำลอง กระบวนการเช่นนี้จะดำเนินไปในทุก ๆ ขั้นตอนจนถึงขั้นตอนที่ 6 ก็จะได้คำตอบสุดท้ายที่ประกอบด้วย 6 คำตอบย่อย (ดังรูปที่ 4.9) การจัดระบบการทดลองให้มีลักษณะนี้ เพื่อให้สามารถประกันว่าคำตอบที่ได้มาโดยใช้เวลาน้อย (ซึ่งเป็นข้อดีของการทดลองในสภาพแวดล้อมจำลอง) นั้นสามารถใช้ได้กับหุ่นยนต์จริงด้วย ซึ่งโดยปกติแล้วคำตอบที่ได้จากสภาพแวดล้อมจำลองนี้จะมีลักษณะหลากหลาย มีทั้งที่ใช้ได้และใช้ไม่ได้กับหุ่น-ยนต์จริง การทดลองกับหุ่นยนต์จริงมีขึ้นเพื่อคัดเลือกเฉพาะคำตอบที่ประสบความสำเร็จกับหุ่นยนต์จริงเท่านั้น และกระบวนการทำงานของระบบการทดลองในรูปที่ 4.9 มีรายละเอียดดังนี้

1. การทดลองในแต่ละขั้นตอนนั้น จะเริ่มด้วยการทดลองในสภาพแวดล้อมจำลองก่อน ซึ่งหุ่นยนต์จะถูกจัดให้อยู่ในท่าทางเริ่มต้น ขั้นตอนวิธีพันธุกรรมถูกใช้ในการสังเคราะห์หาโปรแกรมคำตอบตามเป้าหมายย่อยที่กำหนดไว้ในแต่ละขั้นตอน (ด้วยจำนวนประชากรในแต่ละรุ่นเป็น N) (ดังรูปที่ 4.10) ในการประเมินค่าความเหมาะสมนั้น จะใช้วิธีแปลโปรแกรมคำตอบให้เป็นพฤติกรรมของหุ่นยนต์แล้วจึงนำข้อมูลที่เกิดขึ้นมาคำนวณ

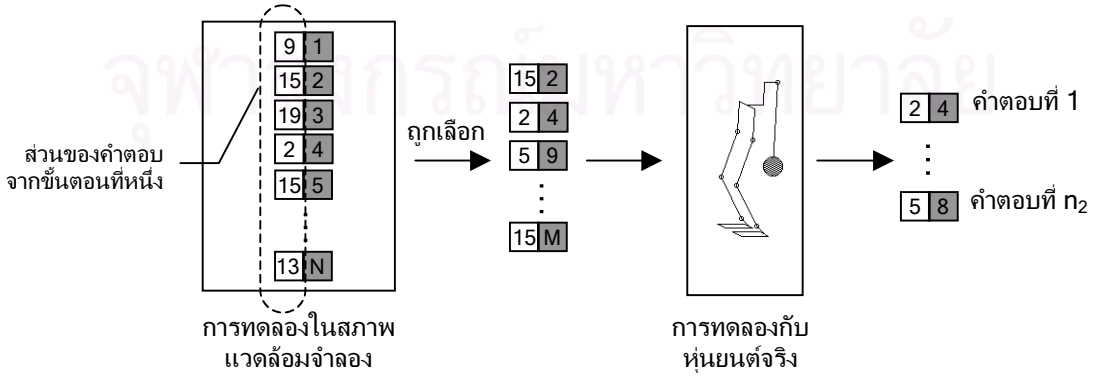
2. หลังจากโปรแกรมคำตอบถูกวิวัฒนาการไปรุ่นแล้วรุ่นเล่าจนถึงรุ่นสุดท้าย ระบบจะเลือกคำตอบมาจำนวนหนึ่ง (M ตัว) จากจำนวนประชากร N ตัว ($M < N$) เพื่อนำมาทดสอบความสำเร็จในการทำงานของโปรแกรมคำตอบเมื่อใช้กับหุ่นยนต์จริง คำตอบที่ประสบความสำเร็จจากการทดลองกับหุ่นยนต์จริง (กล่องแรเงาในรูป 4.9) จะถูกนำมาใช้เป็นท่าทางเริ่มต้นของหุ่นยนต์สำหรับขั้นตอนการเดินถัดไป ซึ่งคำตอบที่ได้ในที่นี้อาจมีมากกว่าหนึ่งคำตอบ และมีรูปแบบของสายคำสั่งมากกว่าหนึ่งลักษณะ (ในที่นี้ กำหนดให้คำตอบที่ได้มี n_1 ลักษณะ, $n_1 < M < N$) รูปที่ 4.10 แสดงภาพการทำงานของการทดลองในขั้นตอนที่หนึ่ง ซึ่งมีท่าทางเริ่มต้นเป็นการยืนสองเท้า



รูปที่ 4.10 : แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่หนึ่ง

3. เนื่องจากคำตอบที่ได้จากขั้นตอนที่หนึ่งถูกนำมาใช้เป็นท่าทางเริ่มต้นของขั้นตอนที่สอง และท่าทางสุดท้ายที่เกิดจากโปรแกรมคำตอบแต่ละตัวในจำนวน n_1 ตัว (ที่ได้จากขั้นตอนที่หนึ่ง) นั้นอาจมีได้มากกว่าหนึ่งรูปแบบ อีกทั้งจากลักษณะของโปรแกรมคำตอบซึ่งเป็นคำสั่งควบคุมมอเตอร์แบบสัมพัทธ์กับสถานะปัจจุบัน (ดังที่อธิบายไว้ในหัวข้อที่แล้ว) ดังนั้นจึงมีความจำเป็นที่จะต้องเชื่อมโยงคำตอบที่จะสังเคราะห์ขึ้นในขั้นตอนที่สองนี้ว่าเป็นคำตอบที่มาจากท่าทางเริ่มต้นใด (ในคำตอบจำนวน n_1 ตัวที่ได้จากขั้นตอนที่หนึ่ง) ดังรูป 4.11

4. ในทางปฏิบัติ ลักษณะการเชื่อมโยงนี้จะทำโดยนำส่วนที่เป็นคำตอบจากขั้นตอนที่หนึ่ง (กล่องสีขาวในรูปที่ 4.11) มา “ต่อ” กับคำตอบที่สังเคราะห์ขึ้นในขั้นตอนที่สองนี้ (กล่องสีเทาในรูปที่ 4.11) เพื่อให้ง่ายต่อการพิจารณาความเชื่อมโยงดังกล่าว แต่เนื่องจากจำนวนคำตอบที่ได้จากขั้นตอนที่หนึ่งนั้นมี n_1 ตัว ซึ่งค่อนข้างน้อยกว่าจำนวนประชากรมาก ($n_1 < N$) ดังนั้นในการเชื่อมโยงกัน คำตอบที่ได้จากขั้นตอนที่หนึ่งจะเชื่อมโยงกับคำตอบในขั้นตอนที่สองมากกว่า 1 คำตอบ ซึ่งในทางปฏิบัตินั้นจะกระทำโดยการนำคำตอบจากขั้นตอนที่หนึ่ง (จำนวน n_1 ตัว) มาขยายจำนวนให้เท่ากับจำนวนประชากรทั้งหมด (N ตัว) โดยใช้หลักที่ว่าคำตอบที่มีค่าความเหมาะสมมากก็จะถูกขยายจำนวนออกมามาก คำตอบที่มีค่าความเหมาะสมน้อยจะถูกขยายออกมาน้อย



รูปที่ 4.11 : แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่สอง

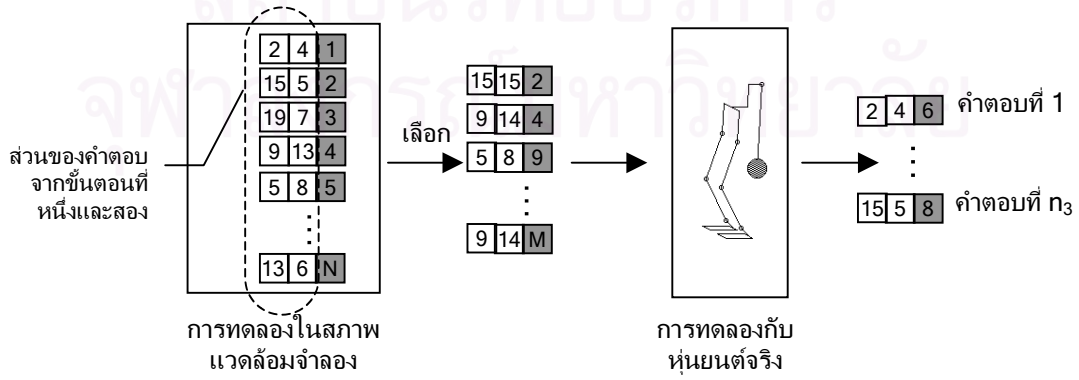
5. จากรูปที่ 4.11 กล่องสีขาว คือ ส่วนของคำตอบที่ได้จากขั้นตอนที่หนึ่ง ตัวเลขที่ระบุอยู่ในกล่องสีเทา แสดงถึงว่าคำตอบที่บรรจุอยู่ภายในกล่องอาจมีรูปแบบเหมือนหรือไม่เหมือนกันก็ได้ ถ้าตัวเลขเหมือนกันก็หมายถึงลักษณะคำตอบที่บรรจุอยู่เหมือนกัน ถ้าตัวเลขต่างกันก็หมายถึงลักษณะคำตอบที่บรรจุอยู่แตกต่างกันไป ส่วนกล่องสีเทา คือ ส่วนของคำตอบที่จะทำการสังเคราะห์ขึ้นและวิวัฒนาการในขั้นตอนที่สอง

6. ส่วนของคำตอบในกล่องสีขาวจะไม่ถูกดำเนินการทางพันธุกรรมใด ๆ ทั้งสิ้น ไม่ว่าจะเป็นการไขว้เปลี่ยนหรือการกลายพันธุ์ก็ตาม ในขั้นตอนที่สองนี้ ขั้นตอนวิธีพันธุกรรมจะดำเนินการทางพันธุกรรมเฉพาะกับส่วนของกล่องสีเทาเท่านั้น ในการคัดเลือกคำตอบไปใช้สำหรับรุ่นถัดไปนั้นจะพิจารณาจากค่าความเหมาะสมหรือค่าความแตกต่างเฉพาะในส่วนของกล่องสีเทาเท่านั้น

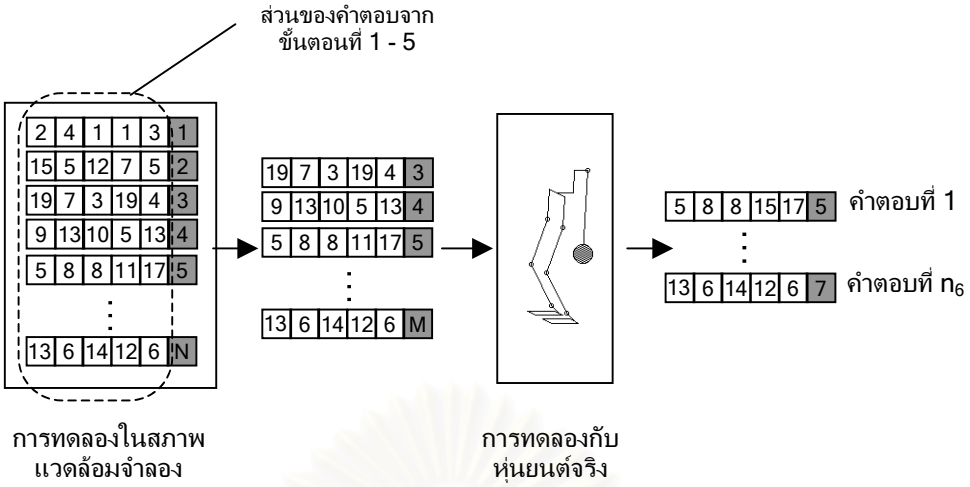
7. ในการประเมินค่าความเหมาะสมของคำตอบจะพิจารณาจากส่วนที่เป็นกล่องสีเทาเท่านั้นเช่นกัน โดยมีวิธีการคือ แปลคำตอบในส่วนกล่องสีขาวทั้งหมดเพื่อแสดงพฤติกรรมหุ่นยนต์ก่อน หลังจากแปลเสร็จหุ่นยนต์จะอยู่ในท่าทางเริ่มต้น (สำหรับขั้นตอนที่สอง) แล้ว จากนั้นจึงแปลคำตอบในส่วนของกล่องสีเทาเป็นพฤติกรรมหุ่นยนต์ แล้วจึงนำข้อมูลที่ได้การแปลผลคำตอบของกล่องสีเทานี้มาคำนวณค่าความเหมาะสม

8. หลังจากการวิวัฒนาการด้วยขั้นตอนวิธีพันธุกรรมในสภาพแวดล้อมจำลองจนได้คำตอบในรุ่นสุดท้ายแล้ว กระบวนการทำงานในช่วงนี้จะเหมือนเดิม คือคำตอบบางส่วนจะถูกเลือกมาเพื่อใช้ทดลองกับหุ่นยนต์จริงเพื่อตรวจสอบดูว่าคำตอบที่ได้นั้นทำให้หุ่นยนต์แสดงพฤติกรรมได้ตามเป้าหมายย่อยของขั้นตอนที่สองนี้โดยไม่มีการล้มหรือบิดตัวหรือไม่ และคำตอบที่ผ่านการตรวจสอบดังกล่าวก็จะถูกนำไปใช้เป็นท่าทางเริ่มต้นสำหรับขั้นตอนการเดินถัดไปเช่นเดิม

9. คำตอบซึ่งได้จากขั้นตอนที่สอง (ซึ่งมีคำตอบจากขั้นตอนที่หนึ่งต่ออยู่ด้วย) จะถูกนำมาใช้เป็นท่าทางเริ่มต้นสำหรับขั้นตอนที่สามต่อไป โดยส่วนของคำตอบจากขั้นตอนที่หนึ่งและ



รูปที่ 4.12 : แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่สาม



รูปที่ 4.13 : แสดงกระบวนการทำงานของการทดลองในขั้นตอนที่หก

สองจะถูกนำมาต่อกับคำตอบที่สังเคราะห์ขึ้นและวิวัฒนาการของขั้นตอนที่สาม จากรูปที่ 4.12 ขั้นตอนพันธุกรรมจะดำเนินการทางพันธุกรรมต่าง ๆ เฉพาะกับกล่องสีเทาเช่นกัน การประเมินค่าความเหมาะสมของคำตอบมีการกระทำในลักษณะเดียวกับขั้นตอนที่สองคือ แปลผลคำตอบที่ได้จากขั้นตอนที่หนึ่งและสอง (กล่องสีขาว) เพื่อทำให้หุ่นยนต์อยู่ในท่าทางเริ่มต้นก่อน จากนั้นจึงแปลผลคำตอบของขั้นตอนที่สามเพื่อนำข้อมูลที่เกิดขึ้นมาคำนวณหาค่าต่อไป การคัดเลือกคำตอบไปใช้สำหรับรุ่นถัดไปก็กระทำเหมือนในขั้นตอนที่สองเช่นกัน

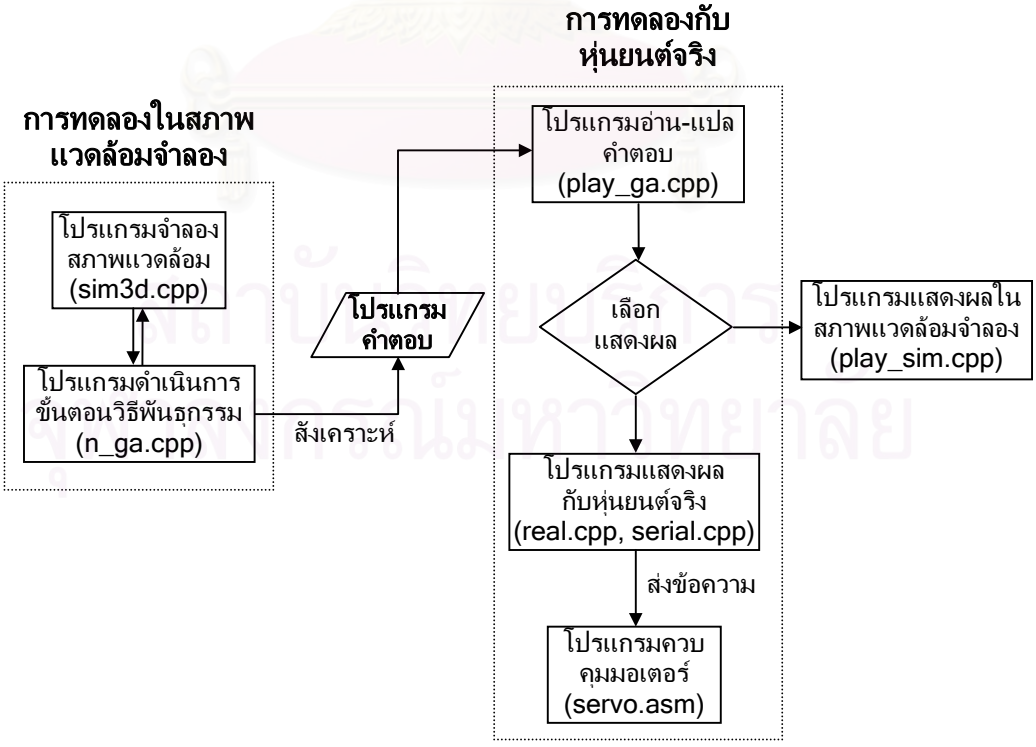
10. กระบวนการทำงานจะกระทำซ้ำในลักษณะเดียวกันนี้ไปทุก ๆ ขั้นตอนจนถึงขั้นตอนที่หก ซึ่งจะมีรายละเอียดดังรูปที่ 4.13 ในส่วนการทดลองในสภาพแวดล้อมจำลองของขั้นตอนนี้จะมีโปรแกรมคำตอบในส่วนหัวซึ่งได้มาจากการทดลองตั้งแต่ขั้นตอนที่ 1 - 5 (กล่องสีขาวต่อกัน 5 กล่อง) มาเป็นท่าทางเริ่มต้นให้กับคำตอบของขั้นตอนที่หกนี้ (กล่องสีเทา)

การทดลองในสภาพแวดล้อมจำลองนั้น ตัวระบบสามารถดำเนินไปได้เองโดยอัตโนมัติ แต่ในส่วนของการทดลองกับหุ่นยนต์จริงนั้น ขั้นตอนบางอย่างยังคงต้องอาศัยมนุษย์ช่วยในการดำเนินงาน ได้แก่ การประเมินพฤติกรรมที่เกิดขึ้นของคำตอบแต่ละตัว ซึ่งต้องประเมินจากสภาวะเดียวกันหรือมีสถานะตั้งต้นเหมือนกัน กล่าวคือ หุ่นยนต์จะต้องอยู่ในท่าทางเริ่มต้นเหมือน ๆ กันทุกครั้งก่อนการอ่านและแปลคำตอบแต่ละตัวให้เป็นพฤติกรรมของหุ่นยนต์ อีกทั้งเนื่องจากหุ่นยนต์มีโอกาสล้มหรือบิดตัวได้ในทุก ๆ คำตอบ ดังนั้นในการจัดวางให้หุ่นยนต์อยู่ในสภาวะดังกล่าวจึงจำเป็นต้องอาศัยมนุษย์ช่วยตรงจุดนี้ หรือกล่าวอีกนัยหนึ่งคือ ในการทดลองกับหุ่นยนต์จริงนั้นจำเป็นต้องอาศัยมนุษย์ในการยกหุ่นยนต์ไปวางไว้ในอยู่ในท่ายืนสองขาทุก ๆ ครั้งก่อนการประเมินคำตอบตัวต่อไป อีกขั้นตอนหนึ่งที่ใช้มนุษย์เข้าไปช่วยเหลือในระบบคือ การตัดสินพฤติกรรมของคำตอบแต่ละตัวว่าใช้ได้หรือไม่ ซึ่งจะกล่าวถึงในส่วนถัดไปของบท

4.1.4 ภาพรวมระบบการทำงาน

ในการทดลองของงานวิจัยนี้ประกอบด้วยโปรแกรมการทำงานต่าง ๆ ซึ่งทำงานร่วมกัน การทำงานของระบบทั้งหมดเพื่อให้ได้มาซึ่งโปรแกรมคำตอบที่สามารถใช้ในการควบคุมหุ่นยนต์จริง ให้เดินได้ โดยใช้ขั้นตอนวิธีพันธุกรรมทำการพัฒนาโปรแกรมคำตอบในสภาพแวดล้อมจำลองเพื่อหาโปรแกรมคำตอบที่มีความเหมาะสมมาจำนวนหนึ่ง จากนั้นจึงนำโปรแกรมคำตอบเหล่านี้มาทดลองกับหุ่นยนต์จริงว่าประสบความสำเร็จหรือไม่ หลังจากผ่านการทดลองกับหุ่นยนต์จริงแล้วก็จะได้โปรแกรมคำตอบสมบูรณ์ที่สามารถใช้ได้กับหุ่นยนต์จริง

ภาพรวมระบบการทำงานต่าง ๆ สามารถแสดงได้ดังรูปที่ 4.14 ซึ่งประกอบด้วยโปรแกรมต่าง ๆ ได้แก่ **"โปรแกรมดำเนินการขั้นตอนวิธีพันธุกรรม"** ซึ่งเป็นโปรแกรมที่ทำหน้าที่ดำเนินการเกี่ยวกับขั้นตอนวิธีพันธุกรรมต่าง ๆ ตั้งแต่ การสร้างกลุ่มคำตอบ การคัดเลือก การไขว้เปลี่ยน และการกลายพันธุ์ ในการประเมินค่าความเหมาะสมของคำตอบแต่ละตัวนั้น โปรแกรมดำเนินการขั้นตอนวิธีพันธุกรรมจะทำงานร่วมกับ **"โปรแกรมจำลองสภาพแวดล้อม"** โดยทดลองควบคุมให้หุ่นยนต์เดินในสภาพแวดล้อมจำลองเพื่อพิจารณาตรวจสอบเงื่อนไขต่าง ๆ ที่เกิดขึ้น เช่น หุ่นยนต์ล้มหรือบิดตัวหรือไม่, หุ่นยนต์แสดงพฤติกรรมเป็นไปตามเป้าหมายย่อย ๆ ของขั้นตอนแต่ละขั้นตอนอย่างไร เป็นต้น โปรแกรม 2 หน่วยนี้ทำงานประกอบกันเป็น "*การทดลองในสภาพแวดล้อมจำลอง*" เพื่อสังเคราะห์โปรแกรมคำตอบในเบื้องต้นจำนวนหนึ่ง



รูปที่ 4.14: แสดงภาพรวมระบบการทำงานต่าง ๆ

หลังจากที่สามารถสังเคราะห์โปรแกรมคำตอบจากการทดลองในสภาพแวดล้อมจำลองแล้ว โปรแกรมคำตอบเหล่านี้จะถูกนำมาทำ *"การทดลองกับหุ่นยนต์จริง"* โดยมี **"โปรแกรมอ่าน-แปลคำตอบ"** เป็นตัวอ่านและแปลโปรแกรมคำตอบซึ่งอยู่ในรูปแบบของสายคำสั่งของการควบคุมมอเตอร์ (ดังรูปที่ 4.14) ทีละโนด ๆ และส่งข้อมูลให้ **"โปรแกรมแสดงผลกับหุ่นยนต์จริง"** โปรแกรมแสดงผลกับหุ่นยนต์จริงจะแปลงข้อมูลคำสั่งเพื่อส่งข้อความไปให้ **"โปรแกรมควบคุมมอเตอร์"** ซึ่งถูกดาวน์โหลดลงในไมโครคอนโทรลเลอร์ที่ติดตั้งอยู่บนตัวหุ่นยนต์ โปรแกรมควบคุมมอเตอร์จะรับข้อความมา และแปลงข้อความให้เป็นสัญญาณพัลส์เพื่อใช้ควบคุมเซอร์โวมอเตอร์ซึ่งติดตั้งอยู่ตามข้อต่อทั้งเจ็ดของหุ่นยนต์ เพื่อให้หุ่นยนต์แสดงพฤติกรรมต่าง ๆ ต่อไป ซึ่งพฤติกรรมต่าง ๆ ที่หุ่นยนต์แสดงออกมาก็จะถูกใช้ในการประเมินและตัดสินโดยมนุษย์ว่าโปรแกรมคำตอบนั้น ๆ สามารถใช้ได้กับหุ่นยนต์จริงหรือไม่

ในกรณีที่ได้โปรแกรมคำตอบสมบูรณ์ที่สามารถใช้ได้กับหุ่นยนต์จริงแล้ว ถ้าต้องการแสดงผลของโปรแกรมคำตอบนั้นอีกครั้ง ก็สามารถทำได้ โดยใช้ชุดโปรแกรมต่าง ๆ ของ "การทดลองกับหุ่นยนต์จริง" เพื่อแสดงพฤติกรรมที่เกิดขึ้นกับหุ่นยนต์จริง นอกจากนี้ ยังมี **"โปรแกรมแสดงผลในสภาพแวดล้อมจำลอง"** ซึ่งมีหลักการทำงานเหมือนกับโปรแกรมจำลองสภาพแวดล้อม แต่สามารถใช้ในกรณีที่ต้องการแสดงผลของโปรแกรมคำตอบใด ๆ ในสภาพแวดล้อมจำลองอีกด้วย

4.4. ขั้นตอนวิธีพันธุกรรม

4.4.1 การเข้ารหัสของคำตอบ (individual representation)

คำตอบจะถูกเข้ารหัสให้ประกอบด้วย 2 ส่วน ได้แก่ 1) ความยาว เป็นตัวกำหนดความยาวของสายลำดับคำสั่ง และ 2) สายลำดับของคำสั่งควบคุมการเดิน (ดังรูปที่ 4.15) ซึ่งความยาวของสายลำดับคำสั่งในการทดลองจะถูกจำกัดไว้ไม่เกิน 10 คำสั่งต่อหนึ่งขั้นตอนย่อย เพราะฉะนั้นความยาวของสายลำดับคำสั่งของทั้ง 6 ขั้นตอนรวมกันจะไม่เกิน 60 คำสั่ง

ความยาว	สายลำดับของคำสั่งควบคุมการเดิน

รูปที่ 4.15: แสดงลักษณะการเข้ารหัสของคำตอบ

4.4.2 การประเมินค่าความเหมาะสม (fitness evaluation)

ในการวัดค่าความเหมาะสมของคำตอบแต่ละตัว โปรแกรมดำเนินการขั้นตอนวิธีพันธุกรรมจะส่ง "สายลำดับของคำสั่งควบคุมการเดิน" ของคำตอบทั้งสายให้โปรแกรมจำลองสภาพแวดล้อม โปรแกรมจำลองสภาพแวดล้อมจะอ่านคำสั่งทีละคำสั่ง และสั่งงานหุ่นยนต์ (จำลอง) ให้แสดงพฤติกรรมต่าง ๆ ตามรหัสของคำสั่ง เป็นเช่นนี้ไปจนกระทั่งครบกำหนดความยาวตามที่ระบุไว้ในส่วน "ความยาว" แล้วจึงประเมินค่าความเหมาะสม ในกรณีที่หุ่นยนต์ "ล้ม" หรือ "บิดตัว" ก่อนครบกำหนดความยาวของสายลำดับ โปรแกรมจำลองสภาพแวดล้อมก็จะหยุดการอ่านเช่นกัน คำสั่งที่เหลือในสายลำดับที่ยังไม่ถูกอ่านจะถือเป็นโมฆะ ไม่ถูกพิจารณา

ในส่วนของเกณฑ์การวัดค่าความเหมาะสม (fitness function) เนื่องจากการแบ่งท่าทางการเดินออกเป็นขั้นตอนย่อย 6 ขั้นตอน ทำให้จะสามารถแบ่งเกณฑ์การวัดค่าความเหมาะสมได้ 2 ชนิดซึ่งเป็นฟังก์ชันแบบหาค่ามากที่สุด ดังนี้

1) <u>เกณฑ์การวัดค่าความเหมาะสมทั่วไป (general fitness function)</u> ใช้ในการวัดค่าความเหมาะสมของคำตอบในทุก ๆ ขั้นตอน เกณฑ์การวัดค่าความเหมาะสมทั่วไปนี้เป็นตัวควบคุมให้คำตอบที่ได้เป็นคำตอบที่ทำให้หุ่นยนต์เดินหน้าได้ตรงและไม่ล้ม ซึ่งมีสมการดังนี้

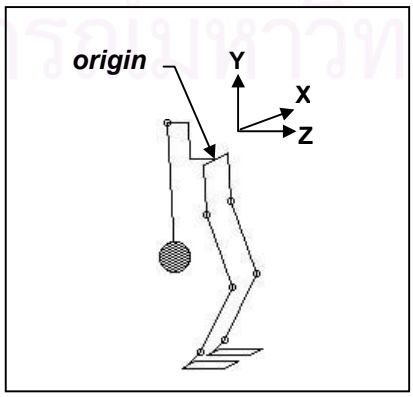
$$Fitness = (k_1F + k_2R) \times T$$

โดยที่ F หมายถึง สถานะการล้มของหุ่นยนต์ มีค่าเป็น 1 เมื่อหุ่นยนต์ล้ม และ เป็น 0 เมื่อหุ่นยนต์ไม่ล้ม

R หมายถึง สถานะการบิดตัวของหุ่นยนต์ มีค่าเป็น 1 เมื่อหุ่นยนต์บิดตัว และเป็น 0 เมื่อหุ่นยนต์ไม่บิดตัว

T หมายถึง ระยะเวลาที่หุ่นยนต์สามารถรักษาสมดุลได้โดยไม่ล้มหรือบิดตัว

$k_1 = -10,\ k_2 = -5.$



รูปที่ 4.16: แสดงการตั้งแกนของหุ่นยนต์

2) <u>เกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอน (particular fitness function)</u> ใช้วัดค่าความเหมาะสมของคำตอบในขั้นตอนย่อยแต่ละขั้นตอน ซึ่งข้อมูลและสมการที่ใช้พิจารณาประเมินค่าความเหมาะสมของคำตอบแต่ละคำตอบตามจะแตกต่างกันไปตามเป้าหมายย่อยของแต่ละขั้นตอน

ในส่วนของการทดลองในสภาพแวดล้อมจำลองนั้น การวัดระยะต่าง ๆ ของหุ่นยนต์จะวัดตามการตั้งแกนดังรูปที่ 4.16 โดยกำหนดให้มีจุดกำเนิด (origin) อยู่ที่จุดกึ่งกลางบริเวณเอวของหุ่นยนต์ (ดังรูปที่ 4.16) ซึ่งการทดลองเบื้องต้นนั้น เกณฑ์การวัดค่าความเหมาะสมนอกจากจะกำหนดอ้างอิงตามเป้าหมายย่อยของแต่ละขั้นตอนแล้ว ยังมีชี้นำจากผู้วิจัยโดยการเพิ่มตัวแปรชี้นำ (guide variable) เข้าไปในสมการด้วย เพื่อทดสอบให้แน่ใจว่าวิธีการแบ่งท่าทางการเดินเป็นขั้นตอนย่อย ๆ นี้สามารถทำให้ขั้นตอนวิธีพันธุกรรมหาคำตอบได้จริง ซึ่งตัวแปรชี้นำที่ใส่เพิ่มเข้าไปในแต่ละขั้นตอนนั้น กำหนดโดยใช้ความรู้พื้นฐานของมนุษย์อ้างอิงตามเป้าหมายย่อยของขั้นตอนนั้น ๆ และสมการเกณฑ์การวัดค่าความเหมาะสมของแต่ละขั้นตอน ซึ่งเป็นฟังก์ชันแบบหาค่ามากที่สุด (maximized function) สามารถแสดงได้ดังตารางที่ 4.2

ตารางที่ 4.2: แสดงเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนของการทดลองในสภาพแวดล้อมจำลอง เบื้องต้น

$F_1 = \frac{1}{\Delta S_R} + n_{tail_rht} - n_{leg}$ $\Delta S_R = \sqrt{(cg_x - pr_x)^2 + (cg_z - pr_z)^2}$
$F_2 = \frac{k}{\Delta z} - \Delta y - n_{left_backward} - n_{tail} + n_{leg}$ $\Delta z = \lvert pl_z - (pr_z + step_size) \rvert ; \ \Delta y = \lvert pl_y - ground \rvert$
$F_3 = \frac{1}{\Delta y} - penalty - n_{left_backward} - n_{tail} + n_{leg}$ $\Delta y = \lvert pl_y - ground \rvert$
$F_4 = \frac{1}{\Delta S_L} + n_{tail_lft} + n_{leg}$ $\Delta S_L = \sqrt{(cg_x - pl_x)^2 + (cg_z - pl_z)^2}$
$F_5 = \frac{1}{\Delta z} - n_{tail} + n_{leg}$ $\Delta z = \lvert pl_z - pr_z \rvert$

$$F_6 = \frac{k}{\Delta y} + \frac{1}{\Delta z} + \frac{1}{\sum_{i=0}^{6} |\Delta \theta_i|}$$

$$\Delta z = |pl_z - pr_z| \; ; \; \Delta y = |pr_y - ground|$$

$\Delta\theta$	คือ ผลต่างระหว่างมุมของแต่ละข้อต่อของหุ่นยนต์ใน“ท่าทางเริ่มต้น” กับท่าทางสุดท้าย
cg_x , cg_z	คือ ตำแหน่งของน้ำหนักหุ่นยนต์ในแนวแกน X และ Z
pl_x, pl_y , pl_z	คือ ตำแหน่งกึ่งกลางเท้าซ้ายของหุ่นยนต์ในแนวแกน X, Y, Z ตามลำดับ
pr_x, pr_y , pr_z	คือ ตำแหน่งกึ่งกลางเท้าขวาของหุ่นยนต์ในแนวแกน X, Y, Z ตามลำดับ
step_size	คือ ความยาวการก้าวเท้า (ถูกกำหนดไว้ ซึ่งในการทดลองใช้ 2.5 ซม.)
penalty	คือ การให้คะแนนลงโทษ เมื่อมีการเคลื่อนย้ายน้ำหนักออกนอกเท้าซ้าย
ground	คือ ตำแหน่งระดับพื้น (แกน Y)
k	คือ ค่าคงที่

<u>ตัวแปรชี้นำ</u>

n_{tail_rht}	คือ ระยะเวลา(สะสม)ที่มีการขยับหางไปทางขวา
n_{tail_lft}	คือ ระยะเวลา(สะสม)ที่มีการขยับหางไปทางซ้าย
n_{tail}	คือ ระยะเวลา(สะสม)ที่มีการขยับหางทั้งทางซ้ายและขวา
n_{leg}	คือ ระยะเวลา(สะสม)ที่มีการขยับข้อต่อขาต่าง ๆ
$n_{left_backward}$	คือ ระยะเวลา(สะสม)ที่มีการขยับขาซ้ายไปด้านหลัง

<u>ในขั้นตอนที่หนึ่ง</u> เป้าหมายย่อยคือการยืนบนเท้าขวา ซึ่งหมายถึงการที่หุ่นยนต์จะต้องย้ายน้ำหนักไปอยู่ที่เท้าขวา ดังนั้นตัวแปรที่ใช้จริงตามเป้าหมายย่อยของขั้นตอนนี้คือ ตัวแปรในพจน์แรก คือระยะระหว่างน้ำหนักหุ่นยนต์กับจุดกึ่งกลางเท้าขวา อย่างไรก็ตาม จะเห็นได้ว่าในสมการ F_1 นี้มีตัวแปรชี้นำคือ $+n_{tail_rht}$ และ $-n_{leg}$ ซึ่งผู้วิจัยมีความคิดเห็นว่าการย้ายน้ำหนักไปที่เท้าขวาต้องมีการขยับหางไปทางขวามาก ๆ และไม่จำเป็นต้องมีการขยับข้อต่อขาใด ๆ เลย ดังนั้นถ้ามีการขยับหางไปทางขวามาก ๆ ตัวแปร n_{tail_rht} ก็จะมีค่าสูงขึ้นตามไปด้วย ซึ่งจะทำให้ค่าความเหมาะสมสูงขึ้นด้วย ในทำนองเดียวกัน ถ้ามีการขยับขามาก ๆ ตัวแปร n_{leg} ก็จะมีค่าสูงขึ้น และทำให้ค่าความเหมาะสมต่ำลงด้วย (เนื่องจากตัวแปรตัวนี้ติดลบ)

<u>ในขั้นตอนที่สอง</u> หุ่นยนต์จะต้องยกเท้าซ้ายไปข้างหน้า ซึ่งจากสมการ F_2 จะเห็นได้ว่าตัวแปรที่ใช้จริงตามเป้าหมายย่อยคือ 2 พจน์แรก ตัวแปรในพจน์ที่หนึ่ง เป็นการวัดระยะห่างระหว่างจุดกึ่งกลางเท้าซ้ายและระยะก้าวไปข้างหน้า (*step_size*) จากเท้าขวาในแนวแกน Z ในที่นี้

ต้องมีการจำกัดความยาวของระยะการก้าวไปข้างหน้าด้วย เพื่อไม่ให้หุ่นยนต์ก้าวเท้ายาวจนเกินไป และจะทำให้ขั้นตอนที่สี่ซึ่งเป็นการย้ายน้ำหนักระหว่างสองเท้าเป็นไปได้ยากขึ้น (รายละเอียดตรงจุดนี้จะกล่าวถึงอีกทีในบทเรื่องผลการทดลอง) และตัวแปรในพจน์ที่สอง ซึ่งเป็นตัววัดความสูงของเท้าซ้ายจากพื้น โดยถ้าตัวแปรทั้งสองตัวนี้มีค่าน้อย (กล่าวคือ ก้าวเท้าซ้ายไปได้ตามระยะก้าวที่กำหนดและเท้าซ้ายอยู่ใกล้พื้น) ก็จะทำให้มีค่าความเหมาะสมมาก นอกจากนี้ ในสมการนี้ยังมีตัวแปรควบคุม 3 ตัวคือ $-n_{left_backward}$, $-n_{tail}$ และ $+n_{leg}$ ตัวแปรตัวแรกถูกกำหนดขึ้นเนื่องจากในการทดลองเบื้องต้นหุ่นยนต์ค่อนข้างมีพฤติกรรมการยกเท้าซ้ายไปข้างหลัง ผู้วิจัยจึงใส่ตัวแปรนี้เพิ่มเพื่อควบคุมไม่ให้หุ่นยนต์มีพฤติกรรมดังกล่าว ตัวแปรตัวที่สองใช้เพื่อควบคุมไม่ให้หุ่นยนต์ขยับหางมากนัก และตัวแปรที่สามใช้ในการกระตุ้นให้หุ่นยนต์มีการขยับข้อต่อต่าง ๆ ของขา

ในขั้นตอนที่สองนี้ จะขอกล่าวถึงรายละเอียดของขั้นตอนที่สองในเบื้องต้นซึ่งการทดลองแบบแบ่งท่าทางการเดินเป็น 4 ขั้นตอนย่อย เพื่อแสดงสาเหตุของการเปลี่ยนจาก 4 ขั้นตอนย่อยเป็น 6 ขั้นตอนย่อยดังนี้ เป้าหมายย่อยของขั้นตอนที่สอง***เดิม***นั้นคือการก้าวเท้าซ้าย ซึ่งหมายถึงการก้าวเท้าซ้ายไปข้างหน้าและวางเท้าซ้ายลงกับพื้น และมีสมการเกณฑ์การวัดค่าความเหมาะสมคือ

$$F_2 = \frac{k_1}{\Delta z} - k_2 \Delta y \; - penalty - n_{left_backward} - n_{tail} + n_{leg}$$

ในการทดลองเบื้องต้นนี้ มีการปรับน้ำหนักของค่าคงที่ k_1, k_2 ซึ่งจากการทดลองพบว่าไม่สามารถหาคำตอบที่ทำให้หุ่นยนต์แสดงพฤติกรรมตามเป้าหมายได้ เมื่อปรับ k_1 มาก จะได้คำตอบที่ทำให้หุ่นยนต์ยกเท้าซ้ายไปข้างหน้ามาก ๆ (ไม่แตะพื้น) และเมื่อปรับ k_2 ให้มาก ก็จะได้พฤติกรรมการวางเท้าซ้ายไว้กับพื้นเหมือนเดิม เพราะท่าต่อเนื่องจากขั้นตอนที่หนึ่งเป็นการยืนโดยน้ำหนักอยู่ที่เท้าขวา แต่เท้าซ้ายก็ยังสัมผัสพื้นอยู่ จากผลการทดลองเบื้องต้นนี้ ทำให้สรุปว่าการก้าวเท้าซ้ายไปวางสัมผัสพื้นด้านหน้าซึ่งเป้าหมายย่อยของขั้นตอนนี้ยังคงเป็นเป้าหมายที่ใหญ่เกินกว่าขั้นตอนวิธีพันธุกรรมจะหาคำตอบให้ได้ เนื่องจากพฤติกรรมดังกล่าวนี้เป็นการกระทำแบบมีลำดับเวลา ด้วยเหตุนี้เป้าหมายย่อยในขั้นตอนนี้จึงถูกแบ่งเป็น 2 ขั้นตอนย่อย และกลายเป็นการแบ่งท่าทางการเดินเป็น 6 ขั้นตอนย่อยนั่นเอง

ขั้นตอนที่สาม หุ่นยนต์วางเท้าซ้ายลงกับพื้น ตัวแปรที่ใช้ในการบรรลุตามเป้าหมายนี้คือพจน์ที่หนึ่งในสมการ $F3$ ซึ่งเป็นการวัดความสูงของเท้าซ้ายจากพื้น ตัวแปรสามพจน์หลังเป็นตัวแปรชี้นำซึ่งใช้ควบคุมพฤติกรรมของหุ่นยนต์ในทำนองเดียวกับขั้นตอนที่สอง นอกจากนี้จะเห็นได้ว่าในสมการนี้มีตัวแปร *penalty* ซึ่งใช้ในการให้คะแนนติดลบ เสมือนเป็นการลงโทษคำตอบนั้น เนื่องจากขั้นตอนนี้ใช้การวัดความสูงของเท้าซ้ายจากพื้นเป็นเกณฑ์ ซึ่งในบางคำตอบหุ่นยนต์จะใช้วิธีการเอียงตัว (ขยับหางไปทางซ้าย) เพื่อให้เท้าซ้ายแตะพื้น ซึ่งจะทำให้ฝ่าเท้าของหุ่นยนต์ไม่ได้สัมผัสพื้น

เต็มฝ่าเท้า และจะมีผลกับขั้นตอนถัดไป กล่าวคือ หุ่นยนต์มีโอกาสมากที่จะล้มหรือบิดตัวในระหว่างการย้ายน้ำหนักจากเท้าขวาไปเท้าซ้าย

ขั้นตอนที่สี่ หุ่นยนต์ต้องยืนด้วยเท้าซ้าย ซึ่งหมายถึงหุ่นยนต์จะต้องย้ายน้ำหนักจากเท้าขวาไปอยู่ที่เท้าซ้าย (ซึ่งในขณะนี้ก้าวมาข้างหน้าเท้าขวาแล้ว) ในขั้นตอนนี้จะสมการเกณฑ์การวัดค่าความเหมาะสมคล้ายกับสมการในขั้นตอนที่หนึ่ง ต่างกันตรงที่เปลี่ยนจากซ้ายเป็นขวาเท่านั้น กล่าวคือ ตัวแปรในพจน์ที่หนึ่ง วัดระยะห่างระหว่างน้ำหนักหุ่นยนต์กับจุดกึ่งกลางเท้าซ้าย และตัวแปรพจน์ที่สองซึ่งเป็นตัวแปรชี้นำและเป็นตัวกระตุ้นให้มีการขยับหางไปทางซ้ายมาก ๆ ตัวแปรตัวที่สามถูกใส่อยู่ในสมการด้วยผู้วิจัยคิดว่าขั้นตอนนี้น่าจะประกอบด้วยการขยับข้อต่อต่างๆที่ขาเพื่อประคองให้น้ำหนักให้อยู่ในพื้นที่รับน้ำหนักตลอดช่วงของการถ่ายน้ำหนัก

ขั้นตอนที่ห้า เป้าหมายคือการก้าวเท้าขวาให้มาอยู่ในระดับเดียวกับเท้าซ้าย ซึ่งคิดค่าความเหมาะสมจากระยะห่างระหว่างเท้าซ้ายกับเท้าขวาในแนวแกน Z ซึ่งเป็นตัวแปรในพจน์แรก สองพจน์หลังใช้ด้วยวัตถุประสงค์เหมือนในขั้นตอนที่ผ่าน ๆ มา คือควบคุมให้ไม่มีการขยับหางมากนักและกระตุ้นให้ขยับขามาก ๆ

ในขั้นตอนที่หก หุ่นยนต์จะต้องก้าวเท้าวางเท้าขวาลงกับพื้น ในระดับเดียวกับเท้าซ้ายตามแนวแกน Z ดังนั้นจึงควบคุมให้ความสูงจากพื้นของเท้าขวา (ตัวแปรพจน์แรก) และระยะห่างระหว่างสองเท้า (ตัวแปรพจน์ที่สอง) นอกเหนือไปจากการวางเท้าดังกล่าวที่ขั้นตอนที่หกจะต้องทำแล้ว ในตอนแรกของการทดลองนั้น ขั้นตอนที่หกซึ่งเป็นขั้นตอนสุดท้ายนี้ยังถูกระบุให้ปรับตำแหน่งมุมของข้อต่อทั้งเจ็ดให้มีขนาดเท่ากับมุมของแต่ละข้อต่อในท่าทางเริ่มต้นอีกด้วย เพื่อให้หุ่นยนต์สามารถเดินในก้าวต่อไปได้อย่างต่อเนื่อง ซึ่งภารกิจนี้ถูกระบุมาในตัวแปรพจน์ที่สาม

อย่างไรก็ตาม หลังจากทำการทดลองไประดับหนึ่ง พบว่าขั้นตอนวิธีพันธุกรรมไม่สามารถบรรลุเป้าหมายทั้งหมดของขั้นตอนนี้ได้ เนื่องจากมีปัญหาย่อยที่กำหนดให้นั้นใหญ่เกินไป ถ้าสังเกตจากสมการจะเห็นได้ว่ามีตัวแปรที่จะต้องขั้นตอนวิธีพันธุกรรมจะต้องแก้ถึง 8 ตัวด้วยกัน คือ สองตัวแปรในสองพจน์แรก และอีก 6 ตัวแปรในพจน์สุดท้าย ด้วยเหตุดังกล่าว พฤติกรรมการปรับมุมของข้อต่อไปสู่ท่าทางเริ่มต้นนั้นจึงไม่ถูกระบุให้เป็นเป้าหมายย่อยของขั้นตอนนี้ด้วย แต่ให้เป็นหน้าที่ของตัวโปรแกรมอ่าน-แปลคำตอบแทน

หลังจากทดลองเบื้องต้นสำเร็จแล้ว ขั้นตอนวิธีพันธุกรรมสามารถหาคำตอบที่ทำให้หุ่นยนต์เดินได้แล้ว ตัวแปรชี้นำจะค่อย ๆ ถูกตัดออกไปจากสมการเพื่อให้มีการใช้ตัวแปรชี้นำน้อยที่สุด ซึ่งหมายความว่าใช้ความรู้พื้นฐานของมนุษย์น้อยลงด้วย และหลังจากแปลงสมการให้อยู่ในรูปที่ง่ายขึ้น ก็จะได้สมการเกณฑ์การวัดค่าความเหมาะสมที่ใช้จริงดังตาราง 4.3 สมการเหล่านี้มีฟังก์ชันแบบหาค่าน้อย (minimized function) กล่าวคือ ค่าน้อยจะหมายถึงคำตอบมีความเหมาะสมมาก และสามารถสรุปเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนได้ดังตารางที่ 4.4

ตารางที่ 4.3: แสดงเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนของการทดลองในสภาพแวดล้อมจำลอง ที่ใช้จริง

$F_1 = \Delta S_R$	$\Delta S_R = \sqrt{(cg_x - pr_x)^2 + (cg_z - pr_z)^2}$
$F_2 = k\Delta z + \Delta y$	$\Delta z = \lvert pl_z - (pr_z + step_size) \rvert$ $\Delta y = \lvert pl_y - ground \rvert$
$F_3 = \Delta y + penalty$	$\Delta y = \lvert pl_y - ground \rvert$
$F_4 = \Delta S_L + penalty$	$\Delta S_L = \sqrt{(cg_x - pl_x)^2 + (cg_z - pl_z)^2}$
$F_5 = k\Delta z + \Delta y$	$\Delta z = \lvert pl_z - pr_z \rvert$ $\Delta y = \lvert pr_y - ground \rvert$
$F_6 = k\Delta y + \Delta z$	$\Delta z = \lvert pl_z - pr_z \rvert$ $\Delta y = \lvert pr_y - ground \rvert$

ตารางที่ 4.4: แสดงสรุปของเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนของการทดลองในสภาพแวดล้อมจำลอง

ขั้นตอนที่	เกณฑ์การวัดค่าความเหมาะสม
1	ระยะน้อยที่สุดระหว่างจุดศูนย์ถ่วง (center of gravity) ของหุ่นยนต์กับจุดกึ่งกลางของเท้าขวา
2	ความยาวการก้าวเท้าซ้ายที่มากที่สุด
3	ความสูงจากพื้นที่น้อยที่สุดของเท้าซ้าย
4	ระยะน้อยที่สุดระหว่างจุดศูนย์ถ่วงของหุ่นยนต์กับจุดกึ่งกลางของเท้าซ้าย
5	ความยาวการก้าวเท้าขวาให้มาชิดเท้าซ้ายให้มากที่สุด
6	ความสูงจากพื้นที่น้อยที่สุดของเท้าขวา

ในส่วนการทดลองกับหุ่นยนต์จริงนั้น การประเมินกระทำโดยมนุษย์เป็นผู้ตัดสิน โดยมีเกณฑ์การตัดสิน 2 ชนิดเช่นเดียวกับการทดลองในสภาพแวดล้อมจำลอง คือ

1) เกณฑ์การตัดสินที่ใช้กับทุกขั้นตอน จะคัดเลือกคำตอบที่ทำให้หุ่นยนต์ "ล้ม" หรือ "บิดตัว" ออกไป
2) เกณฑ์การตัดสินที่ใช้เฉพาะขั้นตอน ซึ่งสามารถแสดงได้ดังตาราง 4.5

ตารางที่ 4.5: แสดงวิธีการประเมินคำตอบแต่ละตัวของแต่ละขั้นตอนในการทดลองกับหุ่นยนต์จริง

ขั้นตอนที่	เกณฑ์การวัดค่าความเหมาะสม
1	น้ำหนักของหุ่นยนต์ตกอยู่ภายในเท้า*ขวา*หรือไม่
2	ก้าวเท้า*ซ้าย*มาข้างหน้าหรือไม่
3	วางเท้า*ซ้าย*ลงกับพื้นหรือไม่
4	น้ำหนักของหุ่นยนต์ตกอยู่ภายในเท้า*ซ้าย*หรือไม่
5	ก้าวเท้า*ขวา*มาชิดเท้าซ้ายหรือไม่
6	วางเท้า*ขวา*ลงกับพื้นหรือไม่

อนึ่ง คำตอบแต่ละตัวของแต่ละรุ่นที่จะนำมาประเมินค่าความเหมาะสมนั้นอาจมีลักษณะสายคำสั่งที่ซ้ำซ้อนกันอยู่ กล่าวคือ มีคำสั่งมอเตอร์ที่สั่งมอเตอร์ตัวเดิมติดกัน 2 โนด ซึ่งปรากฏการณ์นี้อาจเป็นผลมาจากการไขว้เปลี่ยนที่สลับบางส่วนของสายคำสั่งแล้วโนดที่สั่งมอเตอร์ตัวเดียวกันมาอยู่ติดกันพอดี หรือการกลายพันธุ์เปลี่ยนคำสั่งมอเตอร์แล้วบังเอิญไปสั่งมอเตอร์ตัวเดียวกับโนดก่อนหน้า ดังนั้นหลังจากการประเมินค่าความเหมาะสม โปรแกรมดำเนินขั้นตอนวิธีพันธุกรรมจะทำการลดรูป (simplify) ตัวโปรแกรมคำตอบเพื่อให้คำตอบกะทัดรัดขึ้น* โดยการลดรูปนี้ จะพิจารณาเฉพาะคำสั่งมอเตอร์ที่มีการสั่งมอเตอร์ตัวเดิม 2 คำสั่งติดกัน ซึ่งจะเป็นได้ 2 กรณี คือ

1) คำสั่งให้มอเตอร์ขยับไปทิศทางเดียวกัน จะรวมกันให้เป็นคำสั่งเดียว เช่น
   {3-:5, 6+:15, 6+:3, 1-:4} จะรวมกันกลายเป็น {3-:5, 6+:18, 1-:4}
2) คำสั่งให้มอเตอร์ขยับสวนทิศทางกัน จะหักลบให้เป็นคำสั่งเดียว เช่น
   {3-:5, 4+:5, 4-:3, 1-:4} จะหักลบกันกลายเป็น {3-:5, 4+:2, 1-:4}

* การลดรูปโปรแกรมคำตอบสามารถกระทำได้ทั้งก่อนหรือหลังการประเมินค่าความเหมาะสม ถ้ากระทำก่อนการประเมินค่าความเหมาะสม ก็จะมีผลทำให้เวลาที่ใช้ในการประเมินผลของคำตอบแต่ละตัวเร็วขึ้นอีกเล็กน้อย

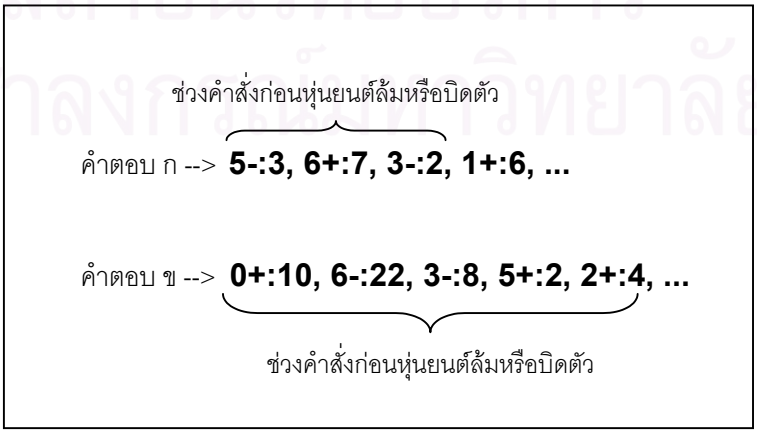
4.1.3 การคัดเลือก (selection)

การคัดเลือกคำตอบเพื่อให้กำเนิดกลุ่มคำตอบในรุ่นถัดไปนั้นใช้วิธีการคัดเลือกแบบ combined rank และคิดค่าความแตกต่าง (diversity value) ของคำตอบ 2 ตัวด้วยวิธี Hamming distance ซึ่งเป็นวิธีเทียบแบบบิตต่อบิต ในที่นี้ได้ประยุกต์ใช้เป็นการเทียบแบบโนดต่อโนด โดยมีขั้นตอนและรายละเอียดดังตารางที่ 4.6 (ดูรูป 4.13 , หน้า 24 ประกอบ)

ตารางที่ 4.6: แสดงขั้นตอนการคิดค่าความแตกต่าง

1. เปรียบเทียบความยาวของสายคำสั่ง โดย ค่าความแตกต่างเป็น “สองเท่าของ ผลต่างของจำนวนโนดของคำตอบทั้งสองตัว” โดยนับเฉพาะโนดก่อนหุ่นยนต์ล้มหรือบิดตัว เท่านั้น
2. เปรียบเทียบโนดต่อโนด (ด้วยจำนวนโนดเท่ากับคำตอบที่สั้นกว่า)
   - 2.1. ถ้าโนดนั้นสั่งมอเตอร์คนละตัวกัน หรือตัวเดียวกันแต่ทิศตรงข้ามกัน --> ค่าความแตกต่างจะเพิ่มขึ้นอีก 2 หน่วย โดยจะไม่พิจารณาถึง “จำนวนกระทำซ้ำ”
   - 2.2. แต่ถ้าโนดนั้นสั่งมอเตอร์ตัวเดียวกัน และ “จำนวนกระทำซ้ำ”ไม่เท่ากัน โดยต่างกันมากกว่า n --> ค่าความแตกต่างจะเพิ่มขึ้นอีก 1 หน่วย โดยมีเงื่อนไขคือ
     - ถ้าสั่งมอเตอร์ “หาง” (มอเตอร์หมายเลข 6) --> n จะต้องมากกว่า 15 จึงจะถือว่าต่างกัน
     - ถ้าสั่งมอเตอร์ตัวอื่น (มอเตอร์หมายเลข 0-5) --> n จะต้องมากกว่า 5 จึงจะถือว่าต่างกัน

ตัวอย่างจากรูป 4.17 กำหนดให้ช่วงของคำตอบที่ใช้ได้ก่อนหุ่นยนต์ล้มหรือบิดตัวเป็นดังรูป จะเห็นได้ว่าในขั้นตอนที่หนึ่ง คำตอบ ก. มีจำนวนโนดที่ใช้ได้เป็น 3 และคำตอบ ข. มีจำนวนโนดที่ใช้ได้เป็น 5 ดังนั้นในกรณีนี้ ค่าความแตกต่างมีค่าเป็น



รูปที่ 4.17 แสดงตัวอย่างประกอบการคิดค่าความแตกต่างของคำตอบ 2 คำตอบ

$2 \times |3-5| = 4$ หน่วย

ในขั้นตอนที่สองนั้น จะทำการเปรียบเทียบเพียง 3 โนดเท่านั้น เนื่องจากคำตอบ ก. มีความยาว 3 โนด ซึ่งมีรายละเอียดดังตารางที่ 4.7

ตารางที่ 4.7: แสดงตัวอย่างการคิดค่าความแตกต่างในขั้นตอนที่สอง

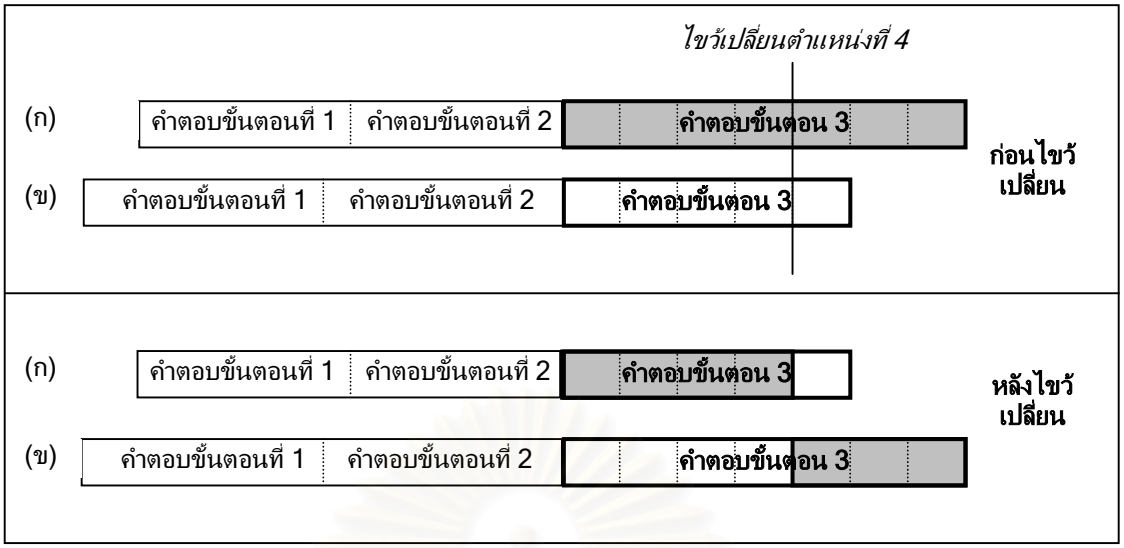
โนดที่	คำอธิบาย	ค่าความแตกต่าง	ค่าความแตกต่างสะสม
1	สั่งมอเตอร์คนละตัวกัน	2	6
2	สั่งมอเตอร์ตัวเดียวกันแต่ทิศตรงข้ามกัน	2	8
3	สั่งมอเตอร์ตัวเดียวกันและทิศทางเดียวกันและสั่งมอเตอร์ตัวอื่นที่ไม่ใช่หาง จำนวนกระทำซ้ำต่างกันเท่ากับ $\|2-8\| = 6$ ถือว่าต่าง	1	9

เพราะฉะนั้นในตัวอย่างนี้ คำตอบสองคำตอบมีค่าความแตกต่างเป็น 9 หน่วย

4.1.4 การไขว้เปลี่ยน (crossover) และการกลายพันธุ์ (mutation)

จากลักษณะของคำตอบที่ประกอบด้วยส่วนของ “ตัวระบุความยาวของคำตอบ” กับส่วนของ “สายลำดับของคำสั่งควบคุมการเดิน” (รูปที่ 4.14 , หน้า 25) นั้น ในการไขว้เปลี่ยนของคำตอบในแต่ละรุ่น จะทำการไขว้เปลี่ยนเฉพาะส่วนที่เป็นสายลำดับคำสั่งเท่านั้น โดยจะใช้วิธี **“การไขว้เปลี่ยนแบบหนึ่งจุด (single point crossover)”** คือสุ่มเลือกตำแหน่งโนดบนสายคำสั่งของคำตอบคู่ที่ต้องการจะไขว้เปลี่ยน โดยคำตอบทั้งคู่จะต้องไขว้เปลี่ยนในตำแหน่งเดียวกันของสายคำสั่ง และในการสุ่มเลือกตำแหน่งนี้จะสุ่มเลือกจากความยาวของคำตอบที่มีสายคำสั่งน้อยกว่า ส่วนของตัวระบุความยาวของคำตอบจะเปลี่ยนแปลงไปตามความยาวของสายลำดับคำสั่งหลังจากการไขว้เปลี่ยนแล้ว

ในการไขว้เปลี่ยนคำตอบของขั้นตอนการเดินที่ 2 ถึง 6 นั้นจะไขว้เปลี่ยนเฉพาะส่วนของคำตอบในขั้นตอนปัจจุบันเท่านั้น โดยยึดถือหลักเกณฑ์ว่าคำตอบของขั้นตอนปัจจุบันในส่วนที่อยู่ก่อนหน้าตำแหน่งการไขว้เปลี่ยนจะยังคง “ต่อ” อยู่กับคำตอบของขั้นตอนก่อน ๆ เหมือนเดิม การไขว้เปลี่ยนจะเป็นการสลับส่วนหลังจากตำแหน่งการไขว้เปลี่ยนเท่านั้น ดังแสดงตัวอย่างในรูปที่ 4.18 ซึ่งสมมติให้เป็นการทดลองในขั้นตอน 3 และมีทำการไขว้เปลี่ยนหลังโนดที่ 4 ของสายคำสั่งเป็นต้นไป หลังจากไขว้เปลี่ยนแล้ว ส่วนที่อยู่ก่อนตำแหน่งไขว้เปลี่ยน ทั้งที่เป็นส่วนของคำตอบในขั้นตอนหนึ่งและสอง และสายคำสั่งส่วนต้นของคำตอบ (ก) และ (ข) ยังคงอยู่เหมือนเดิม มีแต่ส่วนท้ายของคำ



รูปที่ 4.18: แสดงตัวอย่างวิธีการไขว้เปลี่ยนในขั้นตอนการเดินที่ 2 - 6

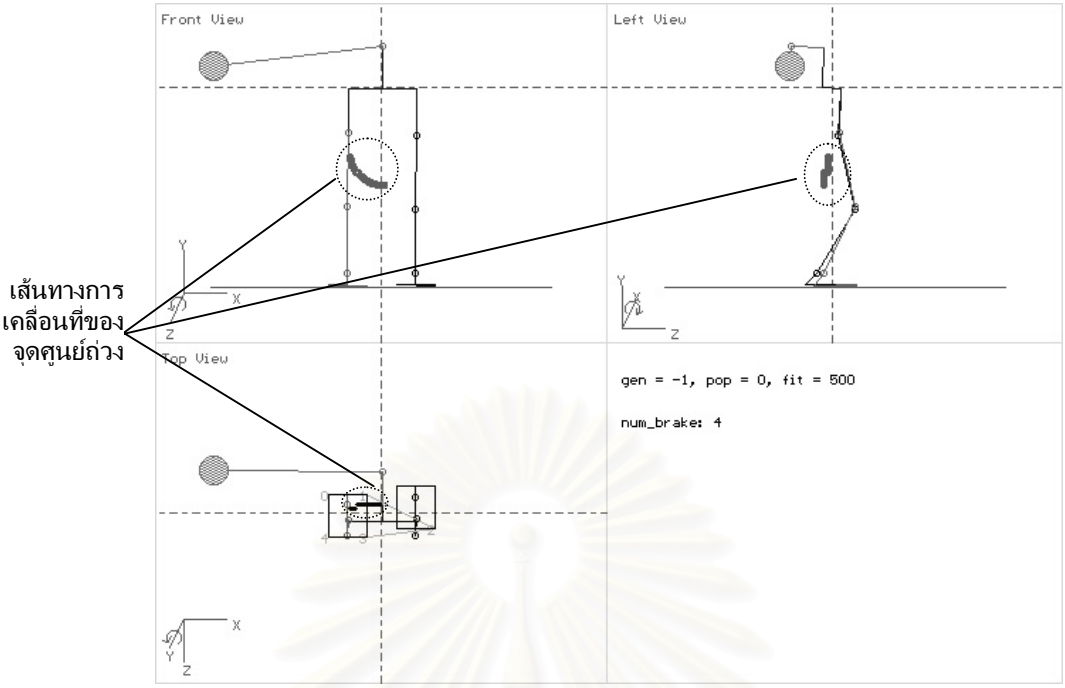
ตอบเท่านั้นที่สลับกัน โดยนัยของหลักเกณฑ์นี้หมายถึงจุดเชื่อมโยงระหว่างคำตอบขั้นตอนปัจจุบันกับขั้นตอนก่อนหน้าจะไม่ถูกเปลี่ยนแปลงใด ๆ

ในส่วนของการกลายพันธุ์จะมีหลักเกณฑ์คล้ายกับการไขว้เปลี่ยน กล่าวคือ การกลายพันธุ์จะเกิดขึ้นเฉพาะกับส่วนของสายลำดับคำสั่งคำตอบในขั้นตอนปัจจุบันเท่านั้น ไม่มีการดำเนินการกลายพันธุ์กับส่วนที่ระบุความยาวสายคำสั่ง และไม่ดำเนินการกลายพันธุ์กับคำตอบที่ได้จากขั้นตอนก่อน ๆ หน้า การกลายพันธุ์จะเกิดขึ้นในทุก ๆ โนดของสายคำสั่ง โดยใช้วิธีสุ่มจากค่าความน่าจะเป็นของการกลายพันธุ์ (ซึ่งกำหนดไว้) เพื่อดูว่าจะดำเนินการกลายพันธุ์ที่โนดนั้น ๆ หรือไม่

ขั้นตอนวิธีพันธุกรรมที่ใช้ในงานวิจัยนี้มีพารามิเตอร์ในการหาคำตอบของทุกขั้นตอน ซึ่งสามารถแสดงได้ดังตารางที่ 4.8

ตารางที่ 4.8: แสดงพารามิเตอร์ของขั้นตอนวิธีพันธุกรรม

ความยาวมากที่สุดของคำตอบในแต่ละขั้นตอนการเดิน	10 โนด
จำนวนประชากร	500 คำตอบ
จำนวนรุ่น	200 รุ่น
วิธีการไขว้เปลี่ยน	single point crossover
ความน่าจะเป็นของการไขว้เปลี่ยน	1.0
ความน่าจะเป็นของการกลายพันธุ์	0.001
วิธีการคัดเลือกคำตอบ	combined rank selection



รูปที่ 4.19: แสดงภาพตัวอย่างหน้าจอการทำงานของสภาพแวดล้อม

4.5. สภาพแวดล้อมจำลอง (simulation)

สภาพแวดล้อมจำลองถูกสร้างขึ้นเพื่อใช้ลดระยะเวลาในการทดลองให้น้อยลง การทดลองในสภาพแวดล้อมจำลองนั้น โปรแกรมดำเนินการขั้นตอนวิธีพันธุกรรม (ดูรูป 4.14 หน้า 25) จะพัฒนาโปรแกรมคำตอบตามหลักการขั้นตอนวิธีพันธุกรรม และในช่วงที่มีจะมีการประเมินค่าความเหมาะสมของคำตอบนั้น โปรแกรมดำเนินการขั้นตอนวิธีพันธุกรรมจะส่งคำตอบที่จะประเมินไปให้โปรแกรมจำลองสภาพแวดล้อมเพื่อพิจารณาดูว่าคำตอบนั้นจะทำให้หุ่นยนต์มีพฤติกรรมอย่างไร จากนั้นจะดึงข้อมูลต่าง ๆ ที่จำเป็นในการประเมินกลับมาใช้ในการประเมินค่าความเหมาะสมต่อไป

รูปที่ 4.19 แสดงภาพตัวอย่างหน้าจอการทำงานของสภาพแวดล้อมจำลอง การคำนวณค่าต่าง ๆ ในสภาพแวดล้อมจำลองจะเป็นแบบสามมิติ โดยมีการตั้งแกนดังรูปที่ 4.16 จากรูปที่ 4.19 จะเห็นได้ว่ามีการแปลผลของโปรแกรมคำตอบและแสดงพฤติกรรมหุ่นยนต์ใน 3 มุมมอง คือ มุมมองด้านหน้า, ด้านข้าง และด้านบน ภาพนี้เป็นภาพตัวอย่างพฤติกรรมของหุ่นยนต์ในขั้นตอนที่หนึ่ง ซึ่งหุ่นยนต์ขยับหางเพื่อให้จุดศูนย์ถ่วงเคลื่อนย้ายไปที่เท้าขวา ซึ่งในทั้งสามมุมมองจะแสดงเส้นทางการเคลื่อนที่ของจุดศูนย์ถ่วงไว้ด้วย (วงกลมเส้นประ)

ในสภาพแวดล้อมจำลองนั้น จะต้องมีการจำลองสถานการณ์บางอย่างที่จำเป็นต่อการพัฒนาให้หุ่นยนต์เดินได้ ซึ่งหมายถึง ในสภาพแวดล้อมจำลองจะต้องสามารถรับรู้ได้ถึงสถานการณ์ที่หุ่นยนต์มีการ “ล้ม” หรือ “บิดตัว” (ซึ่งการบิดตัวนี้จะทำให้หุ่นยนต์เริ่มเดินไม่ตรง และมิใช่จุดประสงค์ของงานวิจัยนี้) หรืออย่างน้อยสภาพแวดล้อมจำลองจะต้องรู้ถึงเงื่อนไขสาเหตุที่ทำให้เกิดเหตุการณ์ดังกล่าว นอกจากนี้สภาพแวดล้อมจำลองยังต้องสามารถรับรู้ได้ถึงตำแหน่งต่าง ๆ (ในสามมิติ) ของจุดศูนย์ถ่วง และข้อต่อทั้งหมดหุ่นยนต์อีกด้วย ในที่นี้จะอธิบายเฉพาะหลักการในการจำลองสถานการณ์การล้มและการบิดตัวของหุ่นยนต์เท่านั้น ซึ่งการตรวจสอบว่าหุ่นยนต์ล้มหรือยังรักษาสมดุลไว้ได้นั้นมีกระบวนการดังนี้

1. หุ่นยนต์จำลองจะถูกจัดให้อยู่ใน“ท่าทางเริ่มต้น”และยืนอยู่บนพื้น และในทุก ๆ ครั้งที่มีคำสั่งให้หุ่นยนต์ขยับข้อต่อใด ๆ จะทำการ *“ยกหุ่นยนต์ (ในแนวดิ่ง) ให้ลอยจากพื้น”* ก่อน แล้วขยับข้อต่อนั้น ๆ ตามคำสั่งที่เกิดขึ้น

2. ปรับค่าสถานะตำแหน่งของข้อต่อต่าง ๆ ที่สัมพัทธ์กับพื้นด้วยวิธีการ *“วางหุ่นยนต์ลงกับพื้น(ในแนวดิ่ง)”* จนกระทั่งจุดต่ำสุดของเท้า (ซึ่งมีอยู่ 8 จุด เนื่องจากหุ่นยนต์มีฝ่าเท้าเป็นรูปสี่เหลี่ยม) แตะพื้น ซึ่งจุดต่ำสุดของเท้าอาจมีได้มากกว่า 1 จุด ถ้ามีมากกว่า 3 จุดที่แตะพื้นพร้อมกัน ก็หมายความว่าเกิดพื้นที่รับน้ำหนักขึ้น

3. ตรวจสอบดูว่าจุดศูนย์ถ่วง (center of gravity) ของหุ่นยนต์ตกอยู่ภายในพื้นที่รับน้ำหนักแล้วหรือไม่ ถ้าอยู่ในพื้นที่รับน้ำหนักแล้ว แสดงว่าหุ่นยนต์สมดุลแล้ว การอัพเดตสถานะของหุ่นยนต์เนื่องจากคำสั่งหุ่นยนต์นี้ก็จะสิ้นสุดลง

4. ถ้ายังไม่อยู่ในพื้นที่รับน้ำหนัก จะตรวจสอบดูว่าหุ่นยนต์จะต้องถ่ายน้ำหนักไปทางใด โดยพิจารณาว่าจุดศูนย์ถ่วงอยู่ทางด้านใดของพื้นที่รับน้ำหนัก หุ่นยนต์ก็จะต้องถ่ายน้ำหนักไปทางนั้น

5. ตรวจสอบต่อไปว่าจุดศูนย์ถ่วงตกในพื้นที่รับน้ำหนักใหม่นี้หรือไม่ ถ้าไม่อยู่ในพื้นที่รับน้ำหนักใหม่นี้ก็ตรวจสอบต่อไปว่าจะต้องถ่ายน้ำหนักไปทางใด (เหมือนในข้อ 4) และทิศทางที่จะถ่ายไปนั้นยังมีจุดใด ๆ (ใน 8 จุด) ของฝ่าเท้าเหลืออยู่หรือไม่ ถ้าไม่มีแล้ว แสดงว่าหุ่นยนต์“ล้ม” แต่ถ้ามีก็จะทำซ้ำในลักษณะเดียวกับข้อ 3 - 5 ต่อไป

6. การ *“วางหุ่นยนต์ลงกับพื้น”* ในข้อ 2. นั้น อาจทำให้หุ่นยนต์มีตำแหน่งเปลี่ยนไปจากเดิมบ้างเล็กน้อย ดังนั้นต้องเคลื่อนย้ายหุ่นยนต์ให้กับมาอยู่ในตำแหน่งเดิม โดยการเคลื่อนย้ายนี้จะอ้างอิงจาก*“ตำแหน่งของจุดรับน้ำหนักเดิม”* ซึ่งเป็นข้อมูลก่อนหุ่นยนต์จะถูก*“ยกขึ้นจากพื้น”*ในข้อ 1

การบิดตัวของหุ่นยนต์นั้นจะใช้วิธีตรวจสอบจากเงื่อนไขสาเหตุที่ทำให้หุ่นยนต์บิด ตัว ในการตรวจสอบจาก “เงื่อนไขการบิดตัว” นั้นกำหนดขึ้นจากการทดลองให้หุ่นยนต์จริงเคลื่อนไหวใน

รูปแบบต่าง ๆ แล้วนำมาสรุปว่าเงื่อนไขใดบ้างที่ทำให้หุ่นยนต์เริ่มเกิดการบิดตัว ซึ่งสรุปได้ว่า **เงื่อนไขที่ทำให้หุ่นยนต์บิดตัวคือ หุ่นยนต์ยืนโดยมีสองเท้าสัมผัสพื้น (ไม่ว่าจะลงน้ำหนักทั้งสองเท้าหรือไม่) และมีการขยับข้อต่อหัวเข่าหรือสะโพกของขาทั้งสองข้างที่ทำให้มุมของขาทั้งสองข้างแตกต่างไปจากเดิม 2.68^{o} (หรือเท่ากับการขยับ 2 ครั้ง)** ในการกำหนดเงื่อนไขดังกล่าวนี้ เป็นการกำหนดในลักษณะการเผื่อความผิดพลาด กล่าวคือการขยับใด ๆ ของหุ่นยนต์ที่ตรงตามเงื่อนไขนี้อาจมีทั้งทำให้หุ่นยนต์บิดหรือไม่บิดตัวก็ได้ แต่การขยับใด ๆ ที่ไม่ตรงตามเงื่อนไขนี้จะไม่ทำให้หุ่นยนต์บิดตัวแน่นอน

อย่างไรก็ตาม เงื่อนไขดังกล่าวที่ได้มานั้น สรุปมาจากข้อมูลสถานะต่าง ๆ ของหุ่นยนต์จริง แต่เงื่อนไขนี้ถูกนำมาใช้ในสภาพแวดล้อมจำลอง ซึ่งยังคงมีความแตกต่างไปจากสภาพแวดล้อมจริงอยู่ ดังนั้นจึงยังไม่สามารถกล่าวได้อย่างแน่ชัดว่าโปรแกรมคำตอบที่เหลือจากการถูกกันออกไปเนื่องจากมีการขยับหุ่นยนต์ตามเงื่อนไขเหล่านี้ จะไม่มีทางทำให้เกิดการบิดตัวขึ้นในหุ่นยนต์จริงได้ แต่พอจะกล่าวได้เพียงว่าพฤติกรรมการบิดตัวที่เกิดขึ้นกับหุ่นยนต์จริงนั้นน่าจะเหลือน้อยเต็มที

ข้อจำกัดของสภาพแวดล้อมจำลองที่ใช้ในงานวิจัยนี้ ได้แก่

1. ไม่สามารถระบุได้ว่าสภาพแวดล้อมจำลองนั้นมีความเหมือนหรือแตกต่างจากสภาพแวดล้อมจริงมากน้อยเพียงใด สิ่งที่พอจะใช้บ่งความเหมือนหรือแตกต่างกันดังกล่าวนั้นมีเพียงโครงสร้างสัดส่วนของหุ่นยนต์จำลองกับหุ่นยนต์จริงเหมือนหรือแตกต่างกันเพียงใด และอีกสิ่งหนึ่งที่พอจะบ่งได้คือสัดส่วนจำนวนคำตอบที่เลือกมาจากสภาพแวดล้อมจำลองและนำมาทดลองกับหุ่นยนต์จริงแล้วประสบความสำเร็จ กล่าวคือคำตอบที่เลือกมาทดลองนั้น ถ้าประสบความสำเร็จทั้งหมด 100% ก็พอจะกล่าวได้ว่าสภาพแวดล้อมจำลองค่อนข้างเหมือนกับสภาพแวดล้อมจริงมากกว่า เหตุการณ์ที่คำตอบที่เลือกมาแล้วสำเร็จเพียง 20% อย่างไรก็ตาม วิธีบ่งชี้ความเหมือนต่างของสภาพแวดล้อมจำลองทั้งสองวิธีดังกล่าว ก็ยังไม่มีความชัดเจนเพียงพอที่จะใช้อ้างอิงได้

2. เนื่องจากวิธีการจำลองสภาพแวดล้อม โดยเฉพาะการตรวจสอบว่าหุ่นยนต์มีการล้มหรือไม่นั้นเป็นวิธีที่ผู้ทำวิจัยคิดขึ้นเอง มิได้อ้างอิงจากของใคร และแม้ว่าผู้ทำวิจัยจะพยายามแก้ไขตรวจสอบความถูกต้องของการทำงานของโปรแกรมอยู่หลายช่วงด้วยกัน แต่ก็ไม่สามารถกล่าวได้ว่าโปรแกรมจำลองสภาพแวดล้อมจะทำงานได้โดยไม่มีข้อผิดพลาดใด ๆ เกิดขึ้นเลย อย่างไรก็ตาม จากการทำการทดลองในสภาพแวดล้อมจำลองควบคู่กับการทดลองกับหุ่นยนต์จริง ทำให้สามารถกล่าวได้ว่าโปรแกรมจำลองสภาพแวดล้อมที่ใช้ในงานวิจัยนี้ สามารถช่วยให้การสังเคราะห์คำตอบของขั้นตอนวิธีพันธุกรรมเป็นไปได้ด้วยดีทีเดียว

บทที่ 5

ผลการทดลอง

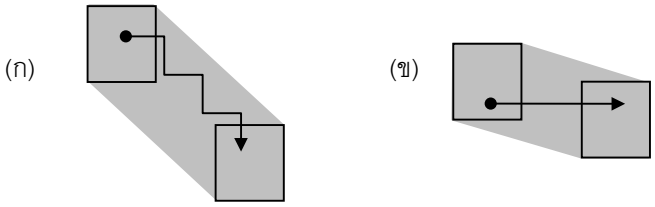
5.1. การทดลองในช่วงต่าง ๆ

5.1.1 การทดลองในระยะที่หนึ่ง

ในระหว่างการทดลองนั้น ได้มีการปรับตัวแปรที่เกี่ยวข้องกับการทดลองต่าง ๆ ไปเรื่อย ๆ เพื่อเป็นการทดสอบความเป็นไปได้ของแนวคิดวิธีการที่ตั้งต้นไว้ และหาทางปรับแต่งส่วนต่าง ๆ การทดลองเพื่อให้ได้คำตอบที่เหมาะสม ซึ่งมีรายละเอียดดังนี้

ในช่วงเริ่มแรกนั้น ทดลองความเป็นไปได้ของแนวคิดการแบ่งท่าทางการเดินเป็น 4 ขั้นตอนย่อย โดยทำการทดลองพัฒนาหาคำตอบของการเดินหนึ่งก้าวในสภาพแวดล้อมจำลองก่อน ซึ่งหมายถึงทำการทดลองในสภาพแวดล้อมจำลองในทุกขั้นตอนติดต่อกันโดยยังไม่ได้เกี่ยวข้องใด ๆ กับการทดลองกับหุ่นยนต์จริงเลย การทดลองในเบื้องต้นนี้ใช้ทั้งเกณฑ์การวัดค่าความเหมาะสมทั่ว-ไป และเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนซึ่งยังเป็นสมการในลักษณะที่มีการชี้นำจากมนุษย์ผสมอยู่ด้วย ดังที่ได้แสดงไว้ในตารางที่ 4.2 (หน้า 28) การทดลองในขั้นตอนย่อยที่หนึ่ง ซึ่งมีเป้าหมายคือการย้ายจุดศูนย์ถ่วงจากท่าทางเริ่มต้นให้ไปอยู่ที่เท้าขวานั้นผ่านไปได้ด้วยดี

แต่ในขั้นตอนที่สองซึ่งมีเป้าหมายย่อยคือการก้าวเท้าซ้ายไปวางไว้ที่ด้านหน้านั้น เกิดปัญหาขึ้นว่าขั้นตอนวิธีพันธุกรรมไม่สามารถหาคำตอบได้ตามเป้าหมายย่อยที่กำหนดขึ้นได้ ได้พยายามแก้ปัญหาโดยการปรับแต่งน้ำหนักระหว่างค่าคงที่ของตัวแปร Z (ซึ่งหมายถึงระยะการก้าวไปข้างหน้า) และตัวแปร Y (ซึ่งหมายถึงความสูงจากพื้นของเท้าซ้าย) รวมทั้งมีการเปลี่ยนสมการที่ใช้ไปในลักษณะต่าง ๆ (ดูตารางที่ 4.2 , หน้า 28 ประกอบ) แต่ก็ยังคงไม่สามารถได้คำตอบตามเป้าหมายได้ ทั้งนี้จึงได้สรุปเหตุผลไว้ว่าเนื่องจากพฤติกรรมของขั้นตอนนี้ยังมีความซับซ้อนอยู่ดังที่ได้กล่าวไว้ในบทที่แล้ว ดังนั้นจึงได้ทำการแบ่งขั้นตอนที่สองให้เป็นอีกสองขั้นตอนย่อย คือ การยกเท้าซ้ายขึ้น และการวางเท้าซ้ายลง (ดังรูปที่ 4.7, หน้า 18) ในทำนองเดียวกันในขั้นตอนที่สี่ (ตามรูป 4.6, หน้า 17) ซึ่งมีเป้าหมายให้หุ่นยนต์ก้าวเท้าขวาไปวางไว้ในระดับเดียวกับเท้าซ้าย ก็เป็นขั้นตอนที่จะต้องแบ่งเป็นอีกสองขั้นตอนย่อยเช่นกัน



รูปที่ 5.1: แสดงลักษณะการถ่ายน้ำหนักในพื้นที่รับน้ำหนักของระยะการก้าวสองแบบ
(ก) ระยะก้าวเท้ายาว (ข) ระยะก้าวเท้าสั้น

ภายหลังการแบ่งขั้นตอนที่สองเป็นขั้นตอนที่สองและสามแล้ว ก็สามารถทำการทดลองได้เป็นปกติ ขั้นตอนวิธีพันธุกรรมสามารถหาคำตอบได้ตามเป้าหมายย่อยของทั้งสองขั้นตอน อย่างไรก็ตาม เมื่อถึงการทดลองขั้นตอนที่สี่ ปรากฏว่าเป้าหมายการย้ายจุดศูนย์ถ่วงจากเท้าขวาไปเท้าซ้ายนั้น ทำได้ยาก เนื่องจากตำแหน่งการวางเท้าซ้ายนั้นอยู่ล้ำหน้าเท้าขวาไปค่อนข้างมาก ซึ่งถ้าพิจารณาตามรูปที่ 5.1 (ก) นั้น การถ่ายน้ำหนักจะทำได้นั้นต้องมีการขยับหางสลับกับการขยับข้อต่ออื่น ๆ อยู่หลายครั้งจึงจะถ่ายน้ำหนักได้สำเร็จ แต่พฤติกรรมการถ่ายโอนน้ำหนักที่เกิดขึ้นนั้น หุ่นยนต์พยายามจะใช้วิธีการที่ง่ายกว่า คือใช้หางในการถ่ายน้ำหนักรวดเดียวติดต่อกัน และมีการขยับข้อต่ออื่น ๆ บ้างเล็กน้อย ทางแก้ที่เป็นไปได้มีอยู่หลายทาง เช่น การเพิ่มความยาวของสายคำสั่ง ซึ่งตอนแรกกำหนดไว้ว่าให้มีความยาว 10 โนดในทุก ๆ ขั้นตอนย่อย, หรืออาจเพิ่มจำนวนประชากรหรือจำนวนรุ่นในการวิวัฒนาการของขั้นตอนวิธีพันธุกรรม เป็นต้น อย่างไรก็ตาม หลังจากทดลองเลือกใช้วิธีต่าง ๆ ดังกล่าวค่อนข้างให้ผลไม่แน่ชัดว่าต้องใช้ตัวเลขเท่าไรในการเพิ่มความยาว การเพิ่มจำนวนประชากร หรือเพิ่มจำนวนรุ่นของการวิวัฒนาการ เพราะฉะนั้นในการแก้ไขปัญหาดังกล่าว จึงใช้วิธีกำหนดความยาวของการก้าวเท้าไว้ไม่ให้ระยะก้าวเท้ายาวเกินไป โดยเลือกให้ระยะการก้าวเท้าเป็นครึ่งหนึ่งของความยาวฝ่าเท้า เพื่อให้หุ่นยนต์สามารถถ่ายน้ำหนักได้ง่ายขึ้น (รูปที่ 5.2 ข) โดยการกำหนดความยาวการก้าวนี้ได้ระบุไว้ในเกณฑ์การวัดค่าความเหมาะสมของการทดลองในขั้นตอนที่สอง

การทดลองในขั้นตอนที่ห้า ผ่านไปโดยไม่มีปัญหาใด ๆ แต่ในการทดลองขั้นที่หกนั้น เป้าหมายที่ตั้งไว้ในตอนแรกคือ การวางเท้าขวาลงกับพื้น พร้อมทั้งท่าทางตอนสุดท้ายของหุ่นยนต์ให้เหมือนกับท่าทางเริ่มต้น ซึ่งหุ่นยนต์ไม่สามารถบรรลุวัตถุประสงค์ประการหลังเนื่องจากมีตัวแปรที่ต้องแก้มากเกินไป ดังที่ได้อธิบายไว้ในบทที่สี่แล้ว ดังนั้นจึงแก้ปัญหาโดยลดวัตถุประสงค์ของขั้นตอนที่หกลงเหลือเพียงการวางเท้าขวาลงกับพื้นเท่านั้น และเนื่องจากท่าทางตอนสุดท้ายมีความแตกต่างจากท่าทางเริ่มต้นไม่มากนัก หน้าที่ในการปรับแต่งท่าทางหุ่นยนต์ให้เหมือนท่าทางเริ่มต้นจึงย้ายไปอยู่ที่โปรแกรมอ่าน-แปลคำตอบแทน และในที่สุดสามารถหาคำตอบที่ทำให้หุ่นยนต์เดินได้หนึ่งก้าว ซึ่งทำให้สามารถยืนยันความเป็นไปได้ของแนวคิดวิธีการแบ่งท่าทางการเดินออกเป็นขั้น-

ตอนย่อยว่าสามารถพัฒนาหาโปรแกรมคำตอบที่ทำให้หุ่นยนต์เดินได้แม้ว่าเป็นการทดลองในสภาพแวดล้อมจำลองเท่านั้นก็ตามที

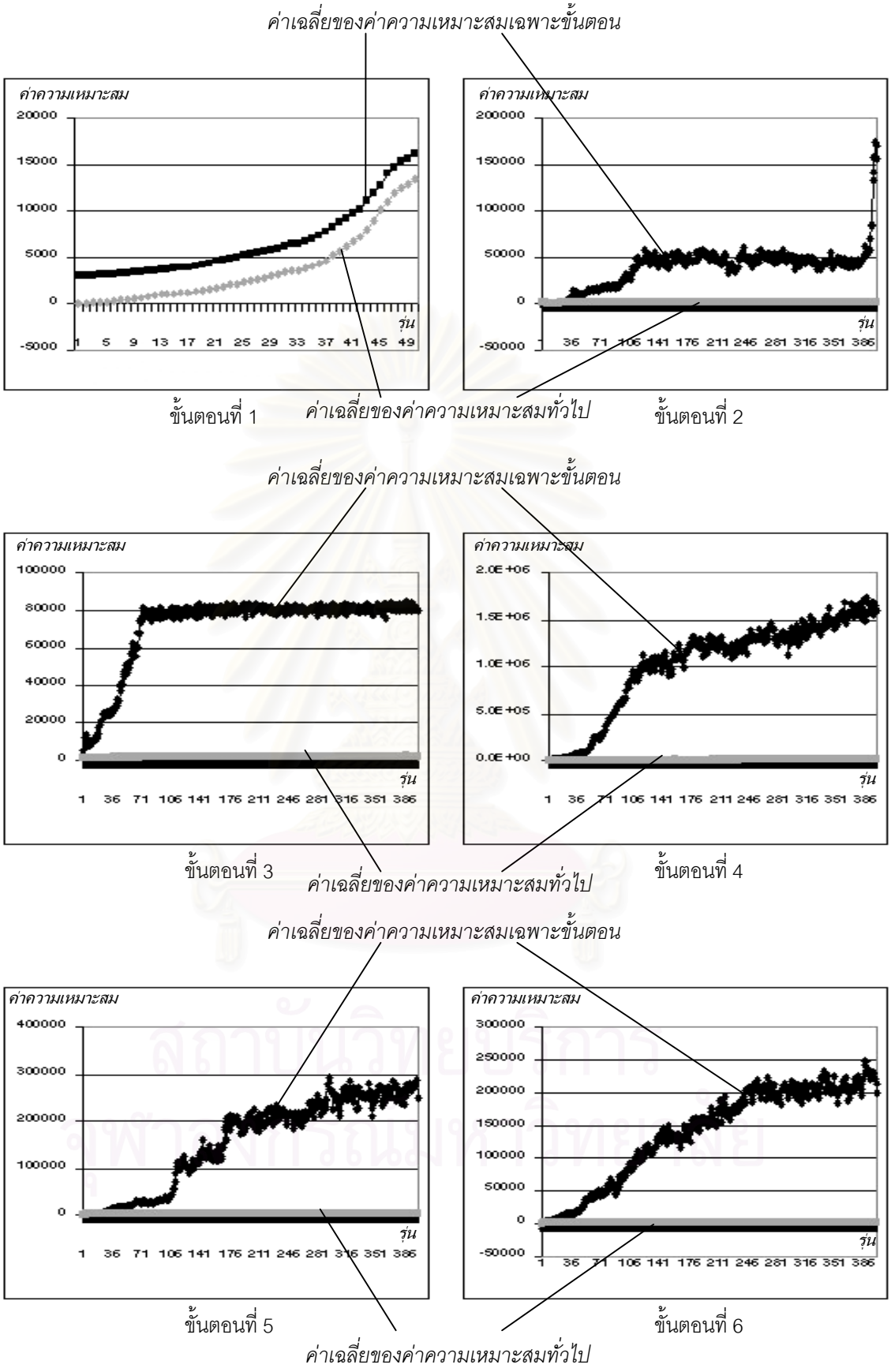
5.1.2 การทดลองในระยะที่สอง

หลังจากหุ่นยนต์สามารถเดินได้ตามเป้าหมายที่วางไว้ในสภาพแวดล้อมจำลองแล้ว การทดลองในระยะถัดมา เป็นการพยายามหาคำตอบที่ควบคุมหุ่นยนต์**จริง**ให้สามารถเดินได้ในสภาพแวดล้อม**จริง** โดยเริ่มทำการทดลองใหม่ตั้งแต่ขั้นตอนที่หนึ่งถึงหก ซึ่งมีการทดลองในสภาพแวดล้อมจำลองควบคู่ไปกับการทดลองกับหุ่นยนต์จริง (ดังแสดงไว้ในรูปที่ 4.9, หน้า 20)

ปัญหาเริ่มแรกของการทดลองในช่วงที่สองนี้คือ โปรแกรมคำตอบในแต่ละขั้นตอนที่สังเคราะห์ขึ้นได้ด้วยขั้นตอนวิธีพันธุกรรมในสภาพแวดล้อมจำลองนั้นมีรูปร่างลักษณะของสายคำสั่งเหมือนกันหมด สามารถกล่าวได้ว่า การทดลองในสภาพแวดล้อมจำลองนั้นสังเคราะห์คำตอบออกได้เพียง**คำตอบเดียว** ซึ่งเมื่อนำมาทดลองกับหุ่นยนต์จริงแล้ว ปรากฏว่าหุ่นยนต์**ล้ม** คำตอบที่ได้จากการทดลองในสภาพแวดล้อมจำลองไม่สามารถใช้ได้จริงกับหุ่นยนต์จริง เนื่องจากคำตอบที่ได้มีหน้าตาเหมือนกันหมด ทางแก้สำหรับปัญหานี้คือ ต้องเพิ่มความหลากหลายของคำตอบที่สังเคราะห์ได้ ซึ่งในการเพิ่มความหลากหลายนี้ต้องแก้ที่วิธีการคัดเลือกคำตอบในแต่ละรุ่นของขั้นตอนวิธีพันธุกรรม

วิธีการคัดเลือกคำตอบที่ใช้ในตอนแรกคือ “การคัดเลือกตามสัดส่วนของค่าความเหมาะสม” ของคำตอบแต่ละตัว ซึ่งวิธีการนี้จะทำให้คำตอบที่มีค่าความเหมาะสมสูงนั้นมีโอกาสอยู่รอดและเป็นต้นแบบของการผลิตคำตอบในรุ่นถัด ๆ ไปในสัดส่วนที่สูงได้ ยิ่งมีค่าความเหมาะสมมากก็ยิ่งมีสัดส่วนในการเป็นต้นแบบของประชากรคำตอบในรุ่นถัดไปได้มาก และทำให้เกิดปรากฏการณ์ชี้นำแนวทางการพัฒนาคำตอบของระบบให้เป็นไปในแนวเดียวหรือรูปแบบเดียวได้ และเป็นเหตุผลที่ทำให้คำตอบที่สังเคราะห์ได้ในงานนี้มีหน้าตาเหมือนกันทั้งหมด

ดังนั้นในงานนี้จึงได้เปลี่ยนวิธีการคัดเลือกคำตอบเป็นการคัดเลือกแบบ combined rank ซึ่งเป็นวิธีการที่มีการจัดอันดับของผลรวมของค่าความเหมาะสมและค่าความแตกต่างกันของประชากรคำตอบแต่ละตัวในการพิจารณาคัดเลือกโปรแกรมคำตอบไปใช้ในรุ่นถัดไป ซึ่งทำให้คำตอบที่สังเคราะห์ขึ้นได้นั้น มีความหลากหลายของหน้าตาหรือรูปลักษณ์มากขึ้น ซึ่งมีทั้งคำตอบที่สามารถใช้ได้และใช้ไม่ได้กับหุ่นยนต์จริงรวมกันอยู่ ดังนั้นด้วยวิธีการทดลองในสภาพแวดล้อมจำลองกับการทดลองกับหุ่นยนต์จริงควบคู่กันไปในแต่ละขั้นตอน ภายหลังการทดลองในขั้นตอนที่หกแล้ว ก็จะได้คำตอบที่ทำให้หุ่นยนต์จริงสามารถเดินได้



รูปที่ 5.2: แสดงกราฟค่าเฉลี่ยของค่าความเหมาะสมทั่วไปและค่าความเหมาะสมเฉพาะของทั้ง 6 ขั้นตอน

5.1.3 การทดลองในระยะที่สาม

หลังจากได้คำตอบที่สามารถทำให้หุ่นยนต์จริงเดินได้แล้ว จึงได้นำค่าเฉลี่ยของค่าความเหมาะสมทั่วไปและค่าเฉลี่ยของค่าความเหมาะสมเฉพาะขั้นตอนของประชากรคำตอบในแต่ละรุ่นมาวาดเป็นกราฟ เพื่อดูแนวโน้มของการพัฒนาคำตอบในแต่ละรุ่นของทั้ง 6 ขั้นตอน โดยแกนนอนแสดงถึงจำนวนรุ่นที่เพิ่มขึ้น และแกนตั้งเป็นค่าความเหมาะสม ดังแสดงในรูปที่ 5.2 โดยมีเส้นสีเทาคือค่าเฉลี่ยของค่าความเหมาะสมทั่วไปในแต่ละรุ่น และเส้นสีดำคือค่าเฉลี่ยของค่าความเหมาะสมเฉพาะขั้นในแต่ละรุ่น ซึ่งจะเห็นได้ว่า นอกจากขั้นตอนที่หนึ่งแล้ว ค่าความเหมาะสมทั่วไปของทุก ๆ ขั้นตอนนั้นมีค่าน้อยมากเมื่อเทียบกับค่าความเหมาะสมเฉพาะขั้นตอน ทำให้เกิดข้อสงสัยว่า *"เกณฑ์การวัดค่าความเหมาะสมทั่วไปนี้ไม่น่าจะมีอิทธิพลใด ๆ กับการวิวัฒนาการคำตอบในแต่ละรุ่นมากนัก"*

ดังนั้นการทดลองในช่วงถัดมาจึงเป็นการทดลองซ้ำในแบบที่ผ่านมา แต่ไม่ใช้เกณฑ์การวัดค่าความเหมาะสมทั่วไป กล่าวคือการทดลองในสภาพแวดล้อมจำลองนั้น ในกระบวนการขั้นตอนวิธีพันธุกรรมของแต่ละขั้นตอนย่อย จะไม่ใช้เกณฑ์การวัดค่าความเหมาะสมทั่วไป ใช้เพียงเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนเท่านั้น และในส่วนการทดลองกับหุ่นยนต์จริงที่กระทำควบคู่ไปกับการทดลองในสภาพแวดล้อมจำลองนั้นยังคงไว้ดังเดิม

หลังจากทำการทดลองในแต่ละขั้นตอนตามวิธีการดังกล่าวแล้ว ปรากฏว่ายังคงสามารถหาคำตอบให้กับการเดินในแต่ละขั้นตอนได้ และภายหลังการทดลองกับหุ่นยนต์จริงในขั้นตอนที่หกแล้ว ก็ยังคงสามารถได้โปรแกรมคำตอบที่ควบคุมให้หุ่นยนต์จริงสามารถเดินได้โดยไม่มีการล้มหรือบิดตัวใด ๆ ซึ่งทำให้สามารถสรุปได้ว่าสมมติฐานของการทดลองในช่วงนี้ที่ว่า *"เกณฑ์การวัดค่าความเหมาะสมทั่วไปไม่มีอิทธิพลใด ๆ กับการพัฒนาโปรแกรมคำตอบของแต่ละขั้นตอนย่อย"* นั้น**เป็นจริง**

ในชั้นต้นนั้น ปรากฏการณ์ของการที่ไม่ต้องใช้เกณฑ์การวัดค่าความเหมาะสมทั่วไปดังกล่าวนั้น ทำให้เกิดสมมติฐานต่อไปว่า *"อาจเป็นเพราะมีเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนเป็นตัวควบคุมพฤติกรรมที่เกิดขึ้นจากโปรแกรมคำตอบไม่ให้เกิดพฤติกรรมการล้มหรือบิดตัวอยู่แล้ว"* กล่าวคือ โปรแกรมคำตอบใดที่ทำให้เกิดการล้มหรือบิดตัวของหุ่นยนต์ขึ้นนั้นจะทำให้ได้รับคะแนนหรือค่าความเหมาะสมน้อยกว่าโปรแกรมคำตอบที่ไม่มีเหตุการณ์ดังกล่าว และโปรแกรมเหล่านี้อาจไม่ถูกเลือกไปใช้เป็นต้นแบบของโปรแกรมคำตอบในรุ่นถัด ๆ ไปก็เป็นได้

อย่างไรก็ตาม ภายหลังได้ทำการวิเคราะห์ดูค่าความเหมาะสมเฉพาะขั้นตอน (ดังตารางที่ 4.2, หน้า 28) แล้ว และตรวจสอบขั้นตอนการทำงานของระบบการดำเนินการทดลองในขั้นตอนต่าง ๆ อย่างละเอียดแล้ว ทำให้สามารถพิสูจน์สมมติฐานข้างต้นว่าเป็นจริงหรือไม่ ดังนี้

ในการทดลองในสภาพแวดล้อมจำลอง (ดูรูปที่ 4.9, หน้า 20 ประกอบ) โปรแกรมคำตอบจะถูกอ่านและทดลองให้แสดงพฤติกรรมกับหุ่นยนต์จำลองทีละโนด และก่อนที่โปรแกรมคำตอบจะถูกอ่านหมดทุกโนด ถ้าหุ่นยนต์เกิด "ล้ม" หรือ "บิดตัว" ที่โนดใด โปรแกรมจำลองสภาพแวดล้อมจะหยุดการทำงาน และทำ *"เครื่องหมายระบุการล้มหรือบิดตัว"* ไว้ที่โนดนั้น เครื่องหมายนี้ระบุไว้เพื่อให้รู้ว่าหุ่นยนต์สามารถขยับข้อต่อต่าง ๆ เป็นเวลาเท่าไร, มีการสั่งมอเตอร์กี่ครั้ง ก่อนที่จะล้มหรือบิดตัว จากนั้นจึงส่งข้อมูลต่าง ๆ ซึ่งได้แก่ ตำแหน่งข้อต่อทั้งเจ็ด, ตำแหน่งจุดศูนย์ถ่วงของหุ่นยนต์ กลับไปให้โปรแกรมดำเนินการขั้นตอนวิธีพันธุกรรมเพื่อประเมินค่าความเหมาะสมตามเกณฑ์การวัดที่กำหนดไว้ต่อไป ข้อมูลที่ส่งเหล่านี้เป็นข้อมูลสถานะของหุ่นยนต์ก่อนที่หุ่นยนต์จะมีการล้มหรือบิดตัว (เนื่องจากหุ่นยนต์ถูกหยุดสั่งการเมื่อมีการตรวจสอบได้ว่ามีการล้มหรือบิดตัวเกิดขึ้นแล้ว) ในทำนองเดียวกัน ถ้าไม่มีการล้มหรือบิดตัวเกิดขึ้น เมื่อโปรแกรมจำลองสภาพแวดล้อมอ่านโปรแกรมคำตอบหมดทุกโนดแล้ว ก็จะส่งข้อมูลสถานะต่าง ๆ ของที่เกิดขึ้นของหุ่นยนต์ไปใช้ประเมินค่าความเหมาะสมเช่นเดียวกัน ซึ่งจากคำอธิบายดังกล่าว แสดงให้เห็นว่า ไม่ว่าหุ่นยนต์จะไม่มีการล้มหรือบิดตัวหรือไม่ ข้อมูลที่ใช้ในการประเมินค่าความเหมาะสมของเป็นข้อมูลสถานะของหุ่นยนต์ในตอนที่หุ่นยนต์ไม่ล้มหรือบิดตัวใด ๆ ดังนั้นจึงสรุปได้ว่าสมมติฐานที่ว่า *"เกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนเป็นตัวควบคุมพฤติกรรมไม่ให้เกิดการล้มหรือบิดตัวของหุ่นยนต์อยู่แล้ว"* นั้น **ไม่เป็นจริง**

อย่างไรก็ตาม ได้มีการตรวจสอบขั้นตอนการดำเนินงานของระบบการทดลองต่อไปหลังจากโปรแกรมคำตอบแต่ละตัวถูกอ่านและประเมินค่าความเหมาะสมดังกล่าวแล้ว ก็ถูกพัฒนาไปเรื่อยๆ จนถึงรุ่นสุดท้าย จากนั้นโปรแกรมคำตอบในรุ่นสุดท้ายนี้จะถูกคัดเลือกมาบางส่วนเพื่อทำการทดลองกับหุ่นยนต์จริงนั้น และภายหลังการทดลองกับหุ่นยนต์จริงแล้ว ก็จะเหลือโปรแกรมคำตอบเพียงบางตัวที่ใช้ได้กับหุ่นยนต์จริง ซึ่งโปรแกรมคำตอบเหล่านี้จะถูกนำไปใช้เป็นท่าทางเริ่มต้นในขั้นตอนการเดินถัดไป ในการนำไปใช้นี้ โปรแกรมดำเนินการขั้นตอนวิธีพันธุกรรมจะ*ตัดตอนเฉพาะส่วนก่อนหน้าเครื่องหมายระบุการล้มหรือบิดตัวของโปรแกรมคำตอบ*ไปใช้เท่านั้น ส่วนที่อยู่หลังจากเครื่องหมายนี้จะไม่ถูกนำไปใช้

จุดนี้เองที่เป็นตัวควบคุมพฤติกรรมดังกล่าว เนื่องจากเครื่องหมายระบุการล้มหรือบิดตัวนี้ นอกจากจะมีความหมายดังกล่าวแล้ว ยังเป็นตัวระบุว่า *"พฤติกรรมใด ๆ ของหุ่นยนต์ที่เกิดจากโปรแกรมคำตอบในส่วนก่อนหน้าเครื่องหมายนี้จะไม่ทำให้หุ่นยนต์ล้มหรือบิดตัวได้"* ซึ่งทำให้สรุปได้ว่า สิ่งที่เป็นตัวควบคุมพฤติกรรมหุ่นยนต์ไม่ให้เกิดการล้มหรือบิดตัวคือตัวกระบวนการทำงานของการทดลองในสภาพแวดล้อมจำลองเอง ที่ทำการกลั่นกรองเอาเฉพาะโปรแกรมคำตอบ หรือถ้าพูดให้

ถูกคือเอาเฉพาะส่วนของโปรแกรมคำตอบ ที่ทำให้หุ่นยนต์แสดงพฤติกรรมต่าง ๆ โดยไม่มีการล้มหรือบิดตัวใด ๆ นั่นเอง

นอกจากนี้ ในการใช้เกณฑ์การวัดค่าความเหมาะสมทั่วไปนั้นยังอาจส่งผลเสียอีกอย่างหนึ่งด้วย กล่าวคือก่อให้เกิดพฤติกรรมการขยับขาไปมาซ้ำ ๆ โดยไม่ก่อให้เกิดผลเกี่ยวกับการเดินแต่อย่างใด ในการวิเคราะห์หาสาเหตุของพฤติกรรมการขยับขาไปมานั้น จะยกสมการของเกณฑ์การวัดค่าความเหมาะสมทั่วไปขึ้นมาพิจารณาอีกครั้ง ดังนี้

$$Fitness = (k_1F + k_2R) \times T$$

เนื่องมากจากว่าในสมการของเกณฑ์การวัดค่าความเหมาะสมทั่วไปมีการพิจารณาตัวแปรเวลา (T) ซึ่งหมายความว่าถ้าหุ่นยนต์สามารถเดินอยู่ได้โดยไม่ล้มและไม่บิดตัวนานเท่าใดก็จะมีค่าความเหมาะสมมากขึ้น ดังนั้นจึงปรากฏออกเป็นพฤติกรรม “ถ่วงเวลา” ของหุ่นยนต์นั่นเอง

5.1.4 การทดลองในระยะที่สี่

การทดลองในช่วงสุดท้ายนั้น เป็นการทดลองโดยการพยายามลดการชี้นำหรือลดตัวแปรชี้นำซึ่งอยู่ในสมการของเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นของแต่ละขั้นตอน (ดูตารางที่ 4.2, หน้า 28 ประกอบ) ให้เหลือน้อยที่สุดเท่าที่จะเป็นไปได้ จากการพยายามลดตัวแปรเหล่านี้ ทำให้จำเป็นต้องมีการปรับพารามิเตอร์ของขั้นตอนวิธีพันธุกรรมเล็กน้อย กล่าวคือ ต้องเพิ่มจำนวนประชากรและจำนวนรุ่นที่ใช้ในแต่ละขั้นตอนให้เพิ่มขึ้น และหลังจากทำการทดลองซ้ำอีก 3 ครั้ง ก็ทำให้สามารถสังเคราะห์โปรแกรมคำตอบได้ตามวัตถุประสงค์ของงานวิจัยนี้ และได้สมการเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนดังแสดงในตารางที่ 4.3 (หน้า 32)

5.1.5 สรุปการทดลองทั้ง 4 ระยะ

จากการทดลองทั้ง 4 ระยะ ทำให้สามารถสรุปการทดลองในแต่ละช่วงได้ดังนี้

1. ทำการทดลองในสภาพแวดล้อมจำลองเท่านั้น โดยใช้เกณฑ์การวัดค่าความเหมาะสมทั่วไป และใช้เกณฑ์การวัดค่าความเหมาะสมแบบมีตัวแปรชี้นำ วิธีการคัดเลือกคำตอบในแต่ละรุ่นนั้นใช้วิธีการคัดเลือกตามสัดส่วนของค่าความเหมาะสม ซึ่งจากการทดลองในระยะที่หนึ่งนี้ทำให้แน่ใจว่าวิธีการแบ่งขั้นตอนการเดินออกเป็นขั้นตอนย่อยนั้น สามารถช่วยทำให้ขั้นตอนวิธีพันธุกรรมสังเคราะห์คำตอบที่ทำให้หุ่นยนต์เดินได้

2. ผนวกการทดลองกับหุ่นยนต์จริงเข้ามาในระบบการทดลองด้วย และเปลี่ยนการคัดเลือกคำตอบเป็นวิธีการคัดเลือกแบบ combined rank เกณฑ์การวัดค่าความเหมาะสมทั้งสองแบบยังคงใช้เหมือนเดิม จากการทดลองในระยะนี้ทำให้แน่ใจว่า ความคิดที่นำการทดลองกับหุ่นยนต์จริง

มาผนวกใช้ควบคู่ไปกับการทดลองในสภาพแวดล้อมจำลองในรูปแบบดังกล่าวนั้น สามารถทำให้ได้คำตอบที่ใช้ได้กับหุ่นยนต์จริง อีกทั้งยังรู้เพิ่มขึ้นอีกว่าระบบการทดลองในรูปแบบดังกล่าวจะสำเร็จได้นั้น คำตอบที่ได้จากการทดลองในสภาพแวดล้อมจำลองแล้วจะนำมาทดลองกับหุ่นยนต์จริง ควรมีลักษณะคำตอบที่หลากหลาย เพื่อให้สามารถมีทางเลือกของคำตอบที่จะใช้กับหุ่นยนต์จริงได้มากขึ้น

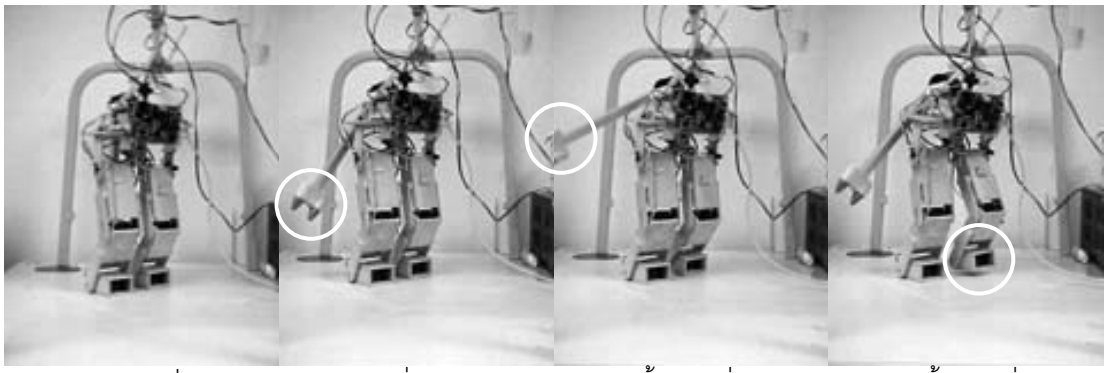
3. มีทั้งการทดลองในสภาพแวดล้อมจำลอง และการทดลองกับหุ่นยนต์จริงเหมือนเดิม แต่ในส่วนการทดลองในสภาพแวดล้อมจำลองนั้น ไม่ใช้เกณฑ์การวัดค่าความเหมาะสมทั่วไป ใช้เพียงเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอน (ซึ่งเป็นแบบมีตัวแปรชี้นำ) ผลจากการทดลองในระยะนี้ทำให้ทราบได้ว่า ตัวเกณฑ์การวัดค่าความเหมาะสมทั่วไปนั้นไม่มีความจำเป็นในการควบคุมพฤติกรรมหุ่นยนต์ให้ไม่มีการล้มหรือบิดตัว เนื่องจากระบบการทำงานของการทดลองเป็นตัวควบคุมพฤติกรรมดังกล่าวเอง ดังนั้นเกณฑ์การวัดค่าความเหมาะสมที่ใช้ในการทดลองเพื่อทำให้หุ่นยนต์สามารถเดินได้ตามระบบดังกล่าว จึงใช้เพียงเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอนเท่านั้นก็เพียงพอแล้ว

4. ทำการลดตัวแปรชี้นำในเกณฑ์การวัดค่าความเหมาะสมเฉพาะขั้นตอน และมีการปรับแต่งรูปสมการให้ดูง่ายขึ้น ซึ่งจากการทดลองระยะนี้ทำให้รู้ว่า แม้ไม่มีตัวแปรชี้นำแล้วโปรแกรมคำตอบที่ต้องการก็ยังคงสังเคราะห์ออกมาได้ เพียงแต่อาจจะต้องเพิ่มความพยายามในการค้นหาคำตอบมากขึ้น ด้วยการเพิ่มจำนวนประชากรหรือจำนวนรุ่นดังที่ได้กล่าวมาแล้ว

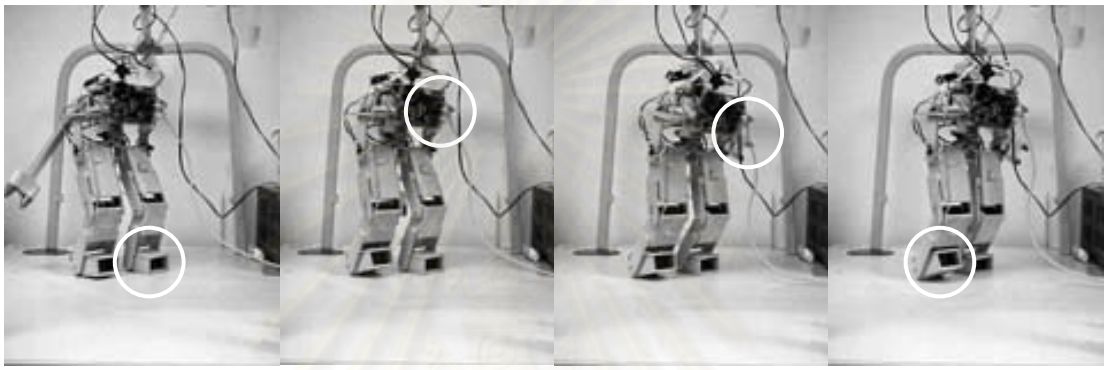
ในท้ายที่สุด สามารถสรุประบบของการทดลองที่ใช้ในงานวิจัยนี้ได้ว่า เป็นระบบที่มีทั้งการทดลองในสภาพแวดล้อมจำลองและการทดลองกับหุ่นยนต์จริงควบคู่กันไปในแต่ละขั้นตอน โดยการทดลองในสภาพแวดล้อมจำลองจะสังเคราะห์คำตอบโดยใช้ขั้นตอนวิธีพันธุกรรม และมีเกณฑ์การวัดค่าความเหมาะสมที่แตกต่างกันไปในแต่ละขั้นตอน โดยไม่มีการชี้นำจากมนุษย์ และจากผลการทดลองในระยะสุดท้าย หุ่นยนต์สามารถเดินได้อย่างต่อเนื่อง ดังจะอภิปรายผลการทดลองไว้ในหัวข้อถัด ๆ ไป

5.2. พฤติกรรมการเดิน

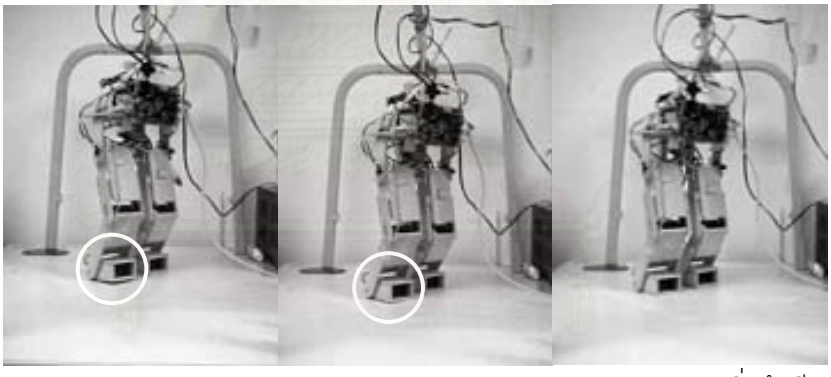
จากการทดลองโดยมีสภาวะเงื่อนไขต่าง ๆ ของระบบตามที่ได้สรุปไว้ในหัวข้อที่แล้วนั้น มีการทำซ้ำการทดลอง 3 ครั้ง ซึ่งก็สามารถได้ผลการทดลองคือ คำตอบที่ได้จากสภาพแวดล้อมจำลองของแต่ละขั้นตอนจะถูกเลือกมาเป็นจำนวน 20 ตัวจาก 500 ตัว จำนวนคำตอบที่ใช้ได้กับหุ่นยนต์จริง (หุ่นยนต์ไม่ล้มและบิดตัว) จะมีค่ามากในช่วงขั้นตอนแรก ๆ และมีจำนวนลดลงในช่วงขั้นตอนหลัง ๆ กล่าวคือ ในขั้นตอนแรก จำนวนคำตอบที่ใช้ได้กับหุ่นยนต์จริงเป็น 15-18 ตัว และลดลงจนในขั้นตอน



ก) ท่าทางเริ่มต้น ข) เริ่มขยับ ค) ขั้นตอนที่ 1 ง) ขั้นตอนที่ 2



จ) ขั้นตอนที่ 3 ฉ) ขยับหาง ช) ขั้นตอนที่ 4 ซ) ขั้นตอนที่ 5



ฌ) เตรียมวางเท้า ญ) ขั้นตอนที่ 6 ฎ) กลับสู่ท่าทางเริ่มต้นอีกครั้ง

รูปที่ 5.3: แสดงภาพตัวอย่างคำตอบการเดินของหุ่นยนต์จริง

สุดท้ายจำนวนคำตอบที่ใช้ได้กับหุ่นยนต์จริงจะเหลือเพียง 3-4 ตัว หุ่นยนต์สามารถเดินได้อย่างต่อเนื่องมากกว่า 15 ก้าวโดยไม่ล้ม ใช้เวลาในการเดินแต่ละก้าวเป็น 40 วินาทีต่อหนึ่งก้าว รูปที่ 5.3 แสดงถึงตัวอย่างการเดินของหุ่นยนต์จริงของคำตอบที่ได้จากการทดลอง โดยแสดงให้เห็นถึงการเคลื่อนไหวในแต่ละขั้นตอนการเดิน วงกลมสีขาวในภาพแสดงถึงพฤติกรรมหลักที่เกิดขึ้นในขั้นตอนแต่ละขั้นตอนดังนี้

รูป ก) หุ่นยนต์เริ่มต้นการทดลองด้วยท่าทางการยืนด้วยสองขา น้ำหนักของหุ่นยนต์ตกลงตรงกลางของพื้นที่รับน้ำหนักที่พื้น จากนั้นในรูป ข) หุ่นยนต์เริ่มขยับหางเพื่อเข้าสู่เป้าหมายย่อยของขั้นตอนที่หนึ่งซึ่งหุ่นยนต์จะต้องยืนด้วยเท้าขวา กล่าวคือหุ่นยนต์ถ่ายย้ายน้ำหนักไปยังเท้าขวา ในรูป ค) น้ำหนักของหุ่นยนต์จะอยู่ตำแหน่งใกล้เคียงกับจุดกึ่งกลางเท้าขวา รูป ง) หุ่นยนต์เข้าสู่ขั้นตอนการเดินที่ 2 คือยกเท้าซ้ายไปข้างหน้า และวางเท้าซ้ายลงกับพื้นในขั้นตอนที่ 3 ซึ่งแสดงในรูป จ) จาก 2 รูปนี้จะสังเกตได้ว่า หุ่นยนต์ยกเท้าแค่ให้พอพ้นพื้นเท่านั้น

ถัดจากนั้นในขั้นตอนที่ 4 หุ่นยนต์ย้ายน้ำหนักจากเท้าขวาไปอยู่ที่เท้าซ้าย ในขั้นตอนนี้ค่อนข้างยาก ถ้าขั้นตอนที่ 3 การวางเท้าของหุ่นยนต์วางในตำแหน่งที่ไกลเกินไปจะทำให้การถ่ายน้ำหนักระหว่างเท้าเป็นไปได้ยาก กล่าวคือ ต้องอาศัยการขยับของข้อต่อต่าง ๆ ทั้งหมดเพื่อประคองให้น้ำหนักของหุ่นยนต์เคลื่อนย้ายโดยยังอยู่ภายในพื้นที่รับน้ำหนักตลอดเวลา แต่ถ้าในขั้นตอนที่ 3 หุ่นยนต์วางเท้าใกล้เกินไป ก็จะมีผลทำให้ระยะก้าวของหุ่นยนต์สั้นไปด้วย ในช่วงแรกของการทดลอง ไม่ได้มีการกำหนดความยาวระยะก้าวไว้ ทำให้ขั้นตอนที่ 4 สำเร็จได้โดยยาก มีบางคำตอบที่หุ่นยนต์ก้าวเท้ายาวเกินไป และในพฤติกรรมที่เกิดขึ้นในขั้นตอนที่ 4 คือหุ่นยนต์สืบเท้านั้นถอยหลังจากระยะหนึ่งพอที่จะถ่ายน้ำหนักได้ง่ายขึ้นก่อน แล้วจึงค่อยถ่ายน้ำหนัก ดังนั้นในช่วงหลังจึงได้มีการกำหนดระยะก้าวของหุ่นยนต์ไว้ตั้งแต่ขั้นตอนที่สอง ซึ่งทำให้การถ่ายน้ำหนักในขั้นตอนที่สี่ทำได้ง่ายขึ้น กล่าวคือหุ่นยนต์เพียงอาศัยการขยับหางกับการขยับข้อต่ออื่นอีกเล็กน้อย ก็สามารถถ่ายน้ำหนักระหว่างเท้าได้แล้ว รูป ฉ) และ ช) แสดงพฤติกรรมที่เกิดขึ้นในช่วงขั้นตอนที่ 4

รูป ซ) แสดงถึงขั้นตอนที่ 5 หุ่นยนต์ก้าวเท้าขวา ซึ่งอยู่ข้างหลังให้มาอยู่ในระดับเดียวกับเท้าซ้าย และวางเท้าลงกับพื้นในขั้นตอนที่ 6 ดังรูป ฌ) และ ญ) หลังจากขั้นตอนที่ 6 หุ่นยนต์จะปรับแต่งท่าทาง ปรับเพิ่มลดมุมในแต่ละข้อต่อเพื่อกลับไปสู่ท่าทางเริ่มต้น ดังรูป ฏ) หลังจากนั้นหุ่นยนต์ก็จะสามารถเดินได้อย่างต่อเนื่องหลาย ๆ ก้าวโดยทำซ้ำพฤติกรรมที่เกิดขึ้นตั้งแต่รูป ก) ถึง ฏ) ไปเรื่อย ๆ กล่าวโดยสรุป ภาพการเดินที่เกิดขึ้นคือ หุ่นยนต์ก้าวเท้าซ้ายนำ แล้วจึงก้าวเท้าขวาตาม (แต่ไม่นำเท้าซ้าย) และก็ก้าวเท้าซ้ายนำก่อน แล้วค่อยก้าวเท้าขวาตาม เป็นเช่นนี้ทุก ๆ ก้าว

หลังจากหุ่นยนต์เดินหน้าได้แล้ว เพื่อเป็นการยืนยันว่าหุ่นยนต์เดินแบบสมดุลสถิต นั่นคือหุ่นยนต์จะอยู่ในสภาพสมดุลอยู่ทุกขณะ ผู้วิจัยได้ทำการทดลองเพิ่มเติมโดยใช้โปรแกรมคำตอบเดิมควบคุมให้หุ่นยนต์เดินถอยหลังในลักษณะคล้ายกับภาพการเดินแบบย้อนกลับ โดยกลับลำดับของโนดในโปรแกรมคำตอบจากหลังมาหน้า และกลับทิศทางการสั่งงานมอเตอร์

ยกตัวอย่างเช่น 3-:2, 1-:4, 0+:6, 6+:6
จะถูกเปลี่ยนเป็น 6-:6, 0-:6, 1+:4, 3+:2

การเดิน 1 ก้าวนั้นได้ถูกแบ่งออกเป็น 6 ขั้น และในการเดินหน้านั้น หลังจากการเดินในขั้นตอนที่หกแล้ว โปรแกรมอ่าน-แปลคำตอบจะปรับเปลี่ยนท่าทางของหุ่นยนต์ไปสู่ท่าทางเริ่มต้น (รูปที่ 4.7, หน้า 18) ดังนั้นในการเดินถอยหลังนั้น หุ่นยนต์จะต้องแสดงพฤติกรรมการเดินแบบย้อนกลับจากขั้นตอนที่หกไปสู่ขั้นตอนที่หนึ่ง หลังจากนั้นโปรแกรมอ่าน-แปลคำตอบจะต้องปรับเปลี่ยนท่าทางของหุ่นยนต์ (ซึ่งขณะนั้นเป็นท่าทางเริ่มต้น) ไปสู่ “ท่าทางสุดท้ายของขั้นตอนที่หก” เพื่อให้หุ่นยนต์สามารถเดินถอยหลังได้หลาย ๆ ก้าวอย่างต่อเนื่อง

หลังจากทำการทดลองให้หุ่นยนต์เดินถอยหลังโดยใช้โปรแกรมคำตอบเดิม และมีการอ่าน-แปลคำตอบในลักษณะดังกล่าวแล้ว ปรากฏว่าหุ่นยนต์สามารถเดินถอยหลังได้อย่างต่อเนื่องและไม่ล้มและไม่หมุน

นอกจากนั้น ได้มีการตรวจสอบดูว่าเมื่อหุ่นยนต์เดินหลาย ๆ ก้าวแล้ว หุ่นยนต์ยังคงเดินเป็นเส้นตรงอยู่หรือไม่ เดินเบี่ยงเบนมากน้อยเพียงไร ซึ่งได้ทดลองให้หุ่นยนต์เดิน 10 ก้าว แล้วทำการวัดระยะเบี่ยงเบน โปรแกรมคำตอบผลลัพธ์มีอยู่ 3 ตัว ทำการทดลองคำตอบละ 2 ครั้ง ในการเดิน 10 ก้าวของหุ่นยนต์ จะทำการวัดระยะทางที่หุ่นยนต์เดินไปข้างหน้าและระยะเบี่ยงเบน จากนั้นจะนำมาหาค่ามุมเบี่ยงเบน ซึ่งได้ผลดังตารางที่ 5.1

ตารางที่ 5.1: แสดงข้อมูลระยะและมุมเบี่ยงเบนของการเดิน 10 ก้าว

คำตอบที่	ครั้งที่	ระยะไปข้างหน้า (ซม.)	ระยะเบี่ยงเบน (ซม.)	มุมเบี่ยงเบน (องศา)
1	1	32.3	+6.4	+11.20
	2	35.0	0.0	+0.00
2	1	43.8	+3.4	+4.44
	2	39.0	+0.8	+1.18
3	1	46.5	-1.0	-1.24
	2	48.7	+4.3	+5.05
			มุมเบี่ยงเบน(สัมบูรณ์)เฉลี่ย	3.85

โดย เครื่องหมายบวก หมายถึงหุ่นยนต์เดินเบี่ยงเบนไปทางขวา
เครื่องหมายลบ หมายถึง หุ่นยนต์เดินเบี่ยงเบนไปทางซ้าย

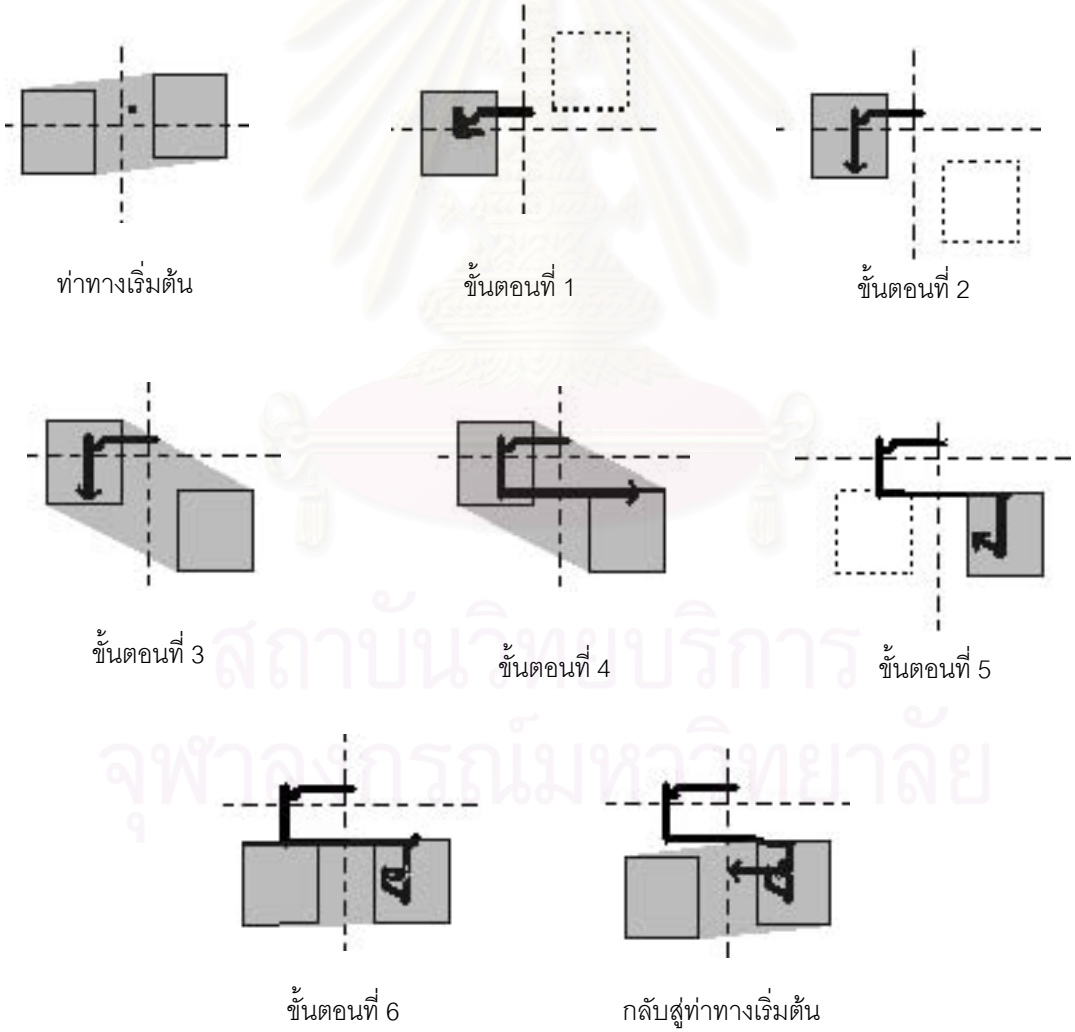
จากตารางที่ 5.1 จะให้เห็นว่าเมื่อหุ่นยนต์เดินมากกว่าหนึ่งก้าว จะเริ่มมีการเบี่ยงเบนบ้างเล็กน้อย ทั้งนี้อาจเนื่องมาจากหลายสาเหตุ เช่น

1) ความผิดพลาดของการตัดสินพฤติกรรมหุ่นยนต์ (ในเรื่องการหมุน) ของการทดลองกับหุ่นยนต์จริงที่ใช้มนุษย์เป็นผู้ตรวจสอบ กล่าวคือหุ่นยนต์อาจมีการหมุนบ้างเล็กน้อย ซึ่งมนุษย์อาจไม่ทันได้สังเกตเห็น (มุมเบี่ยงเบนในแต่ละก้าวที่มากที่สุดไม่เกิน 1.2°)

2) โดยโครงสร้างของหุ่นยนต์จริงนั้น ไม่สามารถสร้างให้หุ่นยนต์มีสัดส่วนถูกต้องสมบูรณ์ได้ ซึ่งหุ่นยนต์จริงที่ใช้ในงานวิจัยนี้ มีช่วงขาไม่ขนานกัน กล่าวคือเมื่อยืดขาทั้งสองข้างจนตึง จะเห็นว่าช่วงขาแยกออกจากกันเล็กน้อย ซึ่งการที่หุ่นยนต์มีลักษณะขาดังกล่าวอาจมีผลทำให้ไม่สามารถเดินตรงได้อย่างสมบูรณ์

5.3. คุณภาพของการเดิน

วิธีหนึ่งที่สามารถใช้ในการพิจารณาคุณภาพของการเดินคือ การพิจารณาความสมดุลของการเดินซึ่งดูได้จากการเคลื่อนที่ของน้ำหนักของหุ่นยนต์ กล่าวคือในการเดินแบบสมดุลสถิต การเดินจะสมดุลที่สุดและมีโอกาสล้มน้อยที่สุดได้นั้น น้ำหนักของหุ่นยนต์จะต้องอยู่ในตำแหน่งกึ่งกลางของพื้นที่รับน้ำหนักมากที่สุดในกรณีที่ไม่มีการถ่ายน้ำหนัก และในกรณีที่มีการถ่ายน้ำหนัก เส้นทางการ



รูปที่ 5.4 แสดงการเคลื่อนที่ของน้ำหนักหุ่นยนต์

เคลื่อนที่ของน้ำหนักจะต้องอยู่ใกล้แนวเส้นกึ่งกลางของพื้นที่รับน้ำหนักมากที่สุด

รูปที่ 5.4 เป็นการแสดงเส้นทางการเคลื่อนที่ของน้ำหนักหุ่นยนต์ที่เคลื่อนไปในแต่ละขั้นตอนของการเดิน รูปนี้วาดจากข้อมูลที่ได้ในสภาพแวดล้อมจำลอง โดยสี่เหลี่ยมจุดไข่ปลาหมายถึงเท้าที่ไม่สัมผัสพื้น สี่เหลี่ยมเส้นทึบหมายถึงเท้าที่สัมผัสพื้น และส่วนที่แรเงาหมายถึงพื้นที่รับน้ำหนัก จะสังเกตได้ว่าในท่าทางเริ่มต้นเท้าทั้งสองข้างของหุ่นยนต์ไม่ได้อยู่ได้ระดับเดียวกัน ทั้งนี้เนื่องมาจากการปรับแต่งของหุ่นยนต์ในสภาพแวดล้อมจำลองให้ใกล้เคียงกับหุ่นยนต์จริงมากขึ้น จากรูปจะเห็นได้ว่าขั้นตอนที่หุ่นยนต์มีโอกาสล้มมากที่สุดคือ ขั้นตอนที่ 2 และรองลงมาคือขั้นตอนที่ 3 และ 4 เนื่องจากหัวลูกศรซึ่งหมายถึงตำแหน่งของน้ำหนักของหุ่นยนต์อยู่ใกล้ขอบของพื้นที่รับน้ำหนักมาก

สาเหตุที่ทำให้หุ่นยนต์มีโอกาสล้มได้มากในขั้นตอนที่ 2, 3 และ 4 นั้น น่าจะมีสาเหตุมาจากการที่โปรแกรมจำลองพฤติกรรมทำ “เครื่องหมายระบุการล้มหรือบิดตัว” ของหุ่นยนต์ แล้วมีการตัดตอนเฉพาะส่วนก่อนหน้าเครื่องหมายระบุการล้มหรือบิดตัว แล้วนำไปใช้เป็นท่าทางเริ่มต้นของขั้นตอนถัดไปของการเดิน (ดังที่ได้อธิบายไว้ในหัวข้อที่ 5.1.3) เนื่องจากเครื่องหมายนี้ระบุตำแหน่งโนดของโปรแกรมคำตอบที่จะทำให้หุ่นยนต์ล้มหรือบิดตัว ดังนั้นจึงหมายความว่าพฤติกรรมที่หุ่นยนต์แสดงออกตามรหัสของโปรแกรมคำตอบในส่วนก่อนหน้าเครื่องหมายนี้จะไม่ทำให้หุ่นยนต์ล้ม **แต่เกือบล้ม** กล่าวคือ โนดสุดท้ายก่อนหน้าเครื่องหมายนี้หุ่นยนต์จะยังไม่ล้ม แต่โนดต่อไปหุ่นยนต์ล้มแล้ว

อย่างไรก็ตาม พฤติกรรมที่หุ่นยนต์แสดงออกว่าเดินได้โดยมีโอกาสล้มได้มากนั้น อาจมองได้อีกแง่หนึ่งว่าถ้าต้องการให้หุ่นยนต์ที่เดินแบบสมดุลสถิตสามารถเดินได้เร็ว ก็จำเป็นที่จะต้องทำให้หุ่นยนต์เดินโดยมีการเคลื่อนไหวของจุดศูนย์ถ่วงอยู่ใกล้ขอบของพื้นที่รับน้ำหนักนั่นเอง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6

บทสรุปและข้อเสนอแนะ

6.1. สรุปงานวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อหาวิธีสังเคราะห์โปรแกรมควบคุมการเดินของหุ่นยนต์แบบเดินสองขา ซึ่งเน้นการสังเคราะห์ให้เป็นแบบอัตโนมัติ และใช้วิธีการแบบขั้นตอนวิธีพันธุกรรมในการสังเคราะห์โปรแกรมควบคุม โปรแกรมควบคุมหรือโปรแกรมคำตอบที่ได้จากขั้นตอนวิธีพันธุกรรมจะใช้ควบคุมการเดินเพียงหนึ่งก้าวเท่านั้น และใช้วิธีสั่งการการเดินหนึ่งก้าวนี้ซ้ำกันไปเรื่อย ๆ เพื่อให้หุ่นยนต์สามารถเดินได้อย่างต่อเนื่อง

ท่าทางการเดินหนึ่งก้าวจะถูกแบ่งเป็น 6 ขั้นตอนย่อยและใช้ขั้นตอนวิธีพันธุกรรมในการสังเคราะห์คำตอบย่อยของแต่ละขั้นตอน นำมาประกอบกันเป็นคำตอบรวม ซึ่งข้อดีของการแบ่งการเดินเป็นขั้นตอนย่อย ๆ คือสามารถกำหนดเกณฑ์การวัดค่าความเหมาะสมเฉพาะ (particular fitness function) ในแต่ละขั้นให้แตกต่างกันไปให้เหมาะสมกับเป้าหมายย่อยของแต่ละขั้นตอนได้ ซึ่งจะทำให้การได้คำตอบใหญ่นั้นง่ายขึ้น

ในงานวิจัยที่เกี่ยวกับการนำขั้นตอนวิธีพันธุกรรมมาใช้ในการหาโปรแกรมควบคุมหุ่นยนต์นั้น ส่วนใหญ่มักทำการทดลองในสภาพแวดล้อมจำลองเพื่อประหยัดเวลาและป้องกันความเสียหายของหุ่นยนต์ จากนั้นจึงนำคำตอบที่ได้มาใช้กับหุ่นยนต์จริง ซึ่งมักไม่ประสบความสำเร็จเนื่องจากความแตกต่างหลายประการของสภาพแวดล้อมจำลองกับสภาพแวดล้อมจริง ดังนั้นในงานวิจัยนี้จึงใช้วิธีการทำการทดลองทั้งในสภาพแวดล้อมจำลองและทดลองหุ่นยนต์จริงผสมผสานกันไป โดยในแต่ละขั้นตอนย่อยจะมีการทดลองในสภาพแวดล้อมทั้งสองแบบ ซึ่งตรงจุดนี้เป็นจุดเด่นของงานวิจัยนี้ในการแก้ไขปัญหาดังกล่าว

จากผลการทดลองที่ได้ แสดงให้เห็นว่า ถึงแม้คุณภาพของการเดินเมื่อพิจารณาจากการรักษาสมดุลตลอดช่วงการเดินจะมีไม่มากนักก็ตาม แต่ก็สามารถกล่าวได้ว่าหุ่นยนต์สามารถรักษาสมดุลของการเดินได้อย่างต่อเนื่องตลอดช่วงของการเดิน

กล่าวโดยสรุป ประเด็นของงานวิจัยนี้อยู่ที่การขยายขอบเขตของการประยุกต์ใช้การคำนวณเชิงวิวัฒน์ในการแก้ไขปัญหาที่ซับซ้อน โดยเลือกการเดินของหุ่นยนต์สองขามาเป็นโจทย์ของงานวิจัย และในกลุ่มงานที่เกี่ยวกับการนำการคำนวณเชิงวิวัฒน์มาใช้แก้ปัญหาการเดินของหุ่นยนต์สองขานั้น

ส่วนใหญ่จะทำการทดลองในสภาพแวดล้อมจำลองเท่านั้น ซึ่งเป็นที่รู้กันว่าคำตอบที่ได้จากการทดลองในสภาพแวดล้อมจำลองมักไม่สามารถทำงานได้ประสบความสำเร็จมากนักเมื่อนำมาใช้กับระบบจริง ดังนั้นการประยุกต์ใช้การคำนวณเชิงวิวัฒน์มาแก้ปัญหาการเดินของหุ่นยนต์สองขาของงานวิจัยนี้ จึงกำหนดให้คำตอบที่ได้รับจากการแก้ปัญหานั้น ต้องเป็นคำตอบสามารถใช้ได้กับหุ่นยนต์จริงด้วย และสามารถสรุปได้ว่าด้วยระบบของการทดลองตามแบบของงานวิจัยนี้ การคำนวณเชิงวิวัฒน์สามารถสังเคราะห์หาคำตอบที่ทำให้หุ่นยนต์สองขาเดินได้ในสภาพแวดล้อมจริงอย่างต่อเนื่อง

6.2. ข้อเสนอแนะ

กล่าวในแง่ของการพัฒนาหุ่นยนต์ในระยะยาว งานวิจัยนี้เป็นเพียงงานในช่วงเริ่มต้นของการนำจุดเด่นในเรื่องความสามารถในปรับตัวเองได้โดยอัตโนมัติของวิธีการคำนวณเชิงวิวัฒน์มาใช้กับงานวิจัยทางด้านหุ่นยนต์ศาสตร์เท่านั้น ตัวอย่างของความสามารถปรับตัวเองได้โดยอัตโนมัติ เช่น หุ่นยนต์สามารถปรับตัวให้ทำงานต่อไปได้ถึงแม้โครงสร้างบางส่วนของหุ่นยนต์หรือสภาพแวดล้อมบางอย่างจะเปลี่ยนแปลงไป เป็นต้น

ในการที่จะปรับตัวได้นั้น เงื่อนไขหนึ่งที่จำเป็นคือหุ่นยนต์จะต้องมีหน่วยรับรู้ (sensor) ชนิดต่าง ๆ ติดตั้งอยู่ในระบบด้วย ยิ่งหุ่นยนต์มีหน่วยรับรู้หรือระบบรับรู้มาก หุ่นยนต์ก็จะมีความสามารถในการปรับตัวได้มากด้วย อย่างไรก็ตาม ในงานวิจัยนี้ตัวหุ่นยนต์จริงไม่มีหน่วยรับรู้ใด ๆ ติดตั้งอยู่เลย การประเมินค่าความเหมาะสมต่าง ๆ ที่กระทำใน “การทดลองกับหุ่นยนต์จริง” ของงานวิจัยนี้กระทำโดยใช้มนุษย์เป็นผู้ตัดสินให้ทั้งสิ้น อีกทั้งใน “การทดลองในสภาพแวดล้อมจำลอง” ยังจำเป็นต้องอาศัยการป้อนข้อมูลโครงสร้าง น้ำหนัก และสัดส่วนต่าง ๆ ของหุ่นยนต์เพื่อใช้คำนวณจุดศูนย์ถ่วงโดยรวมอีกด้วย ซึ่งจุดเหล่านี้เป็นจุดที่ทำให้กล่าวได้ว่าหุ่นยนต์ที่ใช้ในงานวิจัยนี้ยังไม่สามารถปรับตัวเองได้โดยอัตโนมัติหรือโดยไม่มีมนุษย์เข้าไปข้องเกี่ยว ดังนั้น การติดตั้งระบบรับรู้ต่าง ๆ ให้หุ่นยนต์สามารถรับรู้ถึงโครงสร้างของตัวหุ่นยนต์เอง และรับรู้ถึงสภาวะแวดล้อมต่าง ๆ ได้มากขึ้น เป็นเงื่อนไขหนึ่งที่จะทำให้การพัฒนาหุ่นยนต์ให้สามารถปรับตัวเองได้โดยอัตโนมัติมีความเป็นไปได้มากขึ้น

อีกประเด็นหนึ่งของการปรับตัวเองได้ของหุ่นยนต์ คือเรื่องความเร็วหรือระยะเวลาที่ใช้ในการปรับตัว ซึ่งขึ้นอยู่กับความเร็วของระบบประมวลผล กล่าวได้ว่าการที่ความเร็วในการปรับตัวของหุ่นยนต์โดยอัตโนมัติยังไม่สามารถเทียบได้กับการใช้มนุษย์ช่วยในการปรับตัวของหุ่นยนต์นั้น เนื่องจากความเร็วของระบบประมวลผลในปัจจุบันยังมีไม่มากพอ แต่ก็เป็นที่คาดหวังไว้ว่าในอนาคตความเร็วของระบบประมวลผลน่าจะยังเพิ่มขึ้นได้อีกมาก ซึ่งเมื่อถึงเวลานั้นแล้วการปรับตัวโดยอัตโนมัติและการปรับตัวโดยมนุษย์เข้าไปเกี่ยวข้องนั้นน่าจะใช้เวลาใกล้เคียงกัน และอาจไม่มีความจำเป็นที่มนุษย์จะต้องเข้าไปช่วยในเรื่องนี้

ในงานวิจัยนี้ การประเมินค่าความเหมาะสมของโปรแกรมคำตอบแต่ละตัว กระทำจากพฤติกรรมที่แสดงออกของหุ่นยนต์ กล่าวคือ ก่อนที่จะทำการประเมิน หุ่นยนต์จะต้องอยู่ในท่าทางเริ่มต้น แล้วจึงมีการอ่านโปรแกรมคำตอบเพื่อสั่งการหุ่นยนต์ต่าง ๆ จากนั้นจึงค่อยประเมินค่าความเหมาะสม เมื่อจะประเมินโปรแกรมคำตอบถัดไป ก็จำเป็นต้องจัดให้หุ่นยนต์กลับไปอยู่ในสถานะเริ่มต้น ซึ่งในงานวิจัยนี้ ยังต้องอาศัยมนุษย์ยกหุ่นยนต์ไปวางไว้ที่ตำแหน่งเดิม ดังนั้นในการที่จะทำให้หุ่นยนต์ปรับตัวเองได้เป็นอัตโนมัติมากขึ้นนั้น จุดนี้ก็เป็นอีกหนึ่งจุดที่จำเป็นจะต้องนำมนุษย์ออกจากกระบวนการ ซึ่งอาจทำได้โดยจัดสร้างระบบอัตโนมัติขึ้นมาแทน หรืออาจคิดหาวิธีการประเมินโดยไม่ต้องกลับไปอยู่ในสถานะเริ่มต้นก็ได้

นอกจากนี้ ในกรณีที่หุ่นยนต์ล้มในช่วงการประเมินค่าความเหมาะสมของโปรแกรมคำตอบแต่ละตัว อาจทำให้หุ่นยนต์ชำรุดเสียหายได้ ในงานวิจัยนี้ใช้วิธีผูกเชือกโยงตัวหุ่นยนต์ไว้กับคานเหนือหุ่นยนต์ และใช้มนุษย์ช่วยประคองหุ่นยนต์เวลาจะล้ม ดังนั้นจึงควรมีระบบการป้องกันการชำรุดเสียหายของหุ่นยนต์จริงไว้ในกรณีนี้ด้วย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

Arakawa, T., Fukuda, T. (1996). "Natural motion trajectory generation of biped locomotion robot using genetic algorithm through energy optimization", **Proc. of IEEE Int. Conf. on Systems, Man and Cybernetics**, Vol. 2, 1495 - 1500.

Arakawa, T., Fukuda, T. (1997). "Natural motion generation of biped locomotion robot using hierarchical trajectory generation method consisting of GA, EP layers", **Proc. of IEEE Int. Conf. on Robotics and Automation**, Vol. 1, 211 - 216.

Brooks, R. A. (1991a). "Artificial Life and Real Robots", **Proc. of European Conf. on Artifcial Life**, Paris, France, MIT Press, December, 3 - 10.

Brooks, R. A. (1991b). "Intelligence without representation", **Artificial Intelligence**, 47, 139 - 160.

Cheng, M.-Y., Lin, C.-S. (1995). "Genetic algorithm for control design of biped locomotion", **Proc. of Int. Conf. on Intelligent Systems for the 21st Century**, Vol. 2, 1315 - 1320.

Davidor, Y. (1990). "Robot programming with a genetic algorithm", **Proc. of IEEE Int. Conf. on Computer Systems and Software Engineering**, 186 - 191.

Goldberg, D. E. (1989). **Genetic Algorithm in Search, Optimization, and Machine Learning**, Addison Wesley.

Hirai, K., Hirose, M., Haikawa, Y., Takenaka, T. (1998). "The development of Honda humanoid robot", **Proc. of IEEE Int. Conf. on Robotics and Automation**, Vol. 2, 1321 - 1326.

Holland, J. H. (1975). **Adaptation in Natural and Artificial System**, Ann Arbor, University of Michigan Press.

Inaba, M., Kanehiro, F., Kagami, S., Inoue, H. (1995). "Two-Armed Bipedal Robot that can Walk, Roll Over and Stand up", **Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems**, Vol. 3, 297 - 302.

Koza, J. R., and Rice, J. P. (1992). "Automatic programming of Robots using Genetic Programming", **Proc. 10th National Conf. on Artificial Intelligence**, 194 -201.

Kun, A. L., Miller, W. T. (1997). "Adaptive Static Balance of a Biped Robot Using Neural Networks", **Proc. of the IASTED Int. Conf. on Robotics and Manufacturing**, 245 - 248.

Polvichai, J. (1996). **Learning of a robot arm by genetic programming**, Master Thesis, Department of Computer Engineering, Chulalongkorn University. (in Thai)

Rodrigues, L., Prado, M., Tavares, P., Da Silva, K., Rosa, A. (1996). "Simulation and control of biped locomotion-GA optimization", **Proc. of IEEE Int. Conf. on Evolutionary Computation**, 390 - 395.

Winston, P. (1992). **Artificial intelligence**, Addison-Wesley, 505 - 528.

Zheng, Y. F., Shen. J., Sias, R. F., Jr. (1988). "A motion control scheme for a biped robot to climb sloping surfaces", **Proc. IEEE Int. Conf. on Robotic and Automation**, Vol. 2, 814 - 816.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้วิจัย

นายชาญชัย ชัยสุขโกศล เกิดวันที่ 8 พฤศจิกายน พ.ศ. 2520 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาตรีวิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ในปี พ.ศ. 2541 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรม-คอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2542

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย