ระบบหลากหลายตัวกระทำบนพื้นฐานของระบบแบล็กบอร์ด สำหรับหุ่นยนต์เล่นฟุตบอล

THE MULTI-AGENT SYSTEM BASED ON BLACKBOARD SYSTEM FOR ROBOT SOCCER IMPLEMENTATION

นิธิวัฒน์ เสนาะนิ่ยม
NITIWAT SANORNOI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549

ISBN 974-15-2724-1

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบหลากหลายตัวกระทำบนพื้นฐานของระบบแบล็กบอร์ด สำหรับหุ่นยนต์แข่งฟุตบอล

THE MULTI-AGENT SYSTEM BASED ON BLACKBOARD SYSTEM FOR ROBOT SOCCER IMPLEMENTATION

นิธิวัฒน์ เสนาะน้อย

NITIWAT SANORNOI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2549

ISBN 974-15-2724-1

THE MULTI-AGENT SYSTEM BASED ON BLACKBOARD SYSTEM FOR ROBOT SOCCER IMPLEMENTATION

NITIWAT SANORNOI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN INFORMATION ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

ISBN 974-15-2724-1

COPYRIGHT 2006

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	ระบบหลากหลายตัวกระทำบนพื้นฐานของระบบแบล็กบอร์ดสำหรับหุ่นยนต์เล่นฟุตบอล
นักศึกษา	นายนิธิวัฒน์ เสนาะน้อย
รหัสนักศึกษา	46061334
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมสารสนเทศ
พ.ศ.	2548
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร.ปิติเขต สู้รักษา

บทคัดย่อ

งานวิจัยนี้ได้นำเสนอการออกแบบและการประยุกต์การใช้งาน ระบบหลากหลายตัวกระทำบนพื้นของของระบบแบล็กบอร์ดสำหรับการควบคุมหุ่นยนต์เล่นฟุตบอล ระบบควบคุมนี้ทำงานในลักษณะเวลาจริงโดยอาศัยตัวกระทำหลายชนิด ซึ่งประมวลผลแบบขนานทำงานประสานกัน และใช้หน่วยความจำแบล็กบอร์ดที่ออกแบบโดยเฉพาะ เป็นองค์ประกอบของระบบ เนื่องจากสามารถปรับเปลี่ยนแก้ไขระบบตามการใช้งานได้โดยง่าย และยังสะดวกในการปรับปรุงระบบให้มีประสิทธิภาพมากยิ่งขึ้น โดยแสดงถึงการปรับเปลี่ยนระบบและการทดสอบตามแผนการเล่น, ตามการนำร่องและตามลักษณะการขับเคลื่อนของหุ่นยนต์ที่ต่างกัน ผลการทดลองได้แสดงว่าระบบดังกล่าวสามารถนำมาใช้กับการควบคุมหุ่นยนต์เตะฟุตบอลได้เป็นอย่างดี

Thesis Title	The Multi-agent System Based on Blackboard for Robot Soccer Implementation
Student	Mr. Nitiwat Sanornoi
Student ID.	46061334
Degree	Master of Engineering
Program	Information Engineering
Year	2005
Thesis Advisor	Assoc. Prof. Dr. Pitikhate Sooraksa

ABSTRACT

This research presents the design and application of Multi-agent system for soccer robot system control. This real-time system works by using cooperation of parallel various agents and a specifically blackboard memory structure because it's adapted and adjusted easily. The purposed research shows the improvement and test for different playing plans, navigation and model of soccer robot. According to the experiment result, The purposed system can used to be robot soccer control system rightly.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดีเพราะได้รับการแนะนำและคำปรึกษาจาก รศ.ดร.ปิติเขต สู้รักษา ซึ่งเป็นอาจารย์ควบคุมวิทยานิพนธ์ รวมทั้ง ผศ.บุณย์ชนะ ภู่ระหงษ์ ที่ให้คำปรึกษาและสนับสนุนมาตลอด

ขอขอบคุณภาควิชาวิศวกรรมสารสนเทศ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้ให้สถานที่ในการทำวิจัยแก่ข้าพเจ้าจนวิทยานิพนธ์สำเร็จลุล่วงอย่างดี รวมถึงเพื่อน ๆ พี่ ๆ และน้องที่ร่วมงานกันมาและเป็นกำลังใจให้เสมอ

ขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจและให้ทุนทรัพย์เพื่อการศึกษา และให้การสนับสนุนในเรื่องราวต่าง ๆ

คุณงามความดีอันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน รวมทั้งผู้ที่ทำคุณประโยชน์ให้แก่ประเทศชาติและแก่มวลมนุษย์ด้วย

นิธิวัฒน์ เสนาะน้อย

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VIII

สารบัญรูป.....IX

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....1

1.3 สมมติฐานของการศึกษา.....1

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย.....2

1.5 ขอบเขตการวิจัย.....3

1.6 ขั้นตอนการศึกษา.....3

บทที่ 2 ทฤษฎีระบบหลากหลายตัวกระทำ.....4

2.1 ทฤษฎีและหลักการพื้นฐานของระบบแบล็กบอร์ด.....4

2.2 โครงสร้างของระบบแบล็กบอร์ด.....5

2.3 การจัดโครงสร้างระบบแบล็กบอร์ดสำหรับการควบคุม.....7

2.4 การทำงานของตัวกระทำในสภาวะการทำงานจริง.....8

2.4.1 การจัดระเบียบการทำงานของตัวกระทำแบบอนุกรม.....8

2.4.2 การจัดระเบียบการทำงานของตัวกระทำแบบกึ่งขนานกึ่งอนุกรม..... 10

2.4.3 การจัดระเบียบการทำงานของตัวกระทำแบบขนาน.....11

บทที่ 3 ระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอล.....13

3.1 โครงสร้างของระบบ.....13

3.1.1 ชั้นโลกเสมือน (Abstract World Layer).....14

3.1.2 ชั้นคุณสมบัติหุ่นยนต์ (Robot Specific Layer).....17

3.1.3 ชั้นภารกิจ (Mission Layer).....17

3.1.4 ชั้นจุดประสงค์ (Objective Layer).....19

สารบัญ (ต่อ)

หน้า

3.1.5 ชั้นการกระทำ (Action Layer).....22

3.2 รายละเอียดของผู้เชี่ยวชาญ.....22

3.2.1 ตัวกระทำอวัยวะ (Organ Agent).....23

3.2.2 ตัวกระทำปรับข้อมูล (Normalize Agent).....23

3.2.3 ตัวกระทำวางแผน (Strategic Agent).....24

3.2.4 ตัวกระทำข้อมูลจากภารกิจสูงจุดประสงค์ (Mission to Objective Agent).....24

3.2.5 ตัวกระทำข้อมูลจากจุดประสงค์สูงการกระทำ (Objective to Action Agent)....26

บทที่ 4 แผนการเล่นและกลยุทธ์.....27

4.1 การคำนวณตำแหน่งการเล่นของหุ่นยนต์เล่นฟุตบอล.....27

4.1.1 ตำแหน่งตัวไล่บอลและรุก (Chosen).....30

4.1.2 ตำแหน่งตัวประกบ (Marker).....31

4.1.3 ตำแหน่งตัวบังการยิง (Coverer).....32

4.1.4 ตำแหน่งผู้รักษาประตู (Keeper).....33

4.2 การคำนวณสถานการณ์การเล่นของทีม.....34

4.2.1 สถานการณ์ลูกฟุตบอลอยู่ในเขตประตู (Goal zone).....34

4.2.2 สถานการณ์ป้องกัน (Defend).....35

4.2.3 สถานการณ์เก็บบอล (Collect).....36

4.2.4 สถานการณ์รุก (Attack).....37

บทที่ 5 รูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอล.....39

5.1 ตัวกระทำที่ทำงานตามจุดประสงค์ของหุ่นยนต์.....39

5.2 การคำนวณข้อมูลตามจุดประสงค์ของการเล่น.....40

5.2.1 การคำนวณจุดบังการยิงของฝ่ายตรงข้าม.....40

5.2.2 การคำนวณจุดหลบวิถีลูกฟุตบอลจากหุ่นยนต์ผู้รักษาประตู.....46

5.2.3 การคำนวณจุดประกบแบบตัวต่อตัว.....47

5.2.4 การคำนวณจุดประกบแบบกดดัน.....50

สารบัญ (ต่อ)

หน้า

5.2.5 การคำนวณจุดประกบแบบเปิดทาง....54

5.3 การคำนวณข้อมูลตามตำแหน่งของการเล่น....56

5.3.1 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งผู้รักษาประตู....56

5.3.2 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งตัวบังการยิง....61

5.3.3 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งตัวประกบ....61

5.3.4 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก....61

5.4 การคำนวณข้อมูลเพื่อประสิทธิภาพการเล่น....67

5.4.1 การคำนวณจุดที่อยู่นอกเขตการเล่น....67

5.4.2 การคำนวณจุดที่อยู่ในเขตประตู....68

5.4.3 การคำนวณจุดนำร่องเพื่อวิ่งอ้อมเขตประตู....69

5.4.4 การคำนวณจุดนำร่องเพื่อวิ่งอ้อมสิ่งกีดขวาง....71

5.4.5 การคำนวณคุณสมบัติของจุดนำร่อง....76

บทที่ 6 การนำร่องสำหรับหุ่นยนต์เล่นฟุตบอล....78

6.1 แบบจำลองระบบขับเคลื่อนของหุ่นยนต์....78

6.1.1 ระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ....78

6.1.2 ระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ....80

6.1.3 ระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 4 ล้อ....82

6.1.4 แบบจำลองการเคลื่อนที่ของหุ่นยนต์เล่นฟุตบอล....83

6.1.5 การแปลงความเร็วในพิกัดหุ่นยนต์มาเป็นความเร็วล้อ....79

6.2 การควบคุมที่ใช้ในการนำร่อง....85

6.2.1 การควบคุมการหันหน้าแบบพีไอดี (PID)....85

6.2.2 การควบคุมการหันหน้าแบบฟัซซี่ (Fuzzy)....86

6.2.3 การควบคุมความเร็วการวิ่งแบบพีไอดี....87

6.2.4 การควบคุมความเร็วการวิ่งแบบฟัซซี่....87

6.3 ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอล....88

6.3.1 ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลแบบเก้าอี้รถเข็น 2 ล้อ....89

6.3.2 ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลแบบเคลื่อนที่ได้ทุกทิศทาง....90

สารบัญ (ต่อ)

หน้า

บทที่ 7 การทดลอง.....92

7.1 การทดลองการนำร่องสำหรับหุ่นยนต์แบบเก้าอี้รถเข็นสองล้อ.....92

7.1.1 การนำร่องแบบ FTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ.....92

7.1.2 การนำร่องแบบ FTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ.....94

7.1.3 การนำร่องแบบ PTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ.....96

7.1.4 การนำร่องแบบ PTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ.....98

7.2 การทดลองการนำร่องสำหรับหุ่นยนต์เคลื่อนที่ทุกทิศทางสามล้อ.....100

7.2.1 การนำร่องแบบ FTFD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 3 ล้อ.....100

7.2.2 การนำร่องแบบ FTPD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 3 ล้อ.....102

7.2.3 การนำร่องแบบ PTFD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 3 ล้อ.....104

7.2.4 การนำร่องแบบ PTPD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 3 ล้อ.....106

7.3 การทดลองการนำร่องสำหรับหุ่นยนต์เคลื่อนที่ทุกทิศทางสี่ล้อ.....108

7.3.1 การนำร่องแบบ FTFD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง4 ล้อ.....109

7.3.2 การนำร่องแบบ FTPD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 4 ล้อ.....111

7.3.3 การนำร่องแบบ PTFD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 4 ล้อ.....113

7.3.4 การนำร่องแบบ PTPD สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 4 ล้อ.....115

7.4 การวัดประสิทธิภาพของการนำร่อง.....117

7.5 ผลการแข่งขันในการทดสอบบนโปรแกรมจำลองสถานการณ์.....118

7.6 รูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอลในการจำลองสถานการณ์.....119

7.6.1รูปแบบการเล่นของแผน P1301 ในการจำลองสถานการณ์.....120

7.6.2รูปแบบการเล่นของแผน P1211 ในการจำลองสถานการณ์.....123

7.6.3รูปแบบการเล่นของแผน P1121 ในการจำลองสถานการณ์.....127

7.6.4รูปแบบการเล่นของแผน P1031 ในการจำลองสถานการณ์.....131

7.6.5รูปแบบการเล่นของแผน P0041 ในการจำลองสถานการณ์.....135

บทที่ 8 สรุปการวิจัยและข้อเสนอแนะ.....140

8.1 สรุปผลการทดลอง.....140

8.1 ข้อแนะนำในการวิจัยและแนวทางในการพัฒนาในอนาคต.....141

สารบัญ (ต่อ)

หน้า

บรรณานุกรม.....142

ภาคผนวก.....144

ประวัติผู้เขียน.....154

สารบัญตาราง

ตารางที่ หน้า

3.1 ข้อมูลที่กำหนดตายตัวอยู่ในชั้นโลกเสมือน.....14

3.2 ข้อมูลที่ใช้ในการคำนวณในชั้นโลกเสมือน.....15

3.3 ข้อมูลที่ใช้ในการคำนวณในชั้นคุณสมบัติหุ่นยนต์.....17

3.4 ข้อมูลที่ใช้ในการคำนวณในชั้นภารกิจ.....18

3.5 ข้อมูลที่กำหนดตายตัวอยู่ในชั้นจุดประสงค์.....19

3.6 ข้อมูลที่ใช้ในการคำนวณในชั้นจุดประสงค์.....20

3.7 ข้อมูลที่ใช้ในการคำนวณในชั้นการกระทำ.....22

3.8 รายชื่อตัวกระทำอวัยวะ..... 23

3.9 รายชื่อตัวกระทำปรับข้อมูล.....24

3.10 รายชื่อตัวกระทำวางแผน.....24

3.11 รายชื่อตัวกระทำข้อมูลจากภารกิจสู่จุดประสงค์..... 25

3.12 รายชื่อตัวกระทำข้อมูลจากจุดประสงค์สู่การกระทำ.....26

4.1 รายละเอียดของข้อมูล TACTICS ในคุณสมบัติหุ่นยนต์..... 27

5.1 รายละเอียดของตัวกระทำที่คำนวณตามจุดประสงค์ของการเล่น.....39

5.2 รายละเอียดของตัวกระทำที่คำนวณจุดที่หุ่นยนต์วิ่งไปและองศาด้านหน้าของหุ่นยนต์.....40

5.3 รายละเอียดของตัวแปรเพื่อการคำนวณหาจุดนำร่องการบังการยิง.....44

5.4 รายละเอียดของตัวแปรเพื่อการคำนวณหาจุดนำร่องสำหรับหลบวิถีบอล.....47

5.5 รายละเอียดตัวแปรเพื่อการคำนวณหาจุดนำร่องสำหรับการประกบหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านบน.....51

5.6 รายละเอียดตัวแปรเพื่อการคำนวณหาจุดนำร่องสำหรับการประกบหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านล่างบน.....51

5.7 การนำร่องหุ่นยนต์เล่นฟุตบอลตำแหน่งผู้รักษาให้วิ่งอ้อมไปหลังลูกฟุตบอล.....58

5.8 จุดนำร่องเพื่อวิ่งอ้อมด้านหลังลูกฟุตบอล.....58

5.9 คุณสมบัติของจุดนำร่อง.....77

6.1 ทิศทางการวิ่งของระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง.....90

7.1 ผลการวัดประสิทธิภาพของการนำร่องในส่วนระยะทาง.....118

7.2 ผลการวัดประสิทธิภาพของการนำร่องในส่วนมุมด้านหน้าของหุ่นยนต์.....118

สารบัญตาราง (ต่อ)

ตารางที่ หน้า

7.3 ผลการแข่งขันในการทดสอบบนโปรแกรมจำลองสถานการณ์.....119

สารบัญรูป

รูปที่ หน้า

2.1 รูปตัวอย่างการแก้ปัญหา.....5

2.2 โครงสร้างระบบแบล็กบอร์ดแบบดั้งเดิม (Classical Blackboard).....6

2.3 ระบบแบล็กบอร์ดสำหรับการควบคุม.....8

2.4 ตัวอย่างโปรแกรมแกรมเสมือน (Pseudo Code) ของระบบแบบอนุกรม.....9

2.5 การทำงานของระบบแบบอนุกรม.....9

2.6 ตัวอย่างโปรแกรมเสมือนของระบบกึ่งขนานกึ่งอนุกรม.....10

2.7 การทำงานของระบบกึ่งขนานกึ่งอนุกรม.....10

2.8 ตัวอย่างโปรแกรมเสมือนของระบบแบบขนาน.....11

2.9 การทำงานของระบบแบบขนาน.....12

3.1 โครงสร้างที่ได้นำเสนอ.....13

4.1ก ตัวอย่างตำแหน่งการเล่นของแผน P1301.....28

4.1ข ตัวอย่างตำแหน่งการเล่นของแผน P1211.....28

4.1ค ตัวอย่างตำแหน่งการเล่นของแผน P1121.....29

4.1ง ตัวอย่างตำแหน่งการเล่นของแผน P1031.....29

4.1จ ตัวอย่างตำแหน่งการเล่นของแผน P0041.....30

4.2 ตัวอย่างการประกบของหุ่นยนต์ตำแหน่งตัวประกบ.....31

4.3 ตำแหน่งการประกบของหุ่นยนต์ตำแหน่งตัวประกบ.....32

4.4 ตัวอย่างการบังของหุ่นยนต์ตำแหน่งตัวบังการยิง.....33

4.5 ตัวอย่างการป้องกันประตูของหุ่นยนต์ตำแหน่งผู้รักษาประตู.....34

4.6 ตัวอย่างการสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู.....35

4.7 ตัวอย่างการสถานการณ์ป้องกัน.....36

4.8 ตัวอย่างการสถานการณ์เก็บบอล.....37

4.9 ตัวอย่างการสถานการณ์รุก.....38

5.1 ตัวอย่างการจุดอ้างอิงการบัง.....44

5.2 ตัวอย่างการกำหนดจุดนำร่องสำหรับหุ่นยนต์ตำแหน่งบังการยิง.....46

5.3 ตัวอย่างการกำหนดจุดนำร่องสำหรับการหลบวิถีลูกบอล.....47

สารบัญรูป (ต่อ)

รูปที่ หน้า

5.4 การคำนวณตัวประกบของการประกบแบบตัวต่อตัว..... 49

5.5 ตัวอย่างการคำนวณตัวประกบของการประกบแบบตัวต่อตัว..... 50

5.6 ตัวอย่างจุดของหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านบนและด้านล่าง.....52

5.7 ตัวอย่างการคำนวณตัวประกบของการประกบแบบกดดัน..... 53

5.8 การคำนวณตัวประกบของการประกบแบบกดดัน.....54

5.9 การคำนวณตัวประกบของการประกบแบบเปิดทาง.....55

5.10 ตัวอย่างการคำนวณตัวประกบของการประกบแบบเปิดทาง..... 56

5.11 ตัวอย่างตัวอย่างการคำนวณจุดนำร่องเพื่อบังวิถีลูกฟุตบอล.....57

5.12 การแบ่งพื้นที่เพื่อนำร่องให้หุ่นยนต์ตำแหน่งผู้รักษาประตูวิ่งอ้อมไปด้านหลัง.....58

5.13 แผนภาพการทำงานของการนำร่องสำหรับหุ่นยนต์ตำแหน่งผู้รักษาประตู..... 60

5.14 ตัวอย่างการแย่งบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก..... 62

5.15 แผนภาพการทำงานสำหรับหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกในสถานการณ์เก็บบอล.....63

5.16 ตัวอย่างเข้าครอบครองลูกฟุตบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก..... 64

5.17 ตัวอย่างเข้าครอบครองลูกฟุตบอลโดยแตะบอลอ้อมหุ่นยนต์ฝ่ายตรงข้าม..... 65

5.18 ตัวอย่างเข้าครอบครองลูกฟุตบอลในมุมขอบสนามด้านฝ่ายตรงข้าม.....66

5.19 ขอบเขตของจุดนำร่องในสนามสำหรับหุ่นยนต์เล่นฟุตบอล.....68

5.20 ขอบเขตนอกเขตประตูของจุดนำร่องหรับหุ่นยนต์เล่นฟุตบอล.....69

5.21 ตัวอย่างการคำนวณวิถีการวิ่งใหม่เพื่อวิ่งอ้อมเขตประตู..... 70

5.22 ตัวอย่างการคำนวณวิถีการวิ่งใหม่เพื่อวิ่งอ้อมสิ่งกีดขวาง.....72

5.23 ตัวอย่างจุดนำร่องตัดผ่านสิ่งกีดขวางหลายตัว.....73

5.24 จุดนำร่องที่อ้อมสิ่งกีดขวางแรกแล้วยังไปทับกับสิ่งกีดขวางอื่น ๆ.....74

5.25 การคำนวณหาจุดนำร่องสำหรับวิ่งอ้อมสิ่งกีดขวาง.....75

5.26 ตัวอย่างจุดนำร่องที่อ้อมสิ่งกีดขวางอย่างสมบูรณ์..... 76

6.1 รูปแบบจำลองระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ.....78

6.2 รูปแบบจำลองระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ.....80

6.3 ล้อเคลื่อนที่ได้ทุกทิศทาง.....81

6.4 รูปแบบจำลองระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 4 ล้อ.....82

สารบัญรูป (ต่อ)

รูปที่ หน้า

6.5 ระบบนำร่องสำหรับหุ่นยนต์เก้าอี้รถเข็น 2 ล้อ.....89

6.6 ระบบนำร่องสำหรับหุ่นยนต์เคลื่อนที่ได้ทุกทิศทาง.....90

7.1 ระยะในแกน X ของการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ...93

7.2 ระยะในแกน Y ของการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ...93

7.3 องศาด้านหน้าของการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ....94

7.4 ระยะในแกน X ของการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ....95

7.5 ระยะในแกน Y ของการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ...95

7.6 องศาด้านหน้าของการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ....96

7.7 ระยะในแกน X ของการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ...97

7.8 ระยะในแกน Y ของการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ...97

7.9 องศาด้านหน้าของการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ....98

7.10 ระยะในแกน X ของการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ....99

7.11 ระยะในแกน Y ของการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ... 99

7.12 องศาด้านหน้าของการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ...100

7.13 ระยะในแกน X ของการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....101

7.14 ระยะในแกน Y ของการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....101

7.15 องศาด้านหน้าการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....102

7.16 ระยะในแกน X ของการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....103

7.17 ระยะในแกน Y ของการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....103

7.18 องศาด้านหน้าการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....104

สารบัญรูป (ต่อ)

รูปที่ หน้า

7.19 ระยะในแกน X ของการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....105

7.20 ระยะในแกน Y ของการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....105

7.21 องศาด้านหน้าการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....106

7.22 ระยะในแกน X ของการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....107

7.23 ระยะในแกน Y ของการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....107

7.24 องศาด้านหน้าการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 3 ล้อ.....108

7.25 ระยะในแกน X ของการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 4 ล้อ.....109

7.26 ระยะในแกน Y ของการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 4 ล้อ.....110

7.27 องศาด้านหน้าการนำร่องของ FTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 4 ล้อ.....110

7.28 ระยะในแกน X ของการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 4 ล้อ.....111

7.29 ระยะในแกน Y ของการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 4 ล้อ.....112

7.30 องศาด้านหน้าการนำร่องของ FTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 4 ล้อ.....112

7.31 ระยะในแกน X ของการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบ ทิศทาง 4 ล้อ.....113

สารบัญรูป(ต่อ)

รูปที่ หน้า

7.32 ระยะในแกน Y ของการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ.....114
7.33 องศาด้านหน้าการนำร่องของ PTFD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ.....114
7.34 ระยะในแกน X ของการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ.....115
7.35 ระยะในแกน Y ของการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ.....116
7.36 องศาด้านหน้าการนำร่องของ PTPD สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ.....116
7.37 รูปแบบการเล่นของแผน P1301 ในสถานการณ์รุก.....120
7.38 รูปแบบการเล่นของแผน P1301 ในสถานการณ์เก็บบอล.....121
7.40 รูปแบบการเล่นของแผน P1301 ในสถานการณ์ป้องกัน.....122
7.41 รูปแบบการเล่นของแผน P1301 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู.....123
7.42 รูปแบบการเล่นของแผน P1211 ในสถานการณ์รุก.....124
7.43 รูปแบบการเล่นของแผน P1211 ในสถานการณ์เก็บบอล.....125
7.44 รูปแบบการเล่นของแผน P1211 ในสถานการณ์ป้องกัน.....126
7.45 รูปแบบการเล่นของแผน P1211 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู.....127
7.46 รูปแบบการเล่นของแผน P1121 ในสถานการณ์รุก.....128
7.47 รูปแบบการเล่นของแผน P1121 ในสถานการณ์เก็บบอล.....129
7.48 รูปแบบการเล่นของแผน P1121 ในสถานการณ์ป้องกัน.....130
7.49 รูปแบบการเล่นของแผน P1121 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู.....131
7.46 รูปแบบการเล่นของแผน P1031 ในสถานการณ์รุก.....132
7.47 รูปแบบการเล่นของแผน P1031 ในสถานการณ์เก็บบอล.....133
7.48 รูปแบบการเล่นของแผน P1031 ในสถานการณ์ป้องกัน.....134
7.49 รูปแบบการเล่นของแผน P1031 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู.....135
7.50 รูปแบบการเล่นของแผน P0041 ในสถานการณ์รุก.....136

สารบัญรูป(ต่อ)

รูปที่ หน้า

7.51 รูปแบบการเล่นของแผน P0041 ในสถานการณ์เก็บบอล.....137

7.52 รูปแบบการเล่นของแผน P0041 ในสถานการณ์ป้องกัน.....138

7.53 รูปแบบการเล่นของแผน P0041 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู.....139

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

หนึ่งในการทดสอบผลงานการผลิตทรัพยากรบุคคลที่เป็นรูปธรรมด้านหุ่นยนต์เพื่อการศึกษา จะเป็นการทดสอบที่อยู่ในรูปแบบการแข่งขัน ในปัจจุบัน การแข่งขันหุ่นยนต์เล่นฟุตบอลเป็นสิ่งที่ทดสอบประสิทธิภาพการสอนวิชาหุ่นยนต์ศาสตร์ได้ดี [1] โดยสามารถใช้กฎกติกาในการกำหนดสภาพแวดล้อมที่เหมาะสมต่อเทคโนโลยีในปัจจุบันได้ และสามารถปรับเปลี่ยนให้ซับซ้อนได้ตามกาลเวลาและเทคโนโลยีในอนาคต [2] มีแนวโน้มว่าจุดสุดท้ายของการแข่งขันหุ่นยนต์เล่นฟุตบอลจะอยู่ที่การมีการแข่งขันที่ใช้กฎกติกาด้วยกฎเกณฑ์เหมือนมนุษย์โดยขั้นตอนการพัฒนาหุ่นยนต์จะต้องสัมพันธ์กับการออกกฎกติกา โดยแต่ละปีกติกาก็ต้องปรับเปลี่ยนเพื่อเอื้อให้เกิดการพัฒนาหุ่นยนต์ให้มากขึ้น ปัญหาของผู้แข่งขันคือ หากกติกาเปลี่ยนแปลง ระบบต่าง ๆ จะต้องปรับเปลี่ยนตาม การแก้ปัญหาดั้งเดิมในกติกาเก่าอาจจะได้ผลในกติกาใหม่ องค์ประกอบของระบบหุ่นยนต์ที่ใช้ลงแข่งขันจะต้องเปลี่ยนแปลง ความต้องการของผู้พัฒนาคือ การพัฒนาหุ่นยนต์อย่างต่อเนื่อง เป็นสิ่งที่ดีถ้าสิ่งที่ทำในปัจจุบันและเป็นสิ่งที่วางรากฐานต่อระบบในอนาคต โดยไม่ต้องทิ้งสิ่งที่พัฒนามาก่อนแล้ว ซึ่งประเด็นดังกล่าวนี้จึงเป็นหนึ่งในแรงจูงใจที่มีต่อวิทยานิพนธ์นี้ดังจะกล่าวในหัวข้อถัดไป

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

วิทยานิพนธ์ฉบับนี้มุ่งศึกษาและพัฒนาการออกแบบระบบหลากหลายตัวกระทำ (Multi-agent System) บนพื้นฐานระบบแบล็กบอร์ด (Blackboard System) ให้เหมาะสมสำหรับหุ่นยนต์เล่นฟุตบอล (Soccer Robot) ให้มากที่สุด โดยออกแบบระบบให้ให้เหมาะสมกับความต้องการของผู้พัฒนา ซึ่งจะต้องสนับสนุนให้เกิดความสะดวกในการแก้ไขและทดสอบระบบอันเนื่องมาจากการเปลี่ยนแปลงของกติกาและฮาร์ดแวร์ของหุ่นยนต์ ผลลัพธ์ของการวิจัยแสดงโดยการประยุกต์ระบบที่ได้ออกแบบให้สามารถทำงานได้จริงบนระบบจำลองสถานการณ์

1.3 สมมติฐานของการศึกษา

ระบบหลากหลายตัวกระทำบนพื้นฐานระบบแบล็กบอร์ดมีส่วนประกอบที่สำคัญคือตัวกระทำ (Agent) ที่ทำงานเป็นตัวประมวลผลข้อมูลของระบบให้สามารถทำงานตามจุดประสงค์ของ

ผู้พัฒนา โดยระบบหลากหลายตัวกระทำสมัยใหม่จะมีการทำงานอย่างเป็นเอกเทศอย่างแท้จริง ซึ่งทำงานแบบขนานและไม่ต้องอาศัยตัวลำดับการทำงาน ซึ่งในปัจจุบันระบบปฏิบัติการบนคอมพิวเตอร์สนับสนุนการทำงานแบบหลากหลายการทำงาน (Multi-tasking) ซึ่งสามารถแบ่งการทำงานออกเป็นหลายโปรเซสและสามารถทำงานพร้อมกันได้ ดังนั้นตัวกระทำของระบบที่มีคุณสมบัติดังกล่าว สามารถประยุกต์ให้สามารถทำงานบนระบบคอมพิวเตอร์ได้

แบบจำลองหุ่นยนต์เล่นฟุตบอลที่ใช้ในการแข่งขัน จะถูกกำหนดขนาดและความสูงไว้ตามกฎเกณฑ์ แต่ไม่ได้จำกัดรูปแบบการขับเคลื่อนว่าจะใช้รูปแบบไหน ในช่วงเริ่มแรกของการแข่งขันระบบการขับเคลื่อนของหุ่นยนต์นิยมใช้เป็นรูปแบบคล้ายเก้าอี้รถล้อเลื่อนคนพิการซึ่งมี 2 ล้อ แต่ในปัจจุบันระบบขับเคลื่อนได้ถูกพัฒนาเป็นแบบเคลื่อนที่ได้รอบทิศทางที่มีประสิทธิภาพมากกว่า ซึ่งใช้ล้อแบบพิเศษที่ติดตั้งล้อขนาดเล็กไว้ที่ขอบของวงล้อที่หมุนฟรีในแนวข้างได้ โดยระบบขับเคลื่อนดังกล่าวมีทั้งแบบ 3 ล้อและ 4 ล้อ [13]

อัลกอริธึมการนำร่องของหุ่นยนต์มีหลายรูปแบบ ตามแต่ผู้พัฒนาจะนำทฤษฎีใดมาประยุกต์ในการนำร่องหุ่นยนต์เล่นฟุตบอล แม้ว่าจะสามารถนำร่องได้ แต่ประสิทธิภาพในการนำร่องย่อมแตกต่างไปตามจุดประสงค์ เช่น การนำร่องในสถานการณ์ที่จะต้องสกัดการยิงประตูของฝ่ายตรงข้ามจะต้องเน้นที่ความรวดเร็วเป็นสำคัญ ซึ่งแตกต่างจากสถานการณ์การคุมพื้นที่ที่ต้องการความแม่นยำเป็นสำคัญ

นอกเหนือจากความแตกต่างของระบบขับเคลื่อนและการนำร่องแล้ว ในการแข่งขันจะมีการปรับเปลี่ยนแผนการเล่นตามสถานการณ์ แม้ว่าตามกติกา หุ่นยนต์จะต้องทำงานอย่างอัตโนมัติแบบไม่มีการควบคุมโดยมนุษย์ แต่หากผู้ควบคุมระบบในทีมต้องการสั่งการให้ปรับเปลี่ยนรูปแบบการเล่น เสมือนในการแข่งขันฟุตบอลที่ผู้จัดการสามารถสั่งการนักเล่นให้เล่นตามที่ได้สั่งการได้ เช่น การเปลี่ยนแผนการเล่นให้มุ่งเน้นที่การทำประตูมากยิ่งขึ้น เพราะอยู่ในสถานการณ์ที่แต้มน้อยกว่าและกำลังจะหมดเวลาการแข่งขัน เมื่อได้ศึกษาข้อดีและแนวทางการพัฒนาระบบหลากหลายตัวกระทำ ระบบดังกล่าวจะสามารถทำงานควบคุมหุ่นยนต์เล่นฟุตบอลได้และสามารถตอบสนองความต้องการที่ได้กล่าวไว้ในข้างต้นได้

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย

โดยอาศัยการศึกษาทฤษฎีระบบหลากหลายตัวกระทำบนพื้นฐานระบบแบล็กบอร์ดในรูปแบบต่าง ๆ ทั้งโครงสร้างและการประยุกต์ที่ได้นำเสนอจากงานวิจัยมาก่อนหน้านี้ นำมาปรับปรุงแก้ไขให้จำกัดอยู่ที่การควบคุมหุ่นยนต์เล่นฟุตบอล ให้สามารถทำงานได้จริงบนเครื่องคอมพิวเตอร์

อย่างไรก็ตาม เพื่อวัดประสิทธิภาพของตัวกระทำที่อยู่ในระบบ จึงจำเป็นต้องทำการทดลองโดยใช้โปรแกรม Matlab™ ทำการสร้างแบบจำลองขึ้นมาและจำลองสถานการณ์ให้สามารถวัดผล

ได้โดยสะดวก ยกเว้นการวัดผลปัญญาประดิษฐ์ในระบบ ซึ่งจะต้องจำกัดสภาพแวดล้อมและปัจจัยอื่น ๆ จึงจำเป็นจะต้องวัดผลแบบเวลาจริง (Real-time) เท่านั้น

1.5 ขอบเขตการวิจัย

ทำการออกแบบระบบและทดสอบระบบหลากหลายตัวกระทำบนพื้นฐานของระบบแบล็กบอร์ดสำหรับหุ่นยนต์เล่นฟุตบอล โดยการออกแบบจะรวมถึงโครงสร้างของแบล็กบอร์ดและตัวกระทำที่ใช้ในการประมวลผล จากนั้นทำการทดสอบและเปรียบเทียบประสิทธิภาพตัวกระทำตามหน้าที่ที่แตกต่างกัน และใช้เกณฑ์และสถานการณ์ที่เหมาะสมในการทดสอบ โดยมุ่งเน้นที่แสดงแต่ละขั้นตอนและวิธีการพัฒนาระบบโดยรวมด้วยการจำลองผลด้วยตัวจำลอง (Simulator) ที่พัฒนาขึ้นเองบนคอมพิวเตอร์

1.6 ขั้นตอนการศึกษา

มีขั้นตอนการศึกษาดังนี้

1.6.1 ค้นคว้าเอกสารและข้อมูลต่าง ๆ ที่เกี่ยวข้อง

1.6.2 ศึกษาข้อมูลต่าง ๆ

1.6.3 นำข้อมูลที่ได้ศึกษามาออกแบบงานวิจัย

1.6.4 ออกแบบระบบการทดลองและการจำลองสถานการณ์

1.6.5 ทำการทดลอง บันทึกและวัดผล

1.6.6 นำค่าที่ได้จากการทดลองมาแสดงผล และสรุปผลการทดลอง

1.7 โครงสร้างของวิทยานิพนธ์

เพื่อความสะดวกในการนำเสนอ วิทยานิพนธ์นี้มีหัวข้อการนำเสนอโดยสรุปดังนี้ บทที่ 2 อธิบายทฤษฎีระบบหลากหลายตัวกระทำ บทที่ 3 อธิบายหลักการทำงานของระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอล บทที่ 4 กล่าวถึงแผนการเล่นและกลยุทธ์ บทที่ 5 อธิบายรูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอล บทที่ 6 กล่าวถึงการนำร่องสำหรับหุ่นยนต์เล่นฟุตบอล บทที่ 7 แสดงผลการทดลอง และบทสุดท้ายคือบทที่ 8 เป็นการสรุปเนื้อหาของวิทยานิพนธ์และข้อเสนอแนะในอนาคต

บทที่ 2
ทฤษฎีระบบหลากหลายตัวกระทำ

ในบทนี้จะอธิบายทฤษฎี หลักการ ความเป็นมา และพัฒนาการของระบบหลากหลายตัวกระทำ เพื่อแสดงให้เห็นข้อดีในการประยุกต์ระบบปัญญาประดิษฐ์สำหรับหุ่นยนต์เล่นฟุตบอล

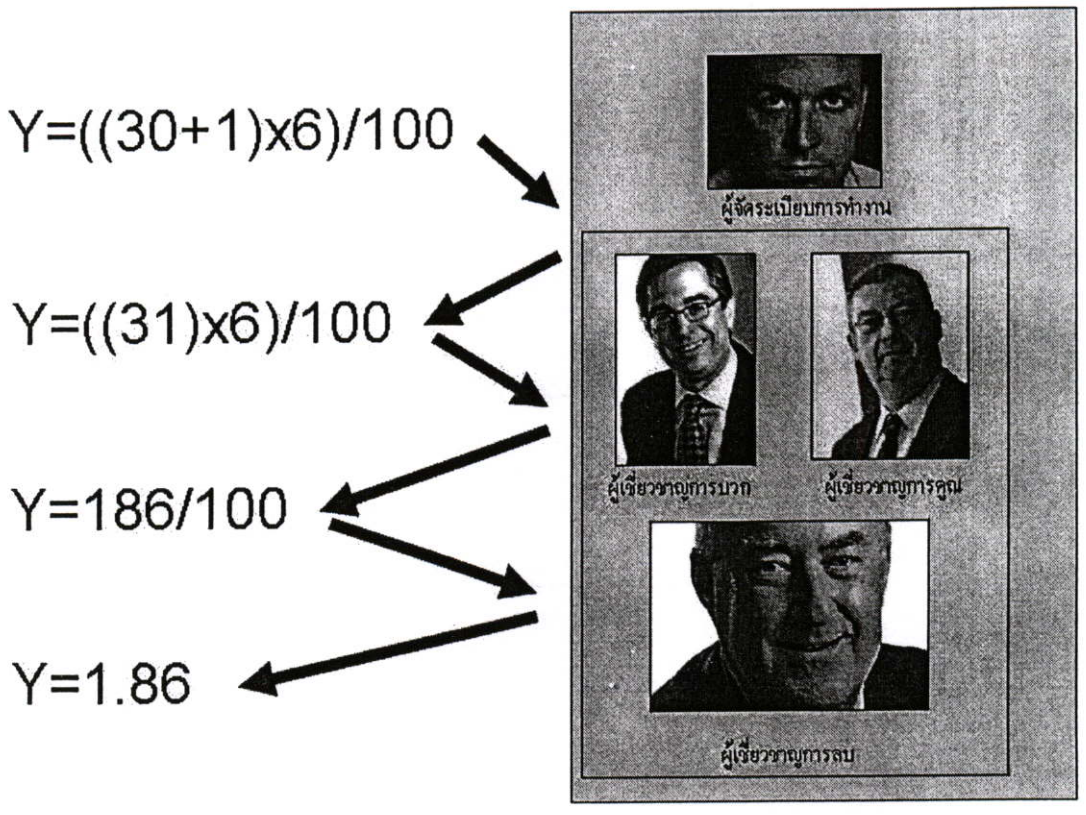
2.1 ทฤษฎีและหลักการพื้นฐานของระบบแบล็กบอร์ด

ระบบแบล็กบอร์ดนั้นดั้งเดิมแล้วจัดเป็นระบบผู้เชี่ยวชาญ (Expert System) ชนิดหนึ่ง ต้นกำเนิดของ ระบบแบล็กบอร์ดมาจากระบบ Hearsay II ซึ่งใช้ในการแก้ปัญหาเกี่ยวกับการเข้าใจเสียงพูด [3] จุดที่น่าสนใจนั้น มิได้อยู่ที่อัลกอริทึมในการแก้ปัญหา ทว่าเป็นโครงสร้างที่ได้รับความสนใจ หลังจากนั้นได้มีการนำไปโครงสร้างนี้ไปวิจัยต่อและประยุกต์กันอย่างแพร่หลาย

โครงสร้างของระบบเป็นเสมือนว่ามีผู้เชี่ยวชาญเฉพาะทางหลาย ๆ คนมาช่วยแก้ปัญหา โดยมีพื้นที่การทำงานอยู่บนกระดานดำ โดยแบ่งปัญหาใหญ่ ๆ ให้เป็นปัญหาย่อยให้ผู้เชี่ยวชาญแก้ปัญหา โดยแต่ละคนจะทำหน้าที่ของตัวเอง โดยการใช้ข้อมูลบนกระดานดำเป็นวัตถุดิบในการแก้ไขปัญหา แล้วจึงเขียนกลับลงไปในกระดานดำ ผลัดกันกระทำตามที่ได้จัดคิวไว้ จนกว่าจะได้คำตอบของโจทย์ปัญหานั้น ๆ

ยกตัวอย่างการแก้โจทย์สมการอย่างง่ายในรูปที่ 2.1 ในการปัญหานี้มีผู้เชี่ยวชาญ 5 คน ได้แก่ ผู้เชี่ยวชาญการบวก ผู้เชี่ยวชาญในการคูณ ผู้เชี่ยวชาญในการหาร ผู้เชี่ยวชาญในการลบ และสุดท้ายคือผู้จัดระเบียบการทำงาน

เมื่อโจทย์ถูกเขียนลงบนกระดาน ผู้จัดระเบียบงานเห็นว่าควรจะให้ผู้เชี่ยวชาญการบวกทำงานคนแรกเพราะตัวเลขที่ทำการบวกกันอยู่ในวงเล็บที่ลึกที่สุด ผู้จัดระเบียบงานจึงให้ผู้เชี่ยวชาญการบวกมาทำงาน ผู้เชี่ยวชาญการบวกจึงคำนวณและเขียนการแก้ปัญหาของตัวเองกลับลงไป เมื่อเขียนเสร็จผู้จัดระเบียบการทำงานจึงผู้เชี่ยวชาญการคูณมาทำงาน ผู้เชี่ยวชาญการคูณก็ได้ทำการแก้ปัญหาและเขียนคำตอบลงไป เมื่อเขียนเสร็จผู้จัดระเบียบการทำงานจึงให้เชี่ยวชาญการหารมาทำงาน เมื่อผู้เชี่ยวชาญการหารแก้ปัญหาและเขียนคำตอบลงไปก็จะได้คำตอบออกมาในท้ายที่สุด

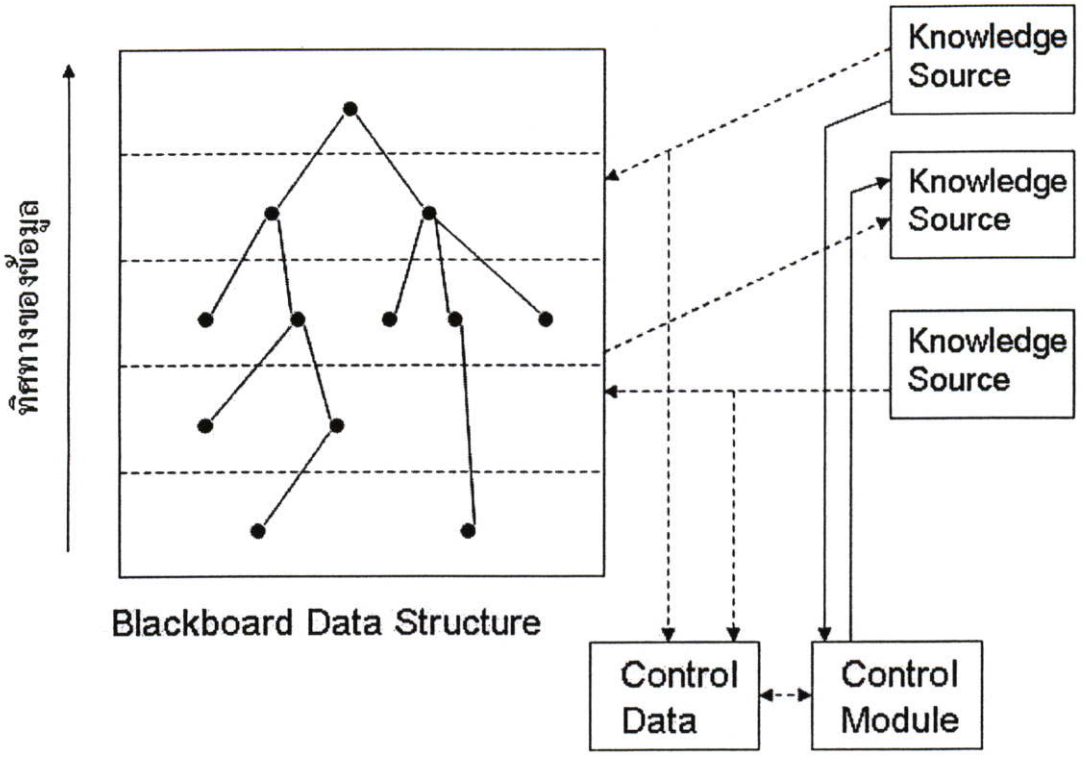


รูปที่ 2.1 รูปตัวอย่างการแก้ปัญหา

สำหรับตัวอย่างในการแก้ปัญหาอีกโจทย์หนึ่ง ซึ่งต้องอาศัยการคำนวณที่มากขึ้น สามารถอธิบายได้อย่างง่ายโดยยกจากตัวอย่างจาก [1] เป็นปัญหาการต่อตัวต่อ (Jigsaw) ซึ่งอาจจัดผู้เชี่ยวชาญในแต่ละหน้าที่ เช่น ผู้เชี่ยวชาญการหาตัวต่อที่เป็นส่วนตรงขอบ ผู้เชี่ยวชาญการแบ่งกลุ่มตัวต่อตามรายละเอียดที่ใกล้เคียงกัน ฯลฯ ซึ่งการแก้ปัญหาจะใช้หลักการแบบเดียวกับการแก้ปัญหาอื่น แต่อาจจะต่างที่กระบวนการและผู้เชี่ยวชาญที่เหมาะสมกับงานที่ต่างกันไป

2.2 โครงสร้างของระบบแบล็กบอร์ด

จุดประสงค์ดั้งเดิมของระบบแบล็กบอร์ดคือ ใช้ในการแก้ปัญหาที่มีความซับซ้อน [1] ข้อมูลในแบล็กบอร์ด (Blackboard) นั้นถูกประมวลผลจากข้อมูลดิบให้กลายเป็นข้อมูลที่เป็นคำตอบของโจทย์นั้น ๆ โดยมีการทำงานลักษณะอนุกรมและมีโครงสร้างดังรูปที่ 2.2



รูปที่ 2.2 โครงสร้างระบบแบล็กบอร์ดแบบดั้งเดิม (Classical Blackboard)

จากรูปจะเห็นได้ว่าส่วนเก็บข้อมูลนั้นเรียกว่าโครงสร้างข้อมูลแบล็กบอร์ด (Blackboard data structure) ซึ่งจะมีโครงเป็นลักษณะหลาย ๆ ชั้น (Layer) โดยตัวมันเองทำหน้าที่เก็บข้อมูลที่ได้จากการประมวลผลจากต้นกำเนิดความรู้ (Knowledge source: KS) โดยเรามักจะทำการประมวลผลข้อมูลจากล่างขึ้นไปบน จนสุดท้ายได้คำตอบการแก้ปัญหาอยู่ชั้นบนสุด แต่ละชั้นจะเห็นได้ว่าชิ้นของข้อมูลจะถูกเชื่อมโยงต่อ ๆ กันเป็นลักษณะความสัมพันธ์

ต้นกำเนิดความรู้ทำหน้าที่เสมือนผู้เชี่ยวชาญที่ใช้ในการแก้ปัญหาเฉพาะด้าน จุดมุ่งหมายของต้นกำเนิดความรู้คือนำข้อมูลที่อยู่ในขอบเขตที่สนใจ มาประมวลผลหรือแก้ปัญหาเฉพาะส่วนที่มันรับผิดชอบ จากนั้นก็นำข้อมูลที่ได้ กลับไปเก็บไว้ในโครงสร้างข้อมูลแบล็กบอร์ดต่อไป ซึ่งการดึงข้อมูลแล้วส่งกลับไปนั้น อาจจะกระทำที่ชั้นเดียวกัน หรือต่างชั้นกันก็ได้ ดังนั้นการกระทำใด ๆ ของต้นกำเนิดความรู้จะเป็นลักษณะโดยอ้อมผ่านทางโครงสร้างข้อมูลแบล็กบอร์ด

ส่วนควบคุม (Control module) เป็นเสมือนผู้ควบคุมการทำงานหรือจัดคิวให้แก่ต้นกำเนิดความรู้โดยดูแลข้อมูลที่อยู่ในโครงสร้างข้อมูลแบล็กบอร์ดเพื่อทำการสั่งการว่าให้ต้นกำเนิดความรู้ตัวไหนทำงานหรือไม่ทำงาน ดังนั้นส่วนควบคุมเป็นตัวจัดให้เกิดการแก้ปัญหาในจุดประสงค์ที่แตกต่างกันได้ สำหรับข้อมูลควบคุม (Control data) นั้นก็คือข้อมูลที่อยู่โครงสร้างข้อมูลแบล็กบอร์ดแต่แยกออกมาเขียนเพื่อให้สะดวกต่อความเข้าใจ

จะเห็นได้ว่าระบบแบล็กบอร์ดทำงานแก้ปัญหาที่ซับซ้อนได้โดยการประมวลผลข้อมูลดิบให้กลายเป็นผลลัพธ์ของปัญหา โดยส่วนควบคุมนั้นสั่งการให้ต้นกำเนิดความรู้ประมวลผลข้อมูลตามจุดประสงค์ที่ได้กำหนดไว้ ซึ่งเหมือนกับที่ได้อธิบายโดยใช้ตัวอย่างปัญหาการต่อตัวต่อด้านบนหรือตัวอย่างการแก้ปัญหาการหาตำแหน่งหมีโคอาล่า [1]

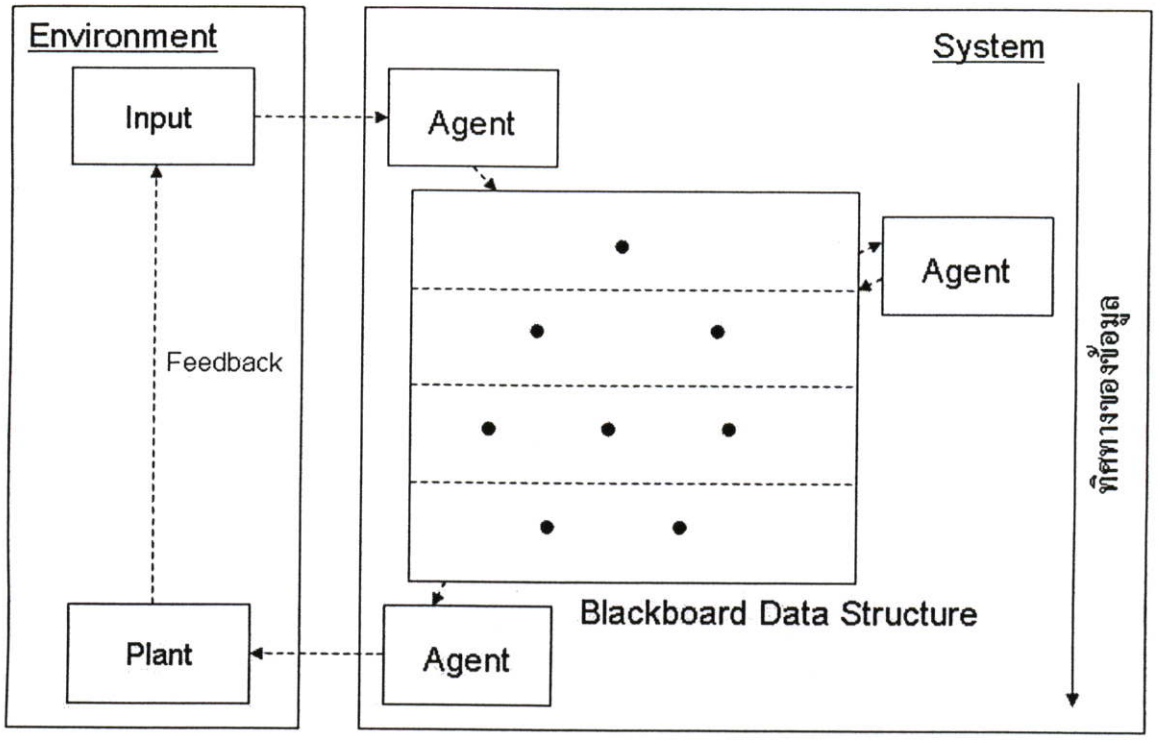
สำหรับโครงสร้างสมัยใหม่ของระบบแบล็กบอร์ดได้ปรากฏในส่วนการวิจัยที่ขยายขอบเขตจากโครงสร้างเดิม โดยจะเป็นลักษณะของการทำงานแบบขนานกันของตัวกระทำต่าง ๆ โดยขจัดส่วนควบคุมออกไป โดยให้ตัวกระทำมีความสามารถที่ทำงานได้อย่างอิสระต่อกัน เรียกว่าว่าโครงสร้างแบบขนาน (Parallel Structure) [4] และนอกจากนี้ในระบบใด ๆ อาจประกอบขึ้นจากระบบแบล็กบอร์ดหลาย ๆ ระบบเข้าด้วยกัน โดยมีการสื่อสารระหว่างแต่ละระบบ เราจะเรียกโครงสร้างแบบนี้ว่าเป็นแบล็กบอร์ดแบบกระจาย (Distributed Blackboard) [5]

2.3 การจัดโครงสร้างระบบแบล็กบอร์ดสำหรับการควบคุม

ในกระบวนการควบคุมสมัยใหม่ ความต้องการของระบบนั้นมีมากกว่าแต่ก่อน ความซับซ้อนที่มากขึ้นทำให้ระบบควบคุมแบบง่ายกลายเป็นระบบควบคุมแบบหลากหลายตัวกระทำซึ่งระบบนั้น ๆ ประกอบขึ้นจากระบบย่อยอีกหลายระบบ ระบบต่าง ๆ ที่เป็นลักษณะอัตโนมัติต้องมีความฉลาดอยู่ในตัว จึงมีการประยุกต์นำระบบแบล็กบอร์ดมาใช้ใน กระบวนการควบคุม [6] ซึ่งจุดประสงค์นั้นแตกต่างจากจุดประสงค์เดิมของระบบแบล็กบอร์ดซึ่งใช้ในการแก้ปัญหาที่ซับซ้อน

สำหรับระบบแบล็กบอร์ดแบบดั้งเดิม ซึ่งจะตั้งชื่อชั้นตามจุดประสงค์ของการนำข้อมูลในชั้นนั้น ๆ ไปใช้งานการลำดับ การเรียงชั้นจะเรียงตามความสัมพันธ์ของข้อมูล ทำให้กระแสของข้อมูลจะวิ่งจากล่างขึ้นสู่บน จึงถือว่าข้อมูลชั้นบนสุดจะถูกนำไปใช้งาน หรืออีกนัยหนึ่งคือจะเรียงลำดับชั้นตามการเข้าใกล้เป้าหมายในการแก้ปัญหา ซึ่งจะเห็นได้ว่าจุดมุ่งหมายของการประมวลผลนั้นเป็นการนำข้อมูลดิบหรือข้อมูลอินพุทนำไปประมวลผลโดยใช้ตัวกระทำให้กลายเป็นข้อมูลที่เป็นคำตอบของปัญหา ซึ่งหากดูจากโครงสร้างดั้งเดิมแล้ว ในชั้นด้านบนสุดจะเป็นส่วนที่เก็บข้อมูลที่เป็นคำตอบและชั้นล่างสุดจะเป็นข้อมูลดิบ โดยชั้นกลาง ๆ นั้นจะเป็นคำตอบของการแก้ปัญหาที่ใช้วัตถุดิบคือข้อมูลจากชั้นล่างกว่า ในขณะเดียวกันนั้นข้อมูลนั้นก็เป็นวัตถุดิบต่อการแก้ปัญหาสำหรับชั้นต่อไป ตามความสัมพันธ์ที่ได้อธิบายไว้ด้านบน

เมื่อจุดประสงค์เปลี่ยนไป ลำดับของกระบวนการย่อมเปลี่ยนไปเช่นกัน สำหรับกระบวนการควบคุมนั้น ข้อมูลที่ต้องการคือข้อมูลที่นำไปป้อนเข้าสู่อินพุทของอุปกรณ์ปฏิบัติ (Actuator) โดยนำข้อมูลอินพุทประมวลผลตามจุดประสงค์ของระบบควบคุมนั้น ๆ จะเห็นได้ว่าโครงสร้างใหม่ดังรูปที่ 2.3 จะเรียงลำดับของชั้นและความสัมพันธ์ของข้อมูลไม่เหมือนกับแบบเดิม กล่าวคือกระบวนการจะเป็นลักษณะที่ประมวลผลจากข้อมูลที่เป็นเป้าหมายให้กลายเป็นข้อมูลดิบ ดังนั้นข้อมูลที่ถูกนำไปใช้งานคือ ข้อมูลที่อยู่ในชั้นล่างสุดและกระแสข้อมูลจะวิ่งจากบนลงสู่ล่าง



รูปที่ 2.3 ระบบแบล็กบอร์ดสำหรับการควบคุม

2.4 การทำงานของตัวกระทำในสภาวะการทำงานจริง

ในการทำงานของระบบแบล็กบอร์ดนั้น การประมวลผลจะเกิดขึ้นได้ต้องอาศัยตัวกระทำเป็นสำคัญ แต่หากต้องการการทำงานที่ถูกต้องจะต้องจัดระเบียบการประมวลผลของตัวกระทำให้สัมพันธ์กัน เพื่อให้ได้ผลลัพธ์ตามจุดประสงค์ของระบบแบล็กบอร์ดนั้น ๆ

2.4.1 การจัดระเบียบการทำงานของตัวกระทำแบบอนุกรม

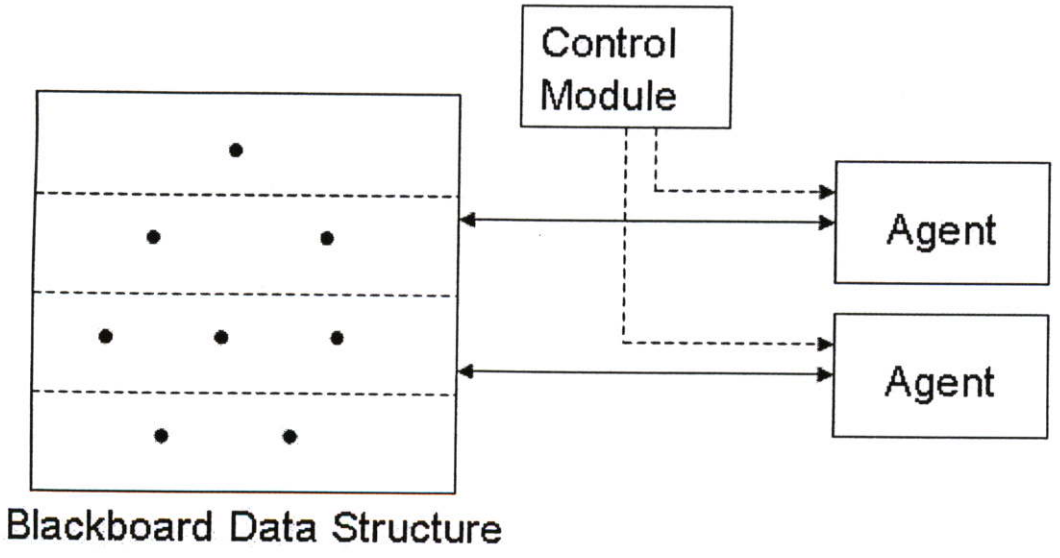
การระเบียบการทำงานแบบอนุกรมเป็นแบบที่ใช้ระบบดั้งเดิม ซึ่งการทำงานของคอมพิวเตอร์ที่ไม่สามารถเขียนโปรแกรมให้มีการประมวลผลพร้อม ๆ กันได้ จึงต้องอาศัยส่วนควบคุมในการจัดลำดับการทำงานของแต่ละตัวกระทำเราสามารถมองได้ว่าส่วนควบคุมเป็นส่วนของโปรแกรมที่ทำให้โปรแกรมทั้งหมดทำงานได้อย่างถูกต้อง ดังเช่นแสดงในรูปที่ 2.5

รูปที่ 2.4 แสดงถึงตัวอย่าง Pseudo Code ของการจัดระเบียบการทำงานของตัวกระทำแบบอนุกรม โดยส่วนควบคุมคือเงื่อนไขต่างๆที่สั่งการตัวกระทำให้ทำงาน ดังนั้นจากระบบตัวอย่าง ในหนึ่งรอบการทำงาน เมื่อมีข้อมูล Data ตัวกระทำที่ 1 (Agent 1) และตัวกระทำที่ 2 (Agent 2) จะทำงาน แต่ถ้าไม่มีข้อมูล Data ตัวกระทำที่ 3 (Agent 3) และตัวกระทำที่ 4 (Agent 4) จะทำงาน และ

เมื่อตัวกระทำทั้งสองทำงานแล้ว ถ้าข้อมูล SubData เท่ากับค่า True ตัวกระทำที่ 5 (Agent 5) จะทำงาน และสุดท้ายเป็นการสั่งให้ชุดคำสั่ง Display ทำงาน

```
Do
        If Data <> Nothing Then
                Agent1.Process
                Agent2.Process
        Else
                Agent4.Process
                Agent3.Process
                If SubData = True Then
                        Agent5.Process
                End if
        End if
        Display (Data)
Loop Until Exit
```

รูปที่ 2.4 ตัวอย่างโปรแกรมแกรมเสมือน (Pseudo Code) ของระบบแบบอนุกรม



รูปที่ 2.5 การทำงานของระบบแบบอนุกรม

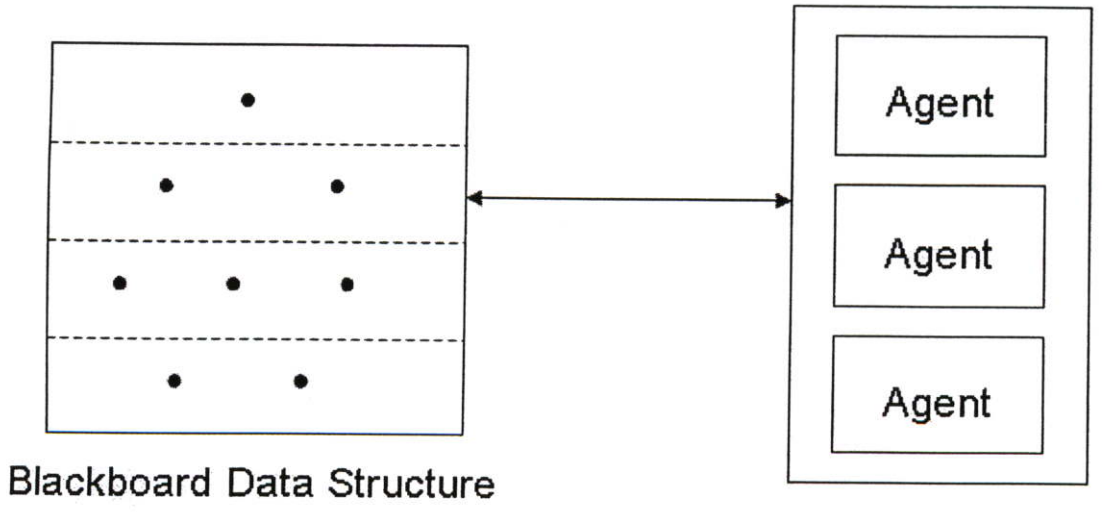
2.4.2 การจัดระเบียบการทำงานของตัวกระทำแบบกึ่งขนานกึ่งอนุกรม

การจัดระเบียบชนิดนี้คล้ายกับแบบอนุกรม แต่ส่วนควบคุมจะไม่มีการโปรแกรมจัดระเบียบการทำงานที่ซับซ้อนอีกต่อไป แต่จะเปลี่ยนความรับผิดชอบการจัดระเบียบจากส่วนควบคุมไปให้ตัวกระทำโดยการแก้ไขให้ตัวกระทำมีความสามารถในการเลือกว่าจะทำงานหรือไม่ทำงานด้วยตัวเองหรือใช้ตัวอื่นช่วย ซึ่งแม้ว่าส่วนควบคุมจะหายไป แต่การทำงานแท้จริงก็ยังเป็นการทำงานแบบอนุกรมอยู่ ดังเช่นแสดงในรูปที่ 2.7

รูปที่ 2.6 แสดงถึงตัวอย่าง Pseudo Code ของการจัดระเบียบการทำงานของตัวกระทำแบบกึ่งขนานกึ่งอนุกรม โดยตัวกระทำที่ 1 (Agent 1) ถึงตัวกระทำที่ 6 (Agent 6) จะทำงานตามลำดับ โดยตัวกระทำทุกตัวจะทำงานโดยตัวใช้ข้อมูลในโครงสร้างข้อมูลแบล็กบอร์ดในการตัดสินใจ

```
Do
        Agent1.Process
        Agent2.Process
        Agent3.Process
        Agent4.Process
        Agent5.Process
        Display (Data)
Loop Until Exit
```

รูปที่ 2.6 ตัวอย่างโปรแกรมเสมือนของระบบกึ่งขนานกึ่งอนุกรม



รูปที่ 2.7 การทำงานของระบบกึ่งขนานกึ่งอนุกรม

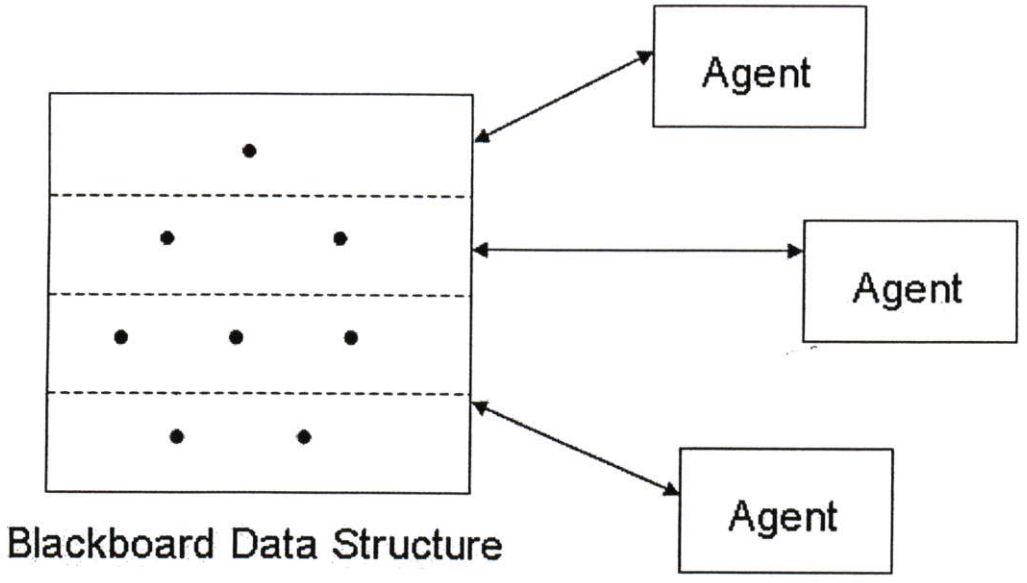
2.4.3 การจัดระเบียบการทำงานของตัวกระทำแบบขนาน

การจัดระเบียบการทำงานแบบขนานนี้เป็นจัดระเบียบที่ให้ตัวกระทำทำงานอย่างเอกเทศโดยสิ้นเชิง ตัวกระทำแต่ละตัวจะทำงานพร้อม ๆ กันอย่างคู่ขนานกันไป โดยความรับผิดชอบในการจัดระเบียบการทำงานขึ้นอยู่ตัวของตัวกระทำเอง โดยคอมพิวเตอร์และภาษาที่ใช้พัฒนานั้นจะต้องสนับสนุนการทำงานแบบขนานหรือการทำงานแบบหลาย ๆ โปรเซสพร้อม ๆ กันด้วย ดังเช่นแสดงในรูปที่ 2.9

รูปที่ 2.8 แสดงถึงตัวอย่าง Pseudo Code ของการจัดระเบียบการทำงานของตัวกระทำแบบขนาน ซึ่งเมื่อสั่งให้ตัวกระทำทำงานแล้ว ตัวกระทำทุกตัวจะทำงานอย่างอัตโนมัติโดยไม่มีการควบคุมโดยส่วนควบคุมเลย

```
Create Agent1
Create Agent2
Create Agent3
Create Agent4
Create Agent5
Agent1.Start
Agent2.Start
Agent3.Start
Agent4.Start
Agent5.Start
```

รูปที่ 2.8 ตัวอย่างโปรแกรมเสมือนของระบบแบบขนาน

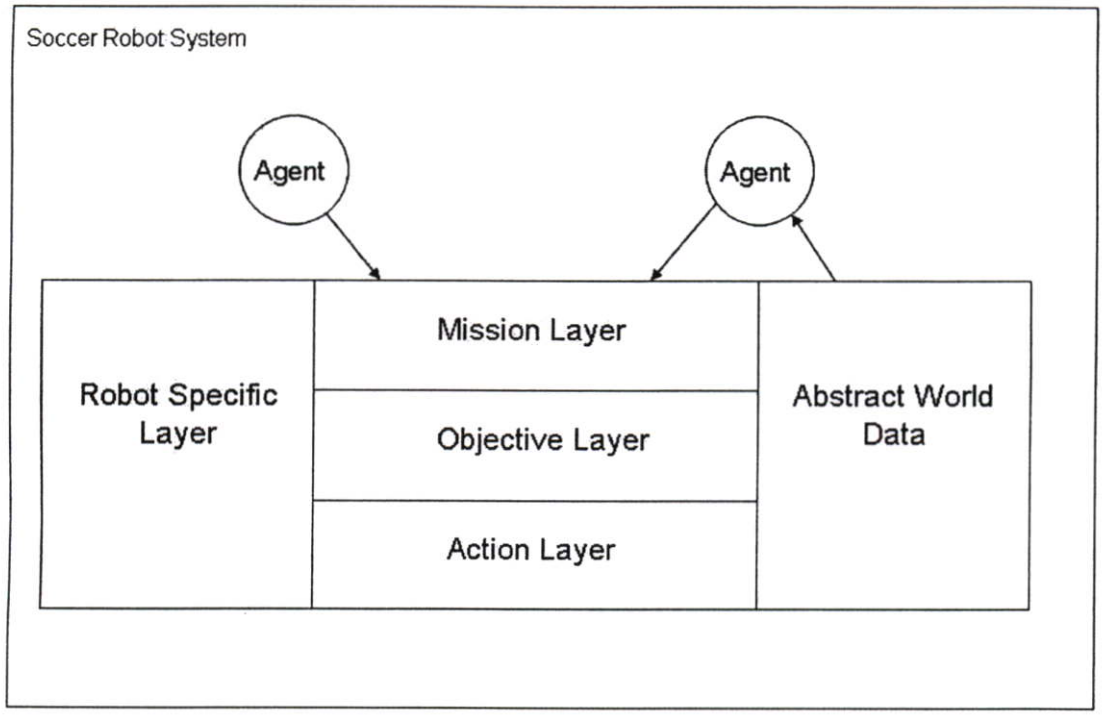


รูปที่ 2.9 การทำงานของระบบแบบขนาน

ในบทที่ 3 ซึ่งเป็นบทต่อไปจะอธิบายการนำระบบหลากหลายตัวกระทำมาประยุกต์ใช้และโครงสร้างใหม่สำหรับหุ่นยนต์เล่นฟุตบอล

บทที่ 3
ระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอล

ในบทนี้จะอธิบายการออกแบบโครงสร้างระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอล โดยได้ใช้แนวคิดการสร้างระบบบนพื้นฐานของระบบแบล็กบอร์ดที่มีอยู่เดิม [2, 3] โดยแต่ละแบล็กบอร์ดนั้นติดต่อสื่อสารเพื่อแลกเปลี่ยนข้อมูลและประมวลผลแบบกระจาย และในแต่ละระบบก็มีการทำงานที่เป็นอิสระต่อกันของตัวกระทำ ซึ่งสามารถพิจารณาดังโครงสร้างที่ [6] เสนอมานั้นเป็นระบบที่ประกอบด้วยสามส่วนหลักคือ Abstract Environment, Expert System และ Low-level control system ซึ่งปฏิสัมพันธ์ต่อกัน หากเปรียบให้เสมือนมนุษย์แล้ว ระบบประกอบด้วยส่วนหน่วยความจำซึ่งเก็บข้อมูลสภาพแวดล้อม, ส่วนสมองซึ่งทำหน้าที่ตัดสินใจ และ อวัยวะซึ่งทำหน้าที่ตามจุดประสงค์ที่ได้กำหนด



รูปที่ 3.1 โครงสร้างระบบหลากหลายตัวกระทำของหุ่นยนต์เล่นฟุตบอล

3.1 โครงสร้างของระบบ

โครงสร้างระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอลในรูปที่ 3.1 เป็นโครงสร้างที่ดัดแปลงจากโครงสร้างที่มีอยู่เดิมโดยระบุชื่อของชั้นและระบุชื่อประเภทของตัวกระทำอย่างชัดเจนเพื่อให้เหมาะสมต่อการพัฒนาหุ่นยนต์เล่นฟุตบอล ซึ่งมีรายละเอียดดังนี้

ระบบหลากหลายตัวกระทำนี้เป็นส่วนกลางในการตัดสินใจของหุ่นยนต์เล่นฟุตบอล ถือว่าเป็นมันสมองของระบบ ทำหน้าที่นำข้อมูลสิ่งแวดล้อมมาประมวลผลตามจุดประสงค์ที่ได้กำหนดให้กลายเป็นข้อมูลสำหรับการควบคุมระดับล่าง โดยประกอบไปด้วยชั้นภารกิจ (Mission Layer), ชั้นจุดประสงค์ (Objective Layer), ชั้นการกระทำ (Action Layer), ชั้นโลกเสมือน (Abstract World Layer) และชั้นคุณสมบัติหุ่นยนต์ (Robot Specific Layer) โดยมีส่วนเพิ่มเติมโครงสร้างเดิม คือมีการแบ่งระดับชั้นในแนวนอนซึ่งหมายถึงความสัมพันธ์ในระดับเดียวกัน

3.1.1 ชั้นโลกเสมือน (Abstract World Layer)

ชั้นโลกเสมือนเป็นชั้นที่มีความสัมพันธ์ต่อชั้นอื่นๆ ทุกชั้นโดยข้อมูลที่อยู่ในส่วนนี้คือข้อมูลสิ่งแวดล้อมที่ถูกแปลงเป็นลักษณะของข้อมูลที่นำไปประมวลผลต่อไปได้ โดยข้อมูลนี้รับข้อมูลจากตัวกระทำอวัยวะรับข้อมูล (Sensory Organ Agent) เพื่อสร้างข้อมูลสภาพแวดล้อม ซึ่งข้อมูลสิ่งแวดล้อมประกอบไปด้วยค่าตัวแปรข้อมูลตำแหน่งหุ่นยนต์ของทั้งฝ่ายตัวเองและฝ่ายตรงข้าม, ตำแหน่งของลูกบอล, ตำแหน่งขอบริม, ตำแหน่งเสาประตู, ฯลฯ

ตารางที่ 3.1 รายละเอียดข้อมูลที่กำหนดในชั้นโลกเสมือน

ข้อมูล	รายละเอียด	กำหนดค่า
BALL_R	รัศมีลูกบอล	22
OBSTACLE_R	รัศมีหุ่นยนต์ฝ่ายตรงข้าม	90
ROBOT_R	รัศมีหุ่นยนต์ฝ่ายเดียวกัน	90
START_X_GROUND	ระยะขอบซ้ายสุดของสนาม	-2750
END_X_GROUND	ระยะขอบขวาสุดของสนาม	2750
START_Y_GROUND	ระยะขอบบนสุดของสนาม	2000
END_Y_GROUND	ระยะขอบล่างสุดของสนาม	-2000
START_X_FIELD	ระยะขอบซ้ายสุดของพื้นที่เล่น	-2750
END_X_FIELD	ระยะขอบขวาสุดของพื้นที่เล่น	2750
START_Y_FIELD	ระยะขอบบนสุดของพื้นที่เล่น	-2000
END_Y_FIELD	ระยะขอบล่างสุดของพื้นที่เล่น	2000
START_X_OPPGOAL	ระยะขอบซ้ายสุดของประตูฝ่ายตรงข้าม	2350
END_X_OPPGOAL	ระยะขอบขวาสุดของประตูฝ่ายตรงข้าม	2450
START_Y_OPPGOAL	ระยะขอบบนสุดของประตูฝ่ายตรงข้าม	350
END_Y_OPPGOAL	ระยะขอบล่างสุดของประตูฝ่ายตรงข้าม	-350

ตารางที่ 3.1(ต่อ) รายละเอียดข้อมูลที่กำหนดในชั้นโลกเสมือน

ข้อมูล	รายละเอียด	กำหนดค่า
START_X_OWNGOAL	ระยะขอบซ้ายสุดของประตูของทีม	-2450
END_X_OWNGOAL	ระยะขอบขวาสุดของประตูของทีม	-2350
START_Y_OWNGOAL	ระยะขอบบนสุดของประตูของทีม	350
START_Y_OWNGOAL	ระยะขอบล่างสุดของประตูของทีม	-350
GOAL_WIDTH	ความกว้างของประตู	180
GOAL_HEIGHT	ความยาวของประตู	700
KICKOFF_P	ตำแหน่งจุดเขี่ยบอลเริ่มเกม	0,0
PENALTY_OUR	ตำแหน่งจุดเล่นลูกโทษของฝ่ายตรงข้าม	2000,0
PENALTY_OWN	ตำแหน่งจุดเล่นลูกโทษของทีม	-2000,0
CENTER_R	รัศมีของวงกลมกลางสนาม	800
SET_PIECE_R	รัศมีของระยะห่างของอีกฝ่ายในขณะเล่นลูกตั้งเล่น	400
TOURCH_R	รัศมีแตะกันระยะระหว่างหุ่นยนต์	180
SAFE_R	รัศมีปลอดภัยระยะระหว่างหุ่นยนต์	270

ตารางที่ 3.2 รายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นโลกเสมือน

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
ROBOT_SIDE	ตัวแปรกำหนดฝ่ายที่ทีมใช้อยู่	[RED,BLUE]
ROBOT_POS_1	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 1	(X,Y,Ø)
ROBOT_POS_2	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 2	(X,Y,Ø)
ROBOT_POS_3	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 3	(X,Y,Ø)
ROBOT_POS_4	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 4	(X,Y,Ø)
ROBOT_POS_5	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 5	(X,Y,Ø)
O_ROBOT_POS_1	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 1 ในโปรเซสก่อนหน้า	(X,Y,Ø)
O_ROBOT_POS_2	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 2 ในโปรเซสก่อนหน้า	(X,Y,Ø)
O_ROBOT_POS_3	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 3 ในโปรเซสก่อนหน้า	(X,Y,Ø)
O_ROBOT_POS_4	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 4 ในโปรเซสก่อนหน้า	(X,Y,Ø)
O_ROBOT_POS_5	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 5 ในโปรเซสก่อนหน้า	(X,Y,Ø)
P_ROBOT_POS_1	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 1 ที่ทำนาย	(X,Y,Ø)

ตารางที่ 3.2(ต่อ) ข้อมูลที่ใช้ในการคำนวณในชั้นโลกเสมือน

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
P_ROBOT_POS_2	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 2 ที่ทำนาย	(X,Y,Ø)
P_ROBOT_POS_3	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 3 ที่ทำนาย	(X,Y,Ø)
P_ROBOT_POS_4	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 4 ที่ทำนาย	(X,Y,Ø)
P_ROBOT_POS_5	ตำแหน่งและองศาของหุ่นยนต์ตัวที่ 5 ที่ทำนาย	(X,Y,Ø)
OPP_POS_1	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 1	(X,Y,Ø)
OPP_POS_2	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 2	(X,Y,Ø)
OPP_POS_3	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 3	(X,Y,Ø)
OPP_POS_4	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 4	(X,Y,Ø)
OPP_POS_5	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 5	(X,Y,Ø)
O_OPP_POS_1	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 1 ใน Process ก่อนหน้า	(X,Y,Ø)
O_OPP_POS_2	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 2 ใน Process ก่อนหน้า	(X,Y,Ø)
O_OPP_POS_3	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 3 ใน Process ก่อนหน้า	(X,Y,Ø)
O_OPP_POS_4	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 4 ใน Process ก่อนหน้า	(X,Y,Ø)
O_OPP_POS_5	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 5 ใน Process ก่อนหน้า	(X,Y,Ø)
P_OPP_POS_1	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 1 ที่ทำนาย	(X,Y,Ø)
P_OPP_POS_2	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 2 ที่ทำนาย	(X,Y,Ø)
P_OPP_POS_3	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 3 ที่ทำนาย	(X,Y,Ø)
P_OPP_POS_4	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 4 ที่ทำนาย	(X,Y,Ø)
P_OPP_POS_5	ตำแหน่งและองศาของคู่ต่อสู้ตัวที่ 5 ที่ทำนาย	(X,Y,Ø)
BALL_POS	ตำแหน่งลูกบอล	(X,Y)
O_BALL_POS	ตำแหน่งลูกบอลใน process ก่อนหน้า	(X,Y)
P_BALL_POS	ตำแหน่งลูกบอลที่ทำนาย	(X,Y)

ตารางที่ 3.1 แสดงรายละเอียดข้อมูลที่กำหนดในชั้นโลกเสมือนและตารางที่ 3.2 แสดงรายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นโลกเสมือน ซึ่งข้อมูลทั้งหมดนี้เก็บและประมวลในหน่วยมิลลิเมตร

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

3.1.2 ชั้นคุณสมบัติหุ่นยนต์ (Robot Specific Layer)

ชั้นคุณสมบัติหุ่นยนต์เป็นชั้นที่มีความสัมพันธ์ระดับเดียวกับชั้นโลกเสมือนเป็นส่วนเก็บข้อมูลที่เป็นข้อมูลเฉพาะของหุ่นยนต์ในระบบซึ่งจำเป็นต่อการประมวลผลในขั้นตอนต่างๆ เช่น มุมระหว่างล้อของหุ่นยนต์ มุมหน้ากว้างของหุ่นยนต์ ฯลฯ โดยตารางที่ 3.3 แสดงรายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นคุณสมบัติหุ่นยนต์

ตารางที่ 3.3 รายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นคุณสมบัติหุ่นยนต์

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
TACTICS	ตัวแปรระบุแผนการเล่น	[P1301, P1211, P1121, P1031, P0041]
DRIVE_TYPE_1	ตัวแปรระบบขับเคลื่อนหุ่นยนต์ตัวที่ 1	[2W,3W,4W]
DRIVE_TYPE_2	ตัวแปรระบบขับเคลื่อนหุ่นยนต์ตัวที่ 2	[2W,3W,4W]
DRIVE_TYPE_3	ตัวแปรระบบขับเคลื่อนหุ่นยนต์ตัวที่ 3	[2W,3W,4W]
DRIVE_TYPE_4	ตัวแปรระบบขับเคลื่อนหุ่นยนต์ตัวที่ 4	[2W,3W,4W]
DRIVE_TYPE_5	ตัวแปรระบบขับเคลื่อนหุ่นยนต์ตัวที่ 5	[2W,3W,4W]
DRIVE_DEGREE_1	มุมองศาระหว่างล้อของหุ่นยนต์ตัวที่ 1	Ø
DRIVE_DEGREE_2	มุมองศาระหว่างล้อของหุ่นยนต์ตัวที่ 2	Ø
DRIVE_DEGREE_3	มุมองศาระหว่างล้อของหุ่นยนต์ตัวที่ 3	Ø
DRIVE_DEGREE_4	มุมองศาระหว่างล้อของหุ่นยนต์ตัวที่ 4	Ø
DRIVE_DEGREE_5	มุมองศาระหว่างล้อของหุ่นยนต์ตัวที่ 5	Ø
FRONT_DEGREE_1	มุมองศาด้านหน้าของหุ่นยนต์ตัวที่ 1	Ø
FRONT_DEGREE_2	มุมองศาด้านหน้าของหุ่นยนต์ตัวที่ 2	Ø
FRONT_DEGREE_3	มุมองศาด้านหน้าของหุ่นยนต์ตัวที่ 3	Ø
FRONT_DEGREE_4	มุมองศาด้านหน้าของหุ่นยนต์ตัวที่ 4	Ø
FRONT_DEGREE_5	มุมองศาด้านหน้าของหุ่นยนต์ตัวที่ 5	Ø

3.1.3 ชั้นภารกิจ (Mission Layer)

ชั้นภารกิจเป็นชั้นบนสุดของการไหลข้อมูล โดยส่วนนี้จะเก็บข้อมูลภารกิจที่ต้องการสั่งการให้หุ่นยนต์เล่นฟุตบอลกระทำ โดยทำหน้าที่รับข้อมูลจากชั้นโลกเสมือนมาประมวลให้กลายเป็นภารกิจที่ต้องกระทำ โดยตารางที่ 3.4 คือรายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นภารกิจ

65468

ตารางที่ 3.4 รายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นภารกิจ

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
PLAYPOS_1	ตำแหน่งการเล่นของหุ่นยนต์ตัวที่ 1	[NONE, CHOSEN, MARKER, COVERER, KEEPER]
PLAYPOS_2	ตำแหน่งการเล่นของหุ่นยนต์ตัวที่ 2	[NONE, CHOSEN, MARKER, COVERER, KEEPER]
PLAYPOS_3	ตำแหน่งการเล่นของหุ่นยนต์ตัวที่ 3	[NONE, CHOSEN, MARKER, COVERER, KEEPER]
PLAYPOS_4	ตำแหน่งการเล่นของหุ่นยนต์ตัวที่ 4	[NONE, CHOSEN, MARKER, COVERER, KEEPER]
PLAYPOS_5	ตำแหน่งการเล่นของหุ่นยนต์ตัวที่ 5	[NONE, CHOSEN, MARKER, COVERER, KEEPER]
PLAY_PRIORITY_1	ลำดับความสำคัญของการเล่นของหุ่นยนต์ตัวที่ 1	[1,2,3,4,5]
PLAY_PRIORITY_2	ลำดับความสำคัญของการเล่นของหุ่นยนต์ตัวที่ 2	[1,2,3,4,5]
PLAY_PRIORITY_3	ลำดับความสำคัญของการเล่นของหุ่นยนต์ตัวที่ 3	[1,2,3,4,5]
PLAY_PRIORITY_4	ลำดับความสำคัญของการเล่นของหุ่นยนต์ตัวที่ 4	[1,2,3,4,5]
PLAY_PRIORITY_5	ลำดับความสำคัญของการเล่นของหุ่นยนต์ตัวที่ 5	[1,2,3,4,5]
GK_USE	มีผู้รักษาประตูในการเล่นหรือไม่	[TRUE,FALSE]
GK_ID	หมายเลขหุ่นยนต์ที่เล่นตำแหน่ง KEEPER	[0,1,2,3,4,5]

ตารางที่ 3.4(ต่อ) รายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นภารกิจ

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
CHOSEN_ID	หมายเลขหุ่นยนต์ที่เล่นตำแหน่ง CHOSEN	[0,1,2,3,4,5]
PLAY_NUM	จำนวนหุ่นยนต์ที่ลงสนาม	[1,2,3,4,5]
MARK_NUM	จำนวนหุ่นยนต์ที่เล่นในตำแหน่ง MARKER	[0,1,2,3,4,5]
COVER_NUM	จำนวนหุ่นยนต์ที่เล่นในตำแหน่ง COVERER	[0,1,2,3,4,5]
IS_PALYPOS_ASSIGNED	มีการประมวลผลตำแหน่งไปหรือยัง	[TRUE, FALSE]
PLAY_STATE	สถานการณ์การเล่น	[NONE, DEFEND, ATTACK, COLLECT, GOALZONE]
O_PLAY_STATE	สถานการณ์การเล่นใน process ก่อนหน้า	[NONE, DEFEND, ATTACK, COLLECT, GOALZONE]

3.1.4 ชั้นจุดประสงค์ (Objective Layer)

ชั้นจุดประสงค์เป็นชั้นที่อยู่ตำแหน่งตรงกลาง เป็นที่เก็บข้อมูลจากการประมวลผลของตัวกระทำที่นำข้อมูลในชั้นภารกิจไปใช้งาน ข้อมูลในส่วนนี้จะเป็นข้อมูลที่แปลงจากข้อมูลเชิงภารกิจให้กลายเป็นข้อมูลเชิงเป้าหมายซึ่งเป็นข้อมูลที่แยกย่อยมากกว่าเดิม ซึ่งข้อมูลนี้จะนำไปประมวลผลโดยตัวกระทำสู่ชั้นการกระทำต่อไป

ตารางที่ 3.5 รายละเอียดข้อมูลที่กำหนดตายตัวอยู่ในชั้นจุดประสงค์

ข้อมูล	รายละเอียด	กำหนดค่า
RANGE_BLOCK	ระยะการบังทางบอล	360
T_LENGHT	ระยะการแย่งบอล	200
CHECK_R	ระยะตรวจสอบการครองบอล	112
CHECK_T	องศาตรวจสอบการครอบครองบอล	60

ตารางที่ 3.6 รายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นจุดประสงค์

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
P_MARK_1	ตำแหน่งประกบของหุ่นยนต์ตัวที่ 1	(X,Y)
P_MARK_2	ตำแหน่งประกบของหุ่นยนต์ตัวที่ 2	(X,Y)
P_MARK_3	ตำแหน่งประกบของหุ่นยนต์ตัวที่ 3	(X,Y)
P_MARK_4	ตำแหน่งประกบของหุ่นยนต์ตัวที่ 4	(X,Y)
P_MARK_5	ตำแหน่งประกบของหุ่นยนต์ตัวที่ 5	(X,Y)
P_COVER_1	ตำแหน่งคุมพื้นที่ของหุ่นยนต์ตัวที่ 1	(X,Y)
P_COVER_2	ตำแหน่งคุมพื้นที่ของหุ่นยนต์ตัวที่ 2	(X,Y)
P_COVER_3	ตำแหน่งคุมพื้นที่ของหุ่นยนต์ตัวที่ 3	(X,Y)
P_COVER_4	ตำแหน่งคุมพื้นที่ของหุ่นยนต์ตัวที่ 4	(X,Y)
P_COVER_5	ตำแหน่งคุมพื้นที่ของหุ่นยนต์ตัวที่ 5	(X,Y)
OBSTACLE_1	ตำแหน่งของสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 1	(X1,Y1) (X2,Y2) (X3,Y3) (X4,Y4) (X5,Y5) (X6,Y6) (X7,Y7) (X8,Y8) (X9,Y9)
OBSTACLE_2	ตำแหน่งของสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 2	(X1,Y1) (X2,Y2) (X3,Y3) (X4,Y4) (X5,Y5) (X6,Y6) (X7,Y7) (X8,Y8) (X9,Y9)
OBSTACLE_3	ตำแหน่งของสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 3	(X1,Y1) (X2,Y2) (X3,Y3) (X4,Y4) (X5,Y5) (X6,Y6) (X7,Y7) (X8,Y8) (X9,Y9)
OBSTACLE_4	ตำแหน่งของสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 4	(X1,Y1) (X2,Y2) (X3,Y3) (X4,Y4) (X5,Y5) (X6,Y6) (X7,Y7) (X8,Y8) (X9,Y9)
OBSTACLE_5	ตำแหน่งของสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 5	(X1,Y1) (X2,Y2) (X3,Y3) (X4,Y4) (X5,Y5) (X6,Y6) (X7,Y7) (X8,Y8) (X9,Y9)
OBSTACLE_COUNT_1	จำนวนสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 1	[1,2,3,4,5,6,7,8,9]
OBSTACLE_COUNT_2	จำนวนสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 2	[1,2,3,4,5,6,7,8,9]

ตารางที่ 3.6(ต่อ) ข้อมูลที่ใช้ในการคำนวณในชั้นจุดประสงค์

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
OBSTACLE_COUNT_3	จำนวนสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 3	[1,2,3,4,5,6,7,8,9]
OBSTACLE_COUNT_4	จำนวนสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 4	[1,2,3,4,5,6,7,8,9]
OBSTACLE_COUNT_5	จำนวนสิ่งกีดขวางเมื่อมองจากมุมมองของหุ่นยนต์ตัวที่ 5	[1,2,3,4,5,6,7,8,9]
PATH_1	ตำแหน่งจุดวิ่งของหุ่นยนต์ตัวที่ 1	(X,Y)
PATH_2	ตำแหน่งจุดวิ่งของหุ่นยนต์ตัวที่ 2	(X,Y)
PATH_3	ตำแหน่งจุดวิ่งของหุ่นยนต์ตัวที่ 3	(X,Y)
PATH_4	ตำแหน่งจุดวิ่งของหุ่นยนต์ตัวที่ 4	(X,Y)
PATH_5	ตำแหน่งจุดวิ่งของหุ่นยนต์ตัวที่ 5	(X,Y)
PATHTYPE_1	ชนิดจุดวิ่งของหุ่นยนต์ตัวที่ 1	[NORMAL, STOP, DRIB, GOAL, PREPARE, BALL, PASS, ENEMY, TOUCH]
PATHTYPE_2	ชนิดจุดวิ่งของหุ่นยนต์ตัวที่ 2	[NORMAL, STOP, DRIB, GOAL, PREPARE, BALL, PASS, ENEMY, TOUCH]
PATHTYPE_3	ชนิดจุดวิ่งของหุ่นยนต์ตัวที่ 3	[NORMAL, STOP, DRIB, GOAL, PREPARE, BALL, PASS, ENEMY, TOUCH]
PATHTYPE_4	ชนิดจุดวิ่งของหุ่นยนต์ตัวที่ 4	[NORMAL, STOP, DRIB, GOAL, PREPARE, BALL, PASS, ENEMY, TOUCH]

ตารางที่ 3.6(ต่อ) ข้อมูลที่ใช้ในการคำนวณในชั้นจุดประสงค์

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
PATHTYPE_5	ชนิดจุดวิ่งของหุ่นยนต์ตัวที่ 5	[NORMAL, STOP, DRIB, GOAL, PREPARE, BALL, PASS, ENEMY, TOUCH]

ตารางที่ 3.5 แสดงรายละเอียดข้อมูลที่กำหนดตายตัวอยู่ในชั้นจุดประสงค์ และตารางที่ 3.6 แสดงรายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นจุดประสงค์

3.1.5 ชั้นการกระทำ (Action Layer)

ชั้นการกระทำเป็นชั้นตำแหน่งล่างสุด ข้อมูลในชั้นนี้จึงย่อยรายละเอียดให้มากที่สุดเพื่อจะได้สามารถนำไปใช้สั่งการในทันที โดยได้รับข้อมูลจากการประมวลผลของตัวกระทำที่ใช้ข้อมูลจากชั้นจุดประสงค์เป็นหลัก ซึ่งตารางที่ 3.7 แสดงรายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นการกระทำ

ตารางที่ 3.7 รายละเอียดข้อมูลที่ใช้ในการคำนวณในชั้นการกระทำ

ข้อมูล	รายละเอียด	รูปแบบข้อมูล
ROBOT_DRIVE_1	ความเร็วล้อของหุ่นยนต์ตัวที่ 1	(W1, W2, W3, W4)
ROBOT_DRIVE_2	ความเร็วล้อของหุ่นยนต์ตัวที่ 2	(W1, W2, W3, W4)
ROBOT_DRIVE_3	ความเร็วล้อของหุ่นยนต์ตัวที่ 3	(W1, W2, W3, W4)
ROBOT_DRIVE_4	ความเร็วล้อของหุ่นยนต์ตัวที่ 4	(W1, W2, W3, W4)
ROBOT_DRIVE_5	ความเร็วล้อของหุ่นยนต์ตัวที่ 5	(W1, W2, W3, W4)

3.2 รายละเอียดของผู้เชี่ยวชาญ

หน้าที่ของตัวกระทำนั้นคือ การประมวลผลข้อมูลตามหน้าที่ที่ได้กำหนดไว้แล้ว โดยระบบจะทำงานตามจุดประสงค์ก็ต่อเมื่อตัวกระทำนั้นสามารถทำงานประสานได้อย่างถูกต้อง จะเห็นได้ว่าหุ่นยนต์เล่นฟุตบอลนั้นได้เกี่ยวข้องกับการวิจัยหลายๆ การวิจัยที่เกี่ยวข้องนั้น [8-10] [12] สามารถนำมาใช้ให้กลายเป็นตัวกระทำเพื่อใช้ในโครงสร้างทางซอฟแวร์นี้ได้ทั้งหมด สามารถแบ่งประเภทของตัวกระทำได้ดังนี้

3.2.1 ตัวกระทำอวัยวะ (Organ Agent)

ตัวกระทำอวัยวะทำหน้าที่ติดต่อกับฮาร์ดแวร์ของหุ่นยนต์เล่นฟุตบอลหรือชุดคำสั่งที่แลกเปลี่ยนข้อมูลที่อยู่ภายนอกของเครื่องคอมพิวเตอร์ โดยทำหน้าที่เสมือนอวัยวะของมนุษย์ ซึ่งแลกเปลี่ยนข้อมูลกับสมอง โดยแบ่งประเภทได้เป็น 2 ประเภทคือ ตัวกระทำอวัยวะรับข้อมูล (Sensory Organ Agent) และ ตัวกระทำอวัยวะดำเนินการ (Actuating Organ Agent)

ตัวกระทำอวัยวะรับข้อมูล (Sensory Organ Agent) ทำหน้าที่เป็นรับข้อมูลจากสิ่งแวดล้อมจริง มาประมวลผลให้ได้ข้อมูลที่สามารถนำไปใช้ประมวลผลได้ต่อไป เปรียบเสมือนประสาทสัมผัสของระบบปัญญาประดิษฐ์ของหุ่นยนต์เล่นฟุตบอลซึ่งทำหน้าที่ประมวลผลข้อมูลที่ได้จากอุปกรณ์รับข้อมูลและการติดต่อกับส่วนฮาร์ดแวร์จะเป็นลักษณะทางเดียว คือรับข้อมูลจากฮาร์ดแวร์เท่านั้น

ตัวกระทำอวัยวะดำเนินการ (Actuating Organ Agent) ทำหน้าที่รับข้อมูลจากชั้นการกระทำมาใช้ในการออกคำสั่งให้หุ่นยนต์เล่นฟุตบอลกระทำการในทางกายภาพ เสมือนอวัยวะที่กระทำทางกายภาพต่อสิ่งแวดล้อม โดยในส่วนของตัวกระทำนี้จะเป็นลักษณะควบคุมอุปกรณ์ของหุ่นยนต์เล่นฟุตบอล ดังนั้นอาจจะพูดได้ว่าอัลกอริทึมที่ใช้ในการควบคุมทั้งหมดนั้นสามารถประยุกต์ให้กลายเป็นตัวกระทำนี้ได้ทั้งสิ้น ส่วนการติดต่อกับส่วนฮาร์ดแวร์นั้นจะเป็นลักษณะส่งข้อมูลออกไป

การแบ่งประเภทตัวกระทำอวัยวะดำเนินการกับตัวกระทำอวัยวะรับข้อมูลนั้น มิได้จำกัดอยู่ที่ว่าตัวกระทำนั้นติดต่อฮาร์ดแวร์ที่เป็นแบบใด แต่อยู่ที่ว่าการแลกเปลี่ยนข้อมูลกับโครงสร้างข้อมูลเป็นลักษณะแบบใด โดยตารางที่ 3.8 แสดงชื่อของตัวกระทำอวัยวะ

ตารางที่ 3.8 รายชื่อตัวกระทำอวัยวะ

ข้อมูล	รายละเอียด	ชนิด
VISIONPORT	แปลงข้อมูลจากเครือข่ายเป็นข้อมูลตำแหน่งและประมวลผลตำแหน่งในอนาคต	Sensory Organ Agent
ROBOTPORT	แปลงข้อมูลจากชั้นการกระทำส่งออกไป	Actuating Organ Agent

3.2.2 ตัวกระทำปรับข้อมูล (Normalized Agent)

ตัวกระทำปรับข้อมูลเป็นตัวกระทำที่ประมวลผลปรับปรุงข้อมูลให้อยู่ในรูปแบบที่เหมาะสมและถูกต้อง เพื่อเตรียมข้อมูลให้ตัวกระทำอื่นๆ นำข้อมูลที่ได้ไปประมวลผลได้โดยไม่ให้เกิดข้อผิดพลาด โดยตัวกระทำชนิดนี้ไม่ได้จำกัดว่าอยู่ชั้นไหน แต่สำหรับในบางตัวกระทำจะมี

หน้าที่ที่ซับซ้อนมากกว่า เช่น หากนำอัลกอริธึมที่เป็นการหลบหลีกสิ่งกีดขวาง [7] มาใช้เป็นตัวกระทำโดยให้ประมวลผลเส้นทางที่ปลอดภัยหลังจากกำหนดจุดเป้าหมายแล้วและตัวกระทำที่เพิ่มค่าความเร็วให้เหมาะสมตามสภาพของพลังงานใน แบตเตอรี่โดยแก้ไขค่าความเร็วก่อนที่จะถูกส่งไปยังตัวกระทำดำเนินการที่เป็นล้อ ก็ถือว่าเป็นตัวกระทำชนิดนี้เช่นกัน ตารางที่ 3.9 แสดงชื่อของตัวกระทำปรับข้อมูลที่ใช้ในระบบหุ่นยนต์เล่นฟุตบอล

ตารางที่ 3.9 รายชื่อตัวกระทำปรับข้อมูล

ข้อมูล	รายละเอียด	ชั้นที่ทำงาน
REDUCE_WORLD	แก้ไขข้อมูลที่ผิดพลาดในชั้น Abstract World Data	Abstract World
REDUCE_PATH	แก้ไขตำแหน่งการวิ่ง ให้วิ่งอ้อมสิ่งกีดขวาง	Objective Layer
REDUCE_DRIVE	แก้ไขข้อมูลความเร็วล้อ ให้มีค่าไม่เกินที่กำหนดไว้	Action Layer

3.2.3 ตัวกระทำวางแผน (Strategic Agent)

ตัวกระทำวางแผนเป็นตัวกระทำวิเคราะห์ข้อมูลต่างๆ ให้กลายเป็นภารกิจที่ระบบจะต้องกระทำ โดยข้อมูลที่จะต้องวิเคราะห์จะเป็นข้อมูลดิบที่อยู่ทั่วไปในชั้นต่างๆ ที่นำมาสรุปว่า ตอนนี้ระบบควรจะทำอะไรต่อไป โดยกำหนดอยู่ในรูปแบบตัวแปรของภาษาพูดอย่างชัดเจน ซึ่งตารางที่ 3.10 แสดงรายชื่อตัวกระทำวางแผน

ตารางที่ 3.10 รายชื่อตัวกระทำวางแผน

ข้อมูล	รายละเอียด	ชั้นที่ทำงาน
PLAYPOS_ASSIGN	คำนวณตำแหน่งการเล่นให้หุ่นยนต์แต่ละตัว	Robot Specific, Abstract World, Mission
PLAYSTATE_ASSIGN	คำนวณสถานการณ์ของทีม	Abstract World, Mission

3.2.4 ตัวกระทำข้อมูลจากภารกิจสู่จุดประสงค์ (Mission to Objective Agent)

ตัวกระทำนี้ทำหน้าที่ประมวลผลข้อมูลเชิงภารกิจที่อยู่ในชั้นภารกิจให้กลายเป็นข้อมูลเชิงเป้าหมายส่งไปที่ชั้นจุดประสงค์ สำหรับข้อมูลภารกิจนั้นจะประกอบไปด้วยเป้าหมายย่อยๆ มากหมาย ดังนั้นการกระทำภารกิจที่กำหนดนั้นให้สำเร็จตามเป้าหมายได้ จะต้องทำเป้าหมายที่ย่อยๆ ที่เป็นส่วนประกอบของภารกิจนั้นก่อน ยกตัวอย่างเช่น ภารกิจการวิ่งทดสอบสนามนั้น จะ

ประกอบด้วยข้อมูลเป้าหมาย ดังนั้นภารกิจการวิ่งทดสอบสนามนั้นก็คือการวิ่งไปตามเป้าหมายทั้งหมดตามลำดับนั้นเอง กล่าวได้ว่าอัลกอริธึมที่เป็นลักษณะการวางแผน [8, 9, 10, 11] นั้นและพฤติกรรม [12, 13] สามารถประยุกต์ให้กลายเป็นตัวกระทำชนิดนี้ได้ทั้งสิ้น

ตารางที่ 3.11 รายชื่อตัวกระทำข้อมูลจากภารกิจสู่จุดประสงค์

ข้อมูล	รายละเอียด	ชั้นที่ทำงาน
COVER_PLAY	คำนวณตำแหน่งการคุมพื้นที่ให้กับหุ่นยนต์ตำแหน่ง COVERER เมื่อ PLAY_STATE ไม่เท่ากับ GOALZONE	Robot Specific, Abstract World, Mission, Objective
UNCOVER_PLAY	คำนวณตำแหน่งการคุมพื้นที่ให้กับหุ่นยนต์ตำแหน่ง COVERER เมื่อ PLAY_STATE เท่ากับ GOALZONE	Robot Specific, Abstract World, Mission, Objective
MAN2MAN_PLAY	คำนวณตำแหน่งการประกบที่ให้กับหุ่นยนต์ตำแหน่ง MARKER เมื่อ PLAY_STATE เท่ากับ ATTACK หรือ COLLECT	Robot Specific, Abstract World, Mission, Objective
PRESSING_PLAY	คำนวณตำแหน่งการประกบที่ให้กับหุ่นยนต์ตำแหน่ง MARKER เมื่อ PLAY_STATE เท่ากับ DEFEND	Robot Specific, Abstract World, Mission, Objective
SCREEN_PLAY	คำนวณตำแหน่งการประกบที่ให้กับหุ่นยนต์ตำแหน่ง MARKER เมื่อ PLAY_STATE เท่ากับ GOALZONE	Robot Specific, Abstract World, Mission, Objective
CHOSEN_PLAY	คำนวณเส้นทางการวิ่งของหุ่นยนต์ตำแหน่ง CHOSEN	Robot Specific, Abstract World, Mission, Objective
COVERER_PLAY	คำนวณเส้นทางการวิ่งของหุ่นยนต์ตำแหน่ง COVERER	Robot Specific, Abstract World, Mission, Objective
MARKER_PLAY	คำนวณเส้นทางการวิ่งของหุ่นยนต์ตำแหน่ง MARKER	Robot Specific, Abstract World, Mission, Objective
GK_PLAY	คำนวณเส้นทางการวิ่งของหุ่นยนต์ตำแหน่ง KEEPER	Robot Specific, Abstract World, Mission, Objective

ตารางที่ 3.11 แสดงรายละเอียดของรายชื่อตัวกระทำข้อมูลจากภารกิจสู่จุดประสงค์ทั้งหมดที่ใช้ในระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอล

3.2.5 ตัวกระทำข้อมูลจากจุดประสงค์สู่การกระทำ (Objective to Action Agent)

ตัวกระทำข้อมูลจากจุดประสงค์สู่การกระทำ (Objective to Action Agent) เป็นตัวกระทำที่ทำหน้าที่แปลงข้อมูลเชิงเป้าหมายที่อยู่ในชั้นจุดประสงค์ให้กลายเป็นข้อมูลเชิงปฏิบัติในชั้นการกระทำซึ่งเป็นการประมวลต่อเนื่องจากตัวกระทำข้อมูลภารกิจสู่จุดประสงค์ซึ่งการประมวลผลโดยตัวกระทำชนิดนี้จะคล้ายกับการทำงานตัวกระทำข้อมูลภารกิจสู่จุดประสงค์แต่ต่างกันตรงที่ชนิดของข้อมูลที่แปลงนั้นเป็นคนละระดับกัน ซึ่งข้อมูลที่เป็นลักษณะเป้าหมายนั้น ยังไม่สามารถใช้ไปสั่งการในระดับล่างได้ ดังนั้นการแปลงให้กลายเป็นข้อมูลเชิงปฏิบัติ ยกตัวอย่างเช่น การนำร่องหุ่นยนต์ จะต้องทำการแปลงข้อมูลเป้าหมายให้กลายเป็นความเร็วเชิงมุมของล้อ เพื่อส่งไปให้ตัวกระทำดำเนินการที่ควบคุมล้อของของหุ่นยนต์เล่นฟุตบอลติดต่อกับฮาร์ดแวร์อีกที กล่าวได้ว่าการใช้อัลกอริธึมที่เป็นการนำร่อง [14,15] นั้น ต้องทำให้กลายเป็นตัวกระทำชนิดนี้

ตารางที่ 3.12 รายชื่อตัวกระทำข้อมูลจากจุดประสงค์สู่การกระทำ

ข้อมูล	รายละเอียด	ชั้นที่ทำงาน
FF_NAVI	นำร่องหุ่นยนต์โดยใช้ Fuzzy คำนวณความเร็วและทิศทาง	Robot Specific, Abstract World, Objective
FP_NAVI	นำร่องหุ่นยนต์โดยใช้ Fuzzy คำนวณความเร็วและใช้ PID คำนวณทิศทาง	Robot Specific, Abstract World, Objective
PF_NAVI	นำร่องหุ่นยนต์โดยใช้ PID คำนวณความเร็วและใช้ Fuzzy คำนวณทิศทาง	Robot Specific, Abstract World, Objective
PP_NAVI	นำร่องหุ่นยนต์โดยใช้ระบบ PID คำนวณความเร็วและทิศทาง	Robot Specific, Abstract World, Objective

* หมายเหตุ: Fuzzy = วิธีการแบบตรรกศาสตร์ฟัซซี

PID = Proportional-Integral-Derivative

ในบทที่ 4 ซึ่งเป็นบทต่อไปจะอธิบายแผนการเล่นและกลยุทธ์ทั้งหมดที่ถูกบรรจุไว้ในตัวกระทำวางแผน โดยประกอบด้วยการคำนวณตำแหน่งการเล่นของหุ่นยนต์เล่นฟุตบอลและการคำนวณสถานการณ์การเล่นของทีม

บทที่ 4
แผนการเล่นและกลยุทธ์

ในบทนี้จะอธิบายแผนการเล่นและกลยุทธ์ที่บรรจุไว้ในตัวกระทำวางแผน ซึ่งได้แก่ PLAYSTATE_ASSIGN ซึ่งเป็นตัวกระทำที่หน้าที่คำนวณสถานการณ์การล่นของทีม และ PLAYPOS_ASSIGN ซึ่งเป็นตัวกระทำที่ทำหน้าที่คำนวณตำแหน่งการเล่นให้กับหุ่นยนต์เล่นฟุตบอล

4.1 การคำนวณตำแหน่งการเล่นของหุ่นยนต์เล่นฟุตบอล

การคำนวณนี้มีจุดประสงค์เพื่อระบุตำแหน่งการเล่นให้กับหุ่นยนต์เล่นฟุตบอลในสถานการณ์ต่างๆ โดยจะอิงอยู่กับข้อมูล TACTICS ที่อยู่ในชั้นคุณสมบัติหุ่นยนต์ ซึ่งระบุการวางแผนตำแหน่งการเล่น โดยมีรายละเอียดดังนี้

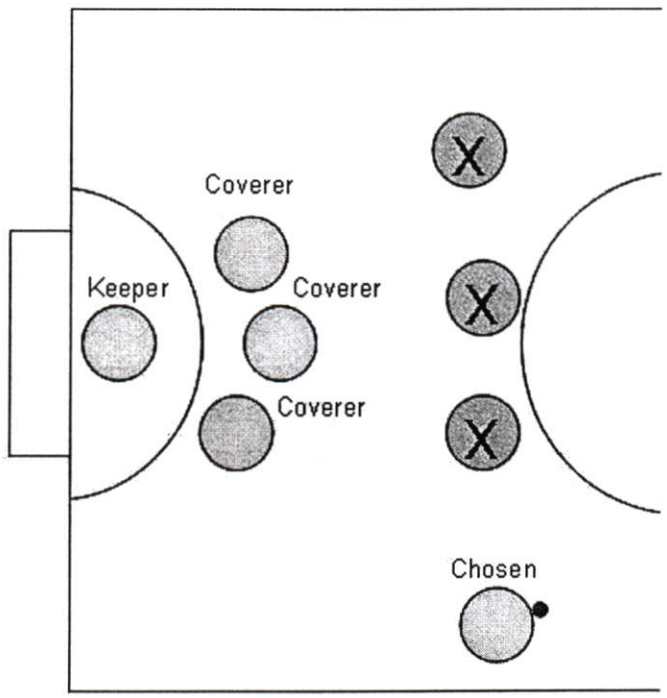
ตารางที่ 4.1 รายละเอียดของข้อมูล TACTICS ในคุณสมบัติหุ่นยนต์

ข้อมูลที่ถูกกำหนด	รายละเอียด	Keeper	Coverer	Marker	Chosen
P1301	เน้นที่ตั้งรับ ไม่ประกบ ไล่บอลอยู่เพียงตัวเดียว	1	3	0	1
P1211	เน้นที่บังการยิงมากกว่าการประกบ	1	2	1	1
P1121	เน้นที่การประกบมากกว่าบังการยิง	1	1	2	1
P1031	เน้นที่การประกบอย่างเดียว	1	0	3	1
P0041	ไม่มีผู้รักษาประตูและบังการยิง เน้นที่การประกบและรุกอย่างเดียว	0	0	4	1

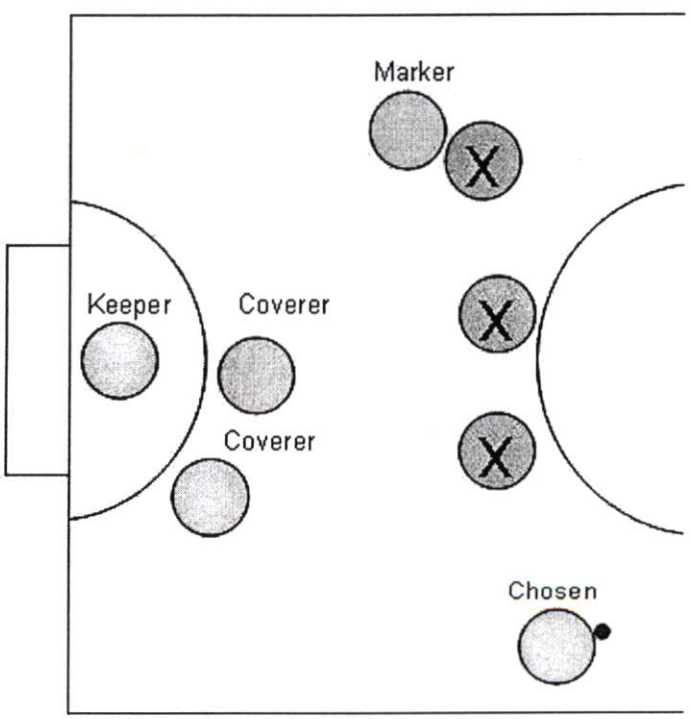
โดยรูปที่ 4.1(ก) แสดงตัวอย่างตำแหน่งการเล่นของแผน P1301 รูปที่ 4.1(ข) แสดงตัวอย่างตำแหน่งการเล่นของแผน P1211 รูปที่ 4.1(ค) แสดงตัวอย่างตำแหน่งการเล่นของแผน P1121 รูปที่ 4.1(ง) แสดงตัวอย่างตำแหน่งการเล่นของแผน P1031 และรูปที่ 4.1(จ) แสดงตัวอย่างตำแหน่งการเล่นของแผน P0041

การระบุว่าหุ่นยนต์ตัวไหนควรจะเล่นตำแหน่งใด ให้ดูที่ระยะห่างของลูกฟุตบอลกับหุ่นยนต์ ตัวที่อยู่ใกล้ลูกฟุตบอลที่สุดจะถูกระบุให้เล่นตำแหน่งตัวไล่บอลและรุก (Chosen) หุ่นยนต์ตัวที่ระยะห่างมากกว่าจะถูกระบุเป็นตำแหน่งตัวประกบ (Marker) ตำแหน่งตัวบังการยิง (Coverer)

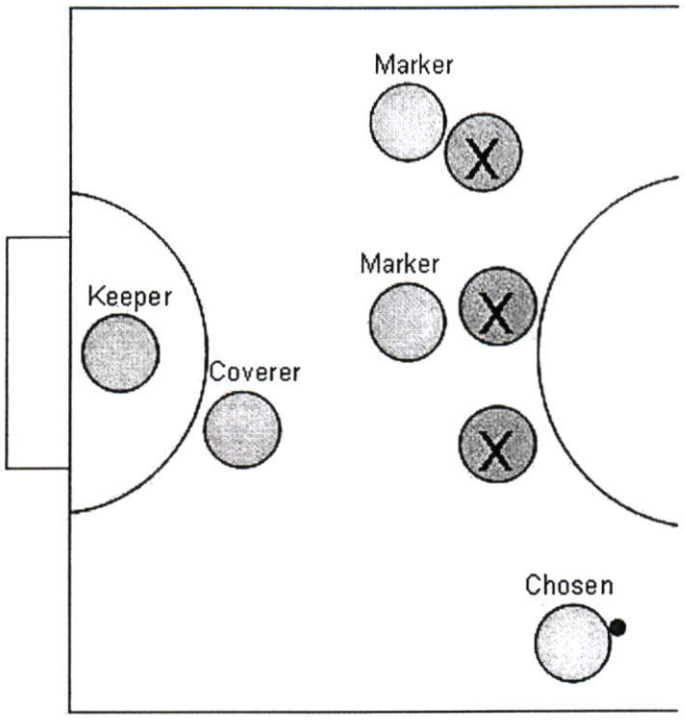
และสุดท้ายคือตำแหน่งผู้รักษาประตู (Keeper) ซึ่งการเปลี่ยนตำแหน่งจะเกิดขึ้นต่อเมื่อมีการเปลี่ยนสถานการณ์เท่านั้น



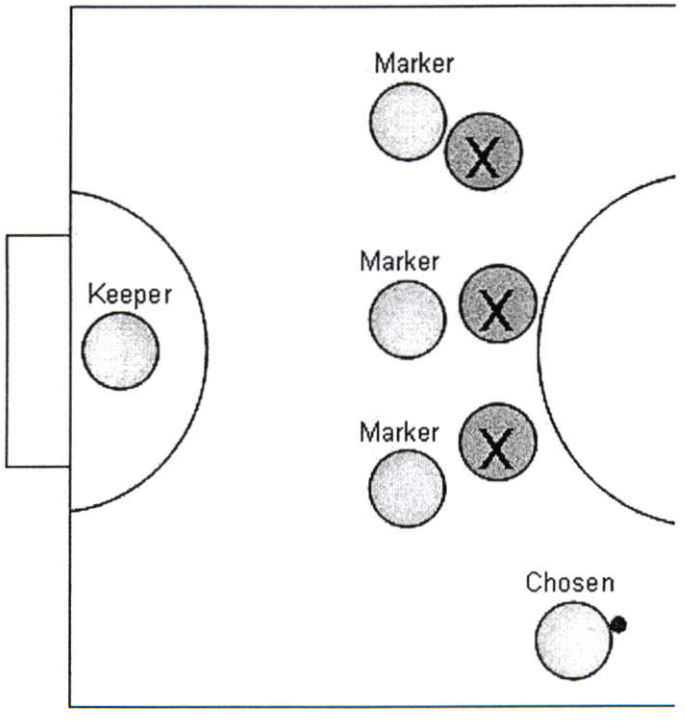
รูปที่ 4.1(ก) ตัวอย่างตำแหน่งการเล่นของแผน P1301



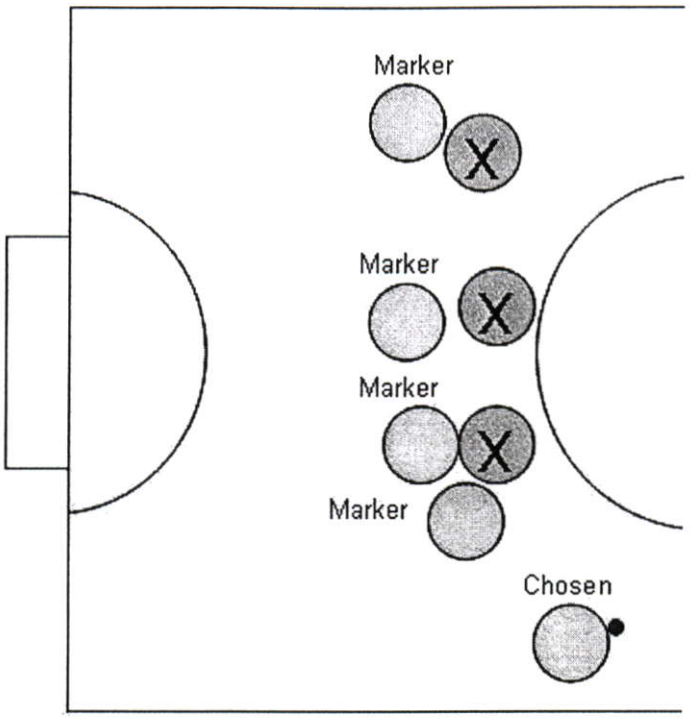
รูปที่ 4.1(ข) ตัวอย่างตำแหน่งการเล่นของแผน P1211



รูปที่ 4.1(ค) ตัวอย่างตำแหน่งการเล่นของแผน P1121



รูปที่ 4.1(ง) ตัวอย่างตำแหน่งการเล่นของแผน P1031



รูปที่ 4.1(จ) ตัวอย่างตำแหน่งการเล่นของแผน P0041

4.1.1 ตำแหน่งตัวไล่บอลและรุก (Chosen)

ตำแหน่งตัวไล่บอลและรุกเป็นตำแหน่งที่ระบุให้กับหุ่นยนต์ที่อยู่ในตำแหน่งใกล้ลูกฟุตบอลมากที่สุด ซึ่งต้องทำหน้าที่ไล่บอลในสถานการณ์ที่บอลไม่ได้อยู่ในการครอบครองของฝ่ายใด ทำหน้าที่แย่งบอลเมื่ออยู่ในสถานการณ์ที่ลูกฟุตบอลอยู่ในการครอบครองของฝ่ายตรงข้าม และทำหน้าที่ยิงประตูหรือพาบอลไปยังจุดที่สามารถยิงประตูได้ในสถานการณ์ที่ลูกฟุตบอลอยู่ในการครอบครองของทีม

ในทุกแผนการเล่นจะต้องมีตำแหน่งนี้อยู่จำนวนหนึ่งตัว เนื่องจากว่าเป็นตำแหน่งแรกที่กำหนดให้กับหุ่นยนต์ หากในกรณีที่ทีมสามารถส่งหุ่นยนต์ลงเล่นได้เพียงตัวเดียว หุ่นยนต์ตัวนั้นจะถูกระบุให้เล่นในตำแหน่งนี้ตลอดเวลา ซึ่งสามารถรุกและรับได้ทั้งสองอย่าง แม้ว่าเหลือหุ่นยนต์เพียงตัวเดียว ทีมก็ยังสามารถแข่งขันต่อไปได้

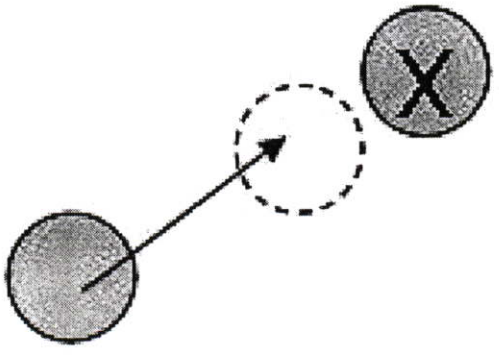
อีกสาเหตุที่ทุกๆ แผนต้องมีหุ่นยนต์ตัวใดตัวหนึ่งรับหน้าที่นี้และมีเพียงตัวเดียว เพราะเนื่องจากในสถานการณ์ที่จะต้องแย่งบอล เก็บบอลหรือยิงประตู จะต้องหลีกเลี่ยงการแย่งกันเล่นของหุ่นยนต์ เพื่อประสิทธิภาพการเล่นที่ดีที่สุด

4.1.2 ตำแหน่งตัวประกบ (Marker)

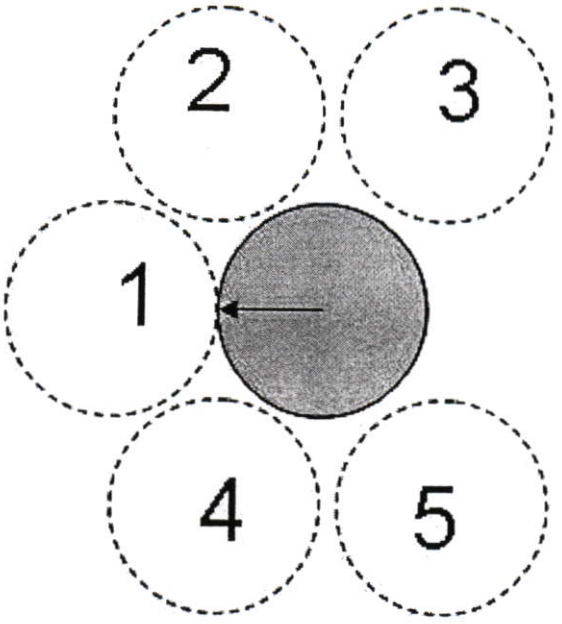
ตำแหน่งตัวประกบเป็นตำแหน่งที่ถูกระบุให้กับหุ่นยนต์ที่อยู่ใกล้กับลูกฟุตบอลรองลงมาจากตำแหน่งตัวไล่บอลและรุก ซึ่งตำแหน่งนี้หน้าที่หลักคือประกบคู่ต่อสู้ โดยวิ่งไปที่ตำแหน่งหุ่นยนต์คู่ต่อสู้ แต่จะพยายามรักษาระยะห่างไม่ให้ชนกับหุ่นยนต์คู่ต่อสู้

ลักษณะของการประกบนั้นจะแตกต่างไปจากสถานการณ์ของทีม หากสถานการณ์ของทีมอยู่ในระหว่างการเก็บลูกฟุตบอลหรือเป็นฝ่ายรุก การประกบจะเป็นลักษณะตัวต่อตัว แต่ถ้าหากอยู่สถานการณ์ที่จะต้องป้องกันการยิงประตู การประกบจะเป็นลักษณะกดดันหุ่นยนต์คู่ต่อสู้ โดยหุ่นยนต์ที่อยู่ในตำแหน่งนี้ทุกตัวจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้ามที่ครองบอลอยู่ แต่ในกรณีที่อยู่ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู หุ่นยนต์ที่อยู่ในตำแหน่งนี้จะประกบหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ใกล้ตัวที่สุด เพื่อป้องกันการวิ่งเข้ามาทำประตูของฝ่ายตรงข้าม

รูปที่ 4.2 แสดงตัวอย่างการประกบของหุ่นยนต์ตำแหน่งตัวประกบ และการประกบหุ่นยนต์คู่ต่อสู้นั้นสามารถแบ่งจุดการประกบได้ที่แสดงไว้ในรูปที่ 4.3 เพื่อประโยชน์ในการประกบโดยใช้หุ่นยนต์หลายๆ ตัวพร้อมๆ กัน



รูปที่ 4.2 ตัวอย่างการประกบของหุ่นยนต์ตำแหน่งตัวประกบ

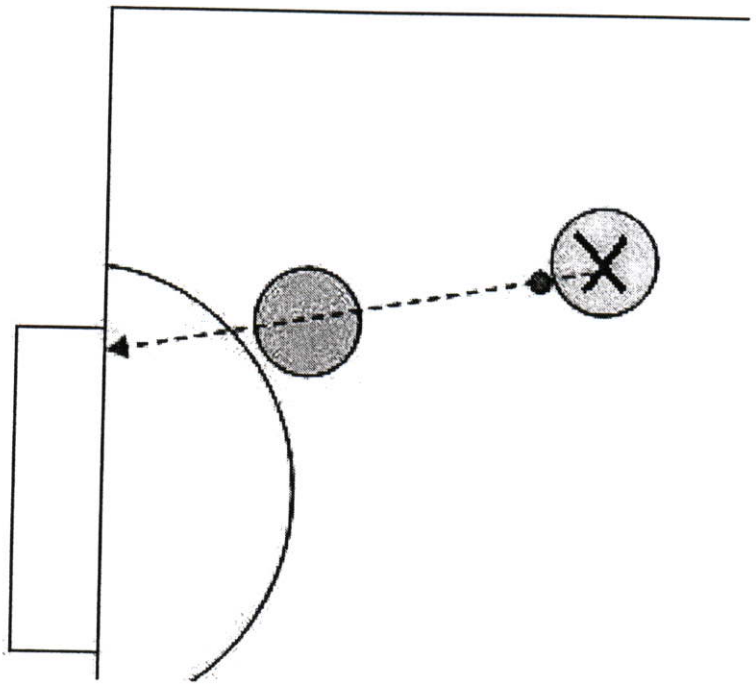


รูปที่ 4.3 ตำแหน่งการประกบของหุ่นยนต์ตำแหน่งตัวประกบ

4.1.3 ตำแหน่งตัวบังการยิง (Coverer)

ตำแหน่งตัวบังการยิงเป็นตำแหน่งที่ระบุให้กับหุ่นยนต์ที่ใกล้กับลูกฟุตบอลรองลงมาจากตำแหน่งตัวประกบ ซึ่งหน้าที่หลักคือบังทางการยิงของหุ่นยนต์ฝ่ายตรงข้าม หรือป้องกันลูกฟุตบอลวิ่งเข้าประตูก่อนถึงหุ่นยนต์ตำแหน่งผู้รักษาประตู โดยหุ่นยนต์ในตำแหน่งนี้จะวิ่งไปยังจุดที่บังทางเพื่อไม่ให้คู่ต่อสู้ยิงประตูได้ ในกรณีที่มีหุ่นยนต์เล่นในตำแหน่งนี้หลายๆ ตัว จุดที่ถูกกำหนดให้วิ่งบังทางจะถูกคำนวณให้สามารถบังทางครอบคลุมพื้นที่การยิงประตูให้มีประสิทธิภาพสูงสุด

โดยทั่วไปหุ่นยนต์ตำแหน่งนี้จะทำหน้าที่บังการยิงอยู่ตลอดเวลา โดยการวิ่งไปตำแหน่งที่คำนวณไว้ว่าสามารถบังทางบอลได้ดีที่สุด โดยรูปที่ 4.4 แสดงตัวอย่างการบังของหุ่นยนต์ตำแหน่งตัวบังการยิง แต่ในกรณีที่อยู่ในสถานการณ์ที่ลูกฟุตบอลอยู่ในเขตประตู หุ่นยนต์ที่อยู่ในตำแหน่งนี้ต้องหลีกทางให้กับหุ่นยนต์ตำแหน่งผู้รักษาประตูเล่นลูกฟุตบอลให้ออกมายังเขตปลอดภัย



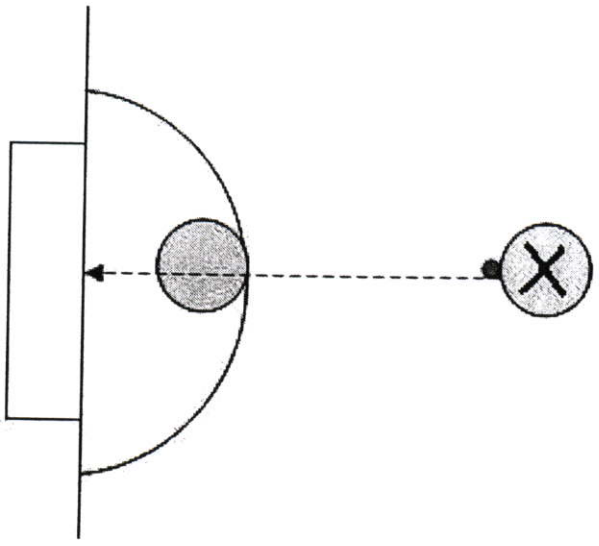
รูปที่ 4.4 ตัวอย่างการบังของหุ่นยนต์ตำแหน่งตัวบังการยิง

4.1.4 ตำแหน่งผู้รักษาประตู (Keeper)

ตำแหน่งผู้รักษาประตูเป็นตำแหน่งที่ระบุให้เป็นตำแหน่งป้องกันสุดท้าย ซึ่งโดยตามกติกาแต่ละทีมจะสามารถมีหุ่นยนต์เพียงตัวเดียวอยู่ในเขตประตู หน้าที่หลักของตำแหน่งนี้จะคล้ายคลึงกับหุ่นยนต์ตำแหน่งตัวบังการยิง โดยจะพยายามวิ่งรักษาตำแหน่งให้อยู่ในจุดที่สามารถบังการยิงได้ดีที่สุด แต่ทำงานเพียงตัวเดียว ไม่มีประสานงานตัวอื่น ดังที่แสดงไว้ในรูปที่ 4.5

แต่ในกรณีที่เกิดสถานการณ์ที่ลูกฟุตบอลอยู่ในเขตประตู ซึ่งจำเป็นจะต้องนำลูกฟุตบอลออกไปยังตำแหน่งที่ปลอดภัยกว่า หุ่นยนต์ตำแหน่งนี้จะวิ่งอ้อมไปด้านหลังและดันลูกฟุตบอลให้ออกจากเขตประตู ซึ่งหุ่นยนต์ตำแหน่งอื่นๆ จะวิ่งเปิดทางให้ลูกฟุตบอลไหลออกมาได้ง่าย ไม่บังทางกันเอง

ในกติกาของการแข่งขันหุ่นยนต์เล่นฟุตบอล หุ่นยนต์ของแต่ละฝ่ายสามารถเข้าไปอยู่ในเขตประตูได้เพียงตัวเดียว แต่ไม่จำเป็นต้องเป็นหุ่นยนต์ตัวเดียวกันตลอดการแข่งขัน หมายความว่าสามารถให้หุ่นยนต์ตัวอื่นๆ มาสลับตำแหน่งผู้รักษาประตูนี้ได้ ดังนั้นจึงให้หุ่นยนต์ตัวที่อยู่ตำแหน่งที่ไกลจากลูกฟุตบอลที่สุดเป็นหุ่นยนต์ตำแหน่งผู้รักษาประตู ซึ่งเมื่อสถานการณ์การเล่นของทีมมีการเปลี่ยนแปลง หุ่นยนต์ที่เล่นในตำแหน่งอื่นสามารถเปลี่ยนมาเล่นในตำแหน่งนี้ได้



รูปที่ 4.5 ตัวอย่างการป้องกันประตูของหุ่นยนต์ตำแหน่งผู้รักษาประตู

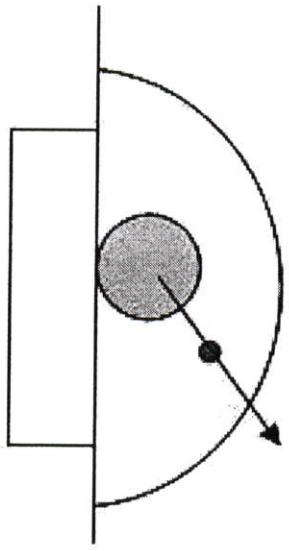
4.2 การคำนวณสถานการณ์การเล่นของทีม

สถานการณ์การเล่นจะต้องคำนวณเป็นอันดับแรก เพราะการคำนวณอื่นๆ ต้องอาศัยการระบุสถานการณ์ของทีมก่อนถึงจะทำงานได้อย่างตรงเป้าหมายและมีประสิทธิภาพ การที่ให้หุ่นยนต์เล่นฟุตบอลตัดสินใจว่าควรจะทำอะไร ควรให้หุ่นยนต์รู้ว่าตอนนี้ทีมตกอยู่สถานการณ์อะไรก่อน เพื่อจะได้ให้หุ่นยนต์ตัดสินใจไปในทิศทางเดียวกันหรือส่งเสริมซึ่งกันและกัน โดยสามารถแบ่งสถานการณ์ของทีมได้ดังนี้

4.2.1 สถานการณ์ลูกฟุตบอลอยู่ในเขตประตู (Goal zone)

สถานการณ์นี้เกิดขึ้นเมื่อลูกฟุตบอลอยู่ในเขตประตูของทีม ซึ่งเป็นสถานการณ์ที่จะต้องระวังที่สุด และจะต้องรีบพาลูกฟุตบอลให้ออกจากเขตนี้ให้เร็วที่สุด

เมื่ออยู่ในสถานการณ์นี้ หุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งอ้อมไปด้านหลังลูกบอลแล้วแล้วดันลูกฟุตบอลให้ออกจาเขต ดังที่แสดงในรูปที่ 4.6 โดยที่หุ่นยนต์ตำแหน่งตัวบังการยิงจะต้องเปิดทางให้หุ่นยนต์ตำแหน่งผู้รักษาประตูดันลูกบอลออกมาได้ ไม่ให้กีดขวางกันเอง ในขณะที่หุ่นยนต์ตำแหน่งตัวประกบและหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกก็จะพยายามกันไม่ให้หุ่นยนต์ฝ่ายตรงข้ามเข้ามาถึงลูกฟุตบอลได้



รูปที่ 4.6 ตัวอย่างการสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู

4.2.2 สถานการณ์ป้องกัน (Defend)

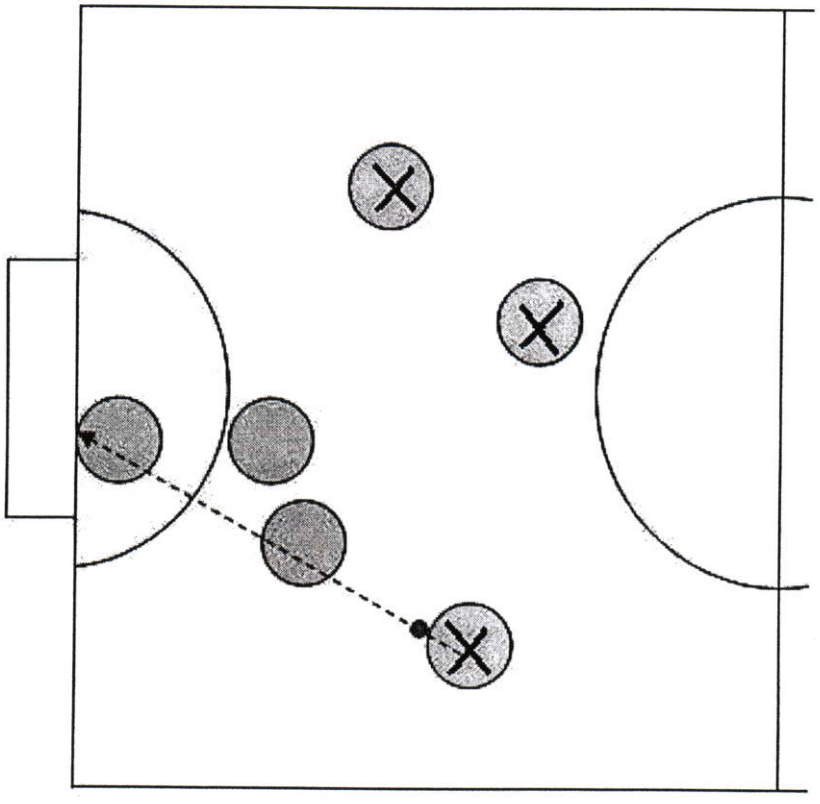
สถานการณ์นี้เกิดขึ้นเมื่อลูกฟุตบอลอยู่ในการครอบครองของหุ่นยนต์ฝ่ายตรงข้าม หรือระยะห่างระหว่างหุ่นยนต์ฝ่ายตรงข้ามกับลูกฟุตบอลมีระยะน้อยกว่า 200 มิลลิเมตรซึ่งเป็นสถานการณ์ที่หุ่นยนต์ฝ่ายตรงข้ามจะต้องเปิดเกมรุกเพื่อทำประตูอย่างแน่นอน

เมื่ออยู่ในสถานการณ์นี้ หุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดที่สามารถบังการยิงประตูได้ดีที่สุด เป็นจุดที่อยู่ในแนวระหว่างหุ่นยนต์ฝ่ายตรงข้ามที่ใกล้ลูกฟุตบอลที่สุดกับลูกฟุตบอล และต้องอยู่ไกลจากประตูมากกที่สุดแต่ไม่เกินเขตประตู เนื่องจากหุ่นยนต์เล่นฟุตบอลมีรูปทรงเป็นทรงกระบอก หากลูกฟุตบอลพุ่งเข้าชนจะกระดอนตามมุมตกกระทบ ซึ่งถ้าหากอยู่ใกล้กับประตูมากเกินไป ลูกฟุตบอลจะพุ่งเข้าประตู และถ้าหุ่นยนต์คู่ต่อสู้ทำประตู จะต้องให้ลูกฟุตบอลชนกับหุ่นยนต์ตำแหน่งผู้รักษาประตูในตำแหน่งที่ห่างจากประตูมากที่สุด เพื่อให้ลูกฟุตบอลวิ่งชนกับหุ่นยนต์ตำแหน่งผู้รักษาประตูในมุมตกกระทบที่ลูกฟุตบอลเบี่ยงออกจากกรอบประตู

ส่วนหุ่นยนต์ตำแหน่งตัวบังการยิงก็จะวิ่งไปจุดที่สามารถบังการยิงประตูได้ดีที่สุดเช่นเดียวกับหุ่นยนต์ตำแหน่งผู้รักษาประตู แต่วิ่งอยู่นอกเขตประตู ระยะห่างจากจุดตกกระทบเมื่อมีการยิงประตูอยู่ไกลกว่าจุดตกกระทบของหุ่นยนต์ตำแหน่งผู้รักษาประตู ซึ่งทำให้ปลอดภัยกว่าการป้องกันโดยหุ่นยนต์ตำแหน่งผู้รักษาประตูเพียงตัวเดียว แต่ถ้าหากหุ่นยนต์ตำแหน่งตัวบังการยิงมีหลายตัว หุ่นยนต์ตัวอื่นๆ ที่เป็นตำแหน่งเดียวกันจะวิ่งไปยังจุดที่ช่วยบังการยิงและปิดมุม ซึ่งเป็นจุดที่เพิ่มขอบเขตการป้องกันการทำประตูได้ดียิ่งขึ้น โดยจุดๆ นั้นจะไม่บังการวิ่งระหว่างหุ่นยนต์ตำแหน่งเดียวกัน

หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกซึ่งอยู่ในตำแหน่งที่ใกล้กับลูกฟุตบอลมากที่สุดจะยิ่งเข้าไปด้านหน้าของหุ่นยนต์ฝ่ายตรงข้ามที่ครอบครองลูกฟุตบอลอยู่เพื่อพยายามแย่งบอล ส่วนหุ่นยนต์

ตำแหน่งตัวประกบ จะวิ่งเข้าไปสนับสนุนการแย่งลูกฟุตบอลโดยการวิ่งเข้าไปขวางเส้นทางการวิ่งในทิศทางรอบหุ่นยนต์ฝ่ายตรงข้ามที่ครอบครองลูกฟุตบอลอยู่ เพื่อไม่ให้หุ่นยนต์ฝ่ายตรงข้ามพาลูกฟุตบอลหนีการแย่งบอลจากหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก โดยรูปที่ 4.7 แสดงตัวอย่างการสถานการณ์ป้องกัน



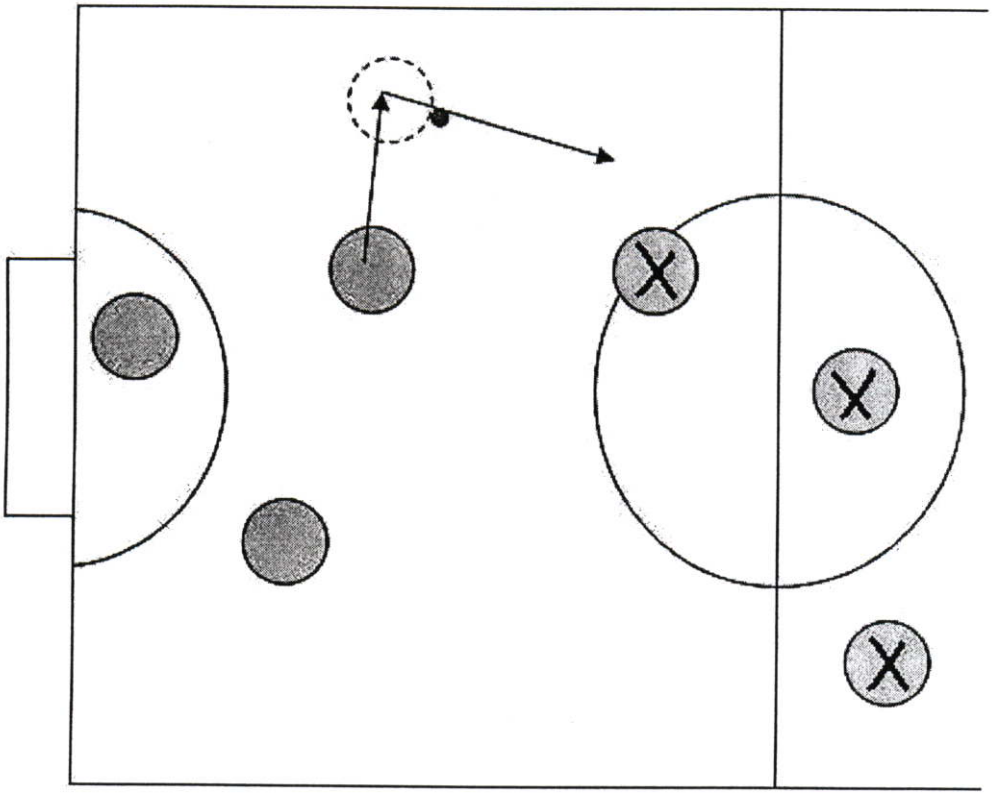
รูปที่ 4.7 ตัวอย่างการสถานการณ์ป้องกัน

4.2.3 สถานการณ์เก็บบอล (Collect)

สถานการณ์เก็บบอลเกิดขึ้นเมื่อลูกฟุตบอลไม่อยู่ในการครอบครองของฝ่ายใด หรืออีกนัยหนึ่งคือ ระยะห่างระหว่างหุ่นยนต์ฝ่ายตรงข้ามกับลูกฟุตบอลและระยะห่างระหว่างหุ่นยนต์ของทีมกับลูกฟุตบอลมีระยะมากกว่า 200 มิลลิเมตรทั้งสองฝ่าย ซึ่งเป็นสถานการณ์ที่หุ่นยนต์ของทีมและฝ่ายตรงข้ามจะต้องวิ่งเข้ามายังลูกฟุตบอลเพื่อทำเกมรุกอย่างแน่นอน

เมื่ออยู่ในสถานการณ์นี้ หุ่นยนต์ตำแหน่งผู้รักษาประตูและหุ่นยนต์ตัวบังการยิงจะทำงานเหมือนกับสถานการณ์ป้องกัน คือวิ่งไปยังจุดที่สามารถบังการยิงได้ดีที่สุด แต่หุ่นยนต์ตำแหน่งตัวประกบจะต้องวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ในตำแหน่งที่ใกล้มากที่สุด โดยไม่ประกบซ้ำตัวกัน ซึ่งเป็นการประกบแบบตัวต่อตัวเพื่อไม่ให้หุ่นยนต์คู่ต่อสู้วิ่งไปขวางการเก็บลูกฟุตบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก

จุดสำคัญของสถานการณ์นี้คือ หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะต้องวิ่งไปเก็บลูกฟุตบอลให้เร็วที่สุดและจะต้องวิ่งไปที่ตำแหน่งที่สามารถทำเกมรุกได้ทันทีได้อีกด้วย ดังนั้นจะต้องวิ่งอ้อมไปด้านหลังของลูกฟุตบอลเสมอตามจุดประสงค์ โดยรูปที่ 4.8 แสดงตัวอย่างการวิ่งเก็บลูกฟุตบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกในสถานการณ์เก็บบอล



รูปที่ 4.8 ตัวอย่างการสถานการณ์เก็บบอล

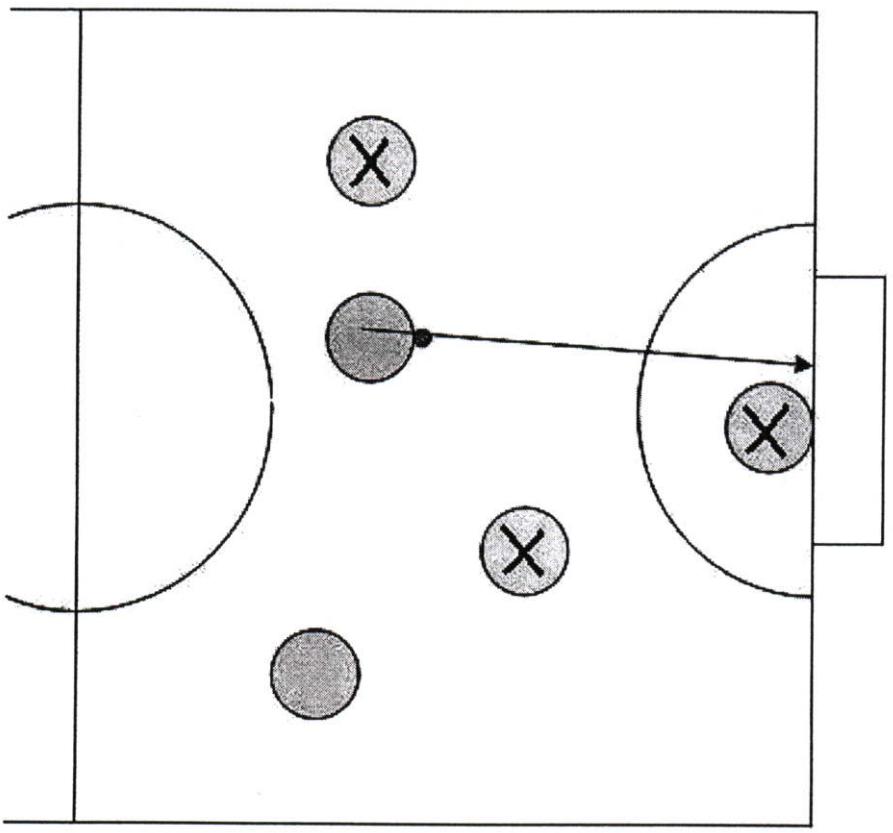
4.2.4 สถานการณ์รุก (Attack)

สถานการณ์รุกเกิดขึ้นเมื่อลูกฟุตบอลอยู่ในการครอบครองของทีม หรืออีกนัยหนึ่งคือ ระยะห่างระหว่างหุ่นยนต์ของทีมกับลูกฟุตบอลมีระยะน้อยกว่า 200 มิลลิเมตรซึ่งเป็นสถานการณ์ที่หุ่นยนต์ของทีมจะต้องเปิดเกมรุกเพื่อทำประตูให้ได้

เมื่ออยู่ในสถานการณ์นี้ หุ่นยนต์ตำแหน่งผู้รักษาประตูและหุ่นยนต์ตัวบังการยิงจะทำงานเหมือนกับสถานการณ์ป้องกัน โดยจะอยู่ในตำแหน่งที่สามารถบังมุมได้ดีที่สุด เผื่อในกรณีที่ลูกฟุตบอลกลิ้งเข้ามาทางประตูของทีม ส่วนหุ่นยนต์ตำแหน่งตัวประกบก็จะประกบหุ่นยนต์ฝ่ายตรงข้ามแบบตัวต่อตัวเพื่อป้องกันไม่ให้หุ่นยนต์ฝ่ายตรงข้ามตัวอื่นๆ เข้าไปแย่งลูกฟุตบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกได้

ในขณะที่หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามดันบอลไปในทิศทางเข้าหาประตูฝ่ายตรงข้ามเพื่อยิงประตู ซึ่งหากมีหุ่นยนต์ฝ่ายตรงข้ามเข้ามาพยายามสกัดหรืออาจจะมีหุ่นยนต์ฝ่าย

เดียวกันกีดขวางมุมการทำประตู ดังนั้นหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะต้องพาลูกฟุตบอลหลบสิ่งกีดขวางและพาบอลไปยังตำแหน่งที่มีมุมในการทำประตูให้ได้ โดยรูปที่ 4.9 แสดงตัวอย่างการสถานการณ์รุก



รูปที่ 4.9 ตัวอย่างการสถานการณ์รุก

ในบทที่ 5 ซึ่งเป็นบทต่อไปจะอธิบายรูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอล ซึ่งเนื้อหาประกอบไปด้วยการอธิบายตัวกระทำที่ทำงานตามจุดประสงค์ของหุ่นยนต์ การคำนวณข้อมูลตามจุดประสงค์ของการเล่น การคำนวณข้อมูลตามตำแหน่งของการเล่น และการคำนวณข้อมูลเพื่อประสิทธิภาพการเล่น

บทที่ 5
รูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอล

ในบทนี้จะอธิบายรูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอล ซึ่งถูกบรรจุไว้ในตัวกระทำข้อมูลภารกิจสู่จุดประสงค์ โดยทำหน้าที่คำนวณจุดที่ควรวิ่งไปและองศาด้านหน้าของหุ่นยนต์ โดยอาศัยข้อมูลสถานการณ์ของทีมจากชั้นภารกิจซึ่งแตกต่างไปตามตำแหน่งของหุ่นยนต์ที่ได้ถูกระบุไว้จากการคำนวณ

5.1 ตัวกระทำที่ทำงานตามจุดประสงค์ของหุ่นยนต์

การคำนวณของตัวกระทำเป็นการคำนวณเพื่อแปลงข้อมูลสถานการณ์ คือข้อมูล PLAY_STATE ในชั้นภารกิจมาคำนวณเป็นข้อมูลจุดที่จะต้องวิ่งไปและองศาด้านหน้าของหุ่นยนต์ในชั้นจุดประสงค์ตามจุดประสงค์ที่กำหนดโดยตำแหน่งการเล่นของหุ่นยนต์ที่กำหนดโดยข้อมูล PLAYPOS_1, PLAYPOS_2, PLAYPOS_3, PLAYPOS_4 และ PLAYPOS_5

รายละเอียดตัวกระทำที่ใช้ในการคำนวณตามจุดประสงค์ของการเล่นแสดงไว้ในตาราง 5.1 โดยแต่ละตัวกระทำนั้นจะประมวลผลตามข้อมูลตำแหน่งการเล่นของหุ่นยนต์และสถานการณ์ของทีม

ตารางที่ 5.1 รายละเอียดของตัวกระทำที่คำนวณตามจุดประสงค์ของการเล่น

ชื่อ	จุดประสงค์	ตำแหน่งการเล่น	สถานการณ์
COVER_PLAY	ขวางแนวการยิงประตูฝ่ายตรงข้าม	COVERER	ทุกๆ สถานการณ์ยกเว้น GOALZONE
UNCOVER_PLAY	เปิดทางให้ผู้รักษาเปิดบอลออกมา	COVERER	GOALZONE
MAN2MAN_PLAY	ประกบตัวต่อตัว	MARKER	COLLECT, ATTACK
PRESSING_PLAY	ประกบแบบรุม	MARKER	DEFEND
SCREEN_PLAY	ประกบตัวที่อยู่ใกล้	MARKER	GOALZONE

โดยข้อมูลที่ได้จากการคำนวณซึ่งเป็นจุดที่หุ่นยนต์จะต้องวิ่งไปและองศาด้านหน้าของหุ่นยนต์เบื้องต้น จากนั้นจะถูกคำนวณอีกทีโดยตัวกระทำที่ทำหน้าที่คำนวณจุดวิ่งแท้จริง โดยรายละเอียดตัวกระทำที่คำนวณจุดวิ่งแท้จริงแสดงไว้ในตาราง 5.2

ตารางที่ 5.2 รายละเอียดของตัวกระทำที่คำนวณจุดที่หุ่นยนต์วิ่งไปและองศาด้านหน้าของหุ่นยนต์

ชื่อ	จุดประสงค์	ตำแหน่งการเล่น
GK_PLAY	คำนวณการวิ่งและองศาด้านหน้าของหุ่นยนต์ตำแหน่งผู้รักษาประตู	KEEPER
CHOSEN_PLAY	คำนวณการวิ่งและองศาด้านหน้าของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก	CHOSEN
MARKER_PLAY	คำนวณการวิ่งและองศาด้านหน้าของหุ่นยนต์ตำแหน่งตัวประกบ	MARKER
COVERER_PLAY	คำนวณการวิ่งและองศาด้านหน้าของหุ่นยนต์ตำแหน่งตัวบังการยิง	COVERER

5.2 การคำนวณข้อมูลตามจุดประสงค์ของการเล่น

สำหรับหุ่นยนต์ตำแหน่งตัวประกบและหุ่นยนต์ตำแหน่งตัวบังการยิงนั้น เบื้องต้นจะต้องคำนวณจุดอ้างอิงก่อน เพื่อเป็นข้อมูลพื้นฐานให้ตัวกระทำที่ทำหน้าที่คำนวณจุดที่หุ่นยนต์วิ่งไปและด้านหน้าของหุ่นยนต์ตามตำแหน่งการเล่นคำนวณเป็นขั้นตอนสุดท้าย สามารถแบ่งตัวกระทำที่ใช้ในการคำนวณข้อมูลตามจุดประสงค์ของการเล่นได้ดังนี้

5.2.1 การคำนวณจุดบังการยิงของฝ่ายตรงข้าม

สำหรับการคำนวณจุดบังการยิงนั้นเป็นหน้าที่ของตัวกระทำ COVER_PLAY ซึ่งคำนวณจุดบังการยิงให้กับหุ่นยนต์ตำแหน่งตัวบังการยิงในกรณีที่เกิดสถานการณ์ใดๆ ก็ตาม ยกเว้นสถานการณ์ลูกฟุตบอลอยู่ในเขตประตูซึ่งจะเป็นหน้าที่ของตัวกระทำที่ชื่อว่า UNCOVER_PLAY

การคำนวณจุดบังการยิงจะต้องคำนวณวิถีการยิงของหุ่นยนต์ฝ่ายตรงข้ามก่อน ซึ่งข้อจำกัดของข้อมูลหุ่นยนต์เล่นฟุตบอลฝ่ายตรงข้าม ทำให้ไม่สามารถรู้ได้ว่าหุ่นยนต์ฝ่ายตรงข้ามนั้นหันหน้าไปทางทิศไหน รู้เพียงตำแหน่งเพียงเท่านั้น ดังนั้นการคำนวณวิถีการยิงนั้นจะต้องอาศัยจุดของลูกฟุตบอลมาช่วยในการหาวิถีการยิง โดยหากอยู่ในสถานการณ์ป้องกัน ลูกฟุตบอลจะอยู่ในตำแหน่งที่ใกล้กับหุ่นยนต์เล่นฟุตบอลของฝ่ายตรงข้ามมาก จะสามารถนำตำแหน่งนั้นมาคำนวณวิถีโดยใช้สมการเส้นตรง (5.1)

$$y = mx + c \tag{5.1}$$

เมื่อ y เป็นตำแหน่งในแนวแกน Y

x เป็นตำแหน่งในแนวแกน X

m เป็นความชันของเส้นตรงนั้น

c เป็นค่าคงที่ของเส้นตรง

โดยความชันของเส้นตรงนั้นสามารถหาได้จากสมการ (5.2)

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)} \tag{5.2}$$

เมื่อ y_1 เป็นตำแหน่งในแนวแกน Y ของจุดแรก

y_2 เป็นตำแหน่งในแนวแกน Y ของจุดที่สอง

x_1 เป็นตำแหน่งในแนวแกน X ของจุดแรก

x_2 เป็นตำแหน่งในแนวแกน X ของจุดที่สอง

โดยค่า c สามารถหาได้จากการแทนค่าจุดใดจุดหนึ่งที่อยู่บนเส้นตรงหลังจากหาค่า m ได้แล้ว

เมื่อได้สมการเส้นตรงจากการคำนวณแล้ว ก็สามารถหาจุดบังการยิงที่อยู่บนเส้นเขตประตูได้โดยการหาจุดตัดระหว่างเส้นตรงวิถีการยิงประตูกับวงกลมที่มีจุดศูนย์กลางอยู่ที่จุดกึ่งกลางขอบเส้นขอบประตูและมีรัศมีเท่ากับรัศมีเขตประตู โดยสมการ (5.3) แทนสมการวงกลมทั่วไป

$$(x-h)^2 + (y-k)^2 = r^2 \tag{5.3}$$

เมื่อ y เป็นตำแหน่งในแนวแกน Y

x เป็นตำแหน่งในแนวแกน X

h เป็นตำแหน่งจุดศูนย์กลางในแนวแกน X

k เป็นตำแหน่งจุดศูนย์กลางในแนวแกน Y

r เป็นรัศมีของวงกลม

เมื่อได้องค์ประกอบทั้งสองอย่างในการหาจุดบังการยิง คือวงกลมของเขตประตูและเส้นวิถีการยิง ก็จะสามารถหาจุดบังการยิงได้โดยหาจุดตัดของเส้นวิถีการยิงกับวงกลมเขตประตู โดยคำนวณจากสมการ (5.4)

$$i = b^2 - 4ac \tag{5.4}$$

โดย

$$b = 2((x_2 - x_1)(x_1 - x_3) + (y_2 - y_1)(y_1 - y_3)) \quad (5.5)$$

$$a = (x_2 - x_1)^2 + (y_2 - y_1)^2 \quad (5.6)$$

$$c = x_3^2 + y_3^2 + x_1^2 + y_1^2 - 2(x_3 x_1 + y_3 y_1) - r^2 \quad (5.7)$$

เมื่อ y_1 เป็นตำแหน่งในแนวแกน Y ของจุดแรกของเส้นตรง

y_2 เป็นตำแหน่งในแนวแกน Y ของจุดที่สองของเส้นตรง

y_3 เป็นตำแหน่งในแนวแกน Y ของจุดศูนย์กลางของวงกลม

x_1 เป็นตำแหน่งในแนวแกน X ของจุดแรกของเส้นตรง

x_2 เป็นตำแหน่งในแนวแกน X ของจุดที่สองของเส้นตรง

x_3 เป็นตำแหน่งในแนวแกน X ของจุดศูนย์กลางของวงกลม

เมื่อคำนวณค่า i ได้แล้ว ก็สามารถตรวจสอบว่าเส้นตรงตัดกับวงกลมหรือไม่โดย

- หาก $i < 0$ แสดงว่าเส้นตรงเส้นนั้นไม่ตัดกับวงกลม
- หาก $i = 0$ แสดงว่าเส้นตรงเส้นนั้นสัมผัสกับวงกลม มีจุดสัมผัสหนึ่งจุด
- หาก $i > 0$ แสดงว่าเส้นตรงเส้นนั้นตัดกับวงกลม มีจุดตัดสองจุด

ดังนั้นในกรณีที่ $i \leq 0$ ซึ่งไม่เกิดการตัดกันระหว่างวงกลมเขตประตูกับเส้นวิถีการยิง จะเปลี่ยนเส้นวิถีการยิงจากที่ใช้ตำแหน่งของหุ่นยนต์คู่ต่อสู้กับลูกฟุตบอลมาสร้างเป็นเส้นตรง มาเป็นสร้างจากตำแหน่งจุดกึ่งกลางของประตูกับลูกฟุตบอล จะทำให้สามารถคำนวณจุดบังการยิงได้ เพราะเส้นที่กำหนดใหม่นั้นจะตัดผ่านวงกลมเขตประตูอย่างแน่นอน

จากนั้นก็สามารถหาจุดตัดได้สองจุดได้โดย

$$xi_1 = x_1 + mu_1(x_2 - x_1) \quad (5.7)$$

$$yi_1 = y_1 + mu_1(y_2 - y_1) \quad (5.8)$$

โดย xi_1 เป็นตำแหน่งในแนวแกน X ของจุดตัดจุดแรก

yi_1 เป็นตำแหน่งในแนวแกน Y ของจุดตัดจุดแรก

เมื่อ

$$mu_1 = -b + \sqrt{\frac{b^2 - 4ac}{2a}} \tag{5.9}$$

$$xi_2 = x_1 + mu_2(x_2 - x_1) \tag{5.7}$$

$$yi_2 = y_1 + mu_2(y_2 - y_1) \tag{5.8}$$

โดย xi_2 เป็นตำแหน่งในแนวแกน X ของจุดตัดจุดที่สอง

yi_2 เป็นตำแหน่งในแนวแกน Y ของจุดตัดจุดที่สอง

เมื่อ

$$mu_2 = -b - \sqrt{\frac{b^2 - 4ac}{2a}} \tag{5.9}$$

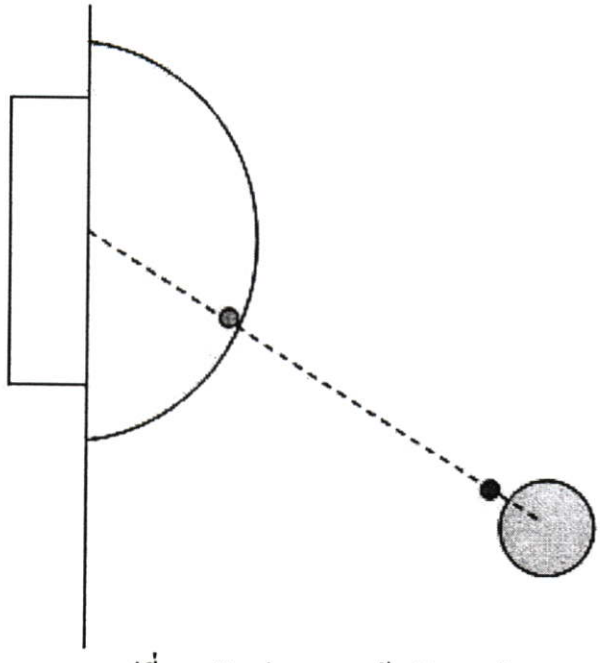
โดยทั้งสองจุดเป็นจุดตัดทั้งคู่ แต่มีเพียงจุดเดียวเท่านั้นที่เป็นจุดบังการยิง เนื่องจากอีกจุดหนึ่งจะอยู่ด้านหลังของประตู จึงไม่สามารถนำมาใช้ สามารถหาจุดบังการยิงได้ดังนี้

$$(xp, yp) = \begin{cases} (xi_1, yi_1); xi_1 \le xi_2 \\ (xi_2, yi_2); xi_1 > xi_2 \end{cases} \tag{5.10}$$

โดย xp เป็นตำแหน่งในแนวแกน X ของจุดบังการยิง

yp เป็นตำแหน่งในแนวแกน Y ของจุดบังการยิง

เมื่อได้คำนวณเรียบร้อยแล้ว จะได้จุดตัดบนวงกลมเขตประตู ซึ่งสามารถนำมาใช้อ้างอิงในการคำนวณหาจุดนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลตำแหน่งตัวบังการยิง โดยตัวอย่างจุดอ้างอิงการบังแสดงในรูปที่ 5.1



รูปที่ 5.1 ตัวอย่างการจุดอ้างอิงการบัง

การนำจุดอ้างอิงมาใช้ในการนำร่องจะต้องคำนวณโดยใช้รัศมีวงกลมที่ขนาดโตกว่าเขตของประตู เนื่องจากกติกาไม่ให้หุ่นยนต์เล่นฟุตบอลเข้าไปในเขตประตูของตนเองพร้อมกันสองตัว ซึ่งตามปกติจะมีหุ่นยนต์ตำแหน่งผู้รักษาประตูอยู่แล้ว ดังนั้นการคำนวณจุดบังการยิงของหุ่นยนต์ตำแหน่งตัวบังการยิงที่มีลำดับความสำคัญมากที่สุดจะใช้การคำนวณเหมือนกับการหาจุดอ้างอิงการบังแต่จะใช้รัศมีที่ใหญ่กว่า และจะคำนวณให้เยื้องกับจุดอ้างอิงเล็กน้อย เพื่อประสิทธิภาพการบังที่ดียิ่งขึ้น

ทั้งนี้สามารถดูตัวแปรการคำนวณหาจุดนำร่องได้จากตารางที่ 5.3

ตารางที่ 5.3 รายละเอียดของตัวแปรเพื่อการคำนวณหาจุดนำร่องการบังการยิง

ลำดับ	ระยะห่างจากจุดกึ่งกลางประตู	องศาเมื่อเทียบกับจุดอ้างอิง
1	1040	+10
2	1220	-10
3	1040	-20

สังเกตตารางที่ 5.3 จะเห็นว่าลำดับมีเพียง 3 ลำดับเท่านั้น เพราะว่าหากสังเกตที่ตาราง 4.2 จะเห็นได้ว่า ไม่ว่าจะใช้แผนอะไรจะมีตำแหน่งตัวไล่บอลและรุก 1 ตัวเสมอ และจำนวนหุ่นยนต์ตำแหน่งตัวบังการยิงจะมีจำนวนสูงสุดที่ 3 ตัวเท่านั้น ไม่ว่าจะใช้แผนใดก็ตาม

เมื่อกำหนดตัวแปรได้แล้ว ก็จะสามารถหาจุดนำร่องให้กับหุ่นยนต์ได้ โดยเริ่มที่การคำนวณหาองศาหรือความชันของเส้นตรงที่สร้างจากจุดกึ่งกลางประตูกับจุดอ้างอิงการบังก่อน โดยอาศัยสมการที่ (5.11)

$$\theta_r = \tan^{-1}\left(\frac{yg - yp}{xg - yp}\right) \tag{5.11}$$

โดย xg เป็นตำแหน่งในแนวแกน X ของจุดกึ่งกลางประตู

yg เป็นตำแหน่งในแนวแกน Y ของจุดกึ่งกลางประตู

θ_r องศาระหว่างจุดกึ่งกลางประตูกับจุดอ้างอิงการบัง

เมื่อได้มุมของเส้นตรงที่สร้างจากจุดกึ่งกลางประตูกับจุดอ้างอิงการบังแล้ว ก็สามารถหาจุดนำร่องของหุ่นยนต์ได้โดย

$$xc = xg + s[\cos(\theta_r + \theta_c)] \tag{5.12}$$

$$yc = yg + s[\sin\theta_r + \theta_c)] \tag{5.13}$$

โดย xc เป็นตำแหน่งในแนวแกน X ของจุดนำร่อง

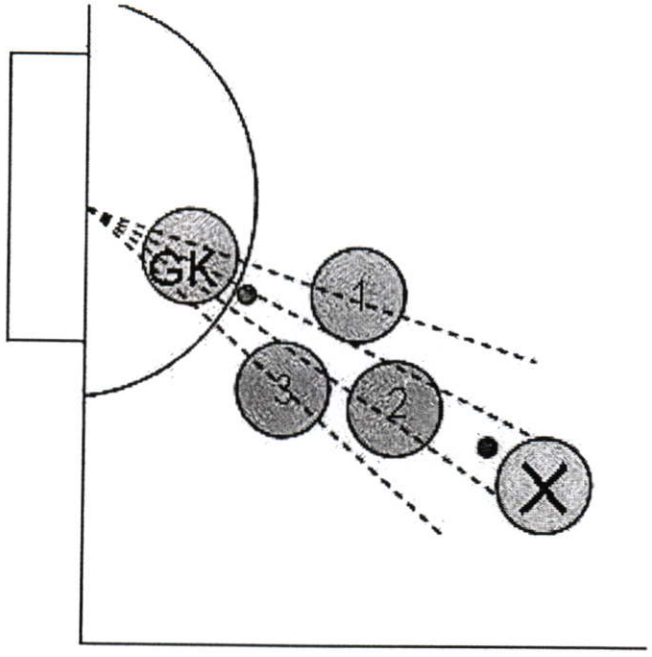
yc เป็นตำแหน่งในแนวแกน Y ของจุดนำร่อง

θ_c องศาเมื่อเทียบกับจุดอ้างอิงการบัง

θ_r องศาระหว่างจุดกึ่งกลางประตูกับจุดอ้างอิงการบัง

s ระยะห่างจากจุดกึ่งกลางประตู

ดังนั้นเมื่อแทนตัวแปรลงในสมการ (5.12) และ (5.13) ก็จะได้จุดนำร่องสำหรับหุ่นยนต์ตำแหน่งตัวบังการยิง ดังเช่นแสดงไว้ในรูปที่ 5.2



รูปที่ 5.2 ตัวอย่างการกำหนดจุดนำร่องสำหรับหุ่นยนต์ตำแหน่งบังการยิง

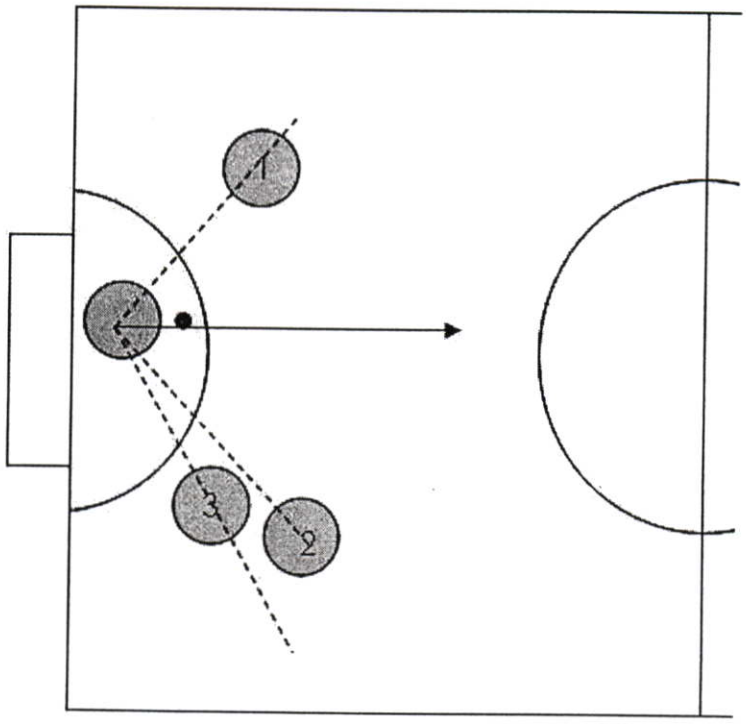
5.2.2 การคำนวณจุดหลบวิถีลูกฟุตบอลจากหุ่นยนต์ผู้รักษาประตู

สำหรับการคำนวณจุดบังการยิงนั้นเป็นหน้าที่ของตัวกระทำ UNCOVER_PLAY ซึ่งจะคำนวณจุดหลบวิถีลูกฟุตบอลจากหุ่นยนต์ผู้รักษาประตูในขณะที่เกิดสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู ซึ่งหุ่นยนต์ตำแหน่งผู้รักษาประตูต้องนำลูกฟุตบอลออกจากเขตของประตูให้เร็วที่สุด ดังนั้นหุ่นยนต์ตำแหน่งตัวบังการยิงจะต้องหลบวิถีลูกฟุตบอลที่ไหลออกมาจากเขตประตูให้ได้

การคำนวณวิถีของลูกฟุตบอลสามารถใช้สมการ (5.4)- (5.10) ในการหาจุดอ้างอิง โดยแทนค่าจุดที่หนึ่งให้เป็นจุดที่อยู่ของหุ่นยนต์ตำแหน่งผู้รักษาประตู และจุดที่สองเป็นลูกฟุตบอล จากนั้นก็ใช้สมการ (5.11) ในการหาองศาของเส้นตรงที่สร้างจากจุดที่อยู่ของหุ่นยนต์ตำแหน่งผู้รักษาประตูกับลูกฟุตบอล จากนั้นก็คำนวณจุดนำร่องของหุ่นยนต์ตำแหน่งตัวบังการยิงโดยใช้สมการ (5.12) และ (5.13) โดยใช้ตัวแปรในตารางที่ 5.4 โดยรูปที่ 5.3 แสดงตัวอย่างการกำหนดจุดนำร่องสำหรับการหลบวิถีลูกบอล

ตารางที่ 5.4 รายละเอียดของตัวแปรเพื่อการคำนวณหาจุดนำร่องสำหรับหลบวิถีบอล

ลำดับ	ระยะห่างจากจุดกึ่งกลางประตู	องศาเมื่อเทียบกับจุดอ้างอิง
1	1040	+45
2	1220	-45
3	1040	-55



รูปที่ 5.3 ตัวอย่างการกำหนดจุดนำร่องสำหรับการหลบวิถีลูกบอล

5.2.3 การคำนวณจุดประกบแบบตัวต่อตัว

การคำนวณจุดนำร่องสำหรับการประกบตัวต่อตัวของหุ่นยนต์ตำแหน่งตัวประกบนั้นเป็นหน้าที่ของตัวกระทำ MAN2MAN_PLAY ซึ่งจะคำนวณจุดประกบให้กับหุ่นยนต์ตำแหน่งตัวประกบในกรณีที่เกิดสถานการณ์เก็บบอลและรุก โดยการประกบแบบนี้จะเป็นการประกบโดยใช้หุ่นยนต์ของทีมหนึ่งตัว ประกบกับหุ่นยนต์ฝ่ายคู่ต่อสู้หนึ่งตัว

จากรูปที่ 4.2 จะเห็นว่าสามารถกำหนดจุดตายตัวโดยอ้างอิงกับจุดของหุ่นยนต์ฝ่ายตรงข้ามที่ต้องการประกบได้ทันที โดยจุดนำร่องสามารถคำนวณได้จากสมการ (5.14) และ (5.15)

$$xm = xo - 2R_{robot} + L_s \tag{5.14}$$

$$ym = yo \tag{5.15}$$

โดย xm เป็นตำแหน่งในแนวแกน X ของจุดนำร่องของการประกบ

ym เป็นตำแหน่งในแนวแกน Y ของจุดนำร่องของการประกบ

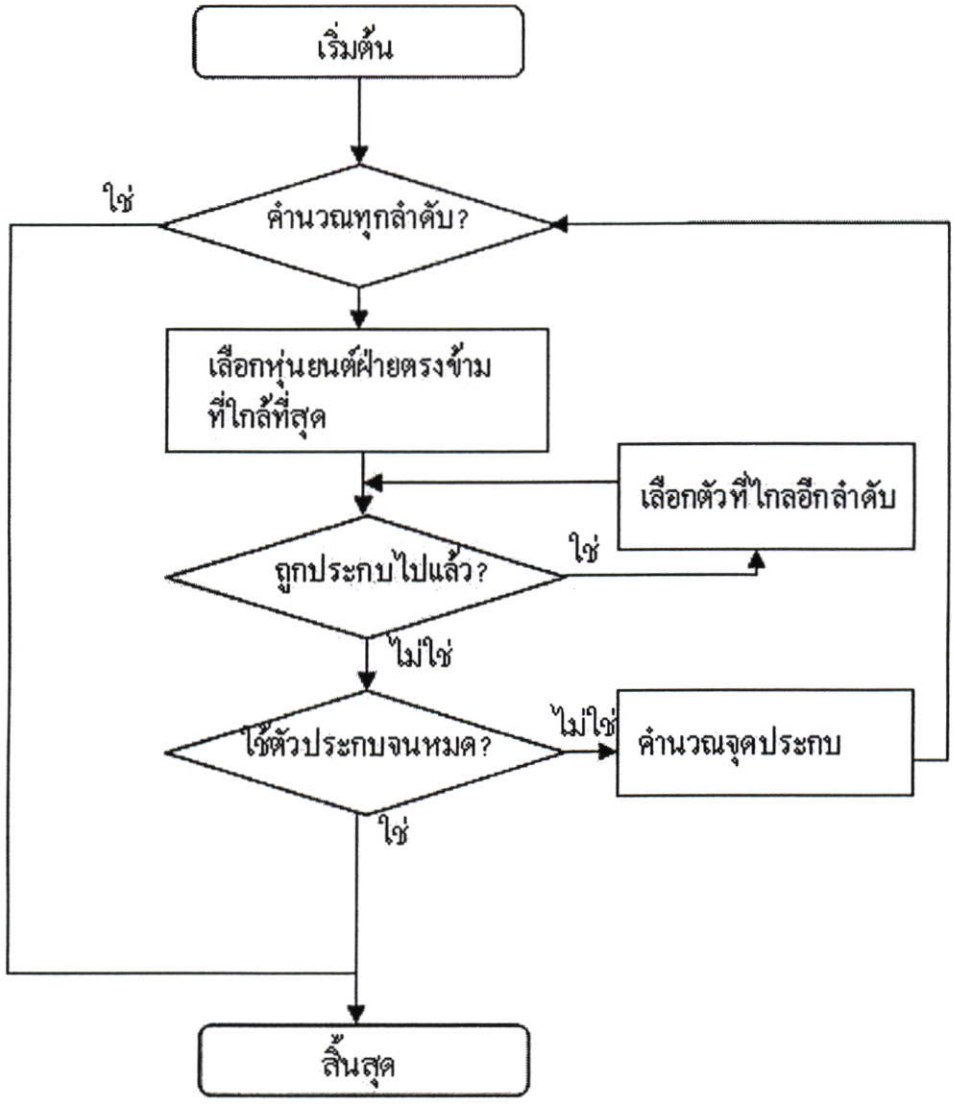
xo เป็นตำแหน่งในแนวแกน X ของจุดที่อยู่ของหุ่นยนต์ฝ่ายตรงข้าม

yo เป็นตำแหน่งในแนวแกน Y ของจุดที่อยู่ของหุ่นยนต์ฝ่ายตรงข้าม

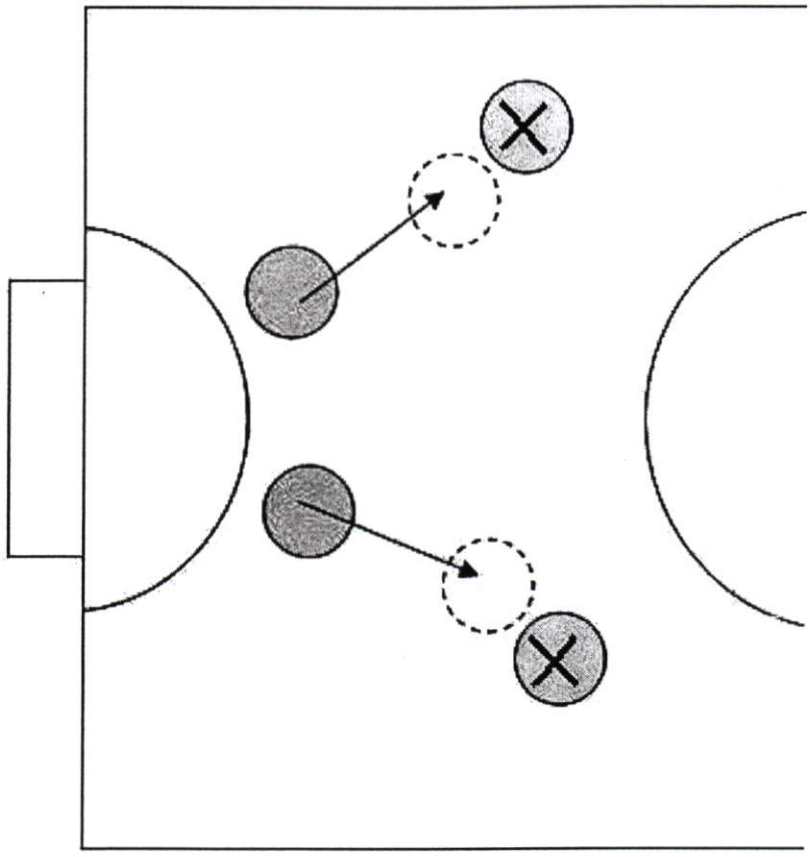
R_{robot} เป็นรัศมีของของหุ่นยนต์ฝ่ายตรงข้าม

L_s เป็นระยะห่างระหว่างหุ่นยนต์ตำแหน่งประกบกับหุ่นยนต์ฝ่ายตรงข้ามที่ต้องการ

โดยวิธีการคำนวณการประกบนั้นจะต้องใช้อัลกอริทึมในการเลือกตัวประกบโดยอาศัยตัวกระทำชื่อ PLAYPOS_ASSIGN คำนวณและบันทึกผ่านข้อมูล PLAYPOS_PRIORITY_1, PLAYPOS_PRIORITY_2, PLAYPOS_PRIORITY_3, PLAYPOS_PRIORITY_4 และ PLAYPOS_PRIORITY_5 ที่อยู่ในชั้นภารกิจ และกระบวนการของวิธีการคำนวณการประกบแบบตัวต่อตัวได้อธิบายในรูปที่ 5.4 และรูปที่ 5.5 แสดงตัวอย่างการคำนวณตัวประกบของการประกบแบบตัวต่อตัว



รูปที่ 5.4 การคำนวณตัวประกบของการประกบแบบตัวต่อตัว



รูปที่ 5.5 ตัวอย่างการคำนวณตัวประกบของการประกบแบบตัวต่อตัว

5.2.4 การคำนวณจุดประกบแบบกดดัน

การคำนวณจุดนำร่องสำหรับการประกบแบบกดดันของหุ่นยนต์ตำแหน่งตัวประกบนั้นเป็นหน้าที่ของตัวกระทำ PRESSING_PLAY ซึ่งคำนวณจุดประกบให้กับหุ่นยนต์ตำแหน่งตัวประกบในกรณีที่เกิดสถานการณ์ป้องกัน โดยการประกบนี้เป็นการประกบโดยใช้หุ่นยนต์ของทีมตำแหน่งตัวประกบทุกตัว ประกบกับหุ่นยนต์คู่ต่อสู้หนึ่งตัวที่อยู่ใกล้กับลูกฟุตบอลมากที่สุด

จุดนำร่องที่ใช้ในการประกบนั้น จะมีตำแหน่งดังรูปที่ 4.2 ซึ่งหุ่นยนต์ฝ่ายตรงข้ามหนึ่งตัวนั้นจะมีจุดประกับ 5 ตำแหน่งด้วยกัน คือ ด้านหน้า ด้านซ้ายหน้า ด้านซ้ายหลัง ด้านขวาหน้า และด้านขวาหลัง โดยการระบุจุดที่ประกบนั้นจะต้องระบุไม่ซ้ำกัน เพื่อไม่ให้หุ่นยนต์ตำแหน่งตัวประกบหลายตัววิ่งไปยังจุดเดียวกันจนเกิดการวิ่งชนกันขึ้น

จุดนำร่องนั้นสามารถคำนวณได้โดยใช้สมการที่ (5.16) และ (5.17) ดังนี้

$$xm = xo - (2R_{robot} + L_s)(\cos\theta_m) \tag{5.16}$$

$$ym = yo - (2R_{robot} + L_s)(\sin\theta_m) \tag{5.17}$$

โดย θ_m มุมการประกบเมื่อเทียบกับจุดที่อยู่ของหุ่นยนต์ฝ่ายตรงข้าม

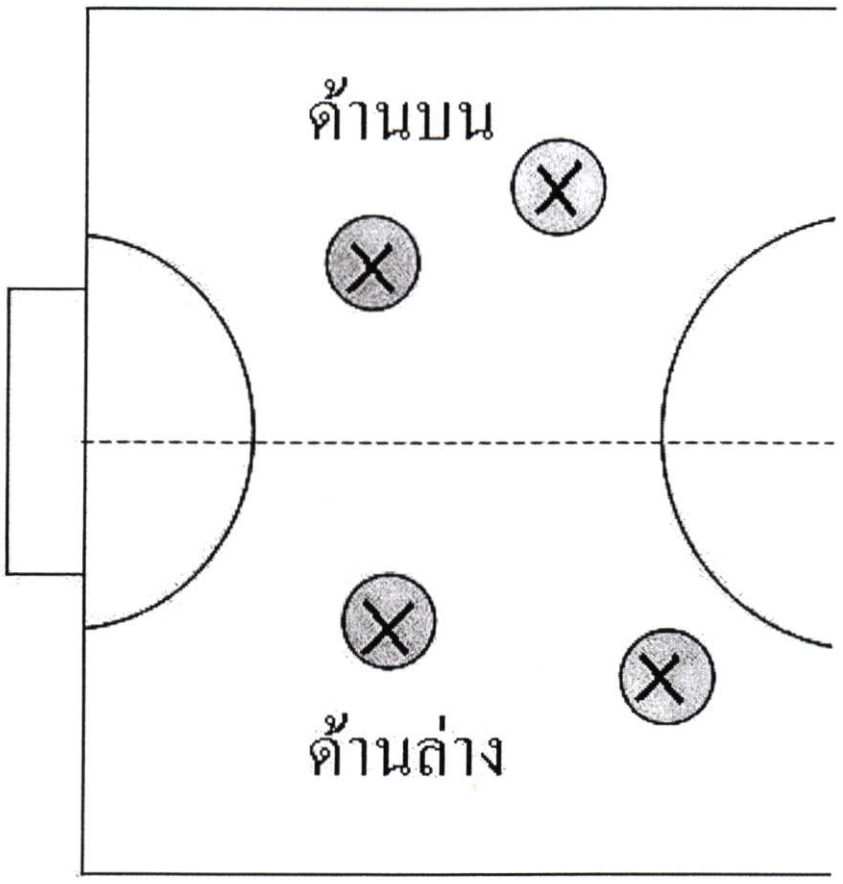
ดังนั้นจะสามารถกำหนดตัวแปรสำหรับจุดประกบได้ดังที่แสดงไว้ในตารางที่ 5.5 สำหรับหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านบนของสนามและตารางที่ 5.6 สำหรับหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านล่างของสนาม โดยรูปที่ 5.6 แสดงตัวอย่างจุดของหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านบนและด้านล่าง

ตารางที่ 5.5 รายละเอียดตัวแปรเพื่อการคำนวณหาจุดนำร่องสำหรับการประกบหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านบน

ลำดับ	จุดการประกบ	องศา
1	2	120
2	3	60
3	4	240
4	5	300

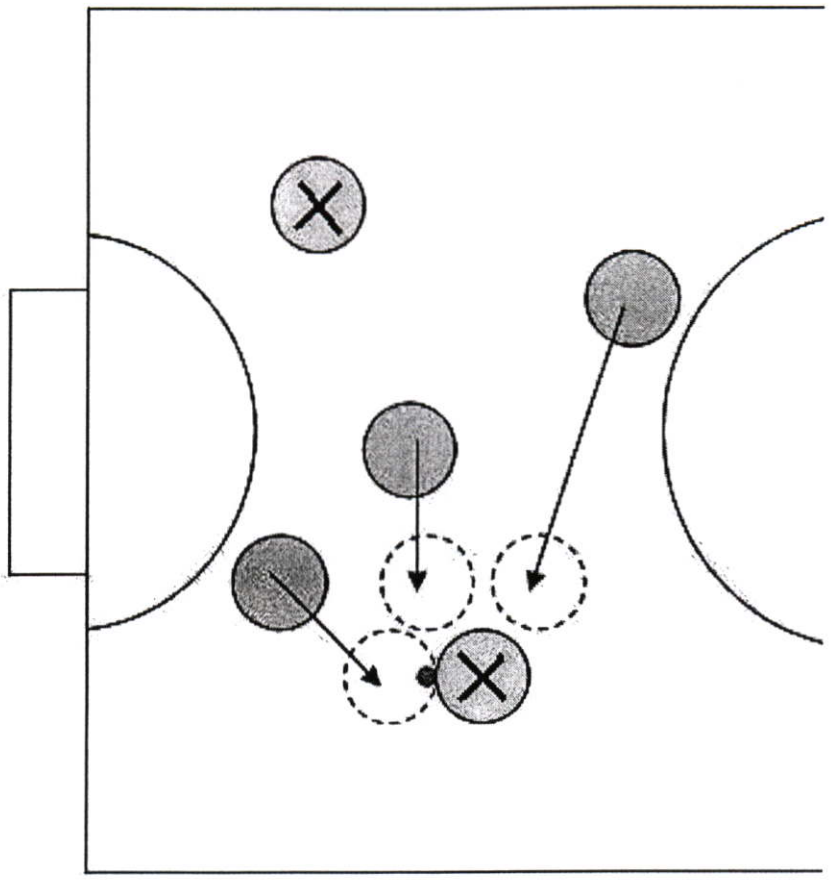
ตารางที่ 5.6 รายละเอียดตัวแปรเพื่อการคำนวณหาจุดนำร่องสำหรับการประกบหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านล่าง

ลำดับ	จุดการประกบ	องศา
1	4	240
2	5	300
3	2	120
4	3	60

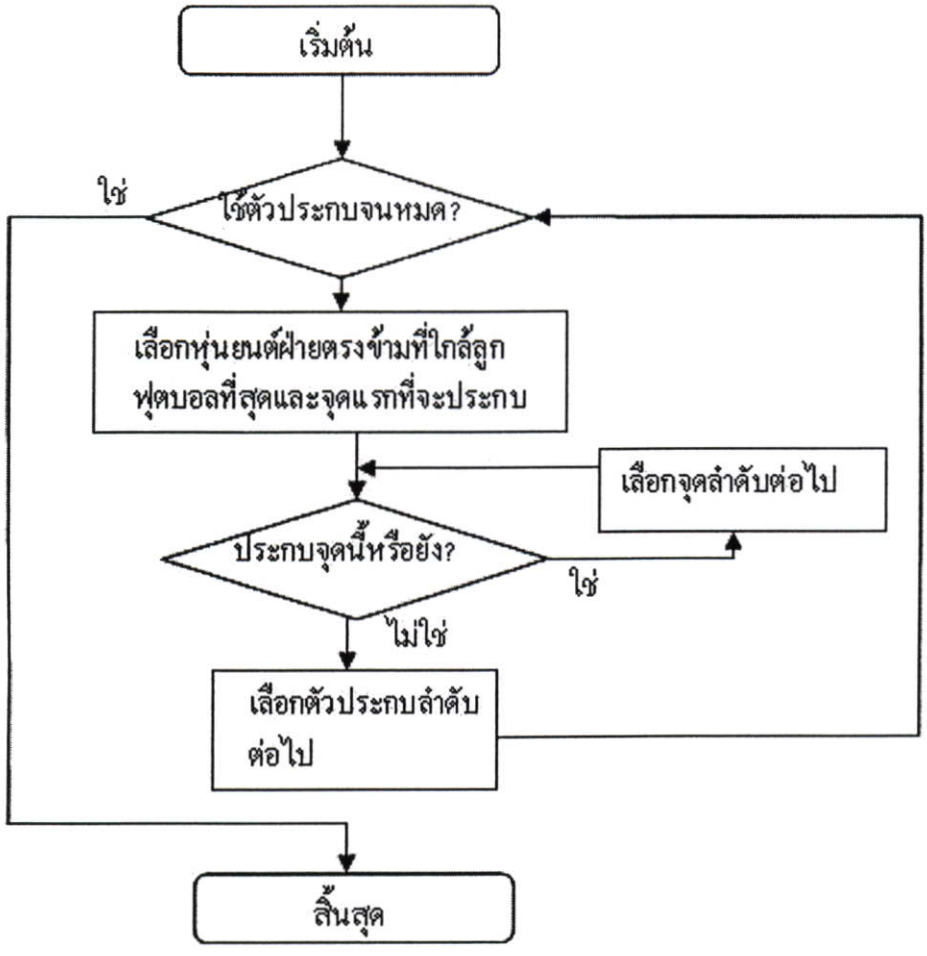


รูปที่ 5.6 ตัวอย่างจุดของหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ด้านบนและด้านล่าง

วิธีการคำนวณจุดประกบแบบกดดันต้องใช้อัลกอริธึมซึ่งอยู่ในตัวกระทำ PRESSING_PLAY และต้องอาศัยตัวกระทำ ชื่อ PLAYPOS_ASSIGN คำนวณและบันทึกผ่านข้อมูล PLAYPOS_PRIORITY_1, PLAYPOS_PRIORITY_2, PLAYPOS_PRIORITY_3, PLAYPOS_PRIORITY_4 และ PLAYPOS_PRIORITY_5 ที่อยู่ในชั้นภารกิจเพื่อนำมาใช้ในการคำนวณด้วย โดยรูปที่ 5.7 แสดงตัวอย่างการคำนวณตัวประกบของการประกบแบบกดดันและรูปที่ 5.8 แสดงวิธีการคำนวณตัวประกบของการประกบแบบกดดัน



รูปที่ 5.7 ตัวอย่างการคำนวณตัวประกบของการประกบแบบกดดัน



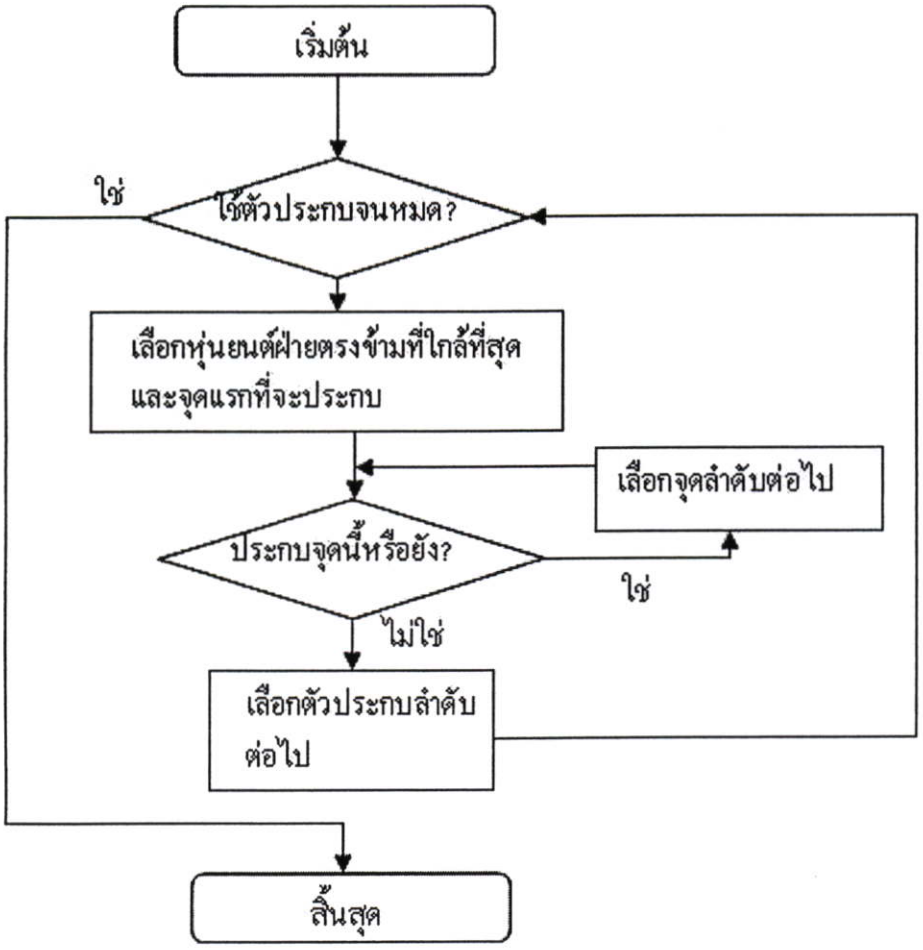
รูปที่ 5.8 การคำนวณตัวประกบของการประกบแบบกดดัน

5.2.4 การคำนวณจุดประกบแบบเปิดทาง

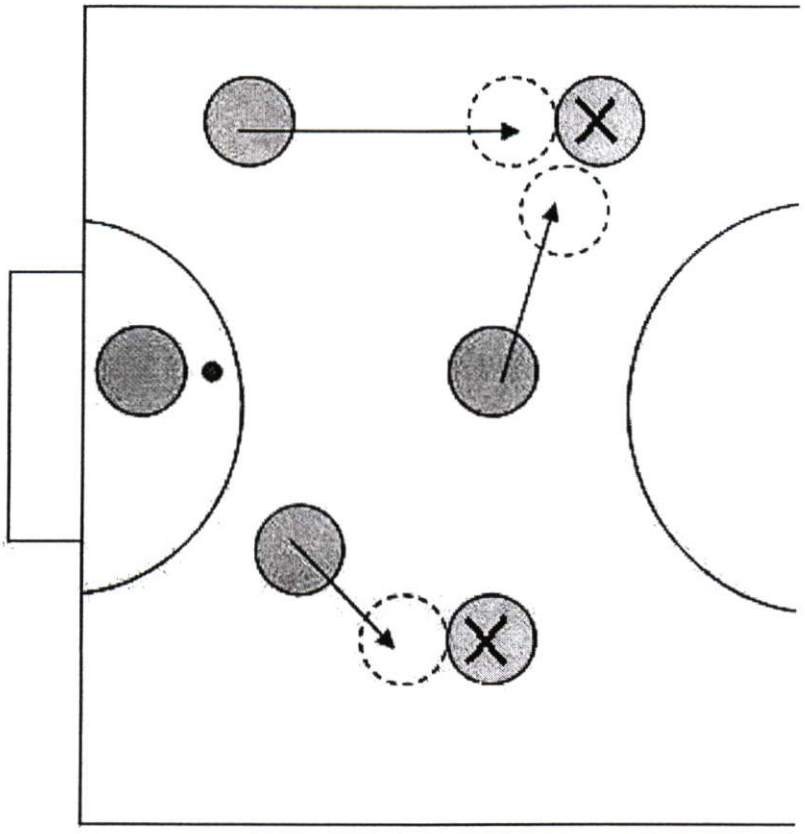
การคำนวณจุดนำร่องสำหรับการประกบแบบเปิดทางของหุ่นยนต์ตำแหน่งตัวประกบนั้นเป็นหน้าที่ของตัวกระทำ SCREEN_PLAY ซึ่งจะคำนวณจุดประกบให้กับหุ่นยนต์ตำแหน่งตัวประกบในกรณีที่เกิดสถานการณ์ที่ลูกฟุตบอลอยู่ในเขตประตู โดยการประกบนี้เป็นการประกบหุ่นยนต์ฝ่ายตรงข้ามที่อยู่ใกล้กับกับหุ่นยนต์ตำแหน่งตัวประกบมากที่สุดในขณะนั้น โดยแตกต่างจากการประกบแบบตัวต่อตัวที่หุ่นยนต์ฝ่ายตรงข้ามอาจโดนประกบโดยหุ่นยนต์ตำแหน่งตัวประกบพร้อมกันหลายๆ ตัวก็ได้

การคำนวณการประกบแบบเปิดทางอาจได้ผลลัพธ์ที่เหมือนกับการประกบแบบตัวต่อตัวหากหุ่นยนต์ฝ่ายตรงข้ามอยู่ใกล้กับหุ่นยนต์ตำแหน่งตัวประกบไม่ซ้ำกัน หรืออาจได้ผลลัพธ์เหมือนกับการประกบแบบกดดันหากหุ่นยนต์ฝ่ายตรงข้ามอยู่ใกล้กับหุ่นยนต์ตำแหน่งตัวประกบทุกๆ ตัว

จุดนำร่องที่ใช้ในการประกบนั้น มีตำแหน่งดังรูปที่ 4.2 ซึ่งหุ่นยนต์ฝ่ายตรงข้ามหนึ่งตัวนั้นจะมีจุดประกับ 5 ตำแหน่งด้วยกัน คือ ด้านหน้า, ด้านซ้ายหน้า, ด้านซ้ายหลัง, ด้านขวาหน้า และด้านขวาหลัง โดยการระบุหุ่นยนต์ฝ่ายตรงข้ามที่ประกบนั้นจะต้องสามารถซ้ำกันได้ แต่จุดประกบต่อหุ่นยนต์ฝ่ายตรงข้ามหนึ่งตัวนั้นต้องไม่ซ้ำกัน จุดนำร่องนั้นสามารถคำนวณได้โดยใช้สมการที่ (5.16) และ (5.17) และตัวแปรต่างๆ เหมือนกับการประกบแบบกดดัน ดังที่แสดงในตาราง 5.5 และ 5.6 โดยรูปที่ 5.9 แสดงการคำนวณตัวประกบของการประกบแบบเปิดทางและรูปที่ 5.10 แสดงตัวอย่างการคำนวณตัวประกบของการประกบแบบเปิดทาง



รูปที่ 5.9 การคำนวณตัวประกบของการประกบแบบเปิดทาง



รูปที่ 5.10 ตัวอย่างการคำนวณตัวประกบของการประกบแบบเปิดทาง

5.3 การคำนวณข้อมูลตามตำแหน่งของการเล่น

หลังจากได้คำนวณข้อมูลตามจุดประสงค์การเล่นแล้ว จะได้ข้อมูลเบื้องต้นที่นำมาคำนวณจุดนำร่องโดยตัวกระทำที่คำนวณตามตำแหน่งของการเล่น ซึ่งรายละเอียดของตำแหน่งต่างๆ ได้กล่าวไว้ในบทที่ 4 ในหัวข้อนี้จะอธิบายถึงการคำนวณอย่างละเอียดมากยิ่งขึ้น

5.3.1 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งผู้รักษาประตู

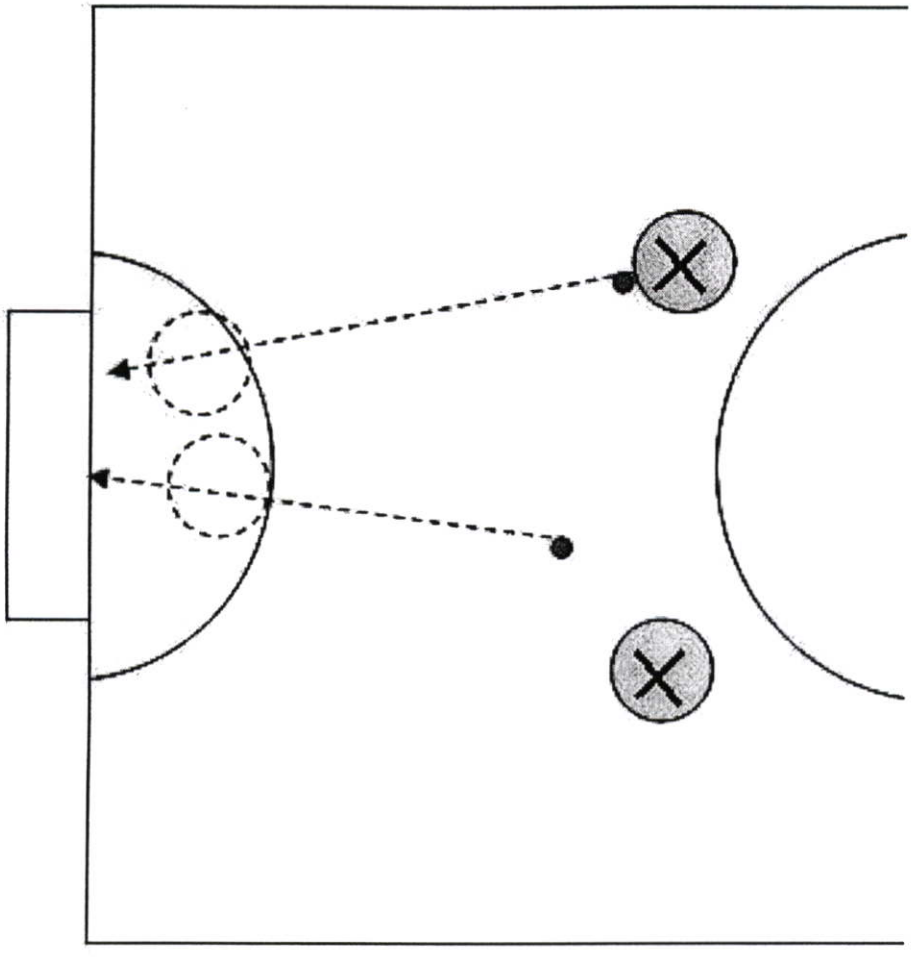
การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งนี้เน้นไปที่การคำนวณการนำร่องเพื่อบังการยิงของหุ่นยนต์ฝ่ายตรงข้ามหรือคำนวณเพื่อป้องกันประตูเป็นสำคัญ แต่ในกรณีที่ลูกฟุตบอลอยู่ในเขตประตูของฝ่ายตัวเอง จะต้องคำนวณจุดนำร่องเพื่อนำลูกบอลออกมาจากเขตประตูให้เร็วที่สุด

การนำร่องจุดบังการยิงของหุ่นยนต์ฝ่ายตรงข้ามนั้น จะสามารถคำนวณเหมือนกับการคำนวณจุดบังการยิงในหัวข้อ 5.2.1 โดยเปลี่ยนแค่ตัวแปรรัศมีวงกลมให้เล็กลง จากเดิมที่ขนาดเท่ากับรัศมีของเขตประตูให้เปลี่ยนเป็นขนาดเล็กลงตามสมการ (5.18) เพื่อให้หุ่นยนต์ตำแหน่งผู้รักษาประตูอยู่ในเขตประตูตลอดเวลา

$$r_k = r - R_{robot} \tag{5.18}$$

โดย r_k คือรัศมีที่ใช้ในการคำนวณจุดบังการยิงของหุ่นยนต์ตำแหน่งผู้รักษาประตู

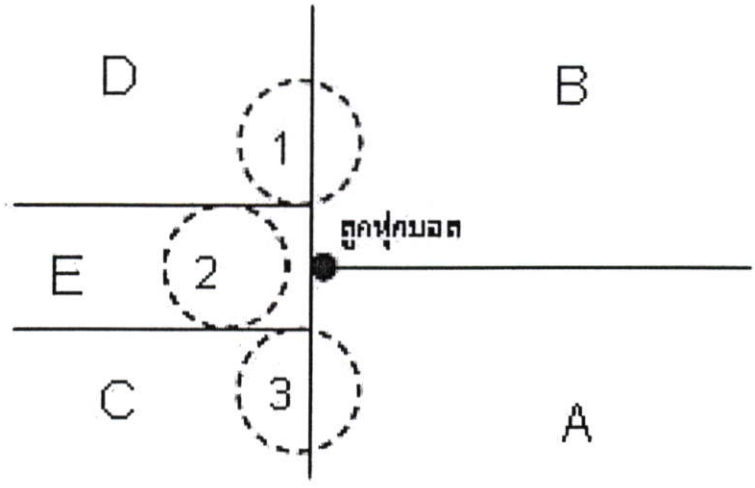
ในกรณีที่หุ่นยนต์ฝ่ายตรงข้ามกับลูกฟุตบอลไม่ได้อยู่ตำแหน่งที่ใกล้กัน การคำนวณวิถีการยิงประตูอาจจะนำมาใช้ไม่ได้ เพราะขณะนั้นไม่ได้อยู่ในสถานการณ์ป้องกัน ดังนั้นการคำนวณวิถีของลูกฟุตบอลโดยสมการ (5.4)-(5.10) จะต้องใช้เส้นตรงที่เกิดจากจุดลูกฟุตบอลกับจุดกึ่งกลางของประตูแทนที่จะใช้จุดตำแหน่งของหุ่นยนต์ฝ่ายตรงข้ามที่ใกล้ลูกฟุตบอลที่สุดกับลูกฟุตบอล โดยรูปที่ 5.11 แสดงตัวอย่างการคำนวณจุดนำร่องเพื่อบังวิถีลูกฟุตบอลทั้งสองวิธี



รูปที่ 5.11 ตัวอย่างการคำนวณจุดนำร่องเพื่อบังวิถีลูกฟุตบอล

กรณีที่ลูกฟุตบอลอยู่ในเขตประตู หุ่นยนต์เล่นฟุตบอลตำแหน่งผู้รักษาประตูจะต้องวิ่งอ้อมไปด้านหลังของลูกฟุตบอลเพื่อเตรียมพร้อมกับการนำลูกฟุตบอลออกจากเขตประตู โดยสามารถ

กำหนดจุดนำร่องเพื่อให้หุ่นยนต์วิ่งอ้อมได้โดยอาศัยการกำหนดจุดนำร่องโดยเทียบกับความสัมพันธ์ระหว่างจุดตำแหน่งของหุ่นยนต์ผู้รักษาประตูกับลูกฟุตบอล ซึ่งจะแบ่งความสัมพันธ์เป็นพื้นที่ตามรูปที่ 5.12 และเทียบจุดนำร่องกับความสัมพันธ์โดยใช้ตารางที่ 5.7 ก็จะสามารถนำคำนวณจุดนำร่องตามจุดประสงค์ได้โดยง่าย



รูปที่ 5.12 การแบ่งพื้นที่เพื่อนำร่องให้หุ่นยนต์ตำแหน่งผู้รักษาประตูวิ่งอ้อมไปด้านหลัง

ตารางที่ 5.7 การนำร่องหุ่นยนต์เล่นฟุตบอลตำแหน่งผู้รักษาให้วิ่งอ้อมไปหลังลูกฟุตบอล

พื้นที่	แกน X	แกน Y	จุดนำร่อง
A	≥ 0	< 0	3
B	≥ 0	≥ 0	1
C	< 0	< -90	2
D	< 0	≥ 90	2
E	< 0	≥ -90 และ < 90	2

ตารางที่ 5.8 จุดนำร่องเพื่อวิ่งอ้อมด้านหลังลูกฟุตบอล

จุดนำร่อง	แกน X	แกน Y
1	-10	+180
2	-110	0
3	-10	-180

เมื่อสามารถทำให้หุ่นยนต์วิ่งอ้อมไปด้านหลังลูกฟุตบอลได้แล้ว จะต้องคำนวณจุดเตรียมพร้อมให้หุ่นยนต์พาลูกฟุตบอลออกจากเขตประตูได้อย่างปลอดภัย โดยกำหนดจุดเตรียมพร้อมให้ระยะที่พาลูกฟุตบอลออกมาจากเขตประตูสั้นที่สุด

เส้นทางในการพาลูกฟุตบอลออกจากเขตประตูนั้นสามารถหาได้โดยใช้สมการ (5.1)-(5.2) โดยใช้จุดกึ่งกลางประตูกับจุดตำแหน่งลูกฟุตบอลสร้างเส้นตรง จากนั้นใช้สมการ (5.11) ในการองศาของเส้นตรงที่เป็นเส้นทาง ก็สามารถหาจุดเตรียมพร้อมได้โดย

$$x_{pk} = x_{ball} + [(s_r + R_{robot})\cos\theta_{pk}] \tag{5.19}$$

$$y_{pk} = y_{ball} + [(s_r + R_{robot})\sin\theta_{pk}] \tag{5.20}$$

โดย x_{pk} เป็นตำแหน่งในแนวแกน X ของจุดนำร่อง

y_{pk} เป็นตำแหน่งในแนวแกน Y ของจุดนำร่อง

x_{ball} เป็นตำแหน่งในแนวแกน X ของจุดที่อยู่ของลูกฟุตบอล

y_{ball} เป็นตำแหน่งในแนวแกน Y ของจุดที่อยู่ของลูกฟุตบอล

s_r ระยะห่างปลอดภัยที่เพิ่มเข้าไป

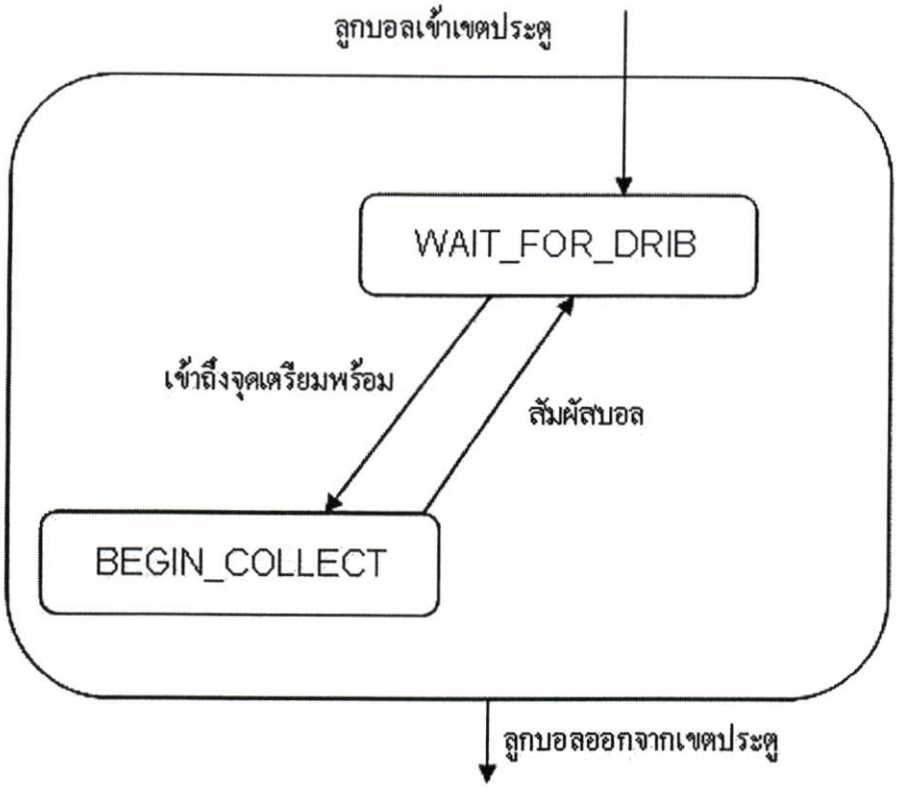
R_{robot} รัศมีของหุ่นยนต์

θ_{pk} เป็นมุมของเส้นทางการพาลูกฟุตบอลออกนอกเขตประตู

จากข้างต้น จะได้จุดเตรียมพร้อมสองจุด ซึ่งต้องเลือกจุดใดจุดหนึ่ง เมื่อจุดตำแหน่งของหุ่นยนต์เล่นฟุตบอลอยู่ด้านหลังลูกฟุตบอลแล้ว จะเลือกได้โดยกำหนดเงื่อนไขให้จุดที่ใกล้กับจุดตำแหน่งหุ่นยนต์มากที่สุดเป็นจุดเตรียมพร้อม

$$p_{prepare} = \begin{cases} p_{pk}; s_{pk} \le s_{pb} \\ p_{pb}; s_{pk} > s_{pb} \end{cases} \tag{5.21}$$

เมื่อคำนวณจุดเตรียมพร้อมได้แล้ว การนำร่องให้หุ่นยนต์เล่นฟุตบอลตำแหน่งผู้รักษาประตูเพื่อนำลูกฟุตบอลออกไปยังเขตประตูจะต้องกำหนดเงื่อนไขที่เหมาะสมกับจุดประสงค์ที่ต้องการโดยการอาศัยกฎการนำร่อง



รูปที่ 5.13 แผนภาพการทำงานของการนำร่องสำหรับหุ่นยนต์ตำแหน่งผู้รักษาประตู

จากรูปที่ 5.13 จะเห็นได้ว่าการทำงานจะแบ่งออกได้เป็น 3 สถานะคือ BEGIN_COLLECT และWAIT_FOR_DRIB โดยเริ่มต้นการทำงานที่ WAIT_FOR_DRIB เมื่อลูกฟุตบอลเข้ามายังเขตประตู ซึ่งในสถานะนี้จะนำร่องให้หุ่นยนต์วิ่งไปยังจุดเตรียมพร้อม ซึ่งเมื่อเข้าถึงจุดเตรียมพร้อมหรือระยะห่างระหว่างหุ่นยนต์เล่นฟุตบอลกับลูกฟุตบอลน้อยกว่า 66 มิลลิเมตรจะเข้าสถานะ BEGIN_COLLECT ซึ่งจุดนำร่องจะถูกกำหนดไปที่ลูกฟุตบอล เมื่อหุ่นยนต์วิ่งไปยังลูกฟุตบอลจนสัมผัสกับลูกฟุตบอลก็จะกลับไปยังสถานะ WAIT_FOR_DRIB อีกครั้ง

การคำนวณระยะห่างระหว่างจุดสองจุดสามารถหาได้โดยใช้สมการ (5.22)

$$s = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{5.22}$$

โดย s เป็นระยะห่างระหว่างจุดสองจุด

x_1 เป็นตำแหน่งในแนวแกน X ของจุดที่ 1

y_1 เป็นตำแหน่งในแนวแกน Y ของจุดที่ 1

x_2 เป็นตำแหน่งในแนวแกน X ของจุดที่ 2

y_2 เป็นตำแหน่งในแนวแกน Y ของจุดที่ 2

ส่วนมุมด้านหน้าของหุ่นยนต์นั้น จะกำหนดให้หุ่นยนต์ตำแหน่งผู้รักษาประตูนั้นหันหน้าเข้าหาลูกฟุตบอลอยู่เสมอ สามารถใช้สมการ (5.11) ใช้จุดตำแหน่งของหุ่นยนต์ตำแหน่งผู้รักษาประตูเทียบกับจุดตำแหน่งของลูกฟุตบอลในการเทียบมุมได้

5.3.2 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งตัวบังการยิง

การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งนี้แทบไม่มีการคำนวณที่ซับซ้อนใดๆ เพราะเนื่องจากมีตัวกระทำ COVER_PLAY และ UNCOVER_PLAY กำหนดจุดนำร่องไว้อยู่แล้ว ในการคำนวณจึงมีแค่การคำนวณมุมด้านหน้าของหุ่นยนต์ตำแหน่งตัวบังการยิงเท่านั้น

มุมหันหน้าที่ดีที่สุดในการบังการยิงนั้นคือ มุมที่ด้านหน้าของหุ่นยนต์หันหน้าไปทางจุดนำร่อง เพราะสามารถวิ่งไปยังจุดเป้าหมายได้เร็วที่สุด โดยสามารถใช้สมการ (5.11) ใช้จุดตำแหน่งของหุ่นยนต์ตำแหน่งตัวบังการยิงเทียบกับจุดเป้าหมายได้

5.3.3 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งตัวประกบ

การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งนี้แทบไม่มีการคำนวณที่ซับซ้อนใดๆ เหมือนกับการคำนวณข้อมูลของหุ่นยนต์ตัวบังการยิง เพราะเนื่องจากมีตัวกระทำ PRESSING_PLAY MAN2MAN_PLAY และ SCREEN_PLAY กำหนดจุดนำร่องไว้อยู่แล้ว ส่วนการคำนวณมุมด้านหน้าของหุ่นยนต์นั้นมิได้คำนวณ ดังนั้นหุ่นยนต์จะวิ่งไปโดยหันหน้าไปในทิศทางเดิมหากระบบขับเคลื่อนของหุ่นยนต์เป็นแบบรอบ

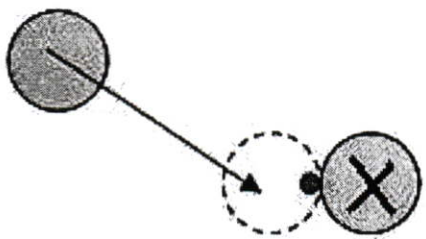
5.3.4 การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก

การคำนวณข้อมูลของหุ่นยนต์ตำแหน่งนี้เน้นไปที่การแย่งลูกฟุตบอลกับฝ่ายตรงข้ามในสถานการณ์ป้องกัน, การประกบตัวหุ่นยนต์ฝ่ายตรงข้ามในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู การเก็บลูกฟุตบอลในสถานการณ์เก็บบอล และการยิงประตูสำหรับสถานการณ์รุก

ในสถานการณ์ป้องกันนั้น ในขณะที่หุ่นยนต์ตำแหน่งอื่นๆ จะทำหน้าที่ตามที่ได้กำหนดหน้าที่ของหุ่นยนต์ตำแหน่งนี้คือการแย่งลูกฟุตบอลจากฝ่ายตรงข้าม หรือทำให้ลูกฟุตบอลหลุดจากการครอบครองของฝ่ายตรงข้าม ดังนั้นการคำนวณจุดแย่งบอลนั่นเป็นสิ่งที่จำเป็นในการทำงานของหุ่นยนต์ตำแหน่งนี้

การคำนวณจุดนำร่องเพื่อการแย่งบอลนั้น จะใช้การคำนวณเหมือนกับการคำนวณจุดประกบของหุ่นยนต์ตำแหน่งตัวประกบทุกประการ ซึ่งหุ่นยนต์ตำแหน่งตัวประกบจะกำหนดตำแหน่งในการประกบแล้ว ซึ่งจะเหลือจุดตำแหน่งสำหรับแย่งลูกฟุตบอลจากหุ่นยนต์ที่ฝ่ายตรงข้ามครอบครองลูกฟุตบอลให้หุ่นยนต์ตำแหน่งตัวไล่บอล ซึ่งก็คือตำแหน่งด้านหน้าของหุ่นยนต์

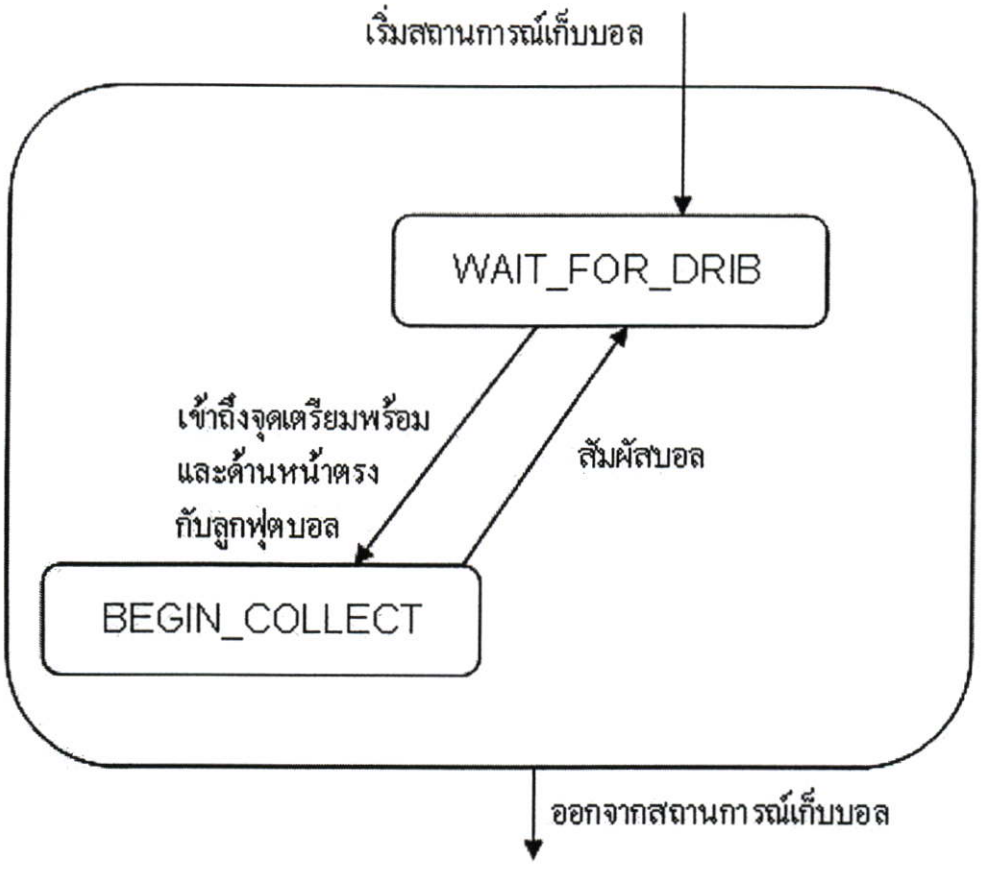
ฝ่ายตรงข้าม หรือตำแหน่งที่ 1 ในรูป 4.2 หรืออาจจะคำนวณจุดแย่งบอลโดยใช้สมการ (5.14)-(5.15) ได้ โดยรูปที่ 5.14 แสดงตัวอย่างการแย่งบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก



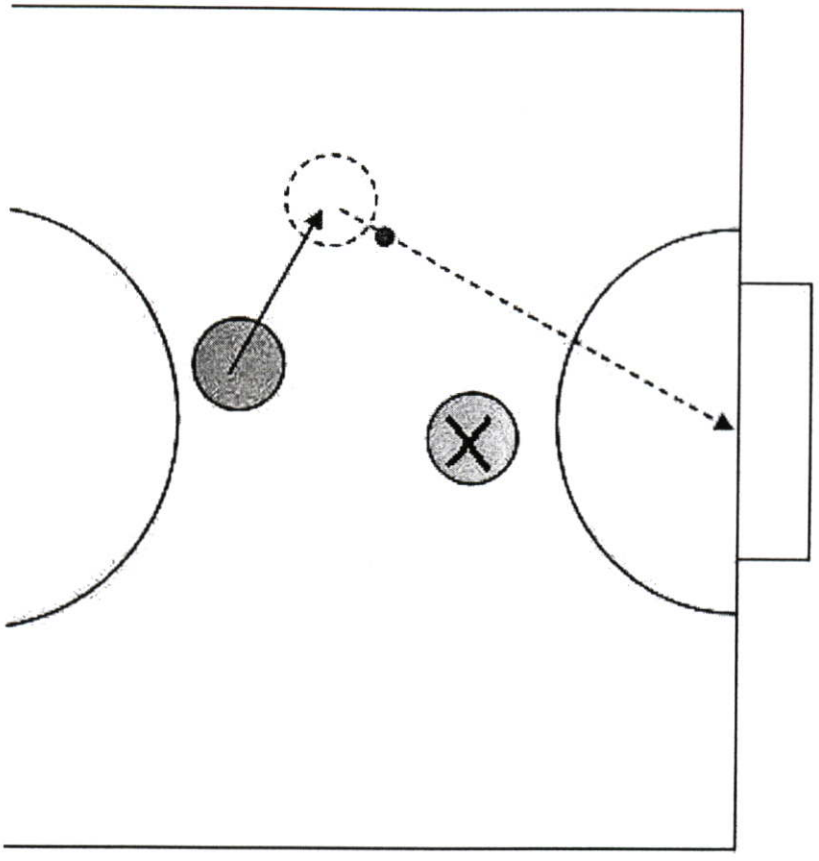
รูปที่ 5.14 ตัวอย่างการแย่งบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก

ในสถานการณ์เก็บบอล ซึ่งลูกบอลไม่ได้อยู่ในการครอบครองของหุ่นยนต์ตัวใด ซึ่งในสถานการณ์นี้หุ่นยนต์ของทีมจะต้องวิ่งไปเก็บบอลก่อนหุ่นยนต์ฝ่ายตรงข้าม ซึ่งเป็นหน้าที่ของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก โดยหุ่นยนต์ตำแหน่งอื่นๆ จะทำหน้าที่คอยสนับสนุนการเข้าเก็บบอล

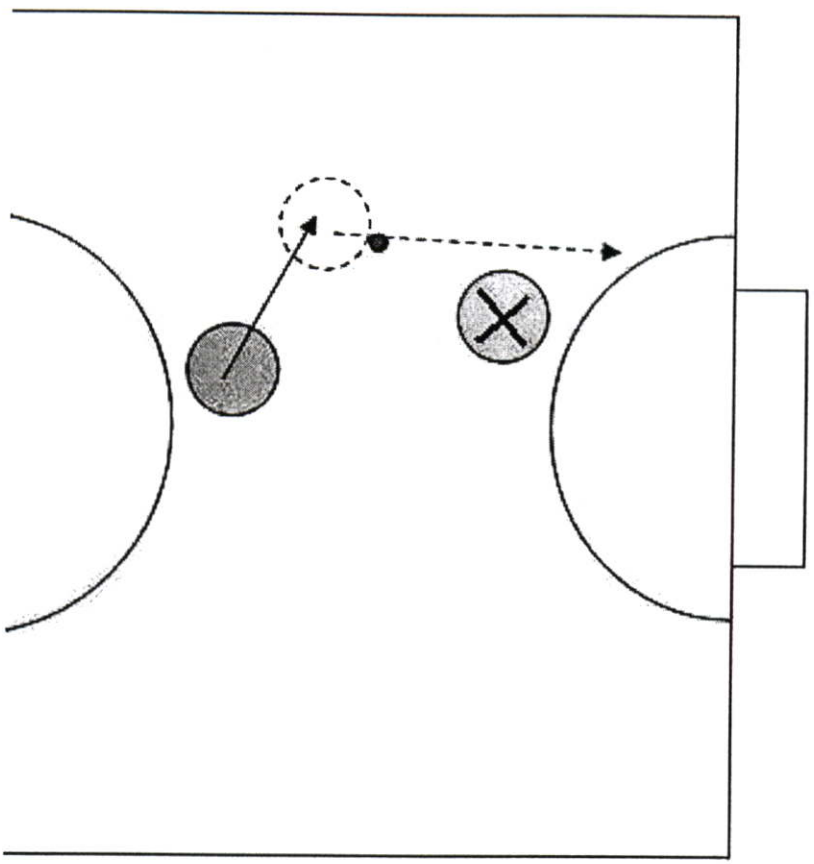
จากรูปที่ 5.15 สำหรับหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกในสถานการณ์เก็บบอลนั้นจะมีรูปแบบคล้ายคลึงกับการนำลูกฟุตบอลออกจากเขตประตูของหุ่นยนต์ตำแหน่งผู้รักษาประตูเกือบทุกประการ แต่วิถีของลูกฟุตบอลนั้นไม่ได้สร้างจากจุดกึ่งกลางประตูของฝ่ายตัวเองกับจุดตำแหน่งลูกฟุตบอล แต่สร้างจากตำแหน่งลูกฟุตบอลกับจุดกึ่งกลางประตูของฝ่ายตรงข้าม โดยแสดงการเข้าครอบครองลูกฟุตบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกในรูปที่ 5.16 ทำให้หุ่นยนต์เล่นฟุตบอลตัวไล่บอลและรุกสามารถอยู่ในสถานะที่ยิงประตูได้ตลอดเวลา แต่ในความเป็นจริงนั้นจะมีหุ่นยนต์ฝ่ายตรงข้ามเข้ามาขวางวิถีการยิงอยู่เสมอ ทำให้ไม่สามารถยิงประตูได้ จะต้องกำหนดวิถีของลูกบอลให้อ้อมหุ่+นยนต์ฝ่ายตรงข้ามไปก่อนเพื่อหามุมยิงที่ไม่มีหุ่นยนต์ฝ่ายตรงข้ามมาขวาง โดยรูปที่ 5.17 แสดงตัวอย่างการเข้าครอบครองลูกฟุตบอลโดยกำหนดวิถีอ้อมหุ่นยนต์ฝ่ายตรงข้าม



รูปที่ 5.15 แผนภาพการทำงานสำหรับหุ่นยนต์ตำแหน่งตัวไล่บอลและรุกในสถานการณ์เก็บบอล

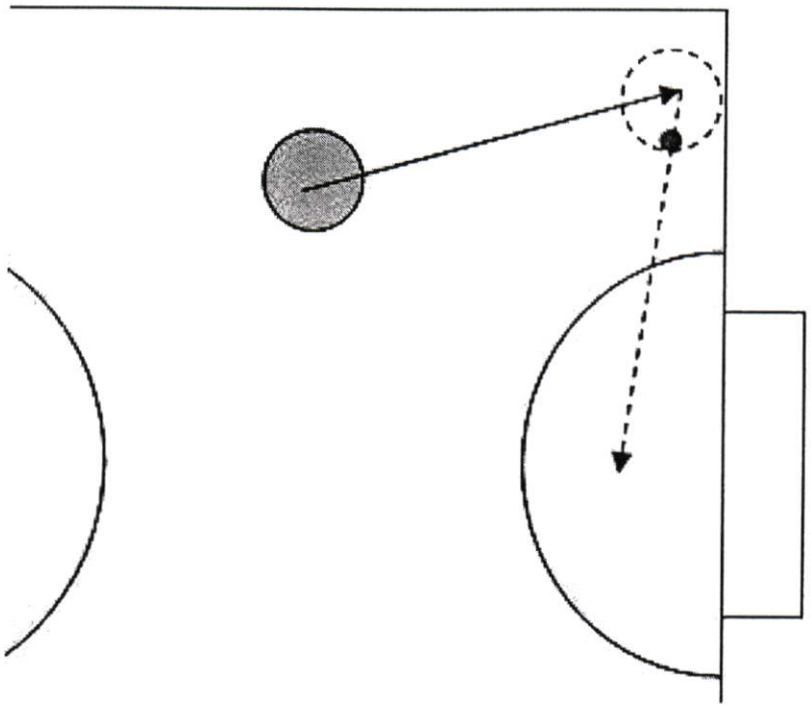


รูปที่ 5.16 ตัวอย่างการเข้าครอบครองลูกฟุตบอลของหุ่นยนต์ตำแหน่งตัวไล่บอลและรุก



รูปที่ 5.17 ตัวอย่างการเข้าครอบครองลูกฟุตบอลโดยกำหนดวิถีอ้อมหุ่นยนต์ฝ่ายตรงข้าม

ในกรณีที่ลูกฟุตบอลอยู่ในเขตมุมของสนามด้านฝ่ายตรงข้าม จำเป็นจะต้องกำหนดวิถีของลูกฟุตบอลใหม่ แทนที่จะสร้างจากจุดตำแหน่งลูกฟุตบอลไปยังจุดกึ่งกลางประตูฝ่ายตรงข้าม เพราะว่าการกำหนดวิถีแบบเดิมนั้นทำให้ไม่สามารถอยู่ในสถานะที่จะยิงประตูได้ เพราะมุมเพื่อการยิงประตูมีน้อยมาก จะต้องนำลูกฟุตบอลมาอยู่ตำแหน่งที่มีมุมในการยิงประตูมากขึ้น ดังนั้นจึงสร้างวิถีของลูกฟุตบอลใหม่โดยแก้ไขจุดปลายเดิมที่เป็นจุดกึ่งกลางประตูฝ่ายตรงข้าม ให้ขยับมาด้านในสนามอีกประมาณ 200 มิลลิเมตรซึ่งแม้ว่าจะเป็นตำแหน่งที่ยิงประตูไม่ได้ แต่ว่าการกำหนดวิถีแบบนี้ทำให้หุ่นยนต์เล่นฟุตบอลตำแหน่งตัวไล่บอลและรุกแตะบอลไปยังตำแหน่งที่มีมุมยิงประตูมากกว่า โดยรูปที่ 5.18 แสดงการเข้าครอบครองลูกฟุตบอลในมุมขอบสนามด้านฝ่ายตรงข้าม



รูปที่ 5.18 ตัวอย่างการเข้าครอบครองลูกฟุตบอลในมุมขอบสนามด้านฝ่ายตรงข้าม

สำหรับสถานการณ์รุกนั้นเป็นสถานการณ์ที่มักเกิดขึ้นหลังจากสถานการณ์เก็บบอล และสลับกลับไปยังสถานการณ์เก็บบอล ซึ่งในสถานการณ์นี้ลูกฟุตบอลจะเข้าใกล้กับหุ่นยนต์เล่นฟุตบอลตำแหน่งตัวไล่บอลและรุกน้อยกว่า 200 มิลลิเมตรแต่การที่ยิงประตูได้นั้น จะต้องมาวิเคราะห์เงื่อนไขในการยิงประตูก่อน

ในการยิงประตูนั้น หุ่นยนต์เล่นฟุตบอลตำแหน่งตัวไล่บอลและรุกจะวิ่งชนกับลูกฟุตบอลด้วยความเร็วสูง เพื่อให้หุ่นยนต์กระแทกกับลูกฟุตบอลจนกลิ้งด้วยความเร็วไปยังทิศทางเข้าหาประตูของฝ่ายตรงข้าม ซึ่งต่างกับในสถานการณ์เก็บบอลที่ความเร็วในการกระแทกจะน้อยกว่าเพราะต้องการให้ลูกฟุตบอลกลิ้งไปด้วยความเร็วที่น้อยกว่า ดังนั้นจะต้องตรวจสอบว่าในสถานะตอนนั้นควรจะปล่อยหุ่นยนต์นำร่องไปยังจุดเดิมหรือควรยิงประตูโดยการตรวจสอบด้านหน้าของหุ่นยนต์ว่าตรงกับลูกฟุตบอลหรือเปล่า โดยใช้สมการ (5.11) คำนวณมุมองศา ซึ่งใช้จุดตำแหน่งของหุ่นยนต์เทียบกับจุดตำแหน่งของลูกฟุตบอลเปรียบเทียบกับมุมองศาด้านหน้าของหุ่นยนต์ในขณะนั้น หากน้อยกว่าค่าที่กำหนดไว้ แสดงว่าหุ่นยนต์อยู่ในสถานะที่สามารถยิงได้

$$\bar{\theta} = \left|\theta_{front} - \theta_{rb}\right| \tag{5.23}$$

โดย $\bar{\theta}$ เป็นมุมส่วนต่างที่คำนวณได้

θ_{front} เป็นมุมด้านหน้าของหุ่นยนต์

θ_{rb} เป็นมุมที่วัดจากจุดตำแหน่งหุ่นยนต์กับจุดตำแหน่งลูกฟุตบอล

โดยสามารถสร้างเงื่อนไขรูปแบบในการยิงประตูได้โดยกำหนดค่าความกว้างมุมที่ต้องการ เช่น หากต้องการให้หุ่นยนต์เล่นฟุตบอลตำแหน่งตัวไล่บอลและรุกเน้นที่ความแม่นยำในการยิงประตู ก็ควรตั้งค่าความกว้างมุมให้ต่ำ แต่ถ้าต้องการการยิงประตูจำนวนครั้งมากๆ ก็ตั้งค่าเริ่มต้นให้มีค่าที่สูง

5.4 การคำนวณข้อมูลเพื่อประสิทธิภาพการเล่น

การคำนวณในข้างต้นนั้นมีจุดประสงค์เพื่อให้หุ่นยนต์เล่นฟุตบอลตามเป้าหมายที่ได้กำหนดไว้ แต่ในความเป็นจริงแล้วข้อมูลที่ได้จากการคำนวณเหล่านั้นก็ยังไม่สมบูรณ์และยังต้องคำนวณเพื่อประสิทธิภาพที่ดีอีกครั้ง

สิ่งที่ต้องคำนวณเพิ่มก็คือการคำนวณแก้ไข้จุดนำร่องที่ได้จากการคำนวณข้างต้น เพราะว่าจุดนำร่องนั้นยังมิได้ตรวจสอบว่ามีสิ่งกีดขวางในการวิ่งหรือไม่ ซึ่งหากไม่คำนวณจุดนำร่องใหม่หุ่นยนต์เล่นฟุตบอลจะวิ่งไปยังจุดเป้าหมายโดยมิได้คำนึงถึงสิ่งกีดขวางเลย ซึ่งจะทำให้เกิดการชนทำให้เกิดความเสียหายและส่งผลให้เกิดการผิดกติกาอีกด้วย

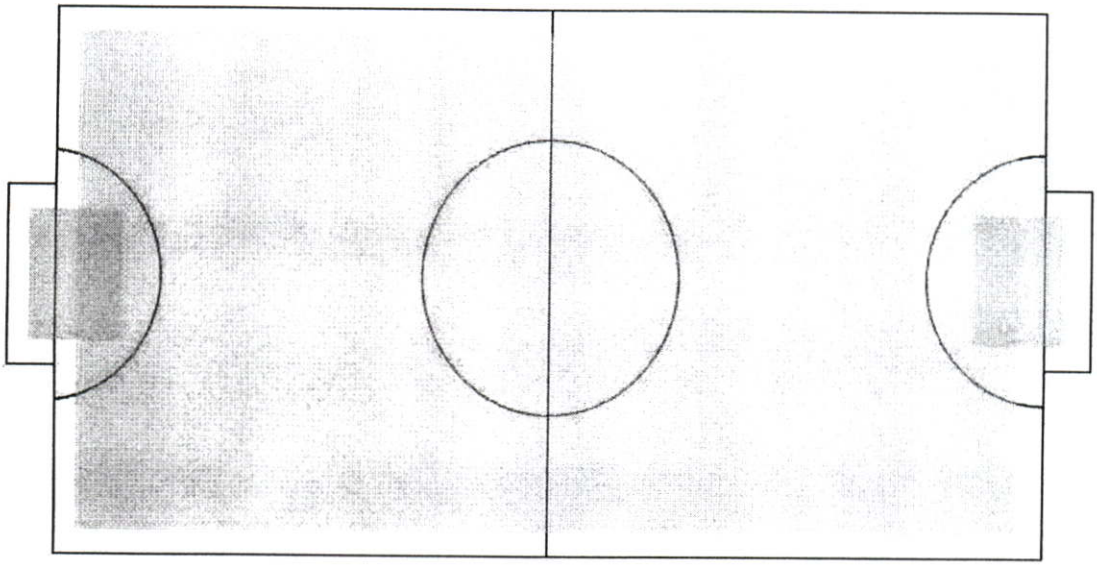
นอกจากสิ่งกีดขวางที่ต้องคำนวณแล้ว ตามกติกานั้นจะอนุญาตให้หุ่นยนต์ฝ่ายเดียวกันเข้าไปยังเขตประตูของฝ่ายตัวเองได้เพียงตัวเดียว ในที่นี้คือหุ่นยนต์ตำแหน่งผู้รักษาประตู และอนุญาตให้หุ่นยนต์ฝ่ายตัวเองเข้าไปยังเขตประตูของฝ่ายตรงข้ามเพียงตัวเดียวเช่นกัน ดังนั้นจะต้องคำนวณแก้ไขจุดนำร่องให้ไม่ทำผิดกติกา นอกจากนี้อาจจะมีโอกาสที่การคำนวณข้างต้นคำนวณจุดนำร่องอยู่นอกพื้นที่การแข่งขันหรือจุดนำร่องไม่อยู่ในจุดที่สามารถเล่นได้ จำเป็นต้องคำนวณให้จุดนำร่องเหล่านี้เป็นจุดนำร่องที่สมบูรณ์ให้ได้

5.4.1 การคำนวณจุดที่อยู่นอกเขตการเล่น

การคำนวณแก้ไขจุดนำร่องที่อยู่นอกเขตการเล่นเป็นสิ่งจำเป็น เพราะการคำนวณจุดนำร่องเบื้องต้นนั้นอาจทำให้หุ่นยนต์วิ่งไปยังนอกเขตการเล่น ซึ่งทำให้วิ่งชนกำแพงหรือประตูได้ ดังนั้นจุดที่ผิดพลาดเหล่านี้จะต้องถูกแก้ไขทั้งหมด

เบื้องต้นนั้นจุดนำร่องทั้งหมดนั้นจะต้องอยู่ในสนาม ซึ่งจะต้องคำนวณเผื่อรัศมีของหุ่นยนต์ด้วย โดยรูปที่ 5.19 แสดงขอบเขตของจุดนำร่องในสนามสำหรับหุ่นยนต์เล่นฟุตบอล แต่ในกรณีที่หุ่นยนต์เล่นฟุตบอลตำแหน่งผู้รักษาประตูจะต้องอ้อมไปด้านหลังลูกฟุตบอลซึ่งอยู่ในเขตประตู หรือในกรณีที่หุ่นยนต์เล่นฟุตบอลตำแหน่งตัวไล่บอลและรุกเข้าไปทำประตูในเขตประตูของฝ่าย

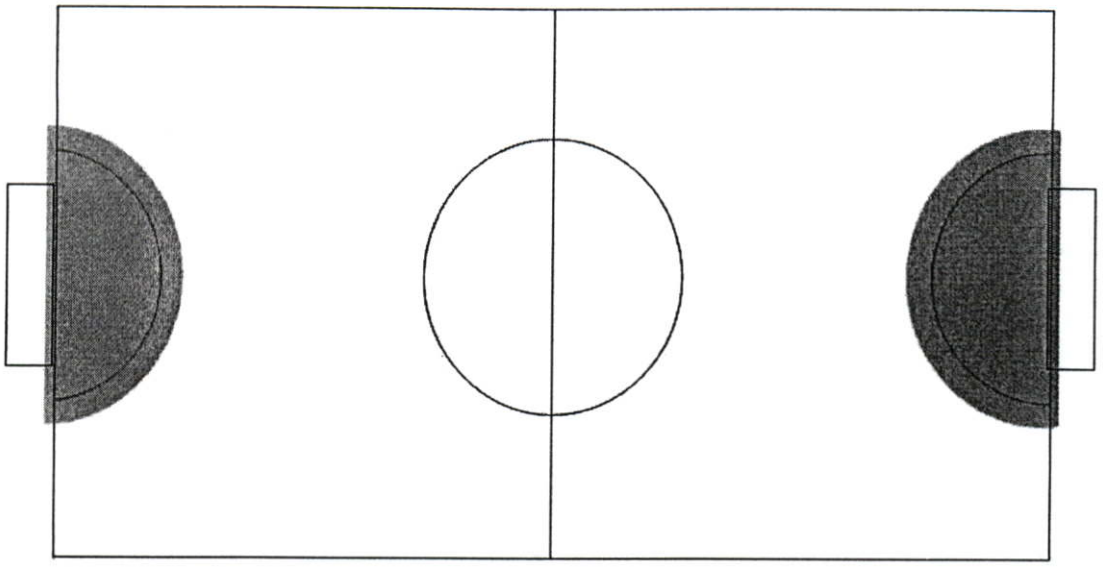
ตรงข้าม จะต้องกำหนดขอบเขตขยายให้กับสองกรณีนี้เป็นพิเศษ เพื่อที่จะทำให้หุ่นยนต์นั้นสามารถวิ่งเข้าไปในส่วนภายในของประตูได้



รูปที่ 5.19 ขอบเขตของจุดนำร่องในสนามสำหรับหุ่นยนต์เล่นฟุตบอล

5.4.2 การคำนวณจุดที่อยู่ในเขตประตู

การคำนวณแก้ไขจุดนำร่องที่อยู่ในเขตประตูนั้นเป็นสิ่งจำเป็น เพราะมิเช่นนั้นหุ่นยนต์อาจวิ่งเข้าไปในเขตประตูจนเกิดการผิดกติกาขึ้น ดังนั้นจะต้องคำนวณแก้ไขจุดนำร่องของหุ่นยนต์เล่นฟุตบอลที่ไม่ได้เป็นตำแหน่งผู้รักษาให้ไม่ให้วิ่งเข้าไปในเขตประตูของทีม และจะต้องคำนวณแก้ไขจุดนำร่องของหุ่นยนต์เล่นฟุตบอลที่ไม่ได้เป็นตำแหน่งตัวไล่บอลและรุกให้ไม่ให้วิ่งเข้าไปในเขตประตูของฝ่ายตรงข้าม



รูปที่ 5.20 ขอบเขตนอกเขตประตูของจุดนำร่องหรับหุ่นยนต์เล่นฟุตบอล

ในรูปที่ 5.20 ซึ่งแสดงขอบเขตนอกเขตประตูของจุดนำร่องหรับหุ่นยนต์เล่นฟุตบอล สังเกตว่าเขตที่ถูกกำหนดนั้นจะใหญ่กว่าเขตประตูเล็กน้อยเพราะว่าจุดนำร่องนั้นแทนจุดศูนย์กลางของหุ่นยนต์ ดังนั้นต้องเผื่อขนาดของหุ่นยนต์ไว้ด้วย เพื่อไม่ให้หุ่นยนต์วิ่งเข้าไปในเขตประตู

5.4.3 การคำนวณจุดนำร่องเพื่อวิ่งอ้อมเขตประตู

แม้ว่าจุดนำร่องจะได้รับการแก้ไขให้อยู่นอกเขตประตูแล้ว แต่กรณีที่หุ่นยนต์วิ่งตัดเขตประตูยังมีโอกาสเกิดขึ้น ดังนั้นจะต้องตรวจสอบว่าทิศทางที่หุ่นยนต์วิ่งไปนั้นตัดผ่านเขตประตูหรือไม่และจะต้องคำนวณจุดนำร่องใหม่ให้หุ่นยนต์วิ่งลัดเลาะตามขอบเขตประตูไปยังจุดเป้าหมายให้ได้

การตรวจสอบว่าการวิ่งของหุ่นยนต์นั้นจะวิ่งตัดเขตประตูหรือไม่ สามารถคำนวณได้โดยใช้สมการ (5.4)-(5.10) โดยสร้างเส้นตรงใช้จุดตั้งต้นเป็นจุดตำแหน่งของหุ่นยนต์เล่นฟุตบอลและจุดปลายเป็นจุดนำร่อง และใช้วงกลมเขตประตูที่มีขนาดใหญ่กว่าขนาดเดิม 180 มิลลิเมตรเป็นวงกลมทดสอบ ทำให้สามารถคำนวณจุดตัดวงกลมได้ เมื่อทดสอบว่าวิถีการวิ่งของหุ่นยนต์เล่นฟุตบอลตัดกับเขตประตู จะต้องคำนวณจุดนำร่องใหม่

เมื่อพบว่าวิถีการวิ่งตัดผ่านเขตประตู จะต้องแก้ไขวิถีการวิ่งโดยเพิ่มองศาของเส้นตรงอีก 45 องศาให้กับวิถีการวิ่ง โดยเริ่มจากการคำนวณความชันใหม่ของเส้นตรง

$$m_n = \begin{cases} m_o + \tan 45^\circ; & m_o \geq 0 \\ m_o - \tan 45^\circ; & m_o < 0 \end{cases} \tag{5.24}$$

โดย m_o เป็นความชันของวิถีการวิ่งเดิม

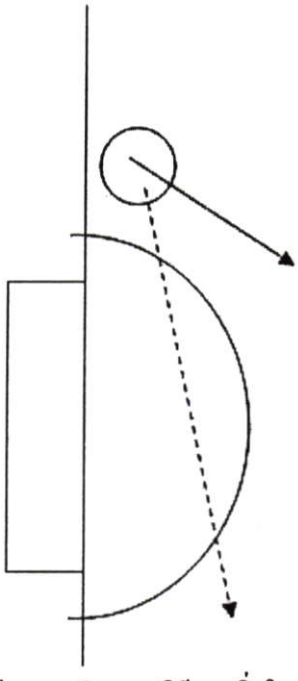
m_n เป็นความชันของวิถีการวิ่งใหม่

ซึ่งสามารถคำนวณสมการวิถีการวิ่งใหม่โดยการแทนค่าจุดตำแหน่งของหุ่นยนต์ลงไปในสมการ (5.1) เพื่อหาค่า c จะได้สมการวิถีการวิ่งใหม่เท่ากับ

$$y = m_n(x - x_{robot}) + y_{robot} \tag{5.25}$$

โดย x_{robot} เป็นจุดตำแหน่งของหุ่นยนต์ในแนวแกน X

y_{robot} เป็นจุดตำแหน่งของหุ่นยนต์ในแนวแกน Y



รูปที่ 5.21 ตัวอย่างการคำนวณวิถีการวิ่งใหม่เพื่อวิ่งอ้อมเขตประตู

เมื่อได้วิถีการวิ่งแล้ว จุดนำร่องจะต้องคำนวณใหม่ทดแทนจุดนำร่องเดิมโดย

$$x_{nt} = x_{robot} + [L_{nt} \cos\theta_n] \tag{5.26}$$

$$y_{nt} = y_{robot} + [L_{nt} \sin\theta_n] \tag{5.27}$$

โดย x_{nt} เป็นตำแหน่งในแนวแกน X ของจุดนำร่องใหม่

y_{nt} เป็นตำแหน่งในแนวแกน Y ของจุดนำร่อง

L_{nt} ระยะห่างของจุดนำร่อง

θ_n เป็นมุมของเส้นทางการพาลูกฟุตบอลออกนอกเขตประตู เท่ากับ $\arctan(m_n)$

5.4.4 การคำนวณจุดนำร่องเพื่อวิ่งอ้อมสิ่งกีดขวาง

ในการคำนวณจุดนำร่องของหุ่นยนต์เบื้องต้นนั้นไม่มีการคำนึงถึงสิ่งกีดขวาง ซึ่งหากไม่มีการคำนวณจุดนำร่องใหม่โดยคำนึงสิ่งกีดขวางแล้ว หุ่นยนต์เล่นฟุตบอลจะวิ่งโดยไม่สนใจสิ่งกีดขวาง(หุ่นยนต์ฝ่ายตรงข้ามและหุ่นยนต์ฝ่ายเดียวกัน) ซึ่งอาจจะเกิดการชนกันเกิดขึ้น และหากวิ่งชนหุ่นยนต์เล่นฟุตบอลฝ่ายตรงข้ามโดยไม่มีการลดความเร็วก็จะทำให้เกิดการผิดกติกาเกิดขึ้น ดังนั้นการคำนวณจุดนำร่องเพื่อวิ่งอ้อมสิ่งกีดขวางเป็นสิ่งที่จำเป็นอย่างยิ่ง

การคำนวณจุดนำร่องให้อ้อมสิ่งกีดขวางนั้นสามารถคำนวณได้โดยคำนวณสมการเส้นตรงของวิถีการวิ่งและจุดตัดวงกลมสิ่งกีดขวางก่อน จากนั้นให้มาคำนวณจุดกึ่งกลางส่วนของเส้นตรงที่อยู่ภายในวงกลมได้จากสมการ (5.28) และ (5.29)

$$x_b = \frac{xi_1 + xi_2}{2} \tag{5.28}$$

$$y_b = \frac{yi_1 + yi_2}{2} \tag{5.29}$$

โดย x_b เป็นจุดกึ่งกลางของส่วนของเส้นตรงภายในวงกลมในแกน X

y_b เป็นจุดกึ่งกลางของส่วนของเส้นตรงภายในวงกลมในแกน Y

จากนั้นสร้างสมการเส้นตรงที่ตั้งฉากกับเส้นตรงที่เป็นวิถีการวิ่งเดิม โดยสมการที่ตั้งฉากสามารถคำนวณค่าความชันได้โดยใช้สมการ (5.26)

$$m_n = \frac{1}{-m_o} \tag{5.26}$$

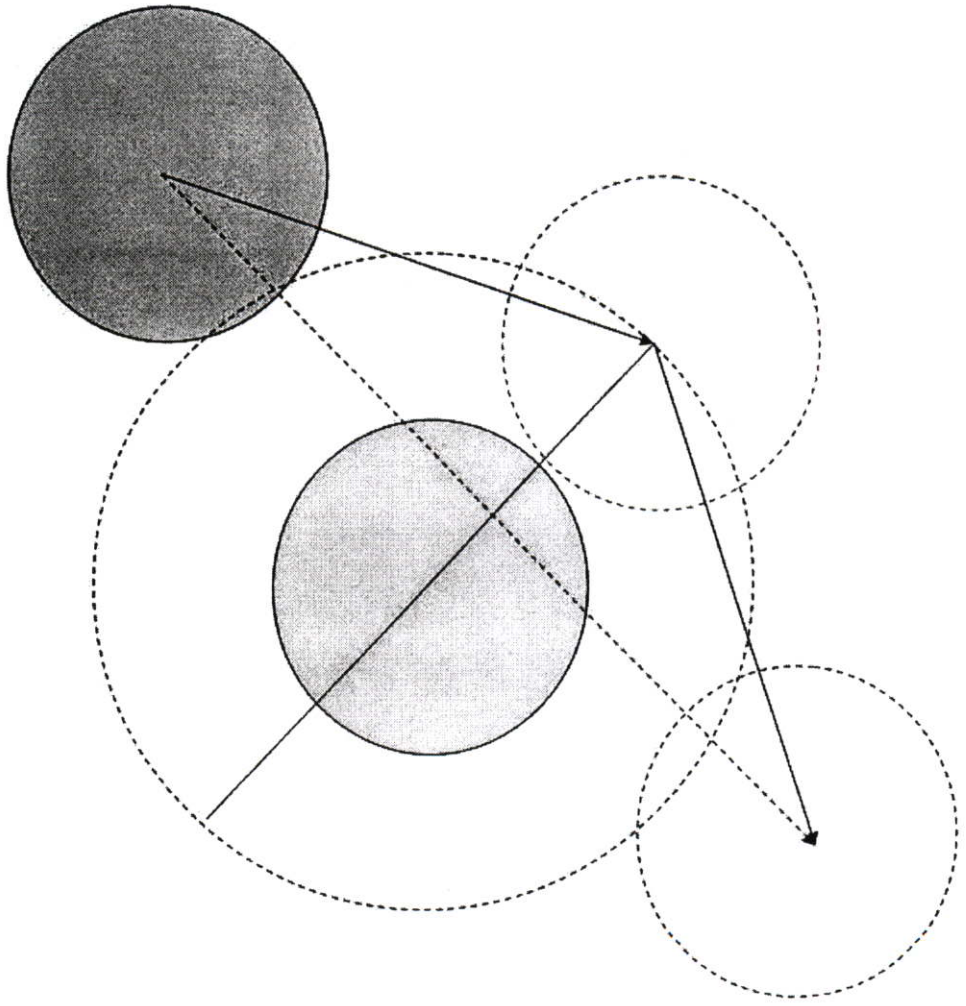
โดย m_n เป็นค่าความชันของเส้นตรงที่ตั้งฉาก

m_o เป็นค่าความชันของเส้นตรงเดิม

เมื่อคำนวณค่าความชันได้แล้ว สามารถนำค่าจุดกึ่งกลางของส่วนของเส้นตรงไปแทนในสมการ (5.1) เพื่อหาค่า c สุดท้ายแล้วจะได้สมการเส้นตรงใหม่ดังนี้

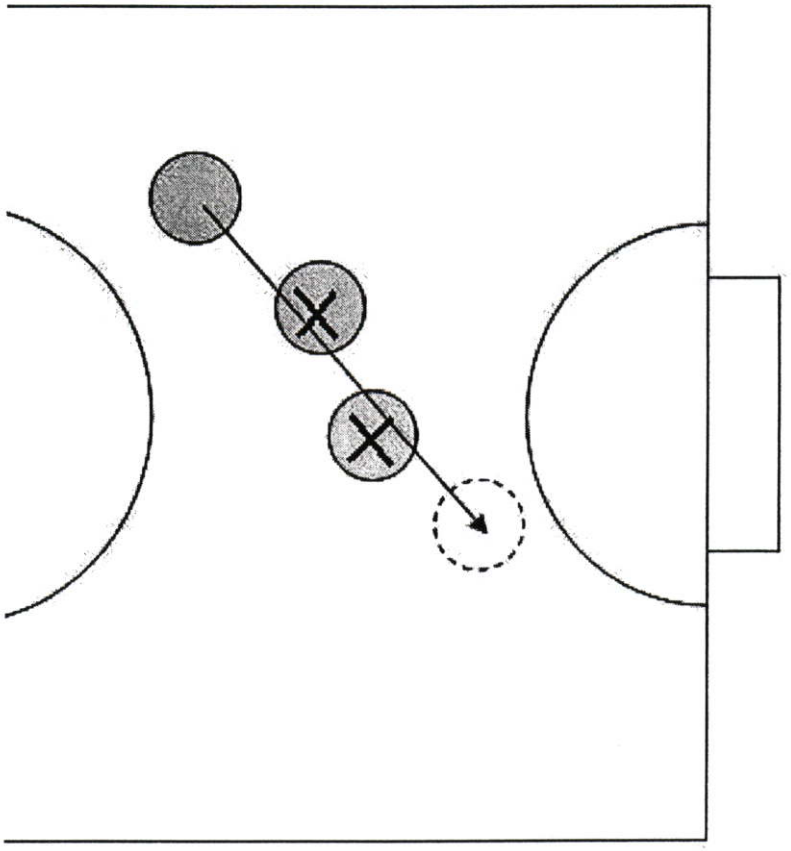
$$y = -\frac{1}{m}x + y_b + \frac{x_b}{m} \tag{5.27}$$

จากนั้นทำการหาจุดตัดบนวงกลมสิ่งกีดขวาง แต่เปลี่ยนให้ใช้เส้นที่ตั้งฉากกับวิถีการวิ่ง ก็จะได้จุดตัดวงกลมมาสองจุด ซึ่งทั้งสองจุดนี้คือจุดที่หุ่นยนต์เล่นฟุตบอลจะวิ่งอ้อมวงกลมไปยังจุดเป้าหมายได้ดังที่แสดงในรูปที่ 5.22 โดยในกรณีนี้ ให้เลือกจุดที่ระยะทางวัดจากจุดตำแหน่งหุ่นยนต์ที่ใกล้กว่าโดยใช้สมการ (5.22) ซึ่งเมื่อสิ้นสุดกระบวนการคำนวณแล้ว จะได้จุดนำร่องที่วิ่งสิ่งกีดขวางอย่างสมบูรณ์

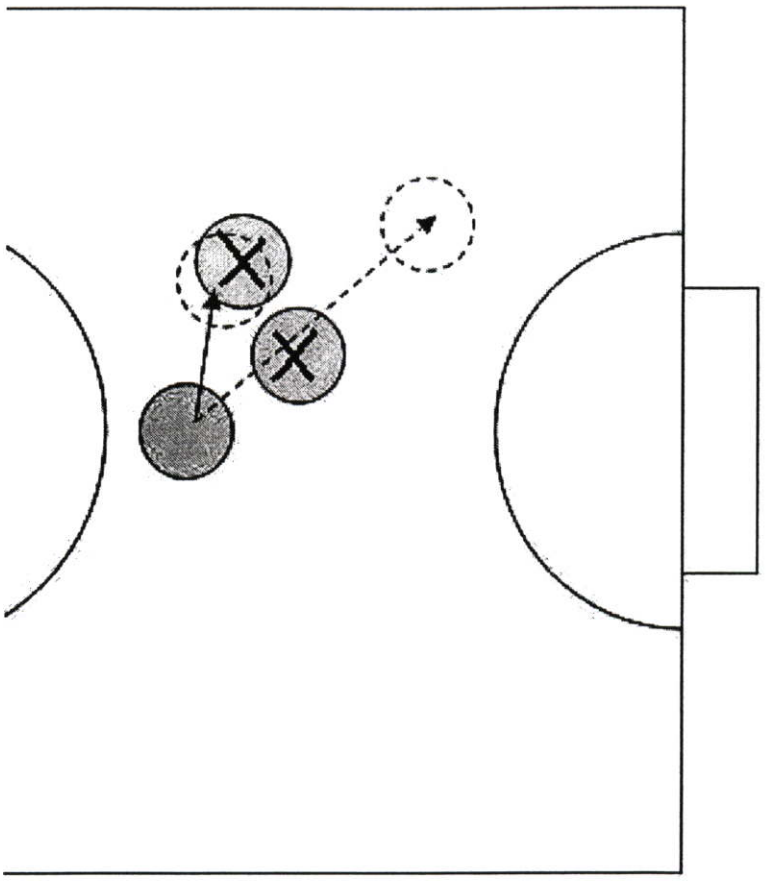


รูปที่ 5.22 ตัวอย่างการคำนวณวิถีการวิ่งใหม่เพื่อวิ่งอ้อมสิ่งกีดขวาง

แต่ในความเป็นจริงแล้ว มักเกิดกรณีที่หาจุดนำร่องวิ่งอ้อมสิ่งกีดขวางแล้ว แต่ก็ยังไปติดกับสิ่งกีดขวางอื่นอีก หรือจุดนำร่องดั้งเดิมนั้นติดสิ่งกีดขวางหลายตัวด้วยกัน ดังนั้นจะต้องคำนวณจุดนำร่องโดยคำนวณสิ่งกีดขวางอื่นๆ ด้วย โดยรูปที่ 5.23 แสดงตัวอย่างจุดนำร่องตัดผ่านสิ่งกีดขวางหลายตัวและรูปที่ 5.24 แสดงตัวอย่างจุดนำร่องที่อ้อมสิ่งกีดขวางแรกแล้วยังไปทับกับสิ่งกีดขวางอื่นๆ

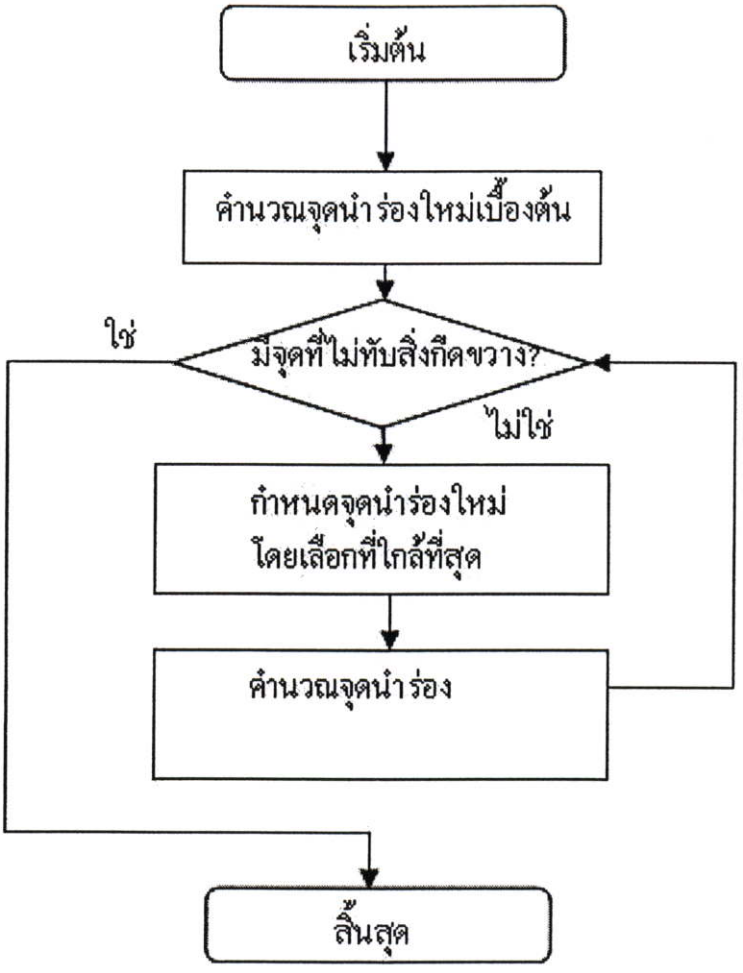


รูปที่ 5.23 ตัวอย่างจุดนำร่องตัดผ่านสิ่งกีดขวางหลายตัว

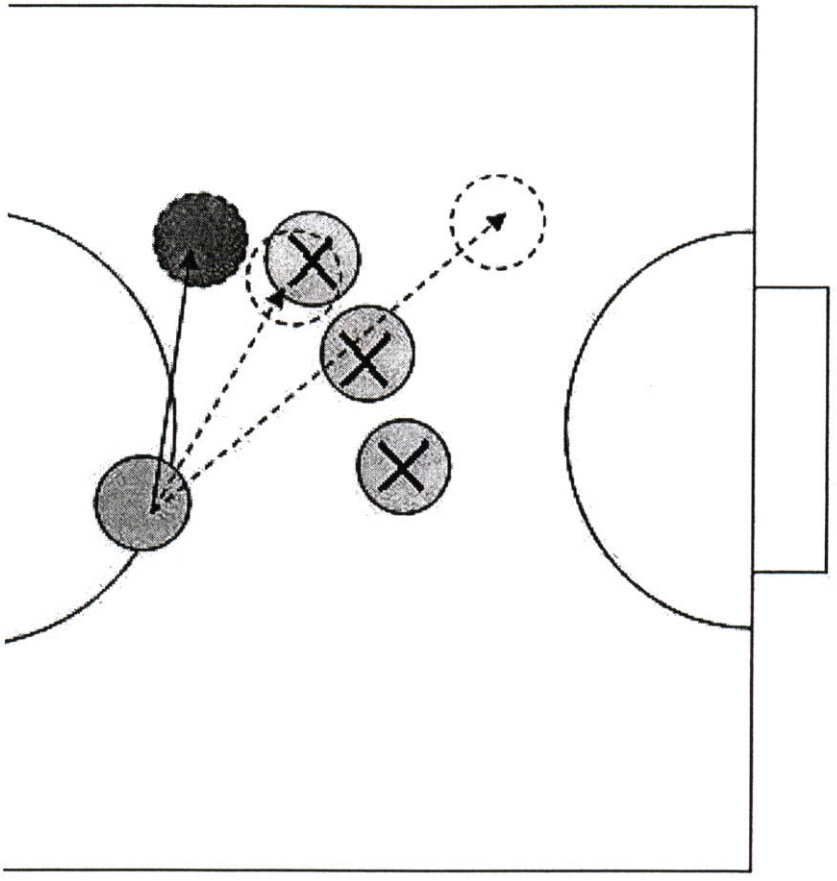


รูปที่ 5.24 ตัวอย่างจุดนำร่องที่อ้อมสิ่งกีดขวางแรกแล้วยังไปทับกับสิ่งกีดขวางอื่นๆ

รูปที่ 5.25 แสดงการคำนวณหาจุดนำร่องสำหรับวิ่งอ้อมสิ่งกีดขวาง เริ่มที่การจัดลำดับสิ่งกีดขวางโดยวัดจากระยะทาง จากนั้นก็คำนวณจุดนำร่องอ้อมสิ่งกีดขวางโดยเริ่มจากสิ่งกีดขวางที่ใกล้ที่สุด โดยจุดๆ นั้นจะต้องไม่ทับอยู่บนสิ่งกีดขวางอื่นๆ แต่หากจุดที่คำนวณทับสิ่งกีดขวางทั้งสองจุดให้คำนวณใหม่ แล้วจากนั้นคำนวณหาจุดนำร่องที่ไม่ทับกับสิ่งกีดขวางไปเรื่อยๆ จนกว่าจะได้จุดนำร่องที่สมบูรณ์ โดยรูปที่ 5.26 แสดงตัวอย่างจุดนำร่องที่อ้อมสิ่งกีดขวางอย่างสมบูรณ์



รูปที่ 5.25 การคำนวณหาจุดนำร่องสำหรับวิ่งอ้อมสิ่งกีดขวาง



รูปที่ 5.26 ตัวอย่างจุดนำร่องที่อ้อมสิ่งกีดขวางอย่างสมบูรณ์

5.4.5 การคำนวณคุณสมบัติของจุดนำร่อง

หากสังเกตที่ตาราง 3.6 จะเห็นได้ว่ามีข้อมูล PATHTYPE_1, PATHTYPE_2, PATHTYPE_3, PATHTYPE_4 และ PATHTYPE_5 เป็นคุณสมบัติของจุดนำร่อง โดยข้อมูลนี้นำไปใช้กำหนดความเร็วสูงสุดในการนำร่องของหุ่นยนต์ ซึ่งคุณสมบัติทั้งหมดแสดงไว้ในตาราง 5.9

ตารางที่ 5.9 คุณสมบัติของจุดนำร่อง

ชนิด	ความเร็ว	จุดประสงค์
NORMAL	100%	ใช้นำร่องทั่วไป
STOP	20%	ใช้สำหรับเพื่อความแน่นอนในการหยุด
DRIB	35%	ใช้สำหรับกระแทกลูกบอลไปในทิศทางที่ต้องการ
GOAL	100%	ใช้สำหรับยิงประตู
PREPARE	70%	ใช้สำหรับเข้าสู่จุดเตรียมพร้อม
BALL	20%	ใช้กระแทกลูกบอลไปในทิศทางที่ต้องการอย่างเบาๆ
PASS	35%	ใช้สำหรับกระแทกลูกบอลไปในทิศทางสำหรับส่ง
ENEMY	50%	ใช้สำหรับการประกบสำหรับหุ่นยนต์ตำแหน่งตัวประกบ
TOUCH	50%	ใช้สำหรับการแย่งบอล

ดังนั้นเมื่อคำนวณตำแหน่งจุดนำร่องได้แล้ว จะต้องกำหนดคุณสมบัติของจุดนำร่องให้เหมาะสมกับการวิ่งของหุ่นยนต์เพื่อประสิทธิภาพการวิ่งที่ดีขึ้น ยกตัวอย่างเช่น การยิงประตูนั้น จุดนำร่องที่กำหนดจะอยู่บนเส้นประตูของฝ่ายตรงข้าม หุ่นยนต์ของทีมจะต้องวิ่งกระแทกบอลด้วยความแรงสูงสุดเพื่อให้ลูกฟุตบอลพุ่งไปในทิศทางนั้นอย่างแรง ดังนั้นคุณสมบัติของจุดนำร่องจะต้องถูกกำหนดเป็น GOAL เพื่อให้หุ่นยนต์วิ่งด้วยความเร็วสูงตามจุดประสงค์

ในบทที่ 6 ซึ่งเป็นบทต่อไปจะอธิบายการนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลซึ่งเป็นการคำนวณขั้นตอนสุดท้าย โดยเนื้อหาประกอบด้วยการอธิบายแบบจำลองระบบขับเคลื่อนของหุ่นยนต์ การควบคุมที่ใช้ในการนำร่อง และระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอล

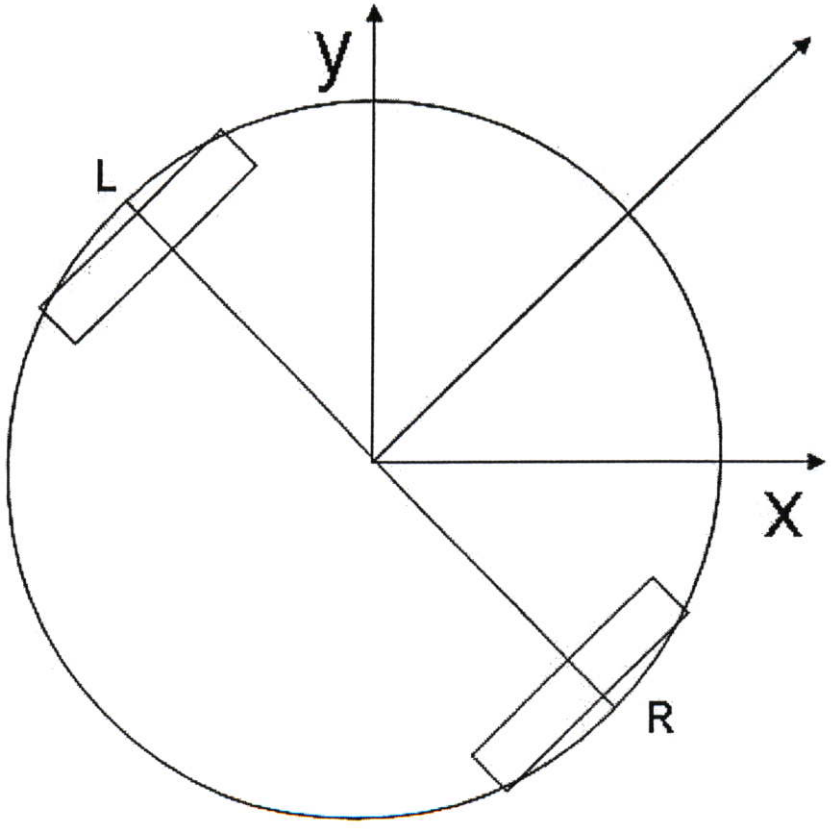
บทที่ 6
การนำร่องสำหรับหุ่นยนต์เล่นฟุตบอล

ในบทนี้จะอธิบายถึงการนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลที่มี 2 ล้อ 3 ล้อและ 4 ล้อซึ่งทำหน้าที่แปลงพิกัดตำแหน่งจุดนำร่องและคุณสมบัติจุดนำร่องที่อยู่ชั้นจุดประสงค์ให้กลายเป็นความเร็วของล้อ เพื่อบังคับให้หุ่นยนต์วิ่งไปยังเป้าหมายได้อย่างถูกต้อง เก้าอี้รถล้อเลื่อนคนพิการ

6.1 แบบจำลองระบบขับเคลื่อนของหุ่นยนต์

ระบบขับเคลื่อนหุ่นยนต์ที่ใช้ในการแข่งขันหุ่นยนต์นั้น แบ่งได้เป็น 3 ชนิดคือ แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ แบบเคลื่อนที่ทุกทิศทาง 3 ล้อ และแบบเคลื่อนที่ทุกทิศทาง 4 ล้อ

6.1.1 ระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 6.1 แบบจำลองระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

แบบจำลองของหุ่นยนต์ชนิดนี้จะมีลักษณะคล้ายกับเก้าอี้รถล้อเลื่อนคนพิการ คือมีล้อจำนวน 2 ล้ออยู่บนแกนกึ่งกลางของหุ่นยนต์ ซึ่งสามารถเคลื่อนที่เดินหน้าถอยหลังได้โดยการควบคุมความเร็วล้อให้มีความเร็วเท่ากัน หากต้องการเลี้ยวก็สามารถทำได้โดยการควบคุมความเร็วล้อข้างใดข้างหนึ่งให้มีความเร็วน้อยกว่าอีกข้างหนึ่ง และหากต้องการหมุนตัวก็ให้ควบคุมความเร็วล้อเท่าๆ กัน แต่กลับทิศทางการหมุน

ระบบขับเคลื่อนแบบนี้เป็นที่นิยมมากสำหรับหุ่นยนต์เคลื่อนที่โดยทั่วไป เพราะสร้างง่ายและควบคุมง่าย แต่มีจุดอ่อนที่ไม่สามารถเคลื่อนที่ในแนวด้านข้างได้ โดยรูปที่ 6.1 แสดงแบบจำลองระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

สำหรับการหาความเร็วในระยะพิกัดของตัวหุ่นยนต์เอง สามารถหาได้โดยใช้สมการ (6.1)-(6.3)

$$V_{xi} = \frac{Vr + Vl}{2} \tag{6.1}$$

$$V_{yi} = 0 \tag{6.2}$$

$$V_{ti} = \frac{Vr - Vl}{2} \tag{6.3}$$

โดย V_{xi} เป็นความเร็วแนวแกน X ในระยะพิกัดภายในหุ่นยนต์

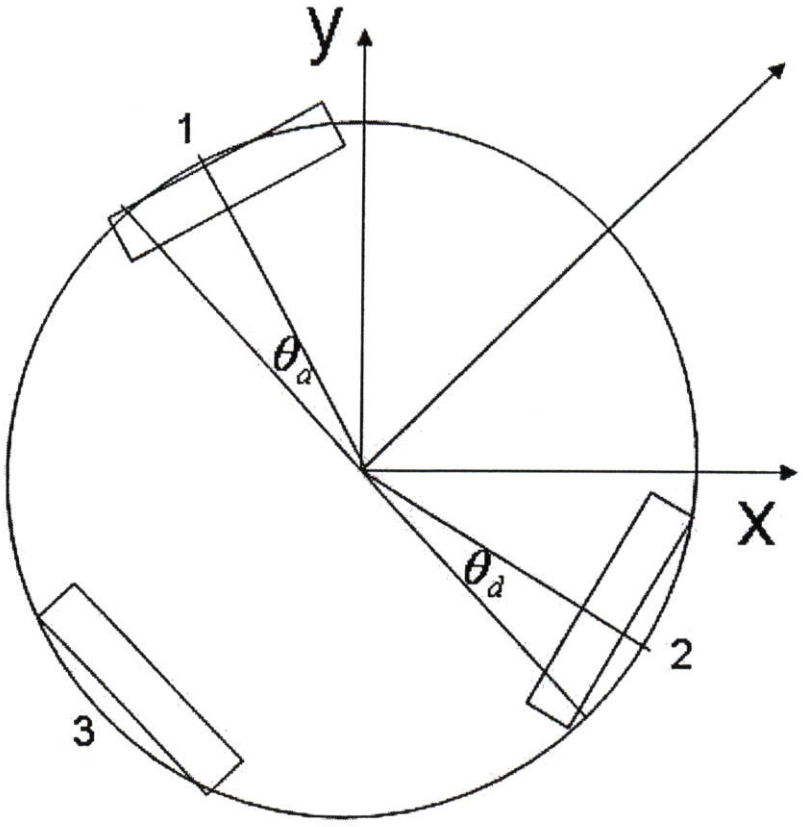
V_{yi} เป็นความเร็วแนวแกน Y ในระยะพิกัดภายในหุ่นยนต์

V_{ti} เป็นความเร็วแนวแกน θ ในระยะพิกัดภายในหุ่นยนต์

Vr เป็นความเร็วของล้อด้านขวา

Vl เป็นความเร็วของล้อด้านซ้าย

6.1.2 ระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ



รูปที่ 6.2 แบบจำลองระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ

แบบจำลองชนิดมีความพิเศษที่สามารถเคลื่อนที่ได้รอบทิศทาง โดยอาศัยล้อแบบเคลื่อนที่ทุกทิศทางซึ่งแสดงในรูปที่ 6.3 มาใช้แทนล้อแบบธรรมดา โดยสามารถควบคุมการวิ่งแบบเดินหน้าถอยหลังและหมุนตัวได้เหมือนกับแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ แต่สามารถเคลื่อนที่ในแนวด้านข้างได้โดยการให้ความเร็วล้อสัมพันธ์กันอย่างเหมาะสม โดยรูปที่ 6.2 แสดงแบบจำลองระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ

สำหรับการหาความเร็วในระยะพิกัดของตัวหุ่นยนต์เอง สามารถหาได้โดยใช้สมการ (6.4)-(6.6)

$$V_{xi} = \frac{(V_1 + V_2) \cdot \cos\theta_d}{3} \tag{6.4}$$

$$V_{yi} = \frac{(V_2 - V_1) \cdot \sin\theta_d}{3} \tag{6.5}$$

$$V_{ti} = \frac{(V_3 + V_2 - V_1)}{3} \tag{6.6}$$

โดย V_{xi} เป็นความเร็วแนวแกน X ในระยะพิกัดภายในหุ่นยนต์

V_{yi} เป็นความเร็วแนวแกน Y ในระยะพิกัดภายในหุ่นยนต์

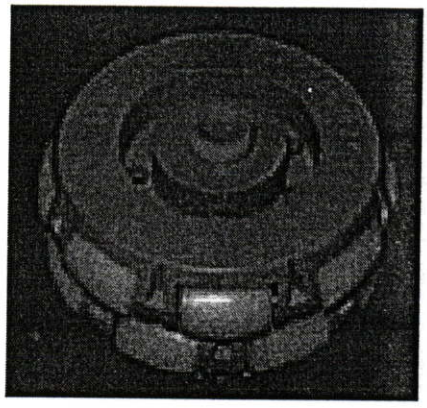
V_{ti} เป็นความเร็วแนวแกน θ ในระยะพิกัดภายในหุ่นยนต์

V_1 เป็นความเร็วของล้อที่ 1

V_2 เป็นความเร็วของล้อที่ 2

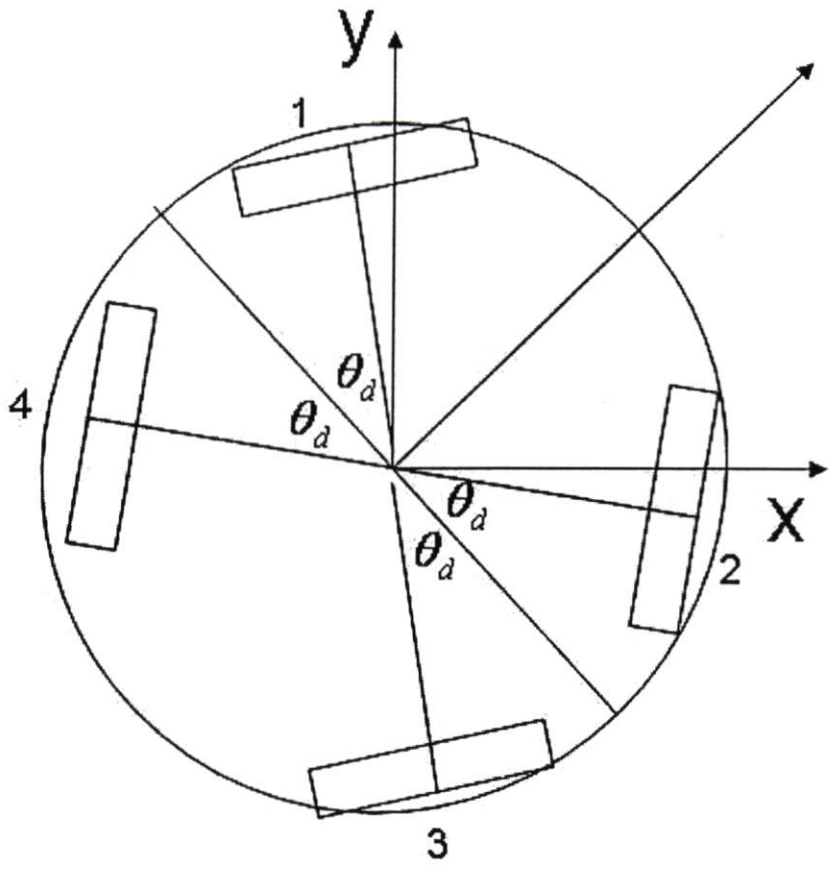
V_3 เป็นความเร็วของล้อที่ 3

θ_d เป็นมุมเบี่ยงเบนของแกนล้อ



รูปที่ 6.3 ล้อเคลื่อนที่ได้ทุกทิศทาง [13]

6.1.3 ระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 4 ล้อ



รูปที่ 6.4 แบบจำลองระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 4 ล้อ

แบบจำลองชนิดนี้มีความพิเศษที่สามารถเคลื่อนที่ได้รอบทิศทางเหมือนกับแบบจำลองแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ โดยเพิ่มล้อด้านหลังขึ้นมาจนกลายเป็น 4 ล้อซึ่งมีผลให้ความสามารถในการเคลื่อนที่แนวด้านข้างดียิ่งขึ้น สามารถควบคุมการวิ่งแบบเดินหน้าถอยหลัง, หมุนตัวและแนวด้านข้างได้เหมือนกับแบบจำลองแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ

สำหรับการหาความเร็วในระยะพิกัดของตัวหุ่นยนต์เอง สามารถหาได้โดยใช้สมการ (6.7)-(6.9)

$$V_{xi} = \frac{(V_1 + V_2 + V_3 + V_4) \cdot \cos\theta_d}{4} \tag{6.7}$$

$$V_{yi} = \frac{(V_2 + V_3 - V_1 - V_4) \cdot \sin\theta_d}{4} \tag{6.8}$$

$$V_{ti} = \frac{(V_2 + V_4 - V_1 - V_3)}{4} \tag{6.9}$$

โดย V_{xi} เป็นความเร็วแนวแกน X ในระยะพิกัดภายในหุ่นยนต์

V_{yi} เป็นความเร็วแนวแกน Y ในระยะพิกัดภายในหุ่นยนต์

V_{ti} เป็นความเร็วแนวแกน θ ในระยะพิกัดภายในหุ่นยนต์

V_1 เป็นความเร็วของล้อที่ 1

V_2 เป็นความเร็วของล้อที่ 2

V_3 เป็นความเร็วของล้อที่ 3

V_4 เป็นความเร็วของล้อที่ 4

θ_d เป็นมุมเบี่ยงเบนของแกนล้อ

6.1.4 แบบจำลองการเคลื่อนที่ของหุ่นยนต์เล่นฟุตบอล

การนำร่องสำหรับแบบจำลองการเคลื่อนที่ของหุ่นยนต์เล่นฟุตบอลนั้น จำเป็นต้องคำนวณหาความเร็วแกนภายในของหุ่นยนต์ก่อน โดยคำนวณได้โดยใช้สมการ (6.1)-(6.9) ตามชนิดของหุ่นยนต์ ซึ่งเมื่อป้อนความเร็วเข้าไปยังล้อของหุ่นยนต์จะทำให้เกิดการเคลื่อนที่เกิดขึ้น สามารถหาความเร็วในการเคลื่อนและมุมที่เคลื่อนที่ไปได้โดยใช้สมการ (6.10) และ (6.11)

$$Vi = \sqrt{V_{xi}^{\ 2} + V_{yi}^{\ 2}} \tag{6.10}$$

$$\theta_i = \arctan\left(\frac{V_{yi}}{V_{xi}}\right) \tag{6.11}$$

โดย Vi เป็นความเร็วในการเคลื่อนที่ของหุ่นยนต์

θ_i เป็นองศาในการเคลื่อนที่ของหุ่นยนต์

จากนั้นก็สามารถหาตำแหน่งและด้านหน้าของหุ่นยนต์ได้โดยใช้นำค่าจากสมการ (6.10) และ (6.11) มาคำนวณโดยใช้สมการ (6.12)-(6.14)

$$x_n = x_o + \left(\cos(\theta_o + \theta_i) \cdot Vi\right) \cdot t \tag{6.12}$$

$$y_n = y_o + \left(\sin(\theta_o + \theta_i) \cdot Vi\right) \cdot t \tag{6.13}$$

$$\theta_n = \theta_o + \left(\frac{V_{ti}}{R_{robot}}\right) \cdot t \tag{6.14}$$

โดย x_n เป็นตำแหน่งในแนวแกน X ซึ่งคำนวณได้ใหม่

x_o เป็นตำแหน่งในแนวแกน X ซึ่งเป็นค่าเดิม

y_n เป็นตำแหน่งในแนวแกน Y ซึ่งคำนวณได้ใหม่

y_o เป็นตำแหน่งในแนวแกน Y ซึ่งเป็นค่าเดิม

θ_n เป็นองศาในการเคลื่อนที่ของหุ่นยนต์ซึ่งคำนวณได้ใหม่

θ_o เป็นองศาในการเคลื่อนที่ของหุ่นยนต์ซึ่งเป็นค่าเดิม

R_{robot} เป็นรัศมีของหุ่นยนต์เล่นฟุตบอล

t เป็นช่วงเวลาที่วัดจากการคำนวณเมื่อครั้งเดิมมาถึงการคำนวณครั้งใหม่

6.1.5 การแปลงความเร็วในพิกัดหุ่นยนต์มาเป็นความเร็วล้อ

การนำร่องจะต้องแปลงคำสั่งที่เป็นความเร็วในพิกัดของหุ่นยนต์ให้กลายเป็นคำสั่งความเร็วล้อ ไม่ว่าจะเป็นการควบคุมโดยมนุษย์หรือควบคุมโดยการนำร่องอัตโนมัติก็ตาม

การควบคุมการนำร่องโดยมนุษย์นั้นคือการป้อนคำสั่งโดยตรง โดยไม่ผ่านการคำนวณโดยใช้ปัญญาประดิษฐ์ โดยการออกคำสั่งจะทำผ่านอุปกรณ์ควบคุมนั้นจะไม่สามารถรู้ได้ว่าการป้อนคำสั่งโดยตรงลงไปยังล้อแต่ละล้อนั้นส่งผลอย่างไร จึงต้องอาศัยฟังชั่นการแปลงความเร็วในการคำนวณ

สำหรับฟังชั่นการแปลงความเร็วของแบบจำลองระบบขับเคลื่อนของหุ่นยนต์เล่นฟุตบอลแบบแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อนั้น แสดงโดยใช้สมการ (6.15) และ (6.16)

$$W_{left} = Vx - Vt \tag{6.15}$$

$$W_{right} = Vx + Vt \tag{6.16}$$

โดย W_{left} เป็นคำสั่งความเร็วล้อด้านซ้าย

W_{right} เป็นคำสั่งความเร็วล้อด้านขวา

Vx เป็นคำสั่งความเร็วในพิกัดแกน X

Vy เป็นคำสั่งความเร็วในพิกัดแกน Y

Vt เป็นคำสั่งความเร็วในพิกัดแกน θ

สำหรับฟังชั่นการแปลงความเร็วของแบบจำลองระบบขับเคลื่อนของหุ่นยนต์เล่นฟุตบอลแบบเคลื่อนที่ทุกทิศทาง 3 ล้อนั้นแสดงโดยสมการ (6.17)-(6.19)

$$W_1 = Vx \cdot \cos(\theta_d) - Vy \cdot \sin(\theta_d) - Vt \tag{6.17}$$

$$W_2 = Vx \cdot \cos(\theta_d) + Vy \cdot \sin(\theta_d) + Vt \tag{6.18}$$

$$W_b = -Vy \cdot \sin(\theta_d) + Vt \tag{6.19}$$

โดย W_1 เป็นคำสั่งความเร็วล้อด้านซ้ายหน้า

W_2 เป็นคำสั่งความเร็วล้อด้านซ้ายขวา

W_b เป็นคำสั่งความเร็วล้อด้านหลัง

และฟังชั่นการแปลงความเร็วของแบบจำลองระบบขับเคลื่อนของหุ่นยนต์เล่นฟุตบอลแบบเคลื่อนที่ทุกทิศทาง 4 ล้อนั้นแสดงโดยสมการ (6.20) และ (6.23)

$$W_1 = Vx \cdot \cos(\theta_d) - Vy \cdot \sin(\theta_d) - Vt \quad (6.20)$$

$$W_2 = Vx \cdot \cos(\theta_d) + Vy \cdot \sin(\theta_d) + Vt \quad (6.21)$$

$$W_3 = Vx \cdot \cos(\theta_d) + Vy \cdot \sin(\theta_d) - Vt \quad (6.22)$$

$$W_4 = Vx \cdot \cos(\theta_d) - Vy \cdot \sin(\theta_d) + Vt \quad (6.23)$$

โดย W_3 เป็นคำสั่งความเร็วล้อด้านซ้ายหลัง

W_4 เป็นคำสั่งความเร็วล้อด้านซ้ายหลัง

6.2 การควบคุมที่ใช้ในการนำร่อง

ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลเป็นสิ่งจำเป็นต่อการให้หุ่นยนต์ทำงานตามจุดประสงค์ ซึ่งจำเป็นจะต้องแปลงข้อมูลที่เป็นจุดตำแหน่งนำร่องที่ชั้นจุดประสงค์มาเป็นคำสั่งควบคุมล้อของหุ่นยนต์เล่นฟุตบอล

ระบบนำร่องเปรียบได้กับระบบควบคุมทั่วไปที่มีจุดประสงค์ให้หุ่นยนต์วิ่งไปยังจุดนำร่องให้ได้มีประสิทธิภาพดีที่สุด ซึ่งนอกจากจะต้องควบคุมหุ่นยนต์วิ่งไปหยุดที่ตำแหน่งนำร่องแล้วจะต้องควบคุมด้านหน้าของหุ่นยนต์หันไปทางที่ได้กำหนดไว้ได้ด้วย

6.2.1 การควบคุมการหันหน้าแบบพีไอดี (PID)

ตัวควบคุมแบบพีไอดี (PID) สามารถนำมาประยุกต์ใช้กับส่วนหนึ่งของการนำร่องได้ โดยใช้ในการควบคุมการหันหน้าโดยการตั้งค่าเป้าหมายเป็นค่าองศาที่ต้องการ และให้ตัวควบคุมนั้นส่งคำสั่งความเร็วของล้อซึ่งแปลงมาจากความเร็วเชิงมุมของหุ่นยนต์ไปบังคับให้หุ่นยนต์เล่นฟุตบอลหันหน้าไปในทิศทางที่ต้องการ

สามารถเขียนระบบควบคุมสำหรับควบคุมการหันหน้าแบบพีไอดีได้โดยใช้สมการ (6.24)

$$u_{\phi_p}(nT) = u_{\phi_p}(nT - T) + \left(K_{\phi_p} e_\theta(nT) + K_{\phi_i} e_{\phi_i}(nT) + K_{\phi_d} e_{\phi_d}(nT)\right) \quad (6.24)$$

และ

$$e_\theta(nT) = \theta_s - \theta_m(nT) \tag{6.25}$$

$$e_{\theta d}(nT) = \frac{e_\theta(nT) - e_\theta(nT-T)}{T} \tag{6.26}$$

$$e_{\theta i}(nT) = \frac{e_\theta(nT) + e_\theta(nT-T)}{T} \tag{6.27}$$

โดย $u_{\theta p}(nT)$ เป็นคำสั่งควบคุมความเร็วเชิงมุม

$u_{\theta p}(nT-T)$ เป็นคำสั่งควบคุมความเร็วเชิงมุมของการคำนวณครั้งก่อน

$K_{\theta p}$, $K_{\theta i}$ และ $K_{\theta d}$ เป็นค่าอัตราขยายของระบบ

$e_\theta(nT)$ เป็นค่าองศาที่ผิดพลาด

$e_\theta(nT-T)$ เป็นค่าองศาที่ผิดพลาดของการคำนวณครั้งก่อน

$e_{\theta i}(nT)$ เป็นผลรวมของค่าองศาที่ผิดพลาด

$e_{\theta d}(nT)$ เป็นอัตราการเปลี่ยนแปลงค่าองศาที่ผิดพลาด

θ_s เป็นค่าองศาที่ได้ตั้งเป้าหมายไว้

$\theta_m(nT)$ เป็นค่าองศาของหุ่นยนต์เล่นฟุตบอลในปัจจุบัน

6.2.2 การควบคุมการหันหน้าแบบฟัซซี (Fuzzy)

สามารถนำการควบคุมแบบฟัซซีมาใช้ในการควบคุมการหันหน้าได้โดยกำหนดกฎการควบคุมตามแบบจำลองของ Takagi-Sugeno [15] จะทำให้ได้ระบบที่สร้างจากกฎการควบคุมการหันหน้าแบบฟัซซี่เสร็จสมบูรณ์

If $e_\theta(nT) > 0$ and $e_\theta(nT) \le L1$ then $u_{\theta p}(nT) = \dfrac{e_\theta(nT)\cdot v_\theta}{L1}$ (6.28)

If $e_\theta(nT) < 0$ and $e_\theta(nT) \ge R1$ then $u_{\theta p}(nT) = \dfrac{-e_\theta(nT)\cdot v_\theta}{R1}$ (6.29)

If $e_\theta(nT) > L1$ then $u_{\theta p}(nT) = v_\theta$ (6.30)

If $e_\theta(nT) < R1$ then $u_{\theta p}(nT) = -v_\theta$ (6.31)

โดย $L1$ เป็นค่าองศาที่กำหนดไว้ด้านบวก

$R1$ เป็นค่าองศาที่กำหนดไว้ด้านลบ

v_θ เป็นค่าความเร็วในการหัน

6.2.3 การควบคุมความเร็วการวิ่งแบบพีไอดี

ตัวควบคุมแบบพีไอดีสามารถนำมาประยุกต์ใช้กับการควบคุมความเร็วสำหรับวิ่งเข้าจุดเป้าหมายคือหุ่นยนต์เล่นฟุตบอลได้ โดยการตั้งค่าเป้าหมายเป็นระยะทางเท่ากับ 0 และให้ตัวควบคุมนั้นส่งคำสั่งความเร็วของล้อซึ่งแปลงมาจากความเร็วในแนวระนาบของหุ่นยนต์ไปบังคับให้หุ่นยนต์เล่นฟุตบอลวิ่งไปด้วยความเร็วที่ต้องการ

สามารถเขียนระบบควบคุมสำหรับควบคุมความเร็วแบบพีไอดีได้โดยใช้สมการ (6.32) โดยเมื่อคำนวณคำสั่งควบคุมความเร็วแล้วจะต้องจำกัดให้ไม่ให้เกินค่าสูงสุดที่ป้อนคำสั่งได้ด้วย

$$u_{sp}(nT) = u_{sp}(nT - T) + \left(K_{sp}e_s(nT) + K_{si}e_{si}(nT) + K_{sd}e_{sd}(nT)\right) \tag{6.32}$$

และ

$$e_s(nT) = s_s - s_{rt}(nT) \tag{6.33}$$

$$e_{sd}(nT) = \frac{e_s(nT) - e_s(nT - T)}{T} \tag{6.34}$$

$$e_{si}(nT) = \frac{e_s(nT) + e_s(nT - T)}{T} \tag{6.35}$$

โดย

$u_{sp}(nT)$	เป็นคำสั่งควบคุมความเร็วแนวระนาบ
$u_{sp}(nT - T)$	เป็นคำสั่งควบคุมความเร็วแนวระนาบของการคำนวณครั้งก่อน
K_{sp}, K_{si} และ K_{sd}	เป็นค่าอัตราขยายของระบบ
$e_s(nT)$	เป็นค่าระยะทางที่ผิดพลาด
$e_s(nT - T)$	เป็นค่าระยะทางที่ผิดพลาดของการคำนวณครั้งก่อน
$e_{si}(nT)$	เป็นผลรวมของค่าระยะทางที่ผิดพลาด
$e_{sd}(nT)$	เป็นอัตราการเปลี่ยนแปลงระยะทางที่ผิดพลาด
s_s	เป็นค่าระยะทางที่ได้ตั้งเป้าหมายไว้หรือเท่ากับ 0
$s_{rt}(nT)$	เป็นค่าระยะทางของหุ่นยนต์เล่นฟุตบอลในปัจจุบัน

6.2.4 การควบคุมความเร็วการวิ่งแบบฟัซซี

สามารถประยุกต์นำการควบคุมแบบฟัซซีมาใช้ในการควบคุมความเร็วการวิ่งได้ โดยในที่นี้จะนำแบบจำลองของ Takagi-Sugeno [15] มาสร้างกฎการควบคุมโดยให้ความเร็วของการวิ่งลดลงตามระยะทางระหว่างหุ่นยนต์เล่นฟุตบอลกับจุดเป้าหมาย

$$\text{If} \quad e_s(nT) > s_d \text{ then } u_{sp}(nT) = \frac{e_s(nT) \cdot v_i}{s_d} \tag{6.36}$$

$$\text{If} \quad e_s(nT) \leq s_d \text{ then } u_{sp}(nT) = v_i \tag{6.37}$$

โดย s_d เป็นระยะทางที่เริ่มลดความเร็ว

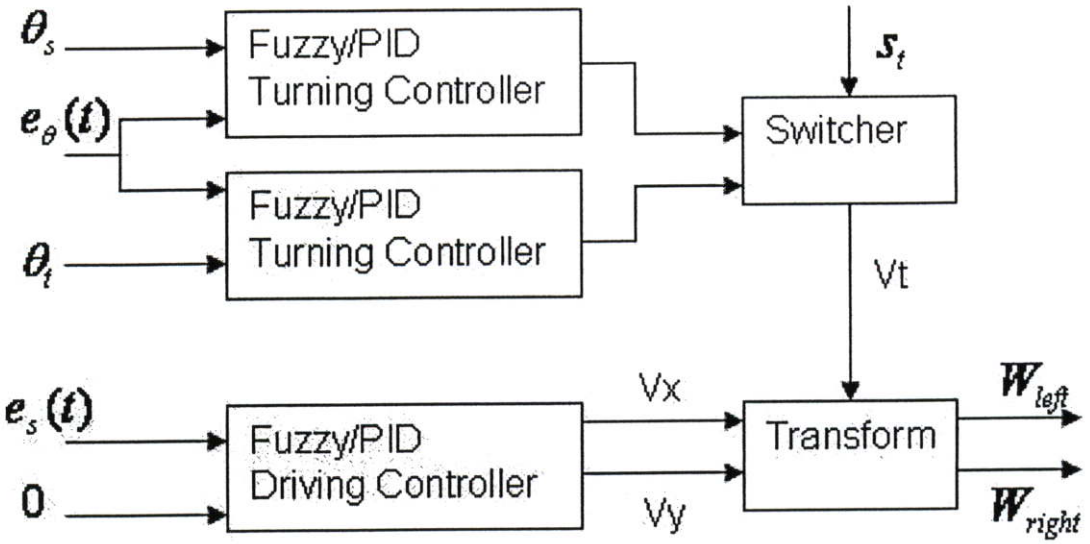
v_i เป็นความเร็วที่ใช้ในการวิ่ง

6.3 ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอล

จากหัวข้อที่ 6.2 สามารถนำการควบคุมทั้งหมดมาประยุกต์ได้เป็น 4 แบบคือ FF ซึ่งใช้ฟัซซีทั้งการหันหน้าและควบคุมความเร็ว PP ซึ่งใช้พีไอดีทั้งหันหน้าและควบคุมความเร็ว FP ซึ่ง ใช้ฟัซซีในการหันหน้าและใช้พีไอดีในการควบคุมความเร็วและ PF ซึ่งใช้พีไอดีในการหันหน้าและใช้ฟัซซีในการควบคุมความเร็ว

แต่หากพิจารณาแบบจำลองระบบขับเคลื่อนของหุ่นยนต์เล่นฟุตบอลแล้ว พบว่าวิธีการนำร่องสำหรับหุ่นยนต์สำหรับเคลื่อนที่ทุกทิศทางกับแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อแตกต่างกันอย่างสิ้นเชิง เพราะว่าแบบเคลื่อนที่ทุกทิศทางนั้น สามารถแยกการควบคุมองศาด้านหน้าหุ่นยนต์กับการควบคุมทิศทางการวิ่งได้ แต่แบบเก้าอี้รถเข็นทำไม่ได้ การนำร่องสำหรับหุ่นยนต์ที่มีการขับเคลื่อนแบบเก้าอี้รถเข็นนั้นจะต้องบังคับให้หุ่นยนต์วิ่งไปยังจุดนำร่องก่อน แล้วถึงจะสามารถหันด้านหน้าหุ่นยนต์ไปในทิศทางที่ต้องการได้ ดังนั้นจะต้องใช้วิธีการนำร่องมาประยุกต์ให้เหมาะสมกับระบบขับเคลื่อนด้วย

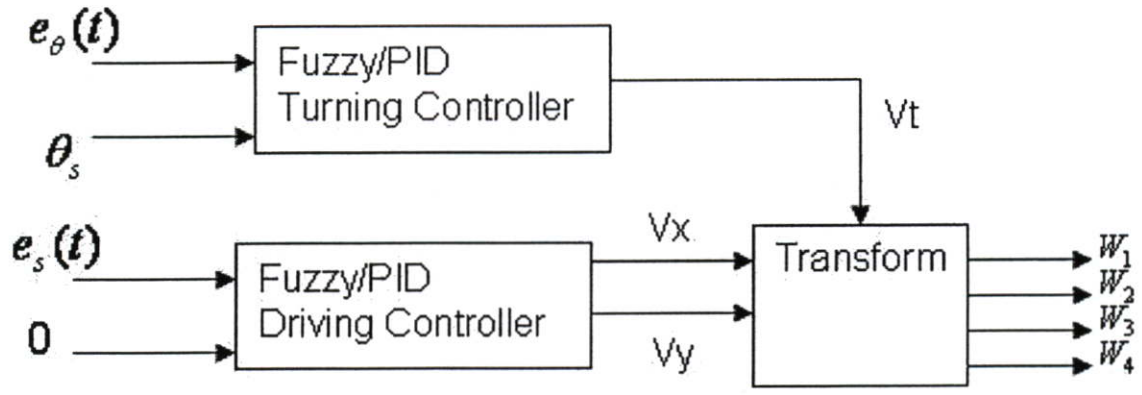
6.3.1 ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลแบบเก้าอี้รถเข็น 2 ล้อ



รูปที่ 6.5 ระบบนำร่องสำหรับหุ่นยนต์เก้าอี้รถเข็น 2 ล้อ

จากรูป 6.5 จะเห็นระบบนำร่องโดยรวม ซึ่งมีตัวควบคุมการหันสองชนิด คือ ตัวควบคุมการหันที่เป้าหมายเป็นมุมที่หุ่นยนต์หันหน้าไปสู่จุดนำร่อง กับตัวควบคุมการหันที่มีเป้าหมายเป็นมุมที่กำหนดไว้ โดยเมื่อเริ่มต้นการวิ่ง ตัวสลับ (Switcher) จะให้เป้าหมายการหันเป็นมุมที่หันหน้าไปยังจุดนำร่องหรือ θ_t เมื่อวิ่งเข้าสู่ระยะการสลับเป้าหมายที่กำหนดไว้คือ s_t ก็จะสลับเป้าหมายการหันเป็นมุมที่กำหนดไว้ หรือ θ_s จากนั้นก็นำค่าคำสั่งมาคำนวณกับคำสั่งที่ได้จากตัวควบคุมความเร็วโดยใช้สมการ (6.15)-(6.16) ซึ่งสุดท้ายก็จะได้คำสั่งสำหรับล้อทั้งสอง

6.3.2 ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลแบบเคลื่อนที่ได้ทุกทิศทาง



รูปที่ 6.6 ระบบนำร่องสำหรับหุ่นยนต์เคลื่อนที่ได้ทุกทิศทาง

จากรูป 6.6 จะเห็นว่าระบบนำร่องมีความเรียบง่ายกว่าแบบเก้าอี้รถเข็น โดยตัวควบคุมการหันหน้าจากคำนวณเหมือนกับระบบนำร่องแบบเก้าอี้รถเข็น แต่ไม่มีการสลับการทำงาน

ตัวควบคุมความเร็วนั้นจะต้องคำนวณทิศทางการวิ่งด้วย โดยกำหนดทิศทางการวิ่งทั้งหมด 8 ทิศทางตามตาราง 6.1 คำนวณจะได้คำสั่งการเคลื่อนที่ในแนวระนาบ จากนั้นก็นำไปคำนวณโดยใช้สมการ (6.17)-(6.19) ในกรณีที่เป็นหุ่นยนต์เล่นฟุตบอลที่มีการขับเคลื่อนแบบ 3 ล้อ หรือใช้สมการ (6.20)-(6.23) ในกรณีที่เป็นหุ่นยนต์เล่นฟุตบอลที่มีการขับเคลื่อนแบบ 4 ล้อ ซึ่งก็จะได้คำสั่งที่ส่งออกไปยังล้อแต่ละล้อในที่สุด

ตารางที่ 6.1 ทิศทางการวิ่งของระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง

การวิ่ง	ช่วงการควบคุม	ทิศทางการวิ่ง
ข้างหน้า	[22.5,-22.5]	0
เฉียงซ้ายหน้า	[22.5,67.5]	45
ซ้าย	[67.5,112.5]	90
เฉียงซ้ายหลัง	[112.5,157.5]	135
ข้างหลัง	[157.5,-157.5]	180
เฉียงขวาหน้า	[-22.5,-67.5]	-45
ขวา	[-67.5,-112.5]	-90
เฉียงขวาหลัง	[-112.5,-157.5]	-135

ในบทที่ 7 ซึ่งเป็นบทต่อไปจะแสดงผลการทดลองของวิทยานิพนธ์ฉบับนี้ โดยแสดงผลการทดลองการนำร่องสำหรับหุ่นยนต์แบบเก้าอี้รถเข็นสองล้อ ผลการทดลองการนำร่องสำหรับหุ่นยนต์เคลื่อนที่ทุกทิศทางสามล้อ ผลการทดลองการนำร่องสำหรับหุ่นยนต์เคลื่อนที่ทุกทิศทางสี่ล้อ ผลการวัดประสิทธิภาพของระบบนำ รูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอลในการจำลองสถานการณ์ และผลการแข่งขันในการทดสอบระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอลบนโปรแกรมจำลองสถานการณ์

บทที่ 7

การทดลอง

ในบทนี้จะแสดงผลการทดลองการนำร่องของหุ่นยนต์เตะฟุตบอลแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ แบบเคลื่อนที่ทุกทิศทาง 3 ล้อ แบบเคลื่อนที่ทุกทิศทาง4 ล้อ ตัวอย่างการเล่นของหุ่นยนต์เล่นฟุตบอลในการจำลองสถานการณ์ และผลการแข่งขันในการทดสอบระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอลบนโปรแกรมจำลองสถานการณ์

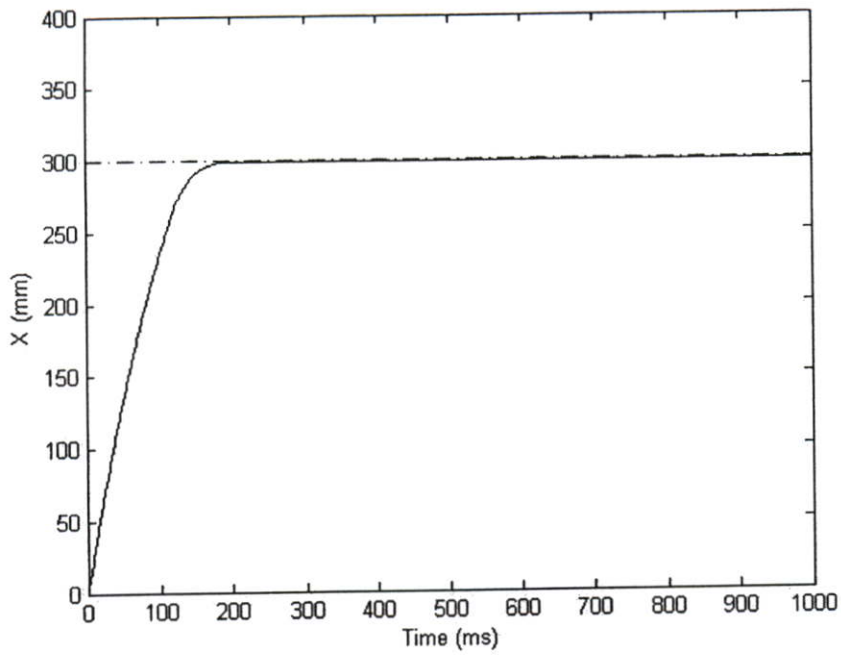
7.1 การทดลองการนำร่องสำหรับหุ่นยนต์แบบเก้าอี้รถเข็นสองล้อ

การทดลองนี้จะใช้การนำร่องสำหรับหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อทั้งสี่แบบ คือ FDFT (Fuzzy Driving and Fuzzy Turning) FDPT (Fuzzy Driving and PID Turning) PDFT (PID Driving and Fuzzy Turning) และ PDPT (PID Driving and PID Turning) ซึ่งกำหนดให้จุดเริ่มต้นของหุ่นยนต์อยู่ที่ตำแหน่ง (0,0) หันหน้าไปที่ 0 องศา และให้จุดเป้าหมายอยู่ที่ตำแหน่ง (300,300) หันหน้าไปที่ 30 องศา

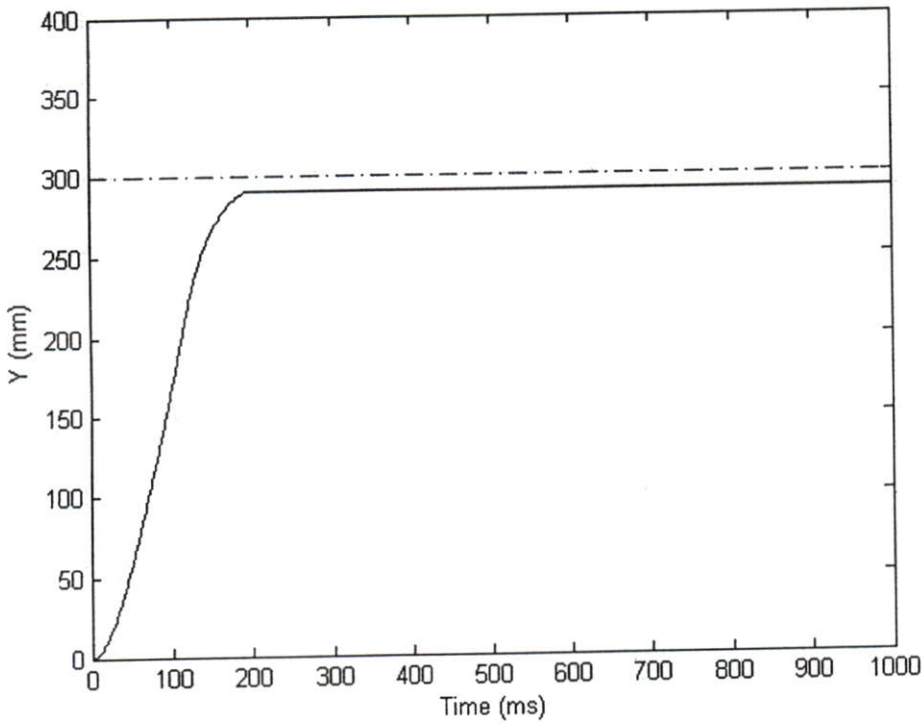
ส่วนตัวแปรอื่นๆ กำหนดให้ s_t เท่ากับ 20 มิลลิเมตร, s_d เท่ากับ 100 มิลลิเมตร, เวลาสุ่ม (Sampling Time) เท่ากับ 0.01 วินาที $K_{\theta p}$ เท่ากับ 1 $K_{\theta i}$ เท่ากับ 0.1 $K_{\theta d}$ เท่ากับ 0.5 K_{sp} เท่ากับ 1 K_{si} เท่ากับ 0.1 K_{sd} เท่ากับ 0.5 และความเร็วสูงสุดไม่เกิน 300 มิลลิเมตร/วินาที่

7.1.1 การนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ

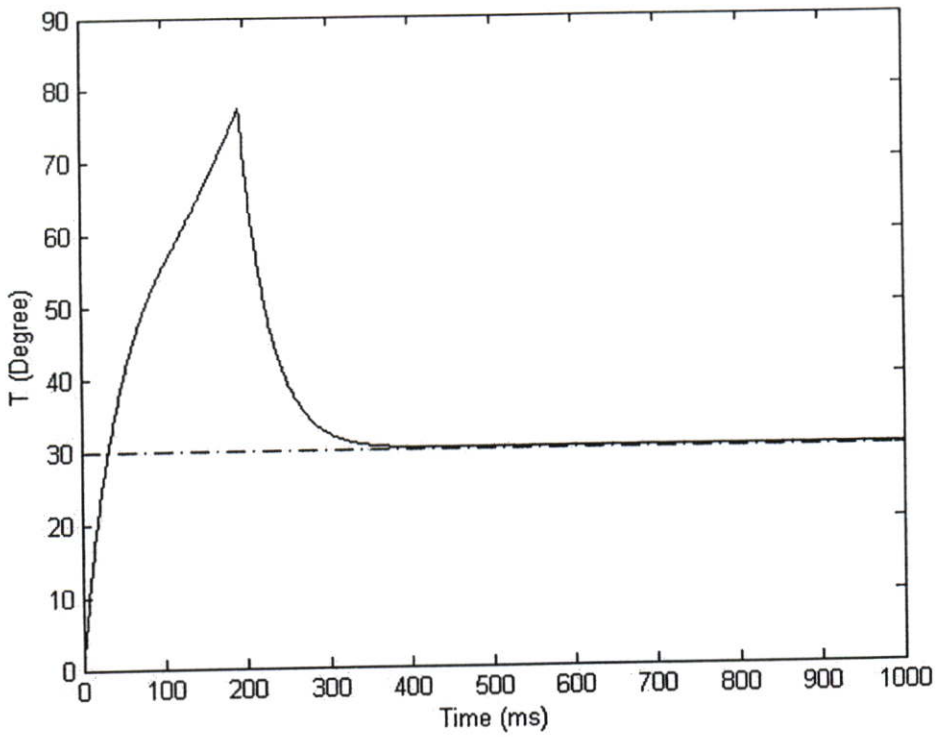
การทดลองนี้แสดงผลการนำร่องแบบ FDFT ซึ่งใช้การควบคุมความเร็วการวิ่งและการหันหน้าแบบฟัซซี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.1 ระยะในแกน X ของการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.2 ระยะในแกน Y ของการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

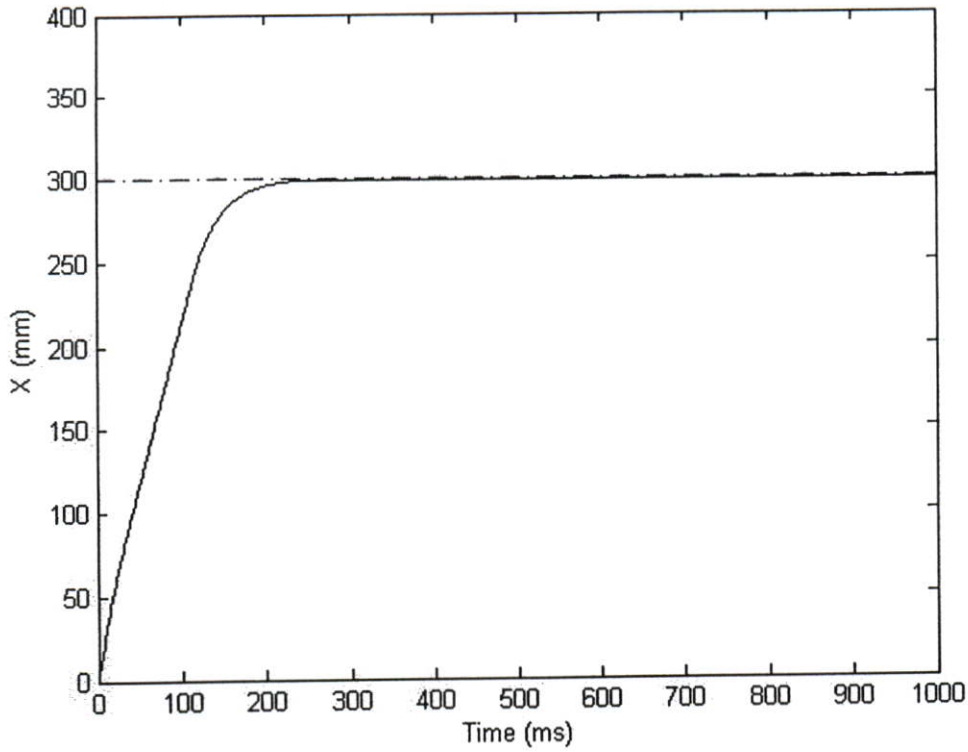


รูปที่ 7.3 องศาด้านหน้าของการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

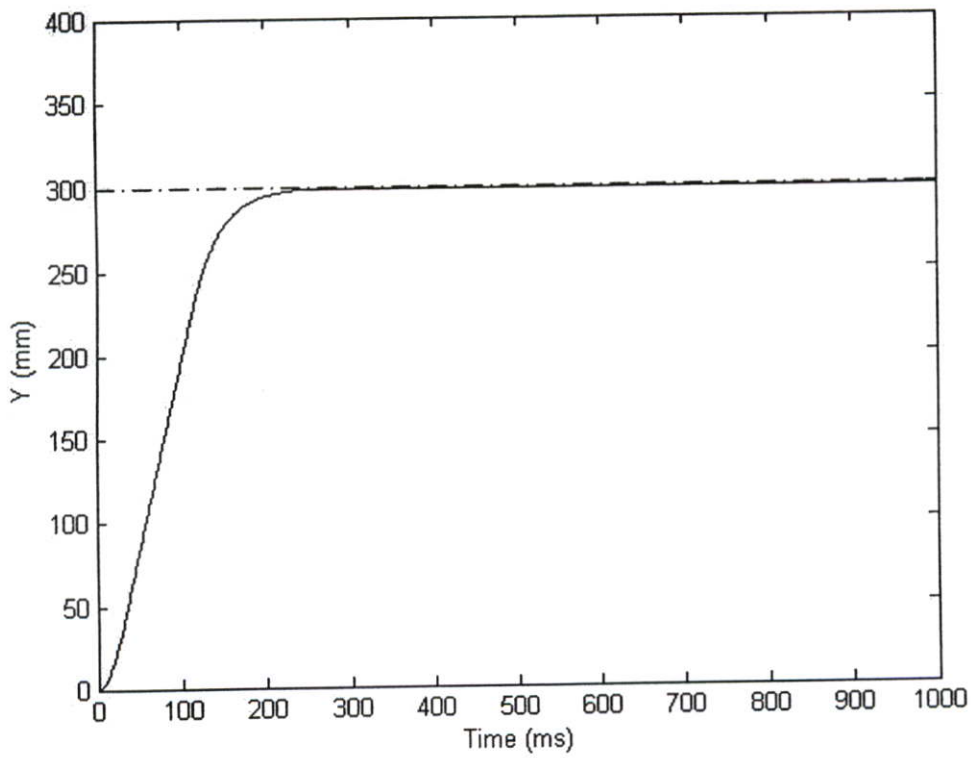
จากรูปที่ 7.1 และ 7.2 จะเห็นได้ว่าหุ่นยนต์วิ่งเข้าจุดเป้าหมายที่อยู่ตำแหน่ง (300,300) ได้อย่างถูกต้อง และรูปที่ 7.3 จะเห็นได้ว่าหุ่นยนต์หันหน้ามุ่งไปที่ 80 องศาก่อน เมื่อถึงตำแหน่ง (300,300) จะหันหน้ากลับมายังที่ 30 องศาตามที่ได้กำหนดไว้

7.1.2 การนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ

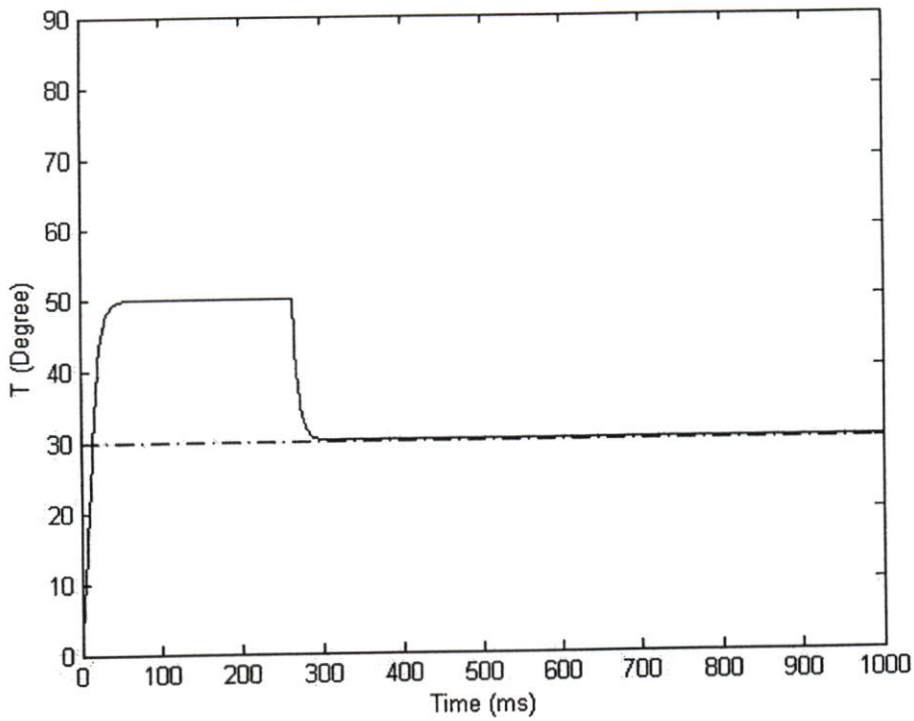
การทดลองนี้แสดงผลการนำร่องแบบ FDPT ซึ่งใช้การควบคุมความเร็วการวิ่งแบบฟัซซี่ และควบคุมการหันหน้าแบบพีไอดี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.4 ระยะในแกน X ของการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.5 ระยะในแกน Y ของการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

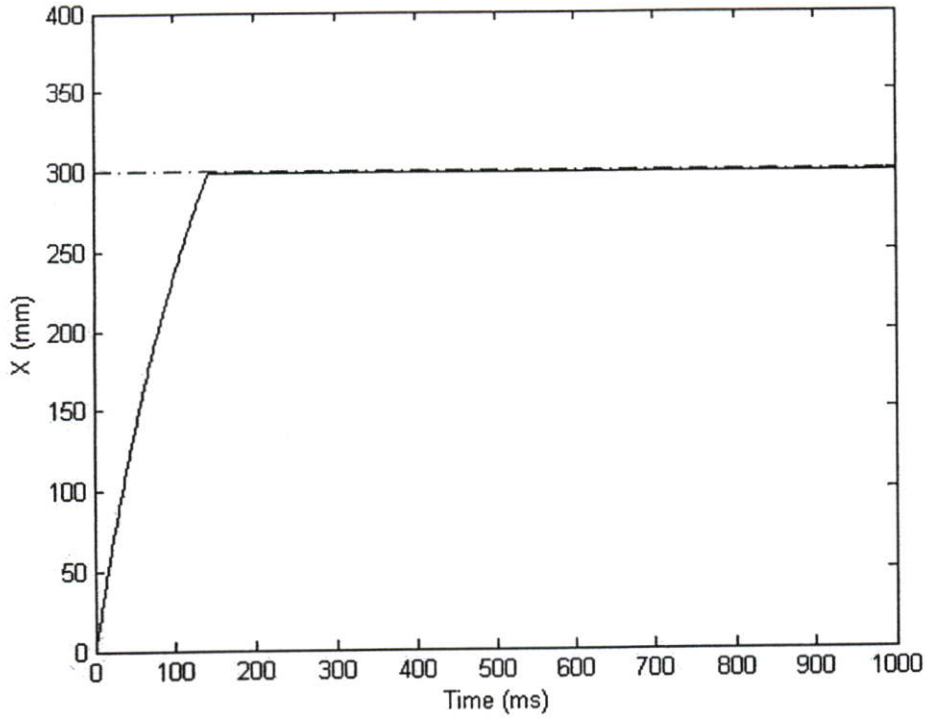


รูปที่ 7.6 องศาด้านหน้าของการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

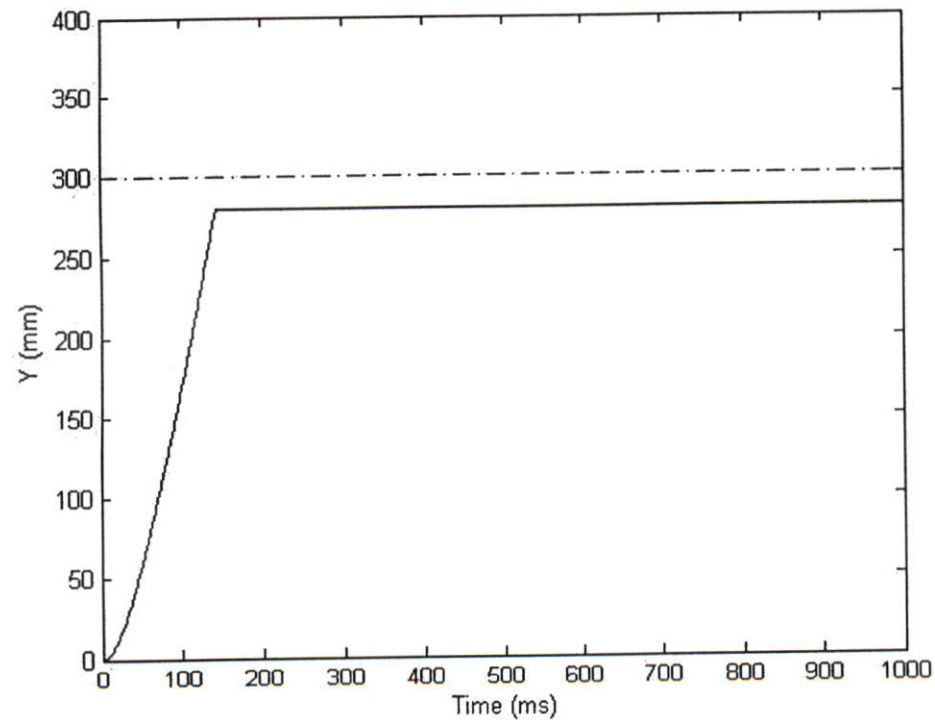
จากรูปที่ 7.4 และ 7.5 จะเห็นได้ว่าหุ่นยนต์วิ่งเข้าจุดเป้าหมายที่อยู่ตำแหน่ง (300,300) ได้อย่างถูกต้อง และรูปที่ 7.6 จะเห็นได้ว่าหุ่นยนต์หันหน้ามุ่งไปที่ 80 องศาก่อน เมื่อถึงตำแหน่ง (300,300) จะหันหน้ากลับมายังที่ 30 องศาตามที่ได้กำหนดไว้ ซึ่งเมื่อเทียบกับรูปที่ 7.3 จะเห็นได้ว่าหุ่นยนต์หันหน้าได้รวดเร็วกว่า

7.1.3 การนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ

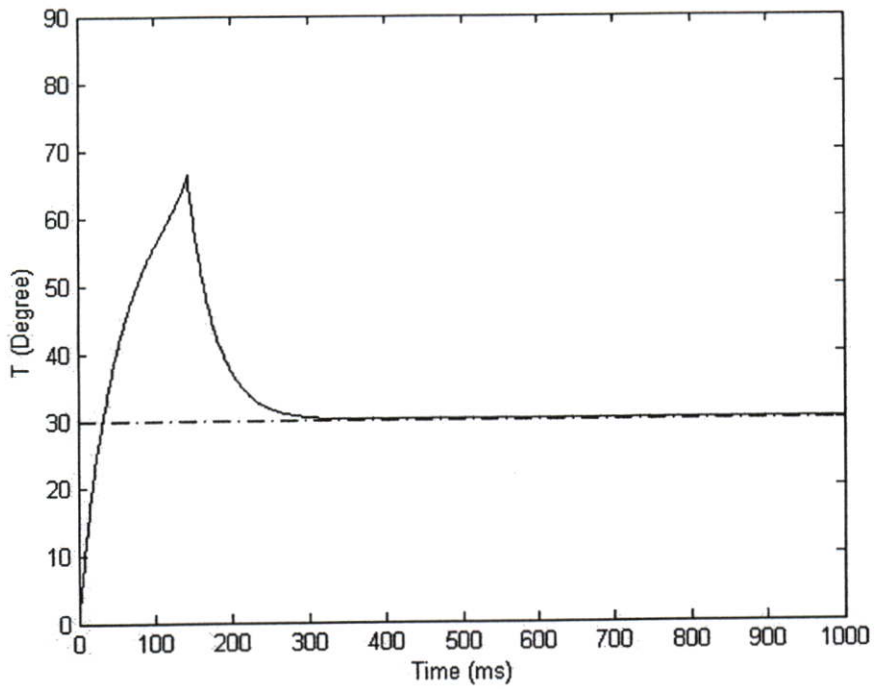
การทดลองนี้แสดงผลการนำร่องแบบ PDFT ซึ่งใช้การควบคุมความเร็วการวิ่งแบบพีไอดีและควบคุมการหันหน้าแบบฟัซซี่ โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.7 ระยะในแกน X ของการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.8 ระยะในแกน Y ของการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

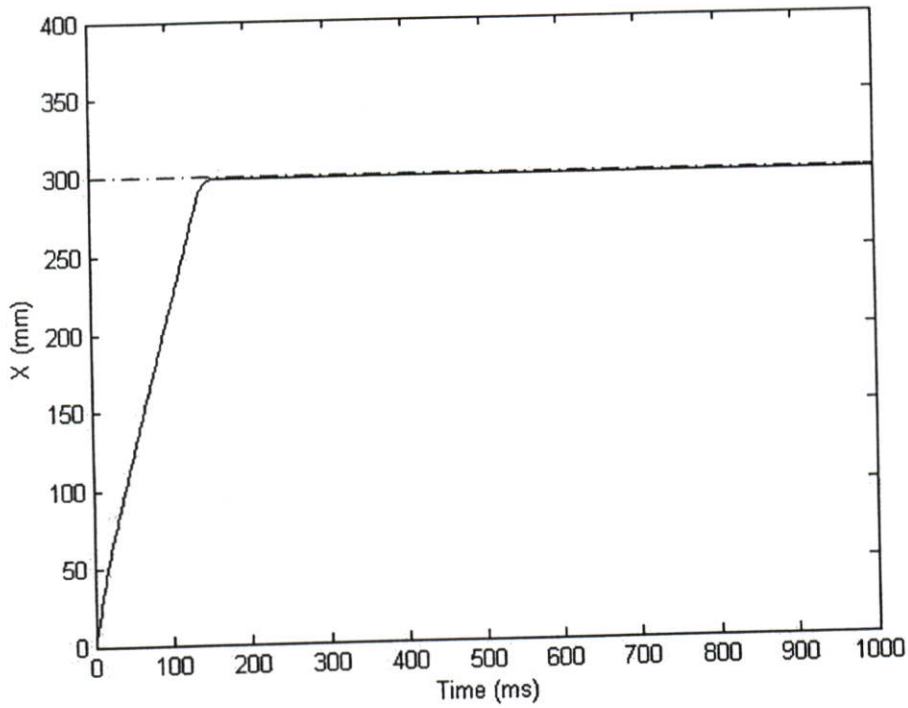


รูปที่ 7.9 องศาด้านหน้าของการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

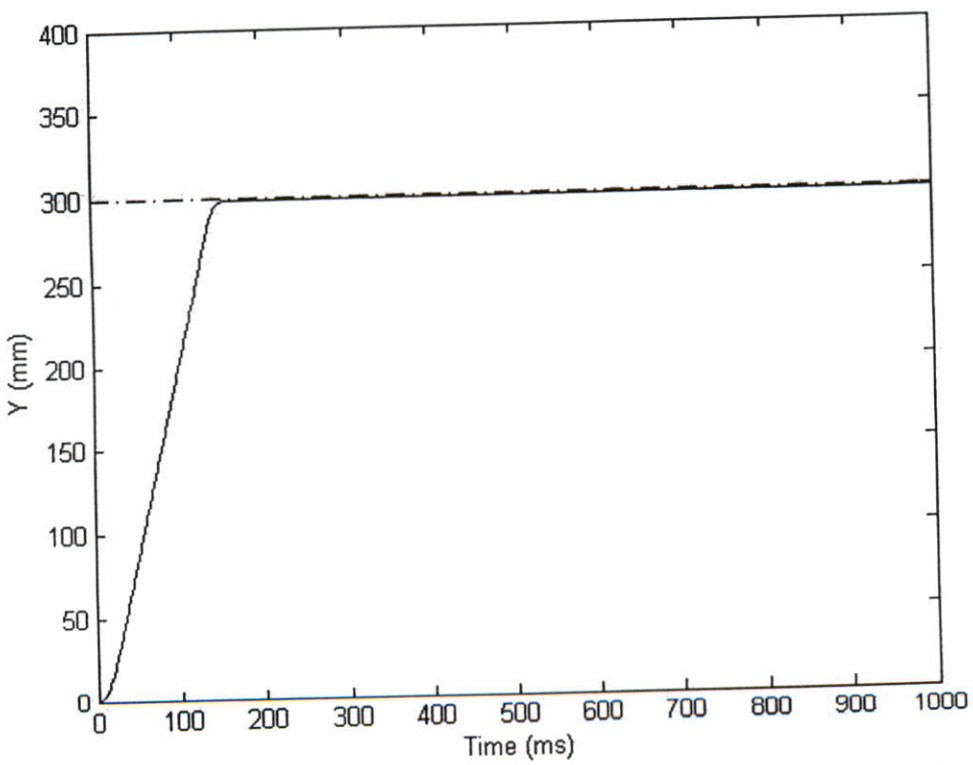
จากรูปที่ 7.7 และ 7.8 จะเห็นได้ว่าหุ่นยนต์ได้วิ่งเข้าจุดเป้าหมายที่อยู่ตำแหน่ง (300,300) ได้อย่างถูกต้อง แต่มีช่องว่างเล็กน้อยเนื่องจากได้กำหนด *s*, เท่ากับ 20 มิลลิเมตรซึ่งเป็นระยะที่สลับการนำร่องจากการวิ่งไปยังจุดเป้าหมายไปเป็นการหันหน้าไปยังองศาที่ได้กำหนดไว้ และรูปที่ 7.9 จะเห็นได้ว่าหุ่นยนต์หันหน้ามุ่งไปที่ 80 องศาก่อน เมื่อถึงตำแหน่ง (300,300) จะหันหน้ากลับมายังที่ 30 องศาตามที่ได้กำหนดไว้ ซึ่งคล้ายกับรูปที่ 7.3 เพราะใช้การควบคุมแบบเดียวกัน

7.1.4 การนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถเข็น 2 ล้อ

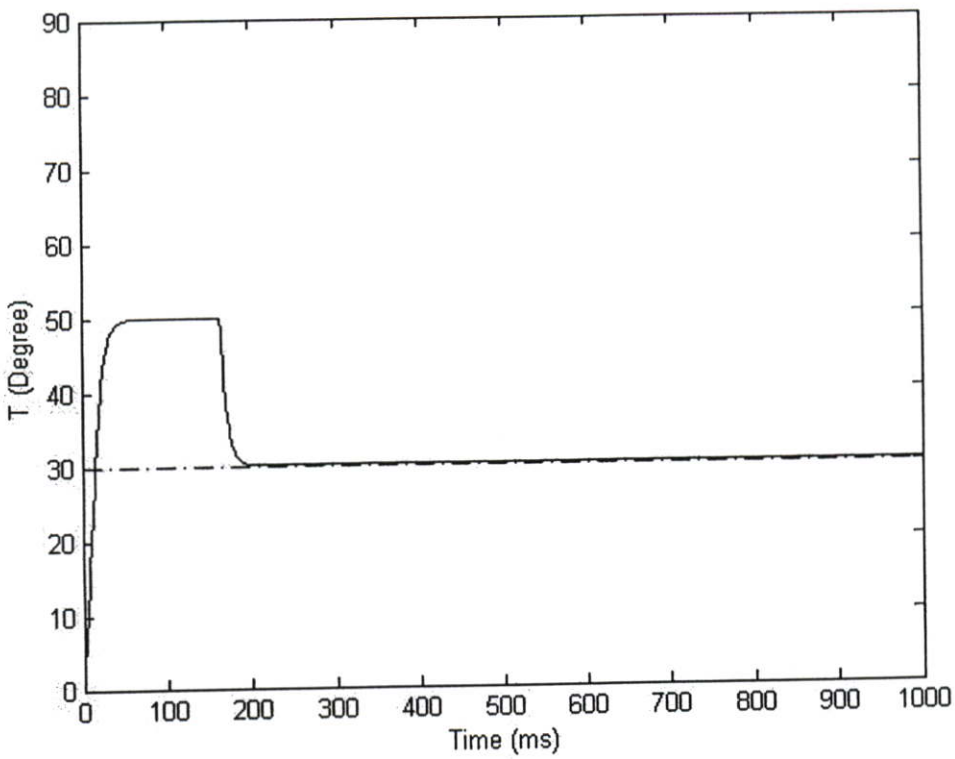
การทดลองนี้แสดงผลการนำร่องแบบ PDPT ซึ่งใช้การควบคุมความเร็วการวิ่งและควบคุมการหันหน้าแบบฟัซซี่ โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.10 ระยะในแกน X ของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.11 ระยะในแกน Y ของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ



รูปที่ 7.12 องศาด้านหน้าของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

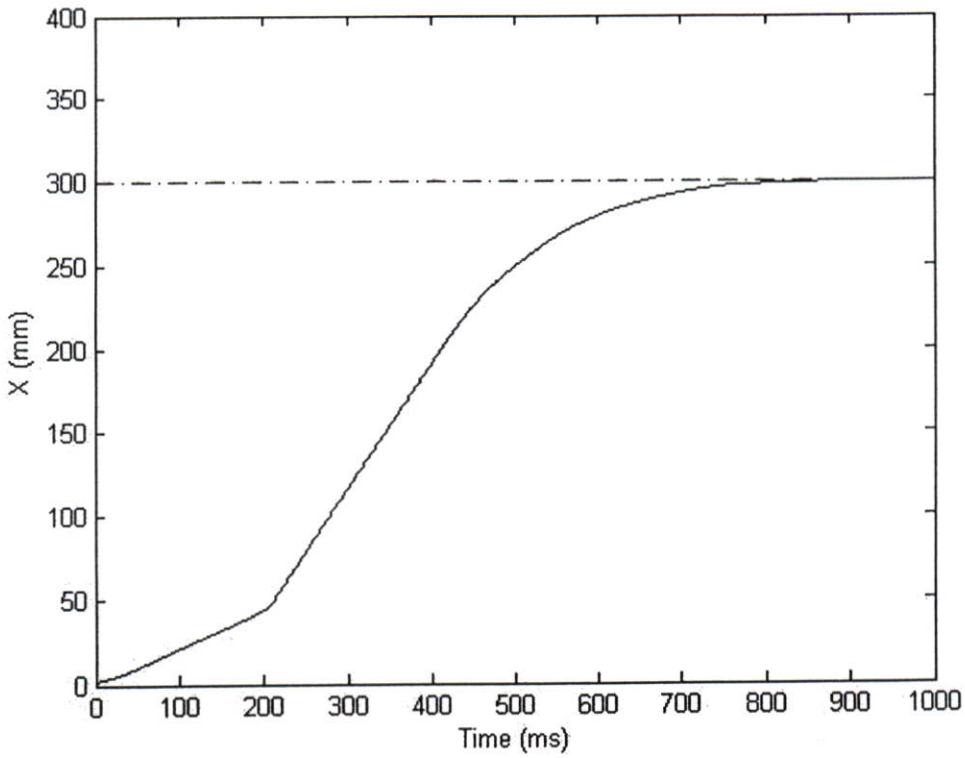
จากรูปที่ 7.10 และ 7.11 จะเห็นได้ว่าหุ่นยนต์ได้วิ่งเข้าจุดเป้าหมายอยู่ที่ตำแหน่ง (300,300) ได้อย่างถูกต้อง และรูปที่ 7.12 จะเห็นได้ว่าหุ่นยนต์หันหน้ามุ่งไปที่ 80 องศาก่อนและเมื่อถึงตำแหน่ง (300,300) จะหันหน้ากลับมายังที่ 30 องศาตามที่ได้กำหนดไว้ ซึ่งรูปแบบคล้ายกับรูปที่ 7.6 เพราะใช้การควบคุมแบบเดียวกัน

7.2 การทดลองการนำร่องสำหรับหุ่นยนต์เคลื่อนที่ทุกทิศทางสามล้อ

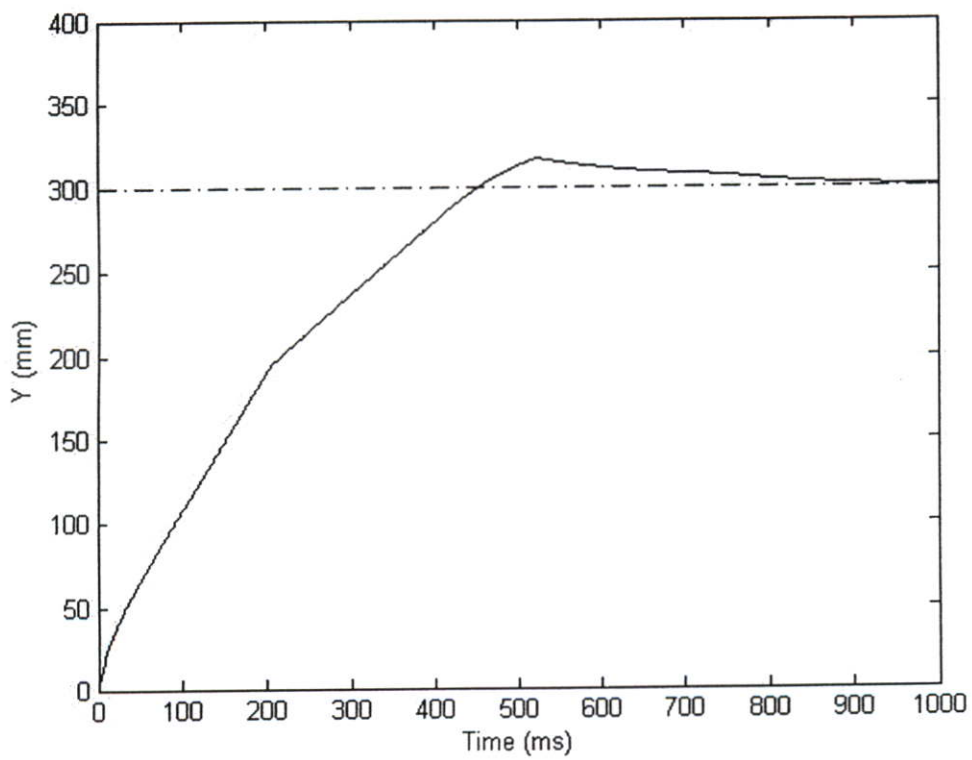
การทดลองนี้จะใช้การนำร่องสำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 3 ล้อ ซึ่งกำหนดค่าตัวแปรเหมือนกับกรณีการทดลองกับหุ่นยนต์แบบเก้าอี้รถเข็นสองล้อทุกประการ ยกเว้นค่า s_r ที่ไม่ได้ใช้เพราะระบบการนำร่องแตกต่างกัน

7.2.1 การนำร่องของ FDFT สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 3 ล้อ

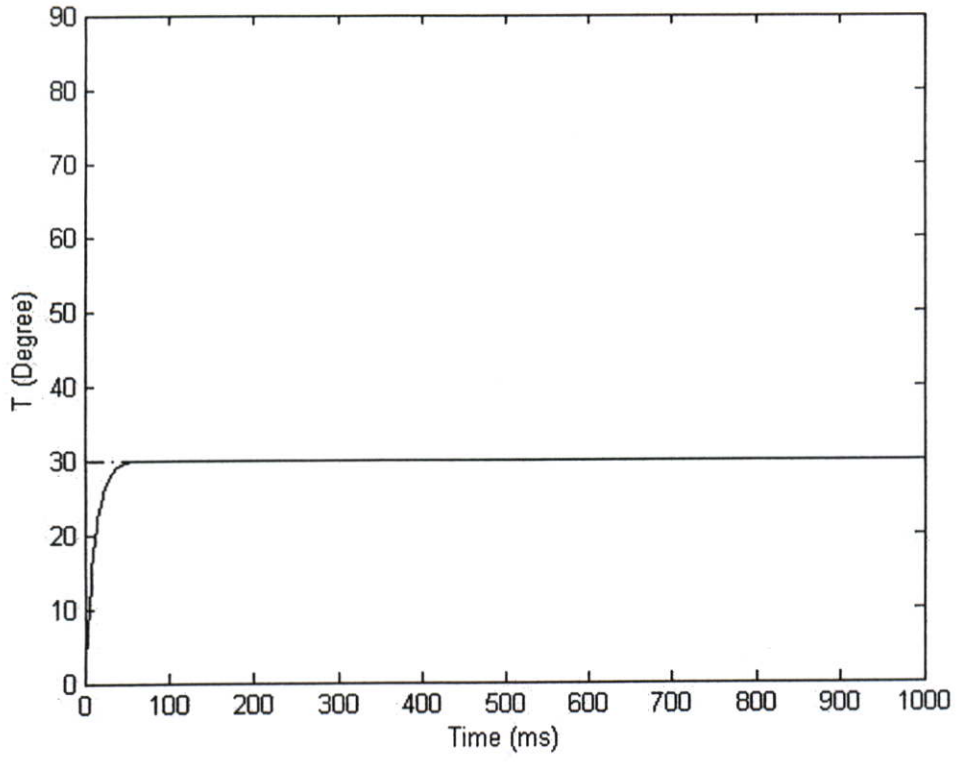
การทดลองนี้แสดงผลการนำร่องแบบ FDFT ซึ่งใช้การควบคุมความเร็วการวิ่งและควบคุมการหันหน้าแบบฟัซซี่ โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 3 ล้อ



รูปที่ 7.13 ระยะในแกน X ของการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



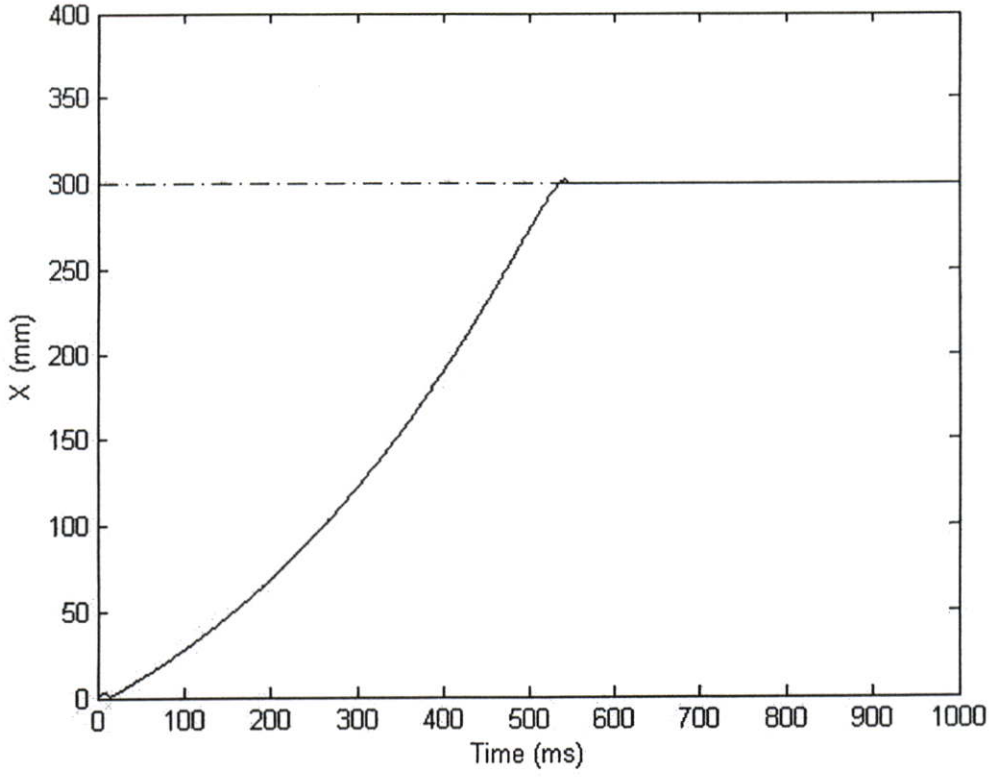
รูปที่ 7.14 ระยะในแกน Y ของการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



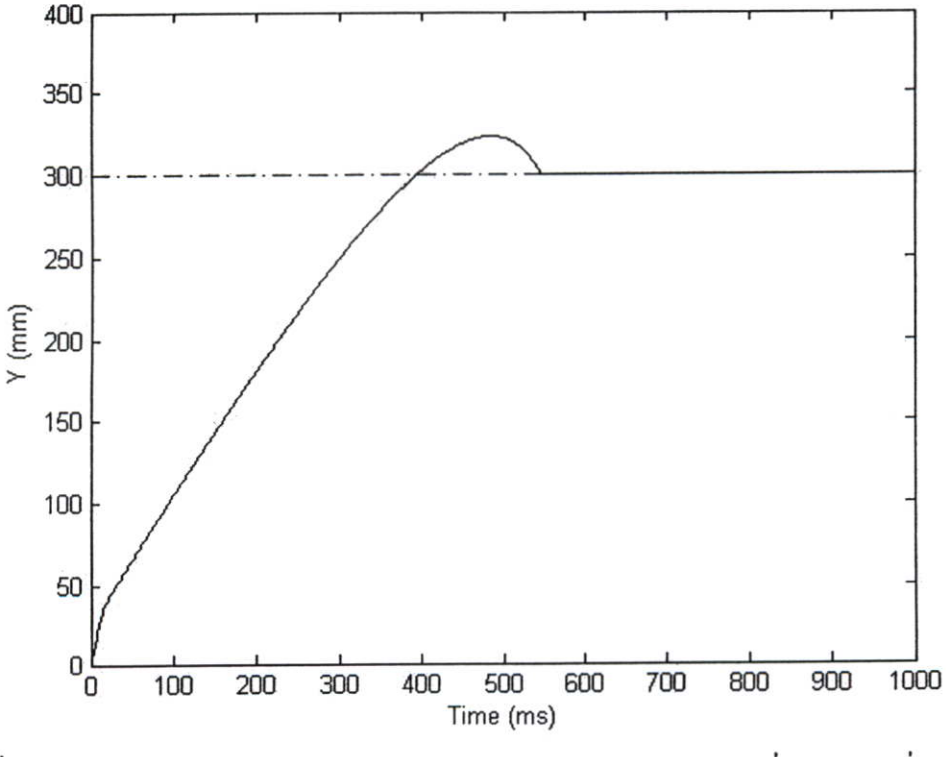
รูปที่ 7.15 องศาด้านหน้าการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ

7.2.2 การนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ

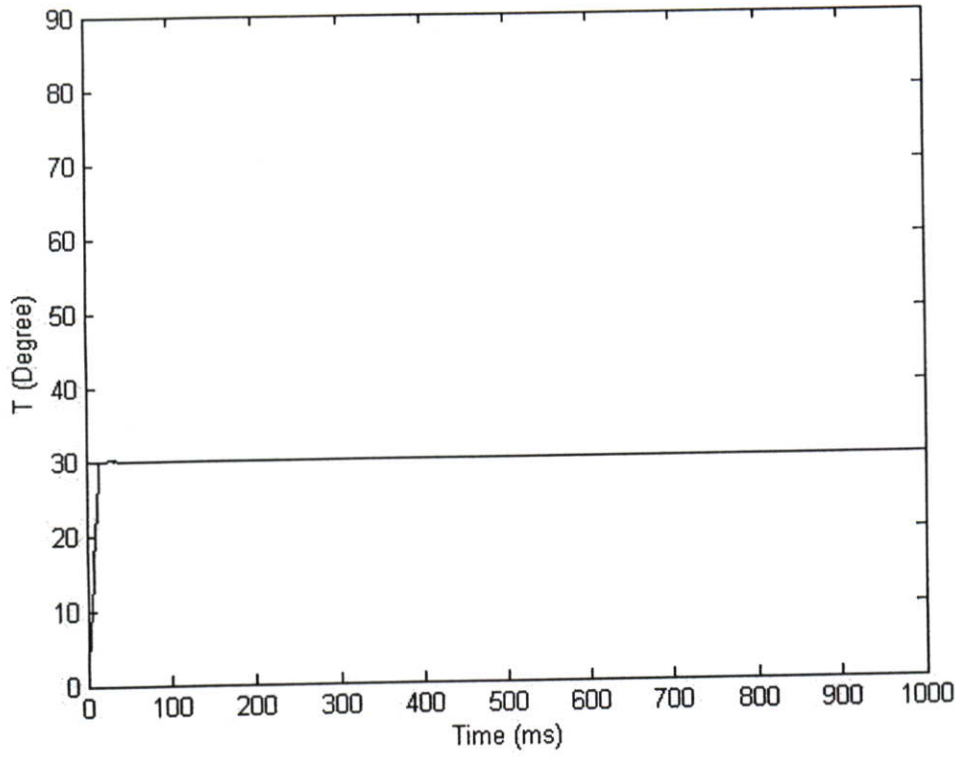
การทดลองนี้แสดงผลการนำร่องแบบ FDPT ซึ่งใช้การควบคุมความเร็วการวิ่งแบบฟัซซีและควบคุมการหันหน้าแบบพีไอดี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 3 ล้อ



รูปที่ 7.16 ระยะในแกน X ของการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



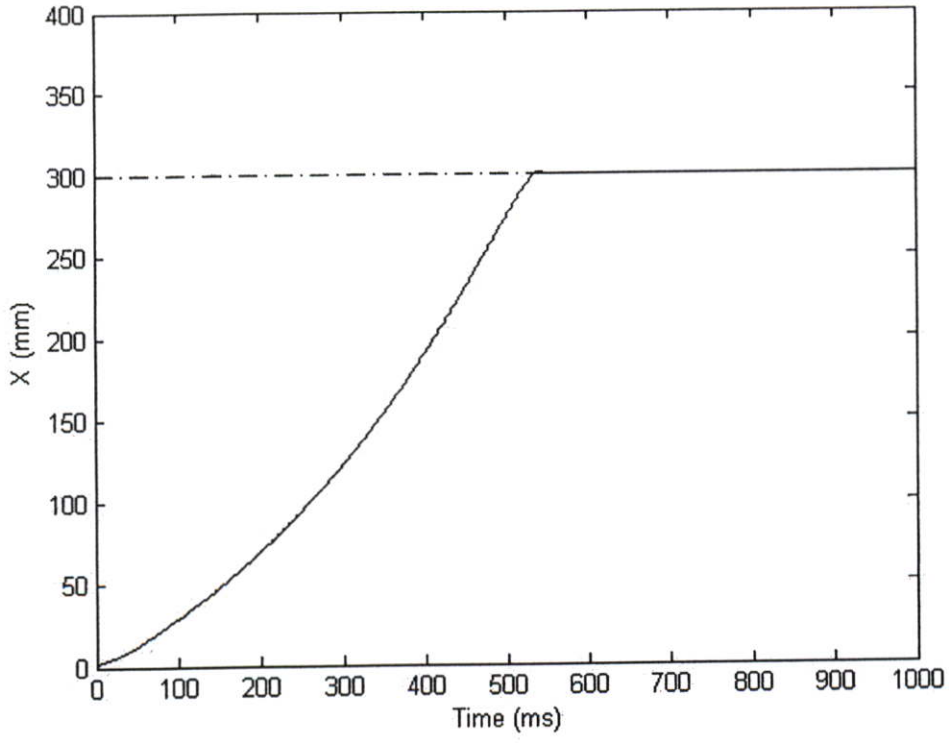
รูปที่ 7.17 ระยะในแกน Y ของการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



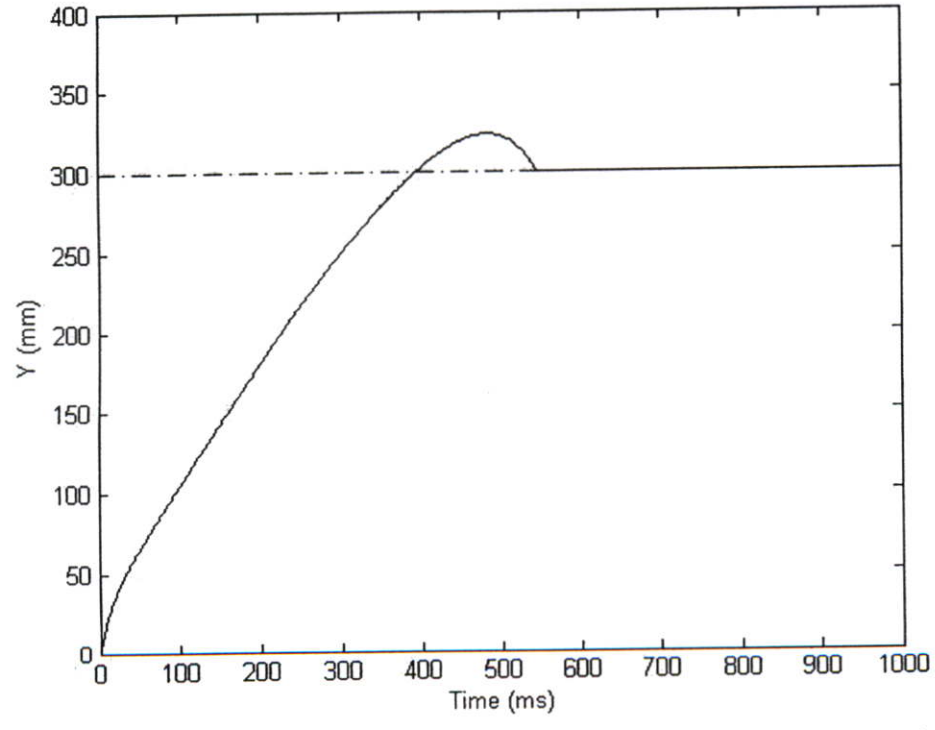
รูปที่ 7.18 องศาด้านหน้าการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ

7.2.3 การนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ

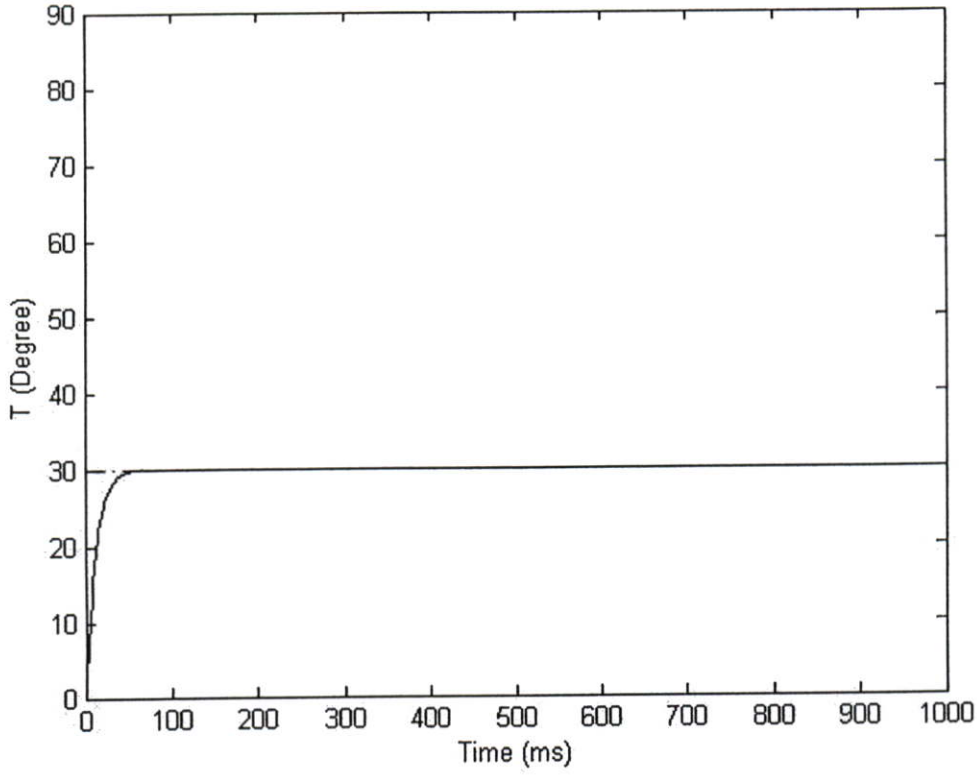
การทดลองนี้แสดงผลการนำร่องแบบ PDFT ซึ่งใช้การควบคุมความเร็วการวิ่งแบบพีไอดี และควบคุมการหันหน้าแบบฟัซซี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 3 ล้อ



รูปที่ 7.19 ระยะในแกน X ของการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



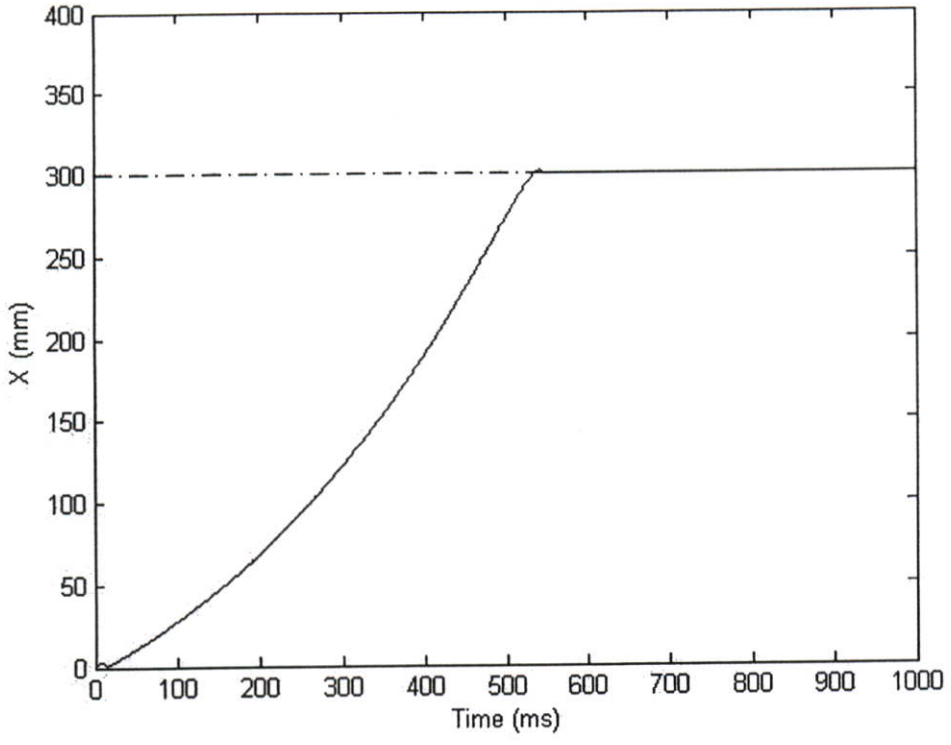
รูปที่ 7.20 ระยะในแกน Y ของการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



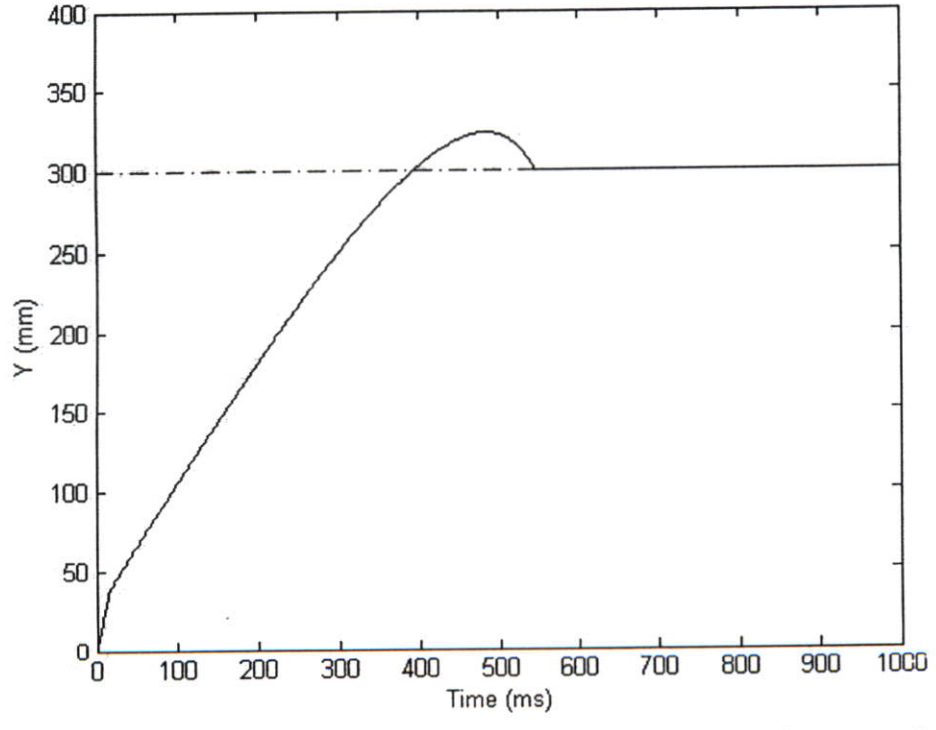
รูปที่ 7.21 องศาด้านหน้าการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ

7.2.4 การนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่ทุกทิศทาง 3 ล้อ

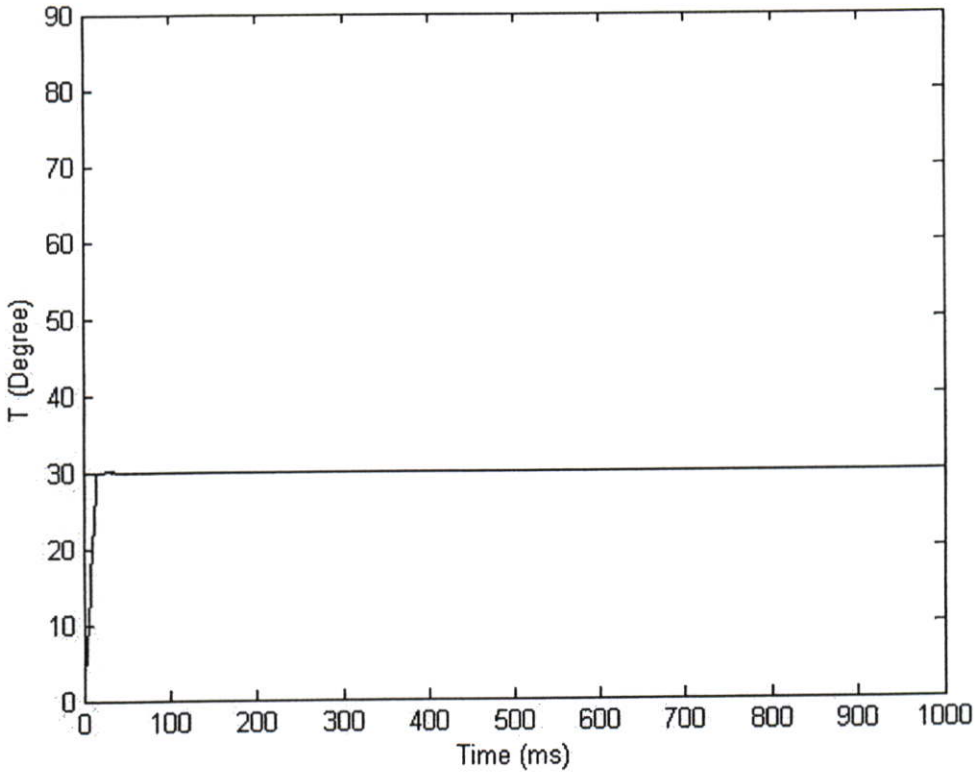
การทดลองนี้แสดงผลการนำร่องแบบ PDPT ซึ่งใช้การควบคุมความเร็วการวิ่งและควบคุมการหันหน้าแบบพีไอดี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 3 ล้อ



รูปที่ 7.22 ระยะในแกน X ของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



รูปที่ 7.23 ระยะในแกน Y ของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ



รูปที่ 7.24 องศาด้านหน้าการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 3 ล้อ

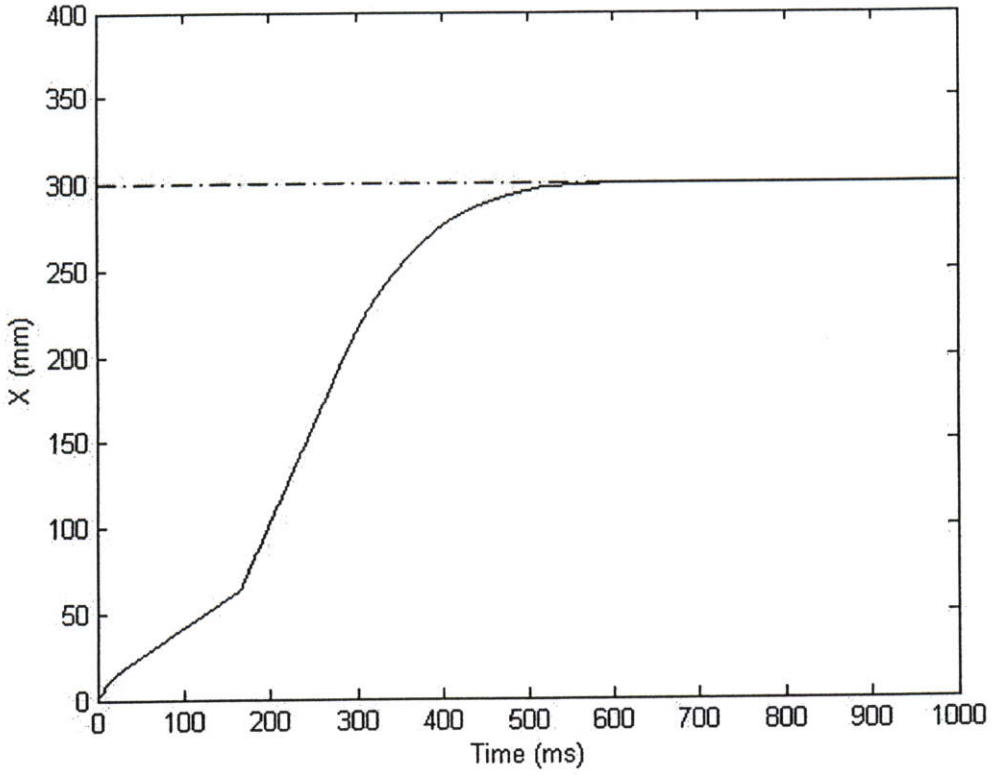
จากรูปที่ 7.13-7.14 รูปที่ 7.16-7.17 รูปที่ 7.19-7.20 และรูปที่ 7.22-7.23 แสดงให้เห็นว่าหุ่นยนต์ได้วิ่งเข้าจุดเป้าหมายอยู่ที่ตำแหน่ง (300,300) ได้อย่างถูกต้อง ซึ่งลักษณะการวิ่งจะแตกต่างกันไปตามกฎการนำร่องและเข้าจุดได้ช้ากว่าหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อเพราะหุ่นยนต์เคลื่อนที่ในแนวด้านข้างตามกฎการนำร่องที่ได้กำหนด ซึ่งเสียเปรียบหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อที่เคลื่อนที่ในแนวด้านหน้า แต่รูปที่ 7.15 รูปที่ 7.18 รูปที่ 7.21 และรูปที่ 7.24 แสดงให้เห็นว่าหุ่นยนต์แบบเคลื่อนที่ทุกทิศทางสามล้อสามารถหันหน้าได้ทันทีและรวดเร็วกว่าหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ เพราะไม่จำเป็นต้องรอให้ถึงตำแหน่งเป้าหมาย ซึ่งเป็นข้อได้เปรียบทางกายภาพ

7.3 การทดลองการนำร่องสำหรับหุ่นยนต์เคลื่อนที่ทุกทิศทางสี่ล้อ

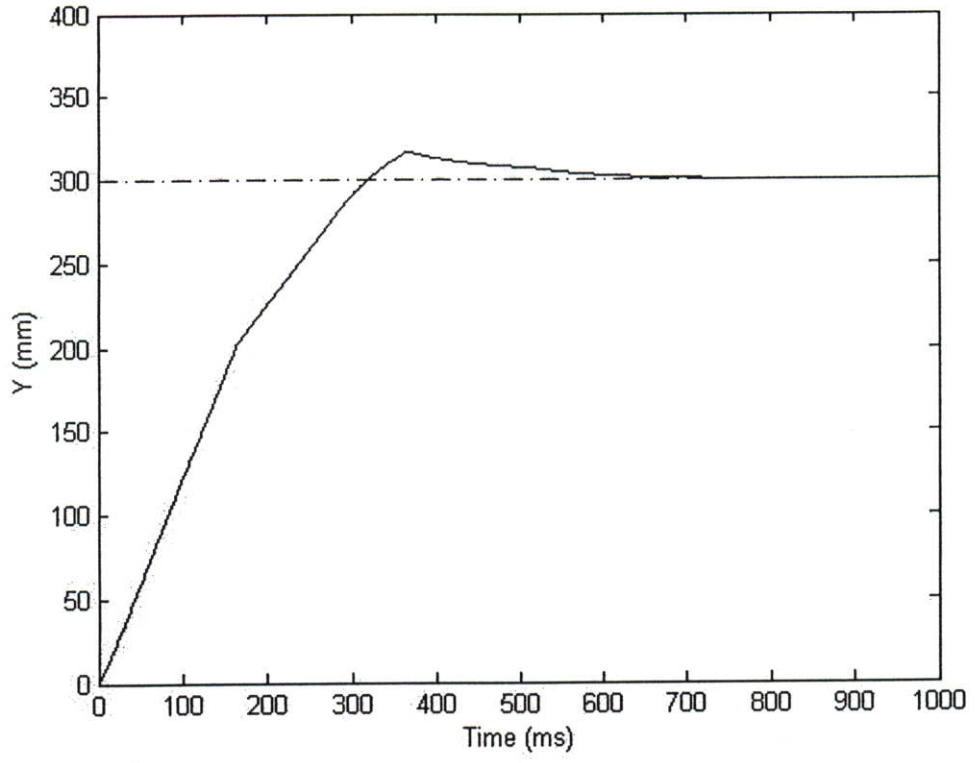
การทดลองนี้จะใช้การนำร่องสำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง4 ล้อ ซึ่งกำหนดค่าตัวแปรเหมือนกับกรณีการทดลองกับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทางสามล้อทุกประการ

7.3.1 การนำร่องของ FDFT สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 4 ล้อ

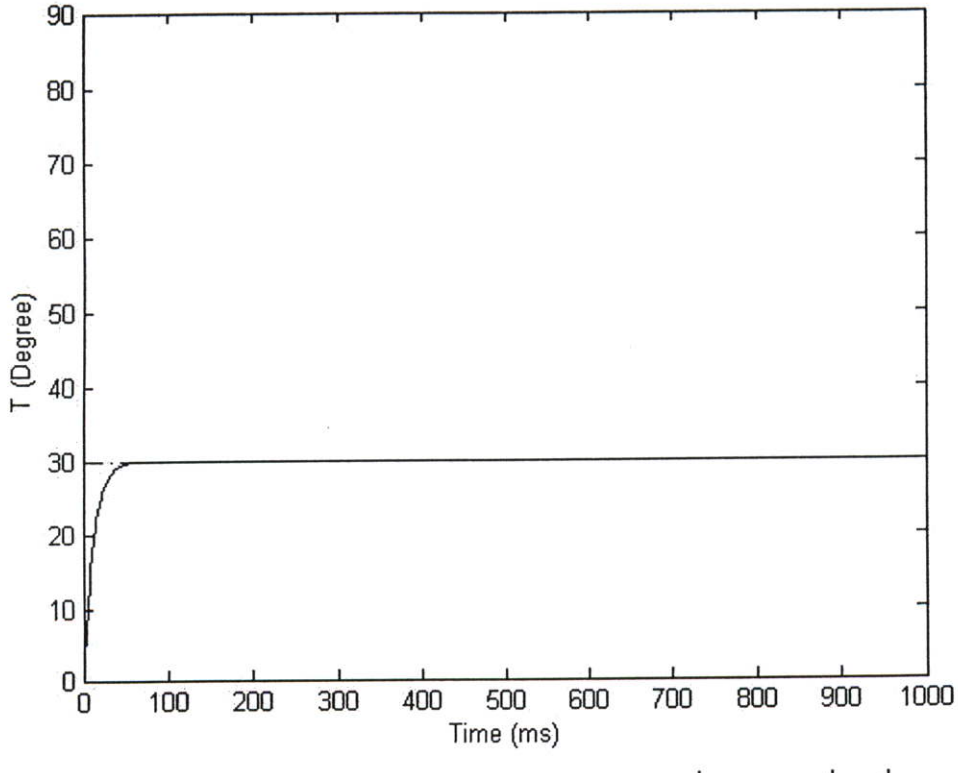
การทดลองนี้แสดงผลการนำร่องแบบ FDFT ซึ่งใช้การควบคุมความเร็วการวิ่งและควบคุมการหันหน้าแบบฟัซซี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 4 ล้อ



รูปที่ 7.25 ระยะในแกน X ของการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



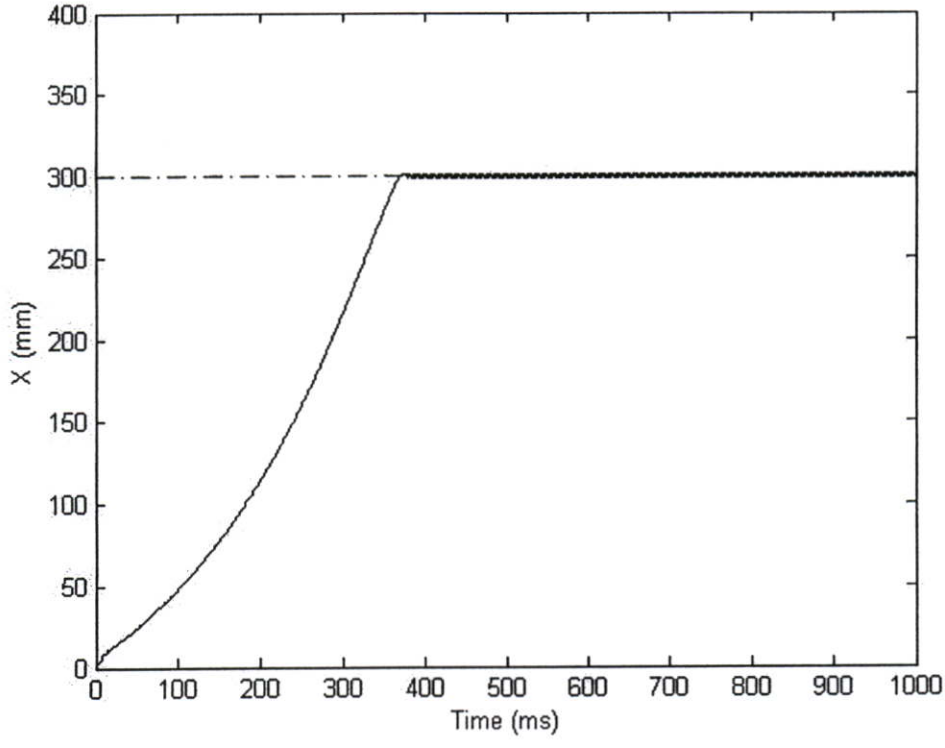
รูปที่ 7.26 ระยะในแกน Y ของการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



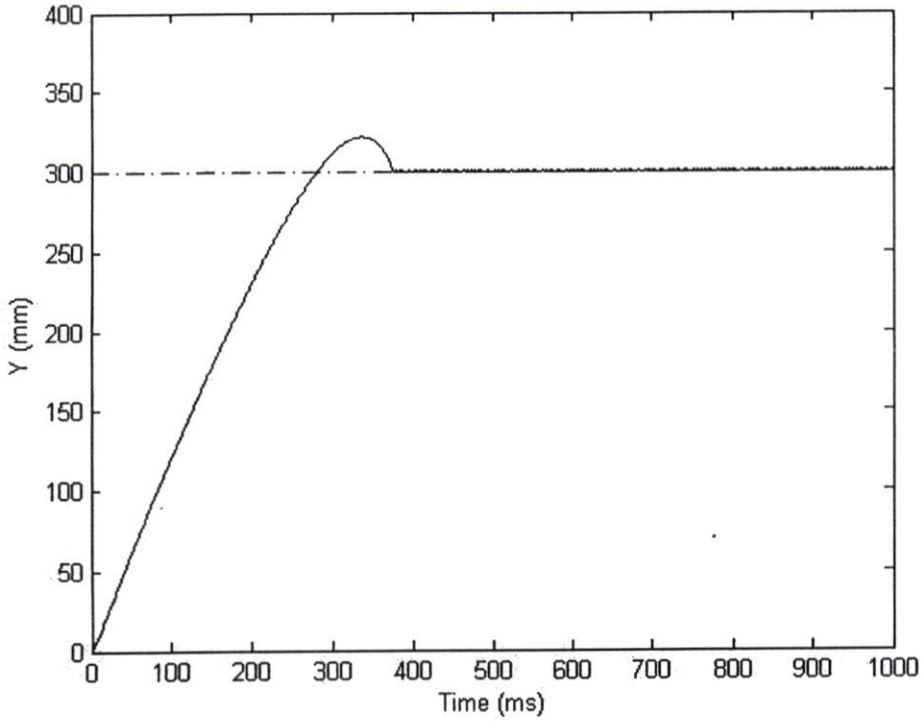
รูปที่ 7.27 องศาด้านหน้าการนำร่องของ FDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ

7.3.2 การนำร่องของ FDPT สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 4 ล้อ

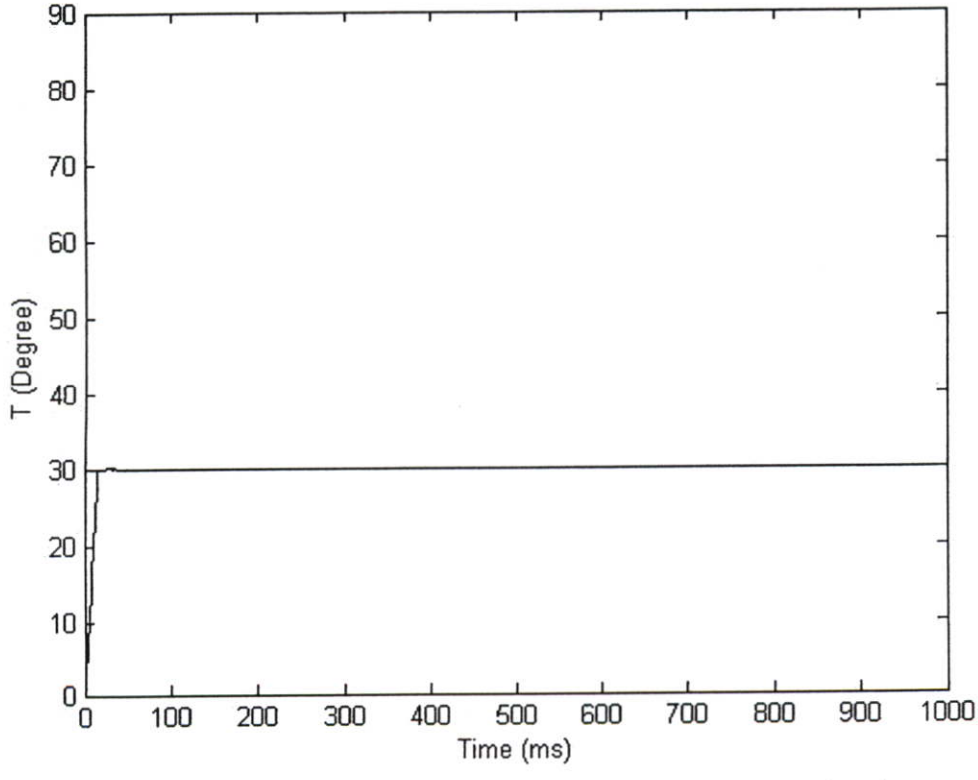
การทดลองนี้แสดงผลการนำร่องแบบ FDPT ซึ่งใช้การควบคุมความเร็วการวิ่งแบบฟัซซี และควบคุมการหันหน้าแบบพีไอดี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 4 ล้อ



รูปที่ 7.28 ระยะในแกน X ของการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



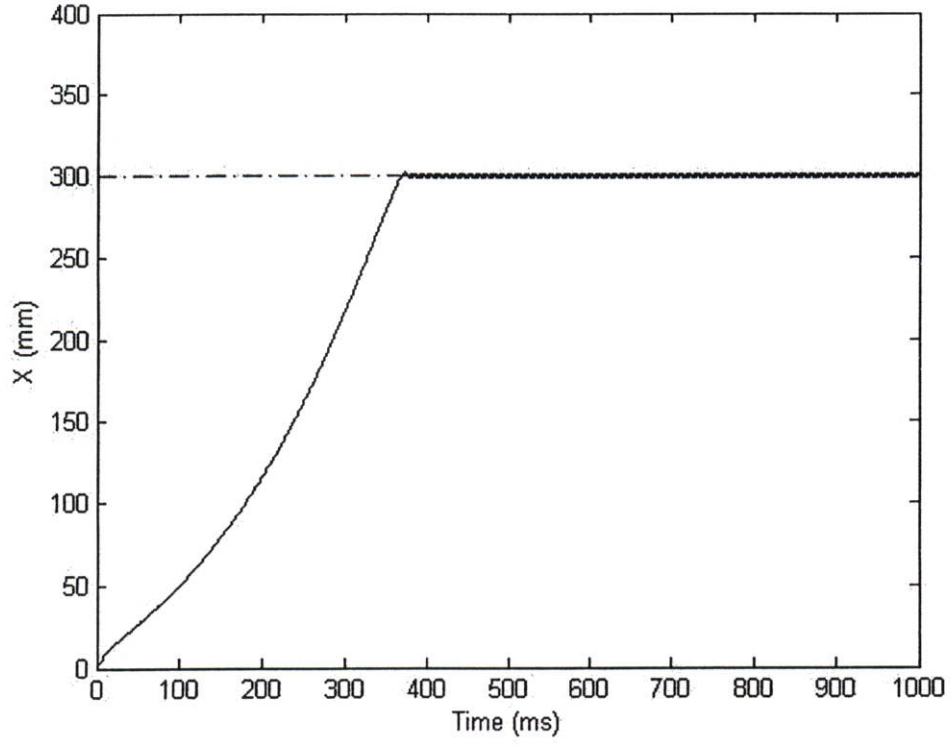
รูปที่ 7.29 ระยะในแกน Y ของการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



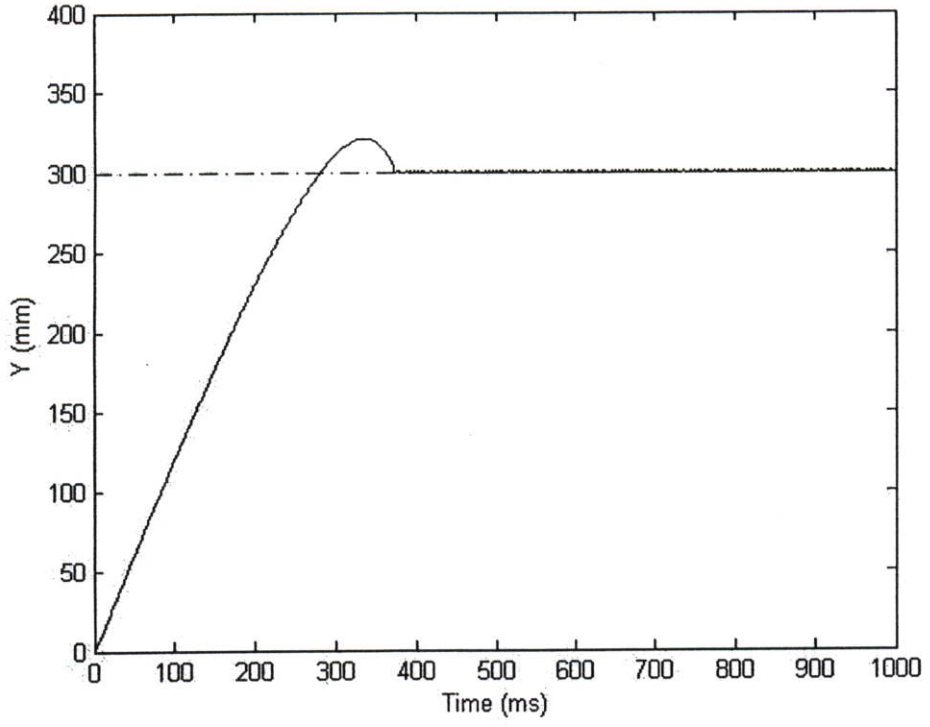
รูปที่ 7.30 องศาด้านหน้าการนำร่องของ FDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ

7.3.3 การนำร่องของ PDFT สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 4 ล้อ

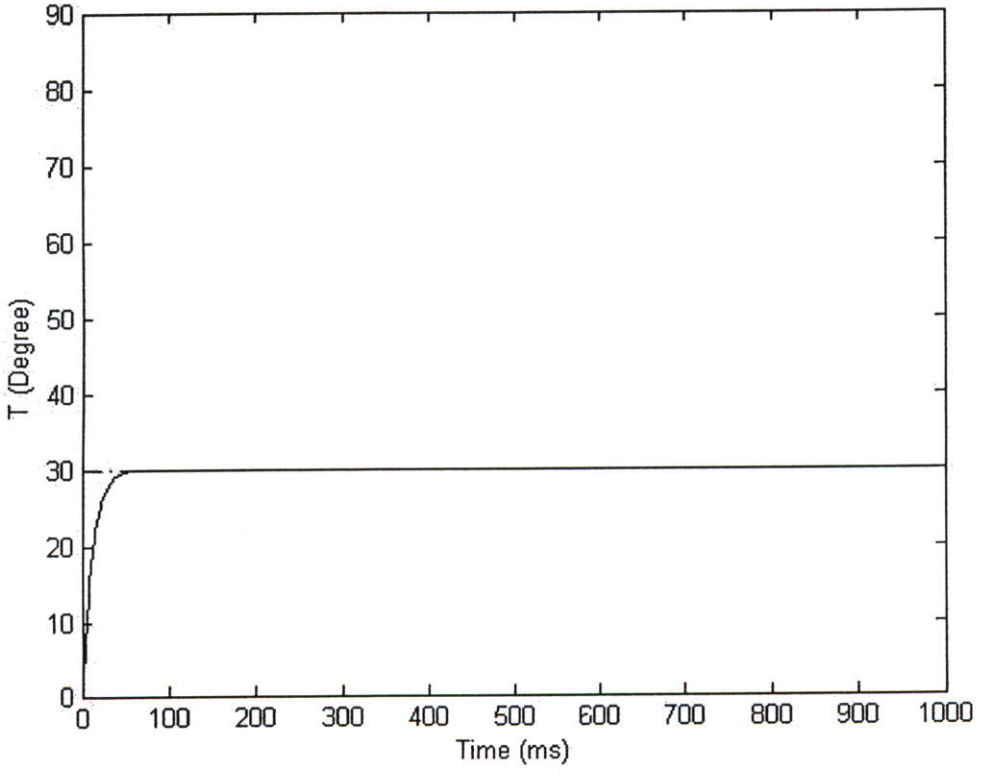
การทดลองนี้แสดงผลการนำร่องแบบ PDFT ซึ่งใช้การควบคุมความเร็วการวิ่งแบบพีไอดี และควบคุมการหันหน้าแบบฟัซซี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 4 ล้อ



รูปที่ 7.31 ระยะในแกน X ของการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



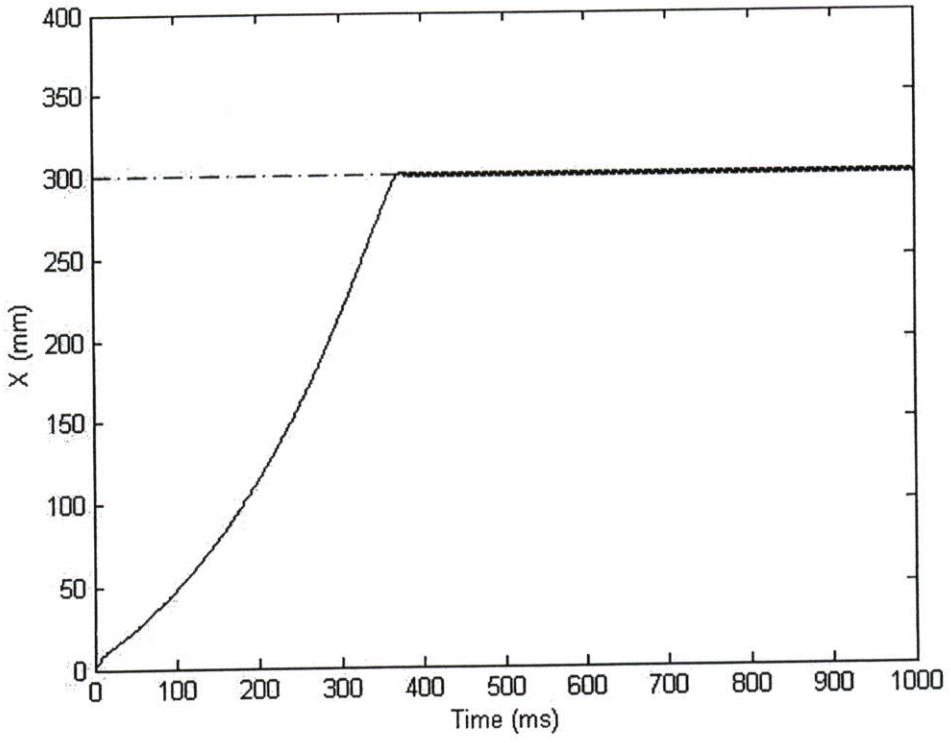
รูปที่ 7.32 ระยะในแกน Y ของการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



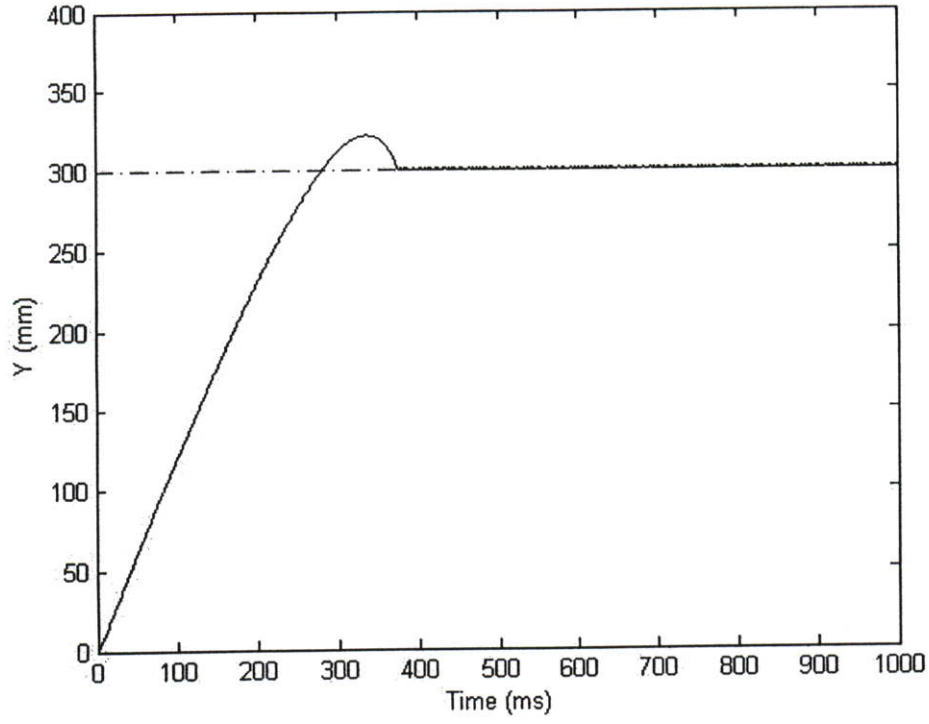
รูปที่ 7.33 องศาด้านหน้าการนำร่องของ PDFT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ

7.3.4 การนำร่องของ PDPT สำหรับระบบขับเคลื่อนเคลื่อนที่ทุกทิศทาง 4 ล้อ

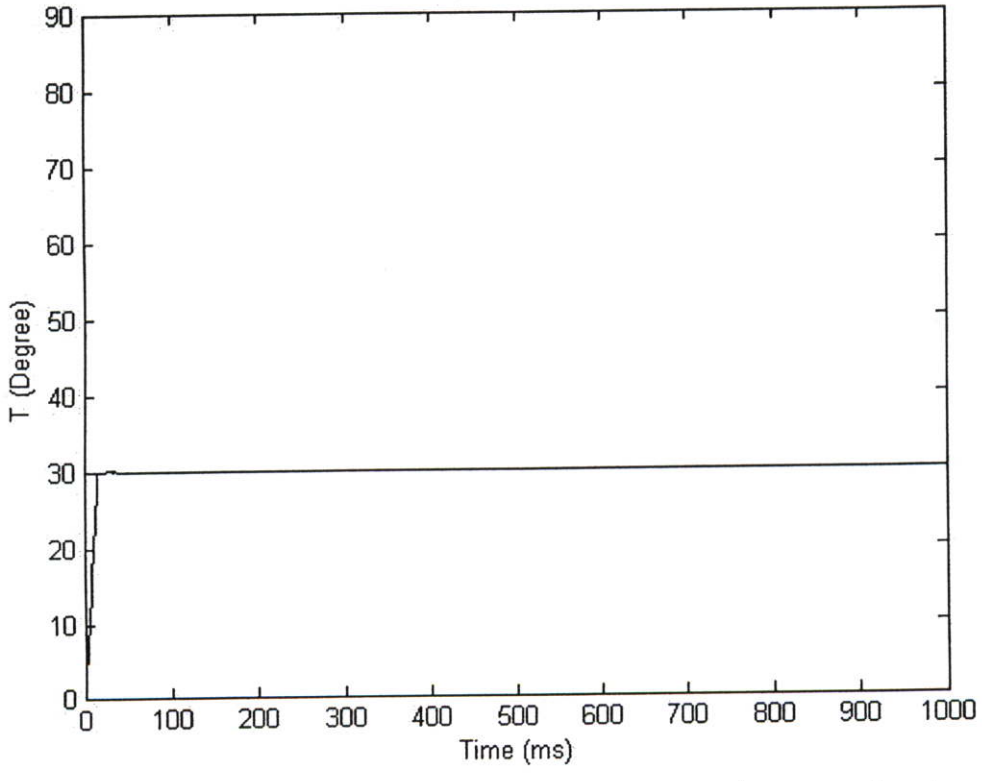
การทดลองนี้แสดงผลการนำร่องแบบ PDPT ซึ่งใช้การควบคุมความเร็วการวิ่งและควบคุมการหันหน้าแบบพีไอดี โดยการนำร่องนี้ใช้สำหรับหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 4 ล้อ



รูปที่ 7.34 ระยะในแกน X ของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



รูปที่ 7.35 ระยะในแกน Y ของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ



รูปที่ 7.36 องศาด้านหน้าของการนำร่องของ PDPT สำหรับระบบขับเคลื่อนแบบเคลื่อนที่รอบทิศทาง 4 ล้อ

จากรูปที่ 7.25-7.26 รูปที่ 7.28-29. รูปที่ 7.31-7.32 และรูปที่ 7.34-7.35 แสดงให้เห็นว่าหุ่นยนต์ได้วิ่งเข้าจุดเป้าหมายอยู่ที่ตำแหน่ง (300,300) ได้อย่างถูกต้อง ซึ่งลักษณะการวิ่งจะแตกต่างกันไปตามกฎการนำร่องและเข้าจุดได้เร็วกว่าหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 3 ล้อเพราะหุ่นยนต์เคลื่อนที่ในแนวด้านข้างเหมือนกันแต่มีคุณสมบัติทางกายภาพที่ดีกว่า(มีล้อมากกว่า) นอกจากนี้ รูปที่ 7.27 รูปที่ 7.30 รูปที่ 7.33 และรูปที่ 7.36 แสดงให้เห็นว่าหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 4 ล้อสามารถหันหน้าได้ทันทีและรวดเร็วกว่าหุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ เพราะไม่จำเป็นต้องรอให้ถึงตำแหน่งเป้าหมาย ซึ่งเป็นข้อได้เปรียบทางกายภาพ

7.4 การวัดประสิทธิภาพของการนำร่อง

ประสิทธิภาพของการนำร่องสามารถวัดได้โดยใช้ดัชนีวัดประสิทธิภาพของระบบควบคุมทั่วไป ซึ่งดัชนีการวัดประสิทธิภาพที่ใช้คือ Integrate Absolute Error (IAE) และ Integrate Square Error (ISE) ซึ่งประสิทธิภาพจะแปรผันกลับกับค่า IAE และ ISE หมายความว่าหากการนำร่องมีประสิทธิภาพมาก ค่า IAE และ ISE จะมีค่าน้อย

สมการ (7.1) แสดงการคำนวณ Integrate Absolute Error และสมการ (7.2) แสดงการคำนวณ Integrate Square Error โดยตาราง 7.1 แสดงผลการวัดประสิทธิภาพของการนำร่องในส่วนระยะทางและตาราง 7.2 แสดงผลการวัดประสิทธิภาพของการนำร่องในส่วนมุมด้านหน้าของหุ่นยนต์เล่นฟุตบอล

$$IAE = \sum_{n=0}^{N} \left| \frac{p_s - p_n}{t_s} \right| \tag{7.1}$$

$$ISE = \sum_{n=0}^{N} \frac{(p_s - p_n)^2}{t_s} \tag{7.2}$$

โดย n เป็นลำดับของช่วงเวลา

N เป็นจำนวนของช่วงเวลาที่วัดผล

p_s เป็นค่าเป้าหมาย

p_n เป็นค่าที่วัดได้ในช่วงเวลานั้น

t_s เป็นเวลาสุ่ม (Sampling Time)

ตารางที่ 7.1 ผลการวัดประสิทธิภาพของการนำร่องในส่วนระยะทาง

	2 ล้อ		3 ล้อ		4 ล้อ	
	IAE	ISE	IAE	ISE	IAE	ISE
FDFT	3.7647×10^4	1.4899×10^7	3.23×10^4	1.205×10^7	3.507×10^4	1.378×10^7
FDPT	3.5623×10^4	1.3353×10^7	3.2874×10^4	1.2430×10^7	3.5403×10^4	1.401×10^7
PDFT	3.6079×10^4	1.37×10^7	3.2863×10^4	1.2430×10^7	3.5413×10^4	1.4016×10^7
PDPT	3.7564×10^4	1.4832×10^7	3.2874×10^4	1.2430×10^7	3.5403×10^4	1.401×10^7

ตารางที่ 7.2 ผลการวัดประสิทธิภาพของการนำร่องในส่วนมุมด้านหน้าของหุ่นยนต์

	2 ล้อ		3 ล้อ		4 ล้อ	
	IAE	ISE	IAE	ISE	IAE	ISE
FDFT	6.43×10^2	1.895×10^4	3.4197×10^1	5.1393×10^2	3.42×10^1	5.14×10^2
FDPT	3.16×10^2	5.977×10^3	2.2146×10^1	4.27×10^2	2.22×10^1	4.27×10^2
PDFT	4.015×10^2	8.997×10^3	3.4197×10^1	5.1393×10^2	3.42×10^1	5.14×10^2
PDPT	2.76×10^2	5.187×10^3	2.2146×10^1	4.27×10^2	2.22×10^1	4.27×10^2

จากตารางที่ 7.1 จะเห็นได้ว่าการนำร่องแบบ FDPT มีประสิทธิภาพสูงสุดในส่วนระยะทางสำหรับหุ่นยนต์เล่นฟุตบอลแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ ส่วนการนำร่องแบบ FDFT มีประสิทธิภาพสูงสุดในส่วนระยะทางสำหรับหุ่นยนต์เล่นฟุตบอลแบบเคลื่อนที่ทุกทิศทาง 3 ล้อและ 4 ล้อ และจากตารางที่ 7.2 จะเห็นได้ว่าการนำร่องแบบ PDPT มีประสิทธิภาพสูงสุดในส่วนมุมด้านหน้าสำหรับหุ่นยนต์เล่นฟุตบอลแบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ ส่วนการนำร่องแบบ FDPT และ PDPT มีประสิทธิภาพสูงสุดในส่วนระยะทางสำหรับหุ่นยนต์เล่นฟุตบอลแบบเคลื่อนที่ทุกทิศทาง 3 ล้อและ 4 ล้อ

7.5 ผลการแข่งขันในการทดสอบบนโปรแกรมจำลองสถานการณ์

การทดสอบระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอลที่พัฒนาขึ้นบนโปรแกรมจำลองสถานการณ์ เป็นการทดสอบประสิทธิภาพของแผนการเล่นต่างๆ ว่ามีประสิทธิภาพเท่าใด โดยกำหนดตัวแปรสิ่งแวดล้อมให้เท่าเทียมกันทั้งหมด

กำหนดให้การทดสอบเป็นการแข่งขันระหว่างทีม RED กับ ทีม BLUE ซึ่งประกอบด้วยหุ่นยนต์เล่นฟุตบอลจำนวน 5 ตัวเท่ากัน ใช้การนำร่องแบบ FDPT เหมือนกัน และในการแข่งขันแต่ละนัดใช้เวลา 5 นาที โดยผลการแข่งขันซึ่งจัดการแบบพบกันหมดแสดงในตารางที่ 7.3

ตารางที่ 7.3 ผลการแข่งขันในการทดสอบบนโปรแกรมจำลองสถานการณ์

แผนการเล่น	1-3-0-1	1-2-1-1	1-1-2-1	1-0-3-1	0-0-4-1
1-3-0-1	0-0	0-0	0-1	0-1	0-2
1-2-1-1	x	0-0	0-0	0-0	0-1
1-1-2-1	x	x	0-1	1-0	0-1
1-0-3-1	x	x	x	0-0	0-1
0-0-4-1	x	x	x	x	2-1

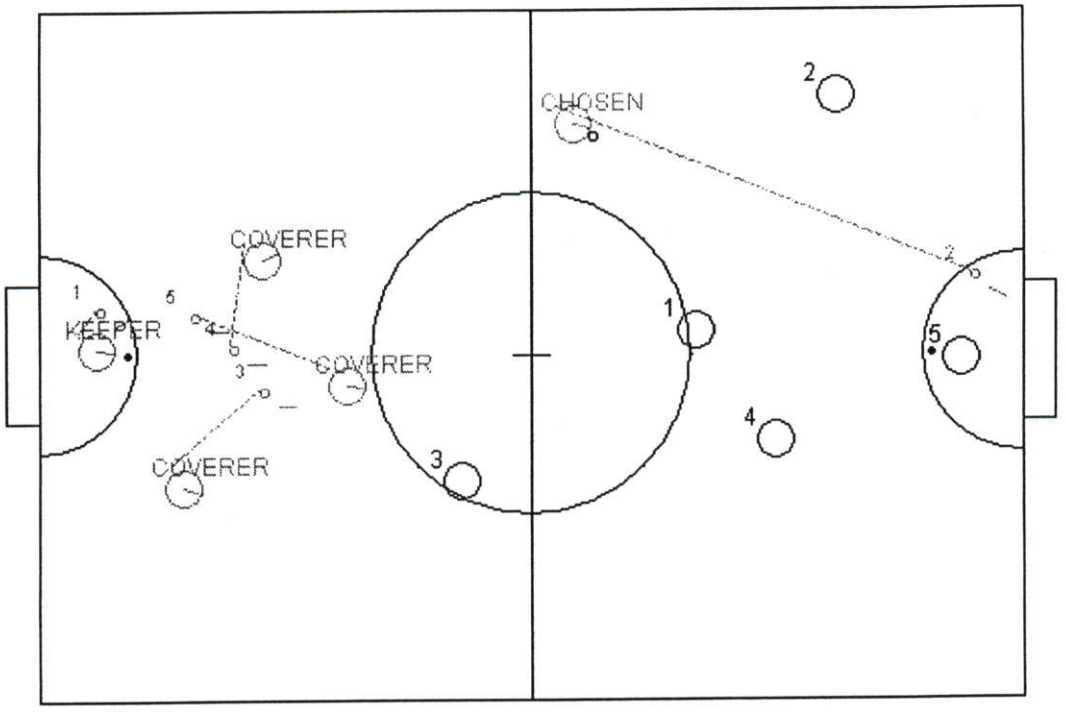
จากตารางที่ 7.3 จะเห็นได้ว่าแผนที่เน้นการรุกให้ผลการแข่งขันที่ดีกว่าแผนที่เน้นการป้องกัน

7.6 รูปแบบการเล่นของหุ่นยนต์เล่นฟุตบอลในการจำลองสถานการณ์

การทดลองนี้เป็นการทดลองว่าระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอลที่พัฒนาขึ้นบนโปรแกรมจำลองสถานการณ์ทำงานตามที่ได้กำหนดไว้หรือไม่ โดยกำหนดตัวแปรสิ่งแวดล้อมให้เท่าเทียมกันทั้งหมด กำหนดให้การทดลองเป็นการแข่งขันระหว่างทีม RED กับ ทีม BLUE ซึ่งประกอบด้วยหุ่นยนต์เล่นฟุตบอลจำนวน 5 ตัวเท่ากัน ใช้การนำร่องแบบ FDPT โดยการตรวจสอบการทำงานโดยจับภาพการเล่นจากโปรแกรมปัญญาประดิษฐ์ของฝ่ายทีม RED

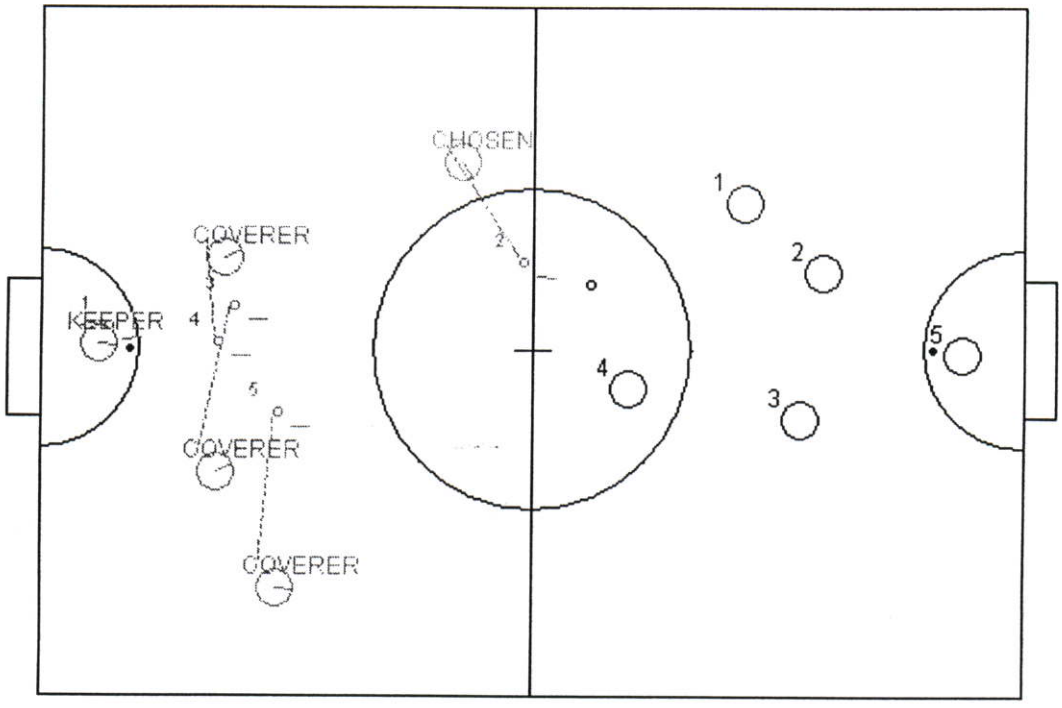
7.6.1 รูปแบบการเล่นของแผน P1301 ในการจำลองสถานการณ์

การทดลองนี้เป็นการทดลองใช้แผน P1301 คือ มีหุ่นยนต์ตำแหน่งผู้รักษาประตู 1 ตัว ตำแหน่งตัวบังการยิง 3 ตัว ตำแหน่งตัวไล่บอลและรุก 1 ตัว โดยไม่มีหุ่นยนต์ตำแหน่งตัวประกบ



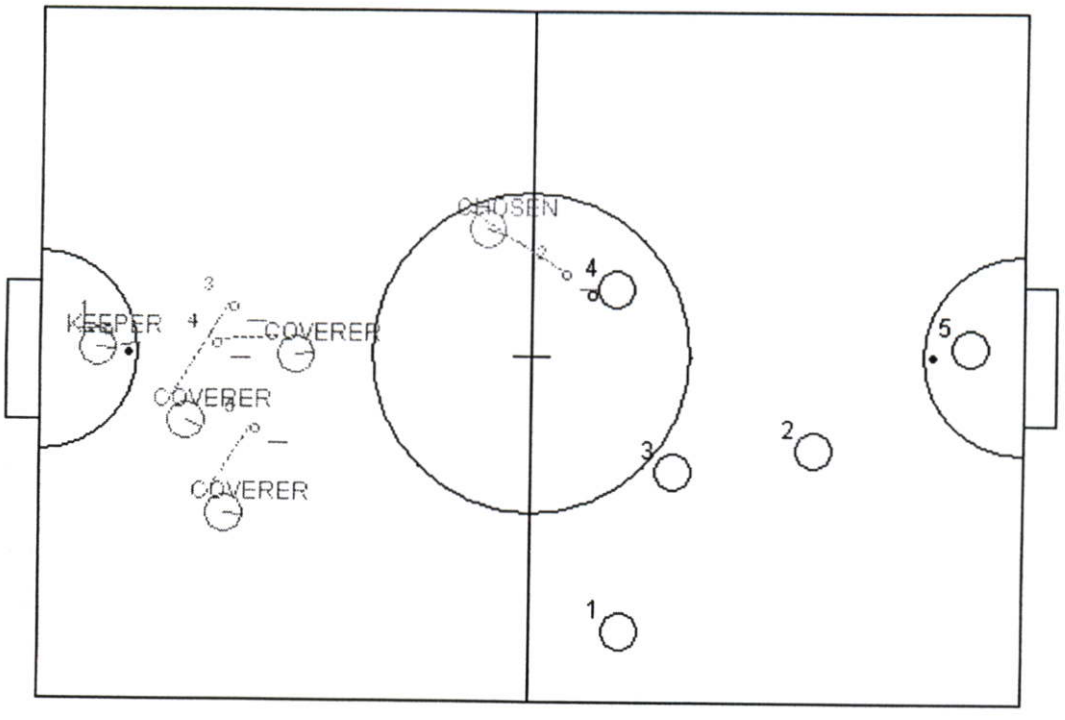
รูปที่ 7.37 รูปแบบการเล่นของแผน P1301 ในสถานการณ์รุก

จากรูปที่ 7.37 แสดงให้เห็นว่าในสถานการณ์รุกเมื่อทีมใช้แผน P1301 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามนำลูกฟุตบอลเข้าไปยังประตูฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



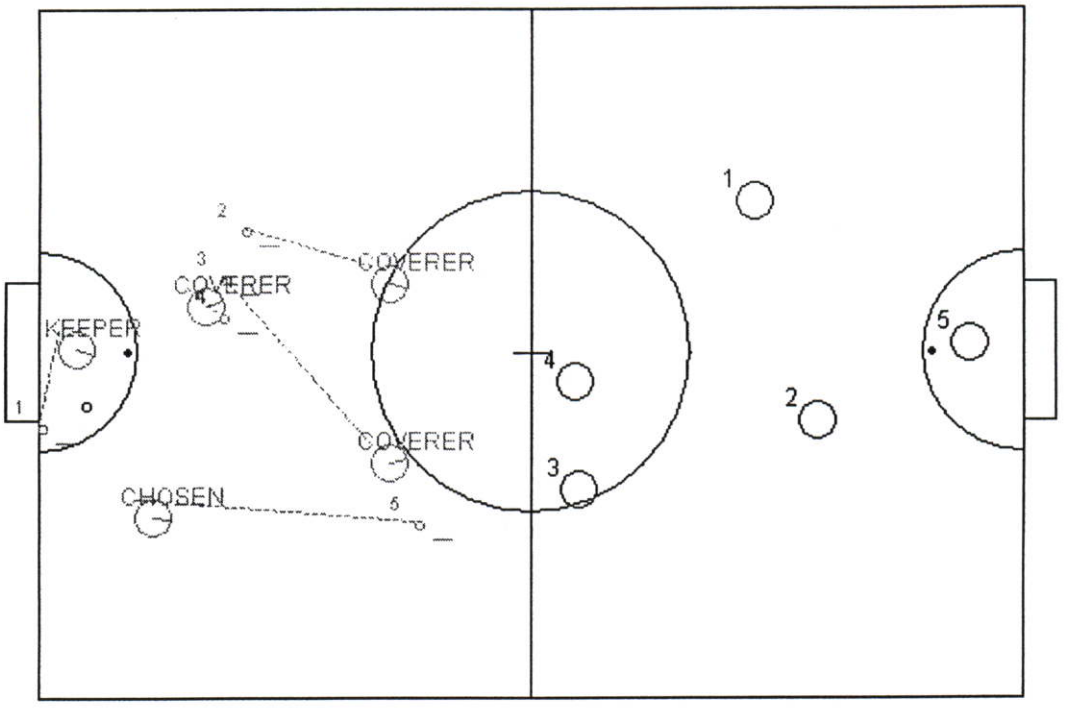
รูปที่ 7.38 รูปแบบการเล่นของแผน P1301 ในสถานการณ์เก็บบอล

จากรูปที่ 7.38 แสดงให้เห็นว่าในสถานการณ์เก็บบอลเมื่อทีมใช้แผน P1301 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปยังตำแหน่งด้านหลังลูกฟุตบอล หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



รูปที่ 7.39 รูปแบบการเล่นของแผน P1301 ในสถานการณ์ป้องกัน

จากรูปที่ 7.39 แสดงให้เห็นว่าในสถานการณ์ป้องกันเมื่อทีมใช้แผน P1301 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปแย่งลูกฟุตบอลจากหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู

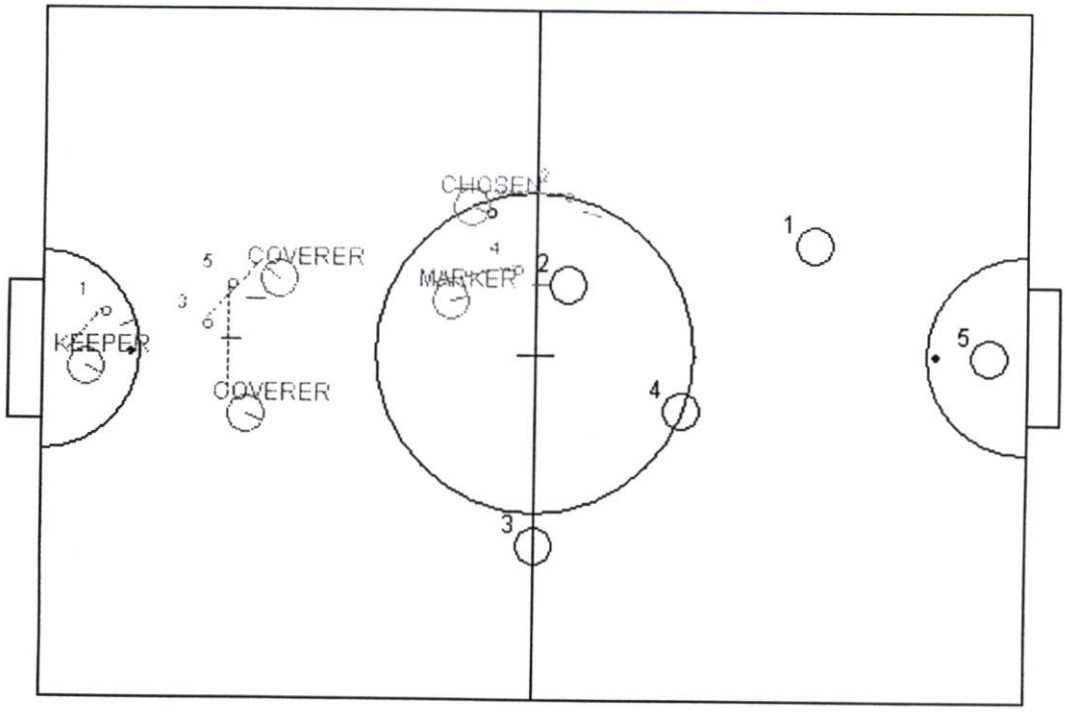


รูปที่ 7.40 รูปแบบการเล่นของแผน P1301 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู

จากรูปที่ 7.40 แสดงให้เห็นว่าในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตูเมื่อทีมใช้แผน P1301 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดเปิดทาง และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งอ้อมด้านหลังลูกฟุตบอล

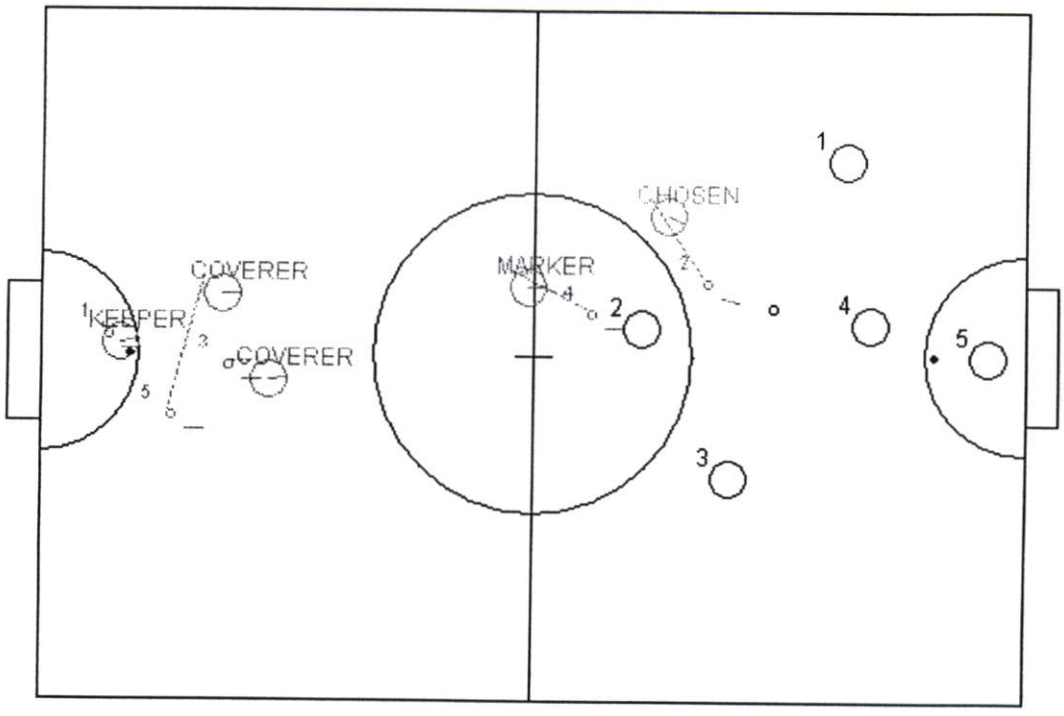
7.6.2 รูปแบบการเล่นของแผน P1211 ในการจำลองสถานการณ์

การทดลองนี้เป็นการทดลองใช้แผน P1211 คือ มีหุ่นยนต์ตำแหน่งผู้รักษาประตู 1 ตัว ตำแหน่งตัวบังการยิง 2 ตัว ตำแหน่งตัวประกบ 1 ตัว และตำแหน่งตัวไล่บอลและรุก 1 ตัว



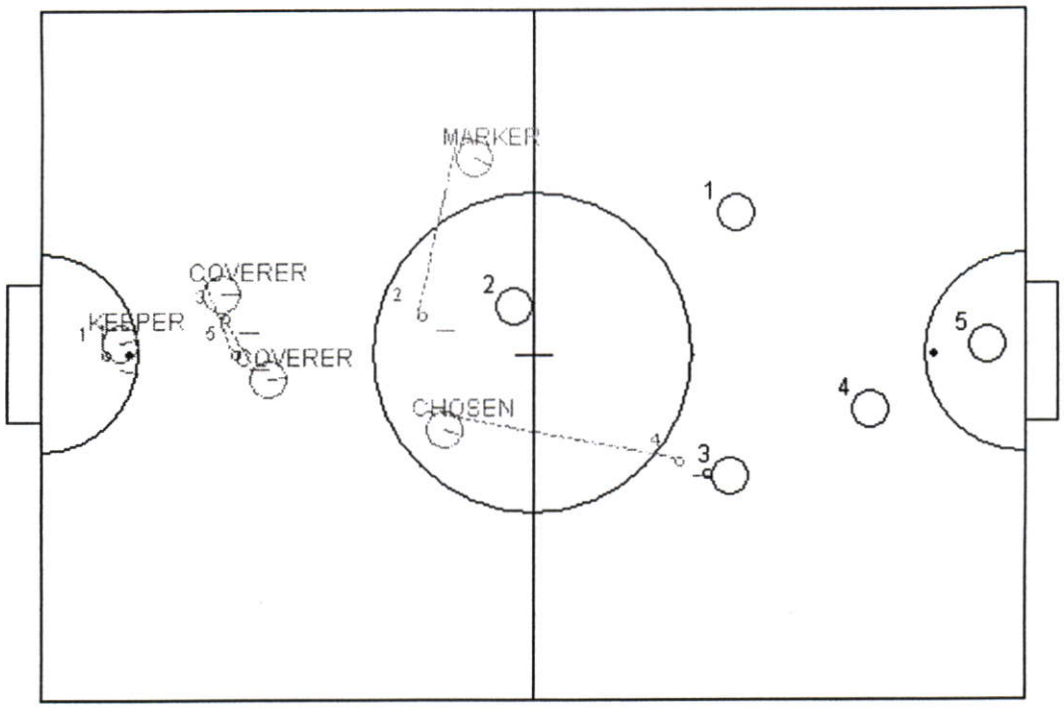
รูปที่ 7.41 รูปแบบการเล่นของแผน P1211 ในสถานการณ์รุก

จากรูปที่ 7.41 แสดงให้เห็นว่าในสถานการณ์รุกเมื่อทีมใช้แผน P1211 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามนำลูกฟุตบอลเข้าไปยังประตูฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



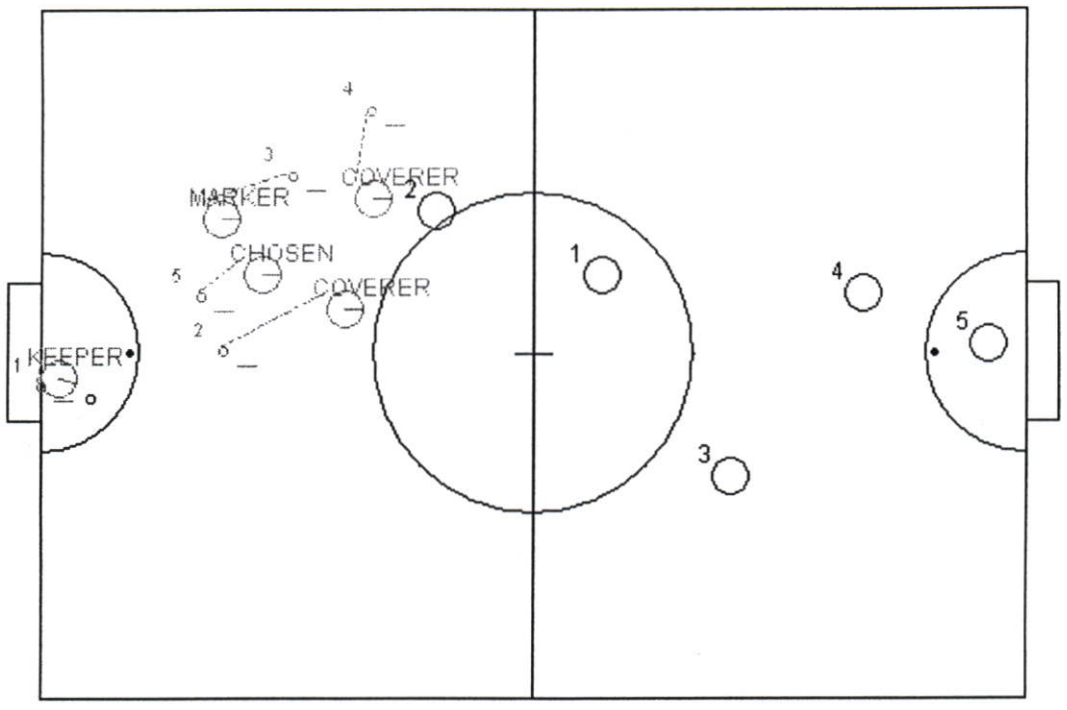
รูปที่ 7.42 รูปแบบการเล่นของแผน P1211 ในสถานการณ์เก็บบอล

จากรูปที่ 7.42 แสดงให้เห็นว่าในสถานการณ์เก็บบอลเมื่อทีมใช้แผน P1211 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปยังตำแหน่งด้านหลังลูกฟุตบอล หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



รูปที่ 7.43 รูปแบบการเล่นของแผน P1211 ในสถานการณ์ป้องกัน

จากรูปที่ 7.43 แสดงให้เห็นว่าในสถานการณ์ป้องกันเมื่อทีมใช้แผน P1211 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปแย่งลูกฟุตบอลจากหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู

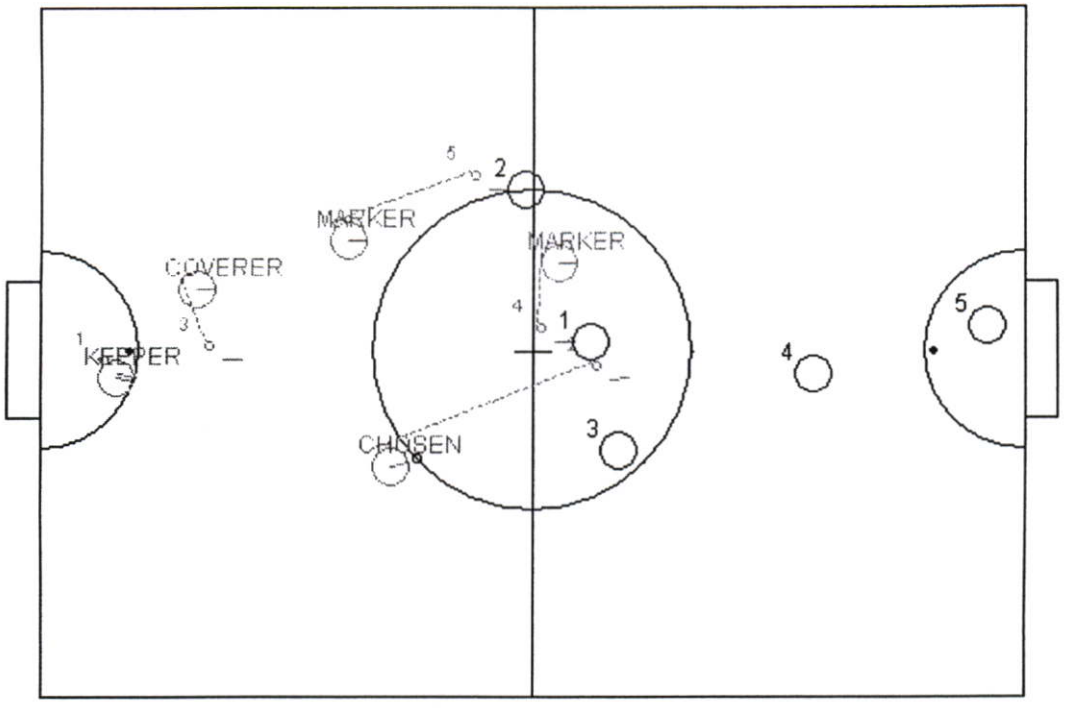


รูปที่ 7.44 รูปแบบการเล่นของแผน P1211 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู

จากรูปที่ 7.44 แสดงให้เห็นว่าในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตูเมื่อทีมใช้แผน P1211 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดเปิดทาง และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งอ้อมด้านหลังลูกฟุตบอล

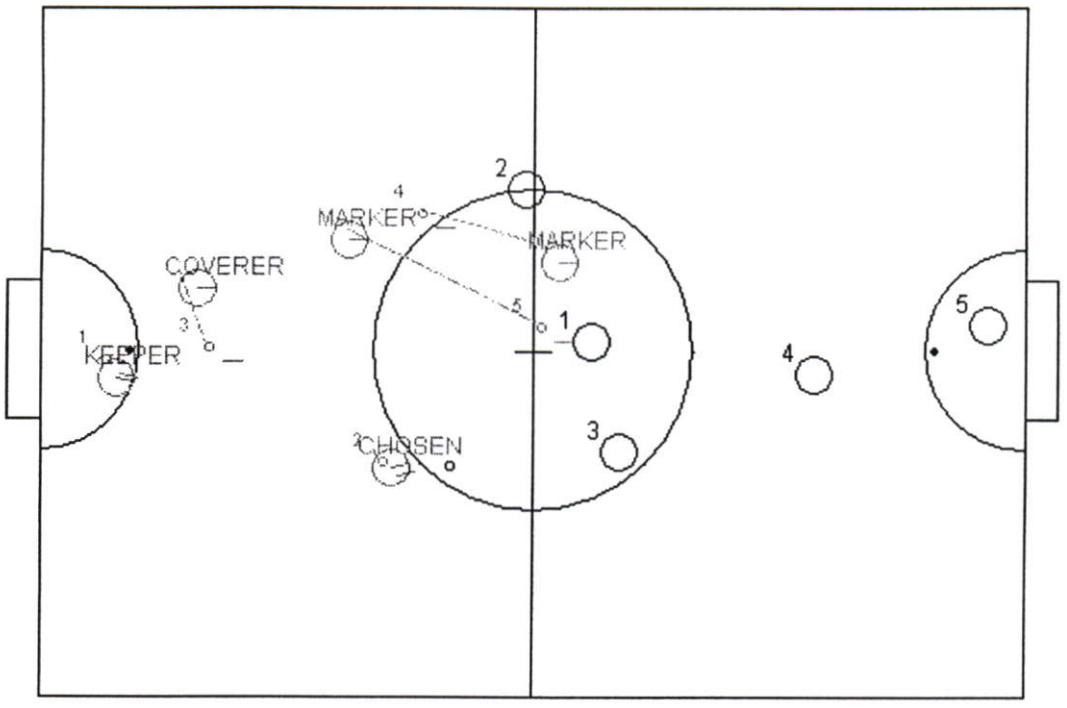
7.6.3 รูปแบบการเล่นของแผน P1121 ในการจำลองสถานการณ์

การทดลองนี้เป็นการทดลองใช้แผน P1121 คือ มีหุ่นยนต์ตำแหน่งผู้รักษาประตู 1 ตัว ตำแหน่งตัวบังการยิง 1 ตัว ตำแหน่งตัวประกบ 2 ตัว และตำแหน่งตัวไล่บอลและรุก 1 ตัว



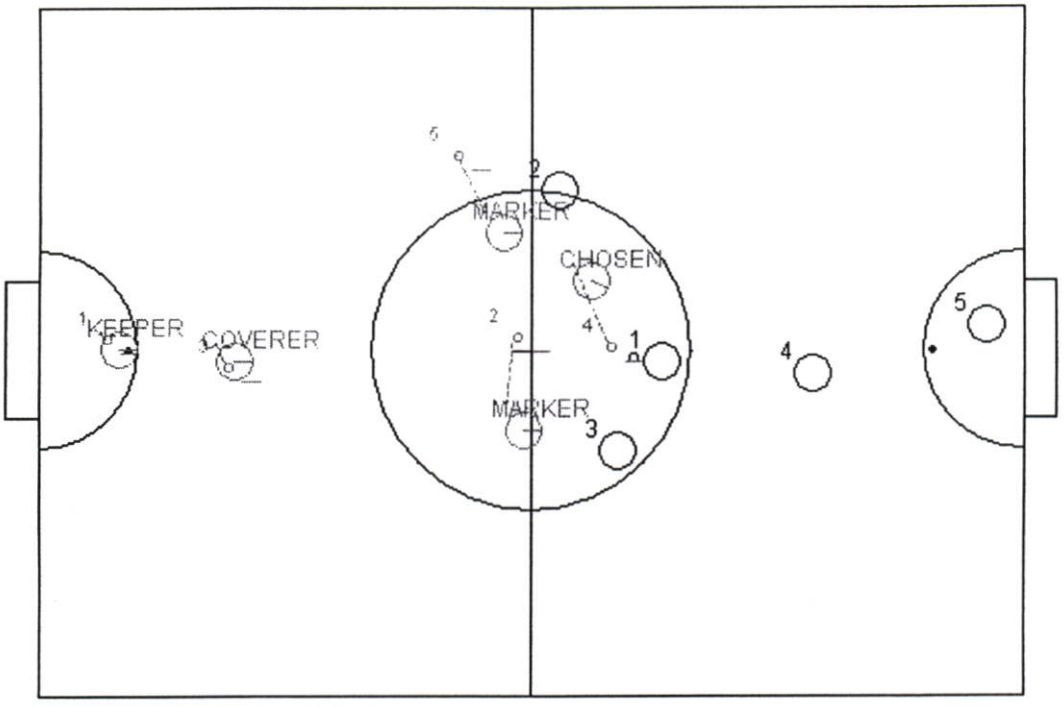
รูปที่ 7.45 รูปแบบการเล่นของแผน P1121 ในสถานการณ์รุก

จากรูปที่ 7.45 แสดงให้เห็นว่าในสถานการณ์รุกเมื่อทีมใช้แผน P1121 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามนำลูกฟุตบอลเข้าไปยังประตูฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



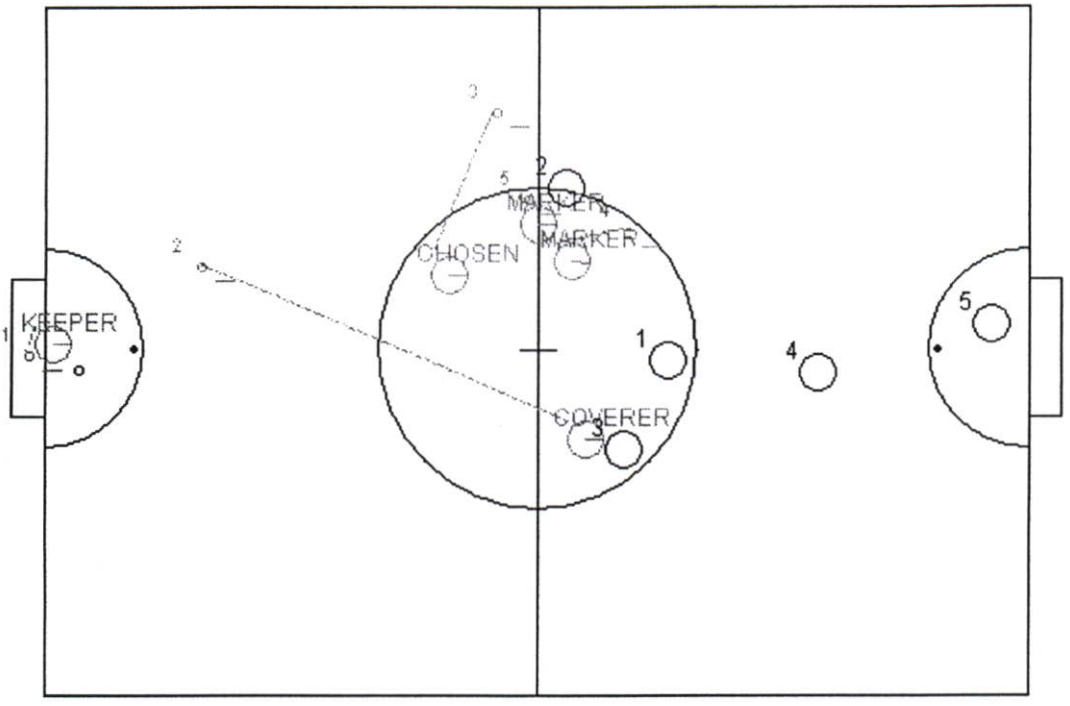
รูปที่ 7.46 รูปแบบการเล่นของแผน P1121 ในสถานการณ์เก็บบอล

จากรูปที่ 7.46 แสดงให้เห็นว่าในสถานการณ์เก็บบอลเมื่อทีมใช้แผน P1121 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปยังตำแหน่งด้านหลังลูกฟุตบอล หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



รูปที่ 7.47 รูปแบบการเล่นของแผน P1121 ในสถานการณ์ป้องกัน

จากรูปที่ 7.47 แสดงให้เห็นว่าในสถานการณ์ป้องกันเมื่อทีมใช้แผน P1121 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปแย่งลูกฟุตบอลจากหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่นอกเขตผู้รักษาประตู และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู

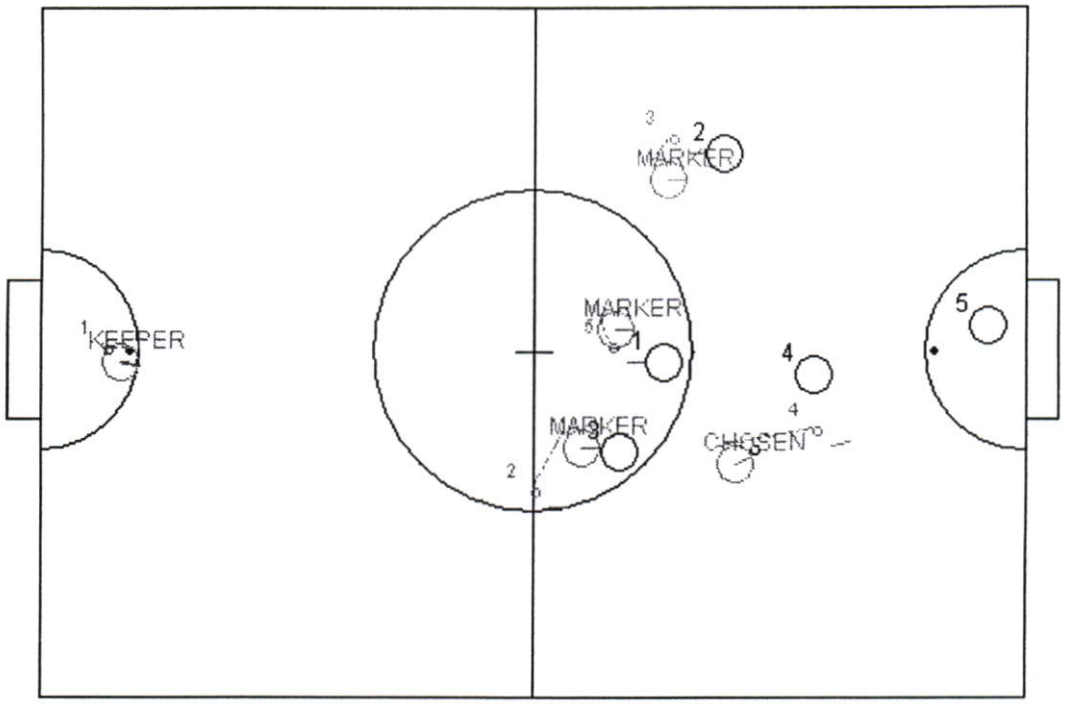


รูปที่ 7.48 รูปแบบการเล่นของแผน P1121 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู

จากรูปที่ 7.48 แสดงให้เห็นว่าในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตูเมื่อทีมใช้แผน P1121 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวบังการยิงจะวิ่งไปยังจุดเปิดทาง และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งอ้อมด้านหลังลูกฟุตบอล

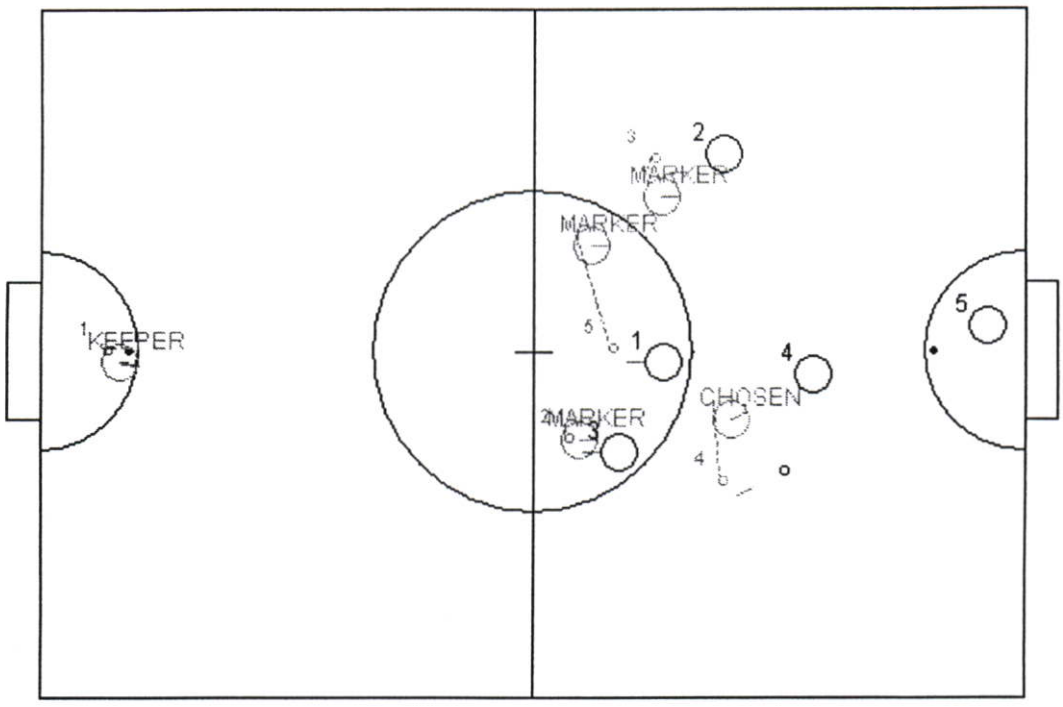
7.6.4 รูปแบบการเล่นของแผน P1031 ในการจำลองสถานการณ์

การทดลองนี้เป็นการทดลองใช้แผน P1031 คือ มีหุ่นยนต์ตำแหน่งผู้รักษาประตู 1 ตัว ตำแหน่งตัวประกบ 3 ตัว ตำแหน่งตัวไล่บอลและรุก 1 ตัว โดยไม่มีหุ่นยนต์ตำแหน่งตัวบังการยิง



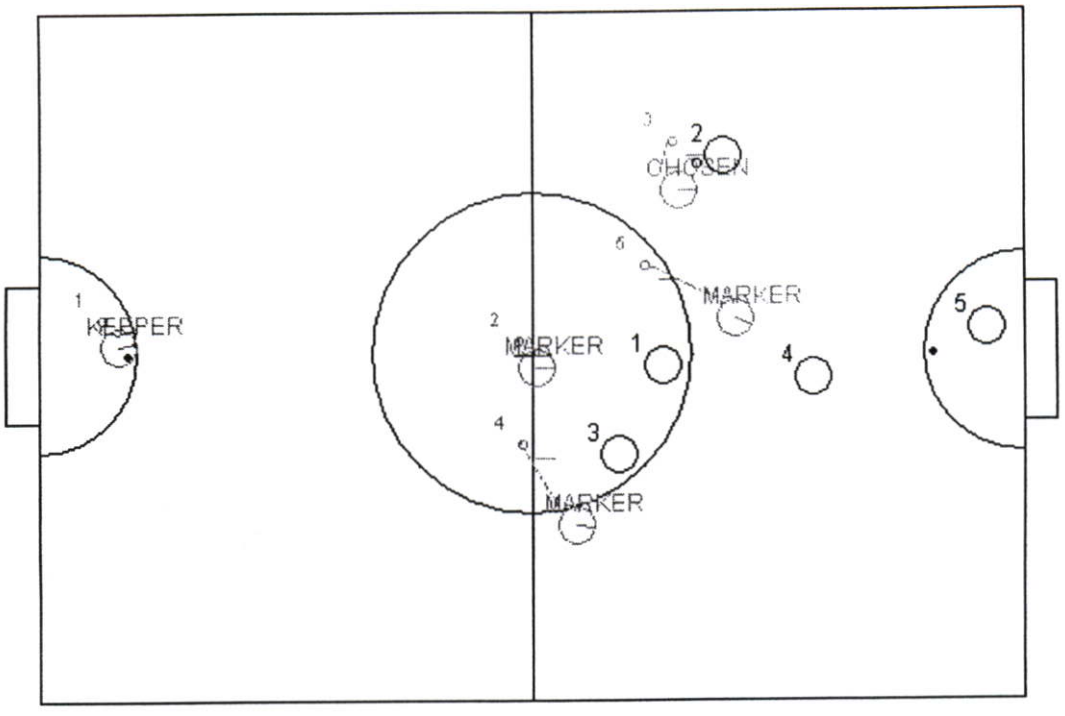
รูปที่ 7.49 รูปแบบการเล่นของแผน P1031 ในสถานการณ์รุก

จากรูปที่ 7.49 แสดงให้เห็นว่าในสถานการณ์รุกเมื่อทีมใช้แผน P1031 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามนำลูกฟุตบอลเข้าไปยังประตูฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



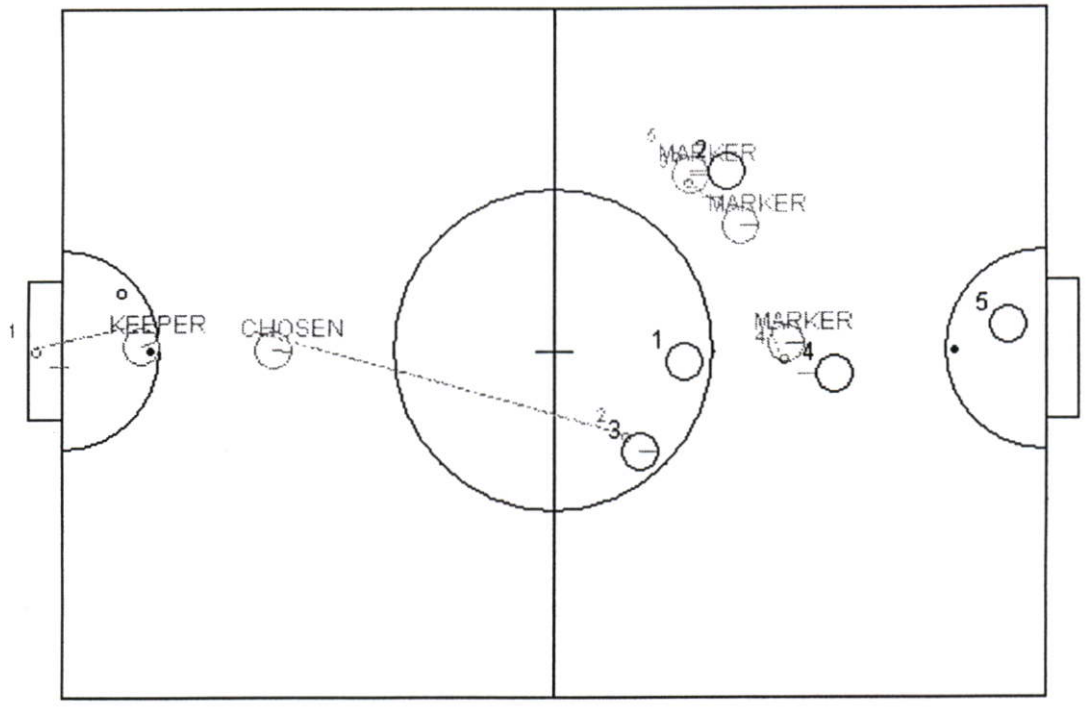
รูปที่ 7.50 รูปแบบการเล่นของแผน P1031 ในสถานการณ์เก็บบอล

จากรูปที่ 7.50 แสดงให้เห็นว่าในสถานการณ์เก็บบอลเมื่อทีมใช้แผน P1031 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปยังตำแหน่งด้านหลังลูกฟุตบอล หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู



รูปที่ 7.51 รูปแบบการเล่นของแผน P1031 ในสถานการณ์ป้องกัน

จากรูปที่ 7.51 แสดงให้เห็นว่าในสถานการณ์ป้องกันเมื่อทีมใช้แผน P1031 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปแย่งลูกฟุตบอลจากหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งไปยังจุดบังทิศทางการยิงประตูที่อยู่ในเขตผู้รักษาประตู

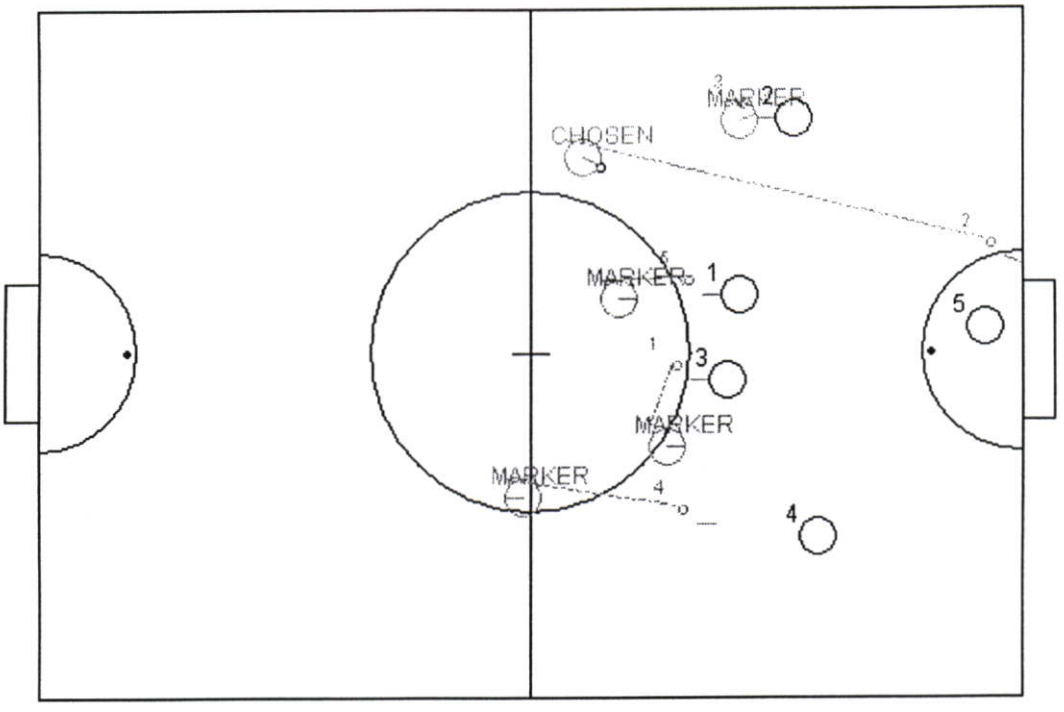


รูปที่ 7.52 รูปแบบการเล่นของแผน P1031 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู

จากรูปที่ 7.52 แสดงให้เห็นว่าในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตูเมื่อทีมใช้แผน P1031 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม หุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม และหุ่นยนต์ตำแหน่งผู้รักษาประตูจะวิ่งอ้อมด้านหลังลูกฟุตบอล

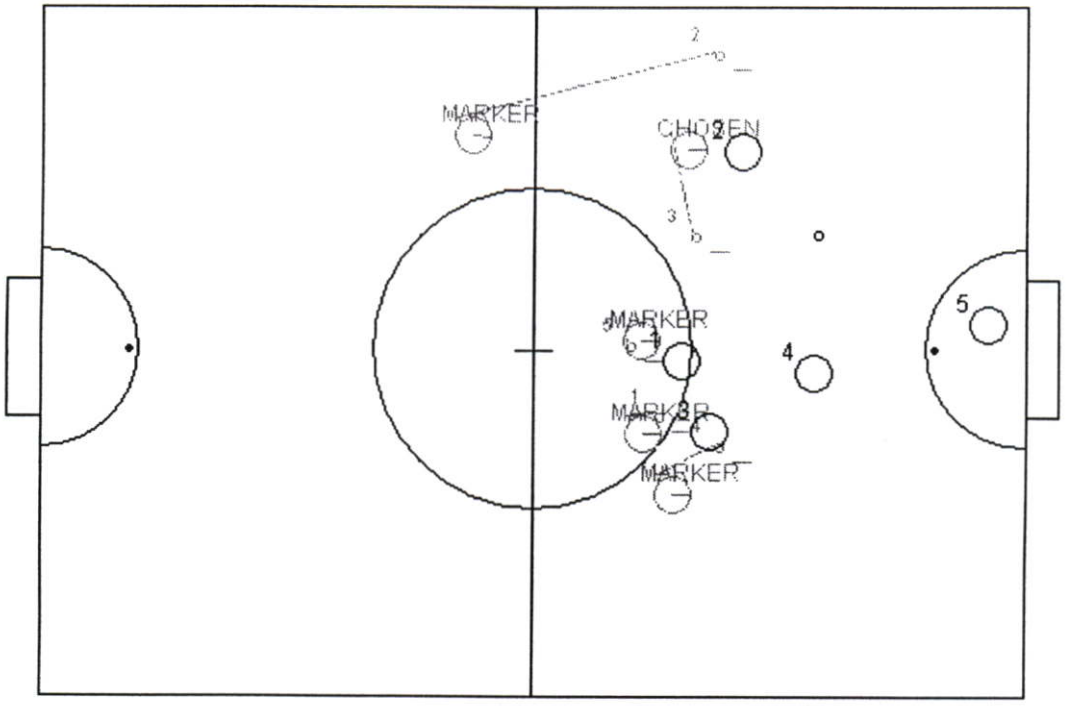
7.6.5 รูปแบบการเล่นของแผน P0041 ในการจำลองสถานการณ์

การทดลองนี้เป็นการทดลองใช้แผน P0041 คือ มีหุ่นยนต์ตำแหน่งตัวประกบ 4 ตัว ตำแหน่งตัวไล่บอลและรุก 1 ตัว โดยไม่มีหุ่นยนต์ตำแหน่งตัวบังการยิง และผู้รักษาประตู



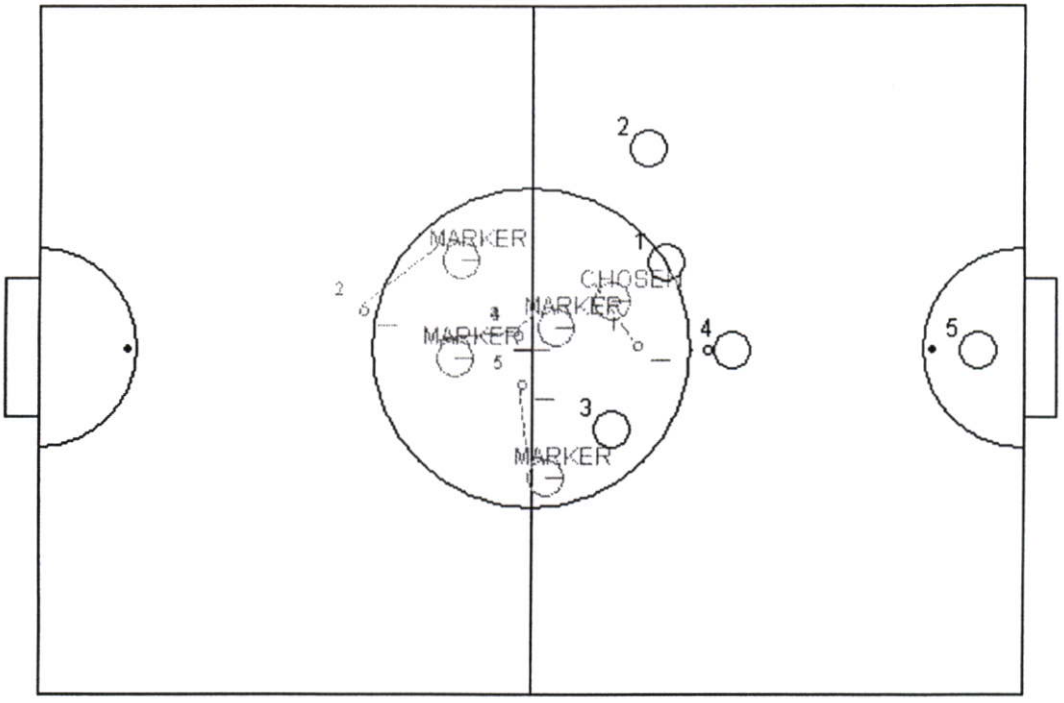
รูปที่ 7.53 รูปแบบการเล่นของแผน P0041 ในสถานการณ์รุก

จากรูปที่ 7.53 แสดงให้เห็นว่าในสถานการณ์รุกเมื่อทีมใช้แผน P0041 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามนำลูกฟุตบอลเข้าไปยังประตูฝ่ายตรงข้าม และหุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม



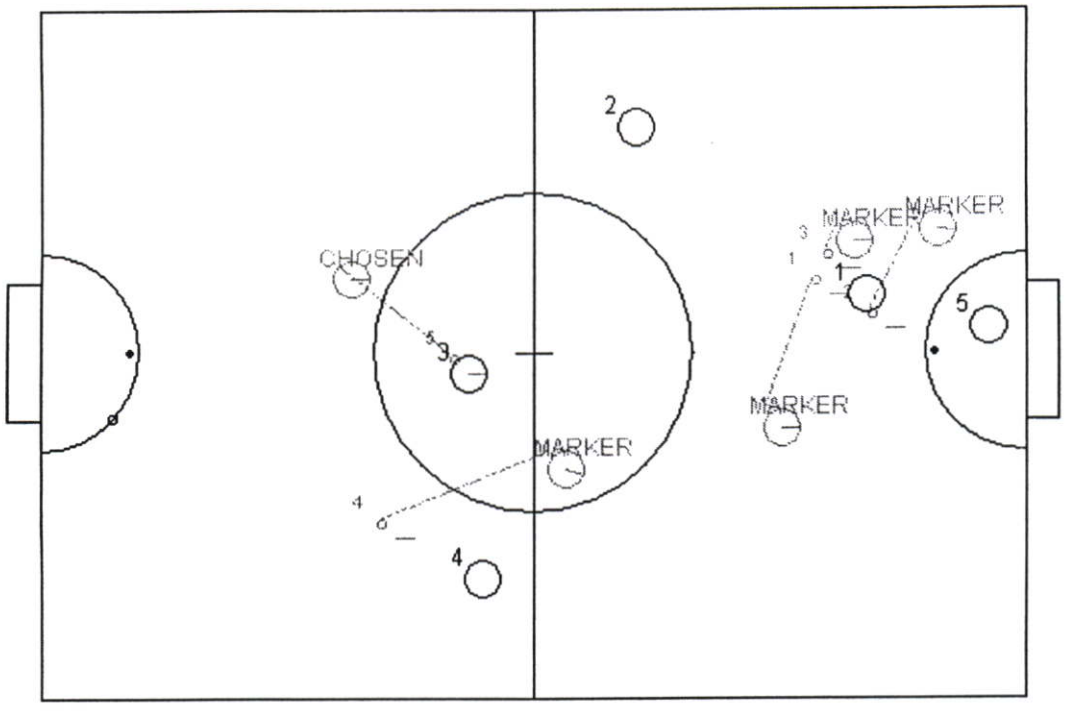
รูปที่ 7.54 รูปแบบการเล่นของแผน P0041 ในสถานการณ์เก็บบอล

จากรูปที่ 7.54 แสดงให้เห็นว่าในสถานการณ์เก็บบอลเมื่อทีมใช้แผน P0041 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปยังตำแหน่งด้านหลังลูกฟุตบอล และหุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม



รูปที่ 7.55 รูปแบบการเล่นของแผน P0041 ในสถานการณ์ป้องกัน

จากรูปที่ 7.55 แสดงให้เห็นว่าในสถานการณ์ป้องกันเมื่อทีมใช้แผน P0041 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปแย่งลูกฟุตบอลจากหุ่นยนต์ฝ่ายตรงข้าม และหุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม



รูปที่ 7.56 รูปแบบการเล่นของแผน P0041 ในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตู

จากรูปที่ 7.56 แสดงให้เห็นว่าในสถานการณ์ลูกฟุตบอลอยู่ในเขตประตูเมื่อทีมใช้แผน P0041 หุ่นยนต์ตำแหน่งตัวไล่บอลและรุกจะพยายามวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม และหุ่นยนต์ตำแหน่งตัวประกบจะวิ่งไปประกบหุ่นยนต์ฝ่ายตรงข้าม

ในบทที่ 8 ซึ่งเป็นบทต่อไปจะสรุปผลการทดลองทั้งหมดรวมทั้งข้อแนะนำในการวิจัยและแนวทางในการพัฒนาในอนาคต

บทที่ 8
สรุปการวิจัยและข้อเสนอแนะ

งานวิจัยระบบหลากหลายตัวกระทำสำหรับหุ่นยนต์เล่นฟุตบอล เป็นการนำระบบหลากหลายตัวกระทำสำหรับการควบคุมมาประยุกต์ใช้งานสำหรับหุ่นยนต์เล่นฟุตบอลโดยเฉพาะ โดยจุดประสงค์ของงานวิจัยนี้เน้นที่การแสดงวิธีการประยุกต์ระบบตัวกระทำและการประยุกต์ระบบการควบคุมต่างๆมาใช้กับงานหุ่นยนต์เล่นฟุตบอลมากกว่าการทำการวิจัยเชิงเปรียบเทียบ ซึ่งผลการทดลองที่ได้แสดงนั้นได้แสดงถึงผลลัพธ์ของงานวิจัยตามที่ได้กำหนดจุดประสงค์เอาไว้

8.1 สรุปผลการทดลอง

จากการทดลองได้นำการประยุกต์วิธีการควบคุมแบบฟัซซีและพีไอดีมาใช้ในการนำร่องหุ่นยนต์เล่นฟุตบอลให้วิ่งไปในตำแหน่งต่างๆได้ตามที่กำหนดจุดเป้าหมายและองศาด้านหน้าของหุ่นยนต์ ผลการทดลองแสดงให้เห็นว่าสามารถนำวิธีการควบคุมดังกล่าวมาใช้ในการนำร่องได้เป็นอย่างดี

สำหรับระบบนำร่องของหุ่นยนต์เล่นฟุตบอลที่มีการขับเคลื่อนแบบเก้าอี้รถเข็นสองล้อ จะเห็นว่าสามารถนำร่องให้ไปยังจุดเป้าหมายได้ แต่ยังวิ่งเข้าจุดนำร่องอย่างไม่พอดี เพราะเนื่องจากหุ่นยนต์จะต้องเปลี่ยนจากการหันหน้าเข้าสู่จุดเป้าหมายมาเป็นการหันหน้าเพื่อเข้าสู่องศาที่ได้กำหนดไว้เมื่อเข้าใกล้จุดเป้าหมาย ซึ่งเป็นข้อจำกัดของแบบจำลองเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ

ระบบนำร่องสำหรับหุ่นยนต์เล่นฟุตบอลที่มีการขับเคลื่อนแบบเคลื่อนที่ได้ทุกทิศทางทั้งแบบ 3 ล้อและ4 ล้อ จะเห็นได้ว่าสามารถนำร่องให้ไปยังจุดเป้าหมาย และวิ่งเข้าจุดนำร่องได้อย่างพอดี เพราะผลจากการที่หุ่นยนต์สามารถเคลื่อนที่ในแนวด้านข้างได้และไม่มีการสลับการควบคุมการหันหน้า

ผลการทดลองในการทดสอบประสิทธิภาพของการนำร้องได้แสดงให้เห็นว่า หุ่นยนต์แบบเก้าอี้รถล้อเลื่อนคนพิการ 2 ล้อ การนำร่องแบบ FDPT จะให้ประสิทธิภาพของการนำร่องในส่วนระยะทางได้ดีที่สุดและการนำร่องแบบ PDPT ให้ประสิทธิภาพของการนำร่องในส่วนมุมด้านหน้าของหุ่นยนต์ได้ดีที่สุด ส่วนหุ่นยนต์แบบเคลื่อนที่ทุกทิศทาง 3 ล้อและ 4 ล้อ การนำร่องแบบ FDFT ให้ประสิทธิภาพของการนำร่องในส่วนระยะทางได้ดีที่สุดและการนำร่องแบบ FDPT และ PDPT ให้ประสิทธิภาพของการนำร่องในส่วนมุมด้านหน้าของหุ่นยนต์ได้ดีที่สุด

ผลการแข่งขันในการทดสอบระบบหลากหลายตัวกระทำบนโปรแกรมจำลองสถานการณ์แสดงให้เห็นว่าระบบสามารถทำงานได้จริง โดยผลการแข่งขันในตาราง 7.3 แสดงให้เห็นว่าเมื่อ

กำหนดสิ่งแวดล้อมให้เท่าเทียมกันแล้ว แผนการเล่นแบบบุกจะให้ผลการแข่งขันที่ดีกว่าแผนการเล่นแบบตั้งรับ

สำหรับผลการทดลองในส่วนรูปแบบของหุ่นยนต์ในการจำลองสถานการณ์นั้น ผลการทดลองซึ่งได้ใช้การจับภาพจากโปรแกรมปัญญาประดิษฐ์แสดงให้เห็นว่า หุ่นยนต์เล่นฟุตบอลนั้นทำงานตามคำสั่งของตัวกระทำได้อย่างถูกต้อง

8.2 ข้อแนะนำในการวิจัยและแนวทางในการพัฒนาในอนาคต

หากดูจากการทดลองแล้ว จะเห็นได้ว่ามีตัวแปรหลายอย่างที่ยังสามารถปรับเปลี่ยนได้ ซึ่งส่งผลให้ได้ผลการทดลองที่แตกต่างจากเดิม ซึ่งเป็นแนวทางในการปรับเปลี่ยนนั้นควรที่จะมุ่งเน้นที่จะให้ได้ประสิทธิภาพที่ดียิ่งขึ้น

หากสังเกตแบบจำลองของหุ่นยนต์เล่นฟุตบอลนั้นซึ่งเป็นแบบจำลองที่ใช้ในการทดลองและการจำลองสถานการณ์ แม้ว่าเป็นแบบจำลองที่ใช้ในการแข่งขันหุ่นยนต์เล่นฟุตบอลในระดับการแข่งขันที่สภาวะแบบจำลองที่เป็นที่ยอมรับทั่วโลก แต่ว่าก็ยังไม่เหมือนหุ่นยนต์เล่นฟุตบอลที่ใช้ในการแข่งขันในระดับที่ใช้หุ่นยนต์จริง ซึ่งส่งผลให้หากนำระบบการนำร่องไปใช้โดยมิได้ปรับแต่ง จะทำให้ระบบทำงานได้ไม่เต็มประสิทธิภาพ

บรรณานุกรม

[1] R. Engelmore, and T. Morgan, **Blackboard Systems**, Addison-Wesley, 1988.

[2] H. Kitano, M. Asada, I. Noda, and H. Matsubara, **"RobotCup: Robot World Cup"**, IEEE Robotics & Automation Magazine, Vol. 5, pp. 30-36, 1998.

[3] V. Lesser, R. Fennell, L. Erman and D. Reddy, **"Organization of the Hearsay II Speech Understanding System"**, IEEE Transactions on Signal Processing, Vol. 23, pp. 11-24, 1975.

[4] M. Occello and Y. Demazeau, **"Building Real Time Agents using Parallel Blackboards and its use for Mobile Robotics"**, IEEE International Conference on System, Man, and Cybernatics, Vol.2, pp. 1610-1615, 1994.

[5] M Occello and M. C. Thomas, **"A Distributed Blackboards Methodology for Designing Robotic Control Softwares"**, IEEE International Conference on Systems Engineering, pp. 147-150, 1992.

[6] M. Occello and M.C. Thomas, **"A New Approach for Process Control"** , Proceedings of Cumputers in Design Manufacturing and Production, pp. 487-492, 1993.

[7] L. Wu, S. Zhao and X. Xu, **"Design of an Obstacle-avoidance Strategy based on Robot Soccer Competition"**, 5th International Conference on Signal Processing, Vol. 3, pp. 1722-1725, 2000.

[8] T.-H S. Li, C. A. Lai and Y. J. Guo, **"Design of Fuzzy Field Control for a One-on-one Robot Soccer System"**, IEEE 28th Annual Conference of the Industrial Electronics Society, Vol. 4, pp. 2605-2610, 2002.

[9] T.-H. S. Li, R. C. Liu and I-F. Lin, **"Fuzzy Shooting Control of Car-like Soccer Robot"**, Proceeding of IEEE Region 10 International Conference on Electrical and Electronic Technology, Vol. 1, pp. 447-452, 2001.

[10] J. H. Kim, K. C. Kim, D. H. Kim, Y. J. Kim and P. Vadakkepat, **"Path Planning and Role Selection Mechanism for Soccer Robot"**, IEEE International Conference on Robotics and Automation, Vol. 4, pp. 3216-3221, 1998.

[11] H. Van Brussel, R. Moreas, A. Zaatri and M. Nuttin, **"A Behaviour-based Blackboard Architecture for Mobile Robots"**, Proceeding of the 24th Annual Conference of the IEEE Industrial Electronics Society, Vol. 4, pp. 2162-2167, 1998.

[12] H. J. Min, K. J. Kim and S. B. Cho, **"Generating Optimal Behaviors of Mobile Robot Using Behavior Network with Planning Capability"**, IEEE International Symposium on Computational Intelligence in Robotics and Automation, Vol. 1, pp. 186-191, 2003.

[13] K. Watanabe, **"Control of an Omnidirectional Mobile Robot"**, Second International Conference on Knowledge-Based Intelligent Electronics Systems, Vol. 1, pp. 51-60, 1998.

[14] F. Solc and B. Honzik, **"Modeling and Control of a Soccer Robot"**, 7th International Workshop on Advanced Motion Control, pp. 506-509, 2002.

[15] T. Takagi and M. Sugeno, **"Fuzzy Indentification of Systems and its Applications to Modeling and Control"**, IEEE Transaction on System, Man and Cybernetics, Vol. 15, pp. 116-132, 1985.

ภาคผนวก ก
ผลงานวิจัยที่ได้รับการตีพิมพ์

1. N. Sanornoi, B. Phurahong and P. Sooraksa. **"Multi-agent System based on Blackboard System for Soccer Robot Implementation"**, International Conference on Control, Automation and System (ICCAS 2004), pp. 2023-2028, Bangkok, Thailand, August 25-27, 2004.

ICCAS '04
ICCAS 2004
2004 International Conference on Control, Automation and Systems
August 25-27, 2004
The Shangri-La Hotel, Bangkok, Thailand
ICASE
http://www.kmitl.ac.th
http://www.iccas.org
Welcome Message
Conference Organization
Conference Information
Sponsors
Table of Contents
Author Index
Search This CD-ROM
Exit
SICE
CAA
AI
ECTI

ICCAS2004 August 25-27, The Shangri-La Hotel, Bangkok, THAILAND

Multi-agent System based on Blackboard System for Soccer Robot Implementation

Nitiwat Sanornoi, Boonchana Phurahong, and Pitikhate Sooraksa

Info-dynamic Laboratory, Department of Information Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

Abstract: This paper reveals the utilization of the multi–agent system that based on the Blackboard system basis as the controller of Soccer Robot. This system is a portion of developing the Soccer Robot team for Robocup 2004 Competition. In this development, the intelligent control system was initiated by the combination of parallel and distributed blackboard structures with the principle design that generated from human body structures, which consists of the combination of two main systems, the organs system and the brain system. The system is designed using the control system theory based on Blackboard basis. Modification of the initial structures to corroborate the Soccer Robot and the structure's constituents are clarified accordingly. To demonstrate the idea, ITE–Gold team is given as a case study.

Keywords: Multi-agent system, Blackboard system, Soccer Robot, Artificial Intelligent

1. INTRODUCTION

A tangible examination for the Robot could be done through the competition that was stipulated with the circumstance to be capable to compete in the competition. At the present, Robot Soccer competition is the most reliable examination field. [17] We would be able to utilize the competition regulations to stipulate the circumstance that appropriately work with the modern technologies. It is adjustable according to a more complicate system depending on the time and new technologies. Obviously the highest accomplishment is the efficiencies in competing with the correspondence to human regulations, which is to develop a robot that would be able to play soccer just like man. Development procedures must be related to the regulations that must be adjusted annually in order to enhance new developments. Competitor's difficulties are occurred with new regulations became the systems needed to be modified accordingly. As a result solutions for the initial regulations could be ineffective to the new regulations, algorithms adaptations, and hardware. The main objective for developers is to continuously develop the robot, but these consequences would be useful if they could access the source of future development without any destruction of the developed systems.

Innovation of multi–agent system on the Blackboard system basis is essential for robot developers, as the system does not only represent the new trend for a more complicated problems, it is also the most effective controlling system. It could satisfactory response to needs of robot developers. In this paper we would like to present architecture on Blackboard system basis for Soccer Robot. As the theme of this development could be related to a general robot development and it is suitable for our architecture examination. Part 2 show the problems and needs for Soccer Robot development covered different circumstances. Part 3 describes the multi–agent system on Blackboard basis. Part 4 proposes a suitable architecture. Part 5, the case study of ITE–Gold team in Thailand's Small League Competition is given as a case study. The final part is the conclusion and the suggestions for future development.

2. DEVELOPMENT EVALUATION

Comparison and the consequences of the multi-agents structure that based on the Blackboard basis shown in [16] have proven that this structure could activate the basic needs such as intelligent behavior, the utilization of various sensor combinations, and the support in destroying the target object, system robustness and system reliability. Using the distributed structures could solve the default of this structure that is non-reactivity to the environment.

2.1 Distributed Development Demands

In system development, various man forces are needed in constructing the Soccer Robot, the responsibilities are then divided into different responsibility areas. The structure that have been mention is capable and could easily be implemented to each part as the structure is logically divided into small parts, helped the development to be carried out individually. Not only that, the error for each particular part could be easily detected. From this consequence, it helps to reduce basic development procedures and time, as well as debug process.

2.2 System Evaluation Demands

A good system should provide a self-learning ability for development, likewise for this Soccer Robot. The structure could support the needs, by implementing algorithm which is used to develop the Evolutionary Agent in learning and it is comparable to a professional that has the self-learning ability. With the increased capabilities, there must be no effect on other parts nor should further adaptation on software.

2.3 System Adaptation Demands

Adaptation is a procedure that could not avoid in development. Structures must be supportive to adapt with the least effects. In this Soccer Robot development, algorithm adaptations in agents are needed to gain the highest efficiencies. Therefore, individual adaptation would effect on particular parts while remaining the rest constantly.

2.4 System Benchmarking Demands

Measurement on efficiency essential for each developer in adaptation. Efficiencies are tested parallel to the development which be could help due the adaptation to have more standard and could be able to compare with the original system.

3. BLACKBOARD SYSTEM THEORY

Blackboard system was originally known as one of the Expert System that was originated from Hearsay II system [1] used to solve the speech understanding problems. The most interesting point in this article is not the utilization of logarithm in problems solving but instead it is the architecture

that is drawn the attention. The utilization was then developed and applied continuously. It was then that this structure is being developed and adapts to meet certain requirement.

This structure is comparable to various specialist professionals that are gathered together to help solve problems with just the basic functions on the blackboard system basis. Problems are individually divided as to be easily solved by particular requirements which based only on the data in the Blackboard system as the raw materials in solving the problems. After finding the solutions, it will be written back to the blackboard. This procedure will be occur continuously until the problems are being answered. This could be easily explained by the example [1] which is the problem in Jigsaw puzzle.

3.1 Original Blackboard System Structure

The original objective of the Blackboard system was to solve the complicated problems [1]. By which the Blackboard will be evaluated from the raw data into data as the answer for that particular questions. It works alphabetical order and the structure as shown in Figure 1.

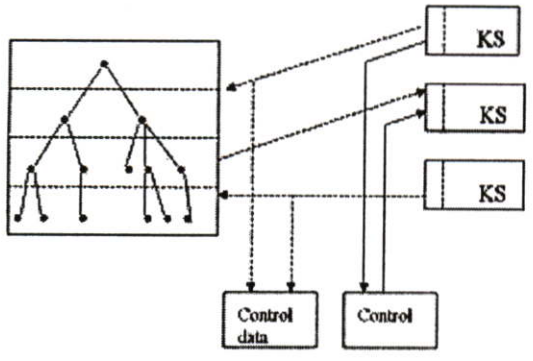


Fig. 1 Classical Blackboard System.

From the Fig. 1, the data will be kept in the Blackboard data structure, which is made up of various layers together (constructed form various layers). This Blackboard data structure will keep the data that are evaluated from the Knowledge source, and the data could be functioned as input data, partial solution, and alternatives or as final solution. Evaluation usually directs from downward to upward and the final solution will be in the top layer. In each layer, the relation of data will be connected as "in-support-of" relation. And in between the same layer, data will be related as "next–to" or "follows" relation.

Knowledge source(KS) will be function as the professional solve specific problems. The objective of KS is to gather all the data in the interested area to evaluate or solve the specific problems in its responsibilities. The data will then be transferred to be kept in the Blackboard data structure. Transferring of data could be taken place in the same or different layers. Therefore, either procedure will be indirectly passed through the Blackboard data structure.

Control Module could be comparable to the controller or to make arrangements for KS by monitoring data in the Blackboard data structure as a command to choose which KS should be in the process or which should not be. Therefore the control module will be the component that commands the solution to the different objectives. The control data is the separated data in the Blackboard data structure for which to support the more convenient in understanding.

Blackboard system could solve the complicated problems by evaluating the data into the solution to the problems, in which the control module will send a command from the KS to evaluate the objective data. This could be easily explained by the above mentioned Jigsaw puzzle problems.

For the new Blackboard system structure will be shown in the extended original structure research. The system will consists of various agents working independently parallel together without control module. This system is called parallel blackboard system [11]. Beside, there might be a system consisted of different Blackboard system combining together by communicating between systems, which is called distributed blackboard structure [10].

3.2 Control Modification of Blackboard System Structure

In the present controlling process, the Blackboard system needs more component than a classical system. Hence, complications takes place and turns the simple control system to a higher qualified system to control, multi–agent control system is then developed as a combination of various sub-systems together. Systems that are autonomous could not avoid the needs of Intelligence in it. Thus, the adaptation of the Blackboard system in process control is implemented to fulfill the requirement [6]. This is where the original objective has been changed form the support to solve the complicated problems.

As for a classical blackboard system, let us we name the layers by the objective of use in each layer accordingly. The layer will be adjusted according to the relations of data allowed the data flows from downstream to upstream (from the bottom to the top) that means the top data will be used. In another words, data will be arranged according to the target's solutions. From this, the objective in evaluation is to evaluate the raw data or the input and convert into the answers to the problems by the agents. By considering the original structure, one can see that the top layer is the answer or the solutions and the bottom layer will be the raw data input. And the middle layers are the answer to the problems that use the lower layer data in solving. At the same time, the data will then be the data to support the solution for the following layer in finding solutions.

When the objectives changed, the sequences of the procedures will be definitely changed as well. For the control process, the data needed are the data that will be fed to the input of the actuator. The data will then be evaluated according to the objectives of each control system. Layers and relations for new structures will be arranged differently from the original structure. The process will be evaluating the target data to input data, the data which is in the bottom layer and the data will flow from the top to the bottom.

4. SOFTWARE ARCHITECTURE FOR SOCCER ROBOT

We got the idea on implementing a robot system that based on the Blackboard system from [2,3] by developing the attribute of both the Distributed Blackboard and the Parallel Blackboard combed together. Each blackboard will communicate in order to exchange data and evaluating the result in distributed. Not only that, agents in each system will work independently from each other. From the structured analysis presented in [6], system consists of three main parts which are the Abstract Environment, Expert System and Low-level control system connected to each other. This could be easily understood by comparing with human beings that consisted of three main parts that are the environments for forming of data, brain part which for

decision making part and the organs for conducting its functions.

Modification upon the original structure could be done by clearly indicated the name of each Layer and specified the type of agents appropriated for the Soccer Robot development (developing the Soccer Robot). The structure as shown in Figure 2 a system represents a robot.

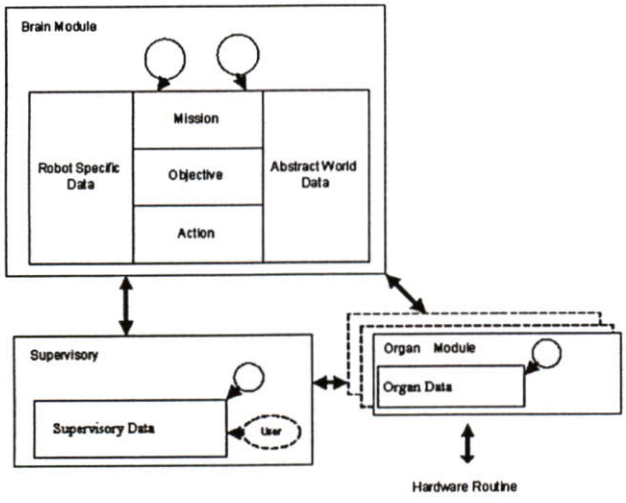


Fig. 2 The presented structure

4.1 Brain Module

A central component in decision making for Soccer Robot could be called as the brain of the system. It is responsible for transferring the environmental data to be evaluated according to the specified objectives into data for controlling the lower part. It is consisted of Mission Layer, Objective Layer, Action Layer, Abstract Layer and Robot Specific Layer. The structure is slightly different from the original in the horizontal layers distribution, which includes the relation and the exchange between the related layers.

Abstract World layer is a layer that has a connection with every other layer. The data in this layer is the environmental data converted into data that are ready for the evaluation. The data is received from the Sensor Organ Module, which is then transformed into environmental data. Robot Specific Layer is another Layer that has the similar connection with the Abstract World Layer. It is a layer that contains the specific data necessary for the system in the evaluation procedures Mission Layer is the top layer in the data flow. It is a place where command mission for Soccer Robot is kept. The data is transferred through the Supervisory Module and is evaluated a mission by Abstract World Layer.

The Objective Layer is the data central layer, and is a place where data evaluated by the agents utilized data in Mission Layer. These data will be transmitting from the mission data to the targeted data. It is digested before being evaluated by agents to Action Layer accordingly. Action Layer is the bottom layer. Data in this layer will be digested in order to be able to carry on to Organ Module immediately. The main data will be formed the evaluated data by the agents in Objective module.

4.2 Organ Module

A module that connected to hardware of the Soccer Robot by imitating human organ for in exchanging data between itself and the brain. These organs could be separated into two types; which are the Sensor Organ Module and Actuate Organ Module.

Sensor Organ Module is responsible for gathering the actual data from the environment and is evaluated into data that could be used in further evaluation. This procedure is the exchange of data between Supervisory Module and the Brain Module, comparable to the Soccer Robot senses.

The agents in this Module will be function in evaluation data from Sensor Hardware and communicate with Hardware by mean of one way communication which is received data from the Hardware only.

Actuate Organ Module receives data from Action Layer of the Brain Module for physical action, as an organ that response physically to the environment. Agents in this Module are the actuator control hardware. From this point, it has been proven that the entire algorithm used in controlling could be modified into this module. The communication with the hardware has be two ways communications due to the necessary use of data from the sensor in controlling process.

We may conclude that the specified types of Actuate Organ Module and Sensor Module do not just depending on the types of communication between the Module and the sensor hardware, but it also concerns in the types of exchanging data between Brain Module and itself.

4.3 Supervisory Module

This Module is a module to communicate with human in order to manage all the procedure in Soccer Robot, by using the User Interface as a transmitter. Communication in exchanging data from Brain Module and Organ Module of Soccer Robot to the understanding and the commanding directly. Agents in this module will be responsible in gathering data from modules transformed into human understandable data (data that could be understood by man) and also receiving the command from user through User Interface back to Soccer Robot.

4.4 Specification of Agent

Agents are responsible for evaluating data according to the assigned function. The system works accordingly only when the agents coordinate accurately. Soccer Robot is involved in many other types of researches and it consists of the combinations of various types of technologies. Involved researches are able to implement into agents by software structures. Agents could be separated into the following types.

Normalize Agent is the agent that evaluate, modified data into an appropriate and correct system pattern as a preparation for other agents in further evaluation without any mistakes or errors occurred. This agent could not be specified into any types of module. It is responsible in preventing any errors that might be occur in evaluation due to the Overflow data. It might also be responsible for data preparation in the following process. But for some other agents, a more complicated function is assigned for transmitting the Algorithm, such as Obstacle Avoidance [20] the waypoint., speed additional in agents to appropriate according to the types of battery, passed to wheel Organ Module.

Strategic agent is responsible for analyzing data from Data Structure into Mission that the system is needed to accomplish. Data to be analysis is the raw data input that could be found generally in the each layer as to conclude in concluding the next procedures which should be in process done by indicted a clear linguistic.

Mission–to-Objective agents could be found in the Brain module, are responsible for evaluating the data as mission data in the Mission Layer into target data and pass to Objective Layer. Mission data is in linguistic which is difficult to

translate data into lower command immediately. Modification is needed in order to an evaluate data. In a Mission, there are various subtract goals. Therefore in accomplishing the mission, we need to have sub target which are the components. For example, Mission in testing the battle / field and it will contain the target data, which are the Waypoints. Mission testing in the field will follow the Waypoint according to Algorithm that will be use as a planning [18,19,21,7] and behavior [15,5] could be able to implement all into this type of agents.

Objective–to–action is an agent responsible for translating the mission data to Objective Layer of the Brain Module into action data in Action Layer which was continuously evaluate from Mission-to-object agent. The evaluation done by this agent is similar to Mission–to–Objective agent, but will be different only with the types of data that will be from the different layer. Therefore, translating into mission data need to transmit data from Waypoint to the wheel speed angle passed through Actuate Organ Module that controls the Soccer Robot wheels. For example, leading the Robot [9] implementing Actuate Organ Module needed to be transmitted into this type of agents.

Data acquire-dispatch is the agent that responsible for receiving and transporting data or both between Blackboards. As for the initial structure transferring data between layers and Blackboard data structure will need agent to be the transmitter. Evaluation will be done by agent in the Blackboard will be the automatically data transformation but t the system structure that consisted of various Blackboard, the transferring of data could be both evaluated and not. Physically, in each Blackboard, might need an indirect transmitter such as LAN, RS-232, etc. Thus these agents are responsible in coordinating with each module to be able to communicate with one another.

5. A CASE STUDY

ITE–Gold has started developing the Soccer Robot for the Robocup Thailand competition in 2003. After the competition, the team was awarded with third place. System used in the competition had the vision part was burried to the Artificial Intelligent and supported only in controlling the "Puthon Kicker " Model, which was the two wheels wheel chair robot using only stepping motor without any other special components.

After the league season, information was gather among the members in our team in order to accomplish the goal for the year 2004, a creation of robot that is capable to challenge with the nationwide competitors according to the national standard regulations is reassigned. Even we encountered with the limitation of only a year period of development, six man forces and a slight amount of budgets, we came to the conclusion of the system specification as follows:

- System should a Distributed System in order to support the development, AI and vision as the parallel structure development.
- System should able to support the control of two wheels or wheel chair robot[8], three wheel Omni-directional[9], and other system in the future [12,13,14].
- Able to support both robots with special equipment (Shooter, Dribbler, etc.), with some special equipment or with non special equipment.
- Able to apply with different models in the same competition.
- AI system must be fixed to only one computer in evaluation by combining the GUI together.

We have used Physical system shown in Figure 4 as to save the budget and to be able to separate the working parts.

After the 2004 League season, we were awarded with third place and received the best technique award. Various types of Robots as shown in Table 1 can be controlled by our system.

From the targeted specification, we have to design the structure that is harmonious with the limited factors. In the regulations, each team is allowed to submit five robots all at once, and it is the reason why we have to separate the Brain Module into five individual modules. There were two Organic modules that are used to communicate with the five robots and the Vision evaluates components. There is only a Supervisory Module that coordinates with all the Brain Modules, structures as shown in Figure 4.

Table 1 ITE-Gold robot model.

Model	Mechanism	Special part	Driving Motor
Phuthon Kicker	Wheel chair	-	Stepping
Big Foot	Wheel chair	Spinning Kicker	DC
Anurak	Wheel chair	Dribbler, Kicker	DC
Goblin	3 Wheel omni-directional	-	Servo
Freedom	3 Wheel omni-directional	Dribbler, Kicker	DC

5.1 Brain Module

We intended to design each robot to be intellectually independence to each other. Therefore the Brain Module is equivalent to the numbers of robots that are in the competition. Independence of this component helps to utilize each module all at the same time. However, even with the independence module, data remains the same.

The Robot Specific Layer keeps only the necessary data for the Robot, which is also necessary for the agent in evolution. The data will be assigned from Supervisory Module in the starting procedure and it could be corrected by the team manager in rearranging the plan according to the situations.

Abstract Word Layer is the data received from Vision Organ Module and is also kept the necessary data factors of the field competition which sets the data/ value from Supervisory Module through the Vision acquired agent. This agent will transmit the raw data from Vision Module into number data and transfer these data to be kept in the Abstract World Layer. Positioning data will also inform us the status of the object such as if the data is shown as remain, it means the robot still remain in the competition field. But if it shows the disappeared, it means that the robot is no longer in the field. However, unavoidable error still be found due to the incorrect data, for example if the target (ball) was blocked by the robot due to the angle of the camera, the data will be shown as disappeared, even with the fact that the target object is still there. Fortunately these errors are solved by the World's Data normalization Agent.

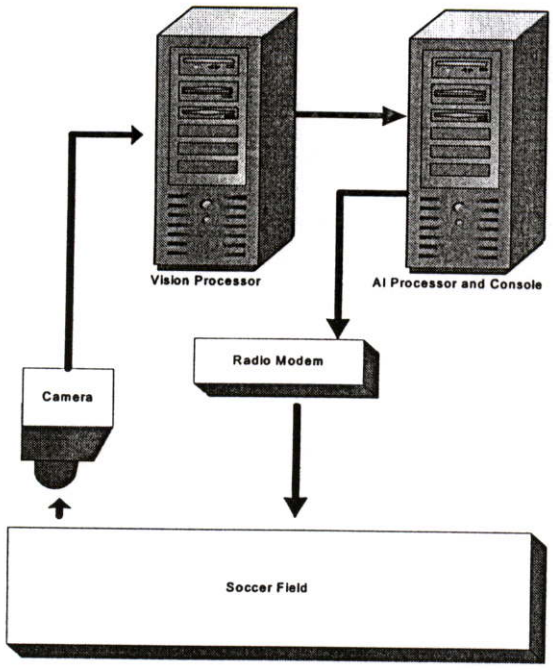


Fig.3 Physical System that was limited

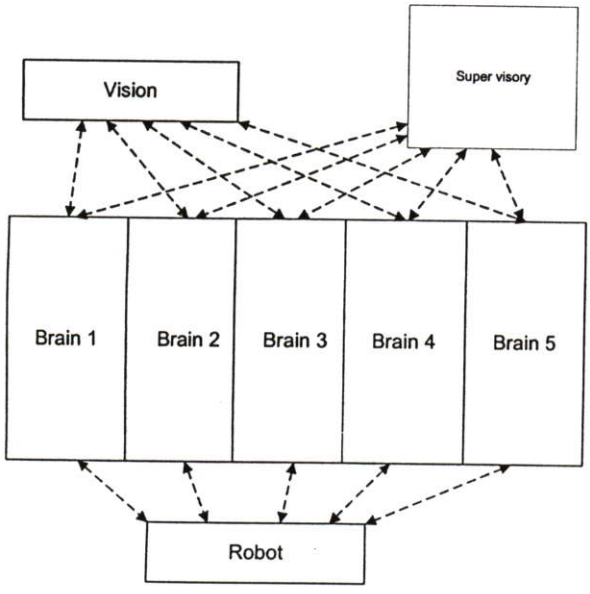


Fig. 4 Software Structures that ITE–Gold team used in Robocup Thailand 2004

Linguistic-mission data is made up of data from Mission Generate agent by evaluating data from both Robot Specific Layer and Abstract World Layer that are suitable for each circumstance. These data will be taken as the events that took place and the Behavior agents will convert the orders to data in Objective Layer which consisted of waypoint and the robot's actions.

In avoiding from the barriers or the defenders in the competition, the new Waypoint is needed to be reconstructed / rebuilt on the original one by Obstacle Avoidance Agent. But if there is no crushing the barriers in the analysis, data changes will not be necessary. Therefore, with the Real–time working system, the robot will automatically avoid the barriers.

The group of Navigation Agent will function as data converter at the Waypoint by using Algorithm [8] for two wheels wheelchair module and [4] for three wheel Omni–direction module. It converts the data into the wheel speed level and will be recorded in Action layer. Types of Waypoint will stipulate the speed of running. And Action layer converts the event data to control data for each organ. And the last procedure is that the Robot Dispatch Agent converts the data to a format that will be transfer to Robot Organic module. From this experiment, we would be able to extract the Linguistic data into a controllable data during the last procedure for the Real–time working system.

5.2 Organic Module

Vision Module is functioned as an eye of the Robot by the Com–port acquisition agent will be gathering the data of the Real–time from the Evaluation part via RS-232.

Robot's organic module will be sharing likewise to the Vision Module and its function is to gather the information from Action Layer, which is the ready data to be passing through to the robot. All data will be passed through the com-port by the wireless system through the Radio modem agent. Data will pass the String data. As for the freeze agent is responsible for correcting all command data to a stop. When the status of Robot Specific Layer is off, the robot will then evaluate the command and act accordingly. It could be noticed by the robot in the competition would function as an organ, which is the Actuator that responses to the environment in the reality.

5.3 Supervisory Module

Supervisory Module is a component that is used to communicate with the Operator. Its function is to gather all the data in other module in terms of GUI (in GUI form) which could be understood by the Operator as shown in Figure 6. Not only that, another function is to be the central transmitter in order and correct the data in Data Structure in other Modules. For example, in the situation that the Operator needs to change the plan while the game is stopped by concentrating on the forwarding. There will be a changes in some of the robot position from the defender to the forwarder that would also need to change the position data in Robot Specific Layer of the Brain Module by passing through the GUI of this Module.

6. CONCLUSION

This article shows the excellent implementation capabilities of the Multi–Agents system structure based on Blackboard System with the Soccer Robot. The advantage of this system can be measured by the results of applications. The outstanding part is the capability to control the robot consisted of different Model together at the same time within the competition. This helps reduction of development period from 5-6 years down to only 2 years.

We are definitely unable to avoid the system developments, due to the change of regulations and new leagues that will be organized. These developments must be consist to each other, which they should be more complicated developments with the Agent and Data structure, but the overall structure should remain the same. From the implementation in this research, the leading system will mainly base on the waypoint which in terms of adaptation into others ways such as Vector Field [21] could be done by just changing parts of the Agent and Data Structure.

Regulations for Robocup Thailand Competition were not much different from the previous competition. Therefore, our ITE Gold team's development approaches were just firstly modified the original into a four wheels Omni–directional robot. Increased the shooting forces, the fetch and throw abilities, vision development were done to support the utility of various cameras as to solve the poor vision problems. These developments were unavoidable effected on the determination part. Agent that was responsible for leading the four wheels Omni–directional model robot was added and other agents were adapted to suite the new model robot. From this point, the new proposed structure shows the excellent support on the developments and we do believe that this structure could definitely support the future challenges.

REFERENCES

[1] R. Engelmore, and T. Morgan, *Blackboard Systems*, Addison-Wesley, 1988.

[2] G. K. H. Pang, "Development of a blackboard system for robot programming", *Proceedings of the third international conference on Industrial and engineering applications of artificial intelligence and expert systems*, Vol. 1, pp. 123-130, 1990.

[3] A. Koeing and E. Crochon,"Tram: a blackboard architecture for autonomous robots", *Proceedings of the first international conference on Industrial and engineering applications of artificial intelligence and expert systems*, Vol. 1, pp. 590-597, 1988.

[4] Y. Liu, X. W, J. Zhu, and J. Lew, "Omni-directional mobile robot controller design by trajectory linearization", *Proceedings of the American Control Conference*, Vol. 4, pp. 3423-3428, 2003.

[5] H. Van Brussel,R. Moreas,A. Zaatri, and M. Nuttin,"A behaviour-based blackboard architecture for mobile robots", *Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society*, Vol. 4, pp. 2162-2167, 1998.

[6] M. Occello and M. C. Thomas, "A new approach for process control", *Proceedings of Computers in Design Manufacturing, and Production*, pp. 487-492, 1993.

[7] K. Y. Chen and A. Liu, "A design method for incorporating multidisciplinary requirements for developing a robot soccer player", *Proceedings of Fourth International Symposium on Multimedia Software Engineering*, pp. 25-32, 2002.

[8] F. Solc and B. Honzik , "Modeling and control of a soccer robot", *7th International Workshop on Advanced Motion Control*, pp. 506-509, 2002.

[9] R. W. Beard,"Robot soccer: an ideal senior design experience", *Proceedings of the American Control Conference*, Vol. 6, pp. 3975-3979, 2000.

[10] M. Occello and M. C. Thomas,"A distributed blackboards methodology for designing robotic control softwares", *IEEE International Conference on Systems Engineering*, pp. 147-150, 1992.

[11] M. Occello and Y. Demazeau ,"Building real time agents using parallel blackboards and its use for mobile robotics", *IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 2, pp. 1610-1615, 1994.

[12] K. Watanabe, "Control of an omnidirectional mobile robot", *Second International Conference on Knowledge-Based Intelligent Electronic Systems*, Vol. 1, pp. 51-60, 1998.

[13] T. B. Park, J. H. Lee, B. J. Yi, W. K. Kim, B. J. You, and S. R. Oh, "Optimal design and actuator sizing of redundantly actuated omni-directional mobile robots", *IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 732-737, 2002.

[14] K. L. Moore, M. Davidson, V. Bahl, S. Rich, and S. Jirgal, "Modelling and control of a six-wheeled autonomous robot", *Proceedings of the American Control Conference*, Vol. 3, pp. 1483-1490, 2000.

[15] H. H. Lund and L. Pagliarini ,"RoboCup Jr. with LEGO MINDSTORMS", *IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 813-819, 2000.

[16] R. E. Fayek, R. Liscano, and G. M. Karam,"A system architecture for a mobile robot based on activities and a blackboard control unit", *IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 267-274. 1993

[17] H. Kitano, M. Asada, I. Noda, and H. Matsubara ,"RoboCup: robot world cup", *IEEE Robotics & Automation Magazine*, Vol. 5, Issue. 3, pp. 30-36, 1998.

[18] T.-H.S. Li, C. A. Lai, and Y. J. Guo, "Design of fuzzy field control for a one-on-one robot soccer system", *IEEE 28th Annual Conference of the Industrial Electronics Society*, Vol. 4, pp. 2605-2610, 2002.

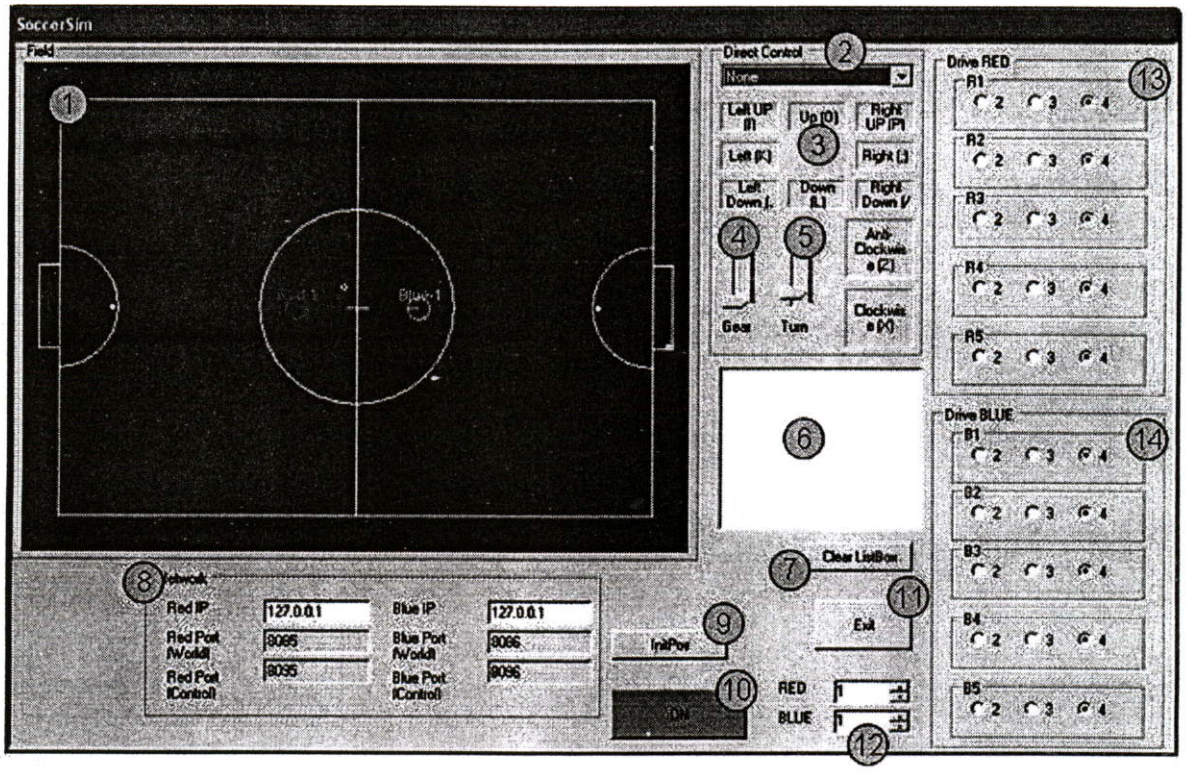
[19] T.-H.S. R. C. Liu, and I-F. Lin, "Fuzzy shooting control of car-like soccer robot", *Proceedings of IEEE Region 10 International Conference on Electrical and Electronic Technology*, Vol. 1, pp. 447-452, 2001.

[20] L. Wu, S. Zhao, and X. Xu, "Design of an obstacle-avoidance strategy based on robot soccer competition", *5th International Conference on Signal Processing*, Vol. 3, pp. 1722-1725, 2000.

[21] J. H. Kim, K. C. Kim, D. H. Kim., Y. J. Kim ,and P. Vadakkepat , "Path planning and role selection mechanism for soccer robot", *IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 3216-3221, 1998.

ภาคผนวก ข

วิธีการใช้งานโปรแกรมจำลองสถานการณ์หุ่นยนต์เล่นฟุตบอล



1. หน้าจอแสดงการแข่งขัน
2. เลือกหุ่นยนต์ที่การบังคับโดยตรงจากผู้ใช้งาน โดยไม่ผ่านการควบคุมจากปัญญาประดิษฐ์
3. แถบแสดงสถานะปุ่มกดจากคีย์บอร์ด สำหรับการควบคุมหุ่นยนต์โดยตรง
   - ปุ่ม I สั่งให้หุ่นยนต์วิ่งไปทางเฉียงหน้าซ้าย
   - ปุ่ม O สั่งให้หุ่นยนต์วิ่งไปทางด้านหน้า
   - ปุ่ม P สั่งให้หุ่นยนต์วิ่งไปทางเฉียงหน้าขวา
   - ปุ่ม K สั่งให้หุ่นยนต์วิ่งไปทางซ้าย
   - ปุ่ม ; สั่งให้หุ่นยนต์วิ่งไปทางขวา
   - ปุ่ม , สั่งให้หุ่นยนต์วิ่งไปทางเฉียงหลังซ้าย
   - ปุ่ม L สั่งให้หุ่นยนต์วิ่งไปทางด้านหลัง
   - ปุ่ม / สั่งให้หุ่นยนต์วิ่งไปทางเฉียงหลังขวา
   - ปุ่ม Z สั่งให้หุ่นยนต์หมุนตัวทวนเข็มนาฬิกา

- ปุ่ม X สั่งให้หุ่นยนต์หมุนตัวตามเข็มนาฬิกา

4. แถบปรับความเร็วของหุ่นยนต์ในการวิ่ง สำหรับการควบคุมโดยตรง
5. แถบปรับความเร็วของหุ่นยนต์ในการหมุน สำหรับการควบคุมโดยตรง
6. หน้าจอแสดงสถานะของหุ่นยนต์
7. ปุ่มล้างหน้าจอแสดงสถานะของหุ่นยนต์
8. หน้าจอแสดงสถานะของโปรแกรมด้านติดต่อสื่อสาร
9. ปุ่มกำหนดตำแหน่งหุ่นยนต์ ให้กลับไปอยู่ตำแหน่งเริ่มต้นการแข่งขัน
10. ปุ่มเริ่มการแข่งขัน/หยุดการแข่งขัน
11. ปุ่มออกจากโปรแกรม
12. เลือกจำนวนหุ่นยนต์ของทีมที่ใช้ในการแข่งขัน
13. เลือกแบบจำลองการเคลื่อนที่ของหุ่นยนต์เล่นฟุตบอลทีม RED
14. เลือกแบบจำลองการเคลื่อนที่ของหุ่นยนต์เล่นฟุตบอลทีม BLUE

ประวัติผู้เขียน

นายนิธิวัฒน์ เสนาะน้อย เกิดเมื่อวันที่ 18 พฤศจิกายน พ.ศ. 2523 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2545 และเข้าศึกษาต่อในระดับปริญญาโท หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมสารสนเทศ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง