

การบอกตำแหน่งในอาคารด้วยวิธีจำแนกขอบเขตพื้นที่ที่ไม่สมมาตร
สำหรับการนำทางหุ่นยนต์เคลื่อนที่

ASYMMETRY REGIONAL BOUNDARY LOCALIZATION FOR
INDOOR ROBOT NAVIGATION



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2562

KMITL-2019-EN-M-060-130

การบอกตำแหน่งในอาคารด้วยวิธีจำแนกขอบเขตพื้นที่ที่ไม่สมมาตร
สำหรับการนำทางหุ่นยนต์เคลื่อนที่

ASYMMETRY REGIONAL BOUNDARY LOCALIZATION FOR
INDOOR ROBOT NAVIGATION



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2562

KMITL-2019-EN-M-060-130

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ASYMMETRY REGIONAL BOUNDARY LOCALIZATION FOR
INDOOR ROBOT NAVIGATION



A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2019

KMITL-2019-EN-M-060-130

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2019

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การบอกตำแหน่งในอาคารด้วยวิธีจำแนกขอบเขตพื้นที่ที่ไม่
	สมมาตร สำหรับการนำทางหุ่นยนต์เคลื่อนที่
นักศึกษา	นายนรินทร์เดช วรรณะเดชะ
รหัสประจำตัว	58601083
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมการวัดคุม
พ.ศ.	2562
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.พงษ์ชัย นิลาศ

บทคัดย่อ

บทความนี้นำเสนอวิธีการบอกตำแหน่งภายในอาคารสำหรับหุ่นยนต์เคลื่อนที่แบบใหม่ โดยการใช้ความแรงสัญญาณจากเครือข่ายโมดูลชิปซึ่งเป็นโมดูลสื่อสารไร้สาย มาเป็นปัจจัยหลักในการระบุว่าหุ่นยนต์อยู่ ณ ตำแหน่งใดในสนามและใช้เพื่อเป็นการนำทางหุ่นยนต์ไปยังเป้าหมาย เนื่องจากโดยปกติความแรงของสัญญาณนั้นจะแปรเปลี่ยนไปตามคุณลักษณะจำเพาะในแต่ละบริเวณ เช่นความสามารถในการสะท้อนและหักเหของคลื่นภายในพื้นที่ใด ๆ สิ่งกีดขวางทางเดินของคลื่นสัญญาณ รวมไปถึงการรบกวนจากแหล่งกำเนิดสัญญาณอื่น ๆ ทำให้เกิดความไม่แน่นอนของสัญญาณอยู่มาก การบอกตำแหน่งที่แม่นยำเฉพาะเจาะจงจึงทำได้ไม่แม่นยำมากนัก บทความนี้จึงใช้ความสามารถในการซ้ำค่าของอุปกรณ์มาประยุกต์ใช้กับโมดูลชิปไร้สาย จากการเก็บข้อมูลความแรงสัญญาณในสนามทดสอบหลาย ๆ ครั้ง เพื่อนำความสามารถในการซ้ำค่ามาสร้างขอบเขตการทำงานที่น่าเชื่อถือของแต่ละโมดูล เมื่อขอบเขตการทำงานของแต่ละเสาสัญญาณนั้นตัดกัน จึงเกิดเป็นพื้นที่ย่อยหรือรีเจี้ยนขึ้นมาหลาย ๆ พื้นที่ที่มีลักษณะไม่สมมาตรกัน ในโหมดการทำงานจริง เมื่อหุ่นยนต์เคลื่อนที่ไปในสนาม ค่าความแรงสัญญาณปัจจุบันที่ตัวมันอ่านค่าได้จะถูกนำไปวิเคราะห์กับฐานข้อมูลที่ประมวลมาแล้วก่อนหน้านี้ เพื่อจำแนกว่าหุ่นยนต์อยู่ในรีเจี้ยนใดในสนามทดสอบ มีสิ่งกีดขวางใด ๆ ในพื้นที่หรือไม่ รีเจี้ยนใดที่ไม่มีสิ่งกีดขวาง จะถูกใช้ในการสร้างเส้นทางการเดินของหุ่นยนต์เพื่อนำทางไปยังพื้นที่เป้าหมายที่ต้องการ

Thesis	Asymmetry Regional Boundary Localization for Indoor Robot Navigation
Student	Mr. Narindech Wanadecha
Student ID.	58601083
Degree	Master of Engineering
Program	Instrumentation Engineering
Year	2019
Thesis Advisor	Asst. Prof. Dr. Pongchai Nilas

ABSTRACT

This thesis proposes new methodology of indoor localization supported for mobile robot navigation using received signal strength indicator or RSSI from Zigbee Wireless Network, locating the robot position respect to the testbed. According to the difference of signal characteristic in each specific area caused by disturbances such as signal multi path, signal reflection and deviation, or interference from other radio sources, these aspects make the signal strength unreliable which leads to low accuracy in positioning process. Therefore, this thesis applies gage repeatability technique to multiple Zigbee modules scattered in the field as reference node. RSSI of each node have been statistically collected entire testbed in order to form a reliable boundary of working range with acceptable repeatability. Then, the intersection of all the boundaries will become close-shape area so-called "Region". The boundaries are not necessary the same range so, in consequently, the region appears to be asymmetry shape. Lastly, when the robot moves in the testbed and read the current RSSIs, the data will be compared with pre-processed database in order to classify which region the robot located in, including the obstacle existence. All of the unoccupied regions will be gathered and formed as a path for the robot to avoid the obstructed zone and reach the target area.

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ผศ.ดร.พงษ์ชัย นิลาศ ที่ให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้และประสบการณ์ที่ดีแก่ข้าพเจ้า

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้กับบิดามารดา และญาติพี่น้องซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า รวมไปถึงเพื่อน ๆ รุ่นพี่และรุ่นน้องที่ห้องปฏิบัติการที่คอยให้กำลังใจเสมอมา

นรินทร์เดช วรรณเดชะ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญรูป.....	XI
บทที่ 1.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	2
1.3 สมมุติฐานของการศึกษา.....	3
1.4 ขอบเขตการวิจัย.....	3
1.5 ขั้นตอนการศึกษา.....	3
บทที่ 2.....	5
2.1 กล่าวนำ.....	5
2.2 แนวคิดการบอกตำแหน่งภายในอาคารด้วยความแรงสัญญาณ (Indoor Localization with RSSI).....	5
2.2.1 การบอกตำแหน่งและติดตามตำแหน่งของอุปกรณ์เป้าหมาย จำแนกจากปัจจัยที่ใช้วัดระยะทาง.....	6
2.2.2 เทคนิคการบอกตำแหน่งด้วยความแรงสัญญาณ.....	6
2.3 วิธีการระบุตำแหน่งพร้อมกับการสร้างแผนที่ (Simultaneous Localization and Mapping: SLAM).....	17
2.4 การเรียนรู้ของเครื่อง Machine learning.....	20
2.4.1 รูปแบบต่าง ๆ ของการเรียนรู้ของเครื่อง (Approaches to Machine Learning).....	21
2.5 เครือข่ายเซนเซอร์ไร้สาย.....	28
2.6 หุ่นยนต์เคลื่อนที่ และตัวควบคุมไมโครคอนโทรลเลอร์.....	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 3	32
3.1 กล่าวนำ	32
3.2 การอ่านค่าความแรงสัญญาณระหว่างตัวหุ่นยนต์เคลื่อนที่กับเสาสัญญาณอ้างอิงในสนาม	33
3.3 การจัดเตรียมสนามทดสอบ	35
3.4 การเก็บบันทึกชุดความแรงสัญญาณและประมวลผลข้อมูลเบื้องต้น	37
3.5 การคำนวณหาความสามารถในการซ้ำค่าของอุปกรณ์ และการสร้างขอบเขตพื้นที่ที่ไม่สมมาตร	38
3.5.1 การคำนวณหาความสามารถในการซ้ำค่าในแต่ละรัศมีการทำงานของอุปกรณ์สื่อสารไร้สาย (Radius repeatability, R_r)	40
3.5.2 การคำนวณหาค่าเฉลี่ยค่าความสามารถในการซ้ำค่าของอุปกรณ์ (Entire field repeatability, R_f)	41
3.5.3 การกำหนดเทรชโฮลด์ให้กับค่าความสามารถในการซ้ำค่า ณ ระยะเวลาการทำงานของอุปกรณ์ที่เหมาะสมที่สุด (Threshold repeatability, R_T)	42
3.6 การสร้างโครงข่ายประสาทเทียมสำหรับเรียนรู้ชุดความแรงสัญญาณ	43
3.6.1 การเตรียมความพร้อมของข้อมูล	44
3.6.2 การปรับปรุงข้อมูลฝึกสอนด้วยเทคนิคการแก้ไขความไม่สมดุลของคลาสและการเพิ่มสัญญาณรบกวนแบบไวท์เกาส์เซียน	48
3.6.3 การลดความซ้ำซ้อนของข้อมูลหรือ نرمัลไลเซชัน (Normalization)	48
3.6.4 การสร้างและฝึกสอนโมเดลจำแนกข้อมูล	50
3.6.5 การเลือกอัลกอริทึมสำหรับฝึกสอนโมเดล	51
3.6.6 การทดสอบไฮเปอร์พารามิเตอร์ (Hyper-parameter testing)	54
3.6.7 วิธีการทำครอสวาไลเดชัน 10 ชุด (10-fold cross validation)	55
3.6.8 ค่าความผิดพลาดเจเนอรัลไลเซชัน แบบ โอเวอร์ฟิตติ้ง และอันเดอร์ฟิตติ้ง (Generalization error: Over-fitting and Under-fitting)	57
3.6.9 การสร้างเส้นทางการเคลื่อนที่ของหุ่นยนต์ด้วยการเรียนรู้แบบเสริมกำลัง	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 4	64
4.1 หุ่นยนต์เคลื่อนที่และระบบควบคุมสำหรับการบอกตำแหน่งเป้าหมายแบบ Asymmetry Regional Boundary	64
4.1.1 หุ่นยนต์เคลื่อนที่	64
4.1.2 เครือข่ายโมดูลซิกบีในสนามทดสอบ	67
4.1.3 สถานีฐาน (Base station)	67
4.1.4 โอเปอเรเตอร์ (Operator)	68
4.1.5 ฐานข้อมูลคลาวด์ไฟร์เอร์เบส (Firebase cloud database)	70
4.2 หุ่นยนต์เคลื่อนที่และระบบควบคุมสำหรับการบอกตำแหน่งเป้าหมายแบบ SLAM	72
4.2.1 แนวคิดพื้นฐานของเฮกเตอร์สแลม (Hector SLAM)	73
4.2.2 เทอมสำคัญในการสร้างหุ่นยนต์เคลื่อนที่ด้วยรอส (ROS)	75
บทที่ 5	79
5.1 กล่าวนำ	79
5.2 ผลการทดลองการเก็บบันทึกชุดความแรงสัญญาณในสนามทดสอบ	79
5.3 ผลการทดลองการคำนวณค่าความสามารถในการเข้าค่าของอุปกรณ์ และการสร้างพื้นที่ที่ไม่ สมมาตร	86
5.4 ผลการทดสอบประสิทธิภาพของโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรใน สนาม ขณะเรียนรู้ชุดข้อมูลฝึกสอน ทดสอบ และการบอกตำแหน่งด้วยพื้นที่ที่ไม่สมมาตรขณะใช้ งานจริง	89
5.5 ผลการทดสอบการค้นหาเส้นทางของหุ่นยนต์เคลื่อนที่ ด้วยการเรียนรู้แบบเสริมกำลัง	109
5.6 ผลการเปรียบเทียบระหว่างการบอกตำแหน่งด้วยวิธี ARBL (Asymmetry Regional Boundary Localization) กับการเทคนิค SLAM (Simultaneous Localization and Mapping) ที่ทำงานบนระบบปฏิบัติการหุ่นยนต์ ROS (Robot Operating System)	123
5.6.1 โครงสร้างของระบบ	123
5.6.2 ความยืดหยุ่นและความซับซ้อนของวิธีการ	124

สารบัญ (ต่อ)

	หน้า
5.6.3 ประสิทธิภาพในการบอกตำแหน่ง.....	125
5.6.4 ความเหมาะสมต่อการนำไปใช้.....	126
5.6.5 ต้นทุนในการผลิต.....	127
บทที่ 6.....	129
6.1 สรุปผลการวิจัย.....	129
6.2 ข้อเสนอแนะ.....	130
เอกสารอ้างอิง.....	132
ภาคผนวก.....	136
ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่.....	140
ประวัติผู้เขียน.....	147



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 ตารางเปรียบเทียบคุณลักษณะของการบอกตำแหน่งภายในอาคารและกลางแจ้ง	12
ตารางที่ 3.1 ตัวอย่างชุดความแรงสัญญาณจากเสาสัญญาณอ้างอิง ณ พิกัดเก็บข้อมูลที่ n ถึง $n+1$	44
ตารางที่ 3.2 ตัวอย่างการสร้างเลเบลด้วยวิธีวันฮ็อตเอนโค้ดดิ้ง (One-hot encoding) สำหรับฝึกสอนโครงข่ายประสาทเทียม.....	45
ตารางที่ 3.3 ความแรงสัญญาณของเสาสัญญาณอ้างอิงในสนาม ณ พิกัดเก็บข้อมูลที่ n ถึง $n+5$	46
ตารางที่ 3.4 ตัวอย่างการสร้างเลเบลของแต่ละพิกัดข้อมูลสำหรับฝึกสอนโครงข่ายประสาทเทียม...	46
ตารางที่ 3.5 ตารางแสดงรายละเอียดกฎนี้ไว้สำหรับกำหนดโครงสร้างเครือข่ายประสาทเทียม ...	54
ตารางที่ 3.6 การตั้งค่าไฮเปอร์พารามิเตอร์สำหรับสร้างและฝึกสอนโครงข่ายประสาทเทียม.....	55
ตารางที่ 5.1 แสดงค่าความสามารถในการซ้ค่าแบบเทอร์ซโฮล ของสนามทดสอบที่ 1 (คำนวณจากชุดข้อมูล 10 ชุด)	86
ตารางที่ 5.2 แสดงค่าความสามารถในการซ้ค่าแบบเทอร์ซโฮล ของสนามทดสอบที่ 2 (คำนวณจากชุดข้อมูล 10 ชุด)	86
ตารางที่ 5.3 คอนฟิวชันเมตริกซ์ของผลจากการฝึกสอนโครงข่ายประสาทเทียมแสดงในรูปแบบจำนวนข้อมูล สนามทดสอบที่ 1 ขนาด 10 เมตร x 10 เมตร.....	99
ตารางที่ 5.4 คอนฟิวชันเมตริกซ์ของผลการฝึกสอนโครงข่ายประสาทเทียมแสดงในรูปแบบร้อยละสนามทดสอบที่ 1 ขนาด 10 เมตร x 10 เมตร	100
ตารางที่ 5.5 คอนฟิวชันเมตริกซ์ของผลจากการทดสอบโครงข่ายประสาทเทียมในรูปแบบจำนวนข้อมูล สนามทดสอบที่ 1 ขนาด 10 เมตร x 10 เมตร.....	101
ตารางที่ 5.6 คอนฟิวชันเมตริกซ์ของผลการทดสอบโครงข่ายประสาทเทียมแบบร้อยละ สนามทดสอบที่ 1 ขนาด 10 เมตร x 10 เมตร.....	102
ตารางที่ 5.7 การตั้งค่าไฮเปอร์พารามิเตอร์ของโครงข่ายประสาทเทียมที่ให้ผลการจำแนกที่ดีที่สุดโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนามทดสอบที่ 2.....	103
ตารางที่ 5.8 จำนวนพิกัดข้อมูลที่ใช้ฝึกสอนในแต่ละคลาสหรือพื้นที่ที่ไม่สมมาตรของโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตร สนามทดสอบที่ 2	104
ตารางที่ 5.9 คอนฟิวชันเมตริกซ์ของผลจากการฝึกสอนโครงข่ายประสาทเทียมในรูปแบบจำนวนข้อมูล สนามทดสอบที่ 2 ขนาด 6 เมตร x 10 เมตร.....	105
ตารางที่ 5.10 คอนฟิวชันเมตริกซ์ของผลการฝึกสอนโครงข่ายประสาทเทียมแบบร้อยละ สนามทดสอบที่ 2 ขนาด 6 เมตร x 10 เมตร.....	106
ตารางที่ 5.11 คอนฟิวชันเมตริกซ์ของผลจากการทดสอบโครงข่ายประสาทเทียมในรูปแบบจำนวนข้อมูล สนามทดสอบที่ 2 ขนาด 6 เมตร x 10 เมตร.....	107

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

หน้า

ตารางที่ 5.12 คอนพิวเตอร์ของผลการทดสอบโครงข่ายประสาทเทียมแบบร้อยละ สนามทดสอบที่ 2 ขนาด 6 เมตร x 10 เมตร.....	108
ตารางที่ 5.13 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป.....	112
ตารางที่ 5.14 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป.....	112
ตารางที่ 5.15 แสดงนโยบายที่ดีที่สุดในการแก้ปัญหา (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป.....	113
ตารางที่ 5.16 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล.....	113
ตารางที่ 5.17 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล.....	114
ตารางที่ 5.18 แสดงนโยบายในการแก้ปัญหาที่ดีที่สุด (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล.....	114
ตารางที่ 5.19 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป.....	116
ตารางที่ 5.20 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป.....	117
ตารางที่ 5.21 แสดงนโยบายในการแก้ปัญหาที่ดีที่สุด (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป.....	117
ตารางที่ 5.22 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล.....	118
ตารางที่ 5.23 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล.....	118
ตารางที่ 5.24 แสดงนโยบายในการแก้ปัญหาที่ดีที่สุด (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล.....	119

สารบัญตาราง (ต่อ)

หน้า

ตารางที่ 5.25 ตารางเปรียบเทียบโครงสร้างของระบบการบอกตำแหน่งหุ่นยนต์ภายในอาคารด้วยวิธี เออาร์บีแอลและวิธีสแลมบนระบบปฏิบัติการรอส.....	123
---	-----



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 การบอกตำแหน่งเป้าหมายด้วยวิธีไตรแองกูลേഷัน (Triangulation) แบบ แบบศูนย์กลางการประมวลผลอยู่กับเครื่องเซิร์ฟเวอร์(ซ้าย) และแบบศูนย์กลางการประมวลผล อยู่กับโหนดเป้าหมาย (ขวา).....	8
รูปที่ 2.2 ภาพแสดงแนวคิดการบอกตำแหน่งแบบ เดอลอเนย์ไตรแองกูลേഷัน (Delaunay Triangulation).....	9
รูปที่ 2.3 ภาพแสดงแนวคิดการบอกตำแหน่งแบบไตรแลทเตอเรชัน (Trilateration).....	10
รูปที่ 2.4 แสดงการบอกตำแหน่งภายในอาคารด้วยเทคนิคไม่อ้างอิงระยะด้วยการรู้จำลายนิ้วมือของ ข้อมูล (Fingerprinting technique).....	13
รูปที่ 2.5 การบอกตำแหน่งเป้าหมายด้วยวิธีการประมาณตำแหน่งจุดในพื้นที่สามเหลี่ยม.....	14
รูปที่ 2.6 กรณีตัวอย่างในการบอกตำแหน่งเป้าหมายด้วยวิธีการประมาณตำแหน่งจุดในพื้นที่ สามเหลี่ยม.....	15
รูปที่ 2.7 แผนผังแสดงประเภทการทำงานรูปแบบต่าง ๆ ที่เป็นส่วนประกอบหลักของแนวคิดแมชชีน เลินนิ่งหรือการเรียนรู้ของเครื่อง.....	20
รูปที่ 2.8 แผนผังแสดงกระบวนการทำงานของวิธีการเรียนรู้แบบเสริมกำลังด้วยเทคนิคคิวเลินนิ่ง...	23
รูปที่ 2.9 ไดอะแกรมแสดงการทำงานของความโลภเอปซิลอน (Epsilon-Greedy).....	27
รูปที่ 2.10 แสดงเครือข่ายเซนเซอร์ไร้สายประกอบไปด้วยโมดูลชิกปีแบบเมฆเน็ตเวิร์ก.....	29
รูปที่ 3.1 ไดอะแกรมแสดงส่วนประกอบทั้งหมดภายในระบบ.....	33
รูปที่ 3.2 กราฟแสดงตำแหน่งการวางเสาสัญญาณอ้างอิงในสนามทดสอบ.....	34
รูปที่ 3.3 โพล์ชาร์ทแสดงกระบวนการรับส่งข้อมูลระหว่างโมดูลชิกปีในเครือข่าย.....	34
รูปที่ 3.4 แสดงเฟรมข้อมูลสำหรับสื่อสารกันภายในเครือข่ายชิกปีทั้งเฟรมประเภทส่งและรับ.....	35
รูปที่ 3.5 สนามทดสอบที่ 1 ขนาดพื้นที่ 100 ตารางเมตร (กว้าง 10 เมตร ยาว 10 เมตร).....	35
รูปที่ 3.6 สนามทดสอบที่ 2 ขนาดพื้นที่ 60 ตารางเมตร (กว้าง 6 เมตร ยาว 10 เมตร).....	36
รูปที่ 3.7 แผนผังสนามทดสอบที่ 1 แสดงพิกัดเก็บข้อมูลในสนาม ตำแหน่งที่ตั้งของเสาสัญญาณ อ้างอิงและเส้นกำหนดการเคลื่อนที่ของหุ่นยนต์.....	37
รูปที่ 3.8 แผนผังสนามทดสอบที่ 2 แสดงพิกัดเก็บข้อมูลในสนาม ตำแหน่งที่ตั้งของเสาสัญญาณ อ้างอิงและเส้นกำหนดการเคลื่อนที่ของหุ่นยนต์.....	38
รูปที่ 3.9 แสดงตัวอย่างการดึงข้อมูลความแรงสัญญาณ จากพิกัดที่บันทึกมาคำนวณ.....	41
รูปที่ 3.10 ไดอะแกรมเปรียบเทียบค่าขอบเขตเทอร์ชโฮลในแต่ละรัศมีการทำงานของอุปกรณ์ชิกปี...	43
รูปที่ 3.11 แสดงตัวอย่างของโครงข่ายประสาทเทียมสำหรับเรียนรู้รูปแบบข้อมูล.....	44
รูปที่ 3.12 แสดงรูปแบบการจัดเก็บชุดความแรงสัญญาณในไฟล์นามสกุล .xlsx.....	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 3.13 แสดงคำสั่งนำเข้าไฟล์นามสกุล .xlsx และอ่านไฟล์พร้อมจัดเก็บในตัวแปรเมทริกซ์ 2 มิติ	45
รูปที่ 3.14 โพลาร์ชาร์ตแสดงกระบวนการทำงานของวิธีการเตรียมข้อมูลสำหรับสร้างและฝึกสอน โครงข่ายประสาทเทียม.....	47
รูปที่ 3.15 สมการไฮเปอร์โบลิกแทนเจนต์ซิกมอยด์ และ โฟซิทฟีลีนีเยร์ทรานส์เฟอร์ฟังก์ชัน	49
รูปที่ 3.16 กราฟแสดงลักษณะความสัมพันธ์ของค่าน้ำหนักและค่าคลาดเคลื่อนในฟังก์ชันสูญเสีย... 52	
รูปที่ 3.17 ไดอะแกรมแสดงขั้นตอนการฝึกสอนและทดสอบโมเดลโครงข่ายประสาทเทียม (ซ้าย) ไดอะแกรมแสดงขั้นตอนการใช้งานโมเดลโครงข่ายประสาทเทียมระหว่างการใช้งานจริง (ขวา).....	57
รูปที่ 3.18 ไดอะแกรมแสดงขั้นตอนการเตรียมสภาพแวดล้อมสำหรับฝึกสอนเอเจนต์ด้วยการเรียนรู้ แบบคิวในการเรียนรู้แบบเสริมกำลัง.....	60
รูปที่ 3.19 แสดงชุดการกระทำของเอเจนต์ในหนึ่งช่องตารางสถานะภายในสนามทดสอบ	61
รูปที่ 3.20 ไดอะแกรมแสดงขั้นตอนการเรียนรู้ของเอเจนต์เพื่อหาเส้นทางไปยังเป้าหมายที่ดีที่สุด... 62	
รูปที่ 3.21 ภาพแสดงอัลกอริทึมการเรียนรู้แบบคิวของระบบค้นหาเส้นทางและส่วนตรวจสอบ ประสิทธิภาพของการเรียนรู้	63
รูปที่ 4.1 แสดงหุ่นยนต์เคลื่อนที่สำหรับทดสอบกับระบบการบอกตำแหน่งแบบเออาร์บีแอล (ARBL)	64
รูปที่ 4.2 ไดอะแกรมแสดงขั้นตอนการทำงานภายในของหุ่นยนต์เคลื่อนที่	66
รูปที่ 4.3 แสดงตัวอย่างการตั้งค่าโมดูลซิกปีบนโปรแกรม X-CTU.....	67
รูปที่ 4.4 แสดงหน้าต่างอินเตอร์เฟซของแอปพลิเคชันบนโทรศัพท์มือถือแอนดรอยด์.....	68
รูปที่ 4.5 โพลาร์ชาร์ตการทำงานของแอปพลิเคชันโอเปอเรเตอร์สำหรับสั่งการส่วนต่าง ๆ ของระบบ69	
รูปที่ 4.6 โพลาร์ชาร์ตการทำงานของแอปพลิเคชันโอเปอเรเตอร์สำหรับสั่งการส่วนต่าง ๆ ของระบบ (ต่อ).....	70
รูปที่ 4.7 สัญลักษณ์ของฐานข้อมูลคลาวด์ไฟร์เออร์เบส	71
รูปที่ 4.8 หุ่นยนต์เคลื่อนที่ที่ใช้กับระบบการบอกตำแหน่งแบบสแลม (SLAM).....	72
รูปที่ 4.9 ไดอะแกรมการทำงานของระบบการบอกตำแหน่งแบบสแลม (SLAM) ในงานวิจัย [42]... 73	
รูปที่ 4.10 แสดงกราฟความสัมพันธ์ในการบอกตำแหน่งต่าง ๆ ด้วยการแปลงพิกัดโคออดิเนตตั้งแต่ แผนที่แสดงสภาพแวดล้อมมายังตัวเฟรมของหุ่นยนต์	78
รูปที่ 4.11 ไดอะแกรมแสดงส่วนประกอบของหุ่นยนต์เคลื่อนที่และระบบควบคุมภายใต้การทำงาน ของเฟรมเวิร์กจัดการหุ่นยนต์หรือออส.....	78

สารบัญรูป (ต่อ)

หน้า

รูปที่ 5.1 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 1 พิกัด (2, 8) ในสนามทดสอบขนาด 10 เมตร x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	80
รูปที่ 5.2 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 2 พิกัด (8, 8) ในสนามทดสอบขนาด 10 เมตร x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	80
รูปที่ 5.3 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 3 พิกัด (2, 2) ในสนามทดสอบขนาด 10 เมตร x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	81
รูปที่ 5.4 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 4 พิกัด (8, 2) ในสนามทดสอบขนาด 10 เมตร x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	82
รูปที่ 5.5 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 5 พิกัด (5, 5) ในสนามทดสอบขนาด 10 เมตร x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	82
รูปที่ 5.6 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 1 พิกัด (1.5, 8.5) ในสนามทดสอบขนาด 6 x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	83
รูปที่ 5.7 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 2 พิกัด (4.5, 8.5) ในสนามทดสอบขนาด 6 x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	83
รูปที่ 5.8 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 3 พิกัด (1.5, 1.5) ในสนามทดสอบขนาด 6 x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	84
รูปที่ 5.9 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 4 พิกัด (4.5, 1.5) ในสนามทดสอบขนาด 6 X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด).....	84
รูปที่ 5.10 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 5 พิกัด (3.5, 5) ในสนามทดสอบขนาด 6 x 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตรตามขนาดของกริด).....	85

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 5.11 ภาพแสดงขอบเขตเทรซโฮลของเสาสัญญาณอ้างอิงแต่ละเสาในสนามและพื้นที่ปิดที่ไม่สมมาตร (Asymmetry Region) ของสนามทดสอบที่ 1	87
รูปที่ 5.12 ภาพแสดงขอบเขตเทรซโฮลของเสาสัญญาณอ้างอิงแต่ละเสาในสนามและพื้นที่ปิดที่ไม่สมมาตร (Asymmetry Region) ของสนามทดสอบที่ 2	88
รูปที่ 5.13 รูปภาพแสดงผลการฝึกสอนโครงข่ายประสาทเทียมสำหรับสนามทดสอบที่ 1 ด้วยโปรแกรมแมทแลป	89
รูปที่ 5.14 กราฟแสดงประสิทธิภาพของโครงข่ายประสาทเทียมของสภาวะฝึกสอนและทดสอบโมเดลของสนามทดสอบที่ 1.....	89
รูปที่ 5.15 แสดงกราฟอาร์ไอซี (ROC) ของโมเดลโครงข่ายประสาทเทียมสำหรับสนามทดสอบที่ 1	91
รูปที่ 5.16 รูปภาพแสดงผลการฝึกสอนโครงข่ายประสาทเทียมสำหรับสนามทดสอบที่ 2 ด้วยโปรแกรมแมทแลป	92
รูปที่ 5.17 กราฟแสดงประสิทธิภาพของโครงข่ายประสาทเทียมของสภาวะฝึกสอนและทดสอบโมเดลของสนามทดสอบที่ 2.....	93
รูปที่ 5.18 แสดงตัวกราฟความแรงสัญญาณที่เก็บบันทึกจากเสาสัญญาณอ้างอิง A1 แบบดั้งเดิมเทียบกับสัญญาณที่ถูกเพิ่มค่าด้วยอัตราส่วนของสัญญาณจริงต่อสัญญาณรบกวน (Signal-to-Noise Ratio) ที่ 35 เพอร์เซ็นต์.....	94
รูปที่ 5.19. แสดงกราฟรีซีฟเวอร์โอเพอเรติงคาแรคเตอร์ิสติกของโมเดลโครงข่ายประสาทเทียมสำหรับสนามทดสอบที่ 2.....	96
รูปที่ 5.20 ขณะหุ่นยนต์เคลื่อนที่จริงพร้อมทั้งบอกตำแหน่งด้วยพื้นที่ที่ไม่สมมาตรในสนามทดสอบที่ 1 (บน) และ สนามทดสอบที่ 2 (ล่าง).....	97
รูปที่ 5.21 โครงสร้างของไฟล์ต่าง ๆ ของแต่ละโปรเจกต์สำหรับการสร้างสนามทดสอบจำลองสนามทดสอบที่ 1	109
รูปที่ 5.22 พื้นที่ที่ไม่สมมาตรพร้อมหมายเลขกำกับภายในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร	110
รูปที่ 5.23 แสดงภาพตัวอย่างการจำลองสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตรแบบกริดสองมิติด้วยเทอร์เทิลไพธอนไลบรารีพร้อมหมายเลขกำกับพื้นที่ที่ไม่สมมาตร.....	110
รูปที่ 5.24 พื้นที่ที่ไม่สมมาตรพร้อมหมายเลขกำกับภายในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร (ซ้าย) ภาพตัวอย่างการจำลองสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตรแบบกริดสองมิติด้วยเทอร์เทิลไพธอนไลบรารีพร้อมหมายเลขกำกับพื้นที่ที่ไม่สมมาตร (ขวา).....	115

สารบัญรูป (ต่อ)

หน้า

รูปที่ 5.25 โครงสร้างของไฟล์ต่าง ๆ ของแต่ละโปรเจกต์สำหรับการใช้ในการสร้างสนามทดสอบจำลองสนามทดสอบที่ 2	115
รูปที่ 5.26 ภาพจำลองการสร้างเส้นทางของระบบการเรียนรู้แบบคิวสำหรับหุ่นยนต์เคลื่อนที่ในสนามจำลอง เปรียบเทียบระหว่างการพิจารณาตำแหน่งของวัตถุในสนามตามปกติ (ซ้าย) และการพิจารณาแบบวิธีเออาร์บีแอล (ขวา) ของทั้งสนามที่สอบที่ 1 (บน) และ 2 (ล่าง) ตามลำดับ	120
รูปที่ 5.27 กราฟเปรียบเทียบการเปลี่ยนแปลงความต่างของค่าคิวในแต่ละรอบการทำงาน ระหว่างวิธีพิจารณาสนามตามปกติ และวิธีพิจารณาสนามแบบเออาร์บีแอลของสนามทดสอบที่ 1	121
รูปที่ 5.28 กราฟเปรียบเทียบการเปลี่ยนแปลงความต่างของค่าคิวในแต่ละรอบการทำงาน ระหว่างวิธีพิจารณาสนามตามปกติ และวิธีพิจารณาสนามแบบเออาร์บีแอลของสนามทดสอบที่ 2	121
รูปที่ 5.29 วิธีการบอกตำแหน่งภายในสภาพแวดล้อมด้วยสแลมบนรอสแพลตฟอร์ม (ซ้าย) ผลลัพธ์จากการสแกนพื้นที่เป้าหมายและแสดงในโปรแกรมอาร์วิส (Rviz) ด้วยข้อมูลประเภทพอยท์คลาวด์ (ขวา) พื้นที่ปิดภายในอาคารที่หุ่นยนต์เคลื่อนที่ทำการสแกน	124
รูปที่ 5.30. แสดงผลลัพธ์การสแกนพื้นที่ทดสอบบนโปรแกรมอาร์วิสด้วยวิธีการบอกตำแหน่งแบบสแลมบนรอสแพลตฟอร์ม	126

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

วิทยาการหุ่นยนต์กลายมาเป็นปัจจัยหนึ่งซึ่งช่วยเพิ่มผลผลิตให้แก่องค์กรหรือหน่วยงานต่าง ๆ ทั้งภาครัฐและภาคเอกชน ช่วยเพิ่มประสิทธิภาพและความรวดเร็วในการทำงานที่ต้องอาศัยความแม่นยำ การทำซ้ำหลาย ๆ ครั้งหรือแบกรับภาระที่เกินกว่ามนุษย์จะสามารถจัดการได้เอง ในงานอุตสาหกรรม การเคลื่อนย้ายวัตถุดิบ ชิ้นส่วนอุปกรณ์ หรือวัตถุเป้าหมายใด ๆ ก็ตามเป็นอีกภารกิจสำคัญภารกิจหนึ่งที่เป็นตัวขับเคลื่อนให้กระบวนการผลิตนั้นดำเนินต่อไปได้ หลายปีมานี้หุ่นยนต์เคลื่อนที่เข้ามามีบทบาทสำคัญในการขนส่งขนถ่ายสินค้าระหว่างสายการผลิตภายในโรงงานคลังสินค้า หรือพื้นที่ภายในอาคารทั่วไป มีการใช้งานกันอย่างแพร่หลายและเป็นที่ยอมรับมากขึ้น การเพิ่มขีดความสามารถในการคิดวิเคราะห์และการตัดสินใจต่อสถานการณ์ต่าง ๆ ของหุ่นยนต์นั้นจึงเป็นสิ่งที่ได้รับความสนใจและถูกพัฒนาอย่างยาวนาน เช่นเดียวกับการบอกตำแหน่งและนำทางหุ่นยนต์ซึ่งเป็นหัวใจสำคัญในการกำหนดวางแผนเส้นทางเพื่อให้หุ่นยนต์เคลื่อนที่ไปยังบริเวณที่ต้องการ

หุ่นยนต์เคลื่อนที่ที่ใช้งานตามโรงงานอุตสาหกรรม สามารถแบ่งประเภทออกได้อย่างคร่าว ๆ เป็นสองกลุ่มด้วยกันคือ หุ่นยนต์เคลื่อนที่อัตโนมัติและหุ่นยนต์เคลื่อนที่ตามเส้น โดยทั้งสองประเภคนั้นมีข้อดีข้อเสียที่แตกต่างกัน หุ่นยนต์เคลื่อนที่ตามเส้นนั้นจะอาศัยสัญลักษณ์ เส้นนำทาง หรือ เสาสัญญาณ เป็นตัวนำทางไปตามพื้นที่ต่าง ๆ ในสนาม ทำให้ง่ายต่อการควบคุมรูปแบบการเคลื่อนที่ การวิเคราะห์เส้นทางและการควบคุมฝูงหุ่นยนต์ในกรณีที่มีหุ่นหลายตัวปฏิบัติการในพื้นที่เดียวกัน อย่างไรก็ตาม ขอบเขตการเคลื่อนที่นั้นจะถูกจำกัดและไม่สามารถตอบสนองต่อสิ่งกีดขวางหรือเหตุฉุกเฉินกระทันหันใด ๆ ได้อย่างอิสระ อีกทั้งการติดตั้งสัญลักษณ์หรือเส้นนำทางต่าง ๆ ในพื้นที่ใช้งานจริงอาจเป็นเรื่องที่ยุ่งยากซับซ้อนและต้นทุนสูงอีกด้วย ในทางกลับกัน หุ่นยนต์เคลื่อนที่อัตโนมัติจะมีความยืดหยุ่นมากกว่าเนื่องจากหุ่นยนต์เคลื่อนที่อัตโนมัติส่วนมากใช้อัลกอริทึมที่เรียกว่า สแลม (Simultaneous Localization and Mapping: SLAM) ซึ่งทำให้หุ่นยนต์มีความสามารถในการแมปปิ้งสภาพแวดล้อมโดยรอบ และระบุตำแหน่งของตัวเองได้ในเวลาเดียวกัน ผนวกกับอัลกอริทึมในการวางแผนเส้นทางเคลื่อนที่ (Path planning) หุ่นยนต์จึงมีความชาญฉลาดในการตัดสินใจขณะเผชิญหน้ากับสิ่งกีดขวางได้ และสามารถสร้างเส้นทางหลบเลี่ยงได้เอง โดยไม่จำเป็นต้องอาศัยเส้นนำทางใด ๆ สามารถขยายขนาดของฝูงและขอบเขตการทำงานหรือการเคลื่อนที่ไปยังพื้นที่อื่น ๆ ได้อย่างอิสระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถึงแม้จะมีแนวคิดมากมายที่สามารถประยุกต์ใช้งานจริงได้อย่างมีประสิทธิภาพ แต่เทคโนโลยีสแลมในปัจจุบันนั้นยังคงมีอุปสรรคด้านความต้องการอุปกรณ์ประสิทธิภาพสูง รวมไปถึงความซับซ้อนในการพัฒนาและประยุกต์ใช้งานจริง ซึ่งส่งผลกระทบต่อต้นทุนการผลิตตัวหุ่นยนต์และระบบควบคุมโดยรวม

1.2 ความหมายและวัตถุประสงค์ของการศึกษา

ในปัจจุบัน เครือข่ายไร้สายนั้นมีบทบาทสำคัญในการบอกตำแหน่งและการนำทางไปยังเป้าหมายที่ต้องการภายในอาคารที่ซึ่งสัญญาณของอุปกรณ์บอกตำแหน่งอย่างจีพีเอส (GPS: Global Positioning System) ไม่สามารถเข้าถึงได้เนื่องจากข้อจำกัดทางการรับส่งข้อมูล ระบบการบอกตำแหน่งภายในอาคาร (Indoor Positioning System: IPS) ที่มีเครือข่ายสื่อสารไร้สายเป็นหัวใจหลักนั้นจึงกลายเป็นทางออกสำคัญของปัญหาดังกล่าว เทคโนโลยีนี้ถูกนำไปใช้งานกันอย่างแพร่หลายไม่ว่าจะในองค์กรต่าง ๆ หรือพื้นที่สาธารณะเช่น ห้างสรรพสินค้า สนามบิน หรือ หอประชุม เป็นต้น ขณะเดียวกัน เครือข่ายไร้สายนั้นก็ยังมีอิทธิพลต่อการบอกตำแหน่งและนำทางหุ่นยนต์ภายในอาคารเช่นกัน หุ่นยนต์เคลื่อนที่บางประเภทอาศัยเสาสัญญาณ (Beacon) หรืออุปกรณ์แท็ก (Tag) มาใช้อ้างอิงในการบอกตำแหน่งและเคลื่อนที่ไปในพื้นที่เป้าหมาย มีงานวิจัยมากมายที่คิดค้นและพัฒนาอัลกอริทึมสำหรับบอกตำแหน่งเป้าหมายด้วยความแรงสัญญาณ ตัวอย่างเช่น การบอกตำแหน่งแบบไม่อ้างอิงระยะทาง (Range-free), การบอกตำแหน่งแบบอ้างอิงพื้นที่ (Area-based), และการบอกตำแหน่งแบบความน่าจะเป็นของข้อมูล (Heuristic information) เป็นต้น อย่างไรก็ตาม แนวคิดการบอกตำแหน่งแต่ละชนิดที่ยกตัวอย่างมาล้วนมีทั้งข้อดีและข้อเสียต่าง ๆ ปะปนกันไป จากสาเหตุนี้เองจึงเป็นจุดเริ่มต้นของแนวคิดที่จะนำข้อได้เปรียบหรือจุดแข็งของแนวคิดการบอกตำแหน่งภายในอาคารด้วยความแรงสัญญาณรูปแบบต่าง ๆ มาประยุกต์ ผสมเข้ากับการใช้เซนเซอร์ตรวจจับวัตถุระยะสั้นอย่างง่าย ราคาถูก และหาได้ตามท้องตลาดทั่วไป เพื่อที่จะลดการใช้ทรัพยากรในการสร้างหุ่นยนต์ อันเนื่องมาจากการใช้ตัวประมวลผลขนาดใหญ่และอุปกรณ์เซนเซอร์ราคาสูง ทั้งยังคงความสามารถในการบอกตำแหน่งของตัวหุ่นยนต์เอง รวมไปถึงตำแหน่งของวัตถุที่ตรวจพบได้อย่างมีประสิทธิภาพ และลดความคลาดเคลื่อนในการแมปผังแผนที่ให้ตำแหน่งที่ถูกต้องที่สุดเท่าที่จะเป็นไปได้

วิทยานิพนธ์นี้ต้องการพัฒนาวิธีการบอกตำแหน่งเป้าหมายด้วยความแรงสัญญาณเพื่อนำมาใช้ทดแทนวิธีการแมปผังพื้นที่แบบเดิมของหุ่นยนต์ เพื่อลดต้นทุนในการใช้อุปกรณ์ควบคุมและเซนเซอร์ตรวจจับสภาพแวดล้อมราคาสูง อาทิเช่น เลเซอร์วัดระยะทาง (LiDAR), กล้องหรืออุปกรณ์เซนเซอร์จับภาพ เป็นต้น อีกทั้งลดความซับซ้อนของแนวคิดแบบเดิมเพื่อให้ง่ายต่อการพัฒนาและปรับใช้กับส่วนการวางแผนเส้นทางการเคลื่อนที่ของหุ่นยนต์ต่อไป

1.3 สมมุติฐานของการศึกษา

จากการทบทวนวรรณกรรมต่าง ๆ การบอกตำแหน่งในอาคารด้วยความแรงสัญญาณจากเครือข่ายสื่อสารไร้สายโดยส่วนมากนั้นจะประสบปัญหากับความไม่แน่นอนในการบอกระยะทางซึ่งมีสาเหตุมาจากความแปรปรวนของความแรงสัญญาณ เนื่องจากเป็นพื้นที่ที่ปิด สัญญาณที่เดินทางระหว่างเสารับและเสาส่งมีโอกาสที่จะเกิดการสะท้อน การดูดซับ และการกระจายของสัญญาณได้สูง แนวโน้มของความแรงสัญญาณที่แปรผันตามระยะห่างของเสาสัญญาณนั้นมีลักษณะไม่เป็นเชิงเส้น ทำให้ผลลัพธ์จากการคำนวณตำแหน่งคลาดเคลื่อนไปจากตำแหน่งที่แท้จริง ถึงแม้จะมีบางวรรณกรรมที่นำเสนอแนวความคิดการใช้ความแรงสัญญาณบอกตำแหน่งเป้าหมายโดยอ้อม ผลปรากฏว่ากระบวนการบางอย่างค่อนข้างซับซ้อน ทำให้เกิดอุปสรรคต่อการออกแบบและนำไปปรับใช้จริง

ดังนั้นในงานวิจัยนี้จึงได้นำเสนอการบอกตำแหน่งภายในอาคารด้วยความแรงสัญญาณแบบจำแนกขอบเขตพื้นที่ที่ไม่สมมาตร โดยสมมุติฐานของงานวิจัยคือผลลัพธ์การบอกตำแหน่งของเป้าหมายจะมีลักษณะเป็นพื้นที่ที่ไม่สมมาตรรูปทรงต่าง ๆ อันเนื่องมาจากเกณฑ์การแบ่งพื้นที่ย่อย ๆ ในสนามทดสอบที่ได้ออกแบบขึ้น งานวิจัยนี้คาดหวังว่าการระบุตำแหน่งในลักษณะนี้จะช่วยทำให้หุ่นยนต์สามารถระบุตำแหน่งของตัวเองและสิ่งแวดล้อมที่ตรวจจับได้ระหว่างการแมปปีง ด้วยค่าความคลาดเคลื่อนที่น้อยที่สุด ซึ่งจากผลการทดสอบนั้นแสดงให้เห็นว่าระบบสามารถระบุตำแหน่งเป้าหมายได้เป็นอย่างดี อีกทั้งยังลดความซับซ้อนในการแมปปีงแผนที่ของหุ่นยนต์อีกด้วย

1.4 ขอบเขตการวิจัย

วิทยานิพนธ์นี้ได้นำเสนอวิธีการบอกตำแหน่งภายในอาคารแบบขอบเขตพื้นที่ที่ไม่สมมาตรซึ่งใช้เกณฑ์ความสามารถในการเข้าถึงความแรงสัญญาณของอุปกรณ์สื่อสารแต่ละตัว มากำหนดขอบเขตพื้นที่ที่ไม่สมมาตรหรือรีเจียน (Region) ต่าง ๆ ในสนาม โดยอุปกรณ์สื่อสารไร้สายนั้นจะเป็นโมดูลเอ็กซ์บีซีรี่ส์ 2 (Xbee series 2) จำนวน 5 ตัว ทำหน้าที่เป็นโหนดอ้างอิง ติดตั้งอยู่ตามตำแหน่งต่าง ๆ ของสนามและอีก 1 ตัวทำหน้าที่เป็นโหนดเป้าหมาย ติดตั้งอยู่กับหุ่นยนต์เคลื่อนที่พร้อมทั้งติดตั้งเซนเซอร์ตรวจจับวัตถุสิ่งกีดขวางระยะสั้น เพื่อให้หุ่นยนต์ส่งชุดความแรงสัญญาณ ณ ตำแหน่งที่ตรวจพบสิ่งกีดขวางใด ๆ ระหว่างการแมปปีงพื้นที่ ไปยังสถานีฐานที่มีโครงข่ายประสาทเทียมสำหรับเรียนรู้และจำแนกความแรงสัญญาณของตำแหน่งที่ตรวจพบว่าอยู่ภายในรีเจียนใด ระบบทั้งหมดจะเชื่อมต่อกับฐานข้อมูลคลาวด์ ทำให้ผู้ทำการทดลองสามารถมอนิเตอร์และควบคุมกระบวนการต่าง ๆ ทั้งหมดผ่านโทรศัพท์มือถือจากภายนอกสนามได้

1.5 ขั้นตอนการศึกษา

1. ศึกษาความเป็นมา ทฤษฎีพื้นฐานที่ใช้ในงานวิจัย และการใช้งานอุปกรณ์ต่าง ๆ
2. ศึกษาแมชชีนเลิร์นนิงการแบ่งประเภทข้อมูล และการเรียนรู้แบบเสริมกำลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. จัดเตรียมอุปกรณ์ ติดตั้งเครือข่ายอินเทอร์เน็ตไร้สาย พร้อมทั้งเก็บข้อมูลชุดความแรงสัญญาณในสนามต่าง ๆ ที่ใช้ทำการทดสอบ เพื่อสร้างฐานข้อมูลสำหรับใช้ในงานวิจัย
4. แบ่งขอบเขตพื้นที่ด้วยเกณฑ์ความสามารถในการซ้ำค่าของอินเทอร์เน็ตไร้สายแต่ละตัวในสนาม
5. สร้างและฝึกสอนโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ในสนามด้วยฐานข้อมูลความแรงสัญญาณที่ได้บันทึกไว้ในขั้นตอนก่อนหน้า
6. หุ่นยนต์ทำการแมปปิ้งจริงในสนามทดสอบ นำแผนที่ที่ได้ไปประยุกต์กับอัลกอริทึมสำหรับวางแผนเส้นทางเพื่อทดสอบประสิทธิภาพ และเก็บผลการทดลอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 กล่าวนำ

ในบทที่สองนี้จะกล่าวถึงหลักการ ทฤษฎี และงานวิจัยที่เกี่ยวข้องในการจัดทำวิทยานิพนธ์ ซึ่งจะประกอบไปด้วย แนวคิดการบอกตำแหน่งภายในอาคารด้วยความแรงสัญญาณรูปแบบต่าง ๆ วิธีการระบุตำแหน่งพร้อมกับการสร้างแผนที่ ทฤษฎีและหลักการของโครงข่ายประสาทเทียมในแมชชีนเลิร์นนิง การเรียนรู้แบบแบ่งประเภทข้อมูลและการเรียนรู้แบบเสริมกำลัง หลักการทำงานของส่วนประกอบหลักในระบบ ทั้งในส่วนของหุ่นยนต์และเครือข่ายสื่อสารไร้สาย รวมไปถึงหลักการหามาตรฐานความสามารถในการซ้ำค่าของอุปกรณ์ซึ่งจะเป็นเกณฑ์สำคัญที่ใช้กำหนดขอบเขตการแบ่งพื้นที่ในสนามทดสอบ

2.2 แนวคิดการบอกตำแหน่งภายในอาคารด้วยความแรงสัญญาณ (Indoor Localization with RSSI)

[1] การบอกตำแหน่งเป้าหมายโดยคร่าว ๆ นั้นจะใช้โหนดสื่อสารไร้สาย หรือเซนเซอร์โหนดจำนวน 2 ตัวขึ้นไปในการตรวจจับตามเส้นทางที่เป้าหมายเคลื่อนที่ผ่านไป พิกัดของเป้าหมายที่ต้องการติดตามนั้นจะสามารถคำนวณได้จากการเฉลี่ยพิกัดที่อ่านค่าได้จากเซนเซอร์ทั้งหมดที่สามารถอ่านค่าตำแหน่งได้ ด้วยการใช้เทคนิคนี้เองทำให้ความถูกต้องแม่นยำในการบอกตำแหน่งและความละเอียดในการบอกที่ตั้งของเป้าหมายจะขึ้นอยู่กับปริมาณความหนาแน่นของโหนดเซนเซอร์ หรือโหนดสื่อสารไร้สายในบริเวณนั้น ๆ ถ้าหากจำนวนโหนดในพื้นที่มีน้อย ความละเอียดในการบอกพิกัดตำแหน่งก็จะน้อยลงไปด้วย

ขณะที่การบอกตำแหน่งเป้าหมายอย่างละเอียดนั้น จะใช้โหนดเซนเซอร์หรือโหนดสื่อสารไร้สายอย่างน้อย 3 ตัวหรือมากกว่านั้นเข้ามาจับผิดชอบในการติดตามการเปลี่ยนแปลงตำแหน่งภายในบริเวณที่กำหนดอุปกรณ์ทั้ง 3 สามารถที่จะบอกระยะห่างระหว่างคูโหนดรับส่งสัญญาณ ด้วยการหาระยะนี้เองทำให้เทคนิคนี้สามารถที่จะบอกตำแหน่งของเป้าหมายที่แน่นอน ซึ่งจะสามารถคำนวณได้จากการหาตำแหน่งเชิงมุมหรือการหาตำแหน่งเชิงพิกัด ดังนั้นการเพิ่มปริมาณความหนาแน่นของโหนดสื่อสารภายในพื้นที่ไม่ได้ช่วยเพิ่มประสิทธิภาพและความแม่นยำในการบอกตำแหน่ง หากแต่ความแม่นยำจะขึ้นอยู่กับวิธีการคำนวณหาระยะห่างระหว่างอุปกรณ์มากกว่า

2.2.1 การบอกตำแหน่งและติดตามตำแหน่งของอุปกรณ์เป้าหมาย จำแนกจากปัจจัยที่ใช้วัดระยะทาง

ปัจจัยที่ใช้วัดระยะทางนั้นมีอยู่ด้วยกัน 4 อย่างคือ มุมองศาจากทิศทางขาเข้าของสัญญาณ หรือเอโอเอ (Angle Of Arrival: AOA), เวลาที่สัญญาณเดินทางมาถึง หรือ ทีโอเอ (Time Of Arrival: TOA), ความแตกต่างของเวลาที่สัญญาณเดินทางมาถึงหรือ ทีดีโอเอ (Time Difference Of Arrival: TDOA), และความแรงของสัญญาณที่ตรวจวัดได้หรือ อาร์เอสเอสไอ (Received Signal Strength: RSS) โดยที่วิธีเอโอเอคือวิธีที่ใช้วัดมุมองศาที่ของคลื่นที่เดินทางมาถึงเสารับสัญญาณโดยการเปรียบเทียบทิศทางของสัญญาณกับมุมอ้างอิง แต่ข้อเสียอย่างหนึ่งของวิธีนี้คืออุปกรณ์รับสัญญาณจำเป็นต้องมีตัวบอกทิศทาง หรือบอกมุมองศาติดตั้งอยู่กับตัว

สำหรับตัวแปรทีโอเออนั้นจะใช้ในกรณีที่ระบบสื่อสารนั้นถูกสร้างขึ้นแบบมีศูนย์กลางควบคุม (centralized) วิธีการวัดระยะทางทำได้โดยจับเวลาที่สัญญาณเดินทางมาถึงจากเสาส่งมายังเสารับ ซึ่งสามารถทำได้สองลักษณะคือ อย่างแรก ให้เสาส่งทำการกระจายสัญญาณไปยังเสารับทุกตัวในบริเวณ จากนั้นเสารับทุกตัวจะตอบข้อมูลเวลาที่สัญญาณมาถึงตนให้กับศูนย์กลางของระบบได้รับทราบ ลักษณะที่สองคือ ให้เสาส่งหลาย ๆ ตัวทำการส่งสัญญาณมายังเสารับเพียงเสาเดียว ตัวรับก็จะทำการจับเวลาที่สัญญาณแต่ละชุดเดินทางมาถึงและทำการเปรียบเทียบความสัมพันธ์ของข้อมูลที่ได้เพื่อหาระยะทาง อย่างไรก็ตามวิธีนี้มีข้อจำกัดทางด้านเทคนิคในเรื่องของความสอดคล้องในการทำงาน โดยเสาส่งทุกตัวจะต้องทำงานประสานกันเพื่อที่จะให้สามารถส่งข้อมูลได้พร้อม ๆ กัน อีกทั้งอาจจะประสบปัญหาได้ในเรื่องของการสูญเสียสัญญาณบางส่วนไปจากการชนกัน เนื่องจากว่าสัญญาณถูกส่งมายังตัวรับพร้อมกันจากหลาย ๆ ที่ในเวลาเดียวกัน

วิธีทีดีโอเออนั้นเป็นการปรับปรุงเพิ่มเติมจากวิธีทีโอเอโดยหลีกเลี่ยงการรับส่งข้อมูลพร้อม ๆ กัน เพื่อไม่ให้เกิดการสูญเสียของข้อมูล สำหรับการประยุกต์ใช้วิธีทีดีโอเอเสาส่งจำเป็นต้องส่งข้อมูลที่แตกต่างกัน 2 ชุดด้วยความเร็วในการส่งที่ไม่เท่ากัน เมื่อเสารับสามารถรับข้อมูลทั้ง 2 ชุดนี้แล้ว มันจะทำการคำนวณค่าต่างของเวลาที่สัญญาณทั้งสองชุดเดินทางมาถึง จากการใช้งานเทคนิคนี้ เราก็จะได้เวลาที่สัญญาณใช้ในการเดินทางหรือทีโอเอฟ (Time of Flight: TOF) ซึ่งก็คล้ายคลึงกันกับการแพร่กระจายของสัญญาณ (propagation time)

การใช้ความแรงสัญญาณอาร์เอสเอสไอ (Received Signal Strength) นั้นเป็นวิธีที่ใช้หาระยะห่างจากการลดทอนลงของสัญญาณที่เคลื่อนที่ไป ถ้าหากเรารู้กำลังส่งของเสาส่งข้อมูล การลดทอนของสัญญาณโดยรวมก็จะสามารถคำนวณได้จากการหาส่วนต่างระหว่าง กำลังส่งที่วัดจากเสาส่งและกำลังส่งที่เสารับวัดได้

2.2.2 เทคนิคการบอกตำแหน่งด้วยความแรงสัญญาณ

[2] เทคนิคที่ใช้ในการบอกตำแหน่งนั้นเป็นสิ่งแรกที่จะต้องพิจารณาเมื่อเริ่มต้นออกแบบระบบ เนื่องจากเทคนิคที่ใช้นั้นจะเป็นตัวกำหนดทิศทางการประมวลผลว่าควรเป็นแบบศูนย์กลางหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระจายตัว วิธีที่ใช้ในการคำนวณหาตำแหน่งและอุปกรณ์ที่เกี่ยวข้องในการประมวลผลของระบบ แม้ว่าจะมีวิธีต่าง ๆ มากมายในการบอกตำแหน่งเป้าหมายภายในอาคารด้วยความแรงสัญญาณ จากงานวิจัยต่าง ๆ ที่ได้ศึกษามานั้น เทคนิคที่หยิบยกมากล่าวถึงในวิทยานิพนธ์สามารถแบ่งออกได้เป็น 4 ประเภท ดังต่อไปนี้

2.2.2.1 การบอกตำแหน่งแบบอ้างอิงระยะทาง (Range-based)

เป็นเทคนิคที่การบอกตำแหน่ง หรือระยะห่างระหว่างเสาสัญญาณนั้นขึ้นอยู่กับปัจจัยที่ใช้วัดซึ่งจะมีอยู่ด้วยกัน 4 อย่างคือมุมมองจากทิศทางขาเข้าของสัญญาณ, เวลาที่สัญญาณเดินทางมาถึง, ความแตกต่างของเวลาที่สัญญาณเดินทางมาถึงและความแรงสัญญาณที่ตรวจวัดได้ ปัจจัยต่าง ๆ เหล่านี้จะเป็นตัวแปรหลักที่ส่งผลต่อการบอกตำแหน่งเป้าหมาย ข้อดีของเทคนิคนี้คือความง่ายในการคำนวณที่ไม่จำเป็นต้องอาศัยความรู้พื้นฐานจากเรื่องอื่นมาก การระบุตำแหน่งที่ชัดเจนซึ่งสามารถให้ผลลัพธ์ในรูปแบบของพิกัดตำแหน่ง และประหยัดต้นทุนในการติดตั้ง เนื่องจากไม่จำเป็นต้องมีอุปกรณ์ในการคำนวณอื่น ๆ เข้ามาเสริมเพิ่มเติม โดยเฉพาะกับการใช้ความแรงสัญญาณ ซึ่งอุปกรณ์สื่อสารส่วนมากจะมีฟังก์ชันวัดความแรงของสัญญาณระหว่างเสาส่งอยู่ในตัว เทคนิคการบอกตำแหน่งแบบอ้างอิงระยะทางที่นิยมใช้กันมากจะมีอยู่ด้วยกัน 2 แบบหลัก ๆ คือ

2.2.2.1.1 การบอกตำแหน่งแบบ ไตรแองกูลेशन (Triangulation)

การบอกตำแหน่งแบบไตรแองกูลेशन [2] นั้นเป็นแนวคิดการใช้ทฤษฎีตรีโกณมิติ มาคำนวณหาตำแหน่งเป้าหมาย โดยใช้การทำมุมมอง 2 มุม และระยะห่างระหว่างเป้าหมายและจุดอ้างอิง ในเครือข่ายเซนเซอร์ไร้สาย โหนดอ้างอิงสองโหนดนั้นจะถูกใช้ในการกำหนดเส้นฐาน (baseline) ของแกน x และโหนดเซนเซอร์อีกสองโหนดใช้ในการกำหนดเส้นฐานของแกน y ระยะห่าง d_r ระหว่างโหนดอ้างอิงทั้งสองโหนดบนเส้นฐานนั้นสามารถวัดและบันทึกลงฐานข้อมูลเพื่อนำไปใช้ในการคำนวณหาตำแหน่งของโหนดเป้าหมายจริง อีกทั้งมุม α_1 และ α_2 นั้นสามารถวัดได้ระหว่างเส้นฐานและเส้นที่ลากขึ้นจาก โหนดอ้างอิงและโหนดเป้าหมาย

โหนดอ้างอิง R_1 และ R_2 ก่อให้เกิดเส้นฐานในแนวแกน x โหนด R_1 เองก็สามารถร่วมกับโหนด R_3 สร้างเส้นฐานในแนวแกน y โหนดเป้าหมาย T_1 นั้นสามารถเคลื่อนที่ได้อย่างอิสระในพื้นที่ พิกัด (x, y) ของโหนดเป้าหมายนั้นสามารถคำนวณได้จากสมการดังนี้

$$x = \frac{d_{ry}}{\tan^{-1}(\alpha_{y1}) + \tan^{-1}(\alpha_{y2})} \quad (2.1)$$

$$y = \frac{d_{rx}}{\tan^{-1}(\alpha_{x1}) + \tan^{-1}(\alpha_{x2})} \quad (2.2)$$

โดยที่ d_{ry} = ระยะห่างระหว่างโหนดอ้างอิงในแนวแกน y

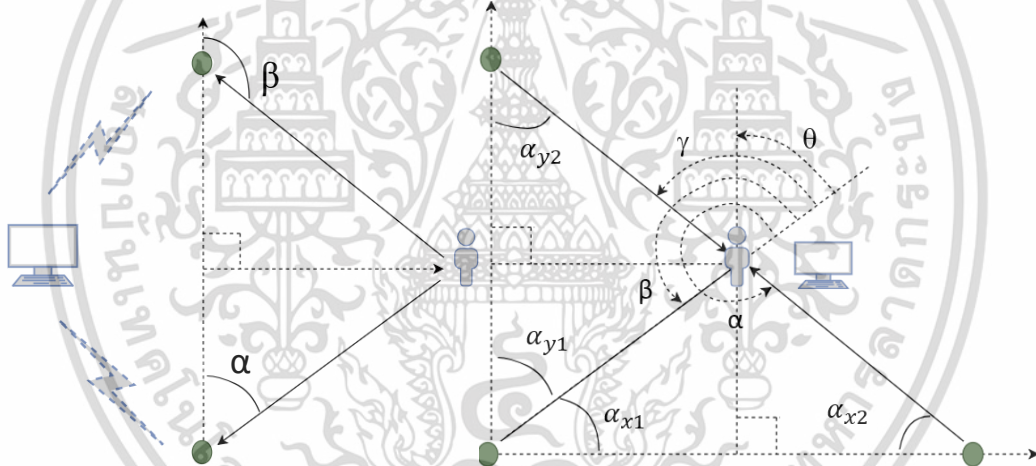
d_{rx} = ระยะห่างระหว่างโหนดอ้างอิงในแนวแกน x

α_x = มุมองศาที่กระทำระหว่างแนวเสาสัญญาณอ้างอิงของแกน x กับเสาสัญญาณเป้าหมาย

α_y = มุมองศาที่กระทำระหว่างแนวเสาสัญญาณอ้างอิงของแกน y กับเสาสัญญาณเป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโครงสร้างของเครือข่าย การคำนวณนั้นสามารถทำได้ในระบบบอกตำแหน่งแบบมีศูนย์กลาง ประมวลผลเก็บค่ามุมจากโหนดอ้างอิงต่าง ๆ หรืออาจจะเก็บโนโหนดเป้าหมายเองเลยก็สามารถทำได้ สำหรับกรณีแรก โหนดเป้าหมายทำการกระจายสัญญาณไปโหนดอื่น ๆ และโหนดอ้างอิงทำการวัดมุมองศาของสัญญาณที่รับเข้ามาได้ จากนั้นโหนดอ้างอิงทำการส่งข้อมูลมุมองศาที่วัดได้ไปยังศูนย์กลางของระบบ ดังแสดงในรูปที่ 2.1 (ซ้าย) ในกรณีนี้ โหนดอ้างอิงโหนดแรกจะทำการวัดมุมแหลมใด ๆ (Acute angle) α และโหนดอ้างอิงโหนดที่สองจะทำการวัดมุมป้าน (Obtuse angle) β ดังนั้น มุมประกอบมุมฉาก (Supplementary angle) ของ β หรือ $(\pi - \beta)$ คือมุมแหลมสำหรับโหนดที่สอง สำหรับกรณีที่สอง การคำนวณหาตำแหน่งแบบไตรแองกูลেশันนั้นทำได้จากภายในโหนดเป้าหมาย ถ้าหากตัวโหนดมีเซ็นเซอร์เข็มทิศติดตั้งอยู่ด้วย เข็มทิศแม่เหล็ก (Magnetic compass) จะทำการวัดมุมองศา ทิศทางที่โหนดเป้าหมายทำมุม โหนดอ้างอิงทุกตัวจะทำการกระจายสัญญาณไปยังโหนดเป้าหมาย ดังนั้น โหนดเป้าหมายจะสามารถทำการวัดมุม α, β และ γ จากสัญญาณที่ได้รับได้ของโหนดอ้างอิงทั้งสามจุด โหนดเป้าหมายทำการคำนวณพิกัดและส่งข้อมูลไปเก็บไว้ยังฐานข้อมูลส่วนกลาง แสดงดังรูปที่ 2.1 (ขวา)

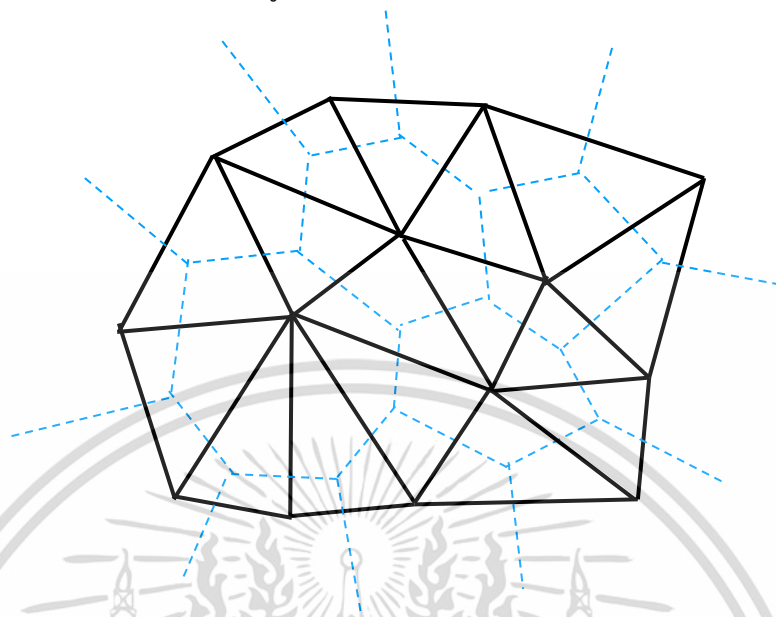


รูปที่ 2.1 การบอกตำแหน่งเป้าหมายด้วยวิธีไตรแองกูลেশัน (Triangulation) แบบ แบบศูนย์กลาง การประมวลผลอยู่กับเครื่องเซิร์ฟเวอร์ (ซ้าย) และแบบศูนย์กลางการประมวลผลอยู่กับโหนดเป้าหมาย (ขวา)

การใช้เซ็นเซอร์เข็มทิศแม่เหล็กแบบอิเล็กทรอนิกส์ (Electronic Magnetic Compass: EMC) ที่ติดตั้งกับโหนดเซ็นเซอร์ไร้สายนั้น ทำให้เราสามารถคำนวณหามุมองศาที่ต้องการได้ โดยการชดเชยค่ามุม (offset) นั้นจะใช้ในการปรับค่าอุปกรณ์เพื่อกำหนดมุมอ้างอิง นอกเหนือจากวิธีไตรแองกูลেশันแบบปกติแล้ว วิธีที่ซับซ้อนแต่สมบูรณ์แบบกว่าในการประยุกต์ใช้แนวคิดไตรแองกูลেশัน คือวิธีไตรแองกูลেশันแบบไดนามิก (Dynamic Triangulation) โดยวิธีนี้จะเน้นการประยุกต์ใช้กับการบอกตำแหน่งแบบสามมิติในเครือข่ายที่มีสเกลที่กว้าง ยิ่งไปกว่านั้นเทคนิคเดอลอนีย์ไตรแองกูลেশัน (Delaunay Triangulation) ดังแสดงในรูปที่ 2.2 สามารถใช้ในการบอกตำแหน่งสำหรับรูปแบบการวางโหนดที่ซับซ้อน และกระจายตัวแบบสุ่มโดยเครือข่ายที่เชื่อมต่อกันเป็นลักษณะพื้นที่สามเหลี่ยมนั้น สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

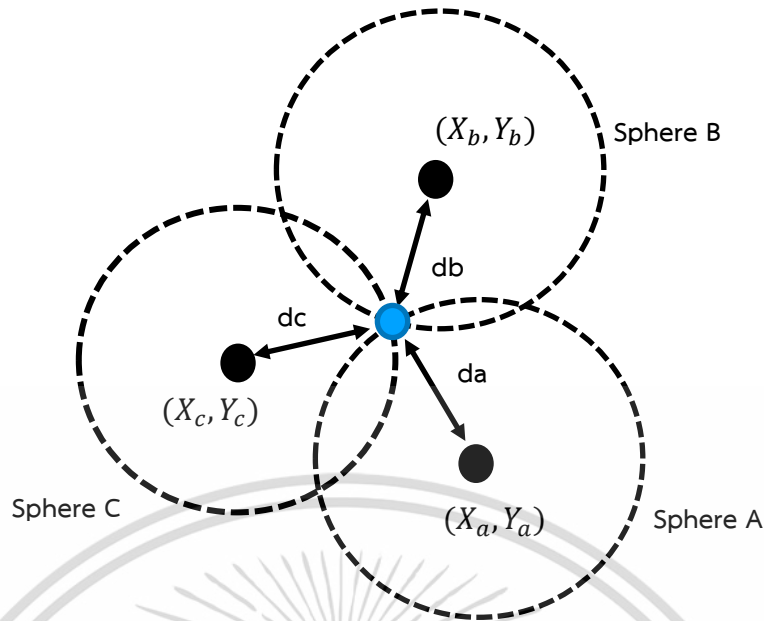
ทำให้มีประสิทธิภาพที่สูงขึ้นได้ (optimize) ได้โดยการใช่วิธีสตีพเพสต์เดสเซนต์ (Steepest Descent method) ซึ่งจะทำให้โครงสร้างนั้นสมบูรณ์มากขึ้น และบอกตำแหน่งเป้าหมายได้แม่นยำยิ่งขึ้น



รูปที่ 2.2 ภาพแสดงแนวคิดการบอกตำแหน่งแบบ เดอลอนนีย์ไตรแองกูลേഷัน (Delaunay Triangulation)

2.2.2.1.2 การบอกตำแหน่งแบบไตรแลทเตอเรชัน (Trilateration)

การบอกตำแหน่งแบบไตรแลทเตอเรชัน ใช้ในการหาตำแหน่งเป้าหมายจากตำแหน่งอ้างอิงหลาย ๆ จุด อย่างไรก็ตาม ความแตกต่างระหว่างอัลกอริทึมไตรแลทเตอเรชันกับไตรแองกูลേഷัน คือข้อมูลที่ใช้ในการประมวลผล แทนที่จะวัดมุมมองระหว่างโหนดเซนเซอร์ไร้สาย วิธีไตรแลทเตอเรชันจะใช้ระยะทางระหว่างตำแหน่งต่าง ๆ มาเป็นข้อมูลในการคำนวณพิกัดเป้าหมาย ซึ่งระยะทางระหว่างโหนดเซนเซอร์ไร้สายนั้นสามารถพิจารณาในรูปของ รัศมีของวงกลมที่มีจุดศูนย์กลางเป็นตำแหน่งของโหนดเซนเซอร์อ้างอิง ดังนั้นพิกัดเป้าหมายนั้นก็เกิดจากจุดตัดของวงกลมแต่ละวง แสดงดังรูปที่ 2.3



รูปที่ 2.3 ภาพแสดงแนวคิดการบอกตำแหน่งแบบไตรแลทเทอเรชัน (Trilateration)

ในรูปที่ 2.3 โหนดเซนเซอร์ไร้สายอ้างอิงนั้นจะถูกสุ่มติดตั้งกันในพื้นที่ ขณะที่โหนดเซนเซอร์เป้าหมายจะเคลื่อนที่ได้อย่างอิสระ ภายในบริเวณโหนดอ้างอิง จากนั้น ตำแหน่งของโหนดเป้าหมายสามารถคำนวณได้จาก พิกัดของโหนดอ้างอิงทั้งสามตัว และระยะห่างระหว่างโหนดอ้างอิงทั้งหมด และโหนดเป้าหมาย สมการที่นำมาใช้คำนวณระยะห่างทั้งหมดนี้สามารถใช้ทฤษฎีพีทาโกรัส ซึ่งสามารถจัดรูปสมการได้ดังต่อไปนี้

$$d_{12} = (x_1 - x)^2 + (y_1 - y)^2 \quad (2.3)$$

$$d_{22} = (x_2 - x)^2 + (y_2 - y)^2 \quad (2.4)$$

$$d_{23} = (x_3 - x)^2 + (y_3 - y)^2 \quad (2.5)$$

จัดรูปสมการใหม่เพื่อหาค่าของตัวแปร x และ y ได้ดังต่อไปนี้

$$x = \frac{AY_{32} + BY_{13} + CY_{21}}{2(x_1Y_{32} + x_2Y_{13} + x_3Y_{21})} \quad (2.6)$$

$$y = \frac{AX_{32} + BX_{13} + CX_{21}}{2(y_1X_{32} + y_2X_{13} + y_3X_{21})} \quad (2.7)$$

โดยที่

$$A = x_1^2 - y_1^2 - d_1^2 \quad (2.8)$$

$$B = x_2^2 + y_2^2 - d_2^2 \quad (2.9)$$

$$C = x_3^2 + y_3^2 - d_3^2 \quad (2.10)$$

และ

$$X_{32} = (X_3 - X_2) \quad (2.11)$$

$$X_{13} = (X_1 - X_3) \quad (2.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X_{21} = (X_2 - X_1) \quad (2.13)$$

$$Y_{32} = (Y_3 - Y_2) \quad (2.14)$$

$$Y_{13} = (Y_1 - Y_3) \quad (2.15)$$

$$Y_{21} = (Y_2 - Y_1) \quad (2.16)$$

การบอกตำแหน่งด้วยสมการไตรแลทเตอร์เช่น นั้นสะดวกกว่าวิธีอื่น ๆ อย่างมาก เนื่องจากว่า ระยะห่างทั้งหมดสามารถคำนวณได้จากวิธีการบอกระยะห่าง (Ranging) และพิกัดของโหนดอ้างอิงต่าง ๆ นั้นก็สามารถระบุและบันทึกไว้ในฐานข้อมูลก่อนจะนำมาใช้คำนวณจริง สำหรับในเครือข่ายขนาดใหญ่ บางเซนเซอร์โหนดอาจจะมีการติดตั้งอุปกรณ์บอกตำแหน่งอย่างจีพีเอสเอาไว้กับตัว ดังนั้น โหนดอื่น ๆ ใกล้เคียงก็อาจจะสามารถรับรู้ตำแหน่งที่แท้จริงได้ จากข้อมูลของเซนเซอร์โหนดที่ติดตั้งจีพีเอสมีความเป็นไปได้ที่เครือข่ายจะมีคุณสมบัติในการเข้าถึงข้อมูลได้ 3 แบบด้วยกันดังนี้

1. โหนดเซนเซอร์ไร้สายสามารถเข้าถึงพิกัดตำแหน่งจีพีเอสได้ทั้งสามโหนด
2. โหนดเซนเซอร์ไร้สายสามารถเข้าถึงพิกัดตำแหน่งจีพีเอสได้เพียง 1 หรือ 2 โหนดเท่านั้น
3. โหนดเซนเซอร์ไร้สายไม่สามารถเข้าถึงอุปกรณ์จีพีเอสได้เลย

ในการใช้เทคนิคไตรแลทเตอร์เช่น ควรจะมีโหนดเซนเซอร์อ้างอิงอย่างน้อย 3 โหนดด้วยกัน กรณีที่ 2 กับ 3 อาจจะไม่เหมาะกับการคำนวณด้วยวิธีนี้ ด้วยเหตุนี้เองแนวคิดการบอกตำแหน่งด้วยวิธีอะตอมิก แอนด์อิตเอเรทีฟ มัลติแลทเตอร์เช่น (Atomic and Iterative Multilateration) นั้นจึงถูกพัฒนาและนำมาปรับใช้กับเครือข่ายขนาดใหญ่ ซึ่งเป็นการให้โหนดที่สามารถเข้าถึงข้อมูลจากจีพีเอสเป็นตัวหลัก ในการกระจายข้อมูลตำแหน่งให้กับโหนดอื่น ๆ ในเครือข่าย ในลักษณะการแลกเปลี่ยนข้อมูลกันเป็นทอด ๆ หรือในกรณีที่เลวร้ายที่สุดที่ไม่มีโหนดใดสามารถเข้าถึงข้อมูลบอกตำแหน่งจากจีพีเอสได้เลย ดังกรณีที่ 3 เทคนิคคอลแลบอเรทีฟมัลติแลทเตอร์เช่น (Collaborative Multilateration) ถูกพัฒนาและนำมาปรับใช้ ช่วยให้โหนดอ้างอิงแต่ละตัวสามารถระบุตำแหน่งกันเองได้ ผ่านการแลกเปลี่ยนข้อมูลซึ่งกันและกันระหว่างโหนดอ้างอิงสองโหนด ทำให้รู้ระยะห่างระหว่างกันส่งผลให้บอกตำแหน่งภายในปริมาตรเดียวกันได้

อย่างไรก็ตาม ข้อเสียของเทคนิคการบอกตำแหน่งแบบอ้างอิงระยะทางนี้ คือ ความแรงสัญญาณนั้นถูกรบกวนจากปัจจัยแวดล้อมอื่น ๆ ได้ง่ายโดยเฉพาะกับการใช้อุปกรณ์สื่อสารไร้สายภายในอาคารซึ่งส่งผลให้การบอกระยะแปรปรวน ทำให้ผลลัพธ์การคำนวณหาพิกัดตำแหน่งคลาดเคลื่อนไปจากความเป็นจริงอยู่บ่อยครั้ง

2.2.2.1.3 วิธีการบอกระยะห่าง (RSS Ranging)

ความแรงของกำลังส่งที่ได้รับมาจากสัญญาณนั้นสามารถนำมาใช้ในการหา ระยะทาง เนื่องจากคลื่นแม่เหล็กไฟฟ้านั้นมีความสัมพันธ์เป็นส่วนกลับยกกำลังสองของระหว่างกำลังส่งที่รับได้และระยะทางแสดงดังสมการที่ 2.17

$$P_r \propto \frac{1}{d^2} \quad (2.17)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ P_r นั้นคือกำลังส่งที่รับได้ของอุปกรณ์ทรานสมิตเตอร์ (Transmitter) ณ ระยะห่าง d สมการนี้ ได้ระบุไว้อย่างชัดเจนว่าระยะที่สัญญาณเดินทางนั้นหาจากความแตกต่างระหว่าง กำลังส่งสัญญาณ และกำลังส่งที่รับได้หรือเรียกอีกอย่างหนึ่งว่าอัตราสูญเสียตามระยะทาง (Path loss) ในทางปฏิบัติจริง การเพิ่มขึ้นของอัตราสูญเสียจะขึ้นอยู่กับ การเพิ่มขึ้นของระยะทางซึ่งอาจจะมีพฤติกรรมที่แตกต่างกันไปตามสภาพแวดล้อมที่ไม่เหมือนกัน นำมาซึ่งการคำนวณหาคุณลักษณะของสิ่งแวดล้อม (Environmental Characterization) ด้วยการใช้ตัวแปรอัตราสูญเสียตามระยะทางแบบเอกซ์โปเนนเชียล (Path loss exponent n) ดังสมการต่อไปนี้

$$P_r = \frac{P_{(d_0)}}{(d/d_0)^n} \quad (2.18)$$

โดยที่ $P_{(d_0)}$ คือกำลังส่งที่รับได้ วัดจากระยะ d_0 โดยทั่วไปแล้ว d_0 จะเป็นค่าคงที่เท่ากับ 1 เมตรตัวแปรอัตราสูญเสียตามระยะทางแบบเอกซ์โปเนนเชียล n ในสมการนั้นเป็นอีกหนึ่งปัจจัยที่สำคัญต่อการบ่งบอกถึงคุณลักษณะของสภาพแวดล้อม ถ้าหากการเพิ่มขึ้นของอัตราสูญเสียนั้นสูงขึ้นอย่างรุนแรงเมื่อระยะทางเพิ่มขึ้น ค่า n ก็จะมีค่าสูงขึ้นตาม

เพื่อให้การวัดระยะด้วยความแรงสัญญาณนั้นมีประสิทธิภาพ เราควรแยกแยะความแตกต่างระหว่างการบอกตำแหน่งภายในอาคารและภายนอกอาคาร โดยอ้างอิงจากลักษณะการออกแบบ การประยุกต์ใช้และการติดตั้งจริงของระบบซึ่งทั้งหมดทั้งหมดแล้ว เราสามารถจำแนกคุณลักษณะออกมาได้ดังตารางต่อไปนี้

ตารางที่ 2.1 ตารางเปรียบเทียบคุณลักษณะของการบอกตำแหน่งภายในอาคารและกลางแจ้ง

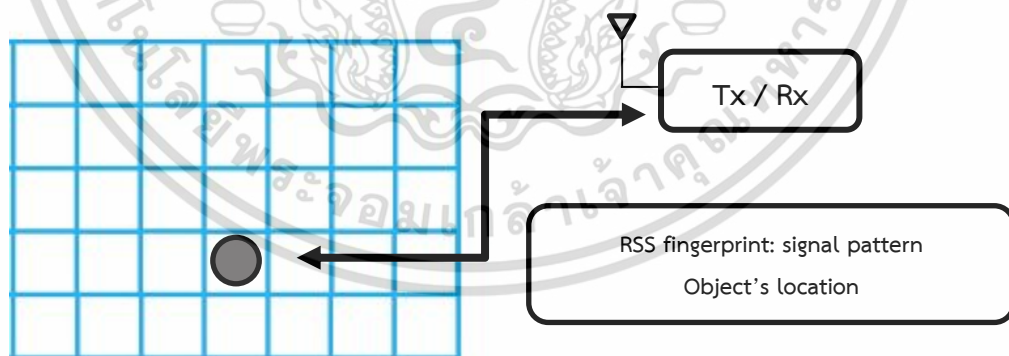
	กลางแจ้ง	ภายในอาคาร
Path loss model	เชิงเส้น	ได้รับผลกระทบ จากการเกิด Multipath และ Shadowing
ความถูกต้อง	แม่นยำสูง แต่ไม่มีความจำเป็นมาก เนื่องจากว่าเป็นพื้นที่เปิด	ความแม่นยำน้อย แต่มีความจำเป็นมาก เนื่องจากพื้นที่ปิดค่อนข้างแคบ
พื้นที่	กว้าง และไม่จำกัดขอบเขต	พื้นที่แคบ และส่วนมากพื้นที่ที่มีรูปทรง เป็นสี่เหลี่ยม
การติดตั้ง	ตำแหน่งที่ใช้ติดตั้งอุปกรณ์เป็นแบบ สุ่ม และ Ad hoc	สามารถวางแผนรูปแบบการจัดวาง อุปกรณ์ได้
กำลังส่งสัญญาณ	กำลังส่งมีได้ตั้งแต่ Link Quality Indicator ไปจนถึง Maximum	ปรับเปลี่ยนได้เพื่อหลีกเลี่ยงการรบกวน กันของสัญญาณ
ความสูงของ โหนดอ้างอิง	ติดตั้งในระดับพื้นดิน	ติดตั้งบนฝ้า หรือเพดาน
สเกลแผนที่	ใช้ได้กับพื้นที่ทั่วไป	พื้นที่ขอบเขตใดขอบเขตหนึ่งเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางด้านบนโมเดลอัตราสูญเสียตามระยะทางคือโมเดลการกระจายสัญญาณวิทยุซึ่งถูกใช้ในการจำลองการลดทอนของสัญญาณที่เคลื่อนที่ไปในตัวกลางต่าง ๆ ในสภาพแวดล้อมใด ๆ เป็นปัจจัยหลักอย่างหนึ่งซึ่งส่งผลต่อการคำนวณหาระยะห่างระหว่างเสาสัญญาณ

2.2.2.2 การบอกตำแหน่งแบบไม่อ้างอิงระยะทาง (Range-free)

การบอกตำแหน่งแบบไม่อ้างอิงระยะทาง [3], [4], [5] เป็นเทคนิคอีกประเภทหนึ่งที่ได้รับคามนิยมอย่างมาก โดยเฉพาะกับการบอกตำแหน่งภายในอาคารด้วยความแรงสัญญาณจากเครือข่ายโทรศัพท์มือถือไร้สาย สาเหตุจากสัญญาณที่เดินทางภายในอาคารนั้นถูกรบกวนได้ง่ายจากสิ่งปลูกสร้างต่าง ๆ สิ่งของภายในอาคาร รวมไปถึงตัวผู้อยู่อาศัยด้วย ทำให้กำลังส่งของสัญญาณนั้นถูกลดทอนไป การใช้ข้อมูลดังกล่าวมาทำการหาระยะทางจึงเป็นเรื่องยากที่จะได้คำตอบที่ถูกต้อง ดังนั้นวิธีการบอกตำแหน่งแบบไม่อ้างอิงระยะทางจึงอาศัยการเก็บบันทึกคุณลักษณะของความแรงสัญญาณภายในพื้นที่ขณะใช้งานจริงมาทำการหารูปแบบของข้อมูล (pattern recognition) จากนั้นจึงนำความแรงสัญญาณที่วัดได้ขณะใช้งานจริงมาเปรียบเทียบกับฐานข้อมูลที่ได้บันทึกไว้ว่าใกล้เคียงกับชุดข้อมูลของตำแหน่งใดในพื้นที่ ด้วยลักษณะของผลลัพธ์ในการบอกตำแหน่งที่ไม่ได้เป็นค่าปริมาณทางตัวเลข ทำให้วิธีนี้สามารถลดความไม่แน่นอนในการบอกตำแหน่งที่เกิดจากการหาระยะทางด้วยความแรงสัญญาณลงไปได้ แต่อย่างไรก็ตาม วิธีนี้มีข้อเสียในแง่ของฐานข้อมูลที่ต้องมีขนาดใหญ่เพียงพอสำหรับรองรับชุดข้อมูลปริมาณมาก กระบวนการบอกตำแหน่งที่ใช้เวลานานกว่าจะได้ผลลัพธ์ เนื่องจากต้องมีขั้นตอนการเก็บข้อมูลดิบ ขั้นตอนการเรียนรู้รูปแบบข้อมูล อีกทั้งในกรณีที่มีการเปลี่ยนแปลงขนาดใหญ่กับพื้นที่เป้าหมาย ตัวระบบอาจจะต้องเริ่มกระบวนการเก็บข้อมูลใหม่เพื่ออัปเดตข้อมูลให้สอดคล้องกับสภาพแวดล้อมที่เปลี่ยนไป



รูปที่ 2.4 แสดงการบอกตำแหน่งภายในอาคารด้วยเทคนิคไม่อ้างอิงระยะด้วยการรู้จำลายนิ้วมือของข้อมูล (Fingerprinting technique)

2.2.2.3 การบอกตำแหน่งแบบอ้างอิงพื้นที่ (Area-based)

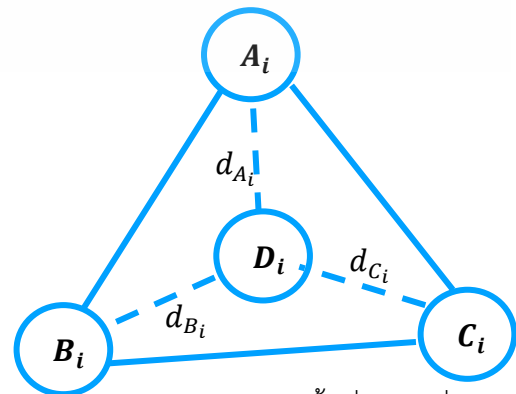
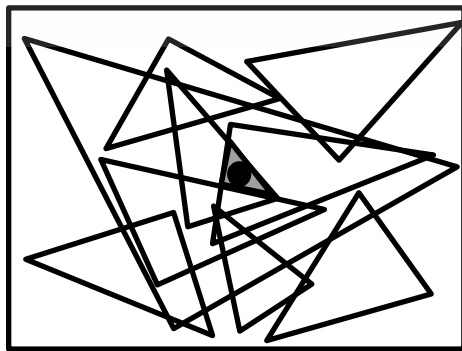
เป็นเทคนิคที่ใช้กันโดยทั่วไปในระบบการบอกตำแหน่งเป้าหมายในเครือข่ายเซนเซอร์ไร้สาย เนื่องจากบางกรณี ธรรมชาติของข้อมูลที่ได้ทำให้ไม่สามารถบอกตำแหน่งเป้าหมายในรูปของพิกัดอย่างชัดเจน แต่ระบบสามารถรับรู้ตำแหน่งของเซนเซอร์ไร้สายโดยรอบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้น เราจึงประมวลผลหาตำแหน่งของเป้าหมายในลักษณะของการประมาณพื้นที่ ว่าโหนดเป้าหมายจะอยู่ในพื้นที่ปิดสมมติใด วิธีนี้อาจจะไม่เหมาะสมสำหรับการติดตามตำแหน่งของเป้าหมายอย่างละเอียด ในทางกลับกันวิธีนี้สามารถให้ผลลัพธ์ที่ดีสำหรับเครือข่ายเซนเซอร์สื่อสารไร้สายขนาดใหญ่ วิธีที่จัดอยู่ในหมวดหมู่การบอกตำแหน่งแบบอ้างอิงพื้นที่ที่มีดังต่อไปนี้

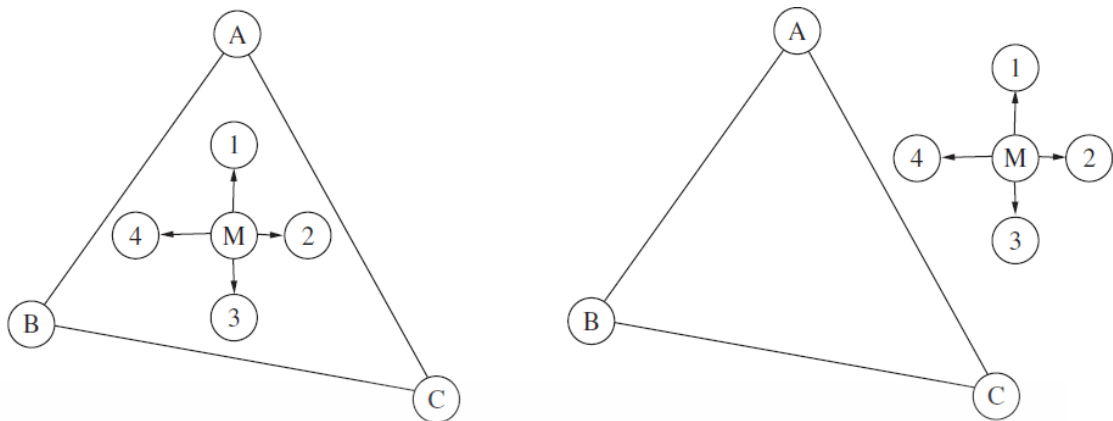
2.2.2.3.1 วิธีการประมาณตำแหน่งจุดในพื้นที่สามเหลี่ยม หรือเอพีไอที (Approximate Point-In-Triangle: APIT)

เทคนิคการประมาณตำแหน่งจุดในพื้นที่สามเหลี่ยม [6], [7] นี้ถูกจัดอยู่ในหมวดหมู่การบอกตำแหน่งแบบอ้างอิงพื้นที่และไม่อ้างอิงระยะทาง (Area-based Range-free) โดยมีหลักการบอกตำแหน่งที่ขึ้นอยู่กับโหนดอ้างอิง (Anchor node) ที่มีความสามารถในการเข้าถึงข้อมูลตำแหน่งของตัวเอง การประกอบกันของโหนดอ้างอิง 3 โหนดจะสร้างพื้นที่สามเหลี่ยมปิดใด ๆ และการตรวจสอบสถานะของโหนดเป้าหมายว่าอยู่ในหรือภายนอกพื้นที่สามเหลี่ยมปิดนั้นจะทำให้โหนดเป้าหมายสามารถจำกัดวงแคบของตำแหน่งที่ตนเองควรจะอยู่ได้ ซึ่งขั้นตอนสำคัญของเทคนิคเอพีไอที นี้ก็คือการทำกระบวนการประมาณตำแหน่งจุดในพื้นที่สามเหลี่ยม (Point In Triangulation: PIT) เพื่อให้โหนดสามารถหาตำแหน่งในพื้นที่สามเหลี่ยมปิดได้ หลังจากที่โหนดเป้าหมายได้รับข้อมูลตำแหน่งของตนจากโหนดอ้างอิงต่าง ๆ มันจะทำการหาตำแหน่งของสามเหลี่ยมที่เป็นไปได้ โหนดเป้าหมายจะถูกจัดว่าอยู่ภายนอกสามเหลี่ยม ABC (ที่สร้างขึ้นจากโหนดอ้างอิง A, B, และ C) ถ้าหากว่าทิศทางการเคลื่อนที่ของโหนดเป้าหมายนั้นออกห่างหรือเข้าใกล้ โหนด A, B, และ C ในเวลาเดียวกัน มิเช่นนั้นแล้ว โหนดเป้าหมายจะถูกกำหนดไว้ว่าอยู่ภายในสามเหลี่ยม ABC แนวคิดนี้สามารถแสดงให้เห็นเป็นรูปธรรมได้ดังภาพประกอบที่ 2.5 อย่างไรก็ตาม การใช้เทคนิคเอพีไอที ได้อย่างถูกต้องแม่นยำนั้นอาจจะไม่สามารถเป็นจริงได้ในทางปฏิบัติ เนื่องจากในความเป็นจริงแล้วโหนดเป้าหมายอาจจะสามารถเคลื่อนที่ไปในทิศทางใด ๆ ก็ได้ เทคนิคเอพีไอทีสามารถใช้กับเครือข่ายที่มีความหนาแน่นของโหนดมาก ซึ่งเทคนิคนี้จะมีประโยชน์มากเมื่อโหนดข้างเคียงอื่น ๆ สามารถรับรู้ถึงการเคลื่อนที่ของโหนดเป้าหมาย และแลกเปลี่ยนข้อมูลซึ่งกันและกันได้ ยกตัวอย่างเช่น ความแรงสัญญาณระหว่างโหนดเป้าหมายและโหนดอ้างอิงนั้นสามารถใช้ในการบอกระยะห่างว่าโหนดเป้าหมายอยู่ใกล้กับโหนดอ้างอิงใด



รูปที่ 2.5 การบอกตำแหน่งเป้าหมายด้วยวิธีการประมาณตำแหน่งจุดในพื้นที่สามเหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิใช่เพื่อเผยแพร่ในเชิงพาณิชย์ การค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 กรณีตัวอย่างในการบอกตำแหน่งเป้าหมายด้วยวิธีการประมาณตำแหน่งจุดในพื้นที่สามเหลี่ยม

จากรูปตัวอย่างที่ 2.6 รูปด้านซ้าย จะเห็นว่า โหนดเป้าหมายถูกล้อมรอบด้วยโหนดใกล้เคียง 4 ตัวเป็นอย่างน้อย ซึ่งไม่มีจุดใดเลยที่โหนดเป้าหมายจะสามารถเข้าใกล้ หรือออกห่างได้พร้อม ๆ กันอย่างน้อย 3 ตำแหน่งอ้างอิง ดังนั้นเราจึงสามารถสรุปได้ว่า โหนดเป้าหมายนั้นอยู่ภายในสามเหลี่ยม ABC ขณะที่รูปด้านขวา จะแสดงให้เห็นว่าโหนดใกล้เคียงหมายเลข 4 นั้นจะมีระยะใกล้กับโหนดอ้างอิงทั้ง 3 มากที่สุด อีกทั้งโหนดหมายเลข 2 ก็จะมีระยะห่างที่ไกลออกไปจากโหนดอ้างอิงยิ่งกว่าโหนดเป้าหมายอีก ดังนั้นจึงสามารถสรุปได้ว่า โหนดเป้าหมายอยู่นอกสามเหลี่ยม ABC แต่อย่างไรก็ตามเทคนิคการบอกตำแหน่งแบบเอพิไอที่นั้นอาจจะไม่ได้ให้ประสิทธิภาพในการบอกตำแหน่งดีเท่าที่ควร เมื่อโหนดเป้าหมายอยู่นิ่งอยู่กับที่ในสนาม ซึ่งจะทำให้การหาเวกเตอร์ในการเคลื่อนที่ของเป้าหมาย ว่าเข้าหาหรือออกห่างจากโหนดอ้างอิงใด ๆ ในสามเหลี่ยมนั้นทำได้ยากขึ้น

2.2.2.3.2 แผนภูมิโวโรนอย (Voronoi diagram)

เทคนิคแผนภูมิโวโรนอย [8], [9] คือการแบ่งพื้นที่รวมของบริเวณที่เราต้องการระบุตำแหน่งเป้าหมายออกเป็นพื้นที่ย่อย ๆ ขนาดเล็ก เพื่อลดความไม่แน่นอนของข้อมูลและลดความซับซ้อนของกระบวนการระบุตำแหน่ง ทำให้โหนดเป้าหมายตรวจสอบได้ว่าอยู่ในพื้นที่ปิดใดจากการวัดระยะห่างกับโหนดอ้างอิงที่ใกล้ที่สุด โดยหลักเกณฑ์การแบ่งนั้นได้มาจากการลากเส้นตั้งฉากสมมุติของโหนดอ้างอิงแต่ละตัวด้วยตัวเอง เส้นแบ่งขอบเขตนี้จะก่อให้เกิดเป็นพื้นที่ปิดย่อยหลาย ๆ ส่วนเรียกว่า ขอบเขตพื้นที่โวโรนอย ซึ่งสมการสร้างขอบเขตพื้นที่โวโรนอยสามารถแสดงได้ดังต่อไปนี้

$$V(P_i) = \left\{ \frac{X \in V(P_i) \mid d(x, P_i) \leq d(x, P_j)}{j = 1, 2, \dots, n, j \neq i} \right. \quad (2.19)$$

โดยให้ x เป็นจุดใด ๆ ในระนาบ และ $d(x, P_i)$ คือค่าระยะทางยูคลิด (Euclidean distance) ระหว่าง x และ จุด P_i ในเครือข่ายเซนเซอร์ไร้สาย ความแรงสัญญาณระหว่างโหนดนั้นเป็นส่วน

กลับกันของค่ากำลังสองของระยะทาง ดังนั้นเราจึงจะสามารถอธิบายขั้นตอนการตำแหน่งของโหนดเป้าหมายว่าอยู่ในขอบเขตไวโรนอยใด ๆ ได้ดังต่อไปนี้

- ให้ P_n แทนตำแหน่งใด ๆ ของโหนดอ้างอิงในเครือข่ายเซนเซอร์ไร้สายที่ติดตั้งอยู่ในสนาม และ S คือตำแหน่งของโหนดเป้าหมาย โดย n คือจำนวนของโหนดอ้างอิงที่มีในเครือข่าย
- โหนดอ้างอิง P_n ติดต่อสื่อสารกับโหนดเป้าหมาย S ทำให้โหนด S สามารถรับรู้ค่าความแรงสัญญาณ RSSI ของแต่ละโหนดโดยจะสามารถเรียงลำดับจากมากไปน้อยได้ดังสมการต่อไปนี้

$$RSSI_{P_1} > RSSI_{P_2} > \dots > RSSI_{P_j}$$
- จากคุณสมบัติของแผนภูมิไวโรนอย และเงื่อนไขการอ่านค่าความแรงสัญญาณจากข้อ B ทำให้โหนดเป้าหมาย S สามารถระบุขอบเขตไวโรนอยที่ตนเองอยู่ได้ว่า อยู่ในพื้นที่ของโหนดอ้างอิง P_1 จากนั้นระบบจะทำการคำนวณขอบเขตไวโรนอยของจุด P_1 และทำเพิ่มค่าน้ำหนัก (weight) ของความแรงสัญญาณ $RSSI_{P_1}$ ให้กับทุก ๆ โหนดในเครือข่าย
- นำโหนด P_1 ออก จากนั้นคำนวณขอบเขตไวโรนอยของจุด P_2 และ ให้ค่าน้ำหนัก (weight) $RSSI_{P_2}$ ด้วยวิธีนี้ทำให้เราสามารถหาตำแหน่งของโหนดอ้างอิงทั้งหมดในขอบเขตไวโรนอยได้
- ท้ายที่สุด เราก็จะสามารถหาขอบเขตพื้นที่ที่มีค่าน้ำหนักสูงที่สุดและพิกัดตำแหน่งของโหนดเป้าหมาย S ได้

อย่างไรก็ตาม มีการพัฒนาอัลกอริทึมชนิดนี้ให้มีความยืดหยุ่นและประสิทธิภาพที่มากขึ้นไปอีก โดยใช้หลักการที่ว่าโหนดที่มีความแรงสัญญาณสูงจะมีระยะห่างระหว่างโหนดที่สั้นลง ดังนั้นเราจึงทำการปรับขอบเขตพื้นที่ไวโรนอยให้เหมาะสมได้จากความสัมพันธ์ของความแรงสัญญาณระหว่างโหนด ซึ่งทำให้ช่วยลดความซับซ้อนในการคำนวณ สามารถระบุตำแหน่งของเป้าหมายได้ถูกต้องแม่นยำมากขึ้น เราเรียกอัลกอริทึมแบบปรับปรุงนี้ว่าแผนภูมิไวโรนอยแบบกำหนดค่าน้ำหนัก

2.2.2.3.3 วิธีการบอกตำแหน่งเป้าหมายด้วยความน่าจะเป็นของข้อมูล (Heuristic Information and Probability)

อัลกอริทึมในรูปแบบนี้สามารถแก้ปัญหาในกรณีที่ระบบไม่สามารถคำนวณหาคำตอบที่ถูกต้องแน่ชัดได้ โดยมันจะทำการคำนวณความน่าจะเป็นของคำตอบที่มีความเป็นไปได้มากที่สุดแทน ซึ่งแนวคิดนี้อาจจะเป็นระบบที่เกิดจากการสร้างและผสมผสานกันระหว่างอัลกอริทึมหลาย ๆ วิธี เช่นการใช้เทคนิคการรู้จำลายนิ้วมือ (Fingerprinting) ผสมกับการใช้ แมชชีนเลิร์นนิงแบบมีผู้ฝึกสอน (Supervised Machine learning) เข้ามาทำหน้าที่เป็นตัวจำแนกประเภทข้อมูล (classifier) ที่เกิดจากการเก็บบันทึกหลาย ๆ ครั้งเพื่อสร้างผลลัพธ์ในการบอกตำแหน่ง วิธีนี้จะทำให้กระบวนการคำนวณมีความยืดหยุ่น ให้ประสิทธิภาพในการบอกตำแหน่งที่ดีกว่า เมื่อเปรียบเทียบกับอัลกอริทึมที่ให้คำตอบเป็นจำนวนตัวเลขที่มีค่าทางคณิตศาสตร์ แต่ถึงกระนั้น กระบวนการคิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนวณจำเป็นต้องอาศัยทรัพยากรและกำลังประมวลผลของอุปกรณ์ค่อนข้างมาก เพื่อให้ได้มาซึ่งผลลัพธ์จากการคำนวณที่ใกล้เคียงกับความเป็นจริงมากที่สุด

ตัวอย่างงานวิจัยที่มีการใช้เทคนิคฮิวริสติกอินฟอร์เมชัน เช่น [11] ใช้โครงข่ายประสาทเทียม (Artificial Neural Network : ANN) สำหรับการคำนวณหาตำแหน่งแบบอะแดปทีฟ (Adaptive position calculation) เนื่องจากรูปแบบข้อมูลความแรงสัญญาณนั้นมีความซับซ้อนเกินกว่าที่จะคำนวณได้จากการหาความสัมพันธ์ทางคณิตศาสตร์หรือสมการตามปกติ [12] ใช้โครงข่ายประสาทเทียมแบบกราเดียนเดสเซนส์ป้อนกลับแบบกระจายตัว (Gradient descent backpropagation) เข้ามาทำหน้าที่เป็นอัลกอริทึมในการเรียนรู้รูปแบบข้อมูลแบบมีผู้ฝึกสอน เพื่อที่จะจำแนกข้อมูลความแรงสัญญาณ ณ ตำแหน่งต่าง ๆ ภายในพื้นที่ [13] ใช้โครงข่ายประสาทเทียมแบบ Particle Swarm Optimization with Backpropagation Algorithm (PSO-BP) ซึ่งเป็นอัลกอริทึมประเภทการสุ่มค้นหา (Random search) ข้อดีของอัลกอริทึมประเภทนี้คือความทนทานต่อการเกิดค่าผิดพลาด อัลกอริทึมแมชชีนเลิร์นนิงอีกประเภทหนึ่งที่เรียกว่าซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) นั้นมีการใช้งานในงานวิจัย [14] ซึ่งแนวคิดพื้นฐานคือการแบ่งรูปแบบข้อมูลออกเป็นคลาสย่อยคลาสต่าง ๆ คล้ายกันกับโครงข่ายประสาทเทียม แต่วิธีซัพพอร์ตเวกเตอร์แมชชีนนี้จะเหมาะกับการใช้งานกับกลุ่มข้อมูลตัวอย่างปริมาณน้อย ในขณะที่ [15] ใช้อัลกอริทึมการเรียนรู้แบบแผนภูมิต้นไม้ (Decision tree) ในการเลือกตัวเลือกความแรงสัญญาณที่เหมาะสมกับสถานการณ์ใด ๆ เพื่อให้ความถูกต้องในการบอกตำแหน่งที่ดีที่สุด ยิ่งไปกว่านั้น วิธีการบอกตำแหน่งเป้าหมายด้วยความน่าจะเป็นนั้นยังรวมไปถึงการบอกตำแหน่งแบบผสมผสาน (Hybrid localization) ซึ่งเป็นการผสมผสานกันระหว่างการคำนวณความน่าจะเป็นของเหตุการณ์, อัลกอริทึมการชดเชยค่าผิดพลาด, และการใช้เซนเซอร์หลาย ๆ ชนิดร่วมกันบอกตำแหน่งดังที่ได้กล่าวถึงในงานวิจัยที่ [10], [11], [13], [14] ซึ่งแนวคิดหลัก ๆ ของเทคนิคนี้คือ หลีกเลี่ยงข้อบกพร่องในการบอกตำแหน่งของแต่ละเทคนิค ยกตัวอย่างเช่น [29] ใช้เทคนิคการบอกตำแหน่งแบบเดอลอนีย์ไตรแองเกิลชันผสมผสานกับเทคนิคฮิวริสติกอินฟอร์เมชัน เพื่อหลีกเลี่ยงความไม่แน่นอนของความแรงสัญญาณที่ใช้ในการบอกตำแหน่งอันเนื่องมาจากความไม่แม่นยำของอุปกรณ์ คุณลักษณะและผลกระทบต่าง ๆ ที่อาจเกิดรบกวนการเดินทางของสัญญาณ ซึ่งจากผลการทดลองในงานวิจัยดังกล่าวได้ระบุไว้ว่า การผสมผสานของทั้งสองวิธีนั้นสามารถให้ค่าผิดพลาดในการบอกตำแหน่งเป้าหมายน้อยกว่า 1.5 เมตร อีกทั้งยังไม่มี การสะสมค่าผิดพลาดในแต่ละขอบเขตพื้นที่สำหรับระบุตำแหน่งเป้าหมายอีกด้วย

2.3 วิธีการระบุตำแหน่งพร้อมกับการสร้างแผนที่ (Simultaneous Localization and Mapping: SLAM)

จากแนวคิดการบอกตำแหน่งหลาย ๆ วิธี สิ่งหนึ่งที่เหมือนกันอย่างหนึ่งคือ ระบบจำเป็นต้องรับรู้ข้อมูลหรือแผนที่ของสภาพแวดล้อมนั้น ๆ ก่อนที่จะทำการประมวลผลใด ๆ ซึ่งบางครั้ง หุ่นยนต์เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตโนมัติก็จำเป็นต้องมีความสามารถในการสร้างแผนที่ของสิ่งแวดล้อมนั้น ๆ ได้เองด้วย หรืออาจจะสรุปได้ว่า การสร้างแผนที่นั้นจำเป็นต้องมีการบอกตำแหน่งเข้ามาเกี่ยวข้องควบคู่กันไป กระบวนการนี้เราจะเรียกว่าการระบุตำแหน่งพร้อมกับการสร้างแผนที่หรือ สแลม (Simultaneous Localization and Mapping: SLAM) [21]

สแลม (SLAM) สามารถนำไปปรับใช้ได้หลายวิธีและสามารถใช้กับอุปกรณ์ฮาร์ดแวร์ได้หลายประเภท ซึ่งสิ่งหนึ่งที่ขาดไม่ได้เลยคือเซนเซอร์รับรู้ค่าทางกายภาพของหุ่นยนต์ เช่น เซนเซอร์วัดระยะ เซนเซอร์วัดทิศทาง เซนเซอร์วัดความเร็วการเคลื่อนที่ เป็นต้น สแลมเป็นแนวคิดที่รวมอัลกอริทึมหลาย ๆ ประเภทเข้ามาทำงานร่วมกัน เช่น การคัดกรองเอาจุดเด่นของพื้นที่ (Landmark extraction), การประยุกต์ใช้งานกลุ่มข้อมูลที่เหมาสะสม (Data association), การคำนวณหาสถานะปัจจุบัน (State estimation), การอัปเดตสถานะปัจจุบัน (State update) และ การอัปเดตพื้นที่เป้าหมายปัจจุบัน (Landmark update) ซึ่งในแต่ละส่วนสามารถนำเทคนิคต่าง ๆ มาปรับใช้ได้เพื่อให้เกิดประสิทธิภาพของผลลัพธ์ที่ดีขึ้น

กระบวนการสแลมประกอบไปด้วยหลายขั้นตอน เป้าหมายของกระบวนการนี้คือ เพื่อใช้สภาพแวดล้อมโดยรอบเป็นตัวบอกตำแหน่งของหุ่นยนต์และอัปเดตการเปลี่ยนแปลงต่าง ๆ เนื่องจากข้อมูลทิศทางการเคลื่อนที่ของหุ่นยนต์ นั้นมีค่าคลาดเคลื่อนปะปนอยู่ เราจึงไม่สามารถอ้างอิงจากข้อมูลนี้ในการบอกตำแหน่งได้เพียงอย่างเดียว การใช้เลเซอร์สแกนเนอร์ในการตรวจจับสภาพแวดล้อมโดยรอบนั้นมีประโยชน์ในขั้นตอนการคัดกรองคุณลักษณะเด่นของสภาพแวดล้อม อีกทั้งเทคนิคตัวกรองคาลมานแบบต่อขยาย (Extended Kalman filter) นั้นเป็นหัวใจหลักของแนวคิดสแลม เนื่องจากว่ามันจะเป็นตัวชดเชยความคลาดเคลื่อนและอัปเดตข้อมูลที่หุ่นยนต์ประมวลผลเสร็จ

ขั้นตอนการทำงานของสแลมโดยสังเขปมีดังต่อไปนี้

1. หุ่นยนต์เคลื่อนที่ไปยังตำแหน่งใหม่ในสนามเกิดการเปลี่ยนแปลงของข้อมูลการเคลื่อนที่และทิศทาง
2. ตัวกรองคาลมาน (EKF) ทำการอัปเดตข้อมูล และข้อมูลพื้นที่เป้าหมาย (Landmark) ในสนามจะถูกอ่านค่าจากตัวเซนเซอร์ เช่น เลเซอร์สแกนเนอร์ เป็นต้น
3. หุ่นยนต์ทำการเปรียบเทียบพื้นที่เป้าหมายที่พบเจอกับฐานข้อมูลของตัวเองเพื่อตรวจสอบว่าพื้นที่นี้ผ่านการสำรวจมาแล้วหรือไม่
4. พื้นที่เป้าหมายที่เคยผ่านการสำรวจมาแล้ว ก็จะถูกนำไปใช้ในการอัปเดตตำแหน่งปัจจุบันของหุ่นยนต์ แต่ถ้าพื้นที่เป้าหมายที่พบ เป็นข้อมูลที่ยังไม่เคยสำรวจ ก็จะถูกส่งไปตัวกรองคาลมานใช้ในการประมวลต่อไป

อัลกอริทึมสแลมหลาย ๆ รูปแบบได้ถูกนำเสนอเรื่อยมาอย่างน่าสนใจ ทิศทางของการสร้างแผนที่ด้วยสแลมนั้นแบ่งออกเป็นหลัก ๆ ได้สามรูปแบบด้วยกันคือแบบโทโพโลจิคอล (Topological), แบบอ้างอิงตารางกริด (Grid-based) และแบบอ้างอิงลักษณะเด่นของพื้นที่ (Feature-based) ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ โทโพโลจิคอลนั้นสภาพแวดล้อมต่าง ๆ จะถูกพิจารณาในลักษณะกราฟเส้นเชื่อมโยงจุดต่าง ๆ เข้าด้วยกัน การบอกตำแหน่ง (Localization) จึงทำได้โดยการจดจำลักษณะของพื้นที่ หรือโหนดต่าง ๆ ไว้ ข้อดีของเทคนิคนี้ คือจะไม่มีค่าผิดพลาดจากการอ่านค่าของเซนเซอร์ในแผนที่ เนื่องจากหุ่นยนต์จะถูกนำทางแบบจุดต่อจุดเท่านั้น อีกทั้งเทคนิคโทโพโลจิคอลนี้ยังสามารถปรับใช้กับสภาพแวดล้อมขนาดใหญ่ได้ โดยใช้ทรัพยากรในการบันทึกข้อมูลต่าง ๆ เพียงเล็กน้อย เนื่องจากข้อมูลที่ใช้ระบุตำแหน่งต่าง ๆ มีเพียงแค่ ข้อมูลที่ใช้ระบุตัวตนของโหนดหรือพื้นที่ย่อยใด ๆ อย่างไรก็ตาม การใช้เทคนิคนี้ควบคู่กับอัลกอริทึมสแลมนั้นก็ยังคงมีข้อเสียคือ ความยากและความซับซ้อนในการจดจำลักษณะของโหนดหรือพื้นที่ย่อยใด ๆ ซึ่งอาจจะทำให้หุ่นยนต์ตัดสินใจได้ลำบากมากขึ้นว่าโหนดดังกล่าวได้เคยสำรวจผ่านมาแล้วหรือไม่

การพิจารณาแผนที่ในลักษณะตารางกริดนั้นถูกใช้กันอย่างแพร่หลายกับเทคนิคสแลมโดยที่แต่ละตารางย่อยจะสามารถระบุได้ว่าพื้นที่นั้นมีสิ่งกีดขวางอยู่หรือไม่ โดยทั่วไปแล้วเทคนิคของการใช้สแลมแบบอ้างอิงตารางกริดนั้นคือการประสานกันระหว่างตารางกริดระดับท้องถิ่น (local grid) และตารางกริดโดยรวม (global grid) โดยที่ตารางกริดโดยรวมนั้นคือแผนที่โดยรวมที่จะถูกจัดเก็บไว้ในฐานข้อมูลกลาง และแผนที่ระดับท้องถิ่นคือแผนที่ที่จะนำไปใช้ในการอัปเดตแผนที่รวม เมื่อเรานำแผนที่โดยรวมและท้องถิ่นมาพิจารณาร่วมกัน เราจะได้ตำแหน่งที่แท้จริงของหุ่นยนต์ ข้อเสียของแนวคิดนี้คือ ต้องการระบบที่มีทรัพยากรในการประมวลผลสูง และมีเนื้อที่หน่วยความจำสำหรับเก็บข้อมูลที่มากพอสมควร

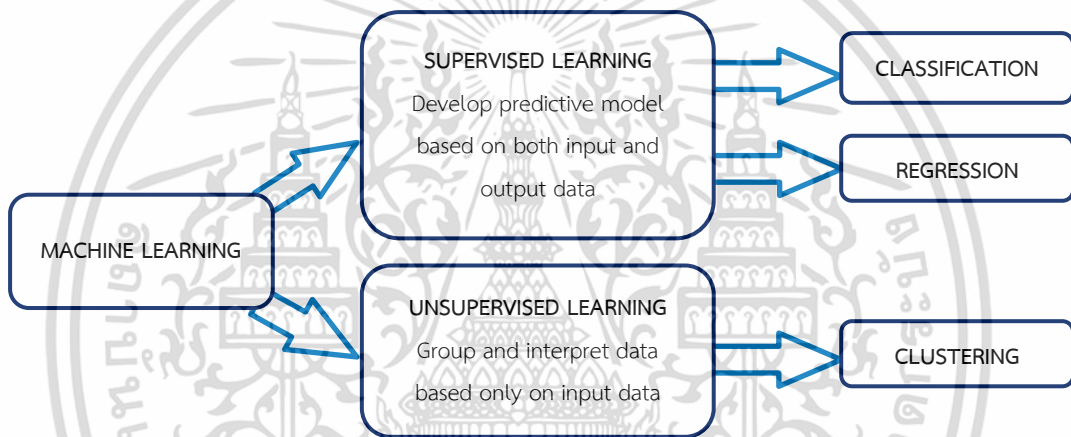
ในการใช้เทคนิคสแลมแบบอ้างอิงคุณลักษณะของพื้นที่ (feature-based) นั้น ลักษณะของสภาพแวดล้อมที่อ่านค่าได้จากอุปกรณ์เซนเซอร์จะเก็บไว้ในรูปของเวกเตอร์หนึ่ง ๆ ทั้งตำแหน่งของทิศทางของหุ่นยนต์ที่กระทำกับแผนที่ที่จะถูกอัปเดตเมื่อคุณลักษณะของแผนที่นั้นเปลี่ยนแปลงไปหรือถูกอ่านค่าใหม่อีกครั้ง เมื่อหุ่นยนต์เคลื่อนที่เข้าไปยังพื้นที่ใหม่ เวกเตอร์สถานะก็จะทำการเก็บข้อมูลคุณลักษณะต่าง ๆ ของพื้นที่นั้น ๆ เมื่อหุ่นยนต์เปลี่ยนตำแหน่งของตัวเอง ข้อมูลตำแหน่งและทิศทางของหุ่นยนต์จะถูกอัปเดตและเก็บไว้ในหน่วยความจำ จากนั้นจึงอ่านค่าคุณลักษณะของแผนที่โดยรวม ขั้นตอนการสร้างแผนที่จะประกอบไปด้วย การคาดการณ์และการอัปเดตข้อมูล ในระหว่างการคาดการณ์ สัญญาณควบคุมการเคลื่อนที่หรือทิศทางของหุ่นยนต์นั้นจะนำมาใช้ในการทำนายสถานะต่อไปที่อาจจะเกิดขึ้น และในระหว่างขั้นตอนการอัปเดตข้อมูลให้เป็นปัจจุบัน คุณลักษณะต่าง ๆ ที่ตรวจวัดได้จากเซนเซอร์นั้นจะนำมาใช้ในการอัปเดตตำแหน่งและทิศทางของหุ่นยนต์ที่กระทำต่อแผนที่ปัจจุบัน เทคนิคตัวกรองคาลมานแบบต่อยอด เข้ามามีบทบาทในการประมาณค่าเชิงเส้นก่อนที่จะทำการอัปเดตข้อมูลแผนที่ให้เป็นปัจจุบัน อย่างไรก็ตาม ในการประมาณค่า ก็ยังมีความเป็นไปได้ที่จะเกิดค่าผิดพลาดในผลลัพธ์ ซึ่งท้ายที่สุดแล้วค่าผิดพลาดในผลลัพธ์ก็อาจจะก่อให้เกิดความคลาดเคลื่อนในการทำนายตำแหน่งของหุ่นยนต์ที่ควรจะเป็น และลักษณะของสิ่งแวดล้อมโดยรอบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัลกอริทึมสแลมแบบอ้างอิงคุณลักษณะเด่นของพื้นที่นั้นถูกปรับแต่งแนวคิดเล็กน้อย เพื่อให้ได้ประสิทธิภาพที่ดีขึ้น โดยการพิจารณาเพียงแคคุณลักษณะของแผนที่ที่สอดคล้องกันเท่านั้น คุณลักษณะบางประการจะถูกคัดออกไปจากเวกเตอร์สถานะ เหลือไว้เพียงคุณลักษณะของแผนที่ที่เด่นที่สุด ที่ทำให้หุ่นยนต์สามารถจดจำรายละเอียดและช่วยในการนำทางได้ เทคนิคนี้จะทำให้สามารถลดจำนวนข้อมูลคุณลักษณะของแผนที่ลงได้มาก

แต่อย่างไรก็ตามการใช้อัลกอริทึมสแลมนั้นก็ยังคงเป็นเรื่องที่ต้องถูกพัฒนาและวิจัยต่อไป เนื่องจากอุปสรรคในการสร้างแผนที่นั้นไม่ได้เกิดจาก ข้อบกพร่องของเทคนิคสแลม เท่านั้น หากแต่อยู่ที่การแปรความหมายของค่าที่วัดได้จากอุปกรณ์ และการนำไปประยุกต์ใช้งานจริง

2.4 การเรียนรู้ของเครื่อง Machine learning



รูปที่ 2.7 แผนผังแสดงประเภทการทำงานรูปแบบต่าง ๆ ที่เป็นส่วนประกอบหลักของแนวคิดแมชชีนเลิร์นนิ่งหรือการเรียนรู้ของเครื่อง

การเรียนรู้ของเครื่อง [1], [22] นั้นกลายมาเป็นหัวข้อสำคัญหัวข้อหนึ่งในหลาย ๆ หน่วยงานหรือองค์กรพัฒนาเทคโนโลยีที่กำลังมองหาการยกระดับความสำคัญของข้อมูลเพื่อช่วยให้กิจการหรือธุรกิจนั้นเติบโตได้มากขึ้น เข้าใจถึงแนวโน้มของการเปลี่ยนแปลงสิ่งต่าง ๆ ได้มากขึ้น ด้วยโมเดลการเรียนรู้ของเครื่องที่เหมาะสมนั้นจะทำให้เราสามารถที่จะคาดการณ์ถึงความเปลี่ยนแปลงของสิ่งต่าง ๆ ได้อย่างต่อเนื่อง อีกทั้งหากโมเดลการเรียนรู้ของเครื่องได้ถูกปรับใช้กับข้อมูลใหม่ ๆ ที่ได้รับเข้ามาอย่างไม่ขาดสาย นั่นก็จะทำให้ตัวโมเดลถูกปรับปรุงประสิทธิภาพให้ทันกับข้อมูลใหม่ ๆ อยู่เสมอ ยิ่งข้อมูลที่น่ามาใช้เรียนรู้ถูกต้องเหมาะสมเท่าไร การทำนายผลของโมเดลก็จะมีประสิทธิภาพและเชื่อถือได้มากขึ้นเท่านั้น

การเรียนรู้ของเครื่องนั้นเป็นรูปแบบหนึ่งของปัญญาประดิษฐ์ที่ทำให้ตัวระบบสามารถเรียนรู้จากชุดข้อมูลที่จัดเตรียมไว้ให้ เพื่อที่จะสามารถสร้างแบบแผนการทำงานหรือประมวลผลกระบวนการต่าง ๆ แทนที่จะเขียนโปรแกรมสั่งการโดยตรง การเรียนรู้ของเครื่องนั้นประกอบไปด้วยอัลกอริทึมมากมายหลายชนิดที่มีกระบวนการเรียนรู้จากข้อมูลแบบซ้ำไปซ้ำมา เพื่อปรับปรุง พัฒนา และแปลเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายของข้อมูลที่ได้รับ เพื่อทำการทำนายผลลัพธ์ที่จะเกิดขึ้นอย่างมีเหตุผล ขณะที่อัลกอริทึมทำการเรียนรู้ชุดข้อมูลที่ใช้ในการฝึกฝน จะมีแนวโน้มที่จะสามารถทำนายผลลัพธ์ต่อไปได้อย่างแม่นยำหลังจากการฝึกฝนเสร็จสิ้น เมื่อทำการป้อนข้อมูลอินพุตใด ๆ ให้กับโมเดล ระบบจะได้ผลลัพธ์จากการประมวลผลมาซึ่งก็คือผลการทำนายของโมเดลที่ผ่านการเรียนรู้จากการฝึกฝนกับชุดข้อมูลมาแล้วในขั้นตอนนี้

การเรียนรู้ของเครื่องนั้นเป็นการสร้างโมเดลขึ้นมาเพื่อเรียนรู้ฝึกสอนด้วยข้อมูลที่ถูกป้อนให้ก่อนจะทำการประมวลผลจริง โมเดลบางประเภทสามารถที่จะทำงานแบบออนไลน์ โดยการเรียนรู้ชุดข้อมูลควบคู่ไปพร้อม ๆ กับการรับข้อมูลใหม่ในเวลาเดียวกัน แต่ในอีกรูปแบบหนึ่ง โมเดลการเรียนรู้ที่เรียกว่าโมเดลการเรียนรู้แบบออฟไลน์ (Offline machine learning model) นั้นมีขั้นตอนการเรียนรู้ที่เหมือนกัน ผ่านการเรียนรู้จากชุดข้อมูลที่ใช้สำหรับฝึกฝนเช่นกัน แต่เมื่อใช้งานจริง จะไม่สามารถปรับตัวตามข้อมูลชุดใหม่ที่ถูกป้อนไปในเวลาเดียวกันกับทำนายผลลัพธ์สำหรับคาดการณ์แนวโน้มข้างหน้าที่จะเกิดขึ้นได้อย่างโมเดลแบบออนไลน์ ยิ่งไปกว่านั้น อัลกอริทึมที่ซับซ้อนของโมเดลการเรียนรู้ สามารถที่จะแปรเปลี่ยนประยุกต์ใช้ไปตามตัวแปรหรือปัจจัยต่าง ๆ ที่ใช้ในการประมวลผล ไม่ว่าจะเป็น ข้อมูลจากเซนเซอร์ เวลา สภาพอากาศ หรือเมทริกซ์ประยุกต์ใด ๆ ตัวอย่างเช่น โมเดลสามารถดึงค่าสภาพอากาศมาทำนายได้ว่า มีแนวโน้มจะเกิดพายุหรือไม่ เพื่อที่จะทำการแจ้งเตือนเหตุฉุกเฉินได้ล่วงหน้า เป็นต้น

2.4.1 รูปแบบต่าง ๆ ของการเรียนรู้ของเครื่อง (Approaches to Machine Learning)

เทคนิคต่าง ๆ ของการเรียนรู้ของเครื่องนั้นเป็นสิ่งจำเป็นที่จะช่วยเพิ่มประสิทธิภาพของความแม่นยำในการทำนายผลลัพธ์ของตัวโมเดล มีวิธีและเทคนิคมากมายในการปรับปรุงคุณภาพของโมเดลขึ้นอยู่กับลักษณะของปัญหาหรือภารกิจ รวมไปถึงประเภทของข้อมูลที่จะต้องนำโมเดลไปประยุกต์ใช้ ซึ่งโดยหลักแล้ว รูปแบบการเรียนรู้ของเครื่องจะสามารถแบ่งหมวดหมู่ออกได้เป็น 4 กลุ่มหลัก ๆ ดังต่อไปนี้

2.4.1.1 การเรียนรู้แบบมีผู้ฝึกสอน (Supervised Learning)

การเรียนรู้ประเภทนี้โดยทั่วไปจะเริ่มต้นจากการสร้างชุดข้อมูลพร้อมทั้งเฉลยหรือผลลัพธ์ที่ถูกต้องแน่นอนและสอดคล้องกับชุดข้อมูล เทคนิคประเภทนี้เป็นการออกแบบมาเพื่อหารูปแบบโครงสร้างภายในชุดข้อมูลที่นำมาใช้ฝึกสอน ยกตัวอย่างเช่น กลุ่มรูปภาพสัตว์ชนิดต่าง ๆ จำนวนหลายล้านรูป พร้อมทั้งมีป้ายชื่อกำกับว่าแต่ละภาพเป็นสัตว์ประเภทอะไร เป็นต้น ชุดข้อมูลเหล่านี้เองที่เราจะสามารถนำมาใช้ฝึกสอนให้กับโมเดล เพื่อสร้างรูปแบบหรือโครงสร้างทางคณิตศาสตร์ที่แสดงความสัมพันธ์ระหว่างรูปภาพกับกลุ่มประเภทของสัตว์แต่ละชนิด เมื่อโมเดลสามารถที่จะเรียนรู้ความสัมพันธ์เหล่านี้ได้ ผลลัพธ์ที่โมเดลทำนายจะอยู่ในรูปแบบของหมวดหมู่ว่ารูปภาพนั้น ๆ คือรูปของสัตว์ประเภทใด โมเดลที่มีลักษณะการประมวลผลเช่นนี้ เราจะเรียกว่าการจัดหมวดหมู่ (classification) ในทางกลับกันหากโมเดลเรียนรู้โครงสร้างและความสัมพันธ์ของข้อมูลกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์ที่ฝึกสอนและทำนายผลเชิงปริมาณ ในลักษณะของค่าจำนวนจริงไม่ใช่การแบ่งกลุ่ม โมเดลที่ประมวลผลเช่นนี้เราจะเรียกว่าโมเดลการวิเคราะห์เชิงถดถอย (Regression) เราจะใช้โมเดลการเรียนรู้ประเภทนี้ในการเรียนรู้ข้อมูลจากอดีตเพื่อคาดการณ์ถึงผลลัพธ์ที่มีความเป็นไปได้ว่าจะเกิดขึ้นในอนาคตข้างหน้า

อัลกอริทึมจะถูกฝึกสอนด้วยข้อมูลที่ผ่านมาการจัดเตรียมอย่างเหมาะสม และหลังจากนั้นประสิทธิภาพของอัลกอริทึมจะถูกทดสอบด้วยข้อมูลชุดใหม่ที่ตัวโมเดลไม่เคยเรียนรู้มาก่อน โดยทั่วไปแล้วโครงสร้างของข้อมูลจะไม่สามารถแสดงออกได้อย่างเห็นชัด หากว่าจำนวนของข้อมูลมีปริมาณมาก ถ้าหากโมเดลจดจำโครงสร้างของข้อมูลจากเฉพาะกลุ่มข้อมูลที่ใช้ฝึกสอนเพียงอย่างเดียว พฤติกรรมนี้จะเรียกว่าโอเวอร์ฟิตติ้ง (Overfitting) ซึ่งก็คือการที่โมเดลถูกปรับแต่งมาให้ใช้งานกับชุดข้อมูลฝึกสอนโดยเฉพาะ แต่จะไม่มีความสามารถที่จะทำนายผลได้อย่างถูกต้องแม่นยำกับชุดข้อมูลใหม่ที่ไม่เคยพบเห็นมาก่อน เพื่อป้องกันการเกิดเหตุการณ์นี้ การทดสอบประสิทธิภาพของโมเดลจึงมีความจำเป็นอย่างมาก เพื่อให้เราสามารถเพิ่มประสิทธิภาพในการประมวลผล ความทนทานต่อการปรับใช้ข้อมูลใหม่ การเรียนรู้แบบมีผู้ฝึกสอนนี้เองสามารถนำไปประยุกต์ใช้กับหลาย ๆ กรณีได้ ตัวอย่างเช่น การวิเคราะห์ความเสี่ยง ตรวจสอบการปลอมแปลงต่าง ๆ การจดจำรูปแบบเสียงพูด

2.4.1.2 การเรียนรู้แบบไม่มีผู้ฝึกสอน (Unsupervised Learning)

การเรียนรู้แบบไม่มีผู้ฝึกสอน จะเหมาะกับภารกิจที่ต้องการปริมาณข้อมูลเป็นจำนวนมากและไม่ได้มีการจัดกลุ่มหรือระบุตัวตนไว้ล่วงหน้าแล้ว ยกตัวอย่างเช่น แอปพลิเคชันจำพวกโซเชียลมีเดียหรือแอปพลิเคชันทางการแพทย์ที่จำนวนข้อมูลที่จำเป็นสำหรับการวินิจฉัยโรคนั้นมีมากกว่าที่จะสามารถจัดกลุ่มได้หมด เป็นต้น การที่โมเดลจะเข้าใจถึงเบื้องหลังโครงสร้างของข้อมูลนั้น จำเป็นต้องอาศัยอัลกอริทึมที่สามารถจัดกลุ่ม หรือรับรู้ถึงรูปแบบการกระจายตัวของข้อมูล และรับมือกับจำนวนตัวแปรที่เกี่ยวข้องเป็นจำนวนมากได้ ยกตัวอย่างเช่น การเรียนรู้ของเครื่องที่ประยุกต์ใช้กับการตรวจจับสแปมอีเมลโดยเงื่อนไขในการตัดสินใจว่าอีเมลนี้เป็นการสแปมหรือไม่ มีข้อกำหนดที่ค่อนข้างซับซ้อนเกินกว่าที่จะเขียนโปรแกรมตรวจจับได้โดยตรง อัลกอริทึมการเรียนรู้แบบไม่มีผู้ฝึกสอนนี้ จะทำการแบ่งชุดข้อมูลออกเป็นกลุ่มตัวอย่างย่อย ๆ ตามประเภทของคุณสมบัติเด่นที่ตรวจพบได้ ด้วยเหตุนี้เองทำให้ในบางครั้งการเรียนรู้แบบไม่มีผู้ฝึกสอนถูกนำมาใช้การจัดกลุ่มประเภทข้อมูลดิบในขั้นแรกก่อนที่จะส่งต่อข้อมูลไปให้ส่วนการเรียนรู้แบบมีผู้ฝึกสอนทำการประมวลผลในขั้นตอนต่อ ๆ ไป

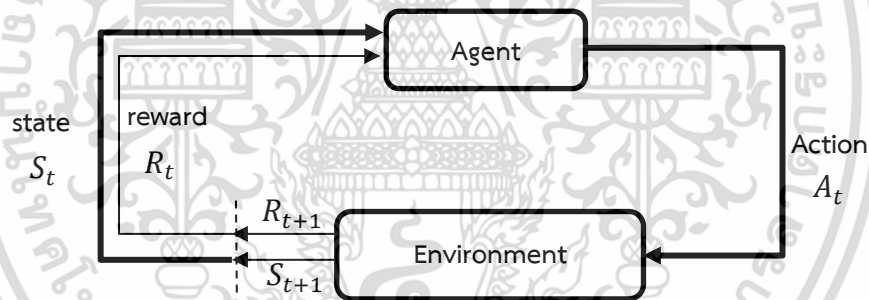
2.4.1.3 การเรียนรู้แบบเสริมกำลัง (Reinforcement Learning)

การเรียนรู้แบบเสริมกำลัง [23], [24] นี้เป็นโมเดลการเรียนรู้ที่วิเคราะห์พฤติกรรมการทำงานของระบบ โดยตัวอัลกอริทึมนี้จะรับข้อมูลป้อนกลับมาจากผลการวิเคราะห์ข้อมูลซึ่งจะสามารถนำพาคำตอบหรือแนวทางการแก้ปัญหาที่ดีที่สุดให้แก่ผู้ใช้ระบบ การเรียนรู้แบบเสริมกำลังนี้จะต่างจากการเรียนรู้แบบมีผู้ฝึกสอนเนื่องจากว่า ตัวระบบจะไม่มีถูกฝึกฝนจากกลุ่มข้อมูลตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่อย่างไรก็ตาม หากแต่ระบบจะทำการเรียนรู้จากการลองผิดลองถูกในแต่ละแนวทางการแก้ปัญหา ดังนั้น กระบวนการตัดสินใจในแต่ละครั้งของโมเดลจะส่งผลให้เกิดการเรียนรู้แบบเสริมกำลัง ซึ่งจะทำให้ในแต่ละครั้งที่เรียนรู้ตัวระบบจะสามารถแก้ปัญหาได้ดีขึ้นเรื่อย ๆ หนึ่งในตัวอย่างการใช้งานที่พบได้มากที่สุดคือ การใช้งานกับหุ่นยนต์หรือการเล่นเกม เป็นต้น สมมติว่าเราต้องการฝึกหุ่นยนต์ให้สามารถเคลื่อนที่ไปตามขั้นบันได หุ่นยนต์จะเปลี่ยนแปลงวิธีการเคลื่อนที่เพื่อนำพาตัวเองไปยังบริเวณต่าง ๆ บนพื้นฐานของผลลัพธ์ที่เกิดจากแต่ละการกระทำ เมื่อหุ่นยนต์ล้มลงหรือตกจากบันได ข้อมูลก็จะถูกสอบเทียบใหม่ซึ่งจะทำให้แต่ละก้าวที่หุ่นยนต์เคลื่อนที่ไปนั้นแตกต่างกัน หุ่นยนต์จะถูกฝึกฝนโดยอาศัยการลองผิดลองถูกแบบนี้ต่อไปเรื่อย ๆ จนกระทั่งเข้าใจถึงพฤติกรรมการขึ้นบันไดในแต่ละขั้น หรืออาจจะเรียกอีกอย่างได้ว่า หุ่นยนต์เรียนรู้จากผลของการกระทำของตนเอง ดังนั้นอัลกอริทึมในการเรียนรู้จะต้องมีความสามารถที่จะค้นหาความสัมพันธ์ของเป้าหมายที่กำหนด ซึ่งก็คือการขึ้นบันไดสำเร็จ และลำดับการเคลื่อนที่ของหุ่นยนต์ว่าแต่ละการกระทำส่งผลให้เกิดการเคลื่อนที่อย่างไรได้บ้าง การเรียนรู้แบบเสริมกำลังนี้อาจจะมองได้อีกรูปแบบหนึ่งว่าคล้ายกับการฝึกสอนสัตว์เลี้ยง เพื่อให้กระทำการบางอย่างตามเป้าหมายที่เราต้องการโดยมีรางวัลเป็นค่าตอบแทนจากการที่ทำเป้าหมายได้สำเร็จ

2.4.1.3.1 การเรียนรู้แบบเสริมกำลังด้วยเทคนิค Q-learning algorithm



รูปที่ 2.8 แผนผังแสดงกระบวนการทำงานของวิธีการเรียนรู้แบบเสริมกำลังด้วยเทคนิคคิวเลิหนึ่ง

การเรียนรู้แบบคิว [25] (Q – learning algorithm) เป็นเทคนิคหนึ่งในรูปแบบระบบการทำงานเชิงปัญญาประดิษฐ์ที่ถูกนำมาปรับใช้ในการแก้ปัญหาที่มีความซับซ้อนสูง ในเทอมของการเรียนรู้แบบคิว ผู้ปฏิบัติการหรือผู้ที่มีปฏิสัมพันธ์กับสิ่งแวดล้อมโดยรอบจะถูกเรียกว่าเอเจนต์ (Agent) มีหน้าที่กระทำตามชุดคำสั่งในการออกกิริยาท่าทางในสิ่งแวดล้อมนั้น ๆ ผลจากการกระทำใด ๆ ของเอเจนต์จะทำให้สิ่งแวดล้อมรอบตัวเปลี่ยนแปลงไป รวมไปถึงการรับรู้ของเอเจนต์ที่ถูกอัปเดตสถานะให้มีความเป็นปัจจุบันอยู่เสมอผ่านการตรวจจับด้วยอุปกรณ์เซนเซอร์ ยิ่งไปกว่านั้นในแต่ละรอบการทำงาน (Epoch) เอเจนต์จะได้รับค่ารางวัล (Reward) จากการออกกิริยากระทำบางอย่างอันเป็นผลให้สถานะของสิ่งแวดล้อมนั้นเปลี่ยนแปลงไป

สำหรับกลยุทธ์การเรียนรู้แบบนี้ จะต้องมีการกำหนดเป้าหมายและกระบวนการเรียนรู้จะดำเนินไปแบบการลองผิดลองถูกกับสิ่งแวดล้อม เอเจนต์จะได้รับค่ารางวัลหรือการลงโทษซึ่งอ้างอิงจากชุดการกระทำและสถานะที่รับรู้ โดยให้ s (State) นั้นแทนสถานะที่มีในเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สภาพแวดล้อมนั้น ๆ a (Action) หรือ ชุดกิจกรรมสำหรับกระทำในสภาพแวดล้อมนั้น ๆ โดยจุดประสงค์ของกลยุทธ์การเรียนรู้คือ ฝึกสอนให้เอเจนต์ค้นหา optimal policy $s \rightarrow a$ หรือนโยบายที่จะแก้ปัญหาได้ดีที่สุด ว่าควรจะทำอะไรให้เหมาะสมต่อสถานะต่าง ๆ ที่เกิดขึ้นในสิ่งแวดล้อม อ้างอิงจากชุดกิจกรรมที่ได้รับเพื่อให้ได้ค่ารางวัลที่สูงที่สุดในระยะยาว ในกระบวนการเรียนรู้ทั้งหมด ค่าการเรียนรู้คิวของทุก ๆ คู่สถานะและการกระทำ (state-action pair หรือ $Q(s, a)$) จะถูกเก็บบันทึกไว้ในฐานข้อมูลและมีการอัปเดตค่าให้เป็นปัจจุบันอยู่เสมอ ค่าการเรียนรู้คิวนั้นเปรียบเสมือนเป็นค่าคุณภาพของกิจกรรม a ที่เราได้ตัดสินใจกระทำลงไป ในสถานะ s ว่ามีประโยชน์ต่อสถานะต่อ ๆ ไปในอนาคต และส่งผลถึงการได้มาซึ่งนโยบายในการแก้ปัญหามากน้อยเพียงใด สามารถอธิบายในรูปแบบสมการดังต่อไปนี้

$$Q(S, A) = Q(S, A) + \alpha [R + \gamma \max_{a'} Q(S', A') - Q(S, A)] \quad (2.29)$$

โดยที่ R นั้นแทนค่ารางวัลที่จะได้รับในแต่ละรอบการทำงาน (Epoch t) โดยที่ $0 < \gamma < 1$ แทนค่าตัวแปรกำหนดความสำคัญของค่ารางวัลในระยะยาว (discount factor) และ α แทนค่าอัตราในการเรียนรู้ (learning rate) ให้ $Q^*(S, A)$ แทนค่าการเรียนรู้คิวที่ดีที่สุด (optimal Q-value) ของคู่สถานะการกระทำ (state-action pair (S, A)) ถ้าหากทุกกิจกรรมที่ได้กระทำลงไปในแต่ละสถานะนั้นมีความเป็นอนันต์ $Q(S, A)$ จะเกิดการผกผัน (converge) กลายเป็น $Q^*(S, A)$ การทำงานของ Q-learning algorithm ทั้งหมดนั้นสามารถอธิบายขั้นตอนการทำงานได้ดังต่อไปนี้

1. ระบบได้รับค่าสถานะปัจจุบัน หรือ state, s_t
2. เลือกกิจกรรม a_t จากชุดกิจกรรมที่ได้กำหนดไว้มากระทำ
3. ระบบได้รับค่าสถานะใหม่ s_{t+1} และค่ารางวัล r ที่เกิดจากการกระทำกิจกรรม a_t
4. อัปเดตแมทริกซ์ค่า Q-value ของ $Q(s_t, a_t)$ ด้วยสมการข้างต้น
5. แทนค่าให้กับ s_t ในสมการ $s_t = s_{t+1}$
6. ขณะที่สถานะใด ๆ ในสภาพแวดล้อมยังไม่เกิดจุดผกผัน หรือ $s_t \neq s_{optimal}$ ให้วนกลับไปทำซ้ำในขั้นตอนที่สอง

โดยที่ α และ γ นั้นคืออัตราในการฝึกสอน และตัวแปรกำหนดความสำคัญของค่ารางวัลในระยะยาวตามลำดับ ซึ่งทั้งสองตัวแปรนี้จะมีค่าเป็นจำนวนเต็มอยู่ในช่วงระหว่าง $[0,1]$

เทอมที่สำคัญในการเรียนรู้แบบเสริมกำลังนั้นคือ

- เอเจนต์ (Agent) คือสิ่งที่สามารถรับรู้ความเป็นไปของสิ่งแวดล้อม สิ่งที่เรายพยายามฝึกสอนให้เรียนรู้และกระทำต่อสิ่งแวดล้อมเพื่อให้เป็นไปตามเป้าหมาย
- สภาพแวดล้อมโดยรอบ (Environment) คือโลกจริงหรือโลกเสมือนที่เอเจนต์อยู่
- เป้าหมาย (Goal) คือเป้าหมายสูงสุดที่กำหนดให้กับเอเจนต์
- สถานะ (State) คือรูปสถานะต่าง ๆ ของสภาพแวดล้อมที่เอเจนต์สามารถตรวจรู้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ค่ารางวัล (Rewards) เป็นสิ่งที่เอเจนต์ต้องการ ไม่เพียงแต่ค่ารางวัลใกล้ ๆ แต่รวมไปถึง ค่ารางวัลในอนาคตด้วย อัลกอริทึมการเรียนรู้แบบเสริมกำลังพยายามที่จะหาวิธีเพื่อที่จะให้ได้มาซึ่ง ค่ารางวัลสูงสุด เราสามารถกล่าวได้อีกอย่างหนึ่งว่า ค่ารางวัลคือตัวบ่งบอกถึงการกระทำ (action) ของเอเจนต์ว่าดีหรือไม่ดี แต่ไม่ได้บอกว่าเป็นการกระทำที่ถูกต้องหรือไม่ถูกต้องแต่อย่างใด สมมติสถานการณ์หนึ่งขึ้นมาโดยหุ่นยนต์หรือเอเจนต์ที่ต้องการออกจากเขาวงกต ดังนั้นเทอมสำคัญที่ต้องกำหนดให้ระบบการเรียนรู้ก็คือ

- เป้าหมาย (Goal) คือการออกจากเขาวงกตให้ได้
- ค่ารางวัล (Reward) มีค่าเป็น 1 สำหรับกรณีที่ออกจากเขาวงกตได้, รางวัลมีค่าเป็น 0 ถ้าออกเขาวงกตไม่ได้
- แนวทางแก้ไขปัญหาที่เป็นไปได้ (Possible solution) คือการเคลื่อนที่แบบสุ่มในสภาพแวดล้อมเป้าหมาย (environment) จนกว่าจะแก้เขาวงกตได้ เนื่องจากเราต้องการให้หุ่นยนต์ออกจากเขาวงกต แต่ไม่ได้บอกให้เลือกรูปวิธีที่มีประสิทธิภาพมากที่สุด การที่หุ่นยนต์เคลื่อนที่แบบสุ่มไปเรื่อย ๆ ท้ายที่สุดหุ่นยนต์ก็สามารถออกจากเขาวงกตได้อยู่ดี ดังนั้นทุก ๆ ครั้งที่หุ่นยนต์เคลื่อนที่ หรือทำให้สถานะของสภาพแวดล้อมที่ตรวจรู้ได้เปลี่ยนแปลงไปนั้น หุ่นยนต์จะได้รับค่ารางวัลเท่ากับ -1 เพื่อเป็นการลงโทษเป้าหมาย (penalty) ทุก ๆ ครั้งที่ยังไม่สามารถแก้ปัญหาได้ หุ่นยนต์จึงต้องใช้จำนวนครั้งในการเคลื่อนที่ให้น้อยที่สุดเพื่อรักษาค่ารางวัลให้มากที่สุดไว้ได้อยู่
- การกระทำของเอเจนต์ (Action) ต่อสภาพแวดล้อมเป้าหมายตัวอย่างเช่น กำหนดให้เอเจนต์คือ ตัวละครในวิดีโอเกม ดังนั้นชุดกิริยา คือ การเคลื่อนที่ขึ้น ลง ซ้าย ขวา และ กระโดด เป็นต้น ซึ่งกิริยาเหล่านี้ส่งผลให้เกิดสถานะต่อไปในสภาพแวดล้อมเป้าหมาย
- ไทม์มิ่ง (Timing) คือลำดับของ สถานะ, กิริยา และค่ารางวัล ที่เกิดขึ้นในสภาพแวดล้อม โดยรอบเอเจนต์ ตัวอย่างเช่น สมมติให้ในสถานการณ์หนึ่ง หุ่นยนต์เริ่มต้นด้วย state, $S(t)$ และเลือกทำกิริยา, $A(t)$ แต่หุ่นยนต์จะได้ค่ารางวัลที่เวลา $R(t+1)$ ดังนั้นรางวัลที่ได้จะมาจากสถานะและกิริยาของลำดับก่อนหน้าเสมอ ซึ่งจะส่งผลให้หุ่นยนต์ก้าวไปยัง state ถัดไปด้วย สามารถเขียนเป็นสมการความสัมพันธ์ได้ดังต่อไปนี้

$$S_{(t+1)} \rightarrow [S_t, A_t, S_{(t+1)}] \text{ หรือเขียนได้อีกกรุปหนึ่งว่า } S_{(t+1)} \rightarrow [S, A, S'] \quad (2.30)$$

สรุปคือเอเจนต์จะมีปฏิสัมพันธ์กับสภาพแวดล้อมเป้าหมายโดยการที่เอเจนต์อยู่ในสถานะหนึ่งเลือกการกระทำใด ๆ ในสถานะนั้น ซึ่งการกระทำดังกล่าวก็จะส่งผลให้เอเจนต์เปลี่ยนสถานะมาอยู่ในสถานะใหม่ ขณะเดียวกันสภาพแวดล้อมเป้าหมายก็ทำให้เอเจนต์ได้ค่ารางวัลมาเป็นค่าตอบแทนในการที่ได้มาอยู่ในสถานะใหม่นี้ อย่างไรก็ตาม ค่ารางวัลนั้นอาจจะเป็นบวก (Reward) หรือเป็นลบ (Penalty) ก็ได้ขึ้นอยู่กับว่าสถานะดังกล่าวจะส่งผลต่อการได้มาซึ่งเป้าหมาย (Goal) อย่างไรบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.3.2 มัลติอาร์มแบนดิต (Multi-armed bandit)

มัลติอาร์มแบนดิต (Multi-armed bandit) มีความหมายมาจากการที่คนหรือเอเจนต์ใด ๆ จะต้องทำการเลือกการกระทำหนึ่งจากกิริยาในหลาย ๆ รูปแบบ โดยที่ไม่รู้ว่าผลลัพธ์จากการกระทำนั้นจะเป็นเช่นไร เป้าหมายคือเพื่อหาผลลัพธ์ที่ดีที่สุดจากการเลือกตัดสินใจในแต่ละการกระทำ ตัวอย่างเช่น การเล่นเกมสล็อตแมชชีนเพื่อเสี่ยงดวง เครื่องสล็อตแมชชีนหนึ่งเครื่องเทียบเท่าได้กับวันอาร์มแบนดิต (one-armed bandit) สมมติว่าผู้เล่นไม่รู้มาก่อนว่าเครื่องสล็อตแมชชีนเครื่องนี้มีอัตราการให้เงินรางวัลเท่าใด ทุกครั้งที่เล่นจึงมีโอกาสที่จะสูญเสียเงินไปโดยเปล่าประโยชน์ และเนื่องจากผู้เล่นต้องการได้เงินรางวัล ผู้เล่นจึงต้องเสี่ยงโชคไปเรื่อย ๆ จนกว่าเครื่องจะจ่ายเงินรางวัล ยิ่งไปกว่านั้นหากมีเครื่องสล็อตแมชชีนให้ผู้เล่นได้ลองเสี่ยงโชคทั้งหมด 3 เครื่องด้วยกัน แน่ใจว่าผู้เล่นต้องการที่จะใช้เครื่องสล็อตแมชชีนที่ให้อัตราการจ่ายเงินรางวัลสูงที่สุดเพื่อลดโอกาสที่จะเสียเงินโดยเปล่าประโยชน์ แต่การที่จะรู้ว่าเครื่องใดให้อัตราสูงที่สุดนั้น ก็ต้องมาจากการเลือกทดลองแต่ละเครื่องหลาย ๆ ครั้ง ดังนั้นปัญหาของมัลติอาร์มแบนดิต คือการที่เราไม่สามารถกำหนดได้ว่าควรจะต้องเล่นเครื่องเล่นเครื่องใด และเป็นจำนวนครั้งเท่าใด จึงจะสามารถประเมินได้ว่าเครื่องเล่นดังกล่าวมีโอกาสให้รางวัลสูงที่สุด

2.4.1.3.3 ภาวะเอกซ์พลอ-เอกซ์พลอยท์ ดิเลมมา (Explore-exploit dilemma)

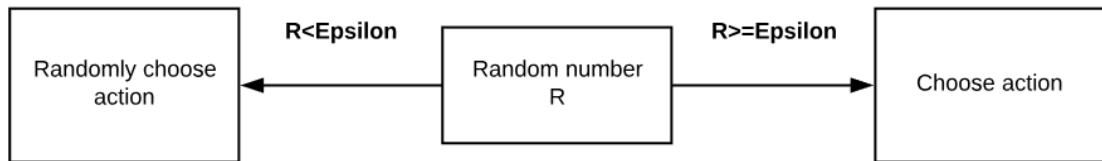
คือนิยามของการที่หุ่นยนต์หรือเอเจนต์ทำการเลือกตัดสินใจต่อสถานการณ์หนึ่ง ๆ ที่มีกิริยาหลาย ๆ กิริยาให้เลือกกระทำและส่งผลให้เกิดสถานะที่แตกต่างกันออกไป โดยหุ่นยนต์นั้นสามารถที่จะเลือกกิริยาที่มีอัตราความสำเร็จสูงซึ่งอาจจะจะเป็นเส้นทางที่นำไปสู่เป้าหมายหรือไม่ก็ได้หรือเลือกกิริยาที่ให้อัตราความสำเร็จที่น้อยกว่าหรือไม่เคยถูกสำรวจมาก่อน ซึ่งเส้นทางนี้อาจจะนำไปสู่เป้าหมายหรือไม่ก็ได้เช่นกัน อย่างไรก็ตาม อัตราความสำเร็จนั้นเกิดจากการที่หุ่นยนต์ผ่านประสบการณ์ต่อเหตุการณ์นั้น ๆ มาก่อนแล้วและบันทึกผลในเชิงสถิติ ดังนั้นเส้นทางที่ยังไม่ถูกสำรวจ ซึ่งให้อัตราความสำเร็จที่น้อยกว่าเส้นทางที่ถูกสำรวจมาแล้วหลายครั้ง ก็อาจจะนำไปสู่เป้าหมายได้ดีกว่าเส้นทางที่มีอัตราสำเร็จสูงก็เป็นได้

2.4.1.3.4 ปัจจัยความโลภเอปซิลอน (Epsilon greedy, ϵ)

เป็นอัลกอริทึมในการแก้ปัญหาภาวะเอกซ์พลอ-เอกซ์พลอยท์ดิเลมมา โดยการใช้ความน่าจะเป็นของการเกิดเหตุการณ์ต่าง ๆ มาเป็นตัวตัดสินใจ ภายในอัลกอริทึมนี้จะกำหนดตัวแปร ϵ ซึ่งเป็นตัวรักษาสมดุลการเลือกสำรวจสถานะใหม่ (Explore) หรือเลือกสถานะที่ให้ผลลัพธ์ในปัจจุบันได้ดีที่สุด (Exploit) ของหุ่นยนต์เพื่อให้เกิดความเท่าเทียมของโอกาสที่จะพบเจอกับเหตุการณ์ต่าง ๆ เพื่อให้ได้มาซึ่งเป้าหมายสูงสุด การทำงานของอัลกอริทึมนี้จะเป็นการสุ่มตัวเลขจำนวนหนึ่งขึ้นมา ถ้าหากตัวเลขนั้นน้อยกว่าค่า ϵ ที่ได้กำหนดไว้ก่อนหน้านี้ ระบบจะทำการสุ่มเลือกการกระทำ (action) ที่มีในสถานะนั้น ๆ เพื่อเพิ่มโอกาสในการสำรวจสถานะใหม่ ๆ ที่ระบบยังไม่ได้เรียนรู้ แต่ถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวเลขที่สุ่มได้นั้นมากกว่าหรือเท่ากับค่า ϵ ระบบจะเลือกการกระทำที่ให้อัตราความสำเร็จสูงที่สุดในสถานะนั้น ๆ



รูปที่ 2.9 ไดอะแกรมแสดงการทำงานของควมโลภเอปซิลอน (EPSILON-GREEDY)

2.4.1.3.5 ปัจจัยการลดทอน (Discount factor, γ)

ปัจจัยการลดทอน (Discount factor หรือ Discount rate) นั้นเป็นตัวแปรที่สื่อถึงความสำคัญของค่ารางวัลในอนาคต กล่าวคือระบบการเรียนรู้จะให้ความสำคัญต่อรางวัลในระยะสั้นที่สามารถพบเจอได้ในสถานะใกล้เคียง มากกว่าหรือน้อยกว่ารางวัลในระยะยาว ซึ่งตัวแปรนี้จะมีค่าระหว่าง 0 ถึง 1 ยิ่งค่า ปัจจัยการลดทอน มีค่ามากก็จะยิ่งให้ความสำคัญต่อรางวัลในระยะยาวมากขึ้น ปัจจัยการลดทอนนี้เป็นอีกหนึ่งปัจจัยที่ส่งผลถึงทิศทางการเรียนรู้ของระบบ ถ้าหากเอเจนต์สนใจต่อค่ารางวัลในระยะสั้น เอเจนต์อาจจะเลือกการกระทำที่ได้ค่ารางวัลในทันทีโดยไม่สนใจว่าการกระทำนั้นสามารถไปถึงเป้าหมายสูงสุดได้หรือไม่ โดยกรณีนี้จะเหมาะกับการกิจที่ต้องการระยะเวลาสั้นในการเรียนรู้แต่ละรอบการทำงาน (Episodic task) แต่ถ้าเอเจนต์สนใจค่ารางวัลระยะยาว มันอาจจะยอมเลือกการกระทำที่ให้ค่ารางวัลน้อยหรือได้รับบทลงโทษ (penalty) บ้างเพื่อให้ได้มาซึ่งเป้าหมายสูงสุด ซึ่งกรณีนี้จะเหมาะกับการกิจที่ในแต่ละรอบการเรียนรู้ต้องใช้เวลาานหรือเรียกได้อีกอย่างว่าเป็นการทำงานอย่างต่อเนื่อง (continuous task)

2.4.1.3.6 อัตราการเรียนรู้ (Learning rate, α)

อัตราการเรียนรู้ของระบบมีค่าระหว่าง 0-1 ตัวแปรนี้จะเป็นตัวกำหนดความถี่ในการอัปเดตค่าการเรียนรู้แบบคิวให้เป็นปัจจุบัน เป็นอีกปัจจัยหนึ่งที่ส่งผลต่อกระบวนการและระยะเวลาที่ใช้ในการเรียนรู้เช่นกัน โดยหากตั้งค่าอัตราการเรียนรู้ให้เป็น 0 แสดงว่าค่าการเรียนรู้แบบคิวจะไม่ถูกอัปเดตเลย แต่ถ้าตั้งค่าเข้าใกล้ 1 มาก แสดงว่าค่าการเรียนรู้แบบคิวจะถูกอัปเดตอย่างรวดเร็วในทุก ๆ รอบการเรียนรู้

2.4.1.4 โครงข่ายประสาทเทียม และการเรียนรู้เชิงลึก

การเรียนรู้เชิงลึก [26] นั้นเป็นเทคนิคเฉพาะของการเรียนรู้ของเครื่องที่อาศัยการทำงานของโครงข่ายประสาทเทียมซ้อนกันหลาย ๆ ชั้นเพื่อที่จะเรียนรู้ข้อมูลที่มีความซับซ้อนมาก ๆ การเรียนรู้เชิงลึกนั้นมีประโยชน์มากโดยเฉพาะเมื่อเราต้องการสร้างโมเดลสำหรับเรียนรู้ชุดข้อมูลที่ไม่ใช่รูปแบบโครงสร้างที่แน่นอน

โครงข่ายประสาทเทียมนั้นจะประกอบไปด้วยจำนวนชั้นในการประมวลผล 3 เลเยอร์หรือมากกว่า ได้แก่ อินพุตเลเยอร์ (Input layer), ฮิดเดนเลเยอร์ (Hidden layer) ที่มีอย่างน้อย

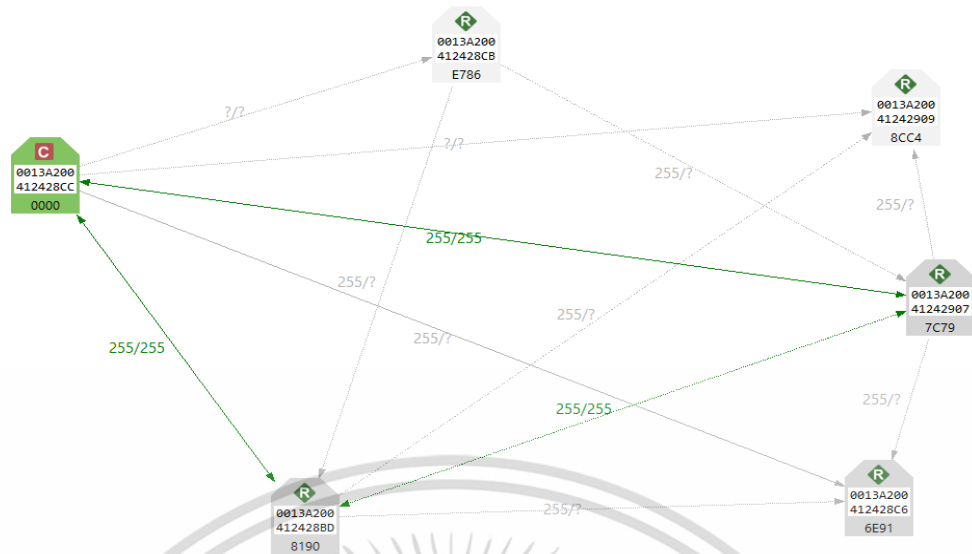
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนึ่งชั้นหรือมากกว่า และเอาต์พุทเลเยอร์ (Output layer) โดยที่ข้อมูลจะถูกรับเข้ามาในโครงข่ายผ่านชั้นอินพุทเลเยอร์ จากนั้นข้อมูลจะถูกปรับแต่งเปลี่ยนแปลงค่าต่าง ๆ ในชั้นฮิดเดนเลเยอร์และเอาต์พุทเลเยอร์ โดยจะอ้างอิงจากค่าน้ำหนักของแต่ละโหนดประมวลผล โดยทั่วไปแล้วโครงข่ายประสาทเทียมจะประกอบไปด้วยโหนดประมวลผลจำนวนมากมายมหาศาลที่จะเชื่อมต่อกันและกัน การเชื่อมต่อเหล่านี้เองจึงทำให้เกิดนิยามของการเรียนรู้เชิงลึกขึ้นมา ซึ่งก็คือการที่มีฮิดเดนเลเยอร์จำนวนมากเชื่อมต่อกัน ด้วยความสามารถในการเรียนรู้แบบซ้ำไปซ้ำมานี้เอง ทำให้โครงข่ายประสาทเทียมสามารถที่จะปรับแต่งค่าอ้างอิงต่าง ๆ ให้มีผลลัพธ์ที่มีความสมเหตุสมผลมากที่สุดจนกว่าจะถึงจุดสิ้นสุดกระบวนการทำงานของโครงข่าย

การเรียนรู้เชิงลึกคือ เทคนิคการเรียนรู้ของเครื่องอีกรูปแบบหนึ่งที่ถูกนำมาผสมผสานกันระหว่าง การเรียนรู้แบบไม่มีผู้ฝึกสอนและแบบมีผู้ฝึกสอน ปกติแล้วการเรียนรู้เชิงลึกสามารถที่จะเรียนรู้ชุดข้อมูลที่ยังไม่ถูกจัดโครงสร้างและยังไม่ถูกจัดกลุ่มได้ ยังมีจำนวนชั้นฮิดเดนเลเยอร์มาประมวลผลมากขึ้นเท่าไร ระบบก็จะสามารถจัดการกับปัญหาที่มีความซับซ้อนได้มากขึ้นเท่านั้น

2.5 เครือข่ายเซนเซอร์ไร้สาย

เครือข่ายเซนเซอร์ไร้สาย [27], [28] เป็นแพลตฟอร์มสำหรับการเก็บบันทึกข้อมูลรูปแบบใหม่ที่มีความสามารถในการจัดการระบบภายในตัวเองและมีกลไกในการปรับค่าโดยอัตโนมัติ ซึ่งเครือข่ายไร้สายนี้จะเข้ามาช่วยในการลดความซับซ้อนของระบบการมอนิเตอร์ข้อมูลและควบคุมกระบวนการแบบเดิมที่ต้องอาศัยการเชื่อมต่อแบบสายซึ่งมีความยุ่งยากในการติดตั้งและดูแลรักษา ระบบ ในการเพิ่มขึ้นของจำนวนเซนเซอร์ที่สื่อสารกันอย่างไร้สายนั้นทำให้ระบบสามารถเก็บข้อมูลได้อย่างอิสระและรวบรวมมายังศูนย์ประมวลผลกลาง นิยามของเซนเซอร์ไร้สายนั้นไม่ได้หมายความว่าเฉพาะส่วนตรวจวัดเท่านั้นแต่รวมถึงส่วนบอร์ดประมวลผลในตัว ส่วนสื่อสาร และพื้นที่จัดเก็บข้อมูลประกอบกันอยู่ภายในตัวโหนด ด้วยความสามารถเหล่านี้เองทำให้โหนดเซนเซอร์แต่ละตัวมีความสามารถในการวิเคราะห์ข้อมูลภายในเครือข่ายได้ในตัว การสื่อสารของโหนดเซนเซอร์ไร้สายนั้น สามารถเกิดขึ้นได้ระหว่างตัวโหนดด้วยกันเองและระหว่างตัวโหนดกับสถานีฐาน ซึ่งเป็นเหมือนตัวกลางที่ทำหน้าที่เผยแพร่ข้อมูลไปยังภายนอกเครือข่าย แสดงผลการวิเคราะห์เป็นรูปภาพ รวมไปถึงระบบจัดเก็บข้อมูลภายนอก



รูปที่ 2.10 แสดงเครือข่ายเซนเซอร์ไร้สายประกอบไปด้วยโหนดชิปแบบเมชเน็ตเวิร์ก

ความสามารถของโหนดเซนเซอร์นั้นมีได้อย่างแพร่หลาย โหนดเซนเซอร์อย่างง่ายอาจจะทำการมอนิเตอร์ตัวแปรทางกายภาพชนิดใดชนิดหนึ่งได้ แต่ตัวโหนดอาจจะมีประสิทธิภาพและการทำงานที่ซับซ้อนมากขึ้นเมื่อประยุกต์ใช้เซนเซอร์หลากหลายชนิดเข้าไปในโหนด ๆ เดียว ตัวอย่างเช่น โหนดเซนเซอร์ไร้สายที่ประกอบไปด้วย เซนเซอร์ตรวจจับการสั่นสะเทือน เซนเซอร์ทางแสง เซนเซอร์ตรวจวัดสนามแม่เหล็ก เป็นต้น ยิ่งไปกว่านั้นแต่ละโหนดอาจมีความสามารถในการสื่อสารที่ต่างกันออกไป ตัวอย่างเช่น สื่อสารผ่านทางคลื่นอัลตราซาวด์ แสงอินฟราเรด หรือสัญญาณความถี่วิทยุ ซึ่งแต่ละรูปแบบก็จะมี ความแตกต่างในด้านของความสามารถในการรับส่งการลดทอนของสัญญาณ หรือประสิทธิภาพในการรับส่งข้อมูล เป็นต้น โหนดเซนเซอร์อย่างง่ายอาจจะทำการมอนิเตอร์ค่าและส่งข้อมูลไปยังโหนดหรือสถานีแม่ในขณะที่โหนดเซนเซอร์ที่มีอุปกรณ์ขั้นสูงเช่น มีหน่วยประมวลผลกำลังสูง มีความสามารถในการประหยัดพลังงาน หรือมีพื้นที่เก็บข้อมูลขนาดใหญ่อาจจะทำการควบคุมกระบวนการต่าง ๆ ได้ภายในตัวเอง โหนดประสิทธิภาพสูงเหล่านี้จะทำหน้าที่เป็นเหมือนแกนกลางของระบบในการควบคุมทรัพยากรและบริหารจัดการข้อมูลให้แก่โหนดลูกตัวอื่น ๆ ในเครือข่าย ที่ไม่สามารถเข้าถึงสถานีได้โดยตรง ท้ายที่สุด โหนดเซนเซอร์บางประเภทอาจจะสามารถในการเข้าถึงเทคโนโลยีการบอกตำแหน่งเป้าหมายได้ ตัวอย่างเช่น ตัวรับสัญญาณจีพีเอส (Global Positioning System: GPS) ซึ่งจะทำให้มันสามารถระบุตำแหน่งของตัวเอง รวมไปถึงใช้ตัวเองเป็นจุดอ้างอิง อ้างอิงตำแหน่งของอุปกรณ์ข้างเคียงอื่น ๆ ได้อีกด้วย

การสื่อสารของระบบสื่อสารไร้สายโดยทั่วไปจะอยู่บนมาตรฐาน IEEE 802.11 ประกอบไปด้วยย่านความถี่ต่าง ๆ อาทิเช่น ความถี่ 2.4 GHz สำหรับมาตรฐาน IEEE 802.11b และ IEEE 802.11g ขณะที่ มาตรฐาน IEEE 802.11a ใช้ย่านความถี่ที่ 5 GHz เป็นต้น อย่างไรก็ตามมาตรฐาน IEEE 802.11 นี้มีอัตราการรับส่งข้อมูลที่ช้าลงสูง ซึ่งไม่สอดคล้องกับประสงค์ในการประหยัดพลังงานของเครือข่ายเซนเซอร์ไร้สายพลังงานต่ำ ดังนั้นจึงได้มีการพัฒนาโปรโตคอลการสื่อสารขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับอัตราการรับส่งข้อมูลต่ำ เช่นโปรโตคอล IEEE 802.15.4 สำหรับการสื่อสารระยะสั้นในเครือข่าย เซนเซอร์พลังงานต่ำ เป็นต้น

โทโพโลยีในการสื่อสารสำหรับเครือข่ายเซนเซอร์ที่มีระยะการสื่อสารเป็นวงกว้าง จะใช้รูปแบบสตาร์โทโพโลยี (star topology) เนื่องจากโหนดเซนเซอร์ไร้สายแต่ละตัวสามารถรับส่งข้อมูลกับสถานีฐานได้โดยตรงภายในการกระโดด (hop) เพียงครั้งเดียว (single hop) อย่างไรก็ตามเครือข่ายเซนเซอร์ที่จำเป็นต้องครอบคลุมพื้นที่สื่อสารเป็นวงกว้างนั้น ควรจะต้องรักษาระดับการใช้พลังงานสำหรับกรณีที่ต้องใช้กำลังส่งสูงไว้ด้วย ดังนั้นการสื่อสารแบบการกระโดดของข้อมูลหลายครั้ง (multi hop) จึงเป็นวิธีการรับส่งข้อมูลที่นิยมมากกว่าสำหรับเครือข่ายเซนเซอร์ สำหรับโทโพโลยีแบบเครือข่ายเมช (Mesh network) นั้น โหนดเซนเซอร์แต่ละตัวไม่เพียงแต่รับรู้และเผยแพร่ข้อมูลของตัวเองได้แต่จะต้องมีความสามารถที่จะถ่ายทอดข้อมูลจากโหนดเซนเซอร์อื่น ๆ ได้ด้วย ทั้งนี้มันจึงต้องมีความสามารถที่จะร่วมงานกับโหนดเซนเซอร์อื่น ๆ ในการกระจายข้อมูลไปยังสถานีฐาน ปัญหาการค้นหาลำเส้นทางนี้เองจึงเป็นอีกหนึ่งอุปสรรคที่สำคัญของการรับส่งข้อมูลกันภายในเครือข่ายแบบการกระโดดของข้อมูลหลายครั้ง อย่างไรก็ตาม ประโยชน์ของรูปแบบการรับส่งข้อมูลลักษณะนี้คือการที่โหนดเซนเซอร์แต่ละตัวที่ทำหน้าที่เป็นตัวถ่ายทอดข้อมูลของโหนดอื่น ๆ นั้นมีโอกาสที่จะวิเคราะห์ข้อมูลหรือจัดการกับข้อมูลใด ๆ ซ้ำ ๆ กันหลาย ๆ ครั้ง ซึ่งก่อให้เกิดการสำรองข้อมูลและตรวจสอบข้อมูลซ้ำ ๆ สร้างความถูกต้องและความน่าเชื่อถือของข้อมูลมากยิ่งขึ้น

2.6 หุ่นยนต์เคลื่อนที่ และตัวควบคุมไมโครคอนโทรลเลอร์

วิทยาการหุ่นยนต์นั้นมีการพัฒนามาอย่างยาวนานจนถึงปัจจุบัน โดยเฉพาะอย่างยิ่งหุ่นยนต์เคลื่อนที่ ในอดีตหุ่นยนต์นั้นถูกควบคุมการเคลื่อนไหวด้วยคอมพิวเตอร์ขนาดใหญ่ที่มีราคาสูงและน้ำหนักมากที่อาจจะไม่สามารถพกพาได้ อีกทั้งยังจำเป็นต้องมีสายสัญญาณเชื่อมต่อเพื่อรับส่งข้อมูลระหว่างกันอยู่เสมอ แต่ในปัจจุบันเทคโนโลยีที่ก้าวหน้าทำให้เราสามารถสร้างหุ่นยนต์เคลื่อนที่ที่มีขนาดเล็ก ควบคุมผ่านคอมพิวเตอร์สมองกลที่ฝังตัวอยู่ในตัวหุ่นยนต์เองด้วยราคาที่ถูกลงและน้ำหนักเบา

สำหรับหุ่นยนต์เคลื่อนที่นั้น [1] รูปแบบที่ง่ายที่สุดในการสร้างการเคลื่อนที่ให้กับหุ่นยนต์ก็คือ ล้อ หุ่นยนต์ที่เคลื่อนที่ด้วยล้อนั้นจะประกอบไปด้วย ล้อขับเคลื่อนหนึ่งล้อหรือมากกว่านั้นและอาจจะมีล้อเสริมที่ไม่สามารถที่จะสร้างแรงขับเคลื่อนได้ เพียงแค่ประคองหุ่นยนต์ไว้ให้ทรงตัวได้เท่านั้น อย่างไรก็ตาม แนวทางการออกแบบหุ่นยนต์ขับเคลื่อนด้วยล้อนั้นที่นิยมมากที่สุดนั้นเรียกว่าแอกเคอร์แมนสตีริง (Ackermann steering) ซึ่งเป็นระบบขับเคลื่อนและควบคุมทิศทางของหุ่นยนต์เคลื่อนที่แบบมาตรฐาน โดยจะให้ระบบขับเคลื่อนแบบอิสระไปติดตั้งอยู่กับล้อคู่หลังแบบแยกมอเตอร์ขับเคลื่อนกัน และล้อคู่หน้าจะทำการติดตั้งแกนเพลลาที่เชื่อมล้อทั้งสองเข้าด้วยกัน พร้อมติดตั้งมอเตอร์ขับเคลื่อนและแกนบังคับที่ใช้ในการหมุนล้อคู่หน้า ให้ตัวหุ่นยนต์สามารถหันไปตามทิศทางที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการได้ สำหรับบางกรณีพิเศษ หุ่นยนต์ขับเคลื่อนด้วยล้อนั้น ตัวล้ออาจจะถูกออกแบบมาให้เคลื่อนที่แบบทุกทิศทาง (Omni-directional) ที่เรียกว่าแมคคานัมไดรฟ์ (Macanumm drive) ซึ่งจะมีการติดตั้งระบบขับเคลื่อนด้วยมอเตอร์เข้ากับล้อที่ออกแบบมาเป็นพิเศษทั้งสี่ข้าง

ข้อเสียของหุ่นยนต์ขับเคลื่อนด้วยล้อคือ จะเคลื่อนที่ได้กับเฉพาะถนนหรือสถานที่ที่มีพื้นผิวที่เป็นระนาบเท่านั้น ถึงแม้ว่าจะมีการออกแบบหุ่นยนต์ที่ใช้ตีนตะขาบสำหรับเคลื่อนที่ในพื้นที่ผิวขรุขระมาแก้ไขปัญหานี้ได้ก็ตาม การใช้ตีนตะขาบนั้นก็ไม่สามารถให้ความแม่นยำในการบอกตำแหน่งได้ แต่อย่างไรก็ตามหุ่นยนต์ขับเคลื่อนด้วยล้อก็ยังคงมีประสิทธิภาพ ง่ายต่อการสร้างและออกแบบจริง

ในงานวิจัยนี้ได้ใช้หุ่นยนต์ประเภทล้อ ขับเคลื่อนด้วยชุดเกียร์และมอเตอร์กระแสตรงที่ติดตั้งให้กับแต่ละล้อเพื่อสร้างแรงขับให้กับหุ่นยนต์และควบคุมทิศทางการหมุนได้อย่างอิสระ และเนื่องจากว่าตัวหุ่นยนต์ไม่ได้มีการออกแบบเพลาร่วมที่ล้อคู่หน้าสำหรับใช้ในการหันควบคุมทิศทางของตัวหุ่นยนต์ เราจึงอาศัยการควบคุมทิศทางการหมุนของมอเตอร์แทน เพื่อให้ความเร็วและทิศทางการหมุนของล้อแต่ละข้างเป็นตัวสร้างทิศทางการหมุน การเลี้ยวของหุ่นยนต์ อย่างไรก็ตาม หลังจากทดสอบกับพื้นผิวของสนามหลาย ๆ ประเภท หุ่นยนต์สี่ล้อในงานวิจัยนี้สามารถใช้งานกับพื้นผิวของสนามที่มีความเป็นระนาบและเรียบเนียนเท่านั้น ไม่สามารถวิ่งบนพื้นผิวขรุขระ หรือเป็นหลุมบ่อได้ เพราะจะทำให้การหมุนของล้อในขณะที่สร้างการเลี้ยว หรือการหันให้กับตัวหุ่นยนต์เกิดการติดขัด และเกิดความเสียหายให้กับตัวรถได้ในเวลาต่อมา

ไมโครคอนโทรลเลอร์นั้นเรียกได้ว่าเป็นชิปคอมพิวเตอร์ขนาดเล็กที่ฝังตัวอยู่ในวงจรไฟฟ้า ทำหน้าที่ในการควบคุมการทำงานต่าง ๆ ภายในอุปกรณ์ แน่นอนว่าชิปคอมพิวเตอร์แบบฝังตัวดังกล่าวย่อมมีประสิทธิภาพในการประมวลผลไม่เทียบเท่ากับชิปคอมพิวเตอร์ทั่วไปที่ประยุกต์ใช้กับคอมพิวเตอร์ตั้งโต๊ะหรือเวิร์กสเตชัน (workstation) หากแต่ว่าชิปคอมพิวเตอร์แบบฝังตัว จะมีราคาถูกกว่า ประหยัดพลังงานกว่า ซึ่งจะเหมาะสำหรับการนำไปใช้เฉพาะทาง เช่น การอ่านค่าดิจิตอลหรืออนาล็อกจากอุปกรณ์เซนเซอร์ การควบคุมอุปกรณ์แอคชูเอเตอร์ (actuator) เป็นต้น และเนื่องจากว่าคุณสมบัติในการใช้งานที่น้อยกว่าเมื่อเทียบกับคอมพิวเตอร์ตั้งโต๊ะ ไมโครคอนโทรลเลอร์จึงถูกออกแบบมาให้ใช้งานง่าย มีส่วนประมวลผลและหน่วยความจำในตัว สามารถต่อพ่วงกับอุปกรณ์ภายนอกได้เพื่อขยายขอบเขตความสามารถทำงานของอุปกรณ์ เช่น พอร์ตขยาย อินพุท/เอาต์พุท, พอร์ตอินเทอร์เฟซที่ใช้เชื่อมต่อสื่อสารกับอุปกรณ์ภายนอกแบบต่าง ๆ เป็นต้น

อาร์ดูโน้เป็นบอร์ดไมโครคอนโทรลเลอร์ในรูปแบบแพลตฟอร์มเปิด ซึ่งเปิดโอกาสให้นักพัฒนาได้เข้ามาปรับแต่งและแก้ไขของค์ประกอบต่างๆได้ตามที่ต้องการ อีกทั้งในส่วนของฮาร์ดแวร์ อาร์ดูโน้ก็ยังรองรับการใช้อุปกรณ์ต่อพ่วงต่าง ๆ เช่น เซนเซอร์ มอเตอร์ หรืออุปกรณ์อื่นมาปรับแต่งและแก้ไขได้อย่างอิสระด้วยอินเทอร์เฟซสื่อสารที่ถูกออกแบบมาตามมาตรฐาน รวมไปถึงสายวงจรรวมและผังโครงสร้างต่าง ๆ ถูกเปิดเผยสู่สาธารณะเพื่อให้ผู้ใช้งานต่าง ๆ สามารถเข้ามาดาวน์โหลดไปประยุกต์ใช้งานกับโปรเจกต์ของตัวเองได้โดยไม่ต้องเสียค่าใช้จ่ายใด ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

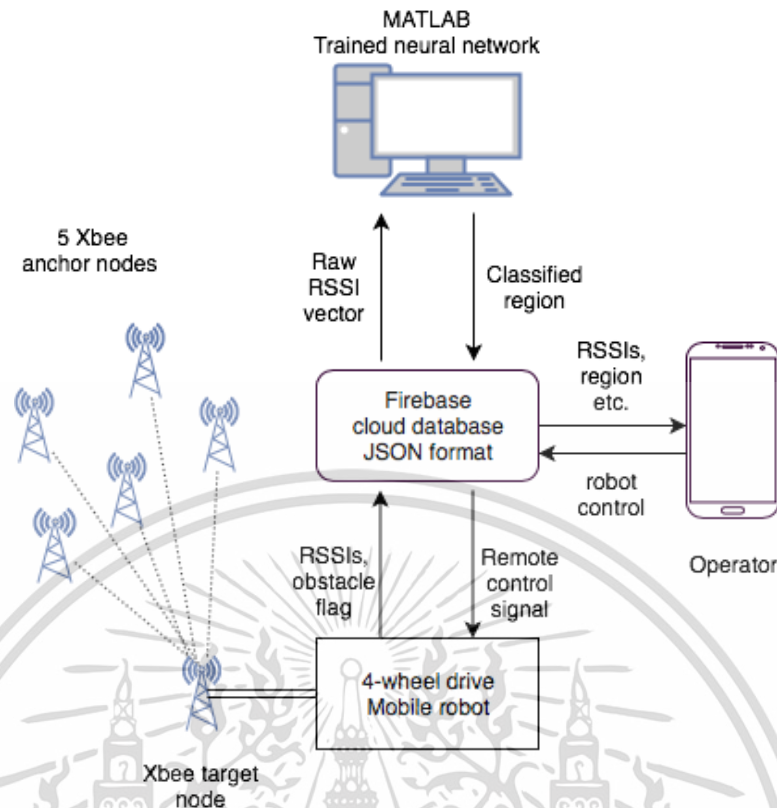
บทที่ 3

การบอกตำแหน่งในอาคารด้วยวิธีจำแนกขอบเขตพื้นที่ที่ไม่ สมมาตรสำหรับการนำทางหุ่นยนต์เคลื่อนที่

3.1 กล่าวนำ

ในงานวิจัย [12], [32], [33], [34] ได้มีการกล่าวถึงการบอกตำแหน่งและสร้างแผนที่ของพื้นที่เป้าหมายพร้อมทั้งวางแผนสร้างเส้นทางสำหรับหุ่นยนต์เคลื่อนที่ภายในสนามทดสอบ หลบเลี่ยงสิ่งกีดขวางต่าง ๆ เพื่อไปยังตำแหน่งตามที่ได้กำหนด ตัวอย่างเช่น การประยุกต์ใช้งานร่วมกันระหว่างข้อมูลความแรงสัญญาณจากเครือข่ายเซนเซอร์ไร้สายที่ติดตั้งในพื้นที่ทดสอบ ผสมกับอัลกอริทึมที่ใช้ในการเคลื่อนที่หลบหลีกสิ่งกีดขวางของหุ่นยนต์ที่เรียกว่าวิธีโพเทนเชียลฟิลด์ (Potential Field method) [31] เพื่อบอกตำแหน่งของเป้าหมายรวมไปถึงสิ่งกีดขวางต่าง ๆ โดยรอบ สร้างรูปแบบการเคลื่อนที่โดยยึดหลักการจำลองเวกเตอร์ตรงกันข้ามกับสิ่งกีดขวาง เพื่อนำทางหุ่นยนต์ไปยังพื้นที่เป้าหมายได้อย่างปลอดภัย หรืองานวิจัย [25] และ [30] ที่ได้กล่าวถึงเทคนิคเฮกเตอร์แมปปิง (Hector Mapping) และการเรียนรู้แบบคิว (Q-learning) ซึ่งเป็นเทคนิคที่มีรากฐานมาจากการบอกตำแหน่งและสร้างแผนที่หรือสแลม (SLAM) โดยจะมีการสร้างแผนที่พร้อมกับบอกตำแหน่งเป้าหมายอ้างอิงจากแผนที่ปัจจุบัน แต่ถึงอย่างไรก็ตาม แนวคิดที่ถูกหยิบยกขึ้นมา นั้นมีความซับซ้อนในการพัฒนา รวมถึงจำเป็นต้องอาศัยการใช้ทรัพยากรในการประมวลผลค่อนข้างมาก จึงจะสามารถบอกตำแหน่งเป้าหมายที่ถูกต้องแม่นยำและส่งผลให้สร้างเส้นทางไปยังเป้าหมายได้ถูกต้อง ดังนั้นจึงได้มีแนวคิดในการสร้างเทคนิคเพื่อบอกตำแหน่งหุ่นยนต์ที่ง่ายไม่ซับซ้อน ทั้งยังคงประสิทธิภาพในการบอกตำแหน่ง เอื้ออำนวยต่อการสร้างเส้นทางเคลื่อนที่ของหุ่นยนต์เป้าหมายให้มีประสิทธิภาพมากที่สุด

ในบทที่ 3 นี้ได้นำเสนอวิธีการบอกตำแหน่งภายในอาคารด้วยวิธีการจำแนกขอบเขตพื้นที่ที่ไม่สมมาตร (Asymmetry Regional Boundary Localization หรือ ARBL) โดยจะนำเสนอหลักการพิจารณาและการสร้างพื้นที่ที่ไม่สมมาตรด้วยความสามารถในการเข้าใจค่าของอุปกรณ์ส่งสัญญาณไร้สาย การเก็บบันทึกและจัดเตรียมชุดข้อมูลสำหรับฝึกสอนโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตร การสร้างแผนที่ในสนามทดสอบและการสร้างเส้นทางในการหลบหลีกสิ่งกีดขวางของหุ่นยนต์เคลื่อนที่ด้วยการเรียนรู้ของเครื่องแบบเสริมกำลังซึ่งกระบวนการทั้งหมดสามารถแสดงด้วยบล็อกไดอะแกรมดังรูปต่อไปนี้



รูปที่ 3.1 ไดอะแกรมแสดงส่วนประกอบทั้งหมดภายในระบบ

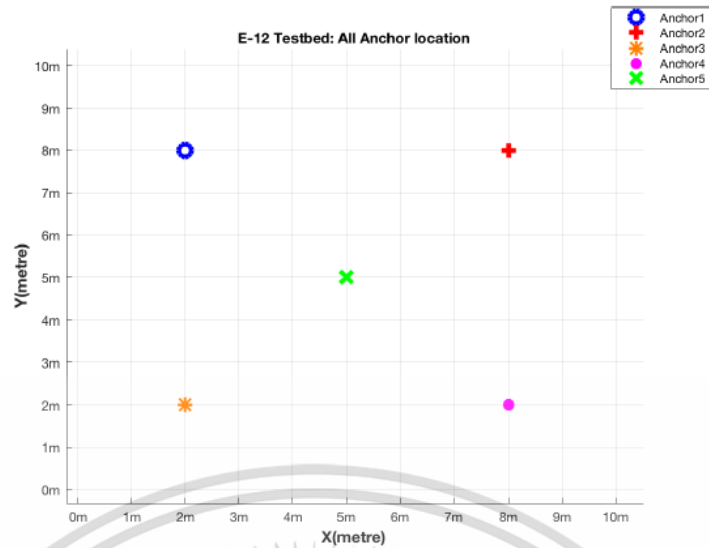
3.2 การอ่านค่าความแรงสัญญาณระหว่างตัวหุ่นยนต์เคลื่อนที่กับเสาสัญญาณอ้างอิงในสนาม

ความแรงสัญญาณนั้นเป็นหนึ่งในปัจจัยสำคัญที่ใช้ระบุตำแหน่งเป้าหมาย ซึ่งโดยทั่วไปแล้วความแรงสัญญาณนั้นคือค่าความต่างระหว่างกำลังส่งที่วัดได้จากเสาสัญญาณต้นทางและจากเสาปลายทางขณะที่สัญญาณเดินทางผ่านตัวกลางด้วยระยะทางใด ๆ ค่าความแรงสัญญาณจะมีลักษณะแปรผกผันกับระยะทาง ในทางอุดมคตินั้น ยิ่งระยะห่างระหว่างเสามีมากความแรงสัญญาณก็จะยิ่งตกลง ในทางกลับกัน ถ้าระยะห่างน้อยความแรงของสัญญาณก็จะมามาก แม้ว่าการใช้ความแรงสัญญาณมาเป็นตัวบอกระยะห่างนั้นมีความไม่แน่นอนอยู่มาก อย่างไรก็ตาม การอ่านค่าความแรงสัญญาณนั้นสามารถกระทำได้จากอุปกรณ์สื่อสารไร้สายแทบทุกชนิด โดยไม่จำเป็นต้องใช้ฟังก์ชันพิเศษ หรืออุปกรณ์เสริมใด ๆ มาเกี่ยวข้อง

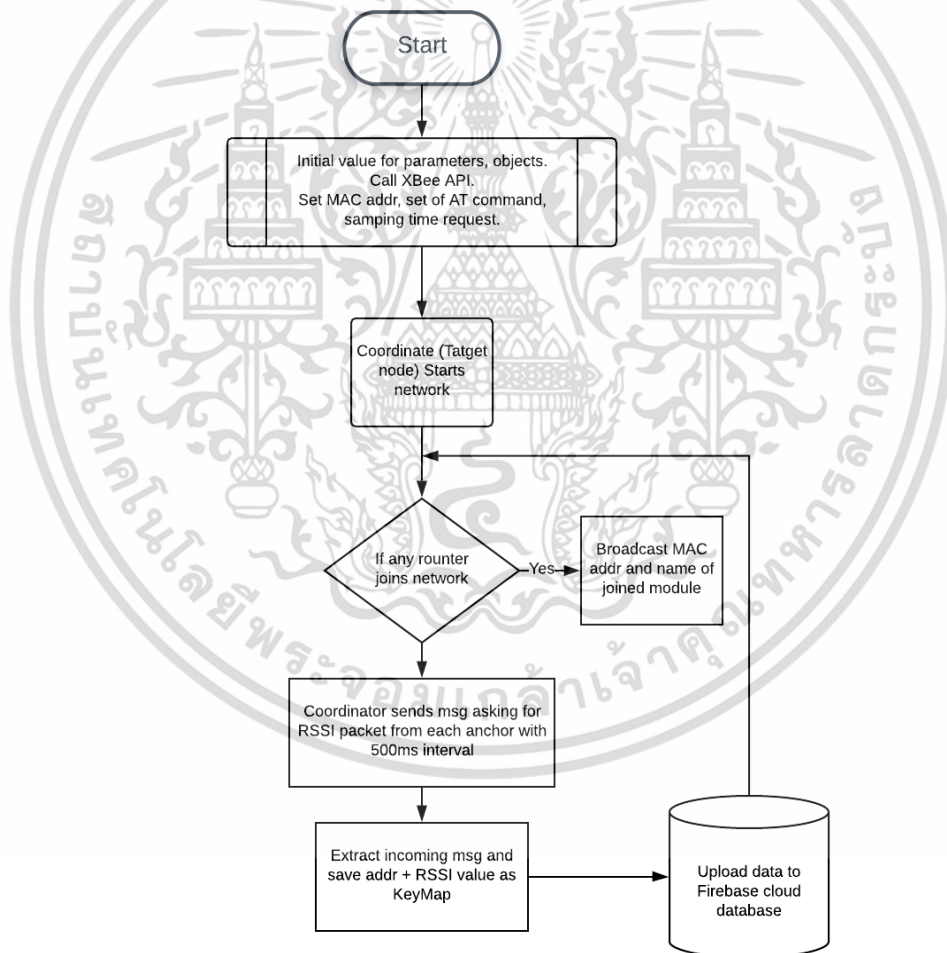
เสาสัญญาณอ้างอิงที่ใช้ในสนามมีทั้งหมด 5 เสาด้วยกัน ถูกจัดวาง ณ ตำแหน่งต่าง ๆ ดังรูปที่ 3.2 ในแต่ละเสาสัญญาณ โมดูลเอ็กซ์บีซีซีรี่ส์สอง (Xbee series 2) นั้นมีอินเตอร์เฟสในการโปรแกรมชื่อว่า เอพีไอ (Application Programming Interface: API) ที่ทำหน้าที่จัดการแพ็กเก็ตข้อมูลสำหรับสื่อสารกันภายในเครือข่าย เมื่อแพ็กเก็ตข้อมูลเดินทางจากเสาส่งต้นทางมายังเสารับปลายทาง หนึ่งในไบนารีข้อมูลของแพ็กเก็ตนั้น ๆ จะประกอบไปด้วย ค่าความแรงสัญญาณที่เสารับอ่านค่าได้แสดงดังรูปที่

3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 กราฟแสดงตำแหน่งการวางเสาสัญญาณอ้างอิงในสนามทดสอบ



รูปที่ 3.3 โฟลว์ชาร์ทแสดงกระบวนการรับส่งข้อมูลระหว่างโมดูลซิกบีในเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MSB/LSB ของ Address อุปกรณ์ปลายทาง	ชนิดของเฟรม ข้อมูล	คำสั่ง AT command เพื่อ ร้องขอค่า RSSI “DB”	Checksum ตรวจสอบความถูกต้อง ครบถ้วน
MSB/LSB ของ Address อุปกรณ์ต้นทาง	ชื่อโมดูลต้น ทาง	ชนิดของเฟรม ข้อมูล	ข้อมูลความแรง สัญญาณ RF data
			Checksum ตรวจสอบความถูกต้อง ครบถ้วน

รูปที่ 3.4 แสดงเฟรมข้อมูลสำหรับสื่อสารกันภายในเครือข่ายชิปที่ทั้งเฟรมประเภทส่งและรับ

โดยระบบจะมีลำดับขั้นตอนการอ่านค่าความแรงสัญญาณเริ่มต้นจาก เสาสัญญาณเป้าหมายที่ติดตั้งอยู่กับตัวหุ่นยนต์เคลื่อนที่ จะทำการยิงแฟล็กเกิดข้อมูลไปขอวัดค่าความแรงสัญญาณจากเสาอ้างอิงแต่ละเสาในสนาม ช่วงเวลาห่างกันเสาสัญญาณละ 500 มิลลิวินาที เมื่อเสาสัญญาณอ้างอิงใด ๆ ได้รับคำสั่งขอมาแล้ว ก็จะส่งแฟล็กเกิดข้อมูลตอบกลับไปให้เสาสัญญาณเป้าหมาย และในท้ายที่สุด ตัวหุ่นยนต์ก็จะทำการคัดกรองเอาเฉพาะข้อมูลความแรงสัญญาณที่เสาเป้าหมายอ่านค่าได้ มาใช้งานในส่วนต่อ ๆ ไป

3.3 การจัดเตรียมสนามทดสอบ

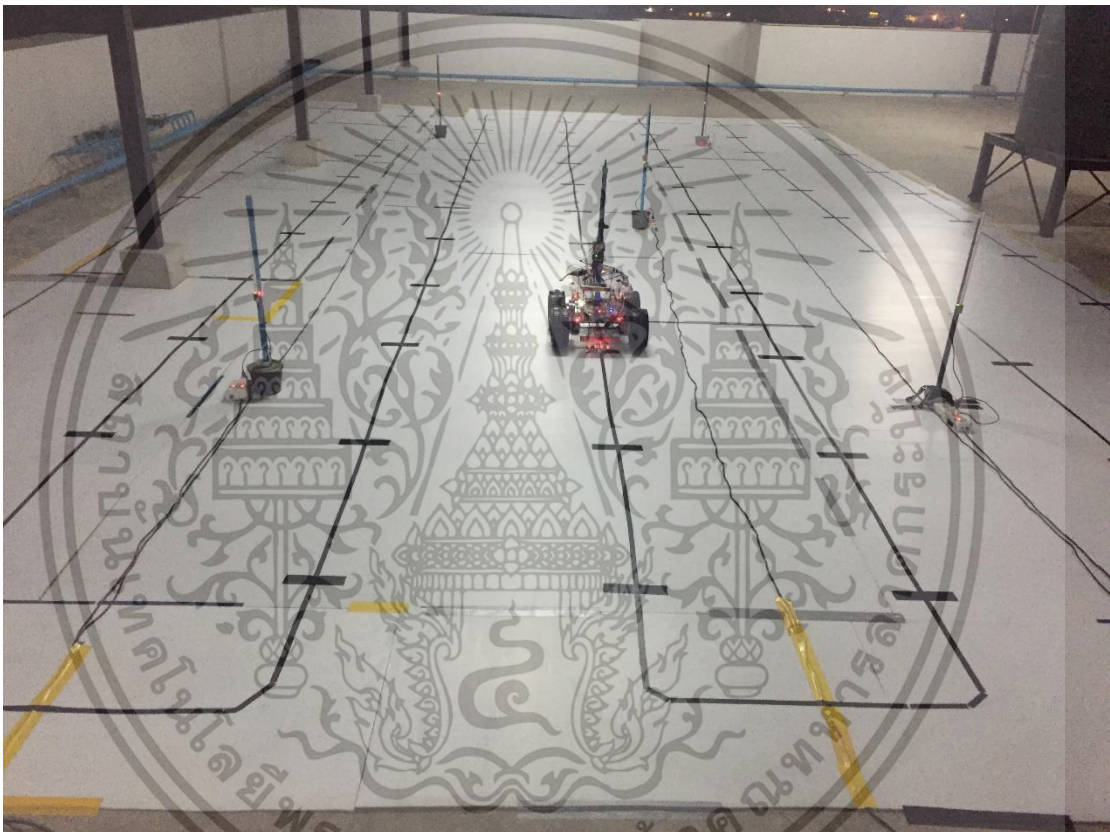


รูปที่ 3.5 สนามทดสอบที่ 1 ขนาดพื้นที่ 100 ตารางเมตร (กว้าง 10 เมตร ยาว 10 เมตร)

เนื่องจากพื้นที่ทดสอบนั้น เป็นพื้นที่ปิดภายในอาคารซึ่งโดยส่วนมากจะเป็นห้องโถง ทางเดิน หรือห้องประชุม ดังนั้นลักษณะของห้องจึงมีรูปทรงเป็นสี่เหลี่ยม อีกทั้งพื้นผิวของห้องมีความเรียบเนียนไม่ใช่ผิวขรุขระ จึงทำให้หุ่นยนต์เคลื่อนที่ที่ออกแบบมาสามารถเคลื่อนที่ได้เป็นอย่างดี อีกทั้งส่วนเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำคัญของการบอกตำแหน่งที่นำเสนอในงานวิจัยนี้มาจากการเก็บข้อมูลความแรงสัญญาณ ณ จุดต่างๆ ในสนามมาวิเคราะห์และสร้างรูปแบบของข้อมูลเพื่อใช้ในการบอกตำแหน่ง ดังนั้นจึงได้มีสร้างตารางขึ้นมาบนสนามเพื่อที่จะกำหนดจุดให้หุ่นยนต์เก็บชุดความแรงสัญญาณทั่วทั้งสนามทดสอบ

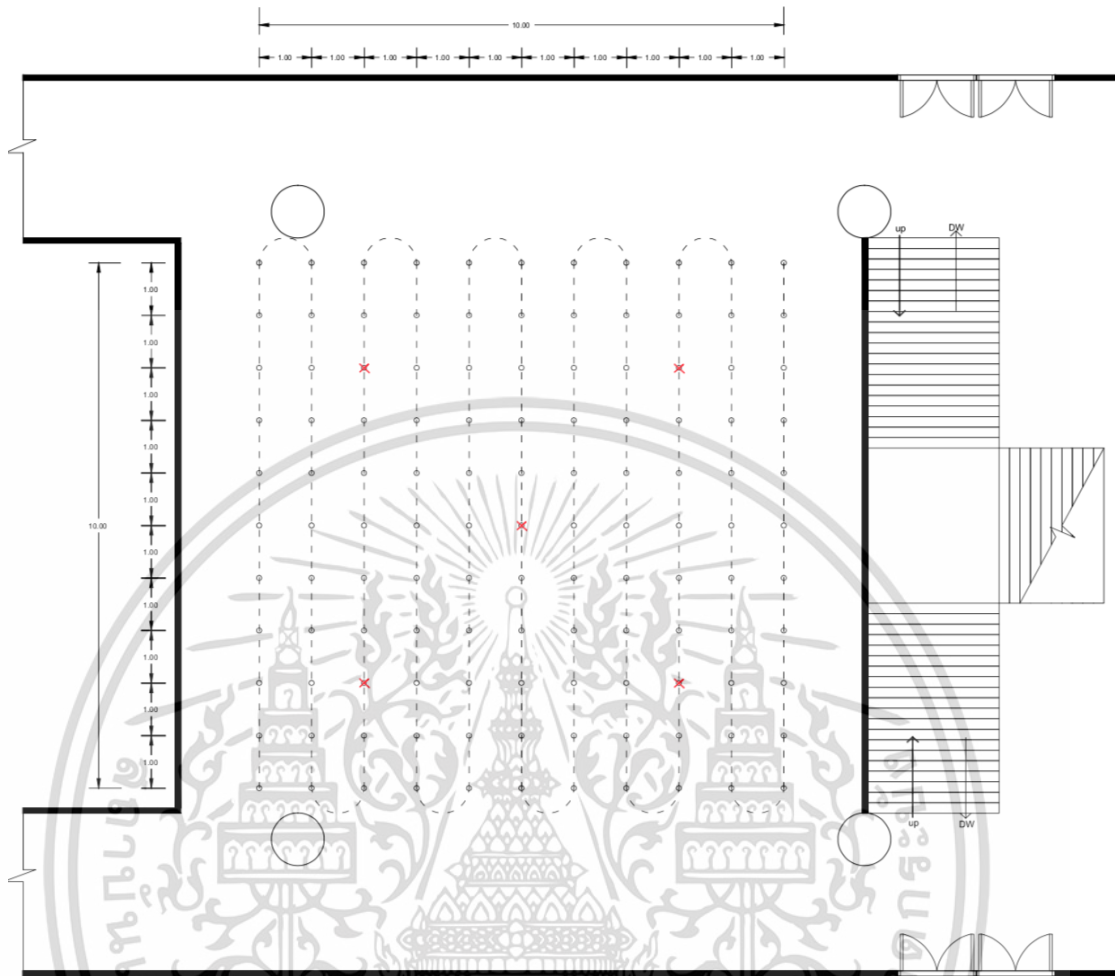
สนามทดสอบในการทดลองนี้ประกอบไปด้วยพื้นที่ปิดในอาคารสองที่ด้วยกัน สนามแรกมีลักษณะเป็นสี่เหลี่ยมจัตุรัส ขนาดความกว้าง 10 เมตรและความยาว 10 เมตร ส่วนสนามทดสอบที่สองมีลักษณะเป็นสี่เหลี่ยมผืนผ้า กว้าง 6 เมตรและยาว 10 เมตร โดยที่ทั้งสองสนามได้มีการสร้างตารางจำลองขึ้นมา ขนาดกริดกว้าง 1 เมตรและยาว 1 เมตรด้วยเทปสีดำ จุดตัดที่ที่เกิดขึ้นในสนามจะกำหนดให้เป็นจุดที่เก็บชุดข้อมูลความแรงสัญญาณ



รูปที่ 3.6 สนามทดสอบที่ 2 ขนาดพื้นที่ 60 ตารางเมตร (กว้าง 6 เมตร ยาว 10 เมตร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การเก็บบันทึกชุดความแรงสัญญาณและประมวลผลข้อมูลเบื้องต้น

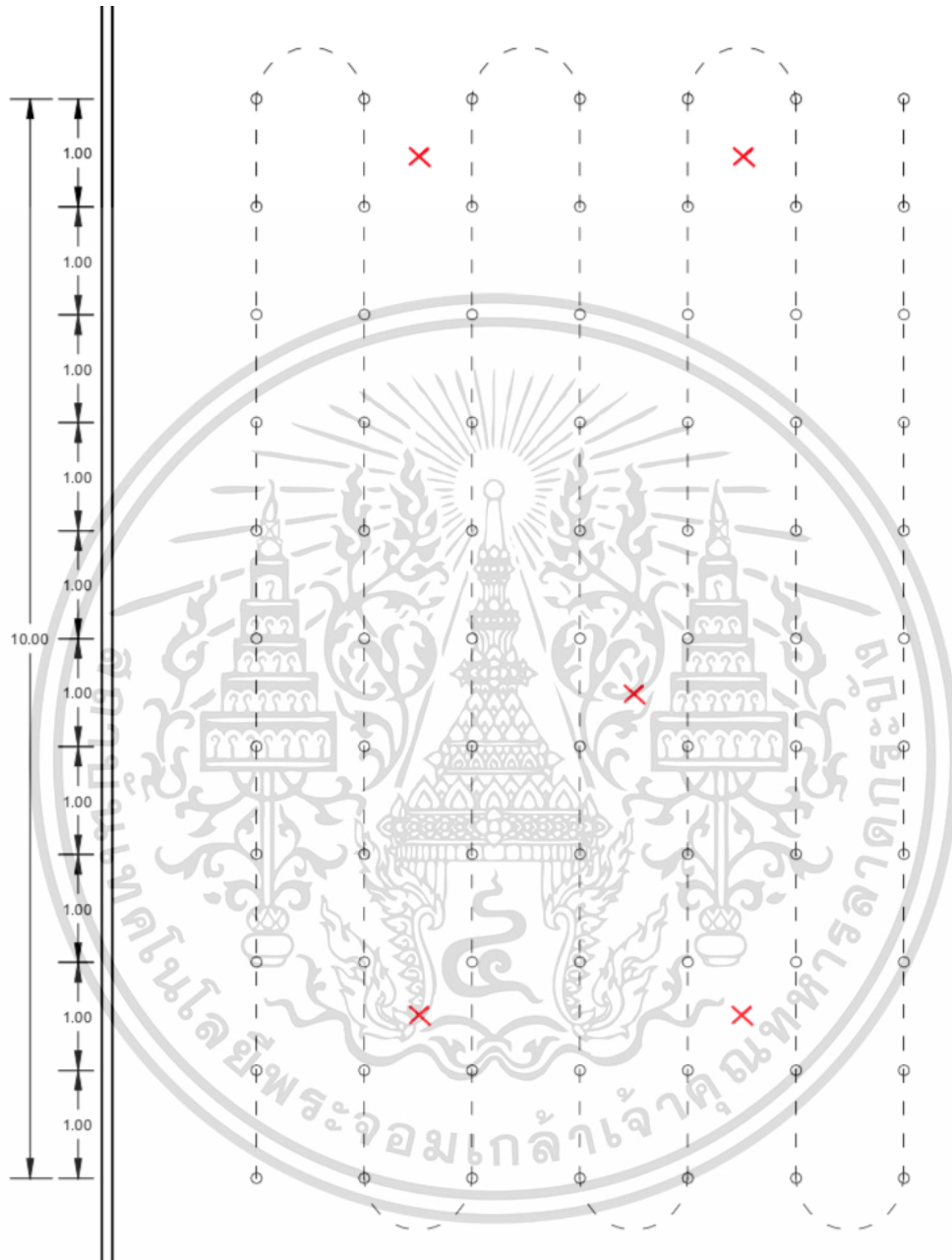


รูปที่ 3.7 แผนผังสนามทดสอบที่ 1 แสดงพิกัดเก็บข้อมูลในสนาม ตำแหน่งที่ตั้งของเสาสัญญาณ อ่างอิงและเส้นกำหนดการเคลื่อนที่ของหุ่นยนต์

ในส่วนของการเก็บบันทึกชุดความแรงสัญญาณในสนาม หุ่นยนต์จะทำการอ่านค่าความแรงสัญญาณจาก เสาสัญญาณอ้างอิงทั้ง 5 เสาตามที่ได้กล่าวไว้ในหัวข้อที่ 3.2 ณ ตำแหน่งจุดตัดของตารางที่ได้ตีเส้นไว้ ทั้งหมด 121 จุดสำหรับตารางที่ 1 (10x10) และ 77 จุดสำหรับตารางที่ 2 (10x6) โดยหุ่นยนต์จะทำการเคลื่อนที่จากจุดเริ่มต้นที่พิกัดแรกทางด้านซ้ายล่างของสนาม ดังแสดงในแผนผังของสนามทดสอบรูปที่ 3.7 และ 3.8 จากนั้นก็จะเคลื่อนที่ตามเส้นเทพีสีดำและหยุดทุก ๆ จุดตัดของตารางเพื่อทำการเก็บบันทึกชุดความแรงสัญญาณ เมื่อบันทึกข้อมูลเสร็จสิ้น หุ่นยนต์จะเคลื่อนที่ตามเส้นเทพีสีดำไปยังพิกัดต่อไปในแนวตั้ง เมื่อหุ่นยนต์เคลื่อนที่เก็บบันทึกความแรงสัญญาณไปจนถึงพิกัดสุดท้ายของสนาม ณ จุดสิ้นสุด หุ่นยนต์จะหยุดเคลื่อนที่และถือว่าชุดข้อมูลทั้งหมดที่เก็บบันทึกได้นั้นเป็น 1 รอบการเก็บข้อมูล

เนื่องจากการสร้างโมเดลสำหรับจำแนกพื้นที่ที่ไม่สมมาตร จำเป็นต้องอาศัยการเรียนรู้ โครงสร้างของความแรงสัญญาณที่แปรเปลี่ยนไปในตำแหน่งต่าง ๆ ของสนาม เราจึงจำเป็นต้องมีการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เก็บข้อมูลหลาย ๆ รอบ เพื่อใช้สร้างและฝึกสอนโมเดล รวมไปถึงทดสอบความถูกต้องแม่นยำของประสิทธิภาพในการจำแนกพื้นที่ขณะใช้งานจริง จำนวนรอบที่ได้ทำการบันทึกข้อมูลนั้น คือ 50 รอบ ต่อสนาม



รูปที่ 3.8 แผนผังสนามทดสอบที่ 2 แสดงพิกัดเก็บข้อมูลในสนาม ตำแหน่งที่ตั้งของเสาสัญญาณ อ่างอิงและเส้นกำหนดการเคลื่อนที่ของหุ่นยนต์

3.5 การคำนวณหาความสามารถในการเข้าค่าของอุปกรณ์ และการสร้างขอบเขตพื้นที่ที่ไม่สมมาตร

วิธีการบอกตำแหน่งที่น่าเสนอในงานวิจัยนี้ มีรากฐานมาจากแนวคิดการบอกตำแหน่งภายในอาคารแบบการรู้จำลายนิ้วมือ (Fingerprinting technique) ซึ่งจะเป็นการจดจำลักษณะเฉพาะของข้อมูลใด ๆ ณ ตำแหน่งหนึ่งในสนาม สาเหตุที่เลือกใช้เทคนิคนี้ก็เพราะการบอกตำแหน่งภายในอาคารเอกสารนี้เป็นเอกสารที่ลงนามไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลระบบระบุเงื่อนไขการคำนวณว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยความแรงสัญญาณโดยเทคนิคการบอกตำแหน่งแบบอ้างอิงระยะทางนั้นถึงแม้จะเป็นวิธีที่ง่ายและนิยมใช้งาน แต่มักจะประสบปัญหาเกี่ยวกับการคลาดเคลื่อนในการบอกระยะ รวมไปถึงความไวต่อการตอบสนองต่อสิ่งรบกวนหรือสิ่งรบกวนกำลังส่งสัญญาณ ดังนั้นสำหรับการจดจำลักษณะเฉพาะของความแรงสัญญาณในสนาม เราจึงต้องมีการกำหนดขอบเขตในการแบ่งส่วนการจดจำข้อมูลให้ระบบรับรู้ตำแหน่งต่าง ๆ ในสนามตรงกับเอกลักษณ์ของชุดความแรงสัญญาณใด

ในงานวิจัย [39] นั้น Silke Feldmann ได้กล่าวถึงความคลาดเคลื่อนของการบอกตำแหน่งด้วยความแรงสัญญาณอันเนื่องมาจากพฤติกรรมของสัญญาณในพื้นที่ปิดรวมไปถึงผลกระทบต่าง ๆ ที่รบกวนต่อการเดินทางของคลื่นสัญญาณ ส่งผลให้การบอกตำแหน่งนั้นคลาดเคลื่อนไป ถึงแม้ว่าอัลกอริทึมในการบอกตำแหน่งที่ใช้ในงานวิจัยดังกล่าวจะถูกจัดอยู่ในหมวดหมู่ของเทคนิคการบอกตำแหน่งแบบอ้างอิงระยะทาง แต่พฤติกรรมของสัญญาณในพื้นที่ปิดที่มีการแปรปรวนอยู่ตลอดนั้นเป็นปัญหาหนึ่งที่เกิดขึ้นอยู่เสมออย่างหลีกเลี่ยงไม่ได้ ดังนั้นในงานวิจัยดังกล่าวจึงได้มีการออกแบบวิธีในการประมาณค่าความแรงสัญญาณโดยการกำหนดช่วงขอบเขตสัญญาณโดยตั้งชื่อว่า โกลเด้นรีซีฟเวอร์พาวเวอร์แรงค์ (Golden receiver power rank) ซึ่งจะเป็นการกำหนดระดับความแรงสัญญาณ 2 ค่าให้เป็นช่วงขอบเขตบน (Upper threshold) ซึ่งมีค่าเป็นลบน้อย และช่วงขอบเขตล่าง (Lower threshold) ซึ่งมีค่าเป็นลบมากตามลำดับ แต่ละคระยะ (threshold) จะมีขอบเขตความคลาดเคลื่อนไม่เกิน $\pm 6 \text{ dBm}$ ซึ่งถ้าหากความแรงสัญญาณที่อ่านค่าได้จริงนั้นมีค่าอยู่ในช่วงขอบเขต (boundary) ที่กำหนดไว้ ก็จะถูกนำมาใช้คำนวณหาระยะห่างของตำแหน่งเป้าหมาย

ยิ่งไปกว่านั้น เทคโนโลยีการบอกตำแหน่งของบลูทูธไอบีคอน (Bluetooth iBeacon) [36], [37], [38] ซึ่งเป็นโมดูลบลูทูธไร้สายประเภทบลูทูธพลังงานต่ำ (Bluetooth 4.0 Low Energy) นั้นได้ถูกนำมาประยุกต์ใช้ในการบอกตำแหน่งหรือระยะห่างระหว่างอุปกรณ์เป้าหมายกับตัวโมดูลเองโดยใช้ความแรงสัญญาณ เพื่อที่จะหลีกเลี่ยงปัญหาความแปรปรวนของสัญญาณ ไอบีคอนได้พัฒนาเทคนิคในการบอกระยะห่างว่าสถานะพร็อกซิมีตี (Proximity states) ซึ่งเป็นการแบ่งขอบเขตของสัญญาณออกเป็น 4 ช่วงด้วยกัน ได้แก่ระยะกระชั้นชิด (Immediate), ระยะใกล้ (Near), ระยะไกล (Far), และเกินขอบเขต (Out of Range) มีค่าความแรงสัญญาณระดับต่าง ๆ กำหนดเป็นเทรชโฮลด์ของแต่ละช่วง ให้แต่ละช่วงเป็นตัวบ่งบอกถึงระดับความมั่นใจ (Level of confident) ในการบอกระยะห่าง ยิ่งค่าความแรงสัญญาณอยู่ในช่วงที่ใกล้กับแหล่งสัญญาณมาก (Immediate) ระดับความน่าเชื่อถือหรือความมั่นใจในการบอกตำแหน่งก็จะยิ่งสูงมากเช่นกัน อย่างไรก็ตาม ทั้งเทคนิคสถานะพร็อกซิมีตีและโกลเด้นรีซีฟเวอร์พาวเวอร์แรงค์นั้นต่างก็มีการทำงานที่ขึ้นอยู่กับปัจจัยการเปลี่ยนแปลงความแรงสัญญาณเป็นหลัก แม้ว่าจะมีการกำหนดขอบเขตค่าคลาดเคลื่อนที่ยอมรับได้หรือระดับขั้นการบอกระยะห่างเพียงใด หากมีการเปลี่ยนแปลงของสัญญาณอย่างกะทันหันก็อาจส่งผลให้การบอกระยะคลาดเคลื่อนไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในงานวิจัยชิ้นนี้ การแบ่งส่วนขอบเขตพื้นที่ที่ไม่สมมาตรอาศัยการใช้ความสามารถในการซ้ำค่าของอุปกรณ์ส่งสัญญาณไร้สายมาเป็นเกณฑ์กำหนด เพราะว่าการใช้งานจริงที่ระยะห่างเดียวกัน ความแรงสัญญาณที่อ่านค่าได้จากอุปกรณ์ส่งสัญญาณไร้สายแต่ละตัวนั้นไม่เท่ากันเป็นปกติอยู่แล้ว สาเหตุมาจากค่าความคลาดเคลื่อนที่มีปัจจัยจากสิ่งแวดล้อม ตัวกลางในการเคลื่อนที่ของสัญญาณ หรือสเปคของอุปกรณ์จากโรงงานผู้ผลิต รวมไปถึงตำแหน่งที่ติดตั้งอุปกรณ์ ซึ่งแต่ละส่วนของสนามอาจจะส่งผลกระทบต่อค่าความแรงสัญญาณได้ต่างกัน ดังนั้นเราจึงต้องการหาความสามารถในการอ่านค่าความแรงสัญญาณของอุปกรณ์แต่ละตัวภายใต้เงื่อนไขการใช้งานจริงของพื้นที่ทดสอบนั้น ๆ ว่า ณ ระยะห่างในการส่งสัญญาณเท่าใด ที่อุปกรณ์จะสามารถให้ค่าความแรงสัญญาณที่คงที่ และน่าเชื่อถือได้ดีที่สุด

การคำนวณหาความสามารถในการซ้ำค่าของอุปกรณ์สำหรับแบ่งขอบเขตพื้นที่ที่ไม่สมมาตรในงานวิจัยนี้ได้ทำการแบ่งขั้นตอนเป็น 3 ขั้นตอนด้วยกันดังต่อไปนี้

3.5.1 การคำนวณหาความสามารถในการซ้ำค่าในแต่ละรัศมีการทำงานของอุปกรณ์สื่อสารไร้สาย (Radius repeatability, R_r)

$$R_r = \sqrt{\frac{D}{N-1}} \quad (3.1)$$

โดยที่ R_r = ค่าความสามารถในการซ้ำค่าของอุปกรณ์ที่รัศมี r ใดๆ

D = ผลรวมของผลต่างยกกำลังสองระหว่างความแรงสัญญาณเฉลี่ยกับความแรงสัญญาณแต่ละชุดข้อมูล N

N = จำนวนชุดความแรงสัญญาณในสนามที่สนใจ

$$D = \sum_{i=1}^N (X_i - M)^2 \quad (3.2)$$

โดย M = ค่าความแรงสัญญาณเฉลี่ยของชุดความแรงสัญญาณในสนามที่สนใจ

X = ค่าความแรงสัญญาณของชุดข้อมูล i ในรัศมีการทำงานใดๆ, $i \in N$

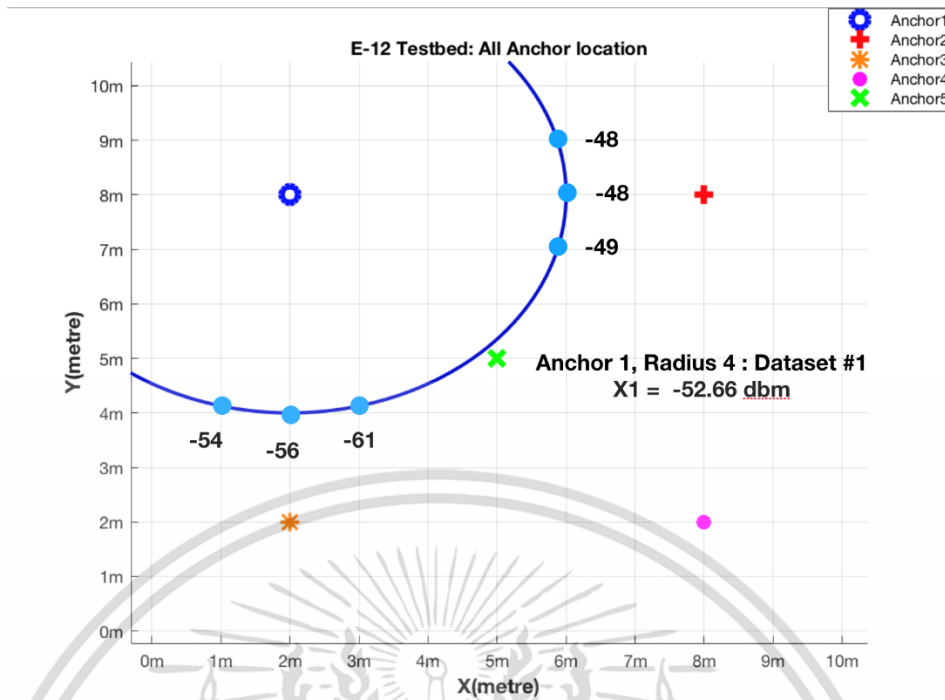
และ

$$M = \frac{1}{N} \sum_{i=1}^N X_i, X = \frac{1}{C} \sum_{c=1}^C x_c \quad (3.3)$$

โดย C = จำนวนพิกัดข้อมูลความแรงสัญญาณในสนาม บนรัศมีใดๆ, $c \in C$

ในการคำนวณหาความสามารถในการซ้ำค่าความแรงสัญญาณของอุปกรณ์สื่อสารไร้สายแต่ละตัวนั้น กระทำได้โดยนำเอาค่าความแรงสัญญาณจากชุดข้อมูล 50 ชุดที่ได้ทำการเก็บบันทึกจากขั้นตอนที่ 3.4 มาคัดแยกข้อมูลของแต่ละเสาสัญญาณอ้างอิงออกจากกัน ซึ่งจะทำให้เราได้ค่าความแรงสัญญาณจากทุกพิกัดในสนามของแต่ละเสาสัญญาณอ้างอิงเสาละ 50 ชุด จากนั้น ในแต่ละชุดข้อมูล ทำการวิเคราะห์ค่าความสามารถในการซ้ำค่าของแต่ละเสาสัญญาณอ้างอิง ณ รัศมีการทำงานทุก ๆ 1 เมตร ซึ่งจะมีขั้นตอนดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แสดงตัวอย่างการดึงข้อมูลความแรงสัญญาณ จากพิกัดที่บันทึกมาคำนวณ

1. สมมติให้ชุดข้อมูลที่หนึ่ง เสาสัญญาณอ้างอิงเสาที่ 1 ที่รัศมีการทำงาน 4 เมตรดังแสดงในรูปที่ 3.9 มีค่าความแรงสัญญาณที่เก็บบันทึกจากพิกัด (x, y) ที่ (1, 4), (2, 4), (3, 4), (6, 7), (6, 8) และ (6, 9) เท่ากับ -54, -56, -61, -49, -48 และ -49 dBm ตามลำดับ
2. ทำการเฉลี่ยค่าความแรงสัญญาณจากพิกัดทั้ง 6 จุด เพื่อหาค่าความแรงสัญญาณของรัศมี 1 เมตรนี้เพียงค่าเดียว
3. ทำซ้ำขั้นตอนที่ 1-2 กับทุก ๆ รัศมีการทำงานที่ 1 เมตรกว้างออกไปเรื่อย ๆ จนครอบคลุมทั้งสนาม (ในการทดลองนี้รัศมีการทำงานของอุปกรณ์สูงสุดที่ครอบคลุมทั้งสนามคือ 10 เมตร)
4. เมื่อได้ค่าความแรงสัญญาณของทุกระยะแล้ว ให้พิจารณาชุดข้อมูลที่ 2 ของเสาสัญญาณอ้างอิงที่ 1 และทำซ้ำขั้นตอนที่ 1-3 จนกว่าจะครบทั้ง 50 ชุดข้อมูล
5. เมื่อได้ค่าความแรงสัญญาณของแต่ละรัศมีการทำงานทั้ง 50 ชุดข้อมูล ให้ทำการหาค่าความสามารถในการซ้ำค่าของแต่ละรัศมีและกับบันทึกค่าลงในฐานข้อมูลไว้สำหรับเสาสัญญาณอ้างอิงที่ 1
6. ทำซ้ำขั้นตอนที่ 1-5 กับเสาสัญญาณอ้างอิงที่เหลือในสนาม

3.5.2 การคำนวณหาค่าเฉลี่ยค่าความสามารถในการซ้ำค่าของอุปกรณ์ (Entire field repeatability, R_f)

$$R_f = \frac{1}{J} \sum_{j=1}^J R_{rj} \quad (3.4)$$

โดยที่ R_{rj} = ค่าความสามารถในการซ้ำค่าของอุปกรณ์ที่รัศมีการทำงาน j ใดๆ

J = จำนวนรัศมีการทำงานที่ครอบคลุมสนามทดสอบ, $j \in J$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่ได้ค่าความสามารถในการซ้ำค่าความแรงสัญญาณของทุก ๆ อุปกรณ์สื่อสารไร้สายทั้งหมดในสนาม ให้ทำการเฉลี่ยค่าความสามารถในการซ้ำค่าจากทุก ๆ รัศมีของแต่ละเสาสัญญาณอ้างอิง เพื่อเป็นการกำหนดเกณฑ์มาตรฐานในการวัดความเหมาะสม หรือความสามารถของอุปกรณ์สื่อสารไร้สายแต่ละตัวในสถานที่ที่ใช้ทดสอบว่าภายใต้เงื่อนไขสภาพแวดล้อมเดียวกันนั้น อุปกรณ์สื่อสารแต่ละตัวจะมีความคลาดเคลื่อนในการอ่านค่าความแรงสัญญาณโดยรวมมากน้อยเพียงใด หรืออาจจะกล่าวได้ว่า เราทำการคำนวณหาสเปคอ้างอิงของอุปกรณ์ขณะใช้งานจริงนั่นเอง ซึ่งขั้นตอนในการคำนวณจะมีดังต่อไปนี้

1. สมมติให้พิจารณาเสาสัญญาณอ้างอิงที่ 1 ให้นำค่าความสามารถในการซ้ำค่าของอุปกรณ์ ทุก ๆ รัศมีการทำงาน มาหาค่าเฉลี่ยโดยรวมและกำหนดค่าเฉลี่ยที่ได้ ให้เป็นค่า R_f
 2. ทำขั้นตอนที่ 1 กับทุกเสาสัญญาณอ้างอิง ดังนั้นเราจะได้ค่า R_f ทั้งหมด 5 ค่าด้วยกันจากเสาสัญญาณอ้างอิงทั้งหมด 5 เสา
- 3.5.3 การกำหนดเทรชโฮลด์ให้กับค่าความสามารถในการซ้ำค่า ณ ระยะเวลาการทำงานของอุปกรณ์ที่เหมาะสมที่สุด (Threshold repeatability, R_T)

$$R_{Tj} = |R_f - R_{rj}| \quad (3.5)$$

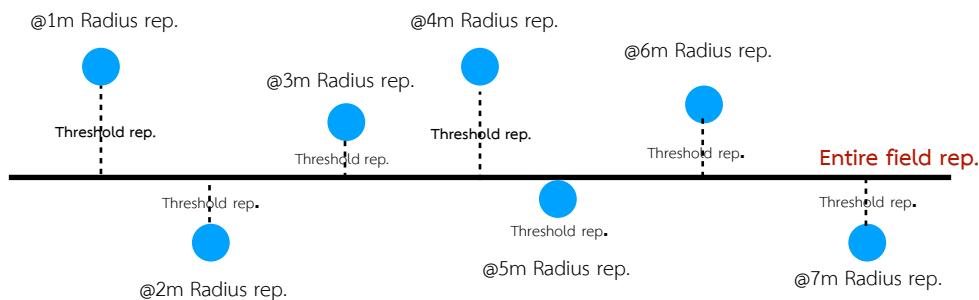
โดยที่ R_{rj} = ค่าความสามารถในการซ้ำค่าของอุปกรณ์ที่รัศมีการทำงาน j ใดๆ

R_f = ค่าเฉลี่ยค่าความสามารถในการซ้ำค่าของอุปกรณ์

J = จำนวนรัศมีการทำงานที่ครอบคลุมสนามทดสอบ, $j \in J$

ในส่วนสุดท้ายจะเป็นการหาค่าเบี่ยงเบนของความสามารถในการซ้ำค่าแต่ละรัศมีการทำงานที่มีค่าน้อยที่สุด กล่าวคือ ในแต่ละเสาสัญญาณอ้างอิง รัศมีการทำงานใดให้ค่าความสามารถในการซ้ำค่าใกล้เคียงกับค่าอ้างอิง (จากขั้นตอนที่ 2) มากที่สุด รัศมินั้นจะถูกกำหนดให้เป็นเทรชโฮลด์แบ่งเขตของเสาสัญญาณนั้น ๆ ยกตัวอย่างเช่น พิจารณาค่าความสามารถในการซ้ำค่าของเสาสัญญาณอ้างอิงที่ 1 เมื่อนำค่าจากทุก ๆ รัศมีการทำงานจาก 1-10 เมตรมาเปรียบเทียบกับค่าอ้างอิงของสนามซึ่งได้มาจากการเฉลี่ยค่าทั้งหมดในขั้นตอนที่ 2 ค่าความสามารถในการซ้ำค่าของรัศมีการทำงานใดที่มีความแตกต่างจากค่าอ้างอิงน้อยที่สุด ค่าดังกล่าวจะถูกกำหนดให้เป็นค่า R_T ของเสาสัญญาณอ้างอิง และรัศมีการทำงานนั้นจะถูกกำหนดให้เป็นขอบเขตเทรชโฮลด์ (Threshold boundary) ซึ่งเราจะทำการวิเคราะห์ด้วยแนวคิดนี้กับเสาสัญญาณอ้างอิงอื่น ๆ ที่เหลือ ทำยที่สุดก็จะสามารถกำหนดขอบเขตการบอกระยะของเสาสัญญาณอ้างอิงทั้ง 5 เสาได้

เมื่อได้ขอบเขตเทรชโฮลด์มาเป็นตัวกำหนดขอบเขตให้แก่เสาสัญญาณอ้างอิงแล้ว ขอบเขตของแต่ละเสาสัญญาณอ้างอิงในสนามจะเกิดการตัดหรือทับซ้อนกัน กลายเป็นพื้นที่ปิดรูปทรงใด ๆ ที่มีลักษณะไม่สมมาตรกันขนาดต่าง ๆ ซึ่งพื้นที่เหล่านี้จะถูกใช้ในการบอกตำแหน่งของหุ่นยนต์ในส่วนต่อ ๆ ไป รูปที่ 3.10 แสดงการหาค่าขอบเขตเทรชโฮลด์ของอุปกรณ์ซิกบีแต่ละตัวในสนามทดสอบ



รูปที่ 3.10 โค้ดโปรแกรมเปรียบเทียบค่าขอบเขตเทรชโอล์ในแต่ละรัศมีการทำงานของอุปกรณ์ซิกบี

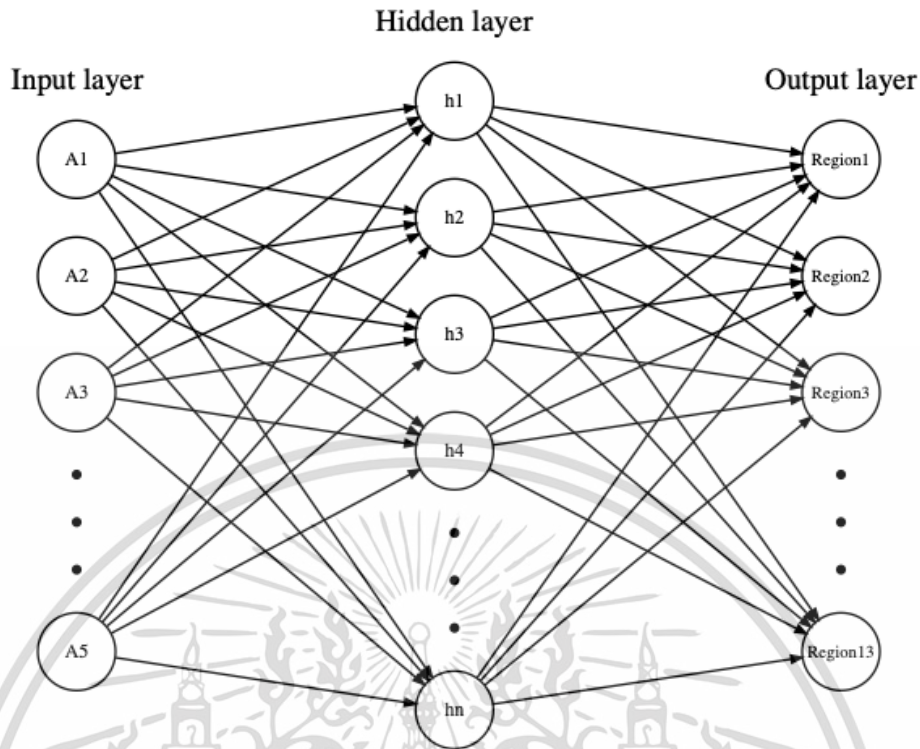
3.6 การสร้างโครงข่ายประสาทเทียมสำหรับเรียนรู้ชุดความแรงสัญญาณ

โครงข่ายประสาทเทียมสำหรับจำแนกข้อมูลที่ใช้ในงานวิจัยนี้เป็นโมเดลที่สร้างขึ้นจากโปรแกรมแมทแลป (MATLAB) ซึ่งมีเครื่องมืออำนวยความสะดวกพร้อมทั้งเอกสารประกอบในการทำความเข้าใจกระบวนการทำงานของอัลกอริทึมภายใน ทำให้เราสามารถสร้างและปรับแต่งโมเดลสำหรับจำแนกข้อมูลให้ได้เหมาะสมกับชุดข้อมูลที่ใช้ฝึกสอนและทดสอบมากยิ่งขึ้น อีกทั้งยังมีเครื่องมือที่ใช้ในการวิเคราะห์ประสิทธิภาพของข้อมูลในลักษณะกราฟที่เรียกว่า กราฟรีซีฟเวอร์โอเปอเรตติ้งคาแรคเตอร์ริสติก (Receiver Operating Characteristic : ROC) และคอนฟิวชันเมทริกซ์ (Confusion Matrix) ซึ่งแสดงประสิทธิภาพของโมเดลระหว่างฝึกสอนและทดสอบกับข้อมูลจริง ทำให้เราสามารถมองเห็นถึงประสิทธิภาพในแต่ละการตั้งค่าปรับแต่งโมเดล

การสร้างโครงข่ายประสาทเทียมสำหรับเรียนรู้ชุดความแรงสัญญาณ สามารถแบ่งกระบวนการออกเป็นขั้นตอนย่อย ๆ ได้ 6 ขั้นตอน ดังต่อไปนี้

1. การเตรียมความพร้อมของข้อมูล
2. การปรับปรุงข้อมูลฝึกสอนด้วยเทคนิคการแก้ไขความไม่สมดุลของคลาส (Class imbalance fix) และการเพิ่มสัญญาณรบกวนแบบไวท์เกาส์เซียน (White Gaussian noise)
3. การลดความซ้ำซ้อนของข้อมูล หรือนอร์มัลไลเซชัน (Normalization) ให้กับข้อมูลฝึกสอน
4. การสร้างและฝึกสอนโมเดลจำแนกข้อมูล
5. การทดสอบไฮเปอร์พารามิเตอร์ทดสอบ (Hyper-parameter testing)
6. วิธีการทำครอสวาไลเดชัน 10 ชุด (10-fold cross validation)
7. เจนเนอรัลไลเซชันเอเรอ แบบโอเวอร์ฟิตติ้งและอันเดอร์ฟิตติ้ง (Generalization error: Overfitting and Underfitting)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 แสดงตัวอย่างของโครงข่ายประสาทเทียมสำหรับเรียนรู้รูปแบบข้อมูล

3.6.1 การเตรียมความพร้อมของข้อมูล

ชุดความแรงสัญญาณที่ถูกเก็บบันทึกในแต่ละพิกัดทั่วทั้งสนามนั้นจะอยู่ในรูปแบบไฟล์ .xlsx ได้จากการทำงานของแอปพลิเคชันที่เขียนขึ้นบนโทรศัพท์มือถือแอนดรอยด์ ตัวแอปพลิเคชันจะบันทึกข้อมูลในแต่ละแถว (row) แทนข้อมูลหนึ่งประเภท ตัวอย่างเช่น แถวที่ 1 จะจัดเก็บข้อมูลความแรงสัญญาณ จากเสาสัญญาณอ้างอิงที่ 1 เป็นต้น เมื่อหุ่นยนต์เก็บบันทึกข้อมูล ณ พิกัดถัดไป ชุดความแรงสัญญาณก็就会被บันทึกต่อท้ายคอลัมภ์ปัจจุบัน ไล่เรียงกันไปเรื่อย ๆ ดังแสดงในตารางที่ 3.1

ตารางที่ 3.1 ตัวอย่างชุดความแรงสัญญาณจากเสาสัญญาณอ้างอิง ณ พิกัดเก็บข้อมูลที่ N ถึง N+1

RSSI (-dBm)	Point n	Point n+1	Point n+2	Point n+3	Point n+4	Point n+5	Point n+6	Point n+7
A1	-64	-65	-54	-51	-55	-58	-65	-52
A2	-57	-57	-61	-56	-55	-71	-54	-54
A3	-50	-50	-55	-48	-52	-48	-50	-58
A4	-58	-58	-68	-55	-53	-53	-56	-61
A5	-57	-58	-57	-53	-71	-56	-57	-55

```
topfloor_file50 = 'topfloor/topfloor_50.xlsx';
```

รูปที่ 3.12 แสดงรูปแบบการจัดเก็บชุดความแรงสัญญาณในไฟล์นามสกุล .XLSX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
topfloor_read_file50 = xlsread(topfloor_file50);
```

รูปที่ 3.13 แสดงคำสั่งนำเข้าไฟล์นามสกุล .XLSX และอ่านไฟล์พร้อมจัดเก็บในตัวแปรเมทริกซ์ 2 มิติ

ไฟล์ข้อมูลหนึ่งไฟล์จะแทนการเก็บบันทึกข้อมูลในแต่ละรอบ ดังนั้น ไฟล์ข้อมูลทั้งหมดจะมีด้วยกัน 50 ไฟล์ ในแต่ละไฟล์จะประกอบไปด้วย 5 แถว, 121 หลัก สำหรับสนามทดสอบที่ 1 และ 77 หลัก สำหรับสนามทดสอบที่ 2

เครื่องมือที่ใช้สร้างโครงข่ายประสาทเทียมในโปรแกรมแมทแล็บ นั้นมีการจัดรูปแบบของข้อมูลในลักษณะของการจับคู่ข้อมูล โดยจะแบ่งประเภทข้อมูลอินพุตเป็น 2 แบบคือ ข้อมูลดิบที่ใช้ฝึกสอนและข้อมูลเลเบล (Label) โดยที่ข้อมูลดิบที่ใช้ฝึกสอนนั้นคือชุดความแรงสัญญาณของทุกพิกัดภายในหนึ่งพื้นที่ปิด และเลเบลคือข้อมูลที่กำหนดเป็นเป้าหมายในการฝึกสอนโมเดล ตัวเลเบลจะมีรูปแบบเป็นเวกเตอร์ขนาด 13x1 (13 แถว, 1 คอลัมน์) แต่ละแถวจะเป็นตัวแทนของแต่ละพื้นที่ปิด ถ้าต้องการฝึกสอนชุดความแรงสัญญาณของพื้นที่ปิดใด กำหนดให้แถวนั้นมีค่าเป็น 1 และแถวอื่น ๆ มีค่าเป็น 0 ดังแสดงในตารางที่ 3.2

ตารางที่ 3.2 ตัวอย่างการสร้างเลเบลด้วยวิธีวันฮ็อตเอนโค้ดดิ้ง (One-hot encoding) สำหรับฝึกสอนโครงข่ายประสาทเทียม

	Region 1	Region 6	Region 13
Class 1	1	0	0
Class 2	0	0	0
Class 3	0	0	0
Class 4	0	0	0
Class 5	0	0	0
Class 6	0	1	0
Class 7	0	0	0
Class 8	0	0	0
Class 9	0	0	0
Class 10	0	0	0
Class 11	0	0	0
Class 12	0	0	0
Class 13	0	0	1

เทคนิคการสร้างเลเบลเวกเตอร์แบบนี้เราจะเรียกว่าวันฮ็อตเอนโค้ดดิ้ง (one-hot encoding) หลังจากนั้นทำการจัดรูปข้อมูลให้อยู่ในลักษณะเมทริกซ์ 2 มิติ โดยจับคู่ชุดข้อมูลดิบของแต่ละคอลัมน์ให้สอดคล้องกับข้อมูลเลเบลที่ใช้ฝึกสอนโมเดลดังแสดงในตารางที่ 3.3 และ 3.4

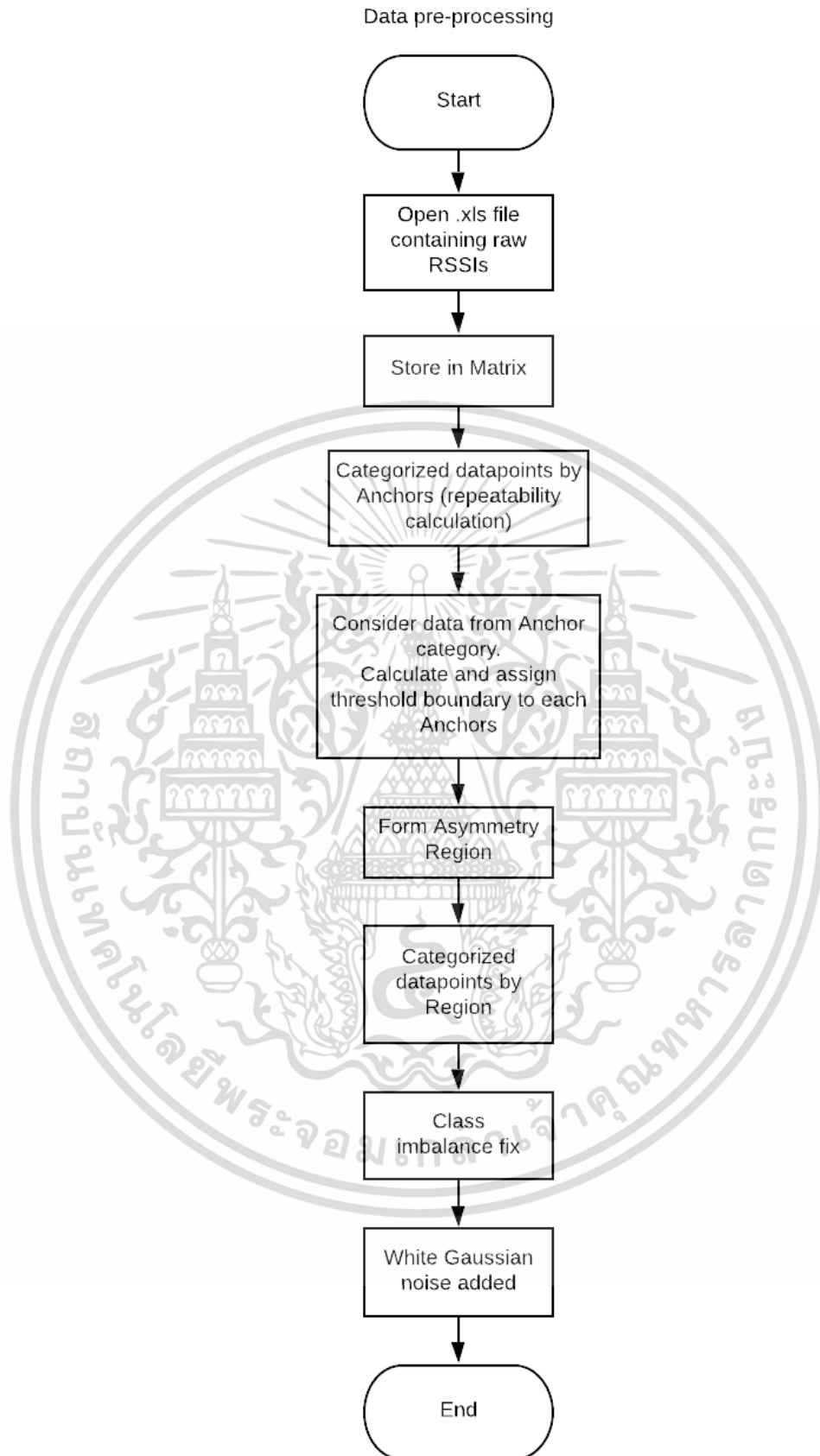
ตารางที่ 3.3 ความแรงสัญญาณของเสาสัญญาณอ้างอิงในสนาม ณ พิกัดเก็บข้อมูลที่ N ถึง N+5

RSSI (-dBm)	Point n	Point n+1	Point n+2	Point n+3	Point n+4	Point n+5
A1	-52	-58	-47	-43	-43	-41
A2	-54	-53	-57	-52	-48	-48
A3	-58	-60	-63	-54	-54	-60
A4	-61	-52	-51	-58	-60	-56
A5	-55	-54	-60	-58	-56	-52

ตารางที่ 3.4 ตัวอย่างการสร้างเลเบลของแต่ละพิกัดข้อมูลสำหรับฝึกสอนโครงข่ายประสาทเทียม

Label	Point n	Point n+1	Point n+2	Point n+3	Point n+4	Point n+5
Class 1	1	1	1	1	1	1
Class 2	0	0	0	0	0	0
Class 3	0	0	0	0	0	0
Class 4	0	0	0	0	0	0
Class 5	0	0	0	0	0	0
Class 6	0	0	0	0	0	0
Class 7	0	0	0	0	0	0
Class 8	0	0	0	0	0	0
Class 9	0	0	0	0	0	0
Class 10	0	0	0	0	0	0
Class 11	0	0	0	0	0	0
Class 12	0	0	0	0	0	0
Class 13	0	0	0	0	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 โฟลว์ชาร์ตแสดงกระบวนการทำงานของวิธีการเตรียมข้อมูลสำหรับสร้างและฝึกสอน
โครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2 การปรับปรุงข้อมูลฝึกสอนด้วยเทคนิคการแก้ไขความไม่สมดุลของคลาสและการเพิ่มสัญญาณรบกวนแบบไวท์เกาส์เซียน

การแก้ไขความไม่สมดุลของคลาส (Class imbalance) เป็นปัญหาหนึ่งที่เกิดขึ้นบ่อยครั้งในการเตรียมความพร้อมของข้อมูลที่ใช้สำหรับฝึกสอนโครงข่ายประสาทเทียม เนื่องจากในการใช้งานจริงนั้น อาจเกิดอุปสรรคบางอย่างที่ทำให้ผู้ทำการทดลองไม่สามารถเก็บบันทึกตัวอย่างข้อมูลมาด้วยจำนวนที่เท่าเทียมกันในทุก ๆ ประเภท เช่นเดียวกับงานวิจัยนี้ที่ขนาดของพื้นที่ปิดที่ไม่สมมาตรนั้นไม่เท่ากันในทุก ๆ โชนจึงทำให้ฝึกที่เก็บบันทึกข้อมูลนั้นมีจำนวนที่ไม่เท่ากัน ส่งผลให้จำนวนข้อมูลในการฝึกสอนไม่เท่าเทียมกันไปด้วย ด้วยข้อมูลที่ใช้ฝึกสอนไม่เท่ากันนั้น โอกาสที่โมเดลจะเรียนรู้จากข้อมูลที่มีจำนวนฝึกสอนมากกว่า ย่อมมากกว่าข้อมูลที่มีจำนวนฝึกสอนน้อย ประสิทธิภาพในการจำแนกข้อมูลที่ต้องการก็จะเอนเอียงไปทางข้อมูลที่มีจำนวนฝึกสอนมาก ด้วยเหตุนี้เอง วิธีการแก้ไขความไม่สมดุลของคลาสที่นิยมใช้กันในการสร้างโครงข่ายประสาทเทียมมีอยู่ด้วยกัน 2 วิธีคือการเพิ่มและการตัดจำนวนข้อมูล

สำหรับการเพิ่มข้อมูลนั้นเป็นการนำคลาสที่มีข้อมูลที่ใช้ฝึกสอนน้อยกว่าคลาสอื่น ๆ มาทำการต่อข้อมูลเพิ่มให้มีจำนวนเท่าเทียมกัน โดยข้อมูลของคลาสที่ต้องการเพิ่มนั้นจะถูกคัดลอกมาในปริมาณที่เหมาะสมแล้วนำมาต่อท้ายกับข้อมูลปัจจุบัน ทำให้ขนาดเมตริกซ์ของข้อมูลทั้งหมดในคลาสนั้นเพิ่มขึ้น เทคนิคนี้ได้รับความนิยมค่อนข้างมากเนื่องจากไม่ทำให้เกิดการสูญเสียข้อมูลดิบ แต่อย่างไรก็ตาม ข้อเสียของเทคนิคนี้คือการคัดลอกข้อมูลเดิมมาต่อท้าย เท่ากับว่าทำให้ข้อมูลที่ใช้ฝึกสอนนั้นซ้ำซ้อน ซึ่งนอกจากจะทำให้โมเดลเรียนรู้แต่ชุดข้อมูลรูปแบบเดิมแล้ว ยังทำให้โมเดลเสียเวลาในการประมวลผลโดยใช้เหตุอีกด้วย

อีกวิธีหนึ่งคือการตัดข้อมูล เป็นการนำคลาสที่มีข้อมูลที่ใช้ฝึกสอนจำนวนมากกว่าคลาสอื่น ๆ มาหั่นข้อมูลทิ้งเพื่อให้เท่าเทียมกัน เทคนิคนี้ง่ายและไม่ซับซ้อน อีกทั้งขนาดข้อมูลที่น้อยทำให้โมเดลลดระยะเวลาในการฝึกสอนอีกด้วย แต่ข้อเสียคือโมเดลจะเสียโอกาสเรียนรู้รูปแบบของข้อมูลส่วนที่ถูกตัดออกไป อีกทั้งจำนวนข้อมูลฝึกสอนที่น้อย อาจจะทำให้ไม่สามารถสร้างโมเดลที่มีประสิทธิภาพในการจำแนกข้อมูลใหม่ได้ไม่สมบูรณ์และยากต่อการจำแนกข้อมูลใหม่ที่โมเดลไม่เคยพบเจอมาก่อน

ในงานวิจัยนี้ได้เลือกใช้เทคนิคการเพิ่มข้อมูล โดยตัดแปลงจากการคัดลอกข้อมูลมาต่อท้ายข้อมูลเดิมตามปกติ มาเป็นการเพิ่มค่าสัญญาณรบกวนแบบไวท์เกาส์เซียนให้กับข้อมูลใหม่ที่ถูกคัดลอกแล้วจึงนำไปต่อท้ายกับข้อมูลปัจจุบัน เทคนิคนี้เป็นที่นิยมอย่างมากในการจำแนกข้อมูลประเภทการประมวลผลภาพ ซึ่งนอกจากจะช่วยลดความซ้ำซ้อนของข้อมูลแล้ว ยังไม่ลดโอกาสในการเรียนรู้ของโมเดล ส่งผลให้โมเดลมีประสิทธิภาพในการทำนายผลได้ดียิ่งขึ้น

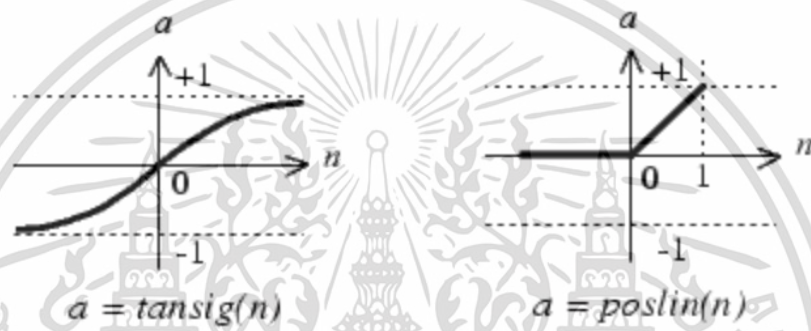
3.6.3 การลดความซ้ำซ้อนของข้อมูลหรือนอร์มัลไลเซชัน (Normalization)

นอร์มัลไลเซชันคือการทำให้ข้อมูลดิบสำหรับป้อนเข้าไปในโมเดลโครงข่ายประสาทเทียมนั้นเหมาะสมสำหรับการประมวลผล ข้อมูลความแรงสัญญาณที่บันทึกได้นั้นเป็นจำนวนเต็มลบที่มีค่าอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่าง -30 ถึง -60 dBm โดยประมาณ ซึ่งค่าเหล่านี้ไม่เหมาะสมสำหรับประมวลผลภายในเครือข่าย เนื่องจากฟังก์ชันกระตุ้น (Activation Function) ของแต่ละนิวรอนในโครงข่ายประสาทเทียมนั้นประมวลผลตัวเลขในช่วงระหว่าง 0 ถึง 1 หรือ -1 ถึง 1 แล้วแต่ชนิดของฟังก์ชัน ดังนั้นเราจึงต้องแปลงให้ข้อมูลดิบที่จะมาประยุกต์ใช้นั้นอยู่ในช่วงที่ฟังก์ชันสามารถประมวลผลได้เสียก่อน มิเช่นนั้นอาจเกิดสถานะ NaN (Not a Number) ซึ่งเป็นผลลัพธ์ที่ไม่สามารถคำนวณได้

ในงานวิจัยนี้ได้มีสมการที่ใช้ในฟังก์ชันกระตุ้นนั้นอยู่สองแบบด้วยกัน คือไฮเปอร์โบลิกแทนเจนท์ซิกมอยด์ (Hyperbolic tangent sigmoid: tansig) และโพซิทีฟลิเนียร์ทรานส์เฟอร์ฟังก์ชัน (Positive Linear Transfer function: poslin) หรือรู้จักในอีกชื่อหนึ่งคือเรคตีไฟเออร์ลิเนียร์ยูนิต (Rectifier Linear Units: ReLu)



รูปที่ 3.15 สมการไฮเปอร์โบลิกแทนเจนท์ซิกมอยด์ และ โพซิทีฟลิเนียร์ทรานส์เฟอร์ฟังก์ชัน สำหรับ tansig นั้น จะมีช่วงของข้อมูลอยู่ระหว่าง -1 ถึง 1 ส่วน poslin จะมีช่วงข้อมูลอยู่ระหว่าง 0 ถึง 1 ซึ่งแต่ละสมการจะมีวิธีการนอร์มัลไลเซชันข้อมูลดังต่อไปนี้

3.6.3.1 การทำนอร์มัลไลเซชันข้อมูลสำหรับ Poslin

- หาค่าความแรงสัญญาณ RSSI ที่ต่ำที่สุดของแต่ละเสาสัญญาณที่อ่านค่าได้ และกำหนดให้เป็น $RSSI_s_m$
- นำค่า RSSI ของเสาสัญญาณนั้น ๆ $RSSI_s$ ทหารด้วย ค่า $RSSI_s_m$ ดังสมการ

$$RSSI_s^* = \frac{RSSI_s}{RSSI_s_m} \tag{3.6}$$

3.6.3.2 การทำนอร์มัลไลเซชันสำหรับ tansig

- หาค่าเฉลี่ย RSSI ทั้งหมดของแต่ละเสาสัญญาณ และกำหนดเป็น $\overline{RSSI_s}$
- หาค่าส่วนเบี่ยงเบนมาตรฐานของ RSSI ทั้งหมดของแต่ละเสาสัญญาณ, σ_{RSSI_s}
- จากนั้นนำค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐานใส่ลงในสมการ

$$RSSI_s^* = (RSSI_s - \overline{RSSI_s}) \cdot \frac{1}{\sigma_{RSSI_s}} \tag{3.7}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.4 การสร้างและฝึกสอนโมเดลจำแนกข้อมูล

จากที่ได้กล่าวไปในข้างต้นแล้วว่าโครงข่ายประสาทเทียมประกอบไปด้วยโหนดนิเวรอนหลายโหนดเข้ามาเชื่อมต่อกันเป็นเครือข่าย เพื่อรับส่งข้อมูลที่ผ่านการประมวลผลภายในแต่ละโหนด พร้อมทั้งปรับแต่งค่าน้ำหนัก (weight) ในตัวเองเพื่อให้ได้มาซึ่งโครงข่ายที่จำแนกข้อมูลได้เหมาะสม พร้อมทั้งจะประยุกต์ใช้งานจริงกับข้อมูลใหม่ที่ไม่เคยพบเจอมาก่อน

การสร้างโครงข่ายประสาทเทียมด้วยเครื่องมือช่วยเหลือในโปรแกรมแมทแลปนั้น เราจำเป็นที่จะต้องกำหนดฟังก์ชันกระตุ้นให้แต่ละเลเยอร์ของเครือข่าย ซึ่งฟังก์ชันกระตุ้นนี้จะเป็นปัจจัยหนึ่งที่มีอิทธิพลอย่างมากต่อผลการจำแนกข้อมูล ฟังก์ชันกระตุ้นแต่ละแบบที่เลือกใช้ในงานวิจัยจะมีรายละเอียดดังต่อไปนี้

3.6.4.1 ไฮเปอร์โบลิกแทนเจนต์ซิกมอยด์ (Hyperbolic tangent sigmoid: tansig)

$$a = \text{tansig}(n) = \frac{2}{(1 + e^{-2n}) - 1} \quad (3.8)$$

สมการทรานสเฟอร์ฟังก์ชันแบบ tansig นั้นเทียบเท่ากับ tanh อาจจะมีควมแตกต่างกันเล็กน้อยในเรื่องความเร็วในการประมวลผลที่มีมากกว่า tanh ส่วนในแง่ของผลลัพธ์จากการคำนวณนั้นมีความแตกต่างกันน้อยมาก เป็นสมการที่ใช้กันเพื่อเริ่มต้นในการสร้างโครงข่ายประสาทเทียมที่ต้องการใช้ความเร็วในการประมวลผลสูงและไม่จำเป็นต้องมีทรานสเฟอร์ฟังก์ชันเฉพาะทาง

3.6.4.2 โพซิทีฟลิเนียร์ทรานส์เฟอร์ฟังก์ชัน (Positive Linear Transfer function: poslin หรือ ReLu)

ทรานสเฟอร์ฟังก์ชันรูปแบบนี้จะส่งค่ากลับมาเป็น เอ๊าท์พุท n ถ้าค่า n มีค่ามากกว่าหรือเท่ากับ 0 และส่งค่ากลับมาเป็นเอ๊าท์พุท 0 ถ้า n มีค่าน้อยกว่าหรือเท่ากับ 0 โพซิทีฟลิเนียร์ทรานส์เฟอร์ฟังก์ชันนั้นจะใช้งานได้ดีและเป็นที่ยอมรับอย่างมากกับโครงข่ายประสาทเทียมที่ประยุกต์ใช้กับงานประมวลผลภาพ เนื่องจากพิกเซลของภาพนั้นส่วนใหญ่จะมีขอบเขตของค่าเป็นจำนวนเต็มศูนย์และเต็มบวก ไม่มีค่าเป็นจำนวนเต็มลบ ทำให้ทรานสเฟอร์ฟังก์ชันนี้ให้ประสิทธิภาพในการประมวลผลที่รวดเร็ว อย่างไรก็ตาม สมการนี้ก็สามารที่จะประยุกต์ใช้กับข้อมูลอื่น ๆ ได้เช่นกัน เพียงแต่ช่วงขอบเขตของค่าอินพุทที่เป็นลบอาจจะถูกปิดให้มีค่าเอ๊าท์พุทเป็นศูนย์ซึ่งทำให้ความละเอียดในการจำแนกข้อมูลขาดหายไป

$$\text{poslin}(n) \begin{cases} = n, n \geq 0 \\ = 0, n \leq 0 \end{cases} \quad (3.9)$$

3.6.4.3 ซอฟท์แมกซ์ (Softmax)

ซอฟท์แมกซ์ฟังก์ชันนั้นเป็นทรานสเฟอร์ฟังก์ชันที่ใช้บ่อยในเลเยอร์ชั้นสุดท้ายของโมเดล (เอ๊าท์พุทเลเยอร์) ในการจำแนกข้อมูลที่สร้างจากโครงข่ายประสาทเทียมซึ่งเป็นโครงข่ายที่ถูกฝึกสอนภายใต้แนวคิดการทำครอสเอนโทรปี (Cross-entropy) โดยแนวคิดดังกล่าวจะทำให้เกิดการ

เปลี่ยนแปลงแบบไม่เป็นเชิงเส้น (non-linear) ของการถดถอยโลจิสติกแบบมัลติโนเมียล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Multinomial logistic regression). สมการที่ 3.10 แสดงสมการซอฟต์แวร์แมกซ์ฟังก์ชัน โดยที่ $x^T w_j$ นั้นคือผลลัพธ์จากการคูณภายใน (inner product) ของ x และ w

$$P(y = j | x) = \frac{e^{x^T w_j}}{\sum_{k=1}^K e^{x^T w_k}} \quad (3.10)$$

ในทางคณิตศาสตร์ ซอฟต์แวร์แมกซ์ฟังก์ชันนั้นจะรับข้อมูลที่ยังไม่ผ่านการนอร์มัลไลเซชันแบบเวกเตอร์ จากนั้นทำการนอร์มัลไลซ์ข้อมูลเพื่อให้สามารถหาการกระจายของความน่าจะเป็น (Probability Distribution) ซึ่งเวกเตอร์ที่นำมาประยุกต์ใช้กับฟังก์ชันซอฟต์แวร์แมกซ์นั้นอาจจะมีตัวประกอบที่มีค่าเป็นลบ หรือบางตัวประกอบที่มีค่ามากกว่า 1 และอาจจะไม่ได้มีผลรวมของตัวประกอบเท่ากับ 1 แต่หลังจากที่นำเวกเตอร์เหล่านั้นมาประยุกต์ใช้กับซอฟต์แวร์แมกซ์ฟังก์ชันแล้ว ตัวประกอบแต่ละตัวของเวกเตอร์จะมีค่าอยู่ระหว่าง 0 ถึง 1 และผลรวมของตัวประกอบทุกตัวจะมีค่าเป็น 1 ฟังก์ชันนี้จะถูกใช้งานบ่อยในโครงข่ายประสาทเทียมเพื่อทำการแมปข้อมูลเอาต์พุท ให้กลายเป็นค่าการกระจายความน่าจะเป็นตามจำนวนคลาสของข้อมูลที่มีในการทำนาย หรืออีกนัยหนึ่งเราอาจจะกล่าวได้ว่า หากซิมมอยด์ฟังก์ชันนั้นให้การจำแนกข้อมูลในลักษณะที่กลุ่มข้อมูลเป็นแบบไบนารี ดังนั้นซอฟต์แวร์แมกซ์ฟังก์ชันก็สามารถจำแนกข้อมูลแบบหลากหลายกลุ่ม (Multi-classification) ได้

ตัวอย่างของการใช้งานซอฟต์แวร์แมกซ์ฟังก์ชัน ถ้าหากเราได้รับอินพุทของเวกเตอร์ [1, 2, 3, 4, 1, 2, 3] จากนั้น ซอฟต์แวร์แมกซ์เอาต์พุทของเราจะได้ [0.024, 0.064, 0.175, 0.475, 0.024, 0.064, 0.175] จะเห็นว่าเวกเตอร์เอาต์พุทจะมีค่า น้ำหนักมากที่สุดที่ตำแหน่งเดียวกับตัวประกอบที่มีค่า '4' ของเวกเตอร์อินพุท ซึ่งหน้าที่ของฟังก์ชันนี้คือ จะทำการไฮไลต์ตัวประกอบที่มีค่ามากที่สุด และจะระงับผลลัพธ์ของค่าตัวประกอบที่ต่ำกว่า ค่าสูงสุด (maximum value) อย่างไรก็ตาม ซอฟต์แวร์แมกซ์นั้นไม่ใช่ฟังก์ชันสำหรับการสเกล (scale) ข้อมูล ดังนั้นถ้าหากเวกเตอร์อินพุทมีค่า [0.1, 0.2, 0.3, 0.4, 0.1, 0.2, 0.3] ซึ่งมีผลรวมเป็น 1.6 ฟังก์ชันซอฟต์แวร์แมกซ์จะให้เอาต์พุทเป็น [0.125, 0.138, 0.153, 0.169, 0.125, 0.138, 0.153] จะเห็นว่าถ้าตัวประกอบของอินพุทเวกเตอร์มีค่าระหว่าง 0 ถึง 1 ฟังก์ชันซอฟต์แวร์แมกซ์ก็จะทำการลดค่าเอาต์พุทลงตามความสัมพันธ์กับเวกเตอร์อินพุท (0.169 นั้นไม่เพียงแต่จะน้อยกว่า 0.457 แต่ยังไม่ถึงสัดส่วนเริ่มต้นของ $0.4/1.6 = 0.25$)

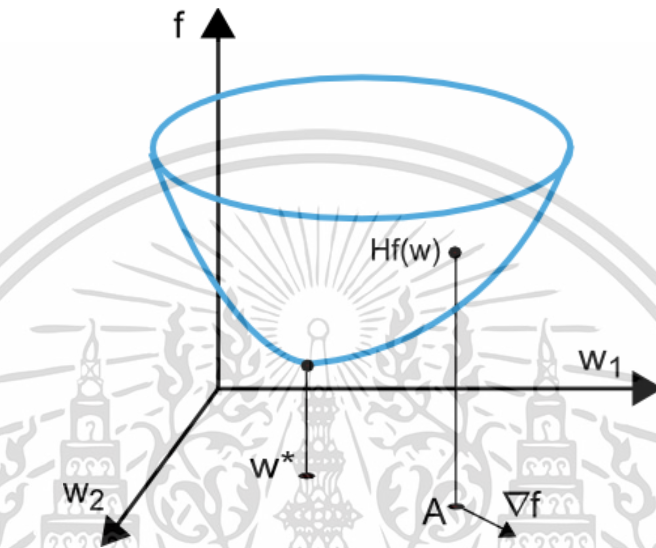
3.6.5 การเลือกอัลกอริทึมสำหรับฝึกสอนโมเดล

ในการสร้างโครงข่ายประสาทเทียมด้วยเครื่องมือช่วยเหลือในโปรแกรมแมทแลบนั้นจะมีส่วนให้เราทำการกำหนดอัลกอริทึมสำหรับเรียนรู้และฝึกสอนโมเดล ซึ่งอัลกอริทึมเหล่านี้จะเป็นปัจจัยหนึ่งที่ส่งผลต่อคุณลักษณะและประสิทธิภาพในการทำงานของโมเดล

การเรียนรู้ปัญหาในโครงข่ายประสาทเทียมนั้นสามารถให้ความหมายได้ในเชิงของการ ลดค่าเป้าหมายให้ต่ำที่สุด (minimization) ของสมการฟังก์ชันสูญเสีย (loss function, f) หรือ ฟังก์ชันสูญเสีย ซึ่งฟังก์ชันดังกล่าวนี้จะประกอบไปด้วยสองเทอมด้วยกันคือ ค่าความคลาดเคลื่อน (error)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่ออนุญาตให้เผยแพร่โดยไม่เสียประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ เทอมเรกูลาไรเซชัน (Regularization) ในเทอมของค่าความคลาดเคลื่อนคือการประเมินว่า โครงข่ายประสาทเทียมจะเหมาะสมกับชุดข้อมูลมากน้อยเพียงใด ในทางกลับกัน เทอมของเรกูลาไรเซชันนั้นจะใช้ในการป้องกันการเกิดพฤติกรรมโอเวอร์ฟิตติ้ง (Overfitting) ฟังก์ชันสูญเสียนั้นจะขึ้นอยู่กับค่าการปรับแต่งพารามิเตอร์ ซึ่งส่วนมากคือค่าไบอัส (Bias) และค่าน้ำหนักของโหนดเชื่อมต่อ (Synaptic weights) ในโครงข่ายประสาทเทียม เราสามารถที่จะจัดกลุ่มตัวแปรเหล่านี้รวมกันเป็น เวกเตอร์น้ำหนัก (weight vector, w) ซึ่งรูปภาพต่อไปนี้เป็นตัวอย่างเป็นตัวอย่างแสดงฟังก์ชันสูญเสีย $f(w)$



รูปที่ 3.16 กราฟแสดงลักษณะความสัมพันธ์ของค่าน้ำหนักและค่าคลาดเคลื่อนในฟังก์ชันสูญเสีย จากภาพจะเห็นได้ว่า ณ จุด w^* คือส่วนที่ต่ำที่สุดของฟังก์ชันสูญเสีย ที่จุด A ใด ๆ เราจะสามารถคำนวณอนุพันธ์ลำดับที่ 1 และ 2 ของฟังก์ชันสูญเสียได้ อนุพันธ์ลำดับที่หนึ่งจะถูกจัดรวมกลุ่มอยู่ในกราเดียนท์เวกเตอร์ (Gradient vector) ซึ่งสามารถเขียนส่วนประกอบของเวกเตอร์ได้ดังต่อไปนี้

$$\nabla_i f(w) = df / dw_i, \quad i = 1, \dots, n \quad (3.11)$$

และอนุพันธ์อันดับสองสามารถเขียนให้อยู่ในรูปของเมทริกซ์ดังต่อไปนี้

$$H_{i,j} f(w) = d^2 f / dw_i \cdot dw_j, \quad i, j = 1, \dots, n \quad (3.12)$$

3.6.5.1 บายเอเซียน เรกูลาไรเซชัน แบ็คพรอพาเกชัน (Bayesian regularization backpropagation)

บายเอเซียน เรกูลาไรเซชัน แบ็คพรอพาเกชันหรือ trainbr (ตัวย่อของอัลกอริทึมนี้ในโปรแกรมแมทแลป) คือฟังก์ชันการฝึกสอนโมเดลโครงข่ายประสาทเทียมที่ทำหน้าที่ในการอัปเดตค่าน้ำหนัก (weights) และไบอัส (bias) ตามหลักการของการทำลาเวนเบิร์ก-มาควอดท์ ออพติไมเซชัน (Levenberg-Marquardt optimization) ซึ่งระบบจะทำการลดค่าสูญเสียของข้อมูลให้ต่ำที่สุดในรูปแบบของค่าคลาดเคลื่อนยกกำลังสองและค่าน้ำหนัก จากนั้นจึงคำนวณผลลัพธ์ที่เหมาะสมที่จะทำให้สามารถปรับแต่งโครงข่ายให้มีความสามารถในการเข้ากับข้อมูลที่ไม่เคยพบเจอมาก่อนหรือการ

ทำเจเนอรัลไลเซชัน (Generalization) ที่ดีได้ กระบวนการดังกล่าวเรียกว่าบายเอเซียน เรกูลาไรเซชัน เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติของวิธีการนี้คือการตั้งค่าตัวแปรสำหรับหยุดการตรวจสอบข้อมูลระหว่างฝึกสอน (Validation stops) นั้นจะถูกยกเลิกออกไปแทนที่โดยใช้ค่า `max_fail = inf` ซึ่งถูกตั้งเป็นค่าเริ่มต้น (default) อยู่แล้ว จึงส่งผลให้การฝึกสอนโมเดลนั้นดำเนินต่อไปเรื่อย ๆ จนกว่าจะปรับแต่งพารามิเตอร์จนให้ค่าคลาดเคลื่อนและค่าน้ำหนักที่ดีที่สุด อย่างไรก็ตาม บางกรณีค่าน้ำหนักและค่าไบอัสที่ดีที่สุดนั้นอาจจะคำนวณได้ก่อนที่หมดเวลาฝึกสอนโมเดลถ้าหากเราเปิดใช้งานการตั้งค่าสำหรับหยุดการฝึกสอนโดยตั้งค่าให้ตัวแปร `max_fail` เท่ากับ 6 หรือเป็นเลขจำนวนเต็มที่มีค่ามากกว่านี้ `trainbr` สามารถที่จะฝึกสอนโครงข่ายใด ๆ ก็ได้ ตรวจจับที่ค่าน้ำหนัก, อินพุตของเครือข่าย และสมการทรานสเฟอร์ฟังก์ชัน ของโครงข่ายนั้น ๆ มีฟังก์ชันอนุพันธ์ประกอบอยู่

บายเอเซียน เรกูลาไรเซชันจะทำการลดค่าคลาดเคลื่อนกำลังสองให้ต่ำลงภายในสมการผลรวมเชิงเส้น (Linear combination) และปรับค่าน้ำหนัก อีกทั้งยังปรับแต่งสมการผลรวมเชิงเส้นเพื่อให้ได้ผลลัพธ์การฝึกสอนที่ดีที่สุดและมีผลลัพธ์ของการเจเนอรัลไลเซชันที่มีคุณภาพ

บายเอเซียน เรกูลาไรเซชันนั้นมีพื้นฐานมาจากอัลกอริทึมลาเวนเบิร์ก-มาควอดท์ ในส่วนการทำ การเรียนรู้แบบย้อนกลับ (Backpropagation) นั้นจะใช้ในการคำนวณจาโคเบียนเมทริกซ์ (Jacobian matrix, JX) เพื่อให้สอดคล้องกับค่าน้ำหนักค่าไบอัสแสดงดังสมการต่อไปนี้

$$jj = jX \cdot jX \quad (3.25)$$

$$je = jX \cdot E \quad (3.26)$$

$$dX = -\frac{(jj + I * \mu)}{je} \quad (3.27)$$

โดยที่ E นั้นคือเทอมของค่าเอเรอ (error) ทั้งหมด และ I คือเมทริกซ์เอกลักษณ์ การฝึกสอนโมเดลจะหยุดกระบวนการทำงานก็ต่อเมื่ออัลกอริทึมพบเจอกับเหตุการณ์ดังต่อไปนี้

- จำนวนรอบการทำงานหรืออีพ็อก (epochs เรียกได้อีกอย่างว่า repetitions) ถึงจุดสูงสุด
- เวลาที่ใช้ในการฝึกสอนเกินกว่าค่าที่ได้กำหนดไว้
- ประสิทธิภาพของโมเดลถึงจุดที่ได้ตั้งค่าเป้าหมาย (goal)
- ค่าของกราเดียนท์ที่คำนวณได้นั้นต่ำกว่าค่าตัวแปร `min_grad` ที่กำหนด
- ตัวแปร `mu` มีค่าเกินกว่าค่าของตัวแปร `mu_max`

3.6.5.2 สเกลคอนจูเกต กราเดียนท์ แบริคพรอพาเกชัน (Scaled conjugate gradient backpropagation)

สเกลคอนจูเกต กราเดียนท์ แบริคพรอพาเกชัน (`trainscg`) คือฟังก์ชันการฝึกสอนโครงข่ายประสาทเทียมที่มีการอัปเดตค่าน้ำหนักและค่าไบอัสบนหลักการของคอนจูเกตกราฟาเดียนท์ อัลกอริทึมนี้สามารถที่จะฝึกสอนโครงข่ายประสาทเทียมใด ๆ ก็ได้ ตรวจจับที่โครงข่ายนั้นมีค่าน้ำหนัก, อินพุตของโครงข่าย, และทรานส์เฟอร์ฟังก์ชันที่มีส่วนประกอบของอนุพันธ์ การทำแบริคพรอพาเกชันนั้นจะใช้ในการคำนวณหาอนุพันธ์ของประสิทธิภาพโครงข่าย (แทนด้วยตัวแปร `perf`) เทียบกับค่าน้ำหนักและค่าไบอัส การฝึกสอนโครงข่ายจะหยุดก็ต่อเมื่อเกิดเหตุการณ์ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จำนวนรอบการทำงานหรืออีพ็อกถึงจุดสูงสุด
- เวลาที่ใช้ในการฝึกสอนเกินกว่าค่าที่ได้กำหนดไว้
- ประสิทธิภาพของโมเดลถึงจุดที่ได้ตั้งค่าเป้าหมาย
- ค่าของกราฟเดียนท์ที่คำนวณได้นั้นต่ำกว่าค่าตัวแปร min_grad ที่กำหนด
- ตัวแปร μ มีค่าเกินกว่าค่าของตัวแปร μ_{max}
- จำนวนค่าประสิทธิภาพของการตรวจสอบโครงข่ายระหว่างฝึกสอน (Validation performance) เพิ่มขึ้นมากกว่าตัวแปร max_fail ที่ได้กำหนดไว้

3.6.6 การทดสอบไฮเปอร์พารามิเตอร์ (Hyper-parameter testing)

การทดสอบไฮเปอร์พารามิเตอร์ คือการทดสอบประสิทธิภาพในการทำงานของโครงข่ายประสาทเทียมที่ผ่านการฝึกสอนมาแล้วว่าการตั้งค่าที่ประยุกต์ใช้กับโครงข่ายนั้นสามารถสร้างโครงข่ายที่ให้ผลลัพธ์ในการจำแนกข้อมูลได้ถูกต้องและเหมาะสมทั้งกับข้อมูลที่ใช้ฝึกสอนและข้อมูลจริงหรือไม่ โดยการทดสอบนี้เปรียบได้กับการลองผิดลองถูก ผู้ทำการทดสอบจะต้องทำการเลือก และทดลองการตั้งค่าไปเรื่อย ๆ จนกว่าจะได้โครงข่ายที่ดีที่สุด

เนื่องจากการสร้างโครงข่ายประสาทเทียมที่ได้นั้น ไม่ได้มีหลักการตายตัวแต่อย่างใด หากแต่จะขึ้นอยู่กับความรู้ความเข้าใจในรูปแบบการทำงานต่าง ๆ ของโครงข่าย รวมไปถึงลักษณะของชุดข้อมูลที่จะนำมาประยุกต์ใช้ ดังนั้น เพื่อเป็นแนวทางปฏิบัติ [40], [42], [41] จึงได้มีการกำหนดขอบเขตแนวคิดในการเลือกการตั้งค่าสำหรับสร้างโครงข่ายประสาทเทียมเบื้องต้น ซึ่งแนวคิดนี้จะมีชื่อเรียกว่ากฎนิ้วโป้ง (Rule of Thumb) ซึ่งจะมีรายละเอียดสำคัญแสดงดังตารางที่ 3.5

ตารางที่ 3.5 ตารางแสดงรายละเอียดกฎนิ้วโป้งสำหรับกำหนดโครงสร้างเครือข่ายประสาทเทียม

การเลือกตัวแปร	ชั้นอินพุตควรจะต้องมีโหนดตามจำนวนอินพุตที่ถูกป้อนเข้ามาในเครือข่าย ซึ่งถ้าหากจำนวนอินพุตมีมาก อาจจะใช้เทคนิคการลดขนาดหรืออัลกอริทึมในการตัดสินใจเลือกใช้อินพุตได้ตามความเหมาะสม
การกำหนดโทโพโลยีของเครือข่าย	โอกาสมากกว่า 90% ในการเลือกใช้จำนวนฮิดเด้นเลเยอร์ 1 ชั้น และ 10% ในการเลือกฮิดเด้นเลเยอร์ 2 ชั้น อย่างไรก็ตาม เราสามารถเพิ่มจำนวนชั้นฮิดเด้นเลเยอร์ได้มากกว่านี้แต่ก็จะเพิ่มโอกาสเสี่ยงที่จะเกิดการโอเวอร์ฟิตติ้งได้
จำนวนของโหนด	ควรมีจำนวนฮิดเด้นโหนดเท่ากับจำนวนคลาสโดยอัตราส่วน 1 ต่อ 1 เป็นอย่างน้อย หรือประมาณ 0.5 ถึง 3 เท่าของจำนวนอินพุตโหนด โดยมีหลักเกณฑ์ในการพิจารณาที่เรียกว่า พีรามิดสมมาตร ได้กล่าวไว้ว่าถ้ามีจำนวนโหนดอินพุต m โหนด และจำนวนโหนดเอาต์พุต n โหนด ดังนั้น โหนดฮิดเด้นควรมีจำนวนเท่ากับ $\sqrt{m \cdot n}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กฎของบอม-เฮาส์เลอร์	ใช้ในการประมาณจำนวนนิวรอลอย่างคร่าว ๆ ในชั้นฮิดเด้นเลเยอร์ว่าควรจะมีกี่โหนด [41] โดยจะมีความสัมพันธ์ของแต่ละปัจจัยดังสมการต่อไปนี้ $N_{Hidden} \leq \frac{N_{train} \cdot E_{tolerance}}{N_{pts} \cdot N_{outputs}}$
---------------------	---

อย่างไรก็ตาม เพื่อให้ลดระยะเวลาในการลองผิดลองถูก ในงานวิจัยนี้จึงได้กำหนดขอบเขตการตั้งค่าโครงข่ายประสาทเทียมให้อยู่ภายในรายการดังตารางต่อไปนี้

ตารางที่ 3.6 การตั้งค่าไฮเปอร์พารามิเตอร์สำหรับสร้างและฝึกสอนโครงข่ายประสาทเทียม

อัตราการฝึกสอนโครงข่าย (Learning rate)	[0.1 0.01 0.001 0.0001 0.00001]
ค่าลงโทษการฝึกสอนแบบแอลทู เรกูลาไรเซชัน (L2 Regularization penalty value)	[0.1 0.01 0.001 0.0001 0.00001]
ชนิดอัลกอริทึมสำหรับฝึกสอน (Learning/Training algorithm)	trainscg, trainrp
ฟังก์ชันกระตุ้น (Activation function)	poslin, tansig
จำนวนชั้นของฮิดเด้นเลเยอร์ (Number of hidden layers)	1-2
จำนวนโหนดของฮิดเด้นโหนด (Number of hidden nodes)	20, 40, 60, 80

การตั้งค่าในตารางจะถูกเลือกมาหั่วข้อละหนึ่งแบบ โดยจะสลับการตั้งค่าไปมาให้ครบทุกรูปแบบการจับคู่ เพื่อให้ได้การตั้งค่าของโครงข่ายที่ให้ประสิทธิภาพในการจำแนกข้อมูลที่ดีที่สุด

3.6.7 วิธีการทำครอสวาไลเดชัน 10 ชุด (10-fold cross validation)

อย่างที่ทราบกันดีว่าโมเดลโครงข่ายประสาทเทียมที่สร้างจะมีประสิทธิภาพในการจำแนกข้อมูลที่ดีก็ต่อเมื่อ ถูกฝึกสอนด้วยวิธีที่เหมาะสมและด้วยข้อมูลฝึกสอนที่เหมาะสม หากล้มเหลวในปัจจุบันใดปัจจัยหนึ่ง ก็อาจจะส่งผลให้โมเดลนั้นไม่มีความสามารถในการจำแนกข้อมูลขณะใช้งานจริงได้ดีเท่าที่ควร ดังนั้นข้อมูลที่ใช้ฝึกสอนจึงมีความสำคัญและควรที่จะให้โครงข่ายสามารถประยุกต์ใช้ชุดข้อมูลได้อย่างครอบคลุม ข้อมูลที่ใช้ฝึกสอนโครงข่ายประสาทเทียมนั้น ในความเป็นจริงแล้ว สามารถแบ่งออกได้เป็น 3 กลุ่มด้วยกันคือ ข้อมูลที่ใช้ฝึกสอน, ข้อมูลที่ใช้ตรวจสอบ และ ข้อมูลที่ใช้ทดสอบ

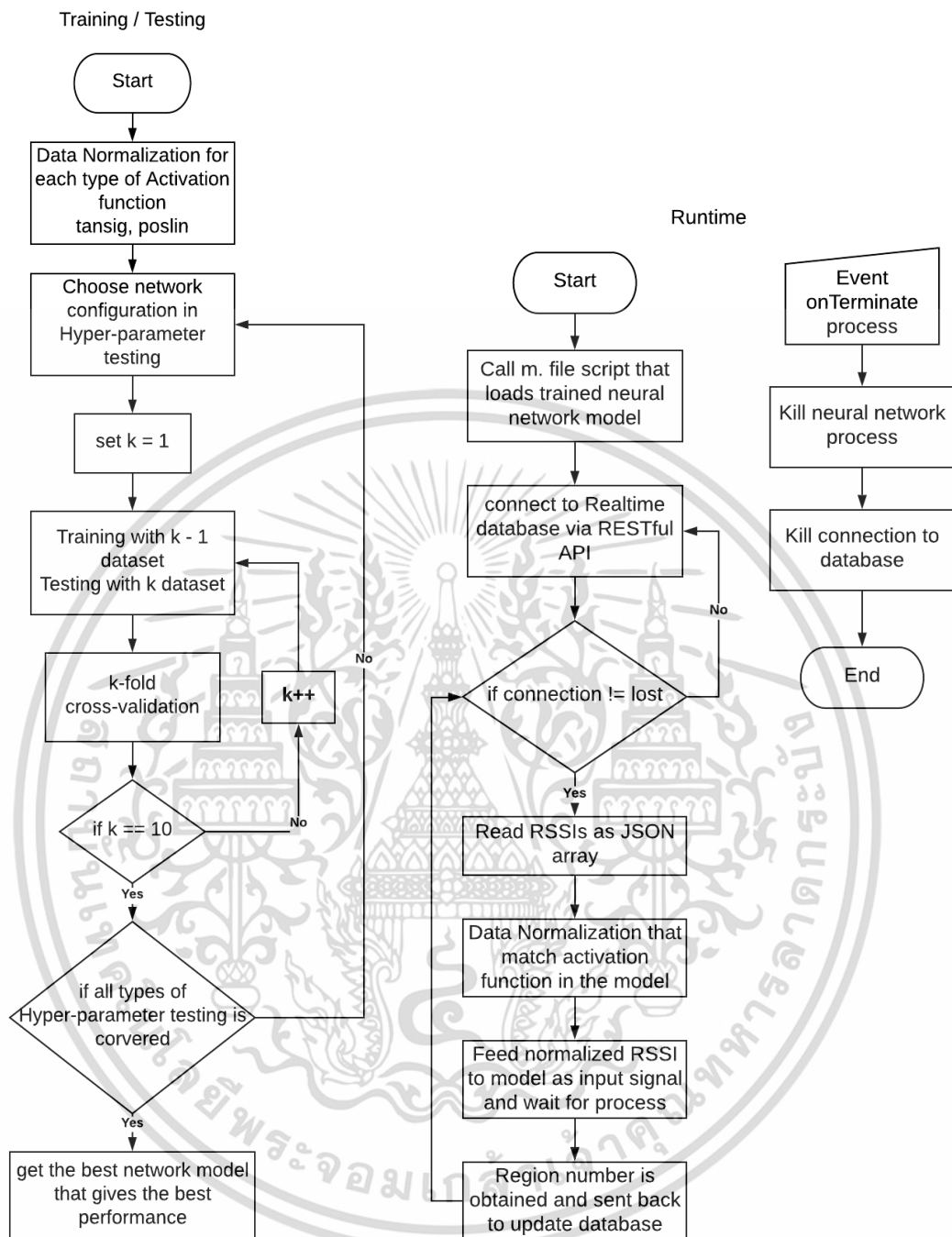
1. ข้อมูลที่ใช้ฝึกสอน (Training set) คือข้อมูลที่ใช้ฝึกสอนโครงข่ายประสาทเทียมให้สามารถจำแนกข้อมูลได้ตามคลาสที่ได้กำหนดไว้ในเวลา
2. ข้อมูลที่ใช้ตรวจสอบ (Validation set) คือข้อมูลที่ใช้ในการตรวจสอบหรือประเมินความสามารถของโมเดลขณะทำการฝึกสอน อีกทั้งยังเป็นชุดข้อมูลที่ใช้ในการกำหนดทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของการประมวลผลว่า ควรจะดำเนินการฝึกสอนต่อหรือหยุดฝึกสอนเมื่อใด ป้องกันการสร้างโมเดลที่ไม่เหมาะสม โดยทั่วไปข้อมูลที่ใช้ตรวจสอบนั้นจะถือว่าเป็นส่วนหนึ่งของข้อมูลที่ใช้ฝึกสอนโมเดล

3. ข้อมูลที่ใช้ทดสอบ (Testing set) คือข้อมูลที่ใช้ทดสอบความสามารถในการจำแนกข้อมูลขณะใช้งานจริง โมเดลจะไม่เคยพบเจอและเรียนรู้ข้อมูลชุดนี้มาก่อน และผลลัพธ์ที่ได้จากการจำแนกข้อมูลประเภทนี้ จะเป็นตัวชี้วัดว่าความสามารถในการจำแนกข้อมูลของโมเดลหลังจากถูกฝึกสอนมาแล้วนั้นมีความน่าเชื่อถือมากน้อยเพียงใด

อย่างไรก็ตาม การแบ่งชุดข้อมูลสำหรับฝึกสอนนั้น ถ้าหากแบ่งข้อมูลฝึกสอน และข้อมูลทดสอบ ออกเป็น 2 ส่วนเท่า ๆ กัน โมเดลจะเสียโอกาสในการเรียนรู้ชุดข้อมูลที่ใช้ทดสอบไป ดังนั้นเพื่อสร้างโอกาสในการเรียนรู้ให้กับโครงข่ายประสาทเทียม จึงได้มีแนวทางการแบ่งส่วนของข้อมูลทั้งหมดออกเป็น k ส่วนเท่า ๆ กันและให้ข้อมูลจำนวน i ส่วน (โดยที่ $i = k-1$) นั้นเป็นข้อมูลที่ใช้สำหรับฝึกสอน ส่วนข้อมูลที่เหลือนั้นกำหนดให้เป็นข้อมูลสำหรับทดสอบ แล้วทำการฝึกสอนโครงข่ายประสาทเทียมด้วยสัดส่วนของข้อมูลดังกล่าว เมื่อฝึกสอนเสร็จสิ้น ให้เก็บผลลัพธ์การทดสอบเอาไว้ จากนั้นสับเปลี่ยนชุดข้อมูลฝึกสอนกับชุดข้อมูลทดสอบ หมุนเวียนจนครบทุกส่วน เป็นจำนวน k รอบ จากนั้นนำผลลัพธ์จากการทดสอบทั้ง k ครั้ง มาทำการหาค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน เพื่อนำมาใช้เป็นเครื่องชี้วัดความแม่นยำในการจำแนกประเภทข้อมูลของโครงข่ายที่ได้ฝึกสอน โดยกระบวนการฝึกสอนและทดสอบโมเดลโครงข่ายประสาทเทียมนี้สามารถแสดงได้ ดังไดอะแกรมในรูปที่ 3.25 (ซ้าย)



รูปที่ 3.17 ไดอะแกรมแสดงขั้นตอนการฝึกสอนและทดสอบโมเดลโครงข่ายประสาทเทียม (ซ้าย)

ไดอะแกรมแสดงขั้นตอนการใช้งานโมเดลโครงข่ายประสาทเทียมระหว่างการใช้งานจริง (ขวา)

3.6.8 ค่าความผิดพลาดเจเนอรัลไลเซชัน แบบ โอเวอร์ฟิตติ้ง และอันเดอร์ฟิตติ้ง (Generalization error: Over-fitting and Under-fitting)

ในการทำแมชชีนเลิร์นนิง อย่างหนึ่งที่ต้องพึงระลึกไว้เสมอคือ ทำอย่างไรที่เราจะสามารถทำให้โมเดลที่สร้างขึ้นมานั้นมีความสามารถในการเจเนอรัลไลเซชัน (Generalization) หรือการทำให้โมเดล

สามารถเข้ากันได้กับข้อมูลที่ตัวโมเดลไม่เคยพบเห็นหรือเรียนรู้มาก่อน เราจะเรียกความผิดพลาดในการทำนายผลลักษณะนี้ว่าเจเนอรัลไลเซชันเอเรอ (Generalization error)

เจเนอรัลไลเซชันเอเรอนั้นเป็นตัวชี้วัดความคลาดเคลื่อนในการจำแนกข้อมูลซึ่งเราต้องการให้ค่าคลาดเคลื่อนนั้นน้อยที่สุดเท่าที่จะเป็นไปได้ การที่โมเดลให้ความคลาดเคลื่อนในการจำแนกข้อมูลได้น้อยที่สุดนั้น เราจะเรียกพฤติกรรมนี้ว่ากูดฟิตติ้ง (Good-fitting) หมายความว่าโมเดลสามารถเข้าใจถึงรูปแบบโครงสร้างของข้อมูลได้เป็นอย่างดีและสามารถปรับใช้กับข้อมูลใหม่ที่ไม่เคยพบเจอมาก่อนได้อย่างเหมาะสม แต่ในอีกทางหนึ่ง หากโมเดลให้ความคลาดเคลื่อนในการจำแนกข้อมูลที่มาก อาจเกิดจากการฝึกสอนที่ไม่เหมาะสม ซึ่งเราสามารถแบ่งประเภทของพฤติกรรมการให้ความคลาดเคลื่อนในการจำแนกได้สองแบบด้วยกันคือโอเวอร์ฟิตติ้ง (Over-fitting) และอันเดอร์ฟิตติ้ง (Under-fitting)

3.6.8.1 โอเวอร์ฟิตติ้ง (Over-fitting)

คือพฤติกรรมในการเรียนรู้ของโมเดลที่ยึดติดอยู่กับข้อมูลฝึกสอนมากเกินไป อาจเกิดขึ้นจากการออกแบบโครงข่ายที่มีความซับซ้อนมาก เช่น กำหนดฮิดเดนโนด / ฮิดเดนเลเยอร์จำนวนมากหรืออัลกอริทึมในการฝึกสอนไม่เหมาะสมกับชุดข้อมูล เป็นต้น ทำให้การปรับค่าน้ำหนักและค่าไบอัสของโครงข่ายนั้นเข้ากับข้อมูลที่ใช้ฝึกสอนเพียงอย่างเดียว การเกิดโอเวอร์ฟิตติ้งจะทำให้โมเดลไม่สามารถจำแนกข้อมูลที่ไม่เคยพบเจอมาก่อนได้เลย ถึงแม้ว่าข้อมูลบางจุดจะใกล้เคียงกับข้อมูลที่ใช้ฝึกสอนมากก็ตาม

3.6.8.2 อันเดอร์ฟิตติ้ง (Under-fitting)

คือพฤติกรรมในการเรียนรู้ของโมเดลอีกรูปแบบหนึ่งที่ไม่สามารถค้นหาหรือเรียนรู้รูปแบบโครงสร้างของข้อมูลฝึกสอนได้เลย อาจเกิดขึ้นจากการที่โมเดลมีการเลือกใช้ฟังก์ชันกระตุ้น (Activation function) ที่ง่ายและไม่ซับซ้อนจนเกินไป หรือการกำหนดฮิดเดนโนด / ฮิดเดนเลเยอร์นั้นมีจำนวนน้อยเกินกว่าที่โครงข่ายจะสามารถประมวลผลรูปแบบของข้อมูลได้ ทำให้ในการใช้งานจริงเกิดความคลาดเคลื่อนในการจำแนกข้อมูลได้เช่นกัน

3.6.9 การสร้างเส้นทางการเคลื่อนที่ของหุ่นยนต์ด้วยการเรียนรู้แบบเสริมกำลัง

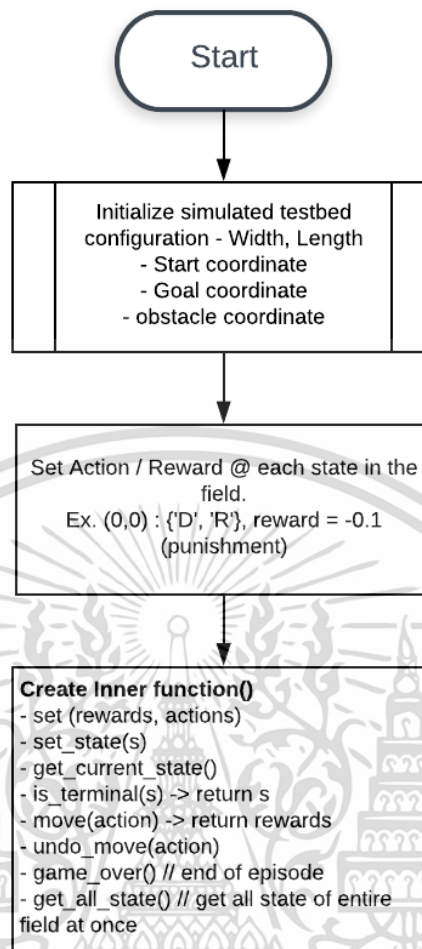
หัวข้อนี้จะกล่าวถึงการสร้างเส้นทางการเคลื่อนที่ของหุ่นยนต์ด้วยการเรียนรู้แบบเสริมกำลัง ซึ่งเป็นเทคนิคที่นิยมใช้กันอย่างแพร่หลายในหลาย ๆ งานวิจัย เพื่อค้นหาเส้นทางไปยังตำแหน่งเป้าหมายที่สั้นที่สุดหรือให้ประสิทธิภาพมากที่สุด การเรียนรู้แบบเสริมกำลังที่ใช้ในงานวิจัยนี้จะหยิบยกอัลกอริทึมการเรียนรู้แบบคิว (Q-learning) มากกล่าวถึงเป็นหลัก เนื่องจากการเรียนรู้แบบคิวเป็นอัลกอริทึมที่ให้ความยืดหยุ่นสูง อีพเทคค่าคิว (Q-value) อย่างต่อเนื่องโดยไม่จำเป็นต้องจบรอบการทำงานในหนึ่งเอพิซอด (Episode) อีกทั้งยังไม่จำเป็นต้องพึ่งพาข้อมูลจากสภาพแวดล้อมทั้งหมดมาทำการคำนวณ ทำให้อัลกอริทึมนี้ประมวลผลได้อย่างรวดเร็ว ซึ่งส่งผลให้หุ่นยนต์สามารถคำนวณหาเส้นทางการเคลื่อนที่ได้ในระยะเวลาอันสั้น ใกล้เคียงเรียลไทม์ (Real-time) มากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.9.1 การสร้างสนามทดสอบแบบจำลองด้วยเทอร์เทิลไลบรารี (Turtle)

ในงานวิจัยนี้ได้จำลองสนามทดสอบทั้ง 2 แบบ (สนามขนาด 10m x 10m และสนามขนาด 10m x 6m) ขึ้นมาเพื่อประเมินความสามารถและผลกระทบของการใช้แนวคิดในการบอกตำแหน่งแบบเออาร์บีแอล (ARBL) ต่อการสร้างเส้นทางเพื่อหลบหลีกสิ่งกีดขวางในสนามว่าวิธีที่นำเสนอขึ้นเพิ่มหรือลดทอนความสามารถในการสร้างเส้นทางของหุ่นยนต์ได้มากน้อยเพียงใดเมื่อเปรียบเทียบกับวิธีการสร้างเส้นทางแบบดั้งเดิม

เทอร์เทิลเป็นหนึ่งในไลบรารีพื้นฐานของภาษาไพธอน (Python) สำหรับใช้ในการวาดภาพกราฟฟิคอย่างง่าย โดยจะมีเคอร์เซอร์ที่มีรูปร่างเป็นหัวลูกศรหรือมีรูปร่างเป็นเต่าแล้วแต่รุ่นของไลบรารีและการตั้งค่า ตัวเคอร์เซอร์จะเปรียบเสมือนกับหัวปากกาหรือพู่กันที่ใช้ในการวาดภาพกราฟฟิคแบบสองมิติลงในพื้นที่ว่างที่เราสร้างขึ้นในโปรแกรม เราสามารถกำหนดการตั้งค่าพื้นฐานของตัวเคอร์เซอร์ได้ ตัวอย่างเช่น สี ลักษณะของเส้นวาด ทิศทางกันหมุนเคอร์เซอร์ ความไวในการเคลื่อนที่ของเคอร์เซอร์ พิกัดการวางตำแหน่งเคอร์เซอร์ หรือการสร้างรูปเรขาคณิต เป็นต้น จากความสามารถของเทอร์เทิลตามที่กล่าวมานั้น ด้วยเหตุนี้เราจึงกำหนดให้เคอร์เซอร์ (Cursor) ของเทอร์เทิลทำหน้าที่เป็นเอเจนต์ (Agent) ในการเรียนรู้แบบเสริมกำลัง (Reinforcement Learning) ด้วยอัลกอริทึมการเรียนรู้แบบคิว หรืออาจกล่าวได้อีกอย่างหนึ่งว่าเคอร์เซอร์จะทำหน้าที่เปรียบเสมือนหุ่นยนต์ที่เคลื่อนที่ในสนามทดสอบจริง



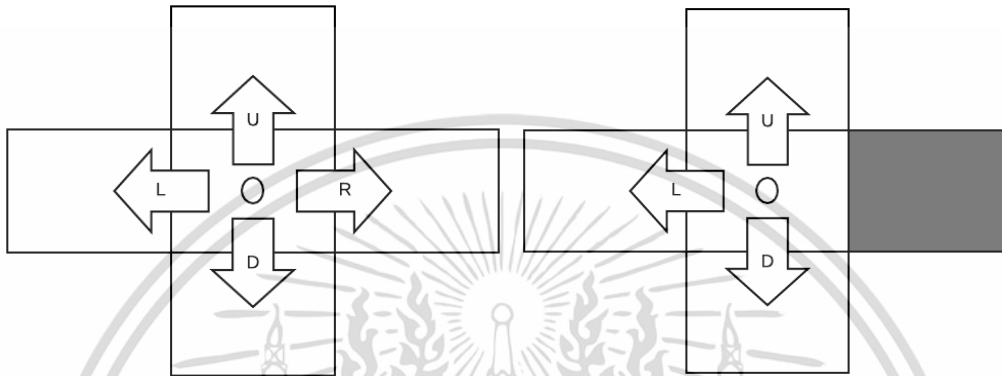
รูปที่ 3.18 ไดอะแกรมแสดงขั้นตอนการเตรียมสภาพแวดล้อมสำหรับฝึกสอนเอเจนต์ด้วยการเรียนรู้แบบคิวในการเรียนรู้แบบเสริมกำลัง

3.6.9.2 เกณฑ์กติกาสำหรับหุ่นยนต์เมื่อเคลื่อนที่ในสนามทดสอบ

เพื่อให้ง่ายต่อการออกแบบและควบคุมการทำงานของเอเจนต์ที่ทำหน้าที่เสมือนหุ่นยนต์เคลื่อนที่จริงในสนามทดสอบนั้น จะถูกกำหนดการกระทำกิริยาใด ๆ ให้มีขอบเขตเพียง 4 รูปแบบการเคลื่อนไหวเท่านั้น คือ ตรงไปข้างหน้า ถอยกลับ ตรงไปทางซ้าย และตรงไปทางขวา ซึ่งข้อมูลเหล่านี้จะถูกแทนด้วยตัวอักษร {U, D, L, R} ตามลำดับและจัดเก็บอยู่ในตัวแปรอาร์เรย์เช่นเดียวกันกับสนามทดสอบ ในแต่ละช่องตารางจะถูกกำหนดการเคลื่อนไหวของเอเจนต์ไว้ว่าสามารถเคลื่อนที่ในทิศทางใดได้บ้างดังแสดงในรูปที่ 3.27

ถ้าหากช่องตารางช่องใดมีตำแหน่งข้างเคียงเป็นสิ่งกีดขวางหรือขอบของสนาม ทิศทางการเคลื่อนที่ทิศนั้นจะไม่ถูกอนุญาตให้หุ่นยนต์เลือกใช้งาน ยิ่งไปกว่านั้น แต่ละช่องตารางจะถูกกำหนดค่าตอบแทนจากการ มาอยู่ในสถานะนี้ ซึ่งค่าตอบแทนดังกล่าวจะเป็นหนึ่งในปัจจัยหลักที่ใช้สอนและควบคุมทิศทางการเรียนรู้แก่เอเจนต์ในทางปฏิบัติ นั้น ค่าตอบแทนเป็นเพียงจำนวนเต็มหรือทศนิยมทางคณิตศาสตร์ที่เราสามารถกำหนดให้ค่าเป็นบวก ศูนย์ หรือลบก็ได้ โดยส่วนมากแล้วเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งหรือสถานะที่เป็นเป้าหมายของการเรียนรู้จะถูกกำหนดให้มีค่าตอบแทนมากที่สุด ซึ่งเราจะเรียกค่าตอบแทนลักษณะนี้ว่า “รางวัล” ขณะที่ตำแหน่งหรือสถานะที่ไม่ต้องการให้เกิดขึ้นกับหุ่นยนต์ จะถูกกำหนดให้มีค่าน้อยที่สุด ซึ่งเราจะเรียกว่า “ค่าลงโทษ” ส่วนตำแหน่งอื่น ๆ ที่เหลือจะถูกกำหนดให้มีค่าเป็นจำนวนเต็มบวกใด ๆ หรือค่าศูนย์ก็ได้ หุ่นยนต์จะเรียนรู้จากค่าเหล่านี้ในการสร้างกระบวนการแก้ปัญหา เพื่อให้ได้มาซึ่งเป้าหมายซึ่งเป็นสถานะที่มีค่ารางวัลสูงสุด พร้อมทั้งหลีกเลี่ยงการเกิดสถานะที่หุ่นยนต์จะได้รับบทลงโทษ

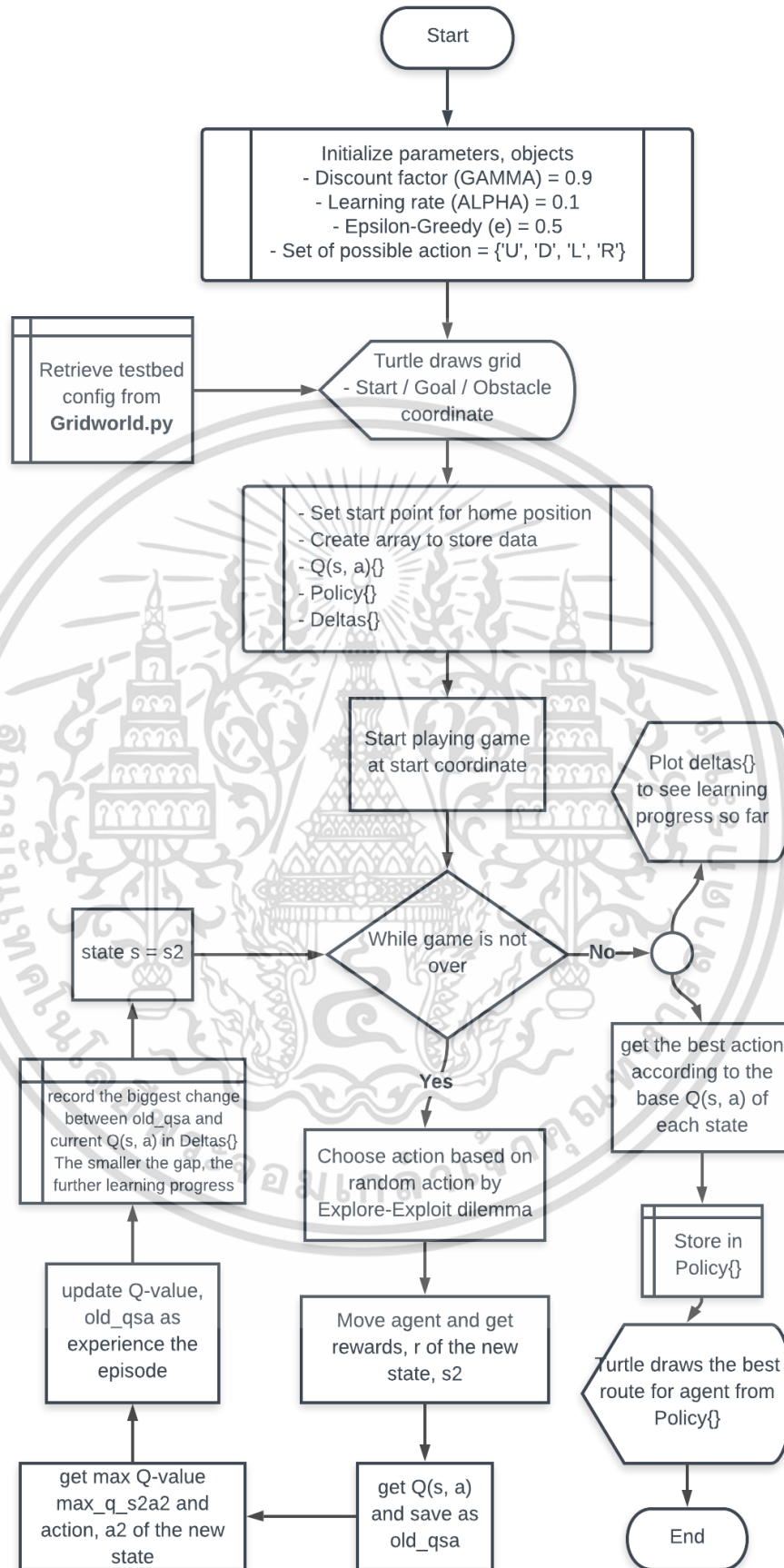


รูปที่ 3.19 แสดงชุดการกระทำของเอเจนต์ในหนึ่งช่องตารางสถานะภายในสนามทดสอบ

อย่างไรก็ตาม ในงานวิจัยนี้ ตำแหน่งต่าง ๆ ในช่องตารางจะถูกกำหนดให้มีค่าตอบแทนของสถานะเท่ากับ -0.1 ยกเว้นตำแหน่งที่เป็นเป้าหมายที่กำหนดให้มีค่าตอบแทนเท่ากับ $+10$ ดังนั้นทุกช่องตารางที่เอเจนต์เคลื่อนที่ไป ตัวเอเจนต์จะได้รับการลงโทษ -0.1 ทุก ๆ ครั้ง จนกว่าเอเจนต์จะไปถึงยังตำแหน่งเป้าหมาย จึงจะได้รับค่ารางวัลเท่ากับ $+10$ เนื่องจากเป้าหมายสำคัญของเอเจนต์คือการให้ได้มาซึ่งรางวัลที่สูงที่สุดในแต่ละรอบการทำงาน ดังนั้น การที่ออกแบบระบบการเรียนรู้แบบนี้ก็เพื่อให้เอเจนต์ค้นหาเส้นทางที่สั้นที่สุดด้วยระยะเวลาที่เร็วที่สุด มิเช่นนั้น หากกำหนดให้ตำแหน่งหรือสถานะอื่น ๆ ที่ไม่ใช่เป้าหมายมีค่าเป็นศูนย์หรือจำนวนเต็มบวก เอเจนต์อาจจะวนเวียนอยู่กับการเคลื่อนที่ในสถานะอื่นในสนาม แทนที่จะมุ่งตรงไปยังเป้าหมายในทันทีหรือใช้เวลาในการค้นหาเป้าหมายนานเกินความจำเป็น เพราะการเคลื่อนที่ไปยังตำแหน่งอื่น ๆ ในสนามก็ส่งผลให้ได้ค่ารางวัลหรือไม่ได้รับบทลงโทษเช่นกัน

3.6.9.3 การสร้างอัลกอริทึมการเรียนรู้แบบคิว (Q-learning) สำหรับเรียนรู้การตอบสนองของเอเจนต์ต่อสิ่งแวดล้อม

ในงานวิจัยนี้จะแบ่งส่วนการสร้างเส้นทางการเคลื่อนที่ของหุ่นยนต์ด้วยการเรียนรู้แบบเสริมกำลังออกเป็นไฟล์ที่รองรับคำสั่งปฏิบัติการ (file executable) นามสกุล .py ทั้งหมด 2 ประเภทด้วยกัน คือ ไฟล์สนามทดสอบและไฟล์การทำอัลกอริทึมการเรียนรู้แบบคิวซึ่งภาพดังต่อไปนี้จะแสดงขั้นตอนการทำงานทั้งหมดของอัลกอริทึม



รูปที่ 3.20 ไดอะแกรมแสดงขั้นตอนการเรียนรู้ของเอเจนต์เพื่อหาเส้นทางไปยังเป้าหมายที่ดีที่สุด เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.9.4 การทดสอบประสิทธิภาพในการเรียนรู้ของเอเจนต์

การเรียนรู้ของระบบที่ได้ทำการสร้างขึ้น สามารถชี้วัดได้จากรอบการทำงานหรือเอพิโซด (Episode) และค่าความแตกต่างระหว่างค่าคุณภาพสูงสุดและต่ำสุด (Maximum / Minimum quality) ในแต่ละรอบการทำงานแสดงดังรูปที่ ค่าความแตกต่างนี้จะลดลงเรื่อย ๆ ซึ่งเป้าหมายของระบบการเรียนรู้แบบเสริมกำลังนั้นจะสิ้นสุดก็ต่อเมื่อ ระบบเข้าสู่จุดอิมิตัว (Convergence) ซึ่งเป็นจุดที่ระบบได้แนวทางการแก้ไขปัญหาคือดีที่สุดแล้ว (Optimal solution) ซึ่งสำหรับงานวิจัยนี้ก็คือเส้นทางที่ดีที่สุดสำหรับหลบหลีกสิ่งกีดขวางในสนามเพื่อไปยังตำแหน่งเป้าหมายนั่นเอง

Algorithm 1 Update Learning Performance Algorithm

```

1:  $t \leftarrow 1.0$ 
2: episode  $\leftarrow 1000$ 
3: deltas = [], diff = 0
4: while iteration < episode do
5:   if iteration % 100 = 0 then
6:      $t = 1e - 2$ 
7:   end if
8:   Grid.state( $s$ )  $\leftarrow$  start point
9:    $a \leftarrow \max(Q(s))$ 
10:  while not Grid.gameOver( $s$ ) do
11:     $a = \text{random}(\epsilon = 0.5 / t)$ 
12:     $r = \text{Grid.move}(a)$ ,  $s_2 = \text{Grid.currentState}()$ 
13:    OldQ( $s, a$ )  $\leftarrow Q(s, a)$ 
14:     $a_2, \max(Q(s_2, a_2)) = \max(Q(s_2))$ 
15:     $Q(s, a) \leftarrow Q(s, a) + \alpha [R + \gamma \max_a(Q(s_2, a_2)) -$ 
16:       $Q(s, a)]$ 
17:    diff = max(|OldQ( $s, a$ ) -  $Q(s, a)$ |)
18:     $s \leftarrow s_2$ 
19:  end while
20:  deltas.append(diff)
21: end while

```

รูปที่ 3.21 ภาพแสดงอัลกอริทึมการเรียนรู้แบบคิวของระบบค้นหาเส้นทางและส่วนตรวจสอบประสิทธิภาพของการเรียนรู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

หุ่นยนต์เคลื่อนที่และระบบควบคุม

4.1 หุ่นยนต์เคลื่อนที่และระบบควบคุมสำหรับการบอกตำแหน่งเป้าหมายแบบ

Asymmetry Regional Boundary



รูปที่ 4.1 แสดงหุ่นยนต์เคลื่อนที่สำหรับทดสอบกับระบบการบอกตำแหน่งแบบเออาร์บีแอล (ARBL)

4.1.1 หุ่นยนต์เคลื่อนที่

หุ่นยนต์เคลื่อนที่ที่ใช้ในงานวิจัยนี้เป็นหุ่นยนต์ขับเคลื่อนด้วยล้อ จำนวน 4 ล้อ ที่ไม่มีจุดหมุนสำหรับเปลี่ยนทิศทางการหันของหุ่นยนต์ แต่จะอาศัยการปรับความเร็วการหมุนของล้อแต่ละข้างเพื่อสร้างการหมุน โดยล้อแต่ละข้างนั้นจะประกอบไปด้วยมอเตอร์กระแสตรงขนาด 12VDC และชุดเฟืองขับ หุ่นยนต์ประกอบไปด้วย 3 ส่วนหลัก ๆ ด้วยกันคือ ส่วนการสื่อสารไร้สาย, ส่วนการควบคุมการเคลื่อนที่ของหุ่นยนต์ และ ส่วนสั่งการโมดูลชิปที่ทำหน้าที่เป็นโหนดเป้าหมาย โดยแต่ละส่วนนั้นมีการทำงานแยกกันอย่างอิสระและจะสื่อสารกันเพื่อแลกเปลี่ยนข้อมูลเป็นระยะ ๆ รายละเอียดของแต่ละส่วนการทำงานภายในหุ่นยนต์จะมีดังต่อไปนี้

4.1.1.1 ส่วนการสื่อสารไร้สาย (Wireless Communication)

เป็นส่วนที่รับผิดชอบการสื่อสารระหว่างหุ่นยนต์และระบบควบคุมภายนอก ส่วนสำคัญของการสื่อสารนั้นคือ มือถือสมาร์ทโฟนแอนดรอยด์ที่เขียนโปรแกรมควบคุมบนแอปพลิเคชัน เรียกใช้การทำงานของโมดูลสื่อสารเช่น 3G/4G, อินเทอร์เน็ตไร้สายหรือวายฟาย (WiFi) และบลูทูธ (Bluetooth) อีกทั้งแอปพลิเคชันนี้จะทำหน้าที่รวมข้อมูลดิบและสร้างเป็นแพ็กเก็ตข้อมูลเพื่อส่งไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

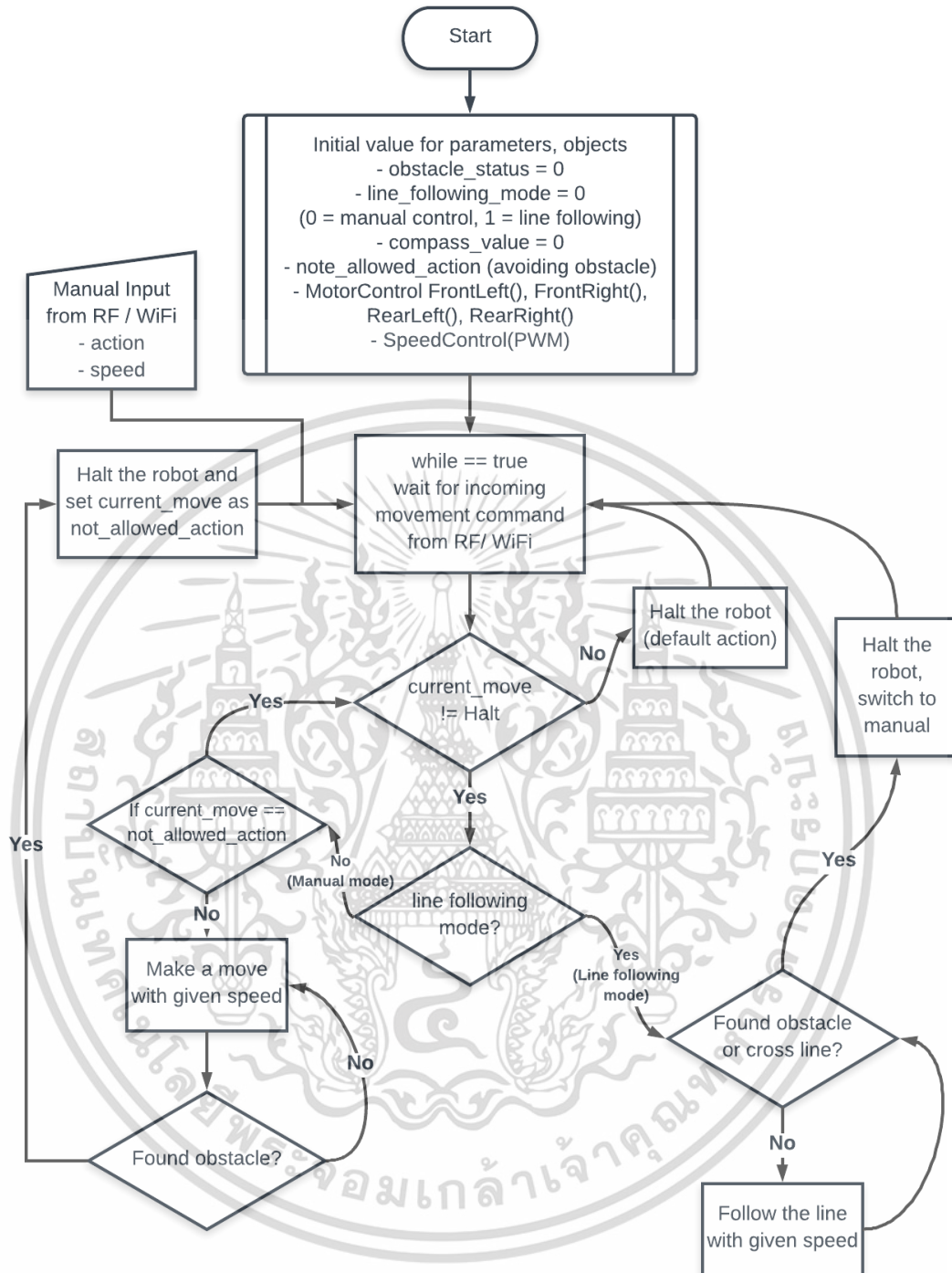
ประมวลผลยังส่วนอื่น ๆ โทรศัพท์มือถือสมาร์ทโฟนนี้จะทำหน้าที่เป็นตัวกลางรับส่งข้อมูลภายในหุ่นยนต์ (ระหว่างอุปกรณ์ควบคุมต่าง ๆ) และเป็นตัวกลางสื่อสารระหว่างหุ่นยนต์กับฐานข้อมูลคลาวด์ไฟร์เออร์เบส (Firebase cloud database) เพื่อมอนิเตอร์และสั่งการทำงานต่าง ๆ อีกด้วย

4.1.1.2 ส่วนควบคุมการเคลื่อนที่ของหุ่นยนต์ (Movement Control)

เป็นส่วนควบคุมการเคลื่อนที่ของหุ่นยนต์ ซึ่งจะประกอบไปด้วยวงจรการควบคุมการหมุนของล้อ ระบบเซนเซอร์ตรวจจับต่าง ๆ แหล่งจ่ายไฟรวมของหุ่นยนต์ รวมไปถึงส่วนสื่อสารข้อมูล ภายในวงจรควบคุมการหมุนของล้อจะประกอบไปด้วย

- ล้อและมอเตอร์พร้อมชุดเฟืองขับเคลื่อนขนาด 12VDC จำนวน 4 ตัว
- วงจรขับมอเตอร์รุ่น VNH5019 4 แผง
- เซนเซอร์ตรวจจับการหมุนของล้อ 2 ตัว
- เซนเซอร์เข็มทิศดิจิทัล 1 ตัว
- บอร์ดควบคุมไมโครคอนโทรลเลอร์ STM32 1 ตัว
- เซนเซอร์ตรวจจับสิ่งกีดขวางแบบอินฟราเรด 1 ตัว
- โมดูลรับส่งสัญญาณวิทยุ (RF) และโมดูลสายพาย WiFi สำหรับรับคำสั่งควบคุมการทำงานจากภายนอกอย่างละ 1 ตัว
- แบตเตอรี่ลิเธียม โพลีเมอร์ (Lithium Polymer) จำนวน 4 ก้อน

หุ่นยนต์จะรอรับคำสั่งในการเคลื่อนที่ผ่านทางโมดูลรับส่งสัญญาณวิทยุ หรือโมดูลสายพายก็ได้แล้วแต่ผู้ใช้งานจะสั่งการ รวมถึงสามารถที่จะส่งข้อมูลความเร็วในการเคลื่อนที่ ทิศทางของหุ่นยนต์และสถานะการตรวจพบสิ่งกีดขวางใด ๆ ขึ้นบนฐานข้อมูลคลาวด์เพื่อนำไปแสดงผลแก่ผู้ใช้งานหรือนำไปประมวลผลในส่วนอื่น ๆ อีกทั้งหุ่นยนต์เคลื่อนที่ที่ยังติดตั้งเซนเซอร์อินฟราเรดสำหรับตรวจจับเส้นบนพื้น (Line-tracking sensor) เพื่อให้หุ่นยนต์เคลื่อนที่ไปตามเส้นบอกตำแหน่งในขั้นตอนการเก็บบันทึกชุดข้อมูลความแรงสัญญาณ ช่วยเพิ่มความสะดวกเมื่อต้องเก็บบันทึกข้อมูลหลาย ๆ รอบ



รูปที่ 4.2 ไดอะแกรมแสดงขั้นตอนการทำงานภายในของหุ่นยนต์เคลื่อนที่

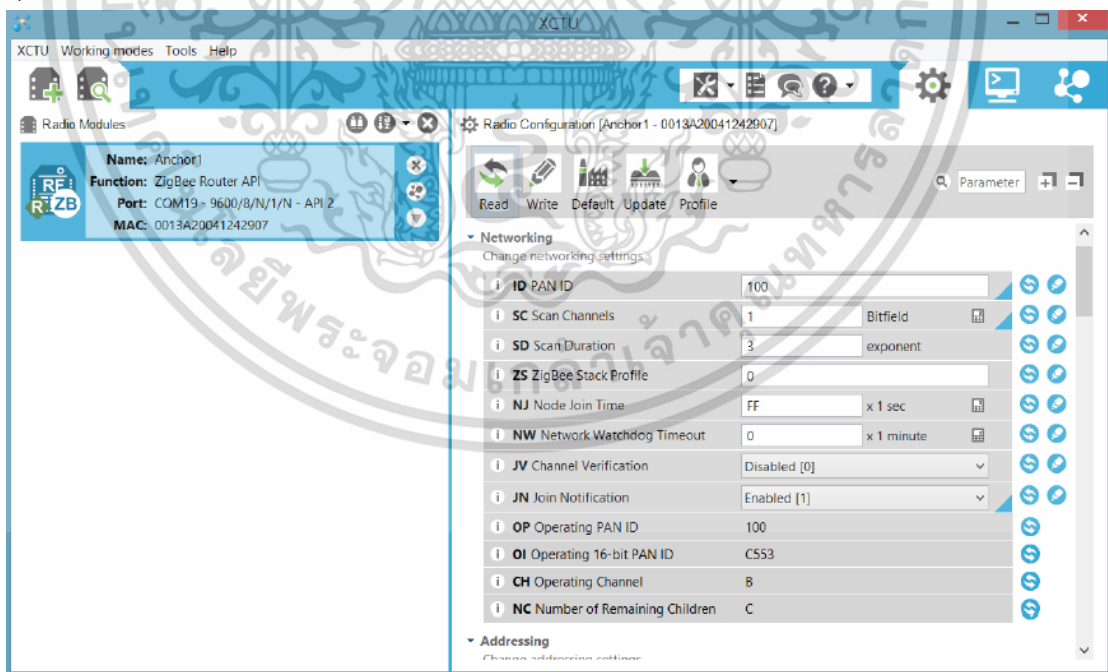
4.1.1.3 ส่วนส่งการโมดูลชิปที่ทำหน้าที่เป็นโหนดเป้าหมาย (Xbee target node)

เป็นส่วนของโหนดเสาสัญญาณเป้าหมายซึ่งจะรวมไปถึงโมดูลชิปซีรีย์ 2 (Xbee Series 2) เสาอากาศแบบอมนิไดเรกชัน (Omni-directional antenna) ขนาด 9 dBi และบอร์ดอาร์ดูโนไมโครคอนโทรลเลอร์รุ่นเมกา (Arduino MEGA) สำหรับส่งการควบคุมการทำงานของโมดูลชิปบอร์ดควบคุมจะทำการส่งคำสั่งเอทีคอมมานด์ (AT command) ไปยังโมดูลชิปเพื่อร้องขอความแรงสัญญาณจากเสาสัญญาณอ้างอิงต่าง ๆ ซึ่งช่วงระยะเวลาการยิงสัญญาณนั้นจะห่างกันเสาละเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

500 มิลลิวัตต์ เพื่อให้ข้อมูลความแรงสัญญาณเป็นปัจจุบันมากที่สุดและสอดคล้องไปกับความเร็วในการเคลื่อนที่ของหุ่นยนต์ เมื่อเสาสัญญาณอ้างอิงแต่ละเสาตอบกลับมานั้น บอร์ดไมโครคอนโทรลเลอร์ก็จะทำหน้าที่กรองข้อมูลความแรงสัญญาณของเสาปลายทางใด ๆ ออกจากแพ็กเก็ตข้อมูล และส่งขึ้นไปเก็บไว้ยังฐานข้อมูลคลาวด์เพื่อประมวลผลต่อไป

4.1.2 เครื่องข่ายโมดูลชิคบีในสนามทดสอบ

เสาสัญญาณอ้างอิงที่ติดตั้งในสนามทดสอบนั้นประกอบไปด้วยโมดูลชิคบีซีรีส์ 2 ชนิดหัวต่อเสาอากาศแบบเอสเอ็มเอ (SMA) จำนวน 5 ตัว กระจายกันอยู่ ณ จุดต่าง ๆ ที่ได้กำหนดไว้ในสนาม แต่ละเสาสัญญาณนั้นจะติดตั้งเสาอากาศแบบออมนิไดเรกชันขนาด 9 dBi และจ่ายไฟเลี้ยงด้วยอแดปเตอร์จ่ายไฟกระแสตรงขนาด 5 โวลต์ 2 มิลลิแอมป์ เสียบเข้ากับรางปลั๊กพ่วงเพื่อให้มีไฟเลี้ยงโมดูลอยู่เสมอ เตรียมรอรับข้อมูลขาเข้า มีเพียงโมดูลชิคบีที่เป็นเสาสัญญาณเป้าหมายติดตั้งกับหุ่นยนต์เคลื่อนที่ที่จะเป็นตัวเริ่มต้นกระบวนการรับส่งข้อมูลกับเสาอ้างอิงอื่น ๆ ด้วยบอร์ดควบคุมโมดูลชิคบีทั้งหมดที่ใช้ในงานวิจัยนี้จะถูกตั้งค่าให้ทำหน้าที่เป็นเราท์เตอร์ (Router) และตั้งโหมดการทำงานแบบเอพีไอ (Application Programming Interface: API) ด้วยโปรแกรมเอ็กซ์ซีทียู (X-CTU) เพื่อที่โมดูลแต่ละตัวจะสื่อสารกันด้วยรูปแบบแพ็กเก็ตข้อมูล ไม่ใช่แบบทรานส์พาเรนต์โหมด (transparent mode) ที่เป็นการส่งข้อมูล ไปต่อไอบีที ซึ่งวิธีเอพีไอ นี้จะทำให้ง่ายต่อการตรวจสอบความถูกต้องของข้อมูล เลขหมายของแหล่งที่มาและที่อยู่ปลายทางของแพ็กเก็ตนำส่ง รวมไปถึงคุณลักษณะของแต่ละแพ็กเก็ต และความแรงสัญญาณของการรับส่งแพ็กเก็ตอีกด้วย



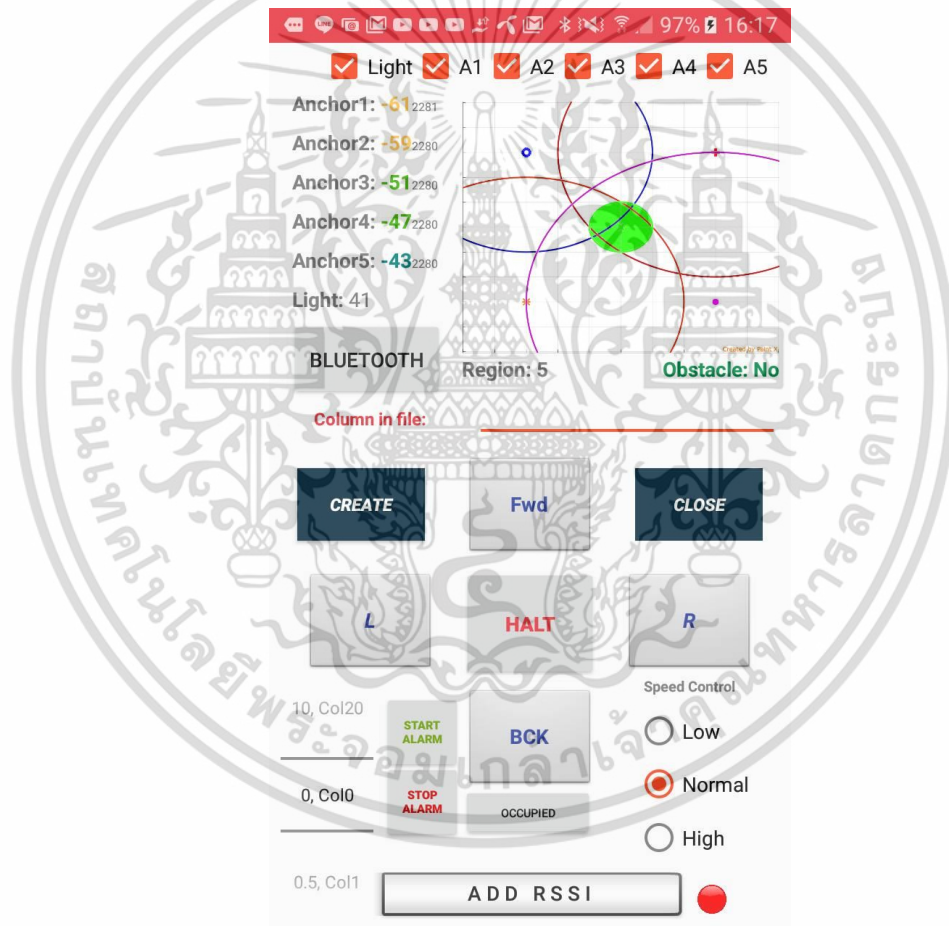
รูปที่ 4.3 แสดงตัวอย่างการตั้งค่าโมดูลชิคบีบนโปรแกรม X-CTU

4.1.3 สถานีฐาน (Base station)

เป็นเครื่องคอมพิวเตอร์ที่เรียกใช้งานโปรแกรมแมทแลป (MATLAB) ขึ้นมาเพื่อสร้าง ฟังก์ชัน และใช้งานโมเดลโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนาม จากข้อมูลความ
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการใช้งานเท่านั้น เมื่อผู้ใดเห็นประโยชน์ของเอกสารนี้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงสัญญาณที่เก็บบันทึกได้ โดยข้อมูลความแรงสัญญาณที่เก็บบันทึกได้ในสนามแต่ละรอบนั้นจะอยู่ในรูปแบบของไฟล์นามสกุล .xls ซึ่งภายในโปรแกรมแมทแลปมีการเขียนสคริปต์ (m file) เพื่อเรียกใช้งานไฟล์ความแรงสัญญาณ ให้สอดคล้องกับการสร้างและฝึกสอนโครงข่ายประสาทเทียม อีกทั้งยังเป็นส่วนรับผิดชอบต่อการวิเคราะห์ข้อมูล แสดงผลข้อมูลในรูปแบบของคอนทัวร์กราฟในออฟไลน์โหมด

อย่างไรก็ตาม ในการทำงานแบบออนไลน์โหมดนั้น สถานีฐานก็เป็นส่วนสำคัญในการเรียกใช้โมเดลโครงข่ายประสาทเทียมที่ถูกฝึกสอนแล้ว เพื่อทำการจำแนกพื้นที่ที่ไม่สมมาตรด้วยข้อมูลความแรงสัญญาณปัจจุบันที่อ่านได้จากหุ่นยนต์เคลื่อนที่ โดยภายในโปรแกรมนั้นจะมีฟังก์ชันที่รองรับการเชื่อมต่อกับฐานข้อมูลคลาวด์ด้วยเรสท์ฟูลเอพีไอ (RESTful API) ทำให้มันสามารถดึงข้อมูลจากฐานข้อมูลคลาวด์ไฟร์เออร์เบสพร้อมทั้งส่งข้อมูลหมายเลขของพื้นที่ที่ไม่สมมาตรกลับขึ้นไปยังฐานข้อมูล เพื่อให้อุปกรณ์อื่น ๆ สามารถนำไปใช้งานต่อได้ในส่วนต่อ ๆ ไป

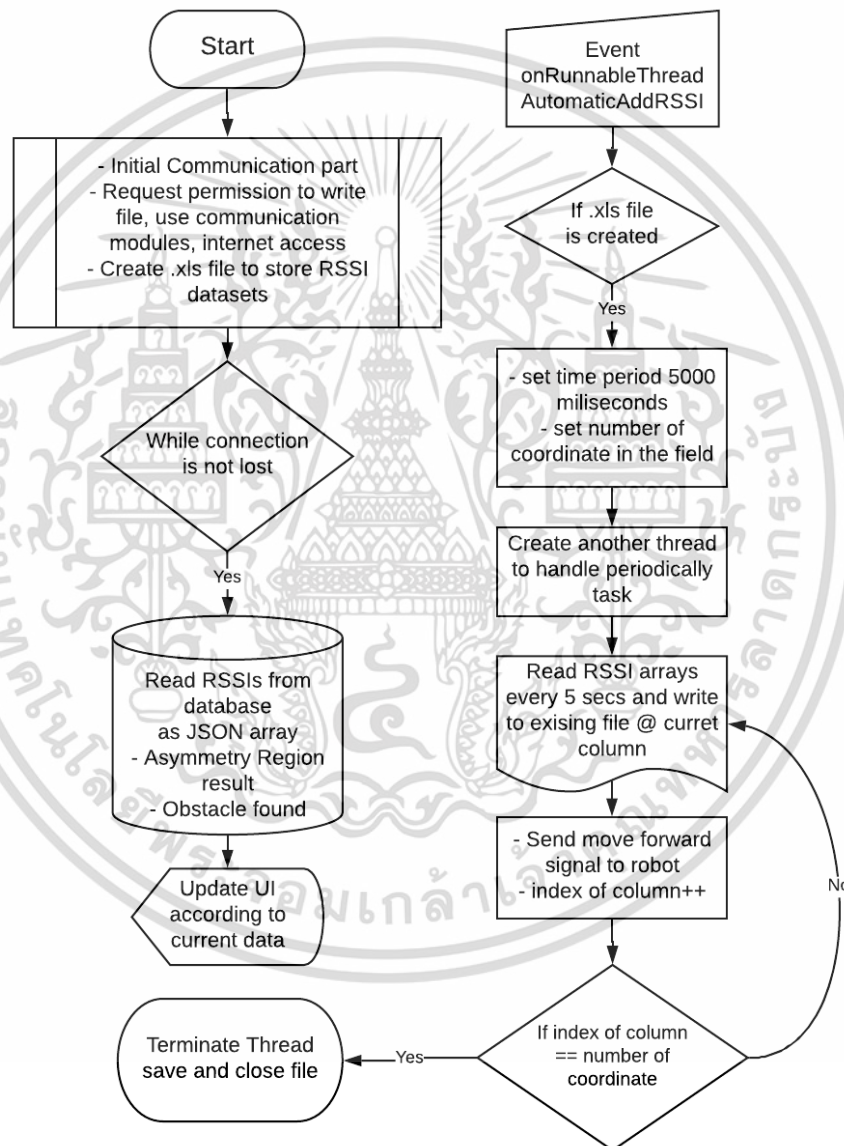


รูปที่ 4.4 แสดงหน้าต่างอินเตอร์เฟซของแอปพลิเคชันบนโทรศัพท์มือถือแอนดรอยด์

4.1.4 โอเปอเรเตอร์ (Operator)

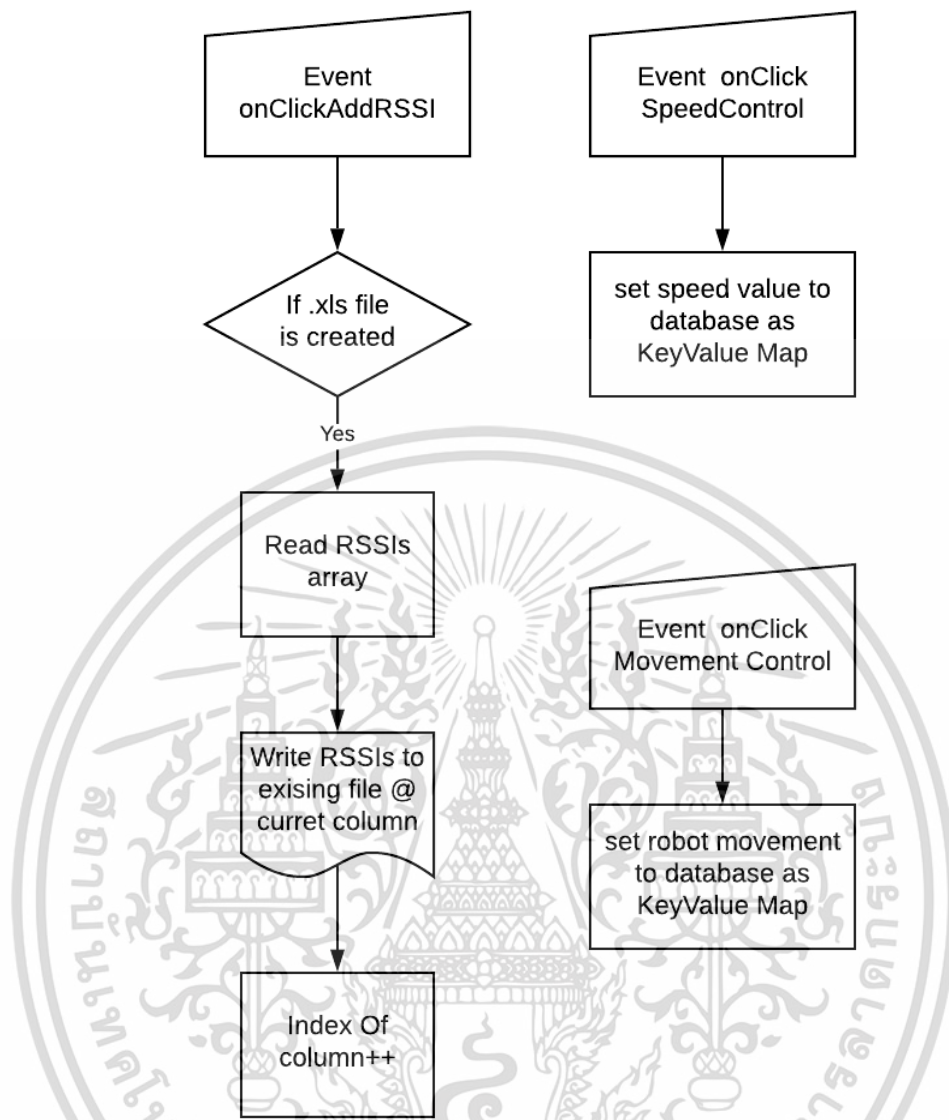
เป็นแอปพลิเคชันบนโทรศัพท์มือถือแอนดรอยด์สมาร์ตโฟนที่จะอยู่ติดตัวผู้ใช้งาน แอปพลิเคชันนี้ถูกเขียนขึ้นมาเพื่อมอนิเตอร์ข้อมูลที่เปลี่ยนแปลงบนฐานข้อมูลคลาวด์ เช่น ผลการจำแนกพื้นที่ที่ไม่สมมาตร ความแรงสัญญาณจากเสาสัญญาณอ้างอิงทั้ง 5 เสา ตำแหน่งที่หุ่นยนต์เคลื่อนที่ตรวจพบวัตถุสิ่งกีดขวาง เป็นต้น พร้อมทั้งยังสามารถสั่งการทำงานหุ่นยนต์ เช่น ควบคุมการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เคลื่อนที่ผ่านฐานข้อมูลคลาวด์ได้ด้วยเช่นกัน ยิ่งไปกว่านั้นตัวแอปพลิเคชันยังเป็นส่วนสำคัญของขั้นตอนการเก็บบันทึกความแรงสัญญาณในสนาม โดยจะทำหน้าที่เป็นตัวควบคุมลำดับขั้นตอนการทำงานของหุ่นยนต์ การหยุดเคลื่อนที่ ณ ตำแหน่งต่าง ๆ ที่กำหนดเพื่อเก็บบันทึกความแรงสัญญาณ ตลอดจนการจัดเก็บข้อมูลลงตารางอย่างเหมาะสมในรูปแบบของไฟล์นามสกุล .xls แอปพลิเคชันนี้มีรูปแบบการสื่อสารกับฐานข้อมูลคลาวด์ผ่านทางอินเทอร์เน็ตไร้สายวายฟาย และอินเทอร์เน็ตสามจีบน โทรศัพท์มือถือ รวมไปถึงแอปพลิเคชันยังสามารถสั่งการหุ่นยนต์ได้โดยตรงผ่านโมดูลรับส่งสัญญาณวิทยุ ซึ่งจะใช้สำหรับกรณีที่หุ่นยนต์ ฐานข้อมูลและผู้ใช้งานขาดการเชื่อมต่อกันจากอินเทอร์เน็ต



รูปที่ 4.5 โฟลว์ชาร์ตการทำงานของแอปพลิเคชันโอเพอเรเตอร์สำหรับสั่งการส่วนต่าง ๆ ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 โฟลว์ชาร์ตการทำงานของแอปพลิเคชันโอเพอเรเตอร์สำหรับสิ่งการส่วนต่าง ๆ ของระบบ
(ต่อ)

4.1.5.ฐานข้อมูลคลาวด์ไฟร์เออร์เบส (Firebase cloud database)

ไฟล์เออร์เบสเป็นฐานข้อมูลคลาวด์แบบโนเอสคิวแอล (NoSQL) ที่เก็บข้อมูลในรูปแบบเจสัน (JSON) และมีการเชื่อมต่อข้อมูลกันอย่างซิงโครนัสแบบเรียลไทม์ (realtime) กับทุก ๆ อุปกรณ์ที่เชื่อมต่อแบบอัตโนมัติในเสี้ยววินาที รองรับการทำงานเมื่ออยู่ในโหมดออฟไลน์ หรือขาดการเชื่อมต่อ โดยข้อมูลจะถูกเก็บไว้ในหน่วยความจำของอุปกรณ์โฮสติ้ง (Hosting) จนกระทั่งกลับมาออนไลน์อีกครั้งระบบก็จะทำการเชื่อมโยงข้อมูลให้อัตโนมัติ สามารถใช้งานได้กับทั้งระบบปฏิบัติการแอนดรอยด์ (Android), ไอโอเอส (iOS) และเว็บแอปพลิเคชัน (Web application platform)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Cloud Storage for Firebase

รูปที่ 4.7 สัญลักษณ์ของฐานข้อมูลคลาวด์ไฟร์เออร์เบส

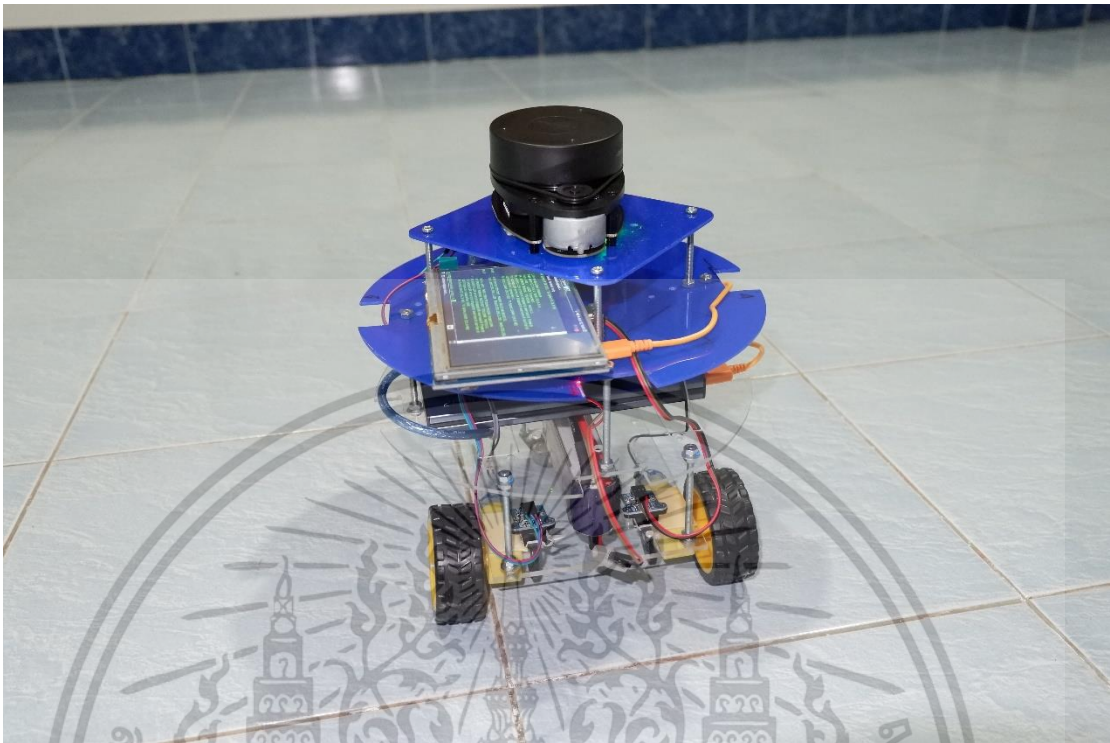
ไฟร์เออร์เบสมีการรองรับการออกแบบเงื่อนไขการเข้าถึงข้อมูลทั้งการอ่านและเขียนซึ่งถือว่ามีความปลอดภัยและมีความน่าเชื่อถืออย่างมาก เนื่องจากมีการตั้งระบบรักษาความปลอดภัยและการตั้งกฎในการเข้าถึงข้อมูล มีการร้องขอออเพนติเคชันคีย์ (Authentication key) หรือโทเคน (Token key) ในการระบุตัวตนว่าการเข้าถึงข้อมูลนั้นมาจากผู้ใช้งานหรืออุปกรณ์ใด จึงจะสามารถอ่านข้อมูล ลบข้อมูล เขียนข้อมูลทับ หรืออัปเดตข้อมูลได้ตามต้องการ

ความแตกต่างของฐานข้อมูลไฟร์เออร์เบสกับฐานข้อมูลเอสคิวแอล (SQL) โดยทั่วไปคือ ฐานข้อมูลไฟร์เออร์เบสนั้นเป็นแบบโนเอสคิวแอล ซึ่งมีโครงสร้างแบบชคิมเลส (Schemeless) กล่าวคือมีความยืดหยุ่นในการออกแบบโครงสร้างการจัดการข้อมูล ไม่จำเป็นต้องมีคำสั่งการเปลี่ยนแปลงสถานะของตารางข้อมูล (ALTER table statement) ก็สามารถที่จะเพิ่ม ลบ หรืออัปเดตข้อมูลได้ ในขณะที่ฐานข้อมูลเอสคิวแอลตามปกตินั้นจะรองรับการทำงานแบบระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database) ซึ่งก็จะมีโครงสร้างเป็นตาราง (Table), หลัก (Column) และแถว (Row) ของข้อมูล อีกทั้งยังต้องมีการสร้างกฎของตาราง (Schema) เสมือนพิมพ์เขียวของผังการทำงาน เข้ามาควบคุมประเภทของข้อมูล คอลัมน์ข้อมูล ความต้องการของข้อมูลในแต่ละฟิลด์ (Field) และคีย์ต้น (Primary Key) ที่จะใช้เป็นเอกลักษณ์ของข้อมูลที่ไม่มีการซ้ำกัน เป็นต้น เพื่อให้โครงสร้างข้อมูลในตารางนั้นมีความสัมพันธ์กันอย่างถูกต้อง

ข้อดีของฐานข้อมูลแบบเอสคิวแอล คือข้อมูลจะมีความถูกต้องตามกฎของตารางที่ได้ ออกแบบไว้ ซึ่งการออกแบบกฎของฐานข้อมูลได้อย่างครบถ้วนสมบูรณ์และถูกต้องนั้นเป็นเรื่องที่ยาก และเปลี่ยนแปลงได้อย่างลำบากเมื่อมีข้อมูลบรรจุลงไปในตารางแล้ว ขณะที่ฐานข้อมูลไฟร์เออร์เบสนั้นจะมีการบันทึกข้อมูลในลักษณะของการจับคู่คีย์และค่าของคีย์ (Key/ Value) ซึ่งทำให้การอ้างอิงถึงข้อมูลบันทึกใด ๆ นั้นมีความยืดหยุ่นมากขึ้น ยิ่งไปกว่านั้นข้อมูลที่บันทึกในแต่ละคีย์นั้นยังไม่ถูกกำหนดรูปแบบตายตัว หากแต่จะเป็นข้อมูลชนิดใดก็ได้อย่างอิสระ เช่น สตริง (String), ตรรกะบูลีน (Boolean), จำนวนเต็ม (Integer), จำนวนทศนิยม (Float) เป็นต้น ดังนั้น ความยืดหยุ่นของโครงสร้างและข้อมูล รวมไปถึงระบบรักษาความปลอดภัยที่เป็นตัวช่วยให้ข้อมูลมีความถูกต้องทำให้ ฐานข้อมูลไฟร์เออร์เบสนั้นยังมีประสิทธิภาพมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 หุ่นยนต์เคลื่อนที่และระบบควบคุมสำหรับการบอกตำแหน่งเป้าหมายแบบ SLAM



รูปที่ 4.8 หุ่นยนต์เคลื่อนที่ที่ใช้กับระบบการบอกตำแหน่งแบบสแลม (SLAM)

หุ่นยนต์เคลื่อนที่ในรูปแบบนี้ถูกกำหนดให้มีระบบการบอกตำแหน่งแบบสแลม (SLAM) ซึ่งจะมีกระบวนการทำงานบนระบบปฏิบัติการรอส (Robot Operating System: ROS) จุดประสงค์ของการสร้างหุ่นยนต์ตัวนี้ขึ้นมาก็เพื่อที่จะประเมินหาประสิทธิภาพในการบอกตำแหน่งรวมไปถึงความเหมาะสมในการใช้งานจริงเมื่อเทียบกับแนวคิดเออาร์บีแอล (ARBL) ที่ได้นำเสนอในงานวิจัย โดยตัวหุ่นยนต์จะประกอบไปด้วยอุปกรณ์ต่าง ๆ ดังต่อไปนี้

1. มอเตอร์พร้อมชุดเฟืองทดรอบ ขนาด 5 โวลต์กระแสตรง จำนวน 2 ตัว
2. ชุดเซนเซอร์ตรวจจับการหมุนของล้อ (Wheel encoder sensor) จำนวน 2 ชุด
3. ล้อลูกกลิ้งสำหรับประกอบตัวถังรถ (Wheel castor) จำนวน 2 ล้อ
4. บอร์ดประมวลผลราสเบอร์รี่พาย 3 โมเดล B (Raspberry Pi 3 Model B) จำนวน 1 บอร์ด
5. จอแสดงผลแบบสัมผัสขนาด 3.5 นิ้ว (3.5 Inch Resistive Touch screen) จำนวน 1 จอ
6. บอร์ดไมโครคอนโทรลเลอร์ (Arduino Uno R3) จำนวน 1 บอร์ด
7. ชีลด์ขับมอเตอร์กระแสตรง (Adafruit Motor shield) จำนวน 1 บอร์ด
8. ชุดเซนเซอร์ไลดาร์ (LiDAR) แบบวัดรอบทิศทาง 360 องศา (SLAMTEC RPLiDAR A1) จำนวน 1 ตัว

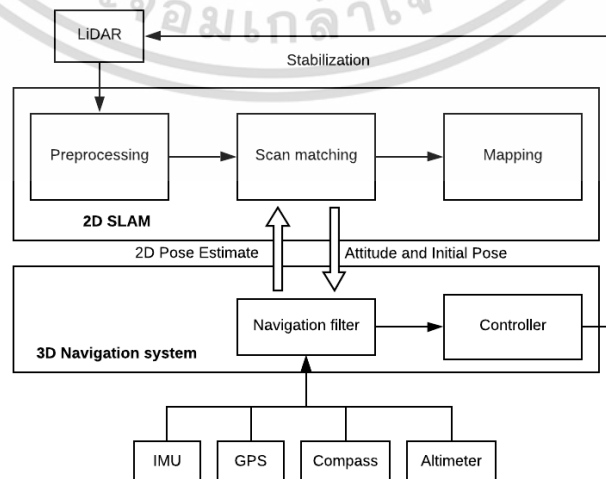
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. แบตเตอรี่พาวเวอร์แบงก์ (Power bank portable charger) ขนาด 5VDC, 20000 mAh จำนวน 1 ก้อน สำหรับจ่ายไฟเลี้ยงให้บอร์ดราสเบอร์รี่พาย จอแสดงผล และเซนเซอร์ไลดาร์ (LiDAR)
10. แบตเตอรี่ลิเทียมพอลิเมอร์ (Lithium Polymer) ขนาด 11.1 VDC/ 3 cell, 2200 mAh จำนวน 1 ก้อน สำหรับจ่ายไฟเลี้ยงให้กับมอเตอร์

ตัวหุ่นยนต์จะมีซิงเกิลบอร์ดคอมพิวเตอร์ราสเบอร์รี่พาย 3 โมเดล B (Raspberry Pi 3 model B) ทำหน้าที่เป็นศูนย์กลางควบคุมการทำงานทั้งหมดของหุ่น ภายในติดตั้งระบบปฏิบัติการอูบันตุเมท (Ubuntu MATE) และรอส คิเนติก เคม (ROS Kinetic Kame) ซึ่งเป็นเฟรมเวิร์กสำหรับจัดการกับแพ็คเกจข้อมูลที่มีการแลกเปลี่ยนสื่อสารระหว่างกัน อีกทั้งยังมีบอร์ดไมโครคอนโทรลเลอร์อาร์ดูโน (Arduino UNO) ทำหน้าที่เป็นอุปกรณ์ต่อพ่วง (Peripheral unit) ที่คอยรับคำสั่งการทำงานจากบอร์ดราสเบอร์รี่พายเพื่อใช้ควบคุมการเคลื่อนที่ของหุ่นยนต์อีกทีหนึ่ง อย่างไรก็ตาม ระบบรอสอนุญาตให้ผู้ใช้สามารถリモテสั่งการควบคุมการทำงานของหุ่นยนต์ได้จากระยะไกล เนื่องจากรอสมีการจัดการแพ็คเกจข้อมูลโดยอิงจากหมายเลขไอพีแอดเดรส (IP address) ของอุปกรณ์ต้นทาง/ปลายทาง ทำให้สามารถกำหนดรูปแบบการสื่อสารในแบบมาสเตอร์/ไคลเอนท์ (Master / Client) ได้

4.2.1 แนวคิดพื้นฐานของเฮกเตอร์สแลม (Hector SLAM)

ในงานวิจัยนี้ได้ทำการจำลองผลการทดสอบการบอกตำแหน่งและสร้างแผนที่ด้วยเทคนิคสแลมขึ้นมาโดยอ้างอิงทฤษฎีและกระบวนการจากงานวิจัย [42] เพื่อนำมาเปรียบเทียบกับการบอกตำแหน่งด้วยวิธีเออาร์บีแอล (ARBL) ที่ได้นำเสนอในวิทยานิพนธ์นี้ ซึ่งในงานวิจัยดังกล่าวได้มีการนำอุปกรณ์เซนเซอร์บอกระยะห่างของสิ่งกีดขวางแบบสแกนรอบตัวเข้ามาประยุกต์ใช้กับเทคนิคสแกนแมตชิ่ง (Scan matching) เพื่อสร้างแผนที่ที่ผ่านการสำรวจตรวจสอบแล้วในลักษณะของการเข้าจับจองพื้นที่กริด (Occupancy grid) และควบคุมการทำงานทั้งหมดด้วยโครงข่ายเฟรมเวิร์กสำหรับจัดการหุ่นยนต์รอส โดยแผนที่ที่ผ่านการประมวลผลแล้วจะแสดงในรูปแบบของข้อมูลกลุ่มจุดข้อมูลหรือ พอยท์คลาวด์ (Point cloud) ในโปรแกรมจำลองภาพแบบสามมิติอาร์วีเอส (Rviz)



รูปที่ 4.9 โดยะแกรมการทำงานของระบบการบอกตำแหน่งแบบสแลม (SLAM) ในงานวิจัย [42]

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่อได้รู้เนื้อหาแล้วโปรดอย่าเผยแพร่ซ้ำโดยไม่ขออนุญาต หรือทำซ้ำโดยไม่ได้รับอนุญาต หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบดังกล่าวมีจุดประสงค์หลักเพื่อเปิดรับให้หุ่นยนต์มีความสามารถที่จะรับรู้สิ่งแวดล้อมต่าง ๆ ได้อย่างถูกต้องแม่นยำและสามารถระบุตำแหน่งของตนเองในสิ่งแวดล้อมดังกล่าวได้ สามารถใช้กับสแลมในสิ่งแวดล้อมที่มีขนาดเล็กและกระบวนการบอกตำแหน่งและสร้างแผนที่นั้นไม่จำเป็นต้องมีการปิดลูบ รวมถึงสามารถปรับใช้กับระบบไลดาร์ (LiDAR system) สมัยใหม่ที่มีอัตราการอัปเดตข้อมูลความถี่สูง วิธีที่นำเสนอจะประกอบไปด้วยกระบวนการทำงานหลัก ๆ 2 กระบวนการด้วยกันคือ

4.2.1.1 ระบบการแมปปีงแผนที่แบบสแลมสองมิติด้วยข้อมูลจากเลเซอร์สแกนเนอร์ (LiDAR) ในแผนที่เชิงระนาบ

เพื่อที่จะสามารถจำลองสภาพแวดล้อมโดยรอบ ในการแมปปีงสถานที่ที่จะใช้เทคนิคที่เรียกว่า การจับจองพื้นที่กริด (Occupancy grid map) เข้ามาบอกตำแหน่งของหุ่นยนต์เคลื่อนที่ด้วยระบบเลเซอร์สแกนเนอร์ ข้อมูลที่ได้จากการสแกนจะถูกแปลงมาเป็นกลุ่มข้อมูลแบบจุด หรือพอยท์คลาวด์ของปลายรัศมีการสแกน (Scan endpoints) เพื่อที่จะทำการคำนวณหามุมมองและความสัมพันธ์ของเฟรมเชื่อมต่อของส่วนประกอบต่าง ๆ ได้อย่างถูกต้อง ในแต่ละสภาพแวดล้อมข้อมูลพอยท์คลาวด์นั้นสามารถที่จะนำมาใช้ประมวลผลล่วงหน้าเพื่อกรองข้อมูลที่เป็นค่ารบกวนออก ตัวอย่างเช่นการลดอัตราการแซมปลิงข้อมูล (Down-sampling) จำนวนของจุดหรือคัดกรองค่าที่ไม่เข้าพวก (outliers) ออกจากข้อมูลหลัก ซึ่งในงานวิจัยนี้เราได้ทำการประยุกต์ใช้ตัวกรองที่แกน z จึงทำให้เราได้ข้อมูลปลายรัศมีการสแกน (endpoints) ที่มีค่าอยู่ภายในเทรซโอสที่กำหนด ส่งต่อผลลัพธ์ระนาบของการสแกนมายังกระบวนการสแกนแมทซิง ซึ่งเป็นการจัดเรียงตัวระหว่างแนวเลเซอร์สแกนเนอร์ด้วยกันเองหรือแนวเลเซอร์กับแผนที่ที่ตรวจพบ

อุปกรณ์เลเซอร์สแกนเนอร์สมัยใหม่นั้นจะมีระยะการเซนเซอร์ค่อนข้างไกล แต่จะมีอัตราการสแกนที่สูง ซึ่งทำให้ข้อมูลจากการสแกนนั้นให้ผลลัพธ์ที่มีความแม่นยำสูงตามไปด้วย สำหรับระบบการทำงานของหุ่นยนต์นั้น การอ่านค่าของเลเซอร์สแกนเนอร์นั้นมีความแม่นยำและความถูกต้องสูงยิ่งกว่าการอ่านค่าจากเซนเซอร์ตรวจจับการเคลื่อนที่ของหุ่นยนต์ ดังนั้นสำหรับการใช้งานจริงในงานวิจัยนี้จะอาศัยการใช้งานและประเมินประสิทธิภาพเฉพาะกรณีที่ใช้ค่าจากเลเซอร์สแกนเนอร์เท่านั้น ไม่มีข้อมูลการเคลื่อนที่ของหุ่นยนต์เข้ามาเกี่ยวข้อง

4.2.1.2 การประยุกต์ใช้ระบบนำทางแบบสามมิติด้วยข้อมูลของอุปกรณ์วัดความเฉื่อยเชิงมุมไอเอ็มยู (Integrated 3D Navigation System based on Inertial Measurement Unit)

ในกระบวนการนี้จะทำหน้าที่รวมข้อมูลทั้งหมดในระนาบ 2 มิติจากสแลมระบบย่อยเข้ามาทำการคำนวณดังแสดงในรูปที่ 4.9 ในขณะที่สแลมทำงานอยู่ตลอดเวลาแบบเรียลไทม์ซึ่งจะถูกกระตุ้นให้มีการรับส่งข้อมูลก็ต่อเมื่อมีจิ้งหะทริกเข้ามา (real-time triggered) เพื่ออัปเดตข้อมูลจาก

อุปกรณ์เลเซอร์สแกนเนอร์นั้น ตัวระบบประยุกต์ใช้การนำทางแบบสามมิติจะทำการประมวลผลอยู่เบื้องหลังเพื่อสร้างโมเดลการควบคุมการเคลื่อนที่ของยานพาหนะในส่วนต่อ ๆ ไป

การพิจารณาแบบสามมิติสมบูรณ์นั้น สามารถแทนได้ดังสมการต่อไปนี้ $x = (\Omega^T, P^T, V^T)$ โดยที่ $\Omega = (\varphi, \theta, \phi)^T$ คือมุมออยเลอร์แบบโรล (Roll), พิทช์ (Pitch), และ ยอร์ (Yaw) ตามลำดับ ส่วน $p = (p_x, p_y, p_z)^T$ และ $v = (v_x, v_y, v_z)^T$ คือ เวกเตอร์ตำแหน่ง (position) และความเร็ว (velocity) ของแพลตฟอร์มแสดงในรูปของเฟรมการนำทาง (navigation frame) ตามลำดับ

ระบบประยุกต์ใช้การนำทางแบบสามมิตินั้นจะมีกระบวนการทำงานแยกย่อยได้สองขั้นตอนด้วยกันคือ 1. ส่วนการกรองข้อมูลการนำทาง (Navigation filter) ซึ่งมีรากฐานมาจากตัวกรองคาลมานแบบต่อยอด (Extended Kalman Filter) ดังแสดงในสมการที่ 1-3 เป็นการกรองเอาเฉพาะข้อมูลที่ต้องการเพื่อนำมาคำนวณหาความเร็วในการเคลื่อนที่และตำแหน่งของหุ่นยนต์ จากนั้นกระบวนการที่ 2. คือการผนวกข้อมูลเข้ากับระบบย่อยสแลมแบบ 2 มิติที่ได้กล่าวไปแล้วข้างต้น เนื่องจากค่ารบกวนที่เกิดขึ้นกับตัวเซนเซอร์ ทำให้ตัวแปรความเร็วและตำแหน่งนั้นอาจจะมีการเลื่อนหรือคลาดเคลื่อนไป ดังนั้นข้อมูลจากเซนเซอร์จึงต้องมีการปรับปรุงเพิ่มเติม การผนวกข้อมูลจากตัวกรองคาลมานต่อยอดแบบสามมิติและสแลมแบบสองมิตินั้นเข้ามาทำหน้าที่ชดเชยค่าที่คลาดเคลื่อนไป โดยอาจจะใช้ข้อมูลจากเซนเซอร์ประเภทอื่น ๆ (ถ้ามี) อาทิเช่น เซนเซอร์ตรวจจับสนามแม่เหล็ก (Magnetic field sensor) สำหรับวัดมุมมองของการหันหน้าของหุ่นยนต์ หรือเซนเซอร์วัดความดันอากาศ (Barometric pressure sensor) สำหรับวัดระดับความสูง อีกทั้งข้อมูลการวัดการหมุนของล้อ (Wheel odometry) ช่วยให้กระบวนการทำสแลมเมทซึ่งนั้นมีประสิทธิภาพในการประมวลผลมากยิ่งขึ้น

อย่างไรก็ตาม เพื่อลดความซับซ้อนในการพิจารณาเปรียบเทียบประสิทธิภาพการใช้งานจริง ในงานวิจัยฉบับนี้ ข้อมูลที่ใช้ในกระบวนการสแกนเมทซึ่งจะมาจากระบบไลดาร์ซึ่งเป็นเซนเซอร์ตรวจจับสภาพแวดล้อมที่เหมาะสมกับการใช้งานภายในพื้นที่ปิดในอาคาร

4.2.2 เทอมสำคัญในการสร้างหุ่นยนต์เคลื่อนที่ด้วยโรส (ROS)

4.2.2.1 รอสมาสเตอร์ (ROS Master)

ภายในโครงสร้างของเฟรมเวิร์ก อุปกรณ์ที่จะมาทำหน้าที่เป็นมาสเตอร์ จะต้องเป็นตัวเริ่มต้นระบบจัดการแพ็กเก็ตข้อมูลที่เรียกว่าโรสคอร์ (Roscore) ซึ่งเป็นเว็บเซอร์วิสแบบเอ็กซ์เอ็มแอล อาร์พีซีเซิร์ฟเวอร์ (XMLRPC server) ทำหน้าที่เป็นศูนย์กลางในการแบ่งปันทรัพยากรต่าง ๆ ให้กับอุปกรณ์ลูกข่ายอื่น ๆ ในระบบ อีกทั้งยังทำหน้าที่จัดการกับแพ็กเก็ตข้อมูล จัดเก็บตัวแปรพารามิเตอร์ต่าง ๆ รีจิสเตอร์ข้อมูลของโหนดอื่น ๆ เช่น ชื่อโหนด ชื่อหัวข้อสื่อสาร และชื่อส่วนบริการกลางเพื่อให้ลูกข่ายสามารถเข้ามาเรียกใช้ และเปลี่ยนแปลงข้อมูลได้อย่างอิสระ เมื่อโรสคอร์เริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงาน หมายเลข (URI) ที่ถูกกำหนดไว้ในตัวแปร “ROS_MASTER_URI” จะถูกเซตค่าให้เป็น “master_URI” ซึ่งเป็นตัวกำหนดว่าอุปกรณ์ใดที่ทำหน้าที่เป็นแม่ข่ายของระบบสื่อสารทั้งหมด โดยเราสามารถกำหนดการตั้งค่า “ROS_MASTER_URI” ได้ในไฟล์การตั้งค่าระบบที่ชื่อว่า “~/.bashrc”

4.2.2.2 รอสไคลเอนท์ (ROS Client)

จะมีความสามารถเทียบเท่ากับรอสมาสเตอร์ทุกประการ เพียงแต่ว่าตัวไคลเอนท์จะไม่มีสิทธิในการสร้างหรือเปลี่ยนแปลงข้อมูลตั้งต้นบางอย่าง รวมไปถึงเริ่มต้นโครงสร้างการสื่อสารข้อมูลได้อย่างมาสเตอร์ อุปกรณ์จำพวกรอสไคลเอนท์นั้นจำเป็นที่จะต้องมีแม่ข่ายควบคุมระบบเป็นผู้ดูแลเสมอ มิเช่นนั้นลูกข่ายใด ๆ ก็จะไม่สามารถเริ่มกระบวนการทำงานตามที่ได้กำหนดไว้ได้

หุ่นยนต์เคลื่อนที่จะถูกกำหนดให้ทำหน้าที่เป็นรอสมาสเตอร์ เพื่อเป็นศูนย์กลางในการจัดการข้อมูลทั้งหมด ส่วนบอร์ดไมโครคอนโทรลเลอร์อาร์ดูโน้ อูโน่ อาร์สาม (Arduino UNO R3) ที่ทำหน้าที่ควบคุมการทำงานของมอเตอร์ทั้ง 4 ตัว และสถานีฐานหรือเวิร์คสเตชัน (Workstation) นั้นจะถูกกำหนดให้มีบทบาทเป็นรอสไคลเอนท์โดยที่บอร์ดไมโครคอนโทรลเลอร์จะติดต่อสื่อสารกับมาสเตอร์ ผ่านพอร์ตอนุกรม (Serial port) แต่เวิร์คสเตชันนั้นจะติดต่อแบบไร้สายจากระยะไกล คอยส่งคำสั่งควบคุมการเคลื่อนที่ของหุ่นยนต์ให้ไปยังทิศทางต่าง ๆ รวมไปถึงการมอนิเตอร์ค่าที่เซนเซอร์ไลดาร์นั้นตรวจจับสิ่งแวดล้อมโดยรอบหุ่นยนต์ได้และการสร้างแผนที่โดยใช้หลักการเฮคเตอร์สแลมอีกด้วย

4.2.2.3 โปรแกรมจำลองภาพแบบสามมิติอาร์วิส (3D Visualization tools : RViz)

อาร์วิสเป็นโปรแกรมสร้างภาพจำลองของระบบปฏิบัติการรอส โดยมีจุดประสงค์หลักคือ แสดงข้อความรอสเมสเสจ (ROS message) ในรูปแบบสามมิติ ทำให้เราสามารถที่จะมองเห็นความเป็นไปของข้อมูลได้ ยกตัวอย่างเช่น เราสามารถที่จะเห็นระยะห่างระหว่างตัวเซนเซอร์วัดระยะทางด้วยเลเซอร์ กับวัตถุสิ่งกีดขวาง เราสามารถที่จะเห็นข้อมูลแบบพอยท์คลาวด์ (Point Cloud, PCD) ของเซนเซอร์วัดระยะแบบสามมิติ อย่าง RealSense, Kinectk, หรือ Xtion หรือเราสามารถเห็นข้อมูลภาพจากเซนเซอร์ประมวลผลภาพหรือกล้องได้โดยที่เราไม่จำเป็นต้องพัฒนาโปรแกรมขึ้นมาด้วยตัวเอง อีกทั้งยังรองรับการแสดงผลภาพหลาย ๆ รูปแบบ โดยการใช้รูปทรงโพลีกอน และ เครื่องหมายอินเทอแครทีฟ (Interactive markers) ทำให้ผู้ใช้งานสามารถที่จะกำหนดการกระทำหรือเคลื่อนที่ด้วยคำสั่งและข้อมูลที่ได้รับมาจากโหนดการทำงานต่าง ๆ

รอสยังสามารถที่จะบรรยายสถานะหรือรูปทรงของหุ่นยนต์ในรูปแบบข้อมูลประกอบยูอาร์ดีเอฟ (Unified Robot Description Format: URDF) ซึ่งจะแสดงอยู่ในรูปของโมเดลสามมิติที่แต่ละโมเดลนั้นสามารถที่จะเคลื่อนที่และทำงานได้สอดคล้องกับดีกรีออฟฟร็ดอม (Degree Of Freedom) ของตัวมันเอง ซึ่งง่ายต่อการสร้างแบบจำลองหรือควบคุมการเคลื่อนที่ โปรแกรมสามารถแสดงตำแหน่งของหุ่นยนต์เคลื่อนที่และรับข้อมูลระยะห่างจากเซนเซอร์วัดระยะทางแบบ

เลเซอร์ เพื่อใช้ในการนำทางหุ่นยนต์ได้ อีกทั้งโปรแกรมอาร์วียังสามารถที่จะรับข้อมูลภาพจากกล้อง ที่ติดอยู่กับตัวหุ่นยนต์แล้วนำมาแสดงบนโปรแกรมได้อีกด้วย

4.2.2.4 การแปลงพิกัดโคออดิเนต (Coordinate Transformation: TF)

แนวคิดการแปลงพิกัดโคออดิเนต (Transform: TF) ถือเป็นแนวคิดหนึ่งที่มีประโยชน์อย่างมากเมื่อทำการอธิบายความสัมพันธ์ระหว่างส่วนต่าง ๆ ของหุ่นยนต์ เช่นเดียวกับ สิ่งกีดขวางหรือวัตถุต่าง ๆ ที่ตรวจพบโดยรอบ ตำแหน่งนั้นสามารถอธิบายได้จาก ชุดข้อมูลตำแหน่งและทิศทางการทำงานของหุ่นยนต์ ตำแหน่งใด ๆ นั้นสามารถแสดงได้โดยเวกเตอร์ 3 เวกเตอร์ด้วยกัน คือ x , y , และ z ในขณะที่ทิศทางการทำงานนั้นอธิบายได้โดยใช้เวกเตอร์ x , y , z และ w ซึ่งจะเรียกว่าควอเทอเนเนียน (Quaternion) การแปลงพิกัดโคออดิเนตมีส่วนประกอบในโครงสร้างของข้อความสำหรับใช้ระบุตัวตนในแต่ละแพ็กเก็ตดังต่อไปนี้

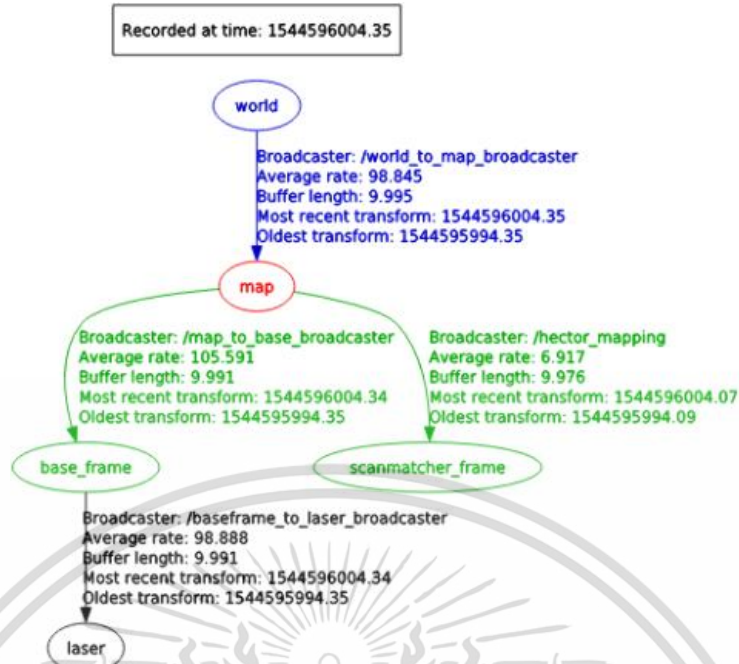
1. ส่วนเฮดเดอร์ (header) นั้นจะใช้บันทึกค่าเวลาและชื่อข้อความ (message name) 'child_frame_id' ซึ่งเป็นฟิลด์ข้อมูลที่ใช้ระบุพิกัดที่เป็นชุดข้อมูลตัวลูก
2. ส่วนข้อมูลดิบ (data) เป็นเฟรมข้อมูลที่บรรจุชุดความสัมพันธ์ของตำแหน่งและมุมมอง ซึ่งมีการจัดเรียงเป็นลำดับของตัวแปรภายในเฟรมข้อมูลหนึ่งเฟรมดังต่อไปนี้
[transform.translation.x / transform.translation.y / transform.translation.z]

4.2.2.5 รอสรัน (ROS run)

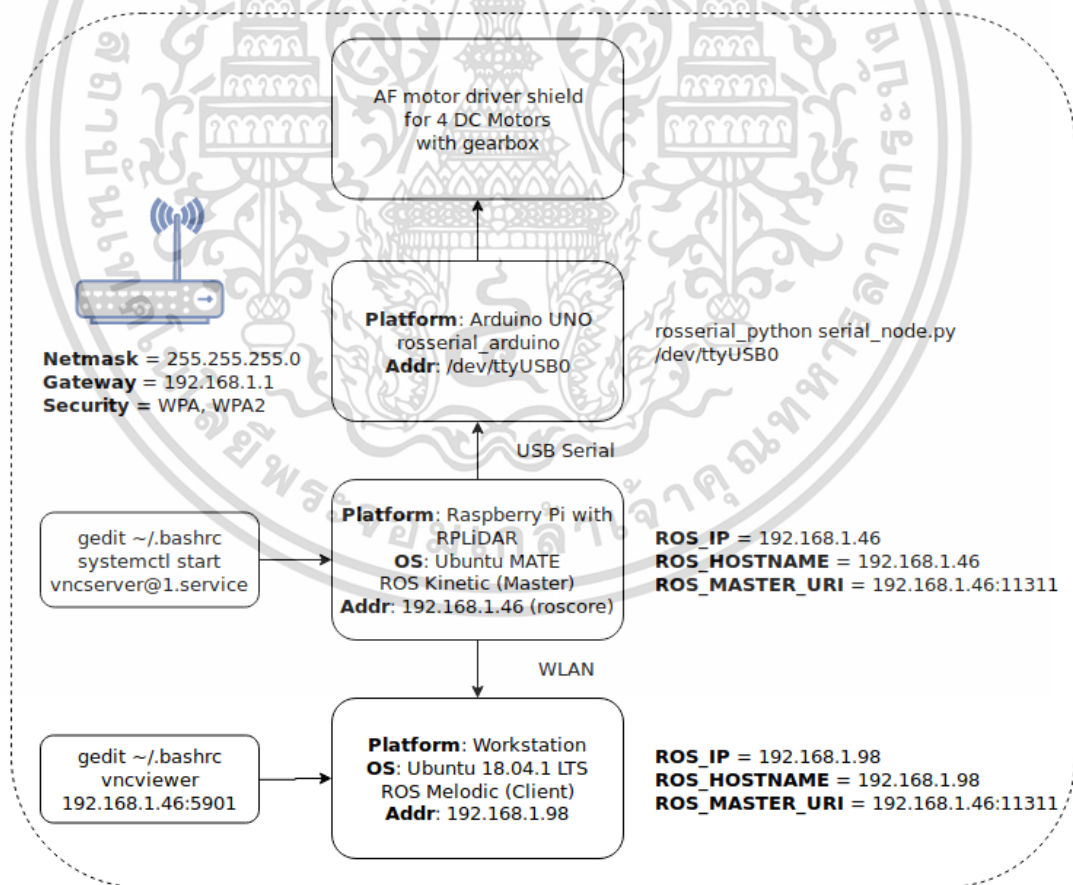
รอสรันนั้นเป็นคำสั่งที่ใช้ในการรันการทำงานของโหนดใดโหนดหนึ่ง ในแพ็กเก็ตจ็ที่ระบุ เหมาะสำหรับการทดสอบหรือดีบักการทำงานในแต่ละส่วนอย่างอิสระจากกัน

4.2.2.6 รอสลันซ์ (ROS launch)

รอสลันซ์เป็นคำสั่งที่ใช้เริ่มต้นการทำงานของโหนดหลาย ๆ โหนดพร้อมกันในคราวเดียว ภายในแพ็กเก็ตจ็ที่ระบุอยู่ในไฟล์นามสกุล .launch นั้นเราสามารถที่จะกำหนดการเริ่มการทำงานของโหนด การเริ่มต้นค่าพารามิเตอร์ เริ่มต้นเซอร์วิสหรือระบบบริการต่าง ๆ ทั้งจากภายในแพ็กเก็ตจ็เดียวกัน หรือต่างแพ็กเก็ตจ็ก็ได้



รูปที่ 4.10 แสดงกราฟความสัมพันธ์ในการบอกตำแหน่งต่าง ๆ ด้วยการแปลงพิกัดโคออดิเนตตั้งแต่แผนที่แสดงสภาพแวดล้อมมายังตัวเฟรมของหุ่นยนต์



รูปที่ 4.11 ไดอะแกรมแสดงส่วนประกอบของหุ่นยนต์เคลื่อนที่และระบบควบคุมภายใต้การทำงานของเฟรมเวิร์กจัดการหุ่นยนต์หรือออส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ผลการทดลองและการวิเคราะห์

5.1 กล่าวนำ

ภายในบทนี้จะกล่าวถึงผลการทดลองและการวิเคราะห์ข้อมูลของการบอกตำแหน่งเป้าหมายด้วยวิธีเออาร์บีแอล (ARBL) ซึ่งจะประกอบไปด้วย ผลการทดลองการเก็บบันทึกชุดความแรงสัญญาณในสนามทดสอบ ผลการทดลองการคำนวณค่าความสามารถในการซ้ค่าของอุปกรณ์และการสร้างพื้นที่ที่ไม่สมมาตร ผลการทดสอบประสิทธิภาพของโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนาม ขณะเรียนรู้ชุดข้อมูลฝึกสอนและขณะใช้งานจริง ผลการทดสอบการค้นหาเส้นทางของหุ่นยนต์เคลื่อนที่ด้วยการเรียนรู้แบบเสริมกำลังและผลการเปรียบเทียบประสิทธิภาพในด้านต่าง ๆ ระหว่างการบอกตำแหน่งด้วยวิธีเออาร์บีแอลและการบอกตำแหน่งและสร้างแผนที่ด้วยเทคนิคผสมระบบปฏิบัติการหุ่นยนต์หรือโรส

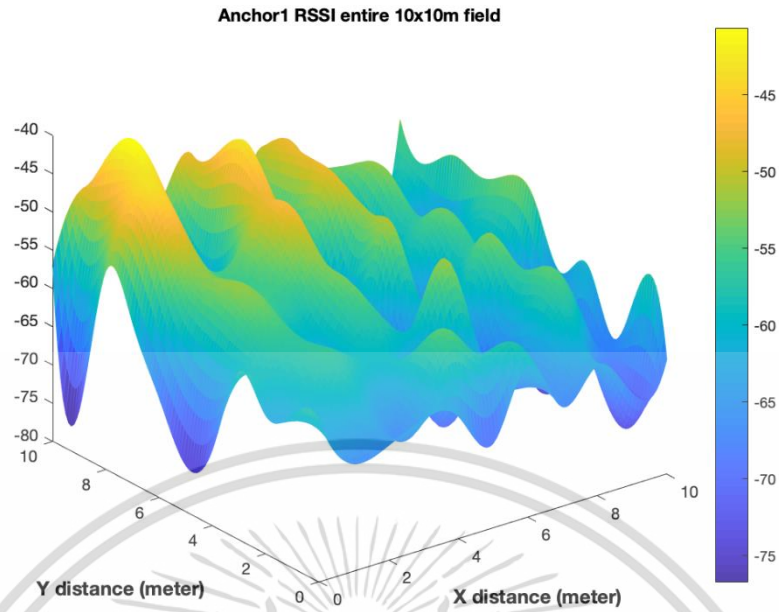
5.2 ผลการทดลองการเก็บบันทึกชุดความแรงสัญญาณในสนามทดสอบ

ตามที่ได้กล่าวไปแล้วข้างต้น พิกัดทั้งหมดที่กำหนดให้หุ่นยนต์ทำการเก็บชุดข้อมูลความแรงสัญญาณ (RSSI) มีด้วยกันทั้งหมด 121 จุดสำหรับสนามทดสอบที่ 1 (10m x 10m) และ 77 จุดสำหรับสนามทดสอบที่ 2 (10m x 6m) โดยจะทำการเก็บข้อมูลทั้งหมด 50 รอบด้วยกัน เท่ากับว่าข้อมูลในแต่ละสนามจะมีจำนวนทั้งหมด 6105 จุดสำหรับสนามที่ 1 และ 3850 จุดสำหรับสนามที่ 2 โดยผลลัพธ์จากการเก็บข้อมูลของสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตรจะแสดงดังรูปภาพตัวอย่างที่ 5.1-5.5

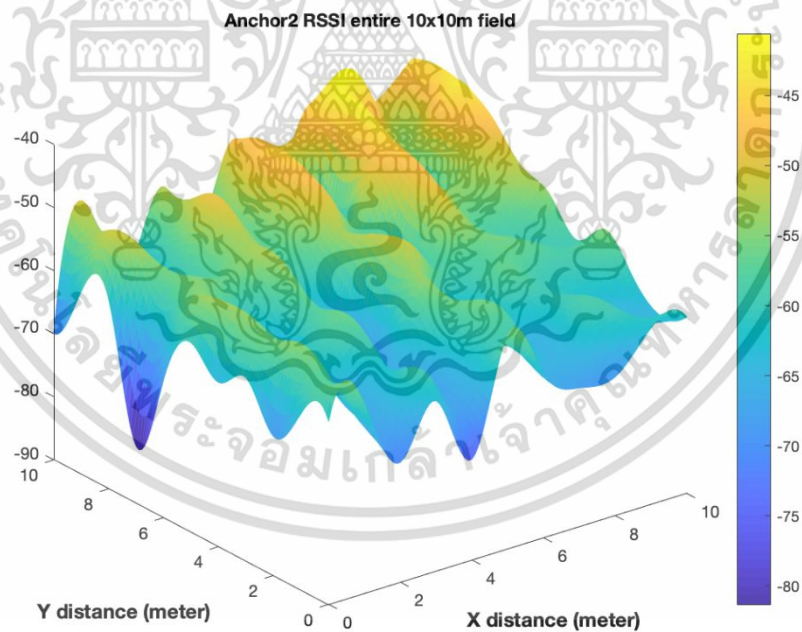
จากภาพที่ 5.1 เป็นตัวอย่างการแสดงข้อมูลความแรงสัญญาณทั่วทั้งสนามของเสาสัญญาณอ้างอิงที่ 1 ซึ่งตั้งอยู่ที่พิกัด (2, 8) ความแรงสัญญาณซึ่งเป็นข้อมูลดิบที่บันทึกได้จะแสดงผลอยู่ในรูปแบบกราฟ คอนทัวร์สามมิติแบบสี (3D filled contour graph) โดยสีแต่ละสีจะแทนความหมายดังต่อไปนี้

1. สีเหลือง: ย่านที่มีความแรงสัญญาณดีมาก มีค่า RSSI ประมาณ -30 ถึง -40 dbm
2. สีส้ม: ย่านที่มีความแรงสัญญาณดี มีค่า RSSI ประมาณ -40 ถึง -45 dbm
3. สีเขียว: ย่านที่มีความแรงสัญญาณปานกลาง มีค่า RSSI ประมาณ -45 ถึง -50 dbm
4. สีฟ้า: ย่านที่มีความแรงสัญญาณพอใช้ มีค่า RSSI ประมาณ -50 ถึง -55 dbm
5. สีน้ำเงิน: ย่านที่มีความแรงสัญญาณต่ำ มีค่า RSSI ประมาณ -55 ถึง -60 dbm
6. สีน้ำเงินเข้ม: ย่านที่มีความแรงสัญญาณต่ำมาก มีค่า RSSI ตั้งแต่ -60 dbm เป็นต้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.1 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 1 พิกัด (2, 8) ในสนามทดสอบขนาด 10 เมตร X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)



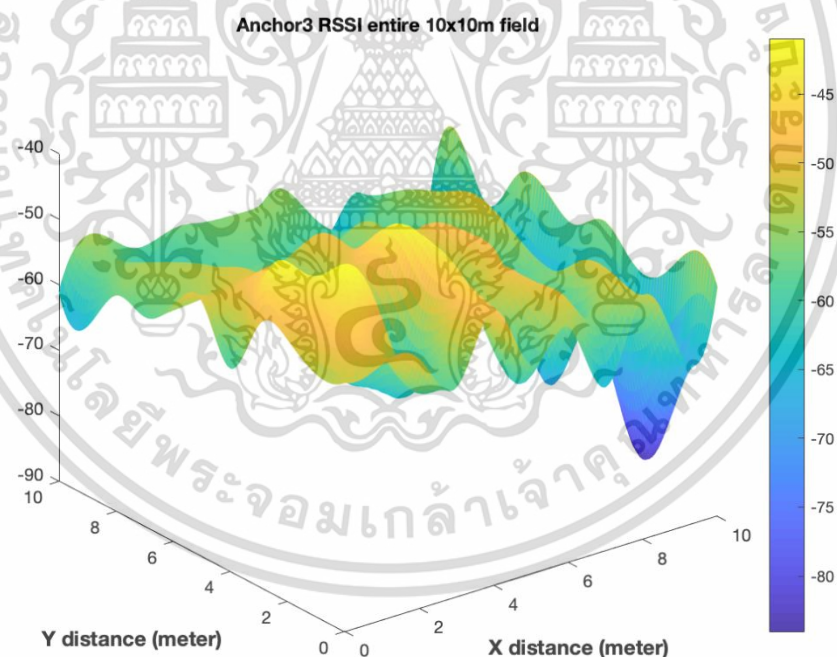
รูปที่ 5.2 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 2 พิกัด (8, 8) ในสนามทดสอบขนาด 10 เมตร X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นได้ว่าบริเวณใกล้เสาสัญญาณ ความแรงสัญญาณจะสูงและเสถียรมาก ในขณะที่ไกลออกจากแหล่งกำเนิดสัญญาณไปนั้น ค่าความแรงสัญญาณจะตกลงและมีความแปรปรวนมากกว่า

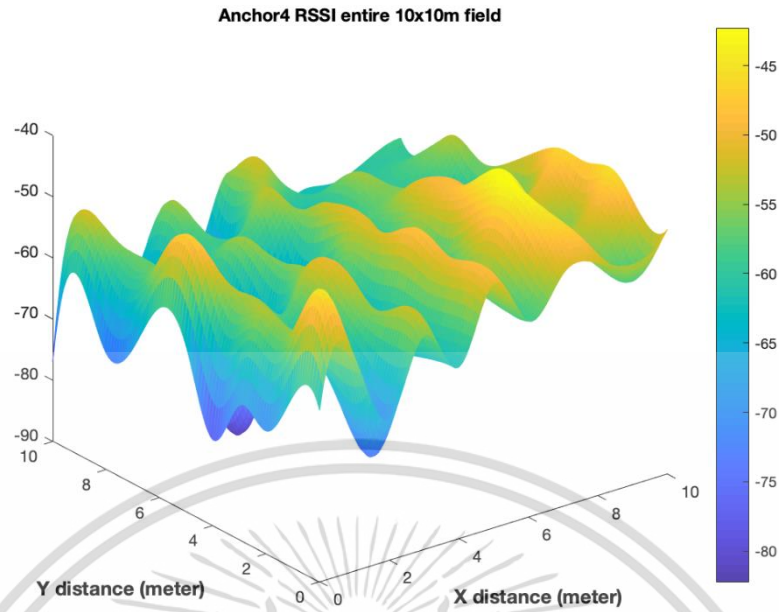
อย่างไรก็ตาม ความแรงสัญญาณที่บันทึกได้นั้นเกิดขึ้นขณะที่หุ่นยนต์หยุดตามพิกัดที่กำหนด แล้วจึงบันทึกผลซึ่งแตกต่างไปจากขณะใช้งานจริงที่ตัวหุ่นยนต์มีการเคลื่อนที่อยู่ตลอดเวลา จึงมีโอกาที่ความแรงสัญญาณจะคลาดเคลื่อนไปจากข้อมูลที่เก็บบันทึกได้ เนื่องจากมีความเร็วในการเคลื่อนที่ของหุ่นยนต์เข้ามาเกี่ยวข้อง ทั้งยังมีเรื่องของระยะเวลาในการอัปเดตข้อมูลของระบบที่ไม่ได้ทำงานกันแบบซิงโครนัสกับหน่วยจัดเก็บข้อมูลบนแอปพลิเคชัน ทำให้บางครั้ง ระบบอาจจะทำการบันทึกความแรงสัญญาณบางค่าลงไปทั้งที่ยังไม่ได้ถูกอัปเดตค่าใหม่ก็ตาม

ในการออกแบบของสนามทดสอบที่ 1 แสดงดังรูปที่ 3.7 จะเห็นได้ว่าตำแหน่งการวางของเสาสัญญาณอ้างอิงในสนามนั้นตั้งอยู่ในแนวการเคลื่อนที่ของหุ่นยนต์ (แนวเส้นประ) ทำให้ระหว่างที่หุ่นยนต์วิ่งเก็บบันทึกข้อมูลความแรงสัญญาณในสนาม ต้องมีมาตรการในการหลีกเลี่ยงหรือแม้แต่บางครั้งผู้ทำการทดลองเองก็จำเป็นต้องสั่งการด้วยมือ ควบคุมให้หุ่นยนต์เคลื่อนที่หลบตำแหน่งของเสาสัญญาณอ้างอิง ด้วยเหตุนี้เองทำให้กระบวนการเก็บบันทึกข้อมูลนั้นไม่เที่ยงตรง และอาจเป็นปัจจัยสำคัญอีกอย่างหนึ่งที่ทำให้เกิดความคลาดเคลื่อนได้

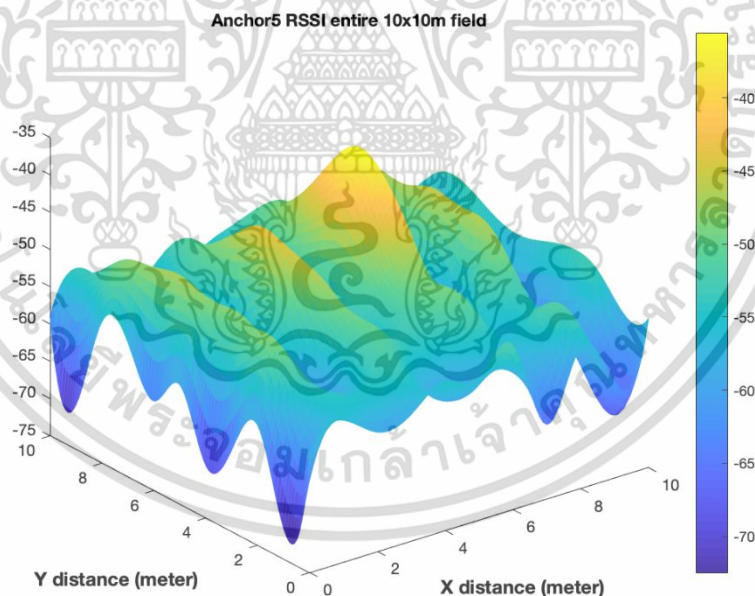


รูปที่ 5.3 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 3 พิกัด (2, 2) ในสนามทดสอบขนาด 10 เมตร X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 4 พิกัด (8, 2) ในสนามทดสอบขนาด 10 เมตร X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)

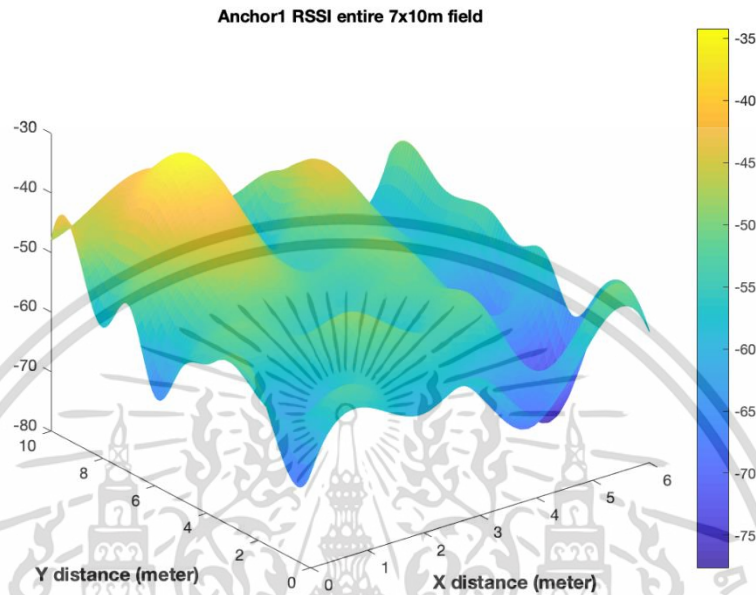


รูปที่ 5.5 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 5 พิกัด (5, 5) ในสนามทดสอบขนาด 10 เมตร X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)

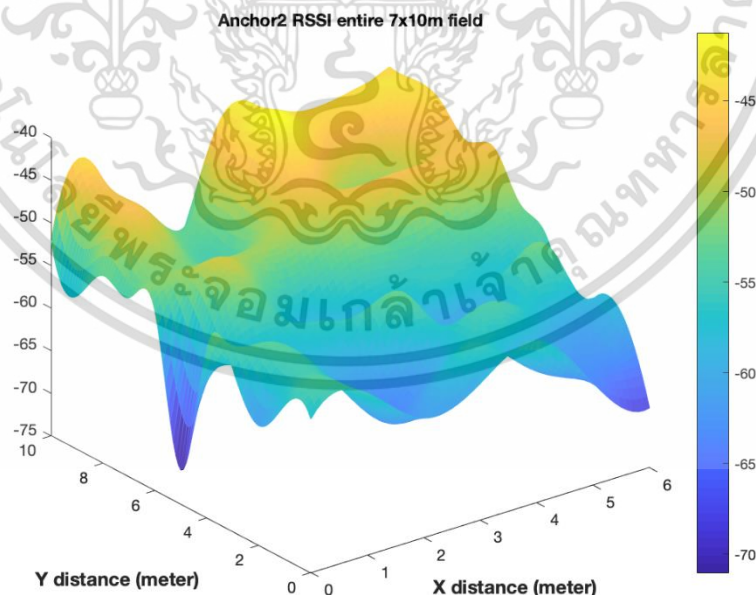
ผลลัพธ์จากการเก็บข้อมูลของสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร จะแสดงดังรูปภาพตัวอย่างที่ 5.6-5.10 ตามลำดับ โดยตำแหน่งการวางเสาสัญญาณอ้างอิงหมายเลข 1-5

จะตั้งอยู่ที่พิกัด (1.5, 8.5), (4.5, 8.5), (1.5, 1.5), (4.5, 1.5), และ (3.5, 5) ตามลำดับ โดยสาเหตุที่ทำให้เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการเรียนการสอนเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวางตำแหน่งเสาให้เลื้อนออกมา 0.5 เมตรก็เพื่อที่จะหลีกเลี่ยงเส้นทางการเคลื่อนที่ของหุ่นยนต์ ขณะทำการเก็บข้อมูลความแรงสัญญาณในสนาม ดังที่ได้เกิดขึ้นกับสนามทดสอบที่ 1 ทำให้หุ่นยนต์เคลื่อนที่ได้ตามแนวเส้นตลอดการเก็บบันทึกข้อมูล ลดโอกาสการเกิดความคลาดเคลื่อนของความแรงสัญญาณที่บันทึกลงไปได้

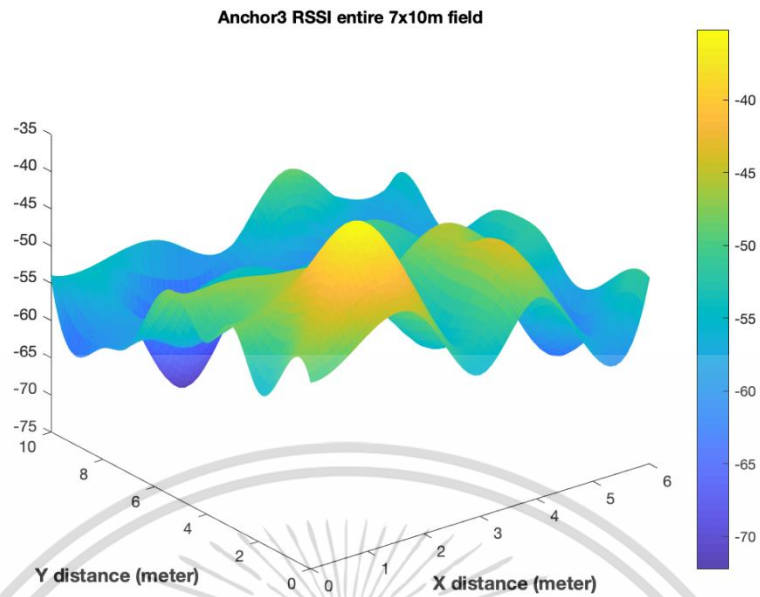


รูปที่ 5.6 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 1 พิกัด (1.5, 8.5) ในสนามทดสอบขนาด 6 X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)

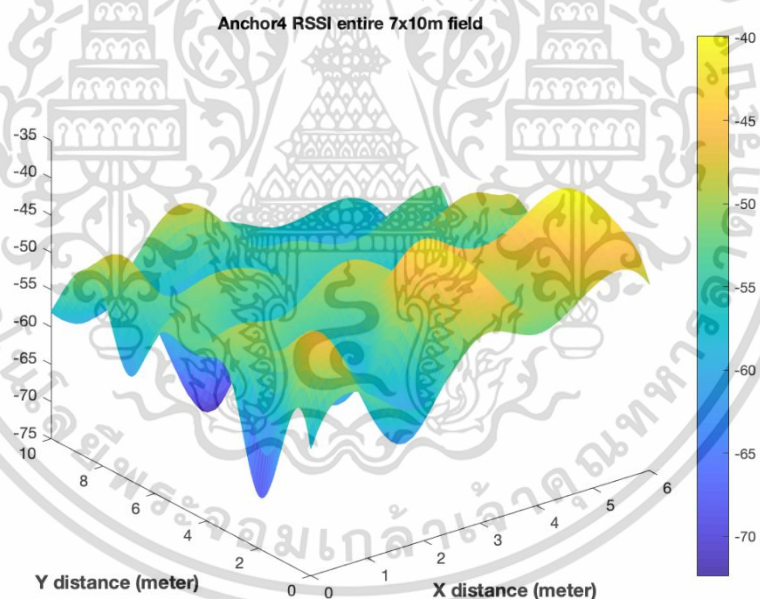


รูปที่ 5.7 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 2 พิกัด (4.5, 8.5) ในสนามทดสอบขนาด 6 X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

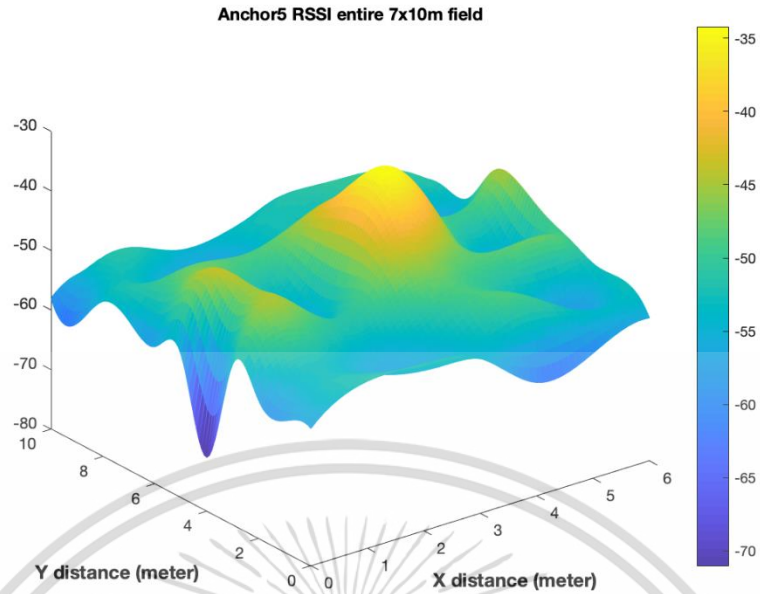


รูปที่ 5.8 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 3 พิกัด (1.5, 1.5) ในสนามทดสอบขนาด 6 X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)



รูปที่ 5.9 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 4 พิกัด (4.5, 1.5) ในสนามทดสอบขนาด 6 X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตร ตามขนาดของกริด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.10 ภาพแสดงคอนทัวร์กราฟแบบสีของ RSSI จากเสาสัญญาณอ้างอิงที่ 5 พิกัด (3.5, 5) ในสนามทดสอบขนาด 6 X 10 เมตร (ความละเอียดในการเก็บข้อมูลทุก ๆ 1 เมตรตามขนาดของกริด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ผลการทดลองการคำนวณค่าความสามารถในการชำระค่าของอุปกรณ์ และการสร้างพื้นที่ที่ไม่สมมาตร

ตารางที่ 5.1 แสดงค่าความสามารถในการชำระค่าแบบเทรชโฮล ของสนามทดสอบที่ 1
(คำนวณจากชุดข้อมูล 10 ชุด)

Anchor Radius	1m	2m	3m	4m	5m	6m	7m	8m	9m	10m
A1	0.93	0.56	0.62	0.54	0.72	0.85	0.98	0.79	0.81	0.77
A2	0.16	0.83	0.3	0.24	0.04	0.3	0.05	0.82	0.39	0.43
A3	0.21	0.42	0.17	0.62	0.06	1.02	0.33	0.22	0.75	0.38
A4	0.39	0.29	0.66	0.29	0.62	0.12	0.35	0.83	0.53	0.59
A5	0.11	0.13	0.12	0.33	0.77	0.45	0.5	-	-	-

ตารางที่ 5.2 แสดงค่าความสามารถในการชำระค่าแบบเทรชโฮล ของสนามทดสอบที่ 2
(คำนวณจากชุดข้อมูล 10 ชุด)

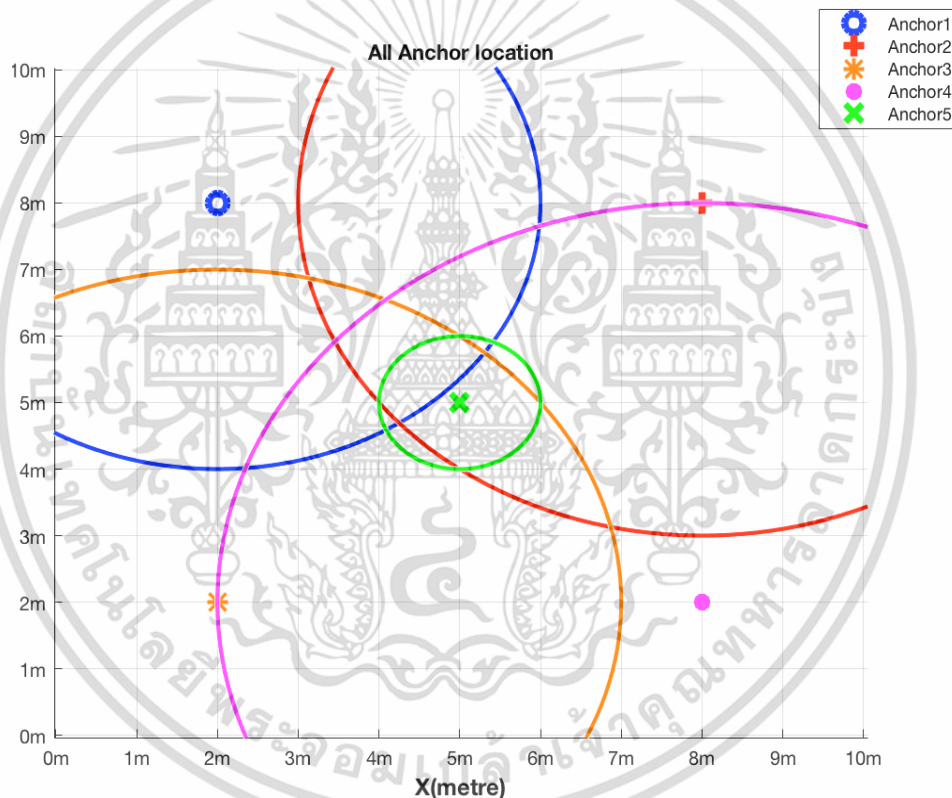
Anchor Radius	1m	2m	3m	4m	5m	6m	7m	8m	9m	10m
A1	-	0.48	0.07	0.42	0.77	0.17	0.17	0.97	0.22	-
A2	-	0.37	0.48	0.64	0.02	0.06	0.99	0.39	0.15	-
A3	-	0.1	0.06	0.21	0.60	0.18	1.31	0.16	0.33	-
A4	-	0.56	0.41	0.41	0.32	0.06	0.25	0.68	0.50	-
A5	0.22	0.16	0.89	0.37	0.21	1.84	-	-	-	-

ความสามารถในการชำระค่าของโมดูลซิกบีในการอ่านค่าความแรงสัญญาณระหว่างเสาสัญญาณเป้าหมายที่ติดตั้งกับหุ่นยนต์เคลื่อนที่และเสาสัญญาณอ้างอิง ณ จุดต่าง ๆ ของสนามนั้นมาจากกระบวนการเก็บข้อมูลเชิงสถิติ เป็นจำนวนหลาย ๆ ครั้ง ตามพิกัดที่ได้กำหนดไว้ล่วงหน้า ทั้งหมด 50 รอบต่อสนาม จากนั้นจึงสุ่มเลือกชุดข้อมูลมาทั้งหมด 10 ชุดเพื่อใช้ในการคำนวณหาความสามารถในการชำระค่า ซึ่งค่านี้เองจะเป็นตัวบ่งบอกถึงความน่าเชื่อถือของอุปกรณ์ ว่าระยะเวลาอ่านค่าระยะใดให้ความเสถียรในการอ่านข้อมูลมากที่สุดภายใต้เงื่อนไขของสภาพแวดล้อมที่อุปกรณ์ใช้งาน และระยะเวลาทำงานที่ให้ค่าเหมาะสมที่สุดจะถูกกำหนดให้เป็นขอบเขตเทรชโฮล (Threshold boundary) ของเสาสัญญาณอ้างอิงนั้น ๆ เพื่อใช้ในการบอกตำแหน่งของเป้าหมายในส่วนต่อ ๆ ไป

ตารางที่ 5.1 และ 5.2 แสดงค่าความสามารถในการชำระค่าของอุปกรณ์แบบเทรชโฮล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้เผยแพร่เอกสารนี้โดยไม่ผ่านการพิจารณาจากผู้เกี่ยวข้อง อาจทำให้เกิดความเสียหายหรือการฟ้องร้องทางกฎหมายได้ ผู้ใช้ควรอ่านเงื่อนไขการใช้งานฉบับเต็มก่อนการใช้งานทุกครั้ง

(Threshold repeatability) ที่ระยะต่าง ๆ ตั้งแต่ 1 เมตร ถึง 10 เมตร เมื่อพิจารณาข้อมูลของแต่ละเสาสัญญาณ จะเห็นว่าแต่ละเสาสัญญาณจะมีระยะการทำงานที่มีค่าความสามารถในการซ้ำค่าแบบเทรซโฮลต่ำที่สุดอยู่ (ค่าที่แสดงด้วยตัวพิมพ์หนา) ตัวอย่างเช่น ในตารางที่ 5.1 เสาสัญญาณอ้างอิงที่ 1 ณ รัศมีการทำงานที่ระยะ 4 เมตรความสามารถในการซ้ำค่ามีค่าเท่ากับ 0.42 ซึ่งเป็นค่าที่ต่ำที่สุดในแถว ดังนั้นระยะที่ 4 เมตรจึงถูกกำหนดให้เป็นขอบเขตเทรซโฮลสำหรับเสาสัญญาณอ้างอิงที่ 1

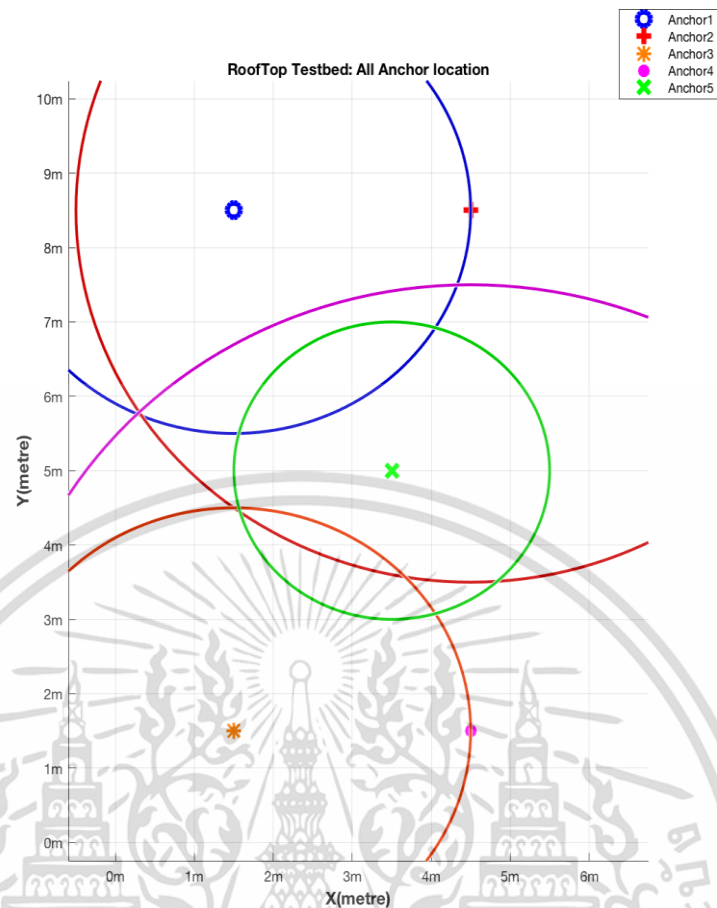
อย่างไรก็ตาม จะเห็นได้ว่าบางรัศมีการทำงานนั้นไม่มีค่าความสามารถในการซ้ำค่าปรากฏสังเกตได้จากช่องตารางของรัศมีการทำงานบางช่องนั้นจะถูกขีดฆ่าออกไป สาเหตุมาจากรัศมีการทำงานนั้นไม่ได้มีตำแหน่งที่ถูกเก็บบันทึกค่าความแรงสัญญาณหรืออาจจะเกินขอบเขตที่รัศมีการทำงานของเสาสัญญาณที่ 10 เมตรครอบคลุมสนามทดสอบ ระยะการทำงานดังกล่าวจึงไม่ถูกนำมาใช้ในการคำนวณ



รูปที่ 5.11 ภาพแสดงขอบเขตเทรซโฮลของเสาสัญญาณอ้างอิงแต่ละเสาในสนามและพื้นที่ปิดที่ไม่สมมาตร (Asymmetry Region) ของสนามทดสอบที่ 1

ดังแสดงในรูปที่ 5.11 และ 5.12 หลังจากที่ได้ขอบเขต (boundary) ของแต่ละเสาสัญญาณอ้างอิงในสนามแล้ว เมื่อนำพล็อตกราฟจะเห็นว่าขอบเขตของเสาสัญญาณต่าง ๆ นั้นทับซ้อนกันเกิดเป็นพื้นที่ปิดที่มีรูปร่างต่าง ๆ อย่างไม่สมมาตร 13 โซนทั้งสนามที่ 1 และ 2 พื้นที่ปิดเหล่านี้จะถูกเรียกว่าพื้นที่ปิดที่ไม่สมมาตร (Asymmetry Region) ที่จะใช้เป็นตัวบ่งบอกตำแหน่งของหุ่นยนต์เคลื่อนที่ และวัตถุต่าง ๆ ในสนามทดสอบ

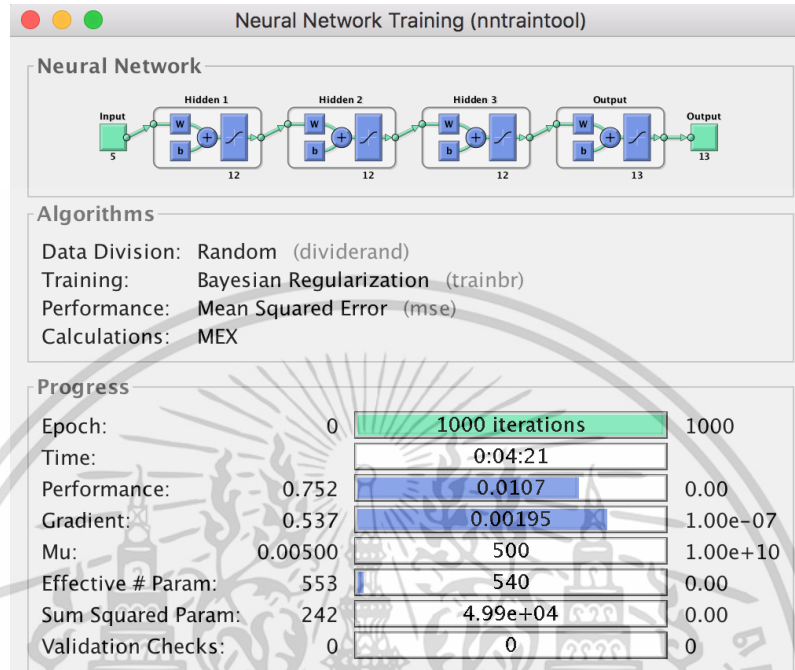
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



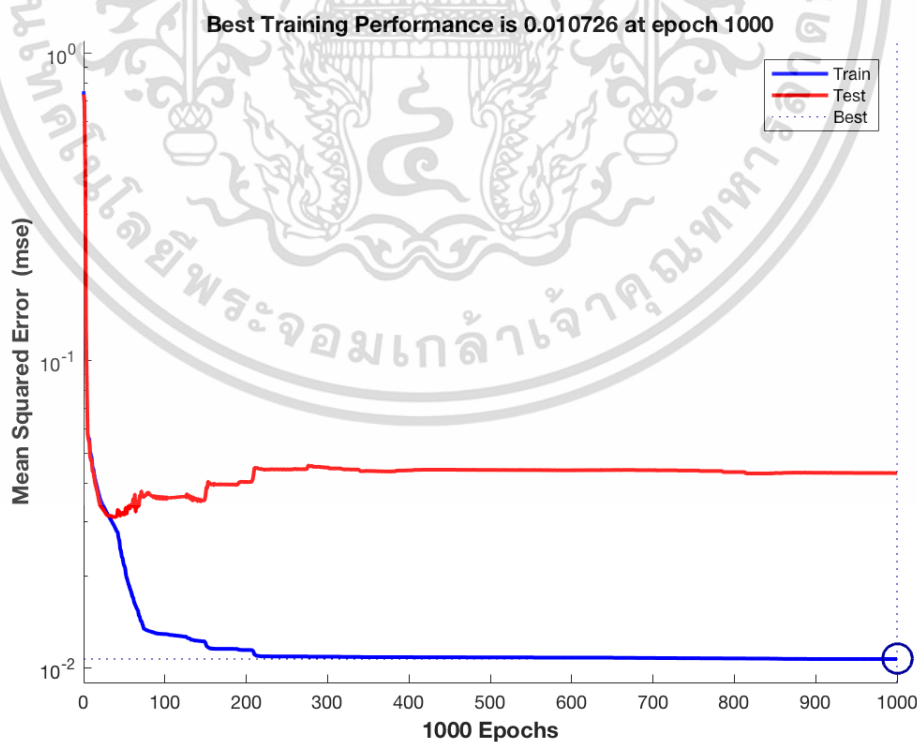
รูปที่ 5.12 ภาพแสดงขอบเขตเทรซโฮลของเสาสัญญาณอ้างอิงแต่ละเสาในสนามและพื้นที่ปิดที่ไม่สมมาตร (Asymmetry Region) ของสนามทดสอบที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 ผลการทดสอบประสิทธิภาพของโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนาม ขณะเรียนรู้ชุดข้อมูลฝึกสอน ทดสอบ และการบอกตำแหน่งด้วยพื้นที่ที่ไม่สมมาตรขณะใช้งานจริง



รูปที่ 5.13 รูปภาพแสดงผลการฝึกสอนโครงข่ายประสาทเทียมสำหรับสนามทดสอบที่ 1 ด้วยโปรแกรมแมทแล็บ

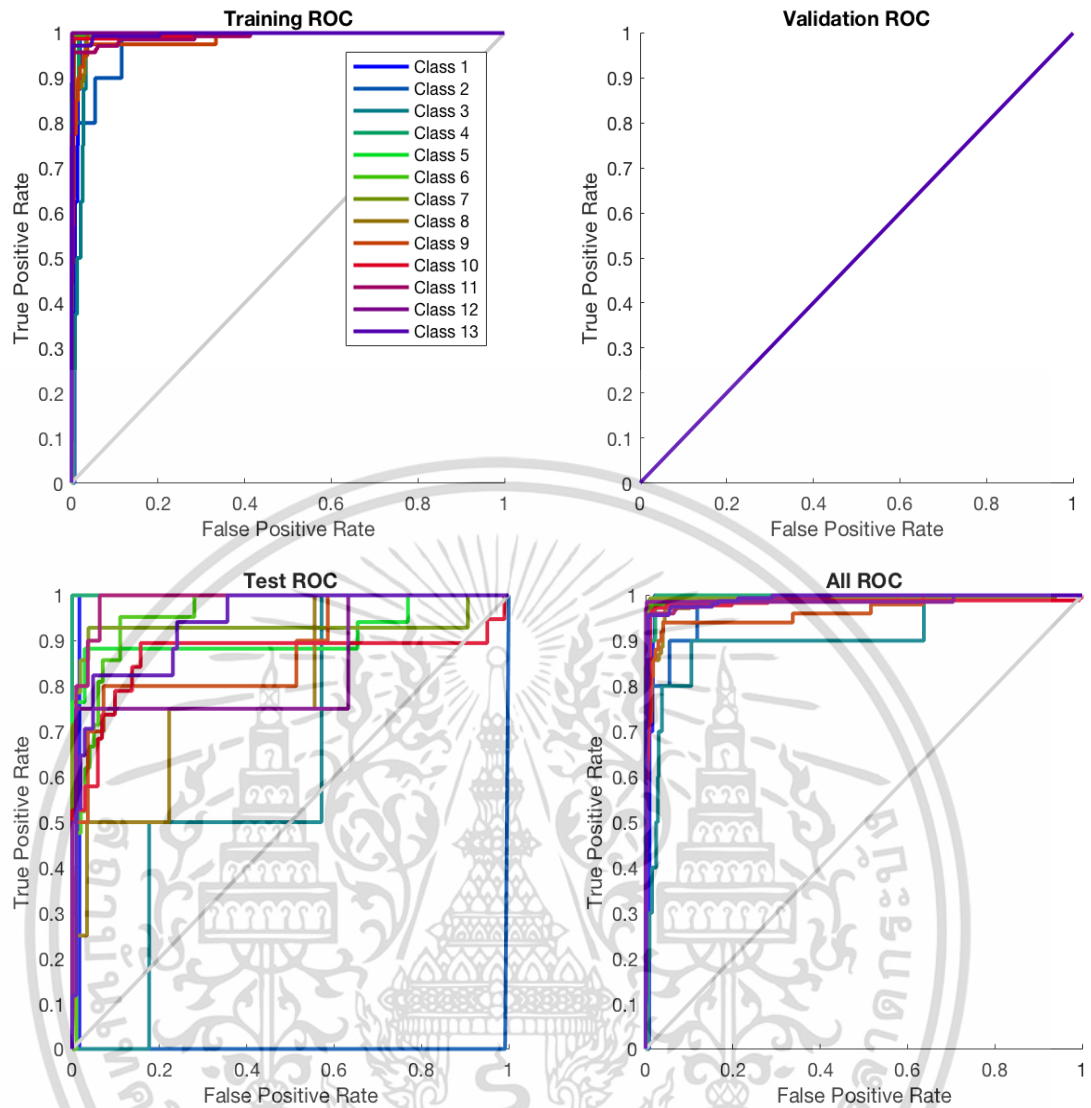


รูปที่ 5.14 กราฟแสดงประสิทธิภาพของโครงข่ายประสาทเทียมของสถานะฝึกสอนและทดสอบโมเดลของสนามทดสอบที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่ได้พื้นที่ที่ไม่สมมาตรจากการทับซ้อนกันของขอบเขตในแต่ละเสาสัญญาณอ้างอิงในสนาม ชุดความแรงสัญญาณจากพิกัดเก็บข้อมูลที่อยู่ภายในพื้นที่ที่ไม่สมมาตรใด ๆ นั้นจะถูกนำมาใช้จำแนกลักษณะเฉพาะของพื้นที่ คล้ายกับการจำแนกปลายนิ้วมือซึ่งต้องผ่านกระบวนการจัดเตรียมข้อมูลและการสร้างโมเดลโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตร ตามที่ได้อธิบายกระบวนการไปแล้วในบทวิธีการวิจัย โดยในส่วนนี้จะกล่าวถึงผลการทดสอบประสิทธิภาพการฝึกสอนโมเดลโครงข่ายประสาทเทียมของทั้งสองสนามทดสอบ ซึ่งรายละเอียดจะมีดังต่อไปนี้

รูปที่ 5.13 และ 5.14 จะเป็นผลการฝึกสอนโมเดลโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนามทดสอบที่ 1 ขนาด 10 เมตร.x 10 เมตร. โดยโครงสร้างของโครงข่ายประสาทเทียมโมเดลสำหรับสนามทดสอบที่ 1 จะมีการตั้งค่าดังนี้คือ กำหนดให้ในอินพุทเลเยอร์ มีจำนวน 5 โหนดอินพุท (แต่ละโหนดแทน RSSI ของแต่ละเสาสัญญาณอ้างอิง) ให้ชั้นฮิดเดนเลเยอร์นั้นประกอบไปด้วย 3 ชั้น ชั้นละ 12 โหนด ซึ่งมีฟังก์ชันกระตุ้นเป็นไฮเปอร์โบลิกแทนเจนท์ซิกมอยด์ (Hyperbolic tangent sigmoid) และในชั้นเอาต์พุทเลเยอร์นั้นจะประกอบไปด้วย 13 โหนดเอาต์พุท (แต่ละโหนดแทนพื้นที่ที่ต้องการจำแนกทั้ง 13 โซน) โดยผลลัพธ์จะถูกจัดรูปแบบให้มีลักษณะเป็นเวกเตอร์วันฮ็อตเอนโค้ดดิ้ง (One-Hot Encoding vector) รวมไปถึงอัลกอริทึมฝึกสอนโครงข่ายประสาทเทียมที่ใช้ในที่นี้คือ บาเยเซียน เรกูลาไรเซชัน (Bayesian Regularization) จากการฝึกสอน จะเห็นได้ว่าผลลัพธ์ไม่ได้เท่าที่ควรจะเป็น ถึงแม้รูปที่ 5.13 จะแสดงให้เห็นว่าตลอดรอบการฝึกสอนทั้งหมด 1000 รอบ ประสิทธิภาพของโมเดลนั้นเข้าใกล้ค่าเป้าหมายที่เป็น 0 ประมาณ 65 ถึง 70 % (ยิ่งเข้าใกล้ศูนย์มากค่าคลาดเคลื่อนยิ่งน้อย) อีกทั้งกราฟเดียนท์สุดท้ายของการฝึกสอนนั้นก็ลดต่ำลงจนเกือบถึงค่าเป้าหมาย อย่างไรก็ตาม รูปที่ 5.14 แสดงให้เราเห็นว่ารูปแบบการฝึกสอนที่สร้างขึ้นส่งผลให้โมเดลยึดติดกับข้อมูลฝึกสอนมากเกินไป ซึ่งทำให้ไม่สามารถปรับใช้กับข้อมูลอื่น ๆ ได้เลย จากกราฟ แนวแกน x จะแสดงค่าคลาดเคลื่อนเฉลี่ยยกกำลังสอง (Mean Squared Error: MSE) ซึ่งเป็นตัวชี้วัดค่าความคลาดเคลื่อนในการจำแนกข้อมูลของอัลกอริทึมบาเยเซียน เรกูลาไรเซชัน และแกน y แสดงรอบการทำงานของอัลกอริทึมในการฝึกสอน ซึ่งตั้งไว้ที่ 1000 รอบ เส้นกราฟน้ำเงินแทนค่า MSE ระหว่างการฝึกสอนโมเดล และเส้นกราฟสีแดงแทนค่าระหว่างการทดสอบ จะเห็นได้ว่าระหว่างการฝึกสอน โมเดลมีความคลาดเคลื่อนในการจำแนกข้อมูลต่ำมาก แต่ตรงกันข้ามกับระหว่างการทดสอบจริง เมื่อประยุกต์ใช้กับข้อมูลชุดอื่น โมเดลให้ความคลาดเคลื่อนในการจำแนกข้อมูลสูงขึ้นจากเดิมอย่างเห็นได้ชัด

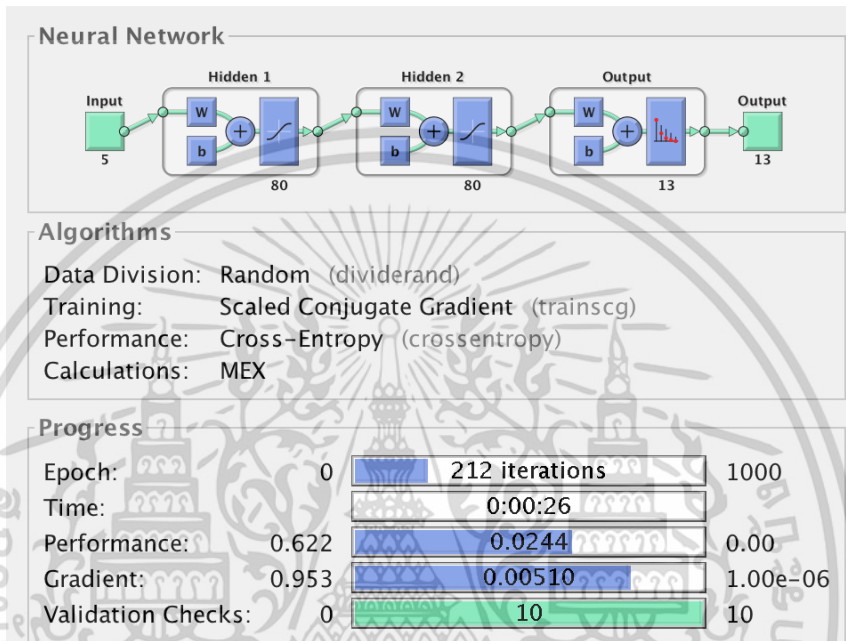


รูปที่ 5.15 แสดงกราฟอาร์โอซี (ROC) ของโมเดลโครงข่ายประสาทเทียมสำหรับสนามทดสอบที่ 1

รูปที่ 5.15 แสดงกราฟฟิซิจัฟเวอร์โอเปอเรตติ้งคาแรคเตอร์ิสติก (Receiver Operating Characteristics: ROC) ซึ่งจะเป็นกราฟที่ใช้บ่งบอกถึงความสัมพันธ์ของผลลัพธ์ในการจำแนกว่ามีความถูกต้องหรือคลาดเคลื่อนมากน้อยเพียงใด โดยที่แกน x จะแทนผลการจำแนกแบบฟอลซีโพซิทีฟเรท (False positive rate) หรืออัตราการจำแนกผิดพลาดหมู่ มีค่าอยู่ในช่วง 0-1 และแกน y แทนผลการจำแนกแบบทรูโพซิทีฟเรท (True positive rate) หรืออัตราการจำแนกถูกหมวดหมู่ มีค่าอยู่ในช่วง 0-1 เช่นกันและเส้นกราฟแต่ละเส้นจะแทนหมวดหมู่ที่ต้องการจำแนก ซึ่งในกรณีนี้ก็คือพื้นที่ที่ไม่สมมาตรทั้งหมด 13 ส่วนในสนามนั่นเอง จากกราฟจะเห็นได้ว่าระหว่างการฝึกสอนนั้น โมเดลสามารถให้อัตราการจำแนกหมวดหมู่ได้ถูกต้อง 100 เปอร์เซ็นต์แทบจะทั้ง 13 หมวดหมู่ มีบางหมวดหมู่เช่น พื้นที่ที่ไม่สมมาตรหมายเลขที่ 2 ที่โมเดลให้อัตราการจำแนกผิดอยู่เล็กน้อย ในทางตรงกันข้าม ระหว่างการทดสอบโมเดลให้ผลลัพธ์ที่ต่างกับการฝึกอย่างสิ้นเชิง จากกราฟจะเห็นได้ว่าอัตราการจำแนกหมวดหมู่ถูกต้องนั้นลดลงอย่างมากโดยเฉพาะกับพื้นที่ที่ไม่สมมาตรหมายเลขที่ 2 ที่โมเดลให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราการทำนายผลผิดพลาดรวม 100 เปอร์เซ็นต์เต็ม นั้นแสดงให้เห็นว่าโมเดลยึดติดกับข้อมูลฝึกสอนมากเกินไปทำให้ไม่สามารถเรียนรู้รูปแบบที่แท้จริงของข้อมูลได้ อย่างไรก็ตาม ในขณะที่ใช้งานจริง (Runtime mode) เมื่อนำโมเดลโครงข่ายประสาทเทียมมาใช้กับชุดความแรงสัญญาณ (RSSI) จริงที่หุ้ยนต์อ่านค่าได้ขณะเคลื่อนที่ผ่านพื้นที่ปิดไม่สมมาตร ณ ตำแหน่งต่าง ๆ ในสนาม ระบบยังคงสามารถที่จะจำแนกพื้นที่ได้อย่างถูกต้องในอัตราส่วนที่ยอมรับได้ แต่จะทำนายผลได้ถูกต้องน้อยลงกับพื้นที่ปิดหมายเลข 2 หรือพื้นที่ปิดอื่น ๆ ที่มีขนาดเล็กซึ่งมีจำนวนข้อมูลตัวอย่างในการฝึกสอนน้อย



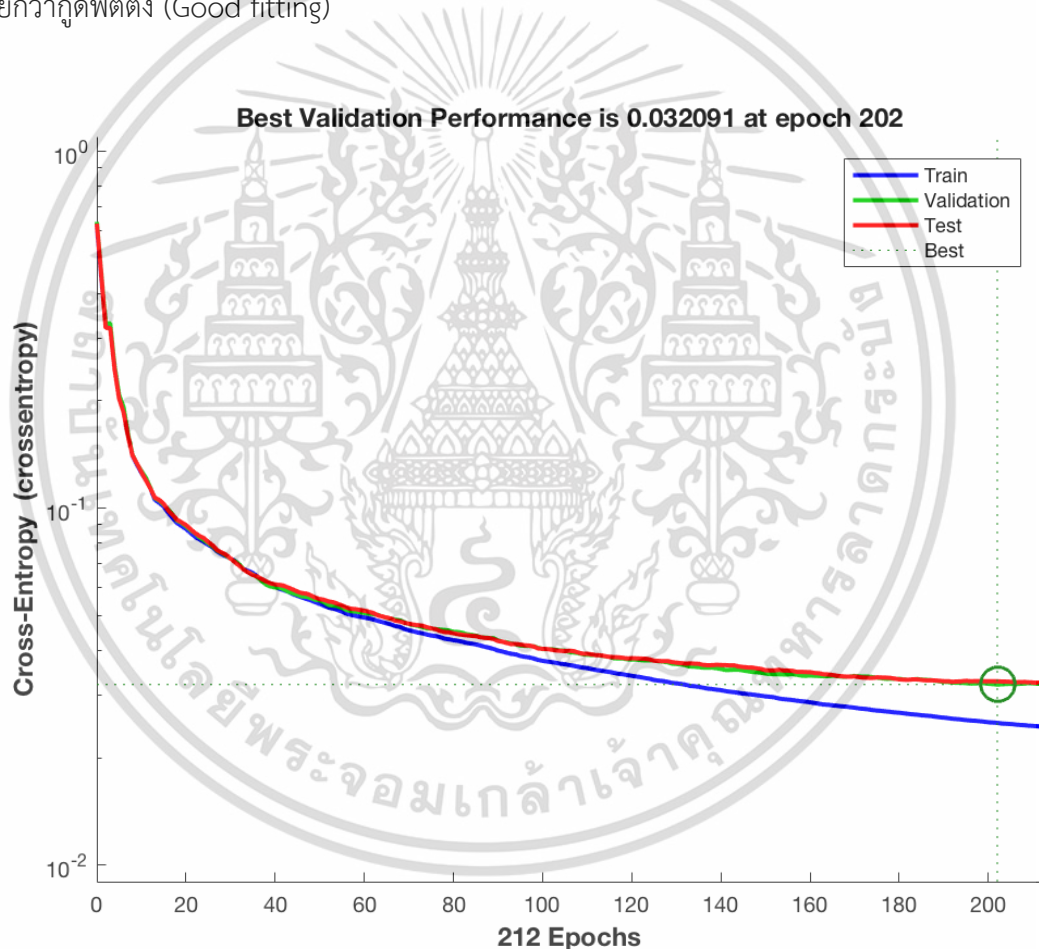
รูปที่ 5.16 รูปภาพแสดงผลการฝึกสอนโครงข่ายประสาทเทียมสำหรับสนามทดสอบที่ 2 ด้วยโปรแกรมแมทแล็บ

เพื่อป้องกันการเกิดพฤติกรรมโอเวอร์ฟิตติงหรือการที่โมเดลยึดติดกับข้อมูลฝึกสอนมากเกินไป ในสนามทดสอบที่ 2 จึงได้ทำการสร้างโมเดลโครงข่ายประสาทเทียมขึ้นมาใหม่พร้อมทั้งประยุกต์ใช้เทคนิคจัดเตรียมข้อมูลและกลยุทธ์การทดสอบต่าง ๆ อาทิเช่น การปรับปรุงปริมาณตัวอย่างข้อมูลที่ใช้ฝึกสอนแต่ละคลาสที่ไม่เท่ากัน การเพิ่มค่าสัญญาณรบกวนแบบไวท์เกาส์เซียน (แสดงดังรูปที่ 5.18) การกำหนดค่าแอลพูเรกูลาไรเซชันเพื่อป้องกันการเรียนรู้ข้อมูลที่ไม่เข้าพวก (outliers) รวมถึงการทำการทดสอบไฮเปอร์พารามิเตอร์ตามที่ได้กล่าวไปแล้วในบทวิธีการวิจัยผลลัพธ์จากการทดสอบที่ดีที่สุดด้วยการปรับจูนพารามิเตอร์ต่าง ๆ นั้นมีค่าดังแสดงในตารางที่ 5.7

ด้วยการตั้งค่าตามที่ได้แสดงดังตารางดังกล่าว ทำให้โมเดลโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนามทดสอบที่ 2 มีประสิทธิภาพระหว่างการทำนายแสดงดังรูปที่ 5.16 และ 5.17 อีกทั้งรูปที่ 5.19 แสดงกราฟรีชีฟเวอร์โอเปอเรตติ้งคาแรคเตอร์สติกของโมเดลโครงข่ายประสาทเทียมอีกด้วย จากรูปที่ 5.16 จะเห็นได้ว่าระบบการฝึกสอนโมเดลนั้นหยุดการฝึกสอน ณ รอบการทำงานที่ 212 เนื่องจากค่าวาลิเดชันหรือค่าการตรวจสอบการเรียนรู้ของโมเดลนั้นครบ 10 ครั้ง

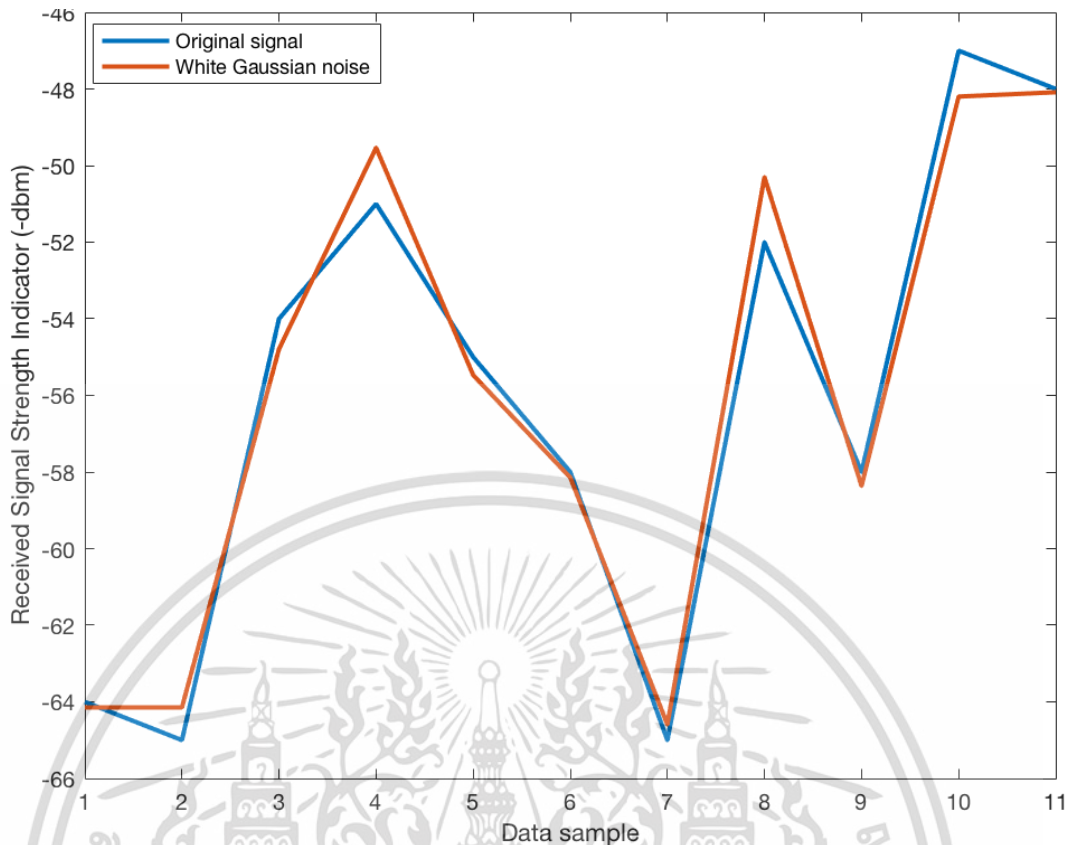
ตามที่ได้กำหนดไว้ล่วงหน้า โดยโมเดลสามารถให้ประสิทธิภาพอยู่ที่ 0.0244 และค่ากราเดียนต์ต่ำสุด เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ 0.00510 ยิ่งไปกว่านั้น จากรูปกราฟ 5.17 จะเห็นได้ว่า เส้นกราฟสีน้ำเงินและสีแดงที่แทนค่าความคลาดเคลื่อนครอสเอนโทรปี (Cross entropy error) ระหว่างการฝึกสอนและการทดสอบโมเดลตามลำดับ นั้นมีความใกล้เคียงกันมาก โดยจะมีแนวโน้มลดต่ำลงเรื่อย ๆ ในทุก ๆ รอบการทำงานของระบบ แนวแกน x คือจำนวนรอบการทำงานหรือเอพิโซด (Episode) ที่มีค่าสูงสุดตั้งไว้ที่ 1000 รอบ และแนวแกน y คือค่าคลาดเคลื่อนครอสเอนโทรปีหรือค่าคลาดเคลื่อนระหว่างโมเดลกับชุดข้อมูล ยิ่งค่าคลาดเคลื่อนน้อยเท่าใดแสดงว่าโมเดลมีความเข้ากันกับชุดข้อมูลมากขึ้นเท่านั้น ณ รอบการทำงานที่ 212 ซึ่งเป็นรอบที่ระบบหยุดฝึกสอนโมเดล จะเห็นได้ว่าค่าคลาดเคลื่อนครอสเอนโทรปี ของกราฟประสิทธิภาพของระบบระหว่างฝึกสอนและทดสอบนั้นมีความใกล้เคียงกันมาก ซึ่งเป็นการสะท้อนให้เห็นว่าโมเดลมีความเข้ากันได้ทั้งกับข้อมูลฝึกสอนและข้อมูลใหม่ที่ใช้ทดสอบ ซึ่งพฤติกรรมเช่นนี้จะเรียกว่ากู่ดฟิตติ้ง (Good fitting)



รูปที่ 5.17 กราฟแสดงประสิทธิภาพของโครงข่ายประสาทเทียมของสภาวะฝึกสอนและทดสอบโมเดลของสนามทดสอบที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.18 แสดงตัวกราฟความแรงสัญญาณที่เก็บบันทึกจากเสาสัญญาณอ้างอิง A1 แบบดั้งเดิม เทียบกับสัญญาณที่ถูกเพิ่มค่าด้วยอัตราส่วนของสัญญาณจริงต่อสัญญาณรบกวน (Signal-to-Noise Ratio) ที่ 35 เเปอร์เซ็นต์

รูปที่ 5.19 แสดงกราฟรีชีฟเวอร์โอเพอเรติงคาแรคเตอร์สติกของโมเดลโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนามทดสอบที่ 2 จากกราฟแสดงให้เห็นว่า โมเดลสามารถที่จะให้อัตราการทำนายผลแบบถูกต้องอยู่ที่ประมาณ 80-100 เเปอร์เซ็นต์ในทุก ๆ คลาสทั้งระหว่างการฝึกสอนและการทดสอบด้วยข้อมูลจริง อีกทั้งในตารางที่ 5.3 ถึง 5.6 และ 5.9 ถึง 5.12 ยังแสดงคอนฟิวชันเมทริกซ์ (Confusion matrix) ซึ่งเป็นเมทริกซ์สำหรับแสดงผลการจำแนกข้อมูลในรูปแบบของ เเปอร์เซ็นต์ความถูกต้อง ความคลาดเคลื่อน และปริมาณของข้อมูลที่ทำนายทั้งถูกและผิดในแต่ละหมวดหมู่ทั้งในรูปของจำนวนเต็มและเปอร์เซ็นต์เทียบจากจำนวนข้อมูลทั้งหมด ของทั้งสนามทดสอบที่ 1 และ 2 ตามลำดับ โดยช่องในแนวทแยงของตารางคอนฟิวชันเมทริกซ์ (ช่องสีเทาตัวอักษรพิมพ์หนา) คือช่องที่โมเดลสามารถให้ผลการจำแนกเอาท์พุทคลาสตรงกับคลาสเป้าหมาย (Target class) ได้อย่างถูกต้อง โดยหมายเลขในแต่ละช่องดังกล่าวคือจำนวนจุดข้อมูลที่ทำนายผลลัพธ์ถูก ส่วนช่องอื่น ๆ ภายในแถวเดียวกันนั้นคือช่องแสดงคลาสที่โมเดลให้ผลการจำแนกที่ผิด ตัวอย่างเช่น จากตารางที่ 5.3 แถวที่ 5 โมเดลให้ผลลัพธ์การจำแนกพื้นที่ที่ไม่สมมาตรผิดเป็นพื้นที่หมายเลข 5 จากข้อมูลความแรงสัญญาณ 4 จุด ทั้งที่ความเป็นจริงแล้ว พื้นที่ที่ถูกต้องของความแรงสัญญาณทั้ง 4 จุด นี้คือพื้นที่หมายเลขที่ 6 เป็นต้น รายละเอียดความสัมพันธ์ของผลลัพธ์ในแต่ละแถวแสดงดังตารางที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.8 ในขณะที่ ตารางที่ 5.4, 5.6, 5.10 และ 5.12 จะแสดงคอนฟิวชันเมตริกซ์ของข้อมูลในรูปแบบเปอร์เซ็นต์ความถูกต้องและความคลาดเคลื่อนโดยรวมของทุกคลาส โดยแต่ละตารางนั้นสามารถแบ่งหัวข้อของผลลัพธ์ออกได้เป็น 4 หัวข้อด้วยกันคือ

1. เปอร์เซ็นต์การทำนายถูกต้อง (Precision หรือ Positive Predictive Value :PPV)

$$PPV = \frac{TP}{TP + FP} = 1 - FDR$$

โดยที่ TP คือผลการทำนายเป็นจริงตามเป้าหมาย หรือ True positive, FP หรือ False Positive คือผลการทำนายผิดจากเป้าหมายในกรณีที่เป้าหมายเป็นเท็จ แต่ผลการทำนายเป็นจริง

2. เปอร์เซ็นต์การทำนายผิดพลาด (False discovery rate: FDR)

$$FDR = \frac{FP}{FP + TP} = 1 - PPV$$

3. อัตราการทำนายถูกหมวดหมู่ (Sensitivity หรือ True positive rate: TPR)

$$TPR = \frac{TP}{TP + FN} = 1 - FNR$$

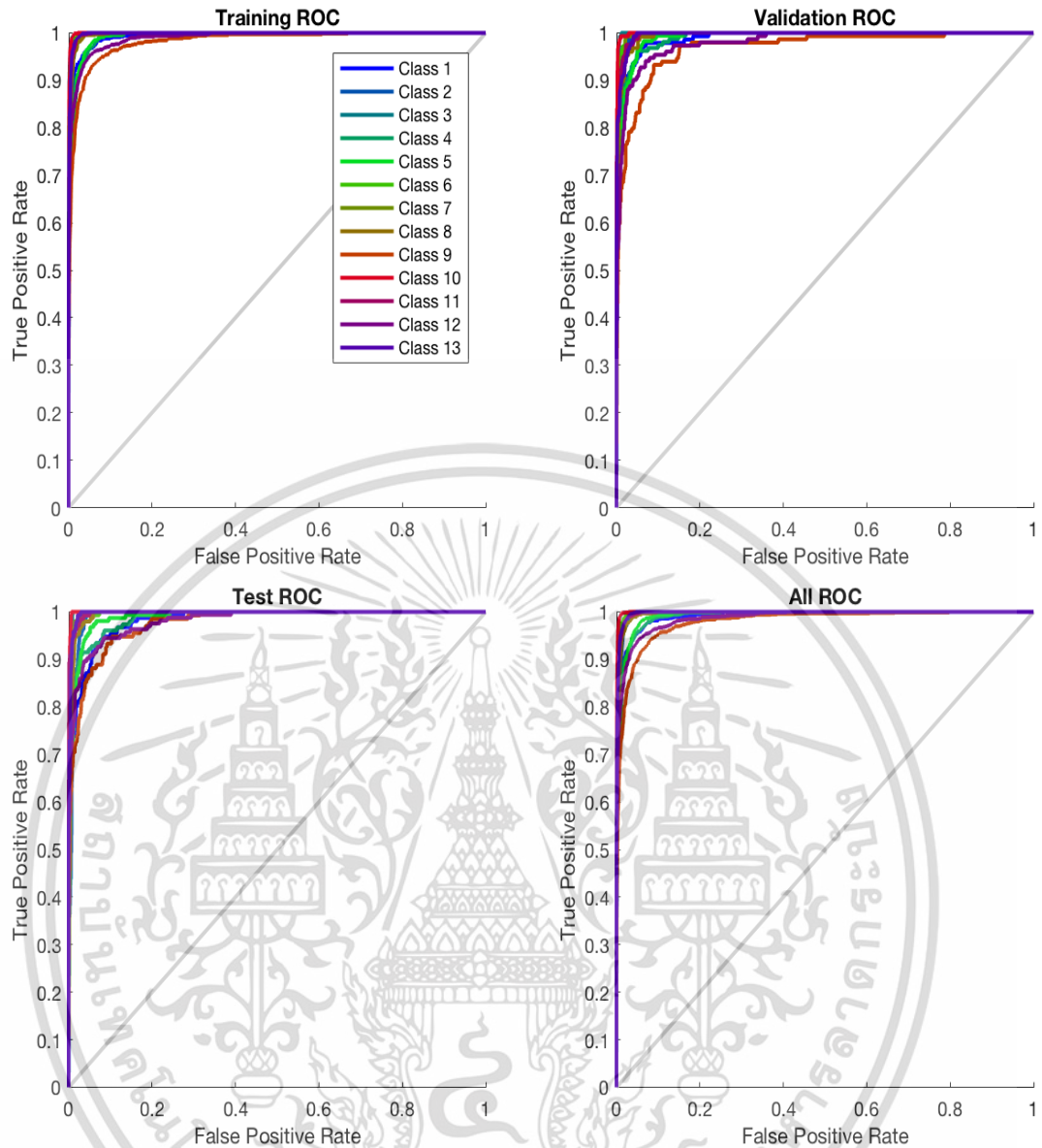
โดยที่ FN หรือ False Negative คือผลการทำนายผิดจากเป้าหมายในกรณีที่เป้าหมายเป็นจริง แต่ผลการทำนายเป็นเท็จ

4. อัตราการทำนายผิดหมวดหมู่ (Miss rate หรือ False negative rate: FNR)

$$FNR = \frac{FN}{FN + TP} = 1 - TPR$$

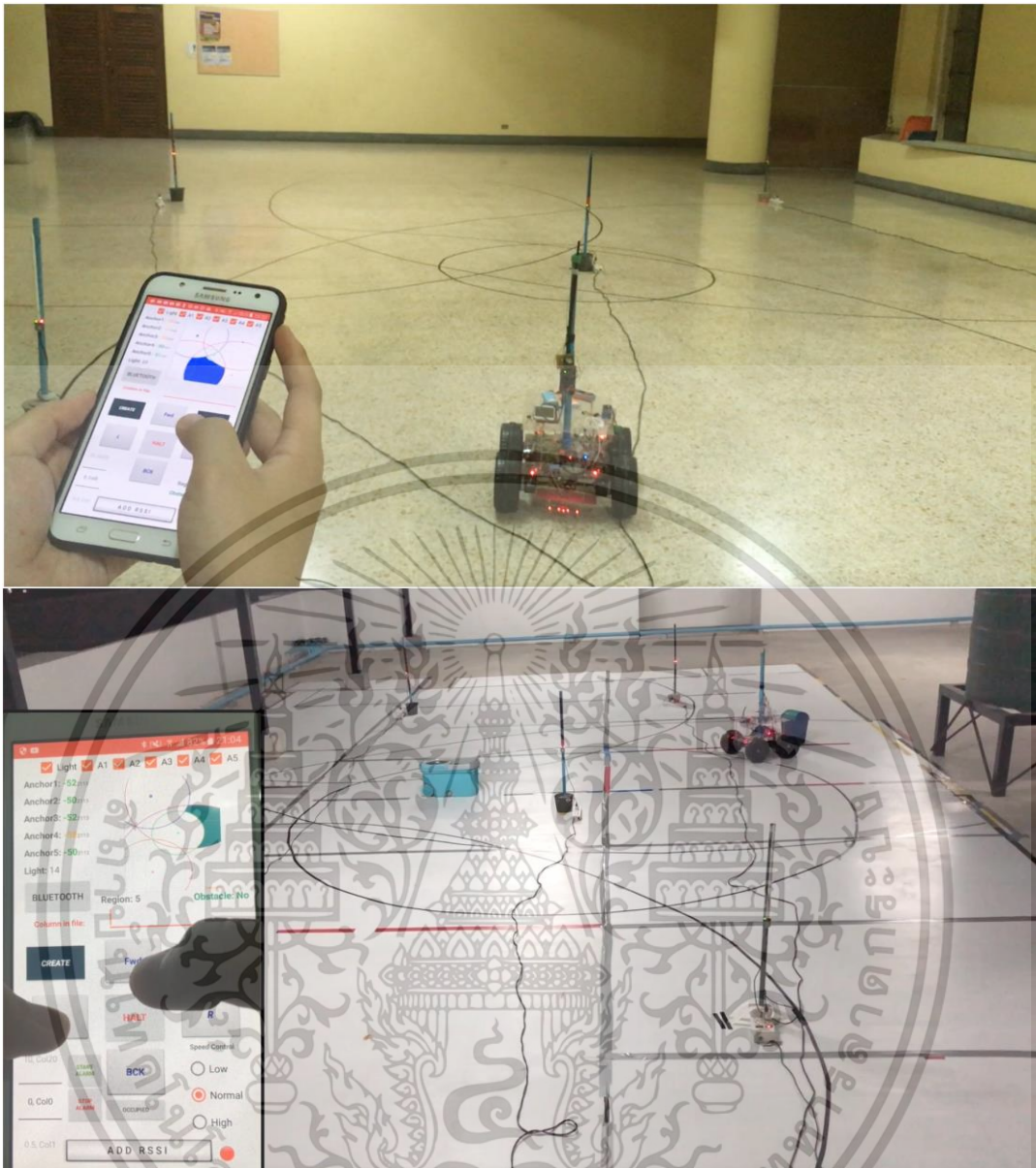
และตารางช่องด้านมุมซ้ายล่างสุดของนั้นจะเป็นเปอร์เซ็นต์รวมจากทุกคลาส แสดงให้เห็นถึงประสิทธิภาพในการจำแนกข้อมูลโดยรวมของทั้งโมเดล ว่ามีเปอร์เซ็นต์การทำนายถูกต้อง (PPV) และเปอร์เซ็นต์ทำนายคลาดเคลื่อน (FDR) ไปเท่าใดบ้าง

จากคอนฟิวชันเมตริกซ์ที่แสดงผลการทำนายของโครงข่ายประสาทเทียมจากสองสนามทดสอบทั้งหลังจากการฝึกสอนและหลังจากการทดสอบกับข้อมูลใหม่ พบว่าผลลัพธ์ของประสิทธิภาพในการจำแนกข้อมูลถูกต้องระหว่างการฝึกสอนและหลังจากการทดสอบของโครงข่ายประสาทเทียมสนามที่ 1 ให้ผลลัพธ์ที่ 92.3 และ 76.0 เปอร์เซ็นต์ตามลำดับ ต่างกันอยู่ถึง 16.3 เปอร์เซ็นต์ซึ่งถือว่าโครงข่ายมีพฤติกรรมโอเวอร์ฟิตติ่งสอดคล้องกับแนวโน้มกราฟรีซีฟเวอร์โอเปอร์เรตติ้งคาแรคเตอร์สติกแต่ในทางตรงกันข้ามนั้น ผลลัพธ์ประสิทธิภาพในการจำแนกข้อมูลถูกต้องระหว่างการฝึกสอนและหลังจากทดสอบของโครงข่ายประสาทเทียมสนามที่ 2 ให้ผลลัพธ์ได้ใกล้เคียงกันที่ 86.6% และ 85.2% ตามลำดับ ต่างกันอยู่เพียง 1.4 เปอร์เซ็นต์เท่านั้น



รูปที่ 5.19. แสดงกราฟรีซีฟเวอร์โอเปอเรตติ้งคาแรคเตอร์สติกของโมเดลโครงข่ายประสาทเทียม สำหรับสนามทดสอบที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.20 ขณะหุ่นยนต์เคลื่อนที่จริงพร้อมทั้งบอกตำแหน่งด้วยพื้นที่ที่ไม่สมมาตรในสนามทดสอบที่ 1 (บน) และ สนามทดสอบที่ 2 (ล่าง)

ในระหว่างการใช้งานจริง เทปสีดำที่แปะลงบนสนามทดสอบลักษณะเป็นโค้งนั้นเป็นเส้นสมมติแทนขอบเขตของแต่ละเสาสัญญาณอ้างอิงที่ผ่านการคำนวณมาแล้ว แน่แน่นอนว่าจุดศูนย์กลางของเส้นโค้งแต่ละเส้นก็คือตำแหน่งของเสาสัญญาณอ้างอิงนั่นเอง จากการทับซ้อนของขอบเขตนั้น เส้นสีดำที่ตัดกันทำให้เกิดพื้นที่ปิดซึ่งจะใช้ในการบอกตำแหน่งหุ่นยนต์ รวมไปถึงวัตถุต่าง ๆ ที่หุ่นยนต์ตรวจพบ จากรูปที่ 5.20 (บน) ขณะหุ่นยนต์เคลื่อนที่ในสนามทดสอบที่ 1 นั้น ค่าความแรงสัญญาณระหว่างเสาสัญญาณเป้าหมายที่ติดตั้งอยู่กับหุ่นยนต์เคลื่อนที่และเสาสัญญาณอ้างอิงทั้ง 5 เสาโดยรอบจะถูกส่งขึ้นไปยังฐานข้อมูลคลาวด์ (Cloud database) พร้อมทั้งจะให้โมเดลโครงข่ายประสาทเทียมที่ถูกฝึกสอนมาแล้วในโปรแกรมแมทแลบดึงไปใช้งานเพื่อจำแนกข้อมูลว่า จากชุดความแรงสัญญาณดังกล่าว หุ่นยนต์อยู่ภายในพื้นที่ปิดใดในสนาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่ได้กล่าวไปแล้วในส่วนของผลทดสอบการฝึกสอนโมเดลโครงข่ายประสาทเทียมถึงแม้ว่าจะมีพฤติกรรมโอเวอร์ฟิตติงและให้ประสิทธิภาพการจำแนกพื้นที่ด้วยข้อมูลทดสอบที่ต่ำกว่าระดับที่ควรจะเป็น แต่ระหว่างการใช้งานจริงโมเดลสามารถให้ผลการจำแนกพื้นที่ที่ไม่สมมาตร ณ ตำแหน่งที่หุ่นยนต์อยู่ได้เป็นอย่างดีมีความคลาดเคลื่อนอยู่บ้างพอประมาณ ซึ่งสาเหตุนี้มาจาก การที่ฝึกสอนโครงข่ายไม่เหมาะสมรวมไปถึงการสื่อสารของข้อมูลภายในเครือข่ายซึ่งมีการเดินทางหลายช่วงต่อ กว่าจะถูกประมวลผลและใช้งานจากอุปกรณ์ปลายทางที่ร้องขอข้อมูล

ในส่วนของการทดสอบที่ 2 (รูปที่ 5.20 ล่าง) เนื่องจากมีการประยุกต์ใช้เทคนิคการสร้างและฝึกสอนโครงข่ายประสาทเทียมรูปแบบต่าง ๆ ทำให้โมเดลสามารถให้ผลลัพธ์ในการจำแนกพื้นที่ที่ไม่สมมาตรได้เป็นอย่างดีและคลาดเคลื่อนน้อยกว่ากรณีของสนามทดสอบที่ 1 อีกทั้งยังมีการทดสอบวางวัตถุสิ่งกีดขวางไว้ในพื้นที่ปิดหมายเลขต่าง ๆ เช่น พื้นที่ปิดหมายเลข 5 และ 9 ของสนามทดสอบที่ 2 เป็นต้น เพื่อให้หุ่นยนต์ทำการระบุว่าพื้นที่ใดมีหรือไม่มีสิ่งกีดขวางปรากฏอยู่ ซึ่งหุ่นยนต์นั้นสามารถตรวจจับวัตถุด้วยการใช้เซนเซอร์อินฟราเรดแบบสัญญาณไม่ต่อเนื่อง (discrete signal) ที่มีระยะการทำงานอยู่ที่ 80-100 เซนติเมตร พร้อมกับทั้งระบุพื้นที่ที่ไม่สมมาตร ณ ตำแหน่งที่ตรวจจับวัตถุได้เป็นอย่างดี แต่อย่างไรก็ตาม ในการใช้งานจริงนั้น หุ่นยนต์เคลื่อนที่จะสามารถตรวจจับวัตถุสิ่งกีดขวางได้ในระยะประมาณ 20-30 เซนติเมตร อันเนื่องมาจาก การสะท้อนลำแสงอินฟราเรดของตัวเซนเซอร์กับผิววัตถุ ที่อาจจะไม่ได้ตั้งฉาก หรือสะท้อนลำแสงอย่างเหมาะสม จึงทำให้บ่อยครั้งที่หุ่นยนต์ต้องเคลื่อนที่เข้าหาวัตถุในระยะใกล้เกินกว่าระยะที่คาดหวัง จึงจะสามารถตรวจจับวัตถุพบ

ตารางที่ 5.3 คอนฟิวชั่นเมทริกซ์ของผลจากการฝึกสอนโครงข่ายประสาทเทียมแสดงในรูปแบบ
จำนวนข้อมูล สนามทดสอบที่ 1 ขนาด 10 เมตร X 10 เมตร

		หมายเลขคลาสหรือหมวดหมู่เป้าหมาย												
		1	2	3	4	5	6	7	8	9	10	11	12	13
หมายเลขคลาสหรือหมวดหมู่ที่ทำนายผลได้	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	7	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	157	4	0	0	0	1	2	0	0
	6	1	0	0	0	1	161	2	0	0	2	0	0	0
	7	0	0	0	0	1	10	146	1	0	0	0	0	0
	8	0	0	0	0	0	1	1	64	0	0	0	1	0
	9	0	0	0	0	0	0	0	1	47	1	0	0	0
	10	0	0	0	1	0	2	0	0	0	174	0	2	2
	11	9	10	9	1	1	2	1	3	2	1	146	5	4
	12	0	0	0	1	0	0	0	0	0	1	0	62	1
	13	0	0	1	0	0	0	0	1	1	0	2	0	153

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.4 คอนฟิวชั่นเมตริกซ์ของผลการฝึกสอนโครงข่ายประสาทเทียมแสดงในรูปแบบร้อยละ
สนามทดสอบที่ 1 ขนาด 10 เมตร X 10 เมตร

		Precision (%)	False discovery rate (%)	True positive rate (%)	False negative rate (%)
หมายเลขคลาสหรือหมวดหมู่ของพื้นที่ที่ไม่ผสมมาตร	1	NaN	NaN	0	100
	2	NaN	NaN	0	100
	3	NaN	NaN	0	100
	4	100	0	70	30
	5	95.7	4.3	98.1	1.9
	6	96.4	3.6	89.4	10.6
	7	92.4	7.6	97.3	2.7
	8	95.5	4.5	91.4	8.6
	9	95.9	4.1	94.0	6.0
	10	96.1	3.9	96.7	3.3
	11	75.3	24.7	97.3	2.7
	12	95.4	4.6	98.6	11.4
	13	96.8	3.2	95.6	4.4
Overall	92.3	7.7	-	-	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.5 คอนฟิวชันเมทริกซ์ของผลจากการทดสอบโครงข่ายประสาทเทียมในรูปแบบจำนวน
ข้อมูล สนามทดสอบที่ 1 ขนาด 10 เมตร X 10 เมตร

		หมายเลขคลาสหรือหมวดหมู่เป้าหมาย												
		1	2	3	4	5	6	7	8	9	10	11	12	13
หมายเลขคลาสหรือหมวดหมู่ที่ทำนายผลได้	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0	0	0	0	0
	4	0	0	0	1	0	0	0	0	0	0	0	0	0
	5	0	0	0	0	12	2	0	0	0	1	0	0	0
	6	0	0	0	0	0	10	0	0	0	2	0	0	0
	7	0	0	0	0	0	1	13	1	0	0	0	0	0
	8	0	0	0	0	0	1	1	2	0	0	0	1	0
	9	0	0	0	0	0	0	0	1	3	1	0	0	0
	10	0	0	0	1	0	2	0	0	0	18	0	1	1
	11	0	1	0	0	0	0	1	0	1	0	15	1	3
	12	0	0	0	1	0	0	0	0	0	1	0	5	1
	13	0	0	0	0	0	0	0	1	0	0	1	0	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.6 คอนฟิวชั่นเมตริกซ์ของผลการทดสอบโครงข่ายประสาทเทียมแบบร้อยละ สนาม
ทดสอบที่ 1 ขนาด 10 เมตร X 10 เมตร

		Precision (%)	False discovery rate (%)	True positive rate (%)	False negative rate (%)
หมายเลขคลาสหรือหมวดหมู่ของพื้นที่ที่ไม่ผสมมาตร	1	NaN	NaN	NaN	NaN
	2	NaN	NaN	0	100
	3	NaN	NaN	NaN	NaN
	4	100	0	33.3	66.7
	5	80.0	20.0	100	0
	6	83.3	16.7	62.5	37.5
	7	86.7	13.3	86.7	13.3
	8	40.0	60.0	40.0	60.0
	9	60.0	40.0	75.0	25.0
	10	78.3	21.7	78.3	21.7
	11	68.2	31.8	93.8	6.2
	12	62.5	37.5	62.5	37.5
	13	86.7	13.3	72.2	27.8
Overall	76.0	24.0	-	-	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 การตั้งค่าไฮเปอร์พารามิเตอร์ของโครงข่ายประสาทเทียมที่ให้ผลการจำแนกที่ดีที่สุด
โครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนามทดสอบที่ 2

ประเภทการตั้งค่า	การตั้งค่า
อินพุทเลเยอร์ / โหนด (Input layer / node)	อินพุทเลเยอร์ = 1 อินพุทโหนด = 5
ฮิดเดนเลเยอร์ / ฮิดเดนโหนด (Hidden layer / Hidden node)	ฮิดเดนเลเยอร์ = 2 ฮิดเดนโหนด = 80
เอาต์พุทเลเยอร์ / เอาต์พุทโหนด (Output layer / Output node)	เอาต์พุทเลเยอร์ = 1 เอาต์พุทโหนด = 13
ฟังก์ชันกระตุ้น (Activation function)	ฟังก์ชัน tansig สำหรับฮิดเดนเลเยอร์ ฟังก์ชัน softmax สำหรับชั้นเอาต์พุทเลเยอร์
อัตราการฝึกสอนโครงข่าย (Learning rate)	0.01
ค่าลงโทษแบบแอลทู เรกูลาไรเซชัน (L2 Regularization)	0.01
อัลกอริทึมฝึกสอนโครงข่ายประสาทเทียม (Training algorithm)	สเกลคอนจูเกตกราเดียนท์ แบ็คพรอพาเกชัน (trainscg)
การแก้ปัญหาความไม่สมดุลของคลาส (Class imbalance fix)	ใช่
เทคนิคที่เลือกใช้แก้ปัญหาความไม่สมดุล (Class imbalance strategy)	เทคนิคการเพิ่มสัญญาณรบกวนแบบไวท์เกาส์ เขียนด้วยอัตราส่วนเท่ากับ 35 SNR
ค่า K ชุดข้อมูล ของการทำครอสวาไลเดชัน (K-fold cross validation)	K = 10 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.8 จำนวนพิกัดข้อมูลที่ใช้ฝึกสอนในแต่ละคลาสหรือพื้นที่ที่ไม่สมมาตรของโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตร สนามทดสอบที่ 2

คลาส / หมายเลขพื้นที่ที่ไม่สมมาตร	จำนวนชุดความแรงสัญญาณในผลการทำนายแต่ละคลาส
คลาส 1 / พื้นที่ที่ไม่สมมาตรหมายเลข 1	1202 จุด
คลาส 2 / พื้นที่ที่ไม่สมมาตรหมายเลข 2	20 จุด
คลาส 3 / พื้นที่ที่ไม่สมมาตรหมายเลข 3	28 จุด
คลาส 4 / พื้นที่ที่ไม่สมมาตรหมายเลข 4	2 จุด
คลาส 5 / พื้นที่ที่ไม่สมมาตรหมายเลข 5	9 จุด
คลาส 6 / พื้นที่ที่ไม่สมมาตรหมายเลข 6	33 จุด
คลาส 7 / พื้นที่ที่ไม่สมมาตรหมายเลข 7	18 จุด
คลาส 8 / พื้นที่ที่ไม่สมมาตรหมายเลข 8	1 จุด
คลาส 9 / พื้นที่ที่ไม่สมมาตรหมายเลข 9	27 จุด
คลาส 10 / พื้นที่ที่ไม่สมมาตรหมายเลข 10	12 จุด
คลาส 11 / พื้นที่ที่ไม่สมมาตรหมายเลข 11	11 จุด
คลาส 12 / พื้นที่ที่ไม่สมมาตรหมายเลข 12	2 จุด
คลาส 13 / พื้นที่ที่ไม่สมมาตรหมายเลข 13	0 จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.9 คอนฟิวชั่นเมทริกซ์ของผลจากการฝึกสอนโครงข่ายประสาทเทียมในรูปแบบจำนวน
ข้อมูล สนามทดสอบที่ 2 ขนาด 6 เมตร X 10 เมตร

		หมายเลขคลาสหรือหมวดหมู่เป้าหมาย												
		1	2	3	4	5	6	7	8	9	10	11	12	13
หมายเลขคลาสหรือหมวดหมู่ที่ทำนายผลได้	1	1202	20	28	2	9	33	18	1	27	12	11	2	0
	2	25	1073	21	29	2	19	22	9	2	3	0	0	0
	3	19	36	1116	49	1	9	61	59	6	21	1	0	0
	4	0	94	79	1180	0	0	23	118	0	0	0	0	0
	5	11	9	0	0	1332	36	0	0	4	1	1	1	0
	6	38	29	1	0	6	1173	10	0	35	13	3	1	1
	7	15	47	31	12	0	20	1114	48	13	37	1	0	0
	8	0	24	41	78	0	2	54	1106	0	22	0	0	0
	9	20	8	7	0	0	39	16	0	1120	22	83	32	1
	10	18	9	26	0	0	12	29	9	46	1196	43	0	0
	11	2	1	0	0	0	4	3	0	44	22	1039	32	39
	12	0	0	0	0	0	3	0	0	53	0	87	1276	5
	13	0	0	0	0	0	0	0	0	0	1	81	6	1304

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.10 คอนฟิวชั่นเมทริกซ์ของผลการฝึกสอนโครงข่ายประสาทเทียมแบบร้อยละ สนาม
ทดสอบที่ 2 ขนาด 6 เมตร X 10 เมตร

		Precision (%)	False discovery rate (%)	True positive rate (%)	False negative rate (%)
หมายเลขคลาสหรือหมวดหมู่ของพื้นที่ที่ไม่ผสมมาตร	1	88.1	11.9	89.0	11.0
	2	89.0	11.0	79.5	20.5
	3	81.0	19.0	82.7	17.3
	4	79.0	21.0	87.4	12.6
	5	95.5	4.5	98.7	1.3
	6	89.5	10.5	86.9	13.1
	7	83.3	16.7	82.5	17.5
	8	83.3	16.7	81.9	18.1
	9	83.1	16.9	83.0	17.0
	10	86.2	13.8	98.6	11.4
	11	87.6	12.4	77.0	23.0
	12	89.6	10.4	94.5	5.5
	13	93.7	6.3	96.6	3.4
Overall	86.8	13.2	-	-	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.11 คอนฟิวชั่นเมทริกซ์ของผลจากการทดสอบโครงข่ายประสาทเทียมในรูปแบบจำนวน
ข้อมูล สนามทดสอบที่ 2 ขนาด 6 เมตร X 10 เมตร

		หมายเลขคลาสหรือหมวดหมู่เป้าหมาย												
		1	2	3	4	5	6	7	8	9	10	11	12	13
หมายเลขคลาสหรือหมวดหมู่ที่ทำนายผลได้	1	177	2	8	0	2	6	2	0	2	0	1	0	0
	2	6	149	2	8	1	4	6	2	0	0	0	0	0
	3	5	6	147	8	0	1	16	3	1	2	0	0	0
	4	0	20	12	176	0	0	4	13	0	0	0	0	0
	5	3	0	0	0	186	9	0	0	1	0	0	0	0
	6	10	7	0	0	1	168	1	0	9	0	1	0	0
	7	1	7	7	1	0	3	179	7	2	7	0	0	0
	8	0	4	8	15	0	0	8	178	0	2	0	0	0
	9	3	0	2	0	0	5	1	0	154	4	8	12	0
	10	2	1	4	0	0	4	5	0	11	183	6	0	0
	11	1	0	0	0	0	0	1	0	11	4	147	6	6
	12	0	0	0	0	0	1	0	0	7	0	15	186	0
	13	0	0	0	0	0	0	0	0	0	0	10	2	213

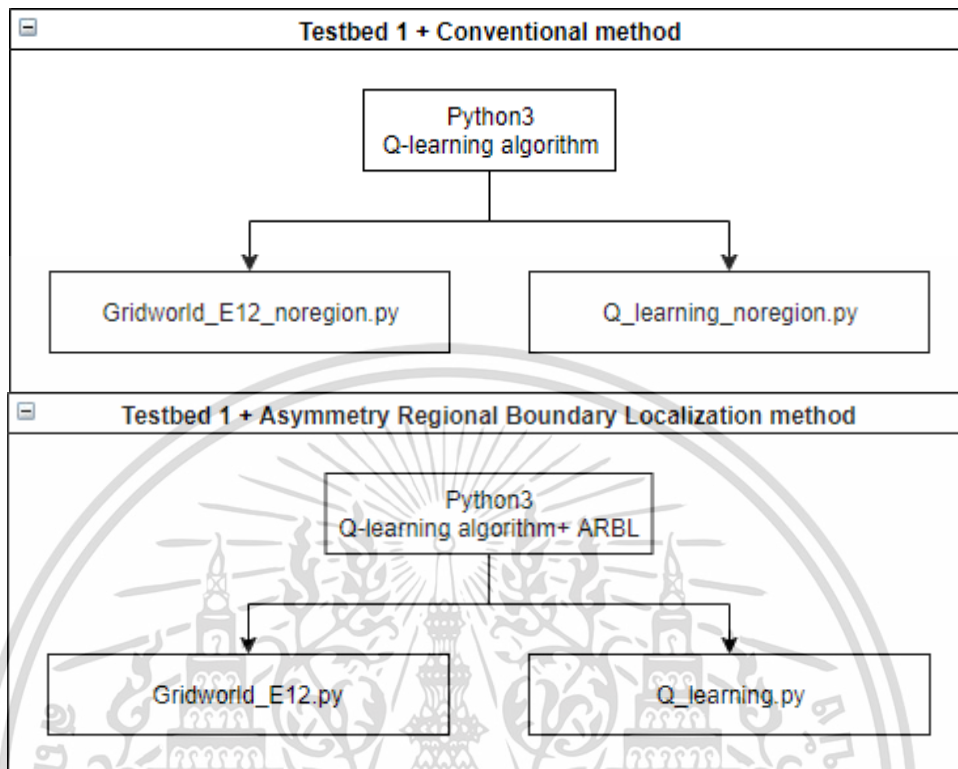
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.12 คอนฟิวชั่นเมทริกซ์ของผลการทดสอบโครงข่ายประสาทเทียมแบบร้อยละ สนาม
ทดสอบที่ 2 ขนาด 6 เมตร X 10 เมตร

		Precision (%)	False discovery rate (%)	True positive rate (%)	False negative rate (%)
หมายเลขคลาสหรือหมวดหมู่ของพื้นที่ที่ไม่ผสมมาตร	1	88.5	11.5	85.1	14.9
	2	83.7	16.3	76.0	24.0
	3	77.8	22.2	77.4	22.6
	4	78.2	21.8	84.6	15.4
	5	93.0	7.0	97.9	2.1
	6	85.3	14.7	83.6	16.4
	7	83.6	16.4	80.3	19.7
	8	32.8	17.2	87.7	12.3
	9	81.5	18.5	77.4	22.6
	10	84.7	15.3	90.6	9.4
	11	83.5	16.5	78.2	21.8
	12	89.0	11.0	90.3	9.7
	13	94.7	5.3	97.3	2.7
Overall	85.2	14.8	-	-	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

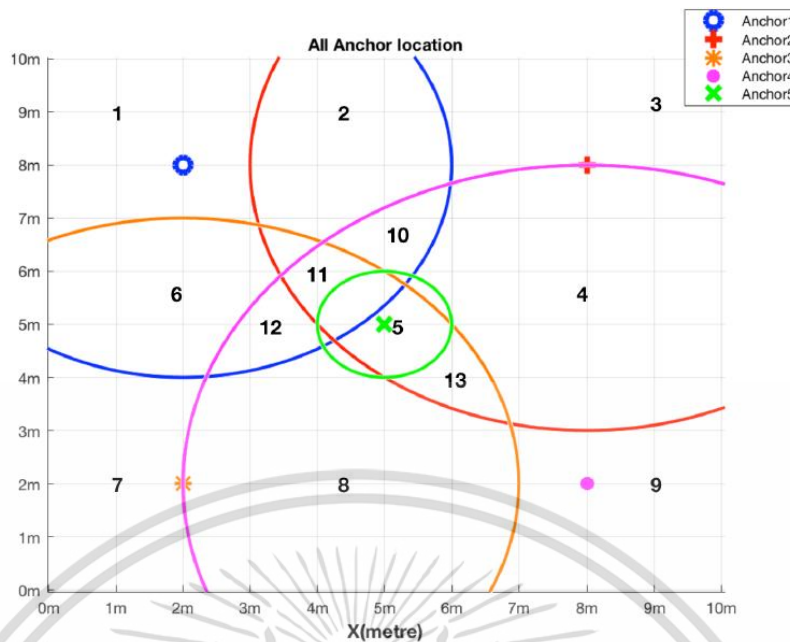
5.5 ผลการทดสอบการค้นหาเส้นทางของหุ่นยนต์เคลื่อนที่ ด้วยการเรียนรู้แบบเสริมกำลัง



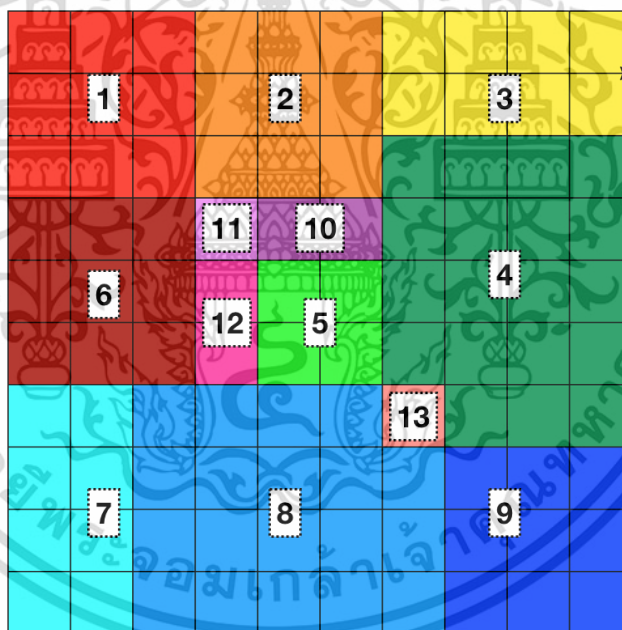
รูปที่ 5.21 โครงสร้างของไฟล์ต่าง ๆ ของแต่ละโปรเจกต์สำหรับการใช้ในการสร้างสนามทดสอบจำลอง สนามทดสอบที่ 1

จากการทดลองที่ผ่านมาแล้วนั้น ทำให้ได้แผนที่ภายในสนามที่บอกที่ตั้งของสิ่งกีดขวางต่าง ๆ ในรูปของพื้นที่ปิดที่ไม่สมมาตร แต่เนื่องจากปัญหาทางด้านสถานที่ทำให้ไม่สามารถทดสอบการสร้างเส้นทางและการเคลื่อนที่หลบหลีกสิ่งกีดขวางของหุ่นยนต์ขณะทำงานจริงได้ ดังนั้นจึงได้มีการออกแบบสนามจำลองขึ้นมาเป็นกริดแบบสองมิติด้วยไลบรารีที่ชื่อว่า เทอร์เทิล ซึ่งเป็นหนึ่งในไลบรารีพื้นฐานของภาษาไพธอน เพื่อให้ง่ายต่อการตรวจสอบและพิจารณาผลจากการทดลอง โดยกำหนดให้ช่องแต่ละช่องในตารางจะเทียบเท่ากับพื้นที่ในสนาม 1 ตารางเมตร

ในการทดลองนี้จะเป็นการเปรียบเทียบประสิทธิภาพของการบอกตำแหน่งแบบทั่วไป (Conventional method) ซึ่งระบบจะพิจารณาสิ่งแวดล้อมต่าง ๆ เป็นพิกัดตำแหน่งกับการบอกตำแหน่งแบบเออาร์บีแอล (ARBL) ซึ่งระบบจะพิจารณาสิ่งแวดล้อมในลักษณะของพื้นที่ที่ไม่สมมาตร ดังนั้นจึงได้มีการแบ่งสภาพแวดล้อมทดสอบเป็นสองโปรเจกต์ออกจากกัน ซึ่งแต่ละโปรเจกต์นั้นได้ทำการสร้างไฟล์นามสกุล .py ออกเป็น 2 ไฟล์คือไฟล์ระบุคุณลักษณะของสนามและไฟล์สร้างระบบการเรียนรู้แบบคิว ซึ่งทั้งสองไฟล์นั้นจะทำงานสอดคล้องกันตั้งแต่เริ่มจนจบกระบวนการ รูปที่ 5.21 แสดงส่วนประกอบของไฟล์ต่าง ๆ ในแต่ละโปรเจกต์



รูปที่ 5.22 พื้นที่ที่ไม่สมมาตรพร้อมหมายเลขกำกับภายในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตร และยาว 10 เมตร



รูปที่ 5.23 แสดงภาพตัวอย่างการจำลองสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร

แบบกริดสองมิติด้วยเทอร์เทิลไพธอนไลบรารีพร้อมหมายเลขกำกับพื้นที่ที่ไม่สมมาตร

จากรูปที่ 5.23 จะเห็นว่าสนามถูกแบ่งออกเป็นพื้นที่สีเหลี่ยมย่อยที่มีสีประจำตัวตามตำแหน่งของพื้นที่ปิดที่ไม่สมมาตรต่าง ๆ ทั้ง 13 โซนที่ได้กำหนดไว้ก่อนหน้านี้แล้ว ตัวอย่างเช่น สีแดง คือพื้นที่ที่ไม่สมมาตรหมายเลข 1 สีส้มคือพื้นที่ที่ไม่สมมาตรหมายเลข 2 เป็นต้น และสำหรับกรณีนี้หุ่นยนต์ตรวจพบสิ่งกีดขวางในพื้นที่หมายเลขใด พื้นที่นั้น ๆ จะถูกกำหนดสีให้เป็นสีดำซึ่งหมายความว่าพื้นที่ห้ามเข้าและหุ่นยนต์ควรจะต้องหาทางหลีกเลี่ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.13 - 5.18 แสดงเมทริกซ์ที่เก็บข้อมูลการประมวลผลภายในอัลกอริทึมการเรียนรู้แบบคิว แต่ละช่องของตารางจะแทนพิกัดหนึ่ง ๆ ในสนามทดสอบที่จำลอง เมทริกซ์ที่ใช้บ่งบอกถึงความคืบหน้าของกระบวนการเรียนรู้จะประกอบไปด้วย 3 เมทริกซ์ด้วยกันคือ 1. ค่ารางวัล (reward) 2. ค่าการเรียนรู้แบบคิว (Q-value) 3. นโยบายในการแก้ไขปัญหา (policy) โดยที่เมทริกซ์รางวัลจะแสดงการกำหนดค่ารางวัลหรือบทลงโทษของหุ่นยนต์เมื่ออยู่ในตำแหน่งหรือช่องใด ๆ ในตาราง จากตารางที่ 5.13 จะเห็นว่าช่องตารางแทบจะทั้งหมดถูกแทนด้วยค่า -0.1 แม้แต่ตำแหน่งเริ่มต้นของหุ่นยนต์เองก็ตาม (ช่องตารางที่มีพื้นสีเขียว) ซึ่งกำหนดให้เป็นบทลงโทษสำหรับหุ่นยนต์ ณ ทุกตำแหน่งที่ไม่ใช่ตำแหน่งเป้าหมาย ขณะที่ตำแหน่งเป้าหมายหรือช่องตารางที่มีพื้นสีแดงมีค่ารางวัลเป็น 10 ส่วนช่องตารางที่มีพื้นสีเทานั้นคือโซนของพิกัดที่ถูกกำหนดไว้ว่ามีสิ่งกีดขวางอยู่จึงกลายเป็นพื้นที่ห้ามเข้า ตำแหน่งดังกล่าวจึงไม่มีค่ารางวัลสำหรับหุ่นยนต์หรือกำหนดให้ค่าเป็น 0

ตารางในส่วนที่ 2 คือเมทริกซ์ค่าคิวซึ่งจะเป็นส่วนที่ใช้ในการประมวลผลค่าคุณภาพของการเลือกกริยาใดกริยาหนึ่งในสถานะหรือช่องตารางนั้น ๆ ว่าจะส่งผลต่อการได้มาซึ่งค่ารางวัลสูงสุดอย่างไร ตัวอย่างเช่น ในตารางที่ 5.14 ช่องตาราง ณ ตำแหน่งเริ่มต้นของหุ่นยนต์ (ช่องสีเขียว) มีค่าคิวเท่ากับ 0.94 และช่องตารางด้านล่างกับด้านบนตำแหน่งเริ่มต้นจะมีค่า 0.65 เท่ากันตามลำดับ ในขณะที่ช่องตารางด้านซ้ายมือของตำแหน่งเริ่มต้นนั้นมีค่าเท่ากับ 1.3 นั้นจึงหมายความว่าหุ่นยนต์จะเลือกเคลื่อนที่จากตำแหน่งเริ่มต้นไปยังช่องทางซ้ายมือมากกว่าที่จะเลือกเคลื่อนที่ไปช่องด้านบนหรือล่าง

ขณะเดียวกัน ในตารางที่ 5.14 ช่องตารางด้านซ้ายและด้านบนของตำแหน่งเป้าหมาย (ช่องสีแดง) มีค่าคิวเท่ากับ 10 และ 9.99 ตามลำดับ นั้นหมายความว่ายิ่งเข้าใกล้เป้าหมายมากเท่าไร ค่าคุณภาพของกริยาที่จะส่งผลให้หุ่นยนต์เปลี่ยนไปเป็นตำแหน่งเป้าหมายนั้นยิ่งมีค่าสูงที่สุดในบรรดาค่าคิวของทุกตำแหน่งในสนาม ส่วนช่องตารางที่เป็นตำแหน่งเป้าหมายและตำแหน่งของพื้นที่สิ่งกีดขวางจะมีค่าคิวเท่ากับ 0 เนื่องจากว่าเป็นสถานะสุดท้ายของหุ่นยนต์หรือเป็นตำแหน่งที่ไม่มีทางให้ไปต่อได้แล้ว

ตารางในส่วนที่ 3 เป็นตารางแสดงเมทริกซ์นโยบายการแก้ไขปัญหาซึ่งจะทำหน้าที่เก็บข้อมูลกริยาต่าง ๆ ที่หุ่นยนต์เลือก ให้สอดคล้องกับเมทริกซ์คิวในหนึ่งตำแหน่ง หุ่นยนต์จะมีกริยาให้เลือกกระทำได้ 4 รูปแบบด้วย ด้วยกันตามที่ได้กล่าวไว้แล้วคือ เดินหน้า ไปทางซ้าย ไปทางขวา หรือถอยกลับ ซึ่งแต่ละกริยานั้นจะมีค่าคิวเป็นค่าประจำของตัวเอง อย่างไรก็ตาม เมื่อระบบคำนวณหากริยาที่ส่งผลให้เกิดค่าคิวดีที่สุดในตำแหน่งนั้น ๆ ค่าคิวและกริยาจะถูกบันทึกลงในเมทริกซ์คิว (Q matrix) และเมทริกซ์นโยบายแก้ไขปัญหา (policy matrix) ตามลำดับ ดังนั้นเมทริกซ์นโยบายแก้ปัญหาดังแสดงในตารางที่ 5.15 จึงทำหน้าที่เก็บบันทึกชุดการกระทำที่ให้ค่าคิวดีที่สุดในระบบคำนวณมาให้ และเมื่อพิจารณาการกระทำของแต่ละช่องในตาราง จะเห็นว่ากริยาของทุก ๆ ตำแหน่งจะส่งผลให้หุ่นยนต์เคลื่อนที่เข้าหาตำแหน่งเป้าหมายทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.13 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป

XY	0	1	2	3	4	5	6	7	8	9
0	-0.1	-0.1	0.00	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1
1	-0.1	-0.1	0.00	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	10.00
2	-0.1	-0.1	0.00	-0.1	-0.1	-0.1	0.00	0.00	0.00	0.00
3	-0.1	-0.1	0.00	-0.1	-0.1	-0.1	0.00	-0.1	-0.1	-0.1
4	-0.1	-0.1	0.00	-0.1	-0.1	-0.1	0.00	-0.1	0.00	0.00
5	-0.1	0.00	0.00	0.00	-0.1	-0.1	0.00	-0.1	-0.1	-0.1
6	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	0.00	-0.1	-0.1	-0.1
7	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1
8	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1
9	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1

ตารางที่ 5.14 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป

XY	0	1	2	3	4	5	6	7	8	9
0	-0.41	-0.5	-	4.62	5.35	6.14	6.96	7.86	8.88	9.99
1	-0.26	-0.38	-	5.34	6.15	6.99	7.9	8.9	10	0
2	-0.09	-0.24	-	4.59	5.37	6.17	-	-	-	-
3	0.12	-0.07	-	3.93	4.67	5.41	-	0.38	0.15	-0.07
4	0.38	0.11	-	3.32	3.99	4.71	-	0.66	-	-
5	0.68	-	-	-	3.33	4.04	-	0.97	0.62	0.31
6	1.03	1.42	1.81	2.27	2.79	3.39	-	1.34	0.95	0.65
7	0.67	0.99	1.39	1.77	2.21	2.8	2.23	1.73	1.3	0.94
8	0.39	0.59	1.01	1.34	1.72	2.23	1.71	1.3	0.98	0.65
9	0.15	0.33	0.64	0.92	1.28	1.71	1.23	0.91	0.64	0.32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.15 แสดงนโยบายที่ดีที่สุดในการแก้ปัญหา (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป

X/Y	0	1	2	3	4	5	6	7	8	9
0	D	D		R	D	D	D	D	D	D
1	D	D		R	R	R	R	R	R	
2	D	D		R	R	U				
3	D	L		R	R	U		D	L	L
4	D	L		R	R	U		D		
5	D				R	U		D	L	D
6	R	R	R	R	R	U		D	L	L
7	U	U	U	U	R	U	L	L	L	L
8	U	R	U	U	R	U	L	U	L	L
9	U	R	U	U	U	L	U	U	U	U

ขณะที่ตารางที่ 5.16 ถึง 5.18 จะแสดงตารางเมทริกซ์ที่เก็บข้อมูลการประมวลผลภายในอัลกอริทึมการเรียนรู้แบบคิวด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล โดยจะอ้างอิงตำแหน่งของสิ่งกีดขวางที่พิกัดเดียวกันกับการทดลองที่แล้ว เนื่องจากการบอกตำแหน่งแบบเออาร์บีแอลจะพิจารณาตำแหน่งของสิ่งต่าง ๆ ในลักษณะของพื้นที่ ทำให้สิ่งกีดขวางภายในสนามทดสอบถูกแปรสภาพกลายเป็นพื้นที่ที่ไม่สมมาตรไปด้วย จะเห็นได้ว่าสิ่งกีดขวางนั้นได้ครอบคลุมพื้นที่ที่ไม่สมมาตรหมายเลข 1, 6, 4 และ 12 ดังแสดงในตารางที่ 5.16 ถึง 5.18 ด้วยช่องตารางที่มีพื้นสีเทา ตำแหน่งเริ่มต้นของหุ่นยนต์ และตำแหน่งเป้าหมายที่ต้องเคลื่อนที่ไปถึงยังคงเป็นพิกัดเดียวกันกับการทดลองที่แล้ว รวมไปถึงการตั้งค่ารางวัลและบทลงโทษในแต่ละพิกัดตำแหน่งที่ยังใช้ค่าเดิมอีกด้วย

ตารางที่ 5.16 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล

X/Y	0	1	2	3	4	5	6	7	8	9
0	0.00	0.00	0.00	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10
1	0.00	0.00	0.00	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	10.00
2	0.00	0.00	0.00	-0.10	-0.10	-0.10	0.00	0.00	0.00	0.00
3	0.00	0.00	0.00	-0.10	-0.10	-0.10	0.00	0.00	0.00	0.00
4	0.00	0.00	0.00	0.00	-0.10	-0.10	0.00	0.00	0.00	0.00
5	0.00	0.00	0.00	0.00	-0.10	-0.10	0.00	0.00	0.00	0.00
6	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	0.00	0.00	0.00	0.00
7	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10
8	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10
9	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

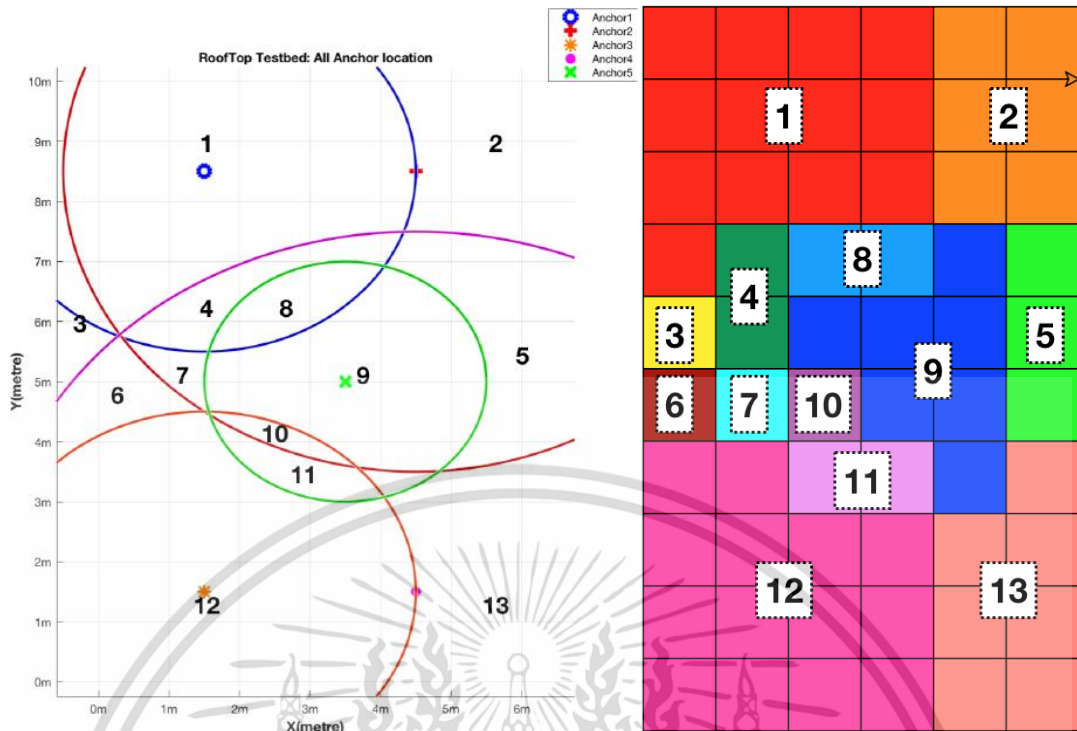
ตารางที่ 5.17 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล

XY	0	1	2	3	4	5	6	7	8	9
0	0.00	0.00	0.00	4.82	5.48	6.21	7.02	7.90	8.90	10.00
1	0.00	0.00	0.00	5.47	6.21	7.02	7.91	8.90	10.00	0.00
2	0.00	0.00	0.00	4.82	5.48	6.21	0.00	0.00	0.00	0.00
3	0.00	0.00	0.00	4.22	4.83	5.49	0.00	0.00	0.00	0.00
4	0.00	0.00	0.00	0.00	4.24	4.83	0.00	0.00	0.00	0.00
5	0.00	0.00	0.00	0.00	3.70	4.24	0.00	0.00	0.00	0.00
6	1.68	2.01	2.38	2.78	3.22	3.70	0.00	0.00	0.00	0.00
7	1.37	1.68	2.01	2.38	2.77	3.21	2.78	2.38	2.01	1.68
8	1.10	1.37	1.68	1.99	2.37	2.77	2.37	2.01	1.67	1.38
9	0.84	1.09	1.37	1.66	1.99	2.36	2.00	1.67	1.37	1.10

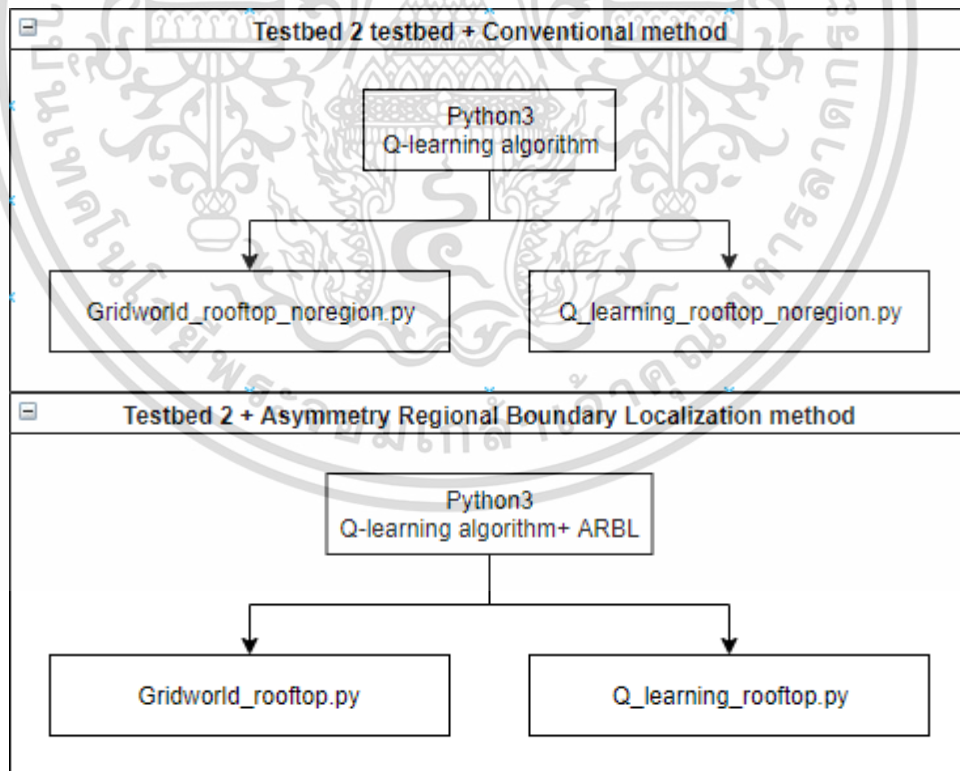
ตารางที่ 5.18 แสดงนโยบายในการแก้ปัญหาที่ดีที่สุด (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 1 ขนาดกว้าง 10 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล

XY	0	1	2	3	4	5	6	7	8	9
0				R	R	D	D	R	R	D
1				R	R	D	D	R	R	
2				R	R	U				
3				R	R	U				
4					U	U				
5					U	U				
6	R	R	R	R	R	U				
7	U	R	U	U	U	U	L	L	L	L
8	U	R	U	U	U	U	U	U	U	U
9	U	R	U	R	R	U	U	U	L	L

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.24 พื้นที่ที่ไม่สมมาตรพร้อมหมายเลขกำกับภายในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร (ซ้าย) ภาพตัวอย่างการจำลองสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตรแบบกริดสองมิติด้วยเทอร์เทิลไพธอนไลบรารีพร้อมหมายเลขกำกับพื้นที่ที่ไม่สมมาตร (ขวา)



รูปที่ 5.25 โครงสร้างของไฟล์ต่าง ๆ ของแต่ละโปรเจกต์สำหรับการใช้ในการสร้างสนามทดสอบจำลองสนามทดสอบที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นเดียวกับกับสนามทดสอบที่ 1 สำหรับสนามทดสอบที่ 2 ได้มีการออกแบบลักษณะการทำงานของอัลกอริทึมเหมือนกัน การตั้งค่าสำหรับกำหนดทิศทางการเล่นรู้เหมือนกันแต่ด้วยขนาดของสนามที่เล็กกว่า ทำให้กระบวนการประมวลผลนั้นเพื่อหานโยบายแก้ปัญหาที่ดีที่สุด (optimal policy) นั้นใช้ระยะเวลาสั้นกว่าสนามทดสอบที่ 1 ตำแหน่งจุดเริ่มต้น / เป้าหมาย รวมไปถึงขนาดของพื้นที่ที่ไม่สมมาตรนั้นถูกกำหนดขึ้นต่างจากสนามทดสอบที่ 1 ซึ่งข้อกำหนดต่าง ๆ จะมีรายละเอียดดังต่อไปนี้

1. ตำแหน่งเริ่มต้นของหุ่นยนต์อยู่ที่พิกัด 4, 0
2. ตำแหน่งเป้าหมายของหุ่นยนต์อยู่ที่พิกัด 0, 5
3. ทุกช่องตารางถูกกำหนดให้มีค่ารางวัลเท่ากับ -0.1 (ค่าลงโทษ) ยกเว้นตำแหน่งเป้าหมายที่มีค่ารางวัลเท่ากับ 10 และตำแหน่งที่มีสิ่งกีดขวางมีค่าเท่ากับ 0
4. สำหรับกรณีสนามทดสอบจำลองแบบปกติ จะถูกกำหนดให้พิกัดดังต่อไปนี้คือ ตำแหน่งที่มีสิ่งกีดขวางอยู่ ได้แก่ พิกัดที่ (4, 0), (3, 1), (3, 2), (3, 3), (3, 4), (3, 5), (3, 6), (3, 7), (3, 8), (3, 2), (3, 3), (3, 4), และ (3, 5) แต่สำหรับกรณีสนามทดสอบจำลองแบบเออาร์พีแอล พิกัดสิ่งกีดขวางจะครอบคลุมพื้นที่ปิดที่ไม่สมมาตร หมายเลข 3, 4, 7, 8, 9 และ 12

ตารางที่ 5.19 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป

X/Y	0	1	2	3	4	5
0	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1
1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1
2	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1
3	10.00	0.00	0.00	0.00	-0.1	-0.1
4	0.00	0.00	-0.1	0.00	-0.1	-0.1
5	-0.1	0.00	-0.1	0.00	-0.1	-0.1
6	-0.1	0.00	-0.1	-0.1	-0.1	-0.1
7	-0.1	0.00	-0.1	-0.1	-0.1	-0.1
8	-0.1	0.00	-0.1	-0.1	-0.1	-0.1
9	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.20 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป

X/Y	0	1	2	3	4	5
0	7.91	7.02	6.22	5.49	4.84	4.26
1	8.90	7.91	7.02	6.22	5.43	4.84
2	10.00	8.90	7.91	7.02	6.22	5.49
3	0.00	0.00	0.00	0.00	5.49	4.84
4	0.00	0.00	2.07	0.00	4.84	4.26
5	0.29	0.00	2.43	0.00	4.26	3.73
6	0.49	0.00	2.82	3.25	3.73	3.25
7	0.71	0.00	2.43	2.82	3.25	2.82
8	0.94	0.00	2.07	2.43	2.82	2.43
9	1.19	1.46	1.76	2.08	2.43	2.08

ตารางที่ 5.21 แสดงนโยบายในการแก้ปัญหาที่ดีที่สุด (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบพิกัดทั่วไป

X/Y	0	1	2	3	4	5
0	D	L	L	L	L	L
1	D	D	D	D	D	D
2	D	L	L	L	L	L
3					U	U
4			D		U	L
5	D		D		U	U
6	D		R	R	U	L
7	D		R	R	U	L
8	D		R	R	U	U
9	R	R	R	R	U	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.22 แสดงค่ารางวัล (Reward) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล

X/Y	0	1	2	3	4	5
0	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10
1	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10
2	-0.10	-0.10	-0.10	-0.10	-0.10	-0.10
3	10.00	0.00	0.00	0.00	0.00	-0.10
4	0.00	0.00	0.00	0.00	0.00	-0.10
5	-0.10	0.00	-0.10	0.00	0.00	-0.10
6	0.00	0.00	-0.10	-0.10	0.00	-0.10
7	0.00	0.00	0.00	0.00	-0.10	-0.10
8	0.00	0.00	0.00	0.00	-0.10	-0.10
9	0.00	0.00	0.00	0.00	-0.10	-0.10

ตารางที่ 5.23 แสดงค่าคิว (Q-value) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล

X/Y	0	1	2	3	4	5
0	7.91	7.01	6.20	5.46	4.77	4.16
1	8.90	7.90	7.00	6.18	5.45	4.79
2	10.00	8.9	7.91	7.01	6.21	5.47
3	0.00	0.00	0.00	0.00	0.00	4.80
4	0.00	0.00	0.00	0.00	0.00	4.20
5	0.00	0.00	0.00	0.00	0.00	3.65
6	0.00	0.00	0.00	0.00	0.00	3.15
7	0.00	0.00	0.00	0.00	2.26	2.69
8	0.00	0.00	0.00	0.00	1.86	2.27
9	0.00	0.00	0.00	0.00	1.54	1.88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

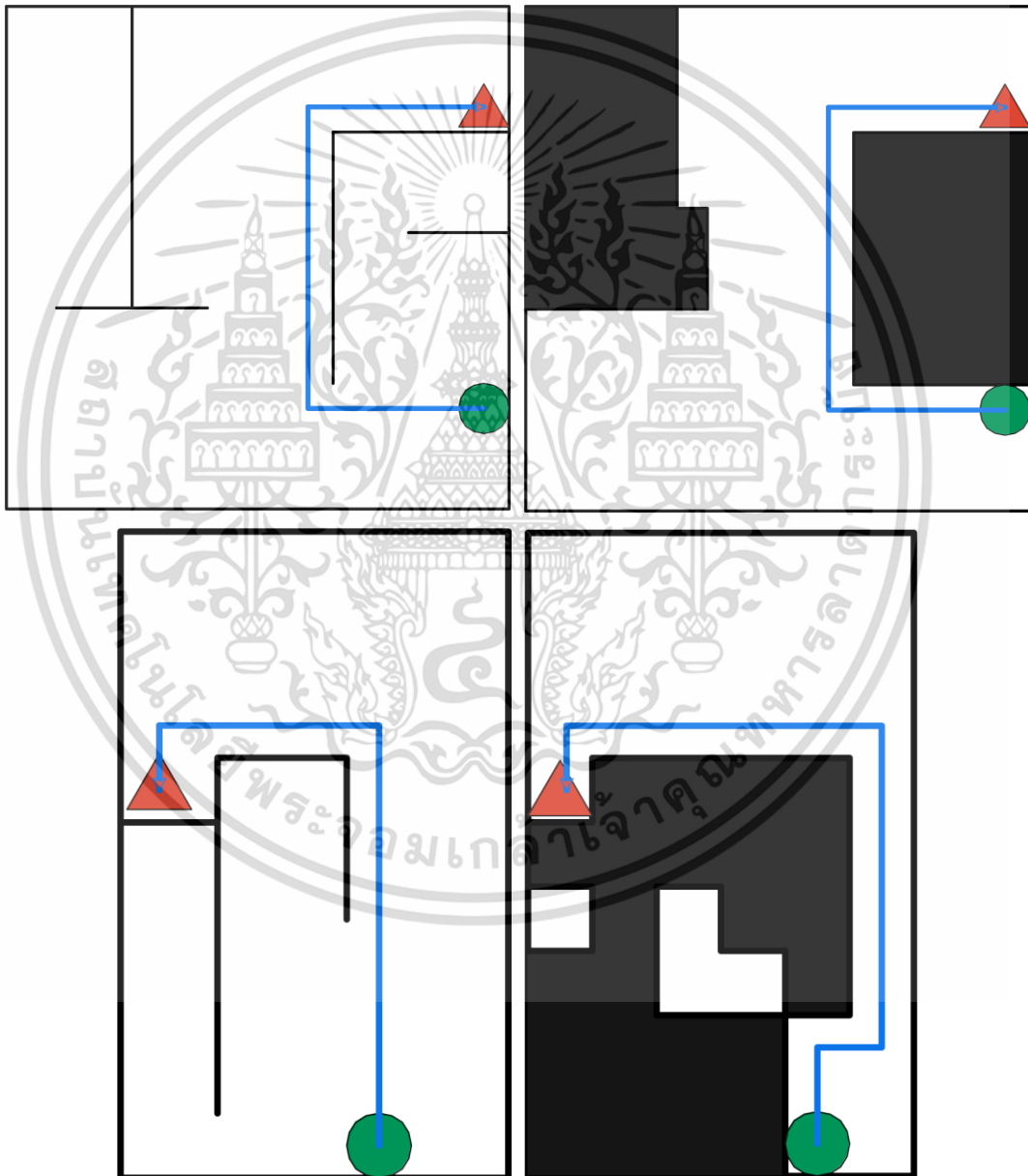
ตารางที่ 5.24 แสดงนโยบายในการแก้ปัญหาที่ดีที่สุด (Optimal Policy) ของแต่ละพิกัด (x, y) ในสนามทดสอบที่ 2 ขนาดกว้าง 6 เมตรและยาว 10 เมตร ด้วยวิธีการบอกตำแหน่งแบบเออาร์บีแอล

X/Y	0	1	2	3	4	5
0	D	D	L	L	L	L
1	D	D	L	L	L	L
2	D	L	L	L	L	L
3		-	-	-	-	U
4	-	--	-	-	-	U
5	X	-	X	-	-	U
6	-	-	X	X	-	U
7	-	-	-	-	R	U
8	-	-	-	-	U	
9	-	-	-	-	U	

จากตารางที่ 5.19 ถึง 5.21 ของวิธีการบอกตำแหน่งแบบทั่วไปและตารางที่ 5.22 ถึง 5.24 ของวิธีการบอกตำแหน่งแบบเออาร์บีแอลจะเห็นได้ว่าเมทริกซ์คิวและเมทริกซ์นโยบายแก้ปัญหา นั้นมีข้อมูลที่สอดคล้องกันโดยจะส่งผลให้แต่ละตำแหน่งมีการกระทำในทิศทางที่ชี้เข้าหาตำแหน่งเป้าหมายทั้งหมด โดยเส้นทางหลีกเลี่ยงสิ่งกีดขวางที่ดีที่สุดที่ระบบคำนวณได้ของสนามทดสอบทั้งสองสนาม นั้นจะแสดงดังรูปที่ 5.26 สัญลักษณ์วงกลมสีเขียวแทนตำแหน่งเริ่มต้นของหุ่นยนต์เคลื่อนที่ สามเหลี่ยมสีแดงแทนตำแหน่งเป้าหมาย พื้นที่สีดำแทนตำแหน่งสิ่งกีดขวางและกำแพงที่จำลองขึ้นมาในสนามทดสอบและเส้นทึบสีน้ำเงินจะแทนเส้นทางที่ดีที่สุดที่ระบบคำนวณได้ สำหรับการสร้างเส้นทางหลีกเลี่ยงสิ่งกีดขวางของสนามทดสอบที่ 1 จะเห็นได้ว่า การบอกตำแหน่งในแผนที่แบบปกติและแบบเออาร์บีแอลนั้นให้เส้นทางหลีกเลี่ยงแบบเดียวกัน ในขณะที่สนามทดสอบที่ 2 การบอกตำแหน่งแบบเออาร์บีแอลนั้นจะทำให้เส้นทางหลีกเลี่ยงอ้อมพื้นที่สิ่งกีดขวางมากกว่าการบอกตำแหน่งในแผนที่แบบปกติ ซึ่งอาจจะเป็นการเพิ่มระยะทางการเคลื่อนที่ให้แก่หุ่นยนต์แทนที่จะเป็นการหาระยะทางที่สั้นที่สุดไปยังเป้าหมาย อีกทั้งบางพิกัดตำแหน่งในสนามทดสอบจำลองนั้นไม่มีโอกาสได้ถูกสำรวจตัวอย่างเช่น ในตารางที่ 5.24 ที่พิกัด $(5, 0)$ และชุดพิกัด $(5, 2)$, $(6, 2)$, และ $(6, 3)$ ซึ่งเป็นพิกัดตำแหน่งภายในพื้นที่ที่ไม่สมมาตรหมายเลข 6, 10 และ 11 นั้นไม่มีการอัปเดตค่าคิวใด ๆ เลย สังเกตได้จากชุดพิกัดดังกล่าวจะถูกใส่เครื่องหมายกากบาทเอาไว้ สาเหตุมาจากพื้นที่ที่ไม่สมมาตรเหล่านั้น แม้จะไม่ถูกกำหนดให้มีสิ่งกีดขวางใด ๆ แต่มันก็ถูกล้อมรอบด้วยพื้นที่ที่ไม่สมมาตรที่เป็นพื้นที่สิ่งกีดขวางหมายเลขอื่น ๆ ทำให้หุ่นยนต์ไม่สามารถเข้าถึงตำแหน่งดังกล่าวได้ ซึ่งในกรณีนี้ หากตำแหน่งดังกล่าวเป็นเส้นทางที่นำไปสู่ตำแหน่งเป้าหมาย ก็เท่ากับว่าเป็นการตัดโอกาสให้ระบบค้นพบเส้นทางเคลื่อนที่ที่สั้นที่สุดของหุ่นยนต์ออกไป

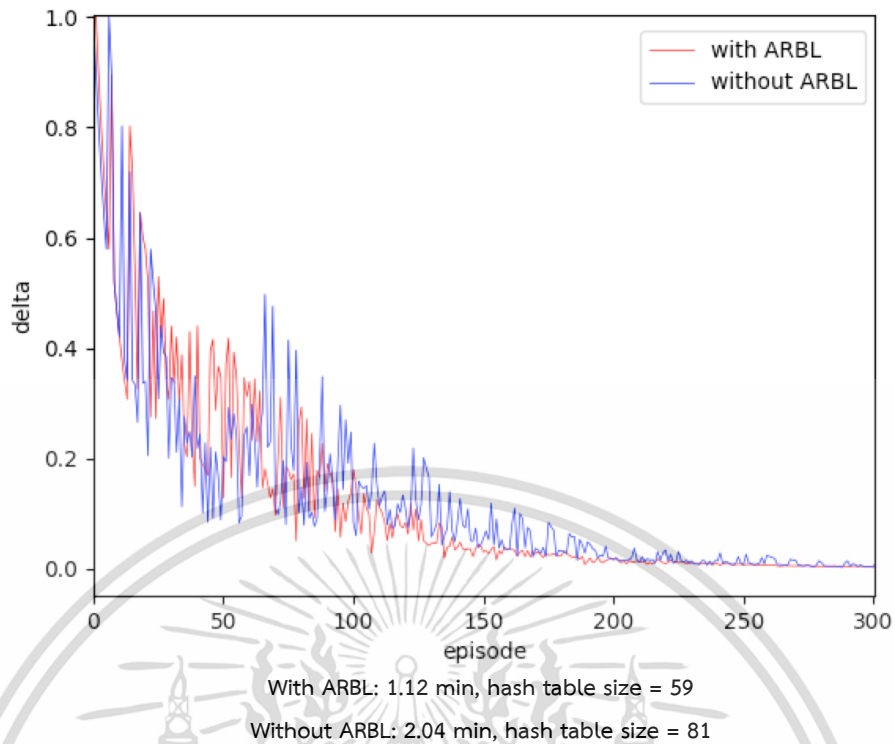
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตาม สนามทดสอบทั้งสองสนามนั้นได้มีการจำลองสิ่งกีดขวางให้มีลักษณะเป็นกับดักล้อมด้านหน้าเป็นลักษณะตัวยู (U-trap) หรือเป็นเขาวงกตที่มีทางตันอยู่ภายใน สถานการณ์นี้จะทำให้หุ่นยนต์เคลื่อนที่ที่เข้าไปสำรวจนั้นติดอยู่ภายในเขาวงกตและต้องหาเส้นทางหลบหลีกออกมา ซึ่งจะทำให้เสียเวลาในการสร้างเส้นทางใหม่ อีกทั้งยังต้องเพิ่มกลยุทธ์ในการเคลื่อนที่ของหุ่นยนต์เมื่อพบเจอสถานการณ์แบบนี้ในกรณีที่เขาวงกตมีความซับซ้อนมาก ดังนั้นจากตัวอย่างของสิ่งกีดขวางแบบกับดักลักษณะตัวยู หรือ ยูแทรป(U-trap) ที่จำลองขึ้นมาในทั้งสองสนาม จะแสดงให้เห็นว่าวิธีการบอกตำแหน่งในแผนที่แบบเออาร์บีแอล (ARBL) นั้นจะช่วยลดความซับซ้อนของสิ่งกีดขวางและป้องกันการเกิดเหตุการณ์ที่หุ่นยนต์จะติดกับดักยูแทรปขึ้นได้

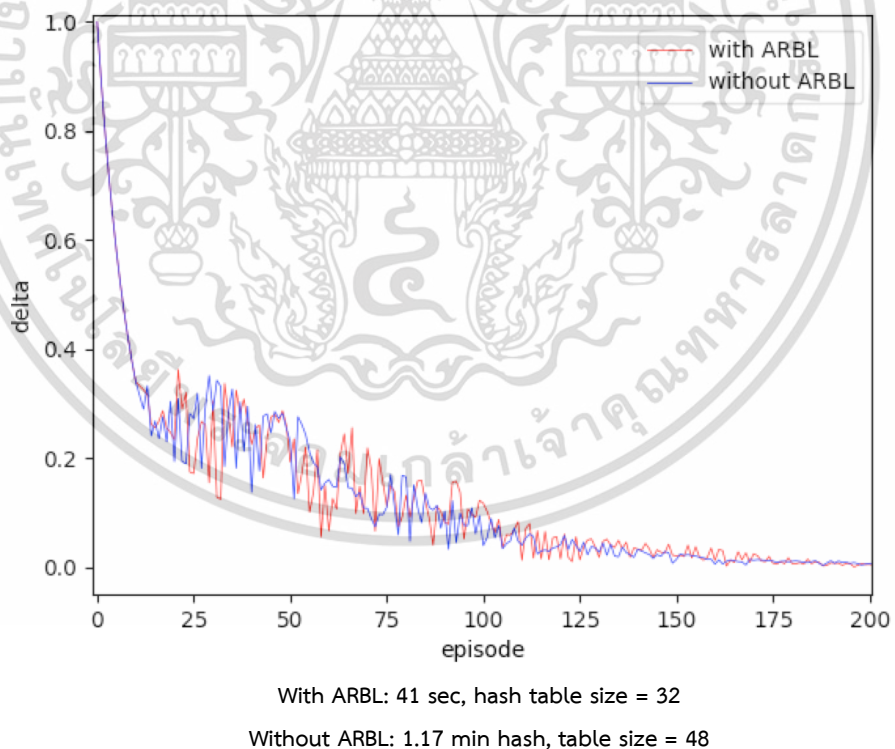


รูปที่ 5.26 ภาพจำลองการสร้างเส้นทางของระบบการเรียนรู้แบบคิวสำหรับหุ่นยนต์เคลื่อนที่ในสนามจำลอง เปรียบเทียบระหว่างการพิจารณาตำแหน่งของวัตถุในสนามตามปกติ (ซ้าย) และการพิจารณา

แบบวิธีเออาร์บีแอล (ขวา) ของทั้งสองสนามที่สอบที่ 1 (บน) และ 2 (ล่าง) ตามลำดับ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.27 กราฟเปรียบเทียบการเปลี่ยนแปลงความต่างของค่าคิวในแต่ละรอบการทำงาน ระหว่างวิธีพิจารณาสนามตามปกติ และวิธีพิจารณาสนามแบบเออาร์บีแอลของสนามทดสอบที่ 1



รูปที่ 5.28 กราฟเปรียบเทียบการเปลี่ยนแปลงความต่างของค่าคิวในแต่ละรอบการทำงาน ระหว่างวิธีพิจารณาสนามตามปกติ และวิธีพิจารณาสนามแบบเออาร์บีแอลของสนามทดสอบที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนสุดท้าย รูปที่ 5.27 และ 5.28 แสดงกระบวนการการเรียนรู้แบบคิวในลักษณะของกราฟเส้นของสนามทดสอบทั้ง 2 สนาม โดยในแต่ละกราฟนั้นเส้นสีแดงจะแทนกระบวนการเรียนรู้ และค้นหาเส้นทางหลักเฉียงสี่กึ่งตัวด้วยการบอกตำแหน่งในแผนที่แบบเออาร์บีแอล เส้นสีน้ำเงินจะแทนกระบวนการเรียนรู้ด้วยการบอกตำแหน่งในแผนที่แบบปกติทั่วไป กราฟในแนวแกน x แทนรอบการทำงานในการเรียนรู้ของระบบ โดยกำหนดไว้ให้มีรอบการทำงานสูงสุด 1000 รอบ และแนวแกน y จะแทนค่าความแตกต่างหรือเดลต้า (delta) ของค่าคิวสูงสุดและต่ำสุดของทุก ๆ การกระทำในสนาม ซึ่งจะเป็นตัวบ่งบอกถึงทิศทางของกระบวนการเรียนรู้ว่าจะเข้าสู่จุดอิมิตัว หรือจุดที่ระบบค้นหานโยบายแก้ปัญหาที่ดีที่สุดพบซึ่งจุดอิมิตัวคือจุดที่ให้ค่าเดลต้าเท่ากับ 0 จากรูปกราฟที่ 5.27 จะเห็นว่ากระบวนการเรียนรู้สำหรับการบอกตำแหน่งในแผนที่ด้วยวิธีปกตินั้นมีการเปลี่ยนแปลงของค่าเดลต้าแกว่งขึ้นลงค่อนข้างมาก เมื่อเทียบกับกระบวนการเรียนรู้สำหรับการบอกตำแหน่งแบบเออาร์บีแอลที่ยังเข้าใกล้จุดอิมิตัวได้ช้ากว่าวิธีเออาร์บีแอล โดยสังเกตได้จากกราฟสีแดงที่มีค่าเดลต้าเป็น 0 ณ รอบการทำงานที่ 200 เป็นต้นไป ระหว่างที่กราฟสีน้ำเงินยังคงมีการแกว่งของค่าเดลต้าอยู่เรื่อย ๆ แม้รอบการทำงานของระบบการเรียนรู้จะสูงกว่า 200 ไปแล้วก็ตาม ยิ่งไปกว่านั้นระหว่างการทำงาน ผู้ทำการทดลองได้มีการจับเวลาตั้งแต่ระบบเริ่มต้นกระบวนการเรียนรู้ไปจนถึงสิ้นสุดกระบวนการที่ 1000 รอบตามที่ได้อ้างไว้ อีกทั้งยังมีการบันทึกขนาดของหน่วยความจำที่ใช้ในการบันทึกข้อมูลต่าง ๆ ของการเรียนรู้เช่น ตำแหน่งต่าง ๆ ในสนาม ค่าคิวของตำแหน่งต่าง ๆ รวมไปถึงชุดกิริยาของแต่ละพิกัดในสนามที่จะนำไปสู่ตำแหน่งเป้าหมายได้ดีที่สุด ซึ่งผลลัพธ์ที่ได้คือระยะเวลาที่ใช้ตั้งแต่เริ่มกระบวนการฝึกสอนเปรียบเทียบระหว่างวิธีที่นำเสนอและวิธีปกตินั้น ระบบใช้เวลาไป 1.12 นาที และ 2.04 นาทีตามลำดับ ในส่วนของหน่วยความจำสำหรับบันทึกข้อมูลในแต่ละพิกัดของสนามจะเห็นได้ว่า วิธีเออาร์บีแอลช่วยลดปริมาณหน่วยความจำแฮชแมป (HashMap) ที่ใช้อยู่ที่ 59 ไบต์ ในขณะที่วิธีปกตินั้นจะใช้หน่วยความจำแฮชแมปไป 81 ไบต์

สำหรับสนามทดสอบจำลองสนามที่ 2 กระบวนการเรียนรู้ทั้งแบบปกติและแบบเออาร์บีแอลนั้นให้แนวโน้มการเปลี่ยนแปลงที่ค่อนข้างใกล้เคียงกัน และเข้าใกล้จุดอิมิตัวเท่าเทียมกัน ณ รอบการทำงานที่ 200 แต่เมื่อพิจารณาถึงระยะเวลาที่ใช้ในการประมวลผลและพื้นที่หน่วยเก็บความจำนั้น วิธีเออาร์บีแอลใช้เวลาและหน่วยเก็บความจำ 41 วินาที กับขนาดหน่วยความจำแฮชแมป 32 ไบต์ ในขณะที่ วิธีปกติใช้เวลาและหน่วยเก็บความจำที่ 1.17 นาที และ 48 ไบต์ตามลำดับ

5.6 ผลการเปรียบเทียบระหว่างการบอกตำแหน่งด้วยวิธี ARBL (Asymmetry Regional Boundary Localization) กับการเทคนิค SLAM (Simultaneous Localization and Mapping) ที่ทำงานบนระบบปฏิบัติการหุ่นยนต์ ROS (Robot Operating System)

จากผลการทดลองการเปรียบเทียบความสามารถในการใช้งานจริงของเทคนิค สแลมและเออาร์บีแอลกับพื้นที่ปิดภายในอาคาร เราสามารถสรุปและแบ่งประเด็นออกได้เป็น 5 หัวข้อดังต่อไปนี้

1. โครงสร้างของระบบ
2. ความยืดหยุ่นและความซับซ้อนของวิธีการ
3. ประสิทธิภาพในการบอกตำแหน่ง
4. ความเหมาะสมต่อการนำไปใช้
5. ต้นทุนในการผลิต

5.6.1 โครงสร้างของระบบ

การบอกตำแหน่งเป้าหมายในอาคารด้วยวิธีเออาร์บีแอลและวิธีสแลมบนรอสแพลตฟอร์มนั้นมีโครงสร้างของระบบการจัดการที่คล้ายคลึงกัน หากแต่ภายหลังจากการทดสอบนั้น พบว่าโครงสร้างของแต่ละระบบต่างมีข้อดีข้อเสียที่ต่างกันไป ตัวอย่างเช่น ระบบการบอกตำแหน่งแบบสแลมบนรอสแพลตฟอร์มนั้นจำเป็นต้องมีส่วนกลางควบคุมการทำงาน มิเช่นนั้นแล้วระบบทั้งหมดจะหยุดการทำงาน ในขณะที่เออาร์บีแอลถูกออกแบบมาให้แต่ละส่วนทำงานแยกออกจากกันได้อย่างอิสระ เป็นต้น ซึ่งส่วนประกอบของโครงสร้างในแต่ละระบบสามารถอธิบายได้ดังตารางต่อไปนี้

ตารางที่ 5.25 ตารางเปรียบเทียบโครงสร้างของระบบการบอกตำแหน่งหุ่นยนต์ภายในอาคารด้วยวิธีเออาร์บีแอลและวิธีสแลมบนระบบปฏิบัติการรอส

	Asymmetry Regional Boundary Localization	Hector SLAM with ROS
ระบบส่วนกลางจัดการข้อมูล	ฐานข้อมูลคลาวด์ไฟร์เออร์เบส (Firebase cloud database)	รอสคอร์ (ROS core)
ไคลแอนต์	หุ่นยนต์เคลื่อนที่ โทรศัพท์มือถือ สถานีฐาน	หุ่นยนต์เคลื่อนที่ สถานีฐาน
รูปแบบเครือข่าย	เครือข่ายทั่วไป (Global)	เครือข่ายภายในท้องถิ่น (Local)
ลักษณะของการสื่อสาร	อะซิงโครนัส (Asynchronous)	ซิงโครนัส (Synchronous)
การทำงานของระบบ	ทำงานแยกส่วนกันอย่างอิสระ	จำเป็นต้องมีส่วนกลางควบคุม

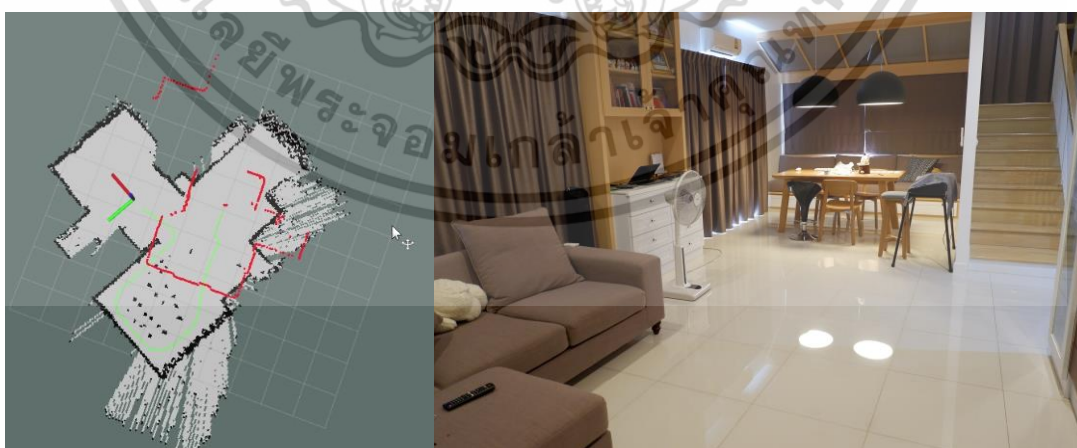
เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สงวนเพื่อการศึกษาเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.2 ความยืดหยุ่นและความซับซ้อนของวิธีการ

การบอกตำแหน่งเป้าหมายในอาคารด้วยวิธีสแลมนั้นสามารถให้ความยืดหยุ่นที่มากกว่า เนื่องจากตัวอัลกอริทึมสามารถใช้เซนเซอร์ตรวจจับสภาพแวดล้อม สร้างแผนที่จำลองโดยรอบไปพร้อมกับการบอกตำแหน่ง ณ ปัจจุบันของเป้าหมายโดยอ้างอิงจากแผนที่ที่ตรวจรู้ได้ จึงทำให้การประมวลผลข้อมูลมีการตอบสนองที่ค่อนข้างไวในระดับเรียลไทม์ ทั้งยังสามารถประยุกต์ใช้กับบริเวณใด ๆ ก็ได้ ไม่จำกัดบริเวณและไม่มีอุปสรรคในการจัดเตรียมสถานที่ ในขณะที่วิธีเออาร์บีแอลนั้นจำเป็นต้องอาศัยขั้นตอนการเตรียมสถานที่ด้วยระยะเวลาค่อนข้างนาน ทั้งในด้านการเก็บบันทึกข้อมูลตัวอย่างสัญญาณ การสร้างโมเดลและแบ่งขอบเขตในการบอกตำแหน่งพื้นที่ที่ไม่สมมาตรในสนาม รวมไปถึงพื้นที่ที่ทำการทดสอบจำเป็นต้องมีอุปกรณ์สื่อสารไร้สายติดตั้งอยู่ด้วยจึงจะสามารถประยุกต์ใช้วิธีนี้ในการบอกตำแหน่งได้ และด้วยเงื่อนไขเหล่านี้จึงทำให้ความยืดหยุ่นในการใช้งานจริงของวิธีเออาร์บีแอลน้อยกว่า

อย่างไรก็ตามในด้านของความซับซ้อนของวิธีการ เทคนิคเออาร์บีแอลที่นำเสนอมีข้อได้เปรียบในด้านการประมวลผลซึ่งอาศัยเพียงเทคนิคการหาความสามารถในการซ้ำค่าของอุปกรณ์และการจำแนกประเภทข้อมูลเป็นหลัก สามารถจัดการได้โดยวิธีการใช้แมชชีนเลิร์นนิงประเภทมีผู้ฝึกสอน (Supervised Machine Learning) เข้ามาทำการจำแนก (Classification) ชุดข้อมูลความแรงสัญญาณที่ได้เก็บบันทึกตัวอย่างไว้แล้วก่อนหน้านี้ในขั้นตอนการเตรียมสถานที่ ทำให้ไม่จำเป็นต้องประมวลผลด้วยอัลกอริทึมที่ซับซ้อนหลาย ๆ ขั้นตอน ขณะที่สแลมนั้นจำเป็นที่จะต้องหาความสัมพันธ์ระหว่างข้อมูลที่ได้จากการตรวจรู้สภาพแวดล้อม และสถานะปัจจุบันของหุ่นยนต์ พร้อมกับอัปเดตผลให้เป็นปัจจุบันสำหรับการคำนวณในรอบการทำงานต่อ ๆ ไป ซึ่งกระบวนการที่ได้กล่าวมาโดยสังเขปนี้ อาศัยอัลกอริทึมทางคณิตศาสตร์หลายประเภทเข้ามาจัดการ เช่น การคัดกรองลักษณะเด่นของข้อมูล สิ่งแวดล้อม (Feature Extraction), ตัวกรองคาลมานแบบต่อยอด (Extended Kalman Filter)



รูปที่ 5.29 วิธีการบอกตำแหน่งภายในสภาพแวดล้อมด้วยสแลมบนรอสแพลตฟอร์ม (ซ้าย) ผลลัพธ์จากการสแกนพื้นที่เป้าหมายและแสดงในโปรแกรมอาร์วิซ (RVIZ) ด้วยข้อมูลประเภทพอยท์คลาวด์ (ขวา) พื้นที่ปิดภายในอาคารที่หุ่นยนต์เคลื่อนที่ทำการสแกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.3 ประสิทธิภาพในการบอกตำแหน่ง

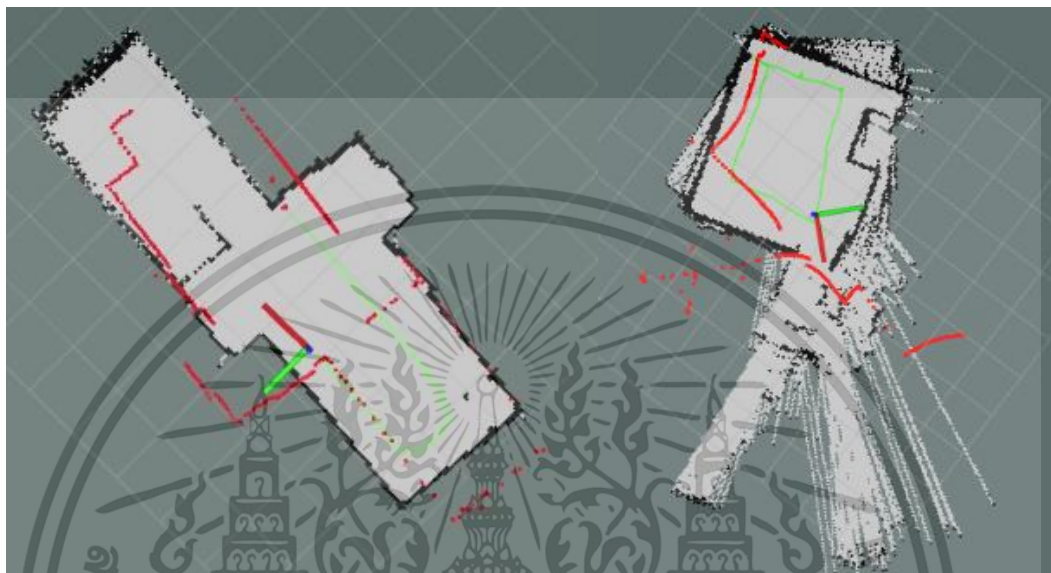
การบอกตำแหน่งด้วยวิธีสแลมนั้นมีลักษณะของข้อมูลเป็นพอยท์คลาวด์ความละเอียดสูง ซึ่งสามารถแปลงเป็นพิกัดในระนาบสองมิติได้อย่างแม่นยำ อีกทั้งตัวเลเซอร์สแกนที่ทำการตรวจรู้สภาพแวดล้อมโดยรอบนั้นมีอัตราการอ่านข้อมูลความถี่สูง ทำให้ระบบสามารถตรวจจับสิ่งต่าง ๆ และจำลองแผนที่ขึ้นมาได้อย่างเรียลไทม์ รูปที่ 5.30 แสดงผลลัพธ์จากการสแกนพื้นที่เป้าหมายของหุ่นยนต์เคลื่อนที่ด้วยวิธีการบอกตำแหน่งแบบสแลมบนรถสแลมแพลตฟอร์ม จากภาพทั้งสองจะเห็นว่าเส้นกรอบสีแดง คือข้อมูลพอยท์คลาวด์แทนจุดที่ลำแสงเลเซอร์สแกนตกกระทบกับวัตถุแบบเรียลไทม์ ด้วยอัตราความถี่สูง ในขณะที่เส้นกรอบสีดำคือข้อมูลพอยท์คลาวด์แทนตำแหน่งที่ผ่านกระบวนการสแลม และถูกระบบจดจำไว้แล้วว่าเป็นตำแหน่งที่มีวัตถุในสภาพแวดล้อมจริง ส่วนตำแหน่งของหุ่นยนต์เมื่ออ้างอิงกับสภาพแวดล้อมเป้าหมายนั้นจะแทนด้วยเส้นตั้งฉากสีเขียวกับสีแดง โดยเส้นสีแดงจะแทนทิศทางที่หุ่นยนต์หันหน้าไป ส่วนเส้นสีเขียวจะแทนทิศทางซ้ายมือของตัวหุ่นยนต์

แต่ขณะเดียวกัน ตัวเลเซอร์สแกนเนอร์มีข้อเสียในการสแกนวัตถุที่มีผลต่อการเดินทางของลำแสง ข้อมูลเซนเซอร์ทั้งหมดที่อ่านได้คือสิ่งที่ลำแสงตกกระทบในแนวระนาบเดียวกันกับตัวเซนเซอร์ ถ้าหากวัตถุเป้าหมายมีสภาพพื้นผิวที่ดูดซับแสงหรือหักเหแสง อันเป็นเหตุให้การสะท้อนของลำแสงเลเซอร์เข้าสู่ตัวรับนั้นไม่ถูกต้อง ก็จะทำให้จุดปลายรัศมีการสแกน (scan endpoint) ที่ระบบคำนวณได้คลาดเคลื่อนไปจากความเป็นจริงและจะส่งผลกระทบต่อโครงสร้างแผนที่ ยิ่งไปกว่านั้นวิธีสแลมโดยทั่วไปแล้วต้องการข้อมูลจากเซนเซอร์อื่น ๆ เข้ามาอ้างอิงเพื่อเพิ่มความถูกต้องแม่นยำของการสร้างแผนที่และการคำนวณตำแหน่งเป้าหมาย เช่น เซนเซอร์วัดตรวจจับการเคลื่อนที่ (odometry sensor), เซนเซอร์วัดมุมองศา (orientation sensor) เป็นต้น เทคนิคสแลมที่นำมาใช้ทดสอบการเปรียบเทียบประสิทธิภาพนั้น ใช้เลเซอร์สแกนเนอร์เป็นอินพุตในการประมวลผลเพียงอย่างเดียว และใช้อัลกอริทึมสแกนแมทซึ่งมาประมวลผลหาความถูกต้องของการจัดเรียงตัว (alignment) ระหว่างลำแสงเลเซอร์สแกนและแผนที่ที่ตรวจรู้ได้ ดังนั้นหากหุ่นยนต์เคลื่อนที่หรือเปลี่ยนมุมมองเร็วเกินไปเป็นเหตุให้การหาความสัมพันธ์ของการจัดเรียงตัวไม่ทันกับการเปลี่ยนแปลง จึงทำให้การสแกนแผนที่รวมไปถึงการอ้างอิงถึงตำแหน่งปัจจุบันของหุ่นยนต์คลาดเคลื่อนไปด้วย

อย่างไรก็ตาม สำหรับวิธีเออาร์บีแอลผลลัพธ์จากอัลกอริทึมนี้มีรูปแบบของข้อมูลเป็นลักษณะไม่ต่อเนื่อง (discrete) ซึ่งเป็นการระบุตำแหน่งในแผนที่แบบขอบเขตพื้นที่ที่ถูกกำหนดหมายเลขไว้แล้วล่วงหน้า ไม่ได้ระบุพิกัดเป้าหมายอย่างเฉพาะเจาะจง เนื่องจากอัลกอริทึมที่ใช้เป็นแมชชีนเลิร์นนิงแบบมีผู้ฝึกสอนประเภทโมเดลจำแนกข้อมูล ที่จะทำให้หน้าที่ในการทำนายผลลัพธ์ความแม่นยำได้ของข้อมูลว่าเป้าหมายควรจะอยู่ในพื้นที่ใดเมื่ออ้างอิงจากชุดความแรงสัญญาณขณะนั้น โมเดลจะให้คำตอบพื้นที่ที่มีความน่าจะเป็นสูงที่สุด จากลักษณะของข้อมูลผลลัพธ์นี้เองที่ไม่ได้เป็นเชิงปริมาณแบบต่อเนื่อง (continuous) ทำให้เราสามารถลดความคลาดเคลื่อนในการบอกตำแหน่งลงได้มาก รวมไปถึงการใช้อินฟราเรดเซนเซอร์ระยะใกล้ในการระบุตำแหน่งเป้าหมายหรือสิ่งกีดขวาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยรอบว่าอยู่ในระยะตรวจจับของหุ่นยนต์หรือไม่ โดยเอาที่พู่ทของอุปกรณ์ก็เป็นแบบสัญญาณไม่ต่อเนื่องเช่นกัน ทำให้หุ่นยนต์สามารถระบุพื้นที่ที่มีสิ่งกีดขวางตั้งอยู่ได้ โดยไม่จำเป็นต้องอาศัยอัลกอริทึมที่ซับซ้อนเข้ามาจัดการ สถานะของหุ่นยนต์ ความสัมพันธ์ของค่าที่ได้จากเซนเซอร์กับตำแหน่งที่หุ่นยนต์เปลี่ยนแปลงไป และการอัปเดตแผนที่การจับของพื้นที่กริด (occupancy grid map) ของแผนที่ที่ตรวจรู้ได้



รูปที่ 5.30. แสดงผลลัพธ์การสแกนพื้นที่ทดสอบบนโปรแกรมอาร์วิสด้วยวิธีการบอกตำแหน่งแบบสแลมบนรอสแพลทฟอร์ม

5.6.4 ความเหมาะสมต่อการนำไปใช้

จากผลการทดสอบ วิธีการบอกตำแหน่งภายในอาคารแบบสแลมเหมาะสำหรับสภาพแวดล้อมภายในอาคารทั่วไปที่มีขนาดเล็กแคบและซับซ้อน มีมุมห้องและเสาต่าง ๆ เป็นจำนวนมาก มีวัตถุต่าง ๆ วางตัวกันอย่างกระจุกกระจาย เช่น อาคารสำนักงานทั่วไป ห้างสรรพสินค้า ภายในบ้านเรือน เป็นต้น เนื่องจากการทำงานของอัลกอริทึมที่สามารถอัปเดตข้อมูลที่ตรวจรู้ได้ ค้นหาพีเจอร์ที่สำคัญของสถานที่เพื่อนำไปเป็นแลนด์มาร์ก (Landmark) ในการอ้างอิงตำแหน่ง พร้อมทั้งหาความสัมพันธ์ระหว่างตัวหุ่นยนต์และข้อมูลเซนเซอร์เพื่อนำมาใช้สร้างโมเดลจำลองสิ่งแวดล้อม และอ้างอิงตำแหน่งปัจจุบันของตัวเองได้แบบเรียลไทม์จึงทำให้วิธีนี้สามารถรับมือต่อการเปลี่ยนแปลงของสิ่งแวดล้อมได้ที่มีอยู่ตลอดเวลาได้

ขณะที่วิธีบอกตำแหน่งภายในอาคารแบบเออาร์บีแอลนั้นเหมาะสำหรับสภาพแวดล้อมภายในโรงงานหรืออาคารเก็บสินค้า เป็นหลัก เนื่องจากเป็นสถานที่ที่มีบริเวณกว้าง มีวัตถุหรือสิ่งของวางอยู่ตายตัว เช่น ชั้นวางสินค้า หรือเครื่องจักรในสายการผลิต เป็นต้น และไม่มี การเปลี่ยนแปลงของตำแหน่งวัตถุอย่างฉับพลันเป็นจำนวนมาก ยิ่งไปกว่านั้น ในปัจจุบัน โกดังเก็บสินค้าหรือแพลนท์ในเขตอุตสาหกรรมต่าง ๆ มีการติดตั้งอุปกรณ์สื่อสารไร้สายขึ้นมาเพื่อรับส่งข้อมูลกันระหว่างภาคส่วนสามารถสั่งการควบคุมและมอนิเตอร์ความเปลี่ยนแปลงของกระบวนการต่าง ๆ ได้อย่างอิสระ ซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครือข่ายสื่อสารไร้สายนี้เอง จะเป็นประโยชน์ต่อวิธีเออาร์บีแอลอย่างมากเนื่องจากสามารถใช้ค่าความแรงสัญญาณมาเป็นตัวบอกตำแหน่งให้กับเป้าหมายได้อย่างดี โดยไม่จำเป็นต้องติดตั้งอุปกรณ์เพิ่มเติม

5.6.5 ต้นทุนในการผลิต

จากการทดลองสามารถสรุปได้ว่าการบอกตำแหน่งด้วยวิธีเออาร์บีแอลนั้นใช้ต้นทุนและทรัพยากรในการผลิตหรือสร้างระบบการทำงานโดยรวมทั้งหมดที่ต่ำกว่า เมื่อเปรียบเทียบกับวิธีการบอกตำแหน่งด้วยสแลมบนรอสแพลตฟอร์ม ซึ่งรายละเอียดของวิธีเออาร์บีแอลนั้นสามารถสรุปเนื้อหาโดยสังเขปได้ดังต่อไปนี้

1. วิธีเออาร์บีแอลใช้ต้นทุนต่ำกว่า เพราะอัลกอริทึมที่ออกแบบมาให้สามารถประมวลผลได้ทั้งบน เซิร์ฟเวอร์และบนตัวไคลแอนท์(หุ่นยนต์)เอง ซึ่งจะเป็นแพลตฟอร์มใด ๆ ก็ได้ไม่มีข้อจำกัด
2. โมเดลในการจำแนกขอบเขตพื้นที่ที่ไม่สมมาตรนั้นจำเป็นต้องสร้าง และฝึกฝนบนเซิร์ฟเวอร์ที่มีความเร็วในการประมวลผลมากเพียงพอ แต่เมื่อฝึกฝนเสร็จสิ้นก็จะสามารถนำโมเดลไปประยุกต์ใช้กับหน่วยประมวลผลใด ๆ ก็ได้และทำการรันระบบเพื่อทำนายผลการคาดการณ์ จึงไม่จำเป็นต้องติดตั้งและรันโมเดลบนพีซีเซิร์ฟเวอร์เสมอไป ลดการใช้ทรัพยากรประมวลผลขนาดใหญ่
3. หุ่นยนต์ไม่จำเป็นต้องมีความสามารถในการประมวลผลสูง แค่ใช้ตัวไมโครคอนโทรลเลอร์ความถี่ต่ำ ที่สามารถสั่งการทำงานอุปกรณ์แอกชูเอเตอร์แบบดิจิทัล/อนาล็อกใด ๆ และมีอินเตอร์เฟสสำหรับรับส่งข้อมูลไร้สาย เพื่อรับคำสั่งจากภายนอกสำหรับกรณีที่มีศูนย์กลางการประมวลผลแยกออกจากตัวหุ่นยนต์ และควบคุมกระบวนการทำงานได้ก็เพียงพอ
4. การตรวจจับสิ่งแวดล้อมจะอาศัยการผสมผสานข้อมูลจากอุปกรณ์ระหว่าง เครือข่ายสื่อสารไร้สายและอุปกรณ์เซนเซอร์ตรวจจับสิ่งกีดขวางระยะใกล้ ซึ่งในส่วนของเครือข่ายไร้สายนั้น เราสามารถประยุกต์ใช้กับอุปกรณ์ที่มีติดตั้งในสถานที่อยู่แล้วไม่จำเป็นต้องทำการติดตั้งเพิ่มเติม เนื่องจากในงานวิจัยนี้ได้ทำการทดสอบกับเครือข่ายไร้สายแบบ 2.4GHz ซึ่งเป็นมาตรฐานเดียวกันกับอุปกรณ์ไร้สาย (WiFi) ตามอาคารทั่วไป อีกทั้งอุปกรณ์เซนเซอร์ตรวจจับสิ่งกีดขวางนั้นไม่จำเป็นต้องมีความละเอียดสูงในการตรวจจับ จึงทำให้สามารถใช้อุปกรณ์ที่มีราคาถูกและหาได้ตามท้องตลาดทั่วไปได้

ในขณะเดียวกัน เทคนิคแฮคเตอร์สแลมที่นำมาใช้ทดสอบนี้ออกแบบมาเพื่อประยุกต์ใช้กับเฟรมเวิร์กสำหรับควบคุมหุ่นยนต์รอส ซึ่งจะมีข้อจำกัดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. รอสต้องประมวลผลบนแพลตฟอร์มที่สามารถรันระบบปฏิบัติการลินุกซ์ (Linux) ได้เท่านั้น เช่น เครื่องพีซี, แลปท็อป, หรือซิงเกิลบอร์ดคอมพิวเตอร์ เป็นต้น ซึ่งทำให้ต้นทุนในส่วนของอุปกรณ์ควบคุมนั้นมีราคาสูง
2. รอสสามารถทำงานได้ทั้งแบบภายในตัวเอง (standalone) และแบบลูกข่ายที่มีการควบคุมจากแม่ข่าย (Client-Master) ได้ โดยมีเงื่อนไขว่า การสื่อสารภายในเครือข่ายนั้นจะอ้างอิงที่อยู่จากไอพีแอดเดรสเท่านั้น ดังนั้นเทคโนโลยีวายฟายจึงเป็นการสื่อสารหลักสำหรับระบบปฏิบัติการ อีกทั้งในการทำแม่ปั๊งดด้วยวิธีสแลมระหว่างลูกข่ายแม่ข่ายโดยให้ตัวหุ่นยนต์ทำหน้าที่เป็นมาสเตอร์ประมวลผลข้อมูลดิบพร้อมกับส่งค่าเลเซอร์สแกนเนอร์ที่ตรวจจับสิ่งแวดล้อมได้ไปยังไคลเอนท์ เพื่อนำข้อมูลที่ได้มาสร้างแผนที่และแสดงผลบนโปรแกรมอาร์วิส กระบวนการนี้จำเป็นต้องอาศัยช่องสัญญาณหรือแบนด์วิธ (bandwidth) ขนาดใหญ่เพื่อรองรับการส่งข้อมูลจำนวนมากในเวลาอันสั้น เพื่อเข้ากระบวนการสแลมได้อย่างถูกต้องและเรียลไทม์มากที่สุด มิเช่นนั้น ความสัมพันธ์ระหว่างแผนที่ที่สร้างขึ้นกับสถานะของหุ่นยนต์ที่เคลื่อนที่ไปในแผนที่นั้นจะไม่สอดคล้องกัน
3. เทคนิคสแลมนั้นจำเป็นต้องอาศัยการทำงานของเซนเซอร์หลากหลายชนิด ซึ่งวิธีที่นำมาใช้ทดสอบนี้ จะใช้เลเซอร์สแกนเนอร์รุ่น RPLiDAR A1M8 ของบริษัท SLAMTEC เพียงอย่างเดียว ซึ่งมีความละเอียดสูงทั้งในด้าน ความถี่ในการสแกน (ส่งผลต่อการใช้อัตราการรับส่งข้อมูล (baudrate) ที่สูงมากขึ้นไปด้วย) จำนวนจุดพอยท์คลาวด์ที่สามารถตรวจจับได้ต่อหนึ่งรอบการทำงาน และค่าความคลาดเคลื่อนของความเร็วการหมุนเชิงมุมของตัวอุปกรณ์ที่น้อยมาก ส่งผลให้ต้นทุนของอุปกรณ์เซนเซอร์สิ่งแวดล้อมตัวนี้มีราคาที่สูงมากขึ้นไปด้วย

ด้วยเหตุผลที่กล่าวมานั้น ทำให้วิธีเออาร์บีแอลใช้ทรัพยากรในการประมวลผล รวมไปถึงต้นทุนในการสร้างหุ่นยนต์เคลื่อนที่ภายในอาคารที่ต่ำกว่ามาก

บทที่ 6

สรุปผลการวิจัย และข้อเสนอแนะ

6.1 สรุปผลการวิจัย

วิทยานิพนธ์ฉบับนี้ได้นำเสนอวิธีการบอกตำแหน่งเป้าหมายด้วยการแบ่งพื้นที่ที่ไม่สมมาตรสำหรับการนำทางหุ่นยนต์ภายในอาคาร โดยจะใช้ความแรงสัญญาณจากอุปกรณ์สื่อสารไร้สายมาเป็นตัวบ่งบอกพื้นที่ที่ไม่สมมาตรที่ได้กำหนดไว้ในสนามหรือพื้นที่ทดสอบ ซึ่งเกณฑ์การแบ่งพื้นที่นั้นจะใช้ความสามารถในการซ้ค่าของอุปกรณ์สื่อสารไร้สายที่ติดตั้งภายในพื้นที่ทดสอบ มาเป็นตัวกำหนดขนาดขอบเขตของพื้นที่ ตำแหน่งของพื้นที่ที่ไม่สมมาตรจะถูกกำหนดเป็นหมายเลขต่าง ๆ เพื่อใช้ในการจำแนกข้อมูลด้วยโครงข่ายประสาทเทียม ซึ่งจะถูกรู้สร้างและฝึกสอนด้วยชุดข้อมูลความแรงสัญญาณระหว่างเสาสัญญาณอ้างอิงในสนามและเสาสัญญาณเป้าหมายบนตัวหุ่นยนต์เคลื่อนที่ที่มีการเก็บบันทึกผลทั่วทั้งสนามทั้งหมด 50 ครั้ง เมื่อสร้างและฝึกสอนโครงข่ายประสาทเทียมจนเหมาะสมกับแต่ละสภาพแวดล้อมในพื้นที่ทดสอบ หุ่นยนต์เคลื่อนที่ในสนามจะอ่านค่าความแรงสัญญาณปัจจุบันและส่งให้กับโมเดลโครงข่ายประสาทเทียม เพื่อจำแนกพื้นที่ที่ไม่สมมาตรที่ตัวเองอยู่ ทั้งยังสามารถตรวจจับได้อีกว่าพื้นที่ปัจจุบันมีสิ่งกีดขวางใด ๆ หรือไม่ จากอุปกรณ์อินฟราเรดเซนเซอร์ตรวจจับวัตถุระยะสั้น โดยพื้นที่ที่มีสิ่งกีดขวางนั้นจะถูกบันทึกข้อมูลให้เป็นพื้นที่ห้ามเข้า หุ่นยนต์จึงต้องทำการสร้างเส้นทางหลีกเลี่ยงโดยการใช้อัลกอริทึมการเรียนรู้แบบเสริมกำลังด้วยเทคนิคการเรียนรู้แบบควิซิงจากผลการทดลองนั้นแสดงให้เห็นว่าเออาร์บีแอลนั้นเป็นวิธีในการบอกตำแหน่งที่ออกแบบง่ายและทนทานต่อความคลาดเคลื่อนในการบอกตำแหน่ง ซึ่งเกิดขึ้นได้โดยทั่วไปกับอัลกอริทึมบอกตำแหน่งเป้าหมายในอาคารด้วยความแรงสัญญาณ

โดยวิธีการบอกตำแหน่งแบบเออาร์บีแอลนี้ให้ความถูกต้องในการบอกตำแหน่งของสนามทดสอบทั้ง 2 สนามอยู่ที่ 71.9 % และ 85.5 % ตามลำดับ อีกทั้งวิธีที่นำเสนอนี้ช่วยลดความซับซ้อนของการตรวจจับสิ่งกีดขวางในแผนที่หรือสนามทดสอบลงได้ ช่วยให้หุ่นยนต์ค้นหาเส้นทางได้ง่ายและรวดเร็วมากขึ้น รวมไปถึงยังช่วยลดโอกาสที่หุ่นยนต์จะติดอยู่ในบริเวณสิ่งกีดขวาง เช่น เขาวงกต เป็นต้น ในด้านของต้นทุนในการผลิต เมื่อเปรียบเทียบกับเทคนิคการบอกตำแหน่งเป้าหมายในอาคารและการสร้างแผนที่อย่างสแลมนั้น วิธีเออาร์บีแอลใช้ต้นทุนและทรัพยากรในการประมวลผลที่น้อยกว่าขณะที่ยังสามารถรักษาประสิทธิภาพในการบอกตำแหน่งและสร้างเส้นทางหลบหลีกสิ่งกีดขวางต่าง ๆ เพื่อไปยังตำแหน่งเป้าหมายได้เหมือนกัน โดยกระบวนการเรียนรู้แบบเสริมกำลังที่ใช้ค้นหาเส้นทางไปยังเป้าหมายที่ดีที่สุดของสนามทดสอบทั้ง 2 สนามนั้น ใช้เวลาและหน่วยเก็บข้อมูลสำหรับประมวลผลไปทั้งสิ้น 1.12 นาที / 59 ไบต์ และ 41 วินาที / 32 ไบต์ ของสนามทดสอบที่ 1 และ 2 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตาม วิธีบอกตำแหน่งแบบเออาร์บีแอลนี้อาจจะมีข้อเสียในด้านของความยืดหยุ่นในการใช้งาน เนื่องจากวิธีนี้มีรากฐานมาจากเทคนิคการรู้จำลายนิ้วมือ (Fingerprinting technique) โดยอาศัยการบันทึกข้อมูลความแรงสัญญาณ ณ ตำแหน่งต่าง ๆ ในสนามหลาย ๆ รอบ เพื่อมาทำการประมวลผล อีกทั้งยังใช้โมเดลโครงข่ายประสาทเทียมเข้ามาจำแนกหมวดหมู่ของข้อมูล ดังนั้นหากเกิดการเปลี่ยนแปลงที่ส่งผลกระทบต่อสัญญาณในสนามทดสอบ ก็จะทำให้ประสิทธิภาพในการบอกตำแหน่งลดน้อยลง และที่สำคัญ สถานที่ที่จะนำวิธีนี้ไปประยุกต์ใช้จำเป็นต้องมีอุปกรณ์ส่งสัญญาณไร้สายประจำอยู่ ณ จุดต่าง ๆ ของพื้นที่ และมีความซับซ้อนของสถานที่ไม่สูงมากจนเกินไป เนื่องจากถ้ามีสิ่งกีดขวางอยู่ในบริเวณเป็นจำนวนมาก การบอกตำแหน่งด้วยวิธีนี้อาจส่งผลให้หุ่นยนต์ไม่สามารถค้นหาเส้นทางที่หลีกเลี่ยงสิ่งกีดขวางได้

6.2 ข้อเสนอแนะ

เพื่อประสิทธิภาพในการบอกตำแหน่งในอาคารด้วยวิธีเออาร์บีแอลที่ดีขึ้นสำหรับการพัฒนาต่อยอดแนวคิดต่อไปในอนาคต จึงได้มีข้อเสนอแนะและข้อปรับปรุงเพิ่มเติมซึ่งจะมีรายละเอียดดังต่อไปนี้

1. ในส่วนการฝึกสอนโครงข่ายประสาทเทียมสำหรับจำแนกพื้นที่ที่ไม่สมมาตรในสนามนั้น ควรจะใช้เทคนิคการทำดรอปเอาต์ (Dropout Neural Network) เพื่อเพิ่มความสามารถในการทนทานต่อข้อมูลความแรงสัญญาณที่ไม่เข้าพวก (outliers) อันเนื่องมาจากคุณลักษณะของสัญญาณในตัวอาคาร
2. ควรติดตั้งเสาสัญญาณอ้างอิงไว้บนเพดาน เพื่อหลีกเลี่ยงไม่ให้เป็นสิ่งกีดขวางการเคลื่อนที่ของหุ่นยนต์ อีกทั้งยังเป็นการจำลองสถานการณ์ให้เหมือนกับการติดตั้งอุปกรณ์สื่อสารไร้สายตามอาคารทั่วไปในชีวิตประจำวันอีกด้วย
3. เปลี่ยนเครือข่ายสื่อสารไร้สายที่ใช้ในการบอกตำแหน่งเป้าหมายจากเครือข่ายซิกบีเป็นเครือข่ายวายฟาย (WiFi)
4. โครงสร้างของตัวหุ่นยนต์ที่ออกแบบมานั้นยังไม่ดีเท่าที่ควร ปัจจุบันเป็นการใช้วัสดุที่หาได้ตามท้องตลาดมาสร้างและประกอบขึ้น ทำให้ระหว่างการใช้งานจริง หุ่นยนต์ยังคงประสบปัญหาในการเคลื่อนที่ เช่น วัสดุที่ใช้ทำจากแผ่นอะคริลิกหรือท่อพีวีซีที่หาซื้อได้ตามร้านขายอุปกรณ์ก่อสร้าง ควรจะต้องมีการออกแบบที่ถูกต้องแม่นยำและสร้างขึ้นด้วยกระบวนการที่ถูกต้อง จึงทำให้โครงสร้างของตัวหุ่นยนต์แข็งแรงและเคลื่อนไหวได้สอดคล้องกับการควบคุมการเคลื่อนที่
5. ในส่วนสั่งการควบคุมการเคลื่อนที่ ปัจจุบันได้มีการใช้การควบคุมไร้สายผ่านวายฟาย (WiFi) และโมดูลส่งสัญญาณวิทยุ (RF) ซึ่งให้ความไวในการตอบสนองที่ดี อย่างไรก็ตามเมื่อเปรียบเทียบกันระหว่างการสั่งการผ่านวายฟายและโมดูลวิทยุ นั้น ผลปรากฏว่าการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมผ่านสายพายทำให้หุ่นยนต์ตอบสนองต่อคำสั่งที่ช้ากว่า ในอนาคตจึงควรมีการรองรับการสื่อสารไร้สายผ่านบลูทูธด้วย (Bluetooth) เพื่อให้สามารถสั่งการควบคุมผ่านโทรศัพท์มือถือด้วยความไวในการตอบสนองต่อคำสั่งที่ไวเทียบเท่าโมดูลวิทยุ

6. เนื่องจากส่วนต่าง ๆ ของระบบตามที่ได้แสดงในรูปแบบไดอะแกรมที่ 3.1 นั้นทำงานแยกส่วนกันอย่างอิสระแบบอะซิงโครนัส ทำให้ในบางครั้งบางส่วนของระบบก็จะดึงข้อมูลที่ยังไม่ถูกอัปเดตให้เป็นปัจจุบันไปประมวลผล ส่งผลให้ผลลัพธ์ที่ได้นั้นไม่ตรงตามความเป็นจริง ดังนั้นในอนาคตควรจะต้องมีการออกแบบโปรโตคอลสื่อสารเพิ่มเติม หรือเลือกใช้โทโพลยีในการเชื่อมต่อกันของเครือข่ายใหม่ ให้ระบบเลือกเฉพาะข้อมูลที่ถูอัปเดตให้เป็นข้อมูลปัจจุบันไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] Th. Bräunl, “**Embedded Robotic: Mobile Robot Design and Applications with Embedded Systems**”, Springer, 3rd ed. 2008, XIV, 546 p.197, 305
- [2] Chuan-Chin Pu, Chuan-Hsian Pu, Hoon-jae Lee “**Indoor Location Tracking using Received Signal Strength Indicator**”, Emerging Communications for Wireless Sensor Networks, July 27, 2015, pp. 229-255.
- [3] R. Yasmine and L. Pei, “**Indoor Fingerprinting Algorithm for Room Level Accuracy with Dynamic Database**”. IEEE UPINLBS, Shanghai, P.R., China, 3-4 Nov 2016, pp. 113-121.
- [4] K. Kaemarungsi, P. Krishnamurthy, “**Modeling of Indoor Positioning Systems Based on Location Fingerprinting**”, INFOCOM 2004, Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, 2004, pp.1012-1022.
- [5] B. Cook, G. Buckberry, I. Scowcroft, J. Mitchell, T. Allen, “**Location by Scene Analysis of Wi-Fi Characteristics**”
- [6] R. Sharma, S. Malhotra, “**Approximate Point in Triangulation (APIT) based Localization Algorithm in Wireless Sensor Network**”, IJRST, Vol. 2, Issue 03, 2008
- [7] Dr. Xuemin (Sherman) Shen, Dr. Yi Pan, “**FUNDAMENTALS OF WIRELESS SENSOR NETWORKS THEORY AND PRACTICE**”, John Wiley & Sons Ltd., p259, 2010
- [8] S. Cai, H. Pan, Z. Gao, N. Yao, Z. Sun “**Research of localization algorithm based on weighted Voronoi diagrams for wireless sensor network**”, EURASIP Journal on Wireless Communications and Networking 2014:50, Springer, 2014, pp.1-5
- [9] D. Blanco, B.L. Boada, and L. Moreno, “**Localization by Voronoi Diagram Correlation**” IEEE International Conference on Robotics & Automation, Seoul, Korea, 21-26 May 2001, pp.4232-4237.
- [10] S. A. Kharidia, Q. Ye, S. Sampalli, J. Cheng, H. Du, L. Wang, “**HILL: A Hybrid Indoor Localization Scheme**”, 10th International Conference on Mobile Ad-hoc and Sensor Networks, 2014, pp. 201- 206.
- [11] Z. Farid, R. Nordin, M. Ismail, N.F. Abdullah, “**Hybrid Indoor-Based WLAN-WSN Localization Scheme for Improving Accuracy Based on Artificial Neural Network**”. Mobile information system, Vol. 2016, May 2016, pp. 123–135.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [12] N.A. Sabto and K. Al Mutib, “**Autonomous mobile robot localization based on RSSI measurements using RFID sensor and neural network BPANN,**” Journal of King Saud University - Computer and Information Science. 23 Oct 2012, pp. 137-143.
- [13] W. Jun, Z. Fu, R. Tiansi, C. Xun, L. Gang, “**A NOVEL HYBRID LOCALIZATION METHOD FOR WIRELESS SENSOR NETWORK**”, INTERNATIONAL JOURNAL ON SMART SENSING AND INTELLIGENT SYSTEMS, Vol.9, No.3, September 2016, pp. 1323-1340.
- [14] H. Sallouha, A. Chiumento, S. Pollin, “**Localization in Long-range Ultra Narrow Band IoT Networks using RSSI**”, Communications (ICC), IEEE International conference, ISSN: 1938-1883, 21-25 May 2017
- [15] A. CHERIET, M. OUSLIM, K. AIZI, “**Localization in a Wireless Sensor Network based on RSSI and a decision tree**”, PRZEGLAD ELEKTROTECHNICZNY, ISSN: 0033-2097, Dec 2013, pp.121-125.
- [16] L-Z. Zhao, X-b. Wen, and D. Li, “**Amorphous Localization Algorithm Based on BP Artificial Neural Network**” International Journal of Distributed Sensor Networks. Vol. 2015, Jan 2015.
- [17] C. Laoudias, D. G. Eliades, P. Kemppe, C.G. Panayiotou, and M. M. Polycarpou, “**Indoor Localization Using Neural Networks with Location Fingerprints**”, ICANN 2009, Part II, 2009, pp. 954-963
- [18] S. Timotheatos, G. Tsagkatakis, P. Tsakalides, and P. Trahanias, “**Feature Extraction and Learning for RSSI based Indoor Device Localization**”.
- [19] L-Z. Zhao, X-b. Wen, and D. Li, “**Amorphous Localization Algorithm Based on BP Artificial Neural Network**” International Journal of Distributed Sensor Networks. Vol. 2015, Jan, 2015.
- [20] S. Timotheatos, G. Tsagkatakis, P. Tsakalides, and P. Trahanias, “**Feature Extraction and Learning for RSSI based Indoor Device Localization**”, ESANN 2017
- [21] S. Riisgaard and M. Rufus Blas, “**SLAM for Dummies A Tutorial Approach to Simultaneous Localization and Mapping**”, 2005
- [22] J. Hurwitz, D. Kirsch, “**Machine Learning For Dummies®**”, IBM Limited Edition, John Wiley & Sons, Inc., ISBN: 978-1-119-45495-3

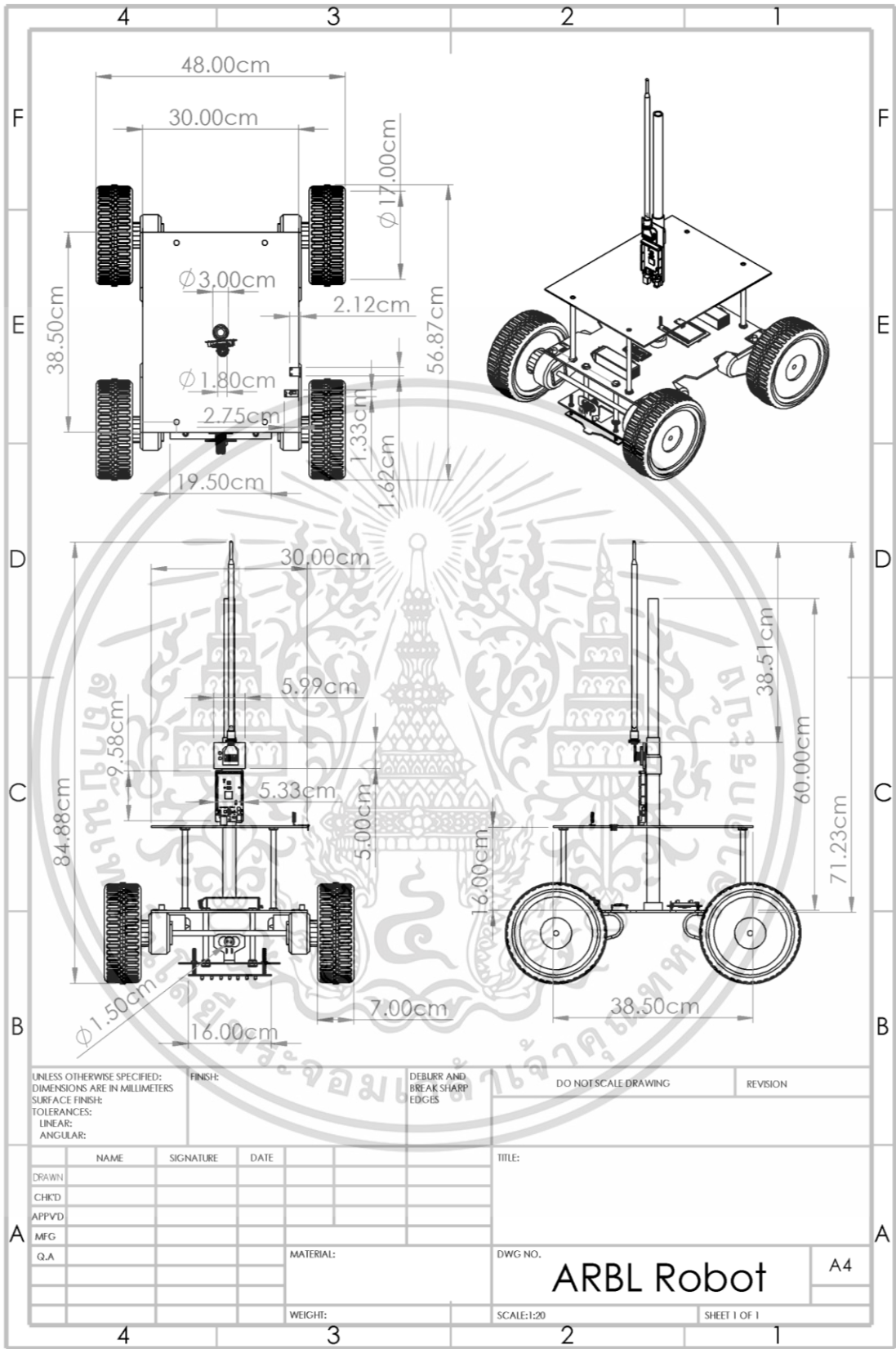
- [23] R. S. Sutton and A. G. Barto, "**Reinforcement Learning: An Introduction**", The MIT Press, Cambridge, Massachusetts, London, England, 2014-2015
- [24] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg and D. Hassabis, "**Human-level control through deep reinforcement learning**", Macmillan Publishers Limited., 26 Feb 2015, vol.518, p529-541
- [25] L. Khriji, F. Touati, K. Benhmed, and A. Al-Yahmedi, "**Mobile robot Navigation Based on Q-learning Technique**" International Journal of Advanced Robotic Systems, Vol. 8, No.1, 2011, pp. 45-51.
- [26] Ian Goodfellow, Yoshua Bengio, Aaron Courville, "**Deep Learning**", MIT Press, 2016
- [27] Dr Xuemin (Sherman) Shen, Dr Yi Pan, "**FUNDAMENTALS OF WIRELESS SENSOR NETWORKS THEORY AND PRACTICE**", 2010 John Wiley & Sons Ltd., p259, 2010
- [28] M. A. Matin, M. M. Islam, "**Overview of wireless sensor network**", Intech, p3-24, September 2012
- [29] J. Qin, S. Sun, Q. Deng, L. Liu, and Y. Tian, "**Indoor Trajectory Tracking Scheme Based on Delaunay Triangulation and Heuristic Information in Wireless Sensor Networks**" Sensor 2017, June 2017
- [30] M. Balcilar, E. Uslu, F. Cakmak, N. Altuntas, S. Marangoz, M. F. Amasyali, S. Yavuz, "**An Architecture for Multi-Robot Hector Mapping**", Innovations in Intelligent Systems and Applications (INISTA), 2016 International Symposium, 2-5 Aug 2016.
- [31] J. Simon, G. Martinovic, "**Navigation of Mobile Robots Using WSN's RSSI Parameter and Potential Field Method**", Acta Polytechnica Hungarica, Vol.10, No.4, 2013, pp. 107-118.
- [32] J. Fink, V. Kumar, "**Online Methods for Radio Signal Mapping with Mobile Robots**", 2010 IEEE International Conference on Robotics and Automation, IEEE, Anchorage, Alaska, USA, May 3-6, 2010, pp.1940-1945.
- [33] N. Jain, Y. P. Kumar, Dr. K S Nagla, "**Corner Extraction from Indoor Environment for Mobile Robot Mapping**", India Conference (INDICON), 2015 Annual IEEE, 2015
- [34] M. L. Rodrigues, L. Filipe M. Vieira and Mario F. M. Campos, "**Mobile Robot Localization in Indoor Environment Using Multiple Wireless Technologies**", IEEE, Brazilian Robotics Symposium and Latin American Robotics Symposium, 2012

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [35] P. Jensfelt, "**Approaches to Mobile Robot Localization in Indoor Environments**", Royal Institute of Technology (KTH), Sweden, 2001
- [36] "**White Paper - Beacons: The Technical Overview**", mibeacons a division of mubaloo
- [37] O. Gedds, "**A Guide to BLE Beacons**", Appflare, September 2014.
- [38] "**Getting Started with iBeacon Version 1.0**", 2 June 2014.
- [39] S. Feldmann, K. Kyamakya, A. Zapater, Z. Lue, "**An indoor Bluetooth-based positioning System: concept, Implementation and Experimental evaluation**", Proceedings of the International Conference on Wireless Networks, ICWN '03, June 23-26, 2003, Las Vegas, Nevada, USA.
- [40] P. Domingos, "**A Few Useful Things to Know about Machine Learning**", Department of Computer Science and Engineering, University of Washington, Seattle, WA
- [41] E. B. Baum, D. Hausler, "**WHAT SIZE NET GIVES VALID GENERALIZATION?**", Advances in Neural Information Processing Systems 1, NIPS Conference, Denver, Colorado, USA, Nov 1988, p.81-89
- [42] Stefan Kohlbrecher, Oskar von Stryk, Johannes Meyer, Uwe Klingauf, "**A Flexible and Scalable SLAM System with Full 3D Motion Estimation**", IEEE International Symposium on Safety, Security, and Rescue Robotics, 1-5 Nov 2011



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

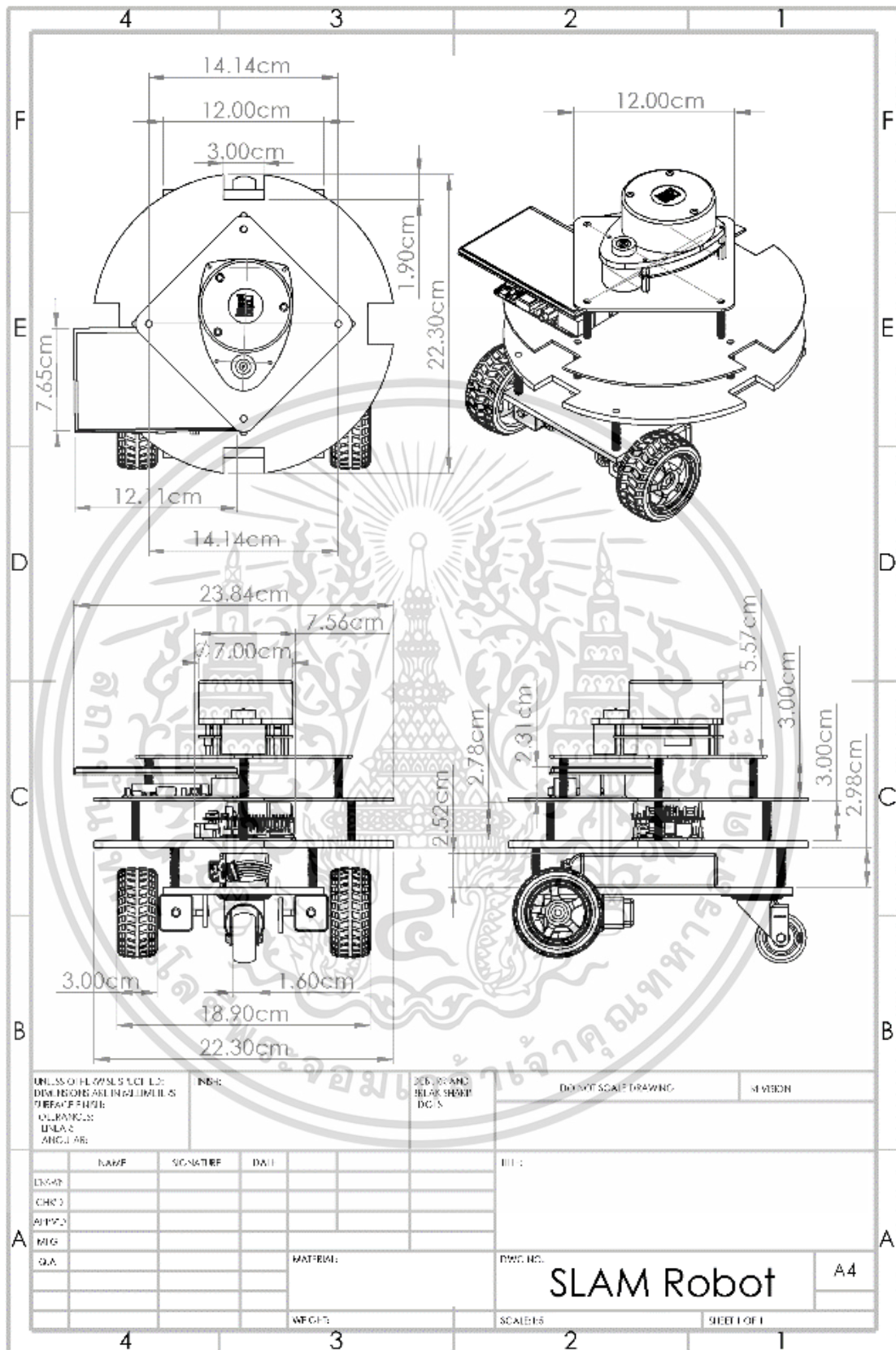


UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN MILLIMETERS
 SURFACE FINISH:
 TOLERANCES:
 LINEAR:
 ANGULAR:
 FINISH:
 DEBURR AND
 BREAK SHARP
 EDGES
 DO NOT SCALE DRAWING
 REVISION

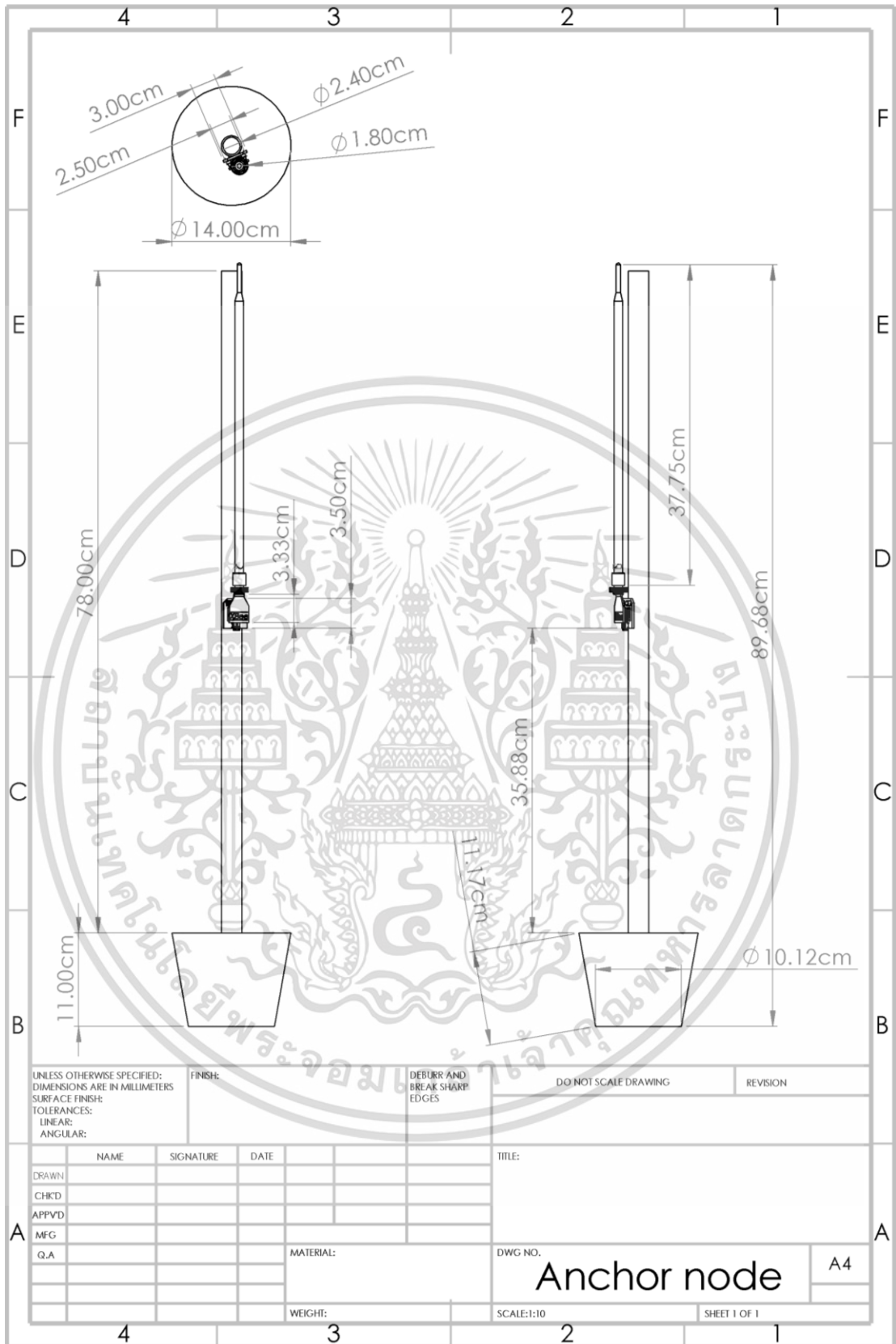
NAME	SIGNATURE	DATE
DRAWN		
CHKD		
APPVD		
MFG		
Q.A		

TITLE:
 DWG NO.
ARBL Robot
 A4
 SCALE:1:20
 SHEET 1 OF 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

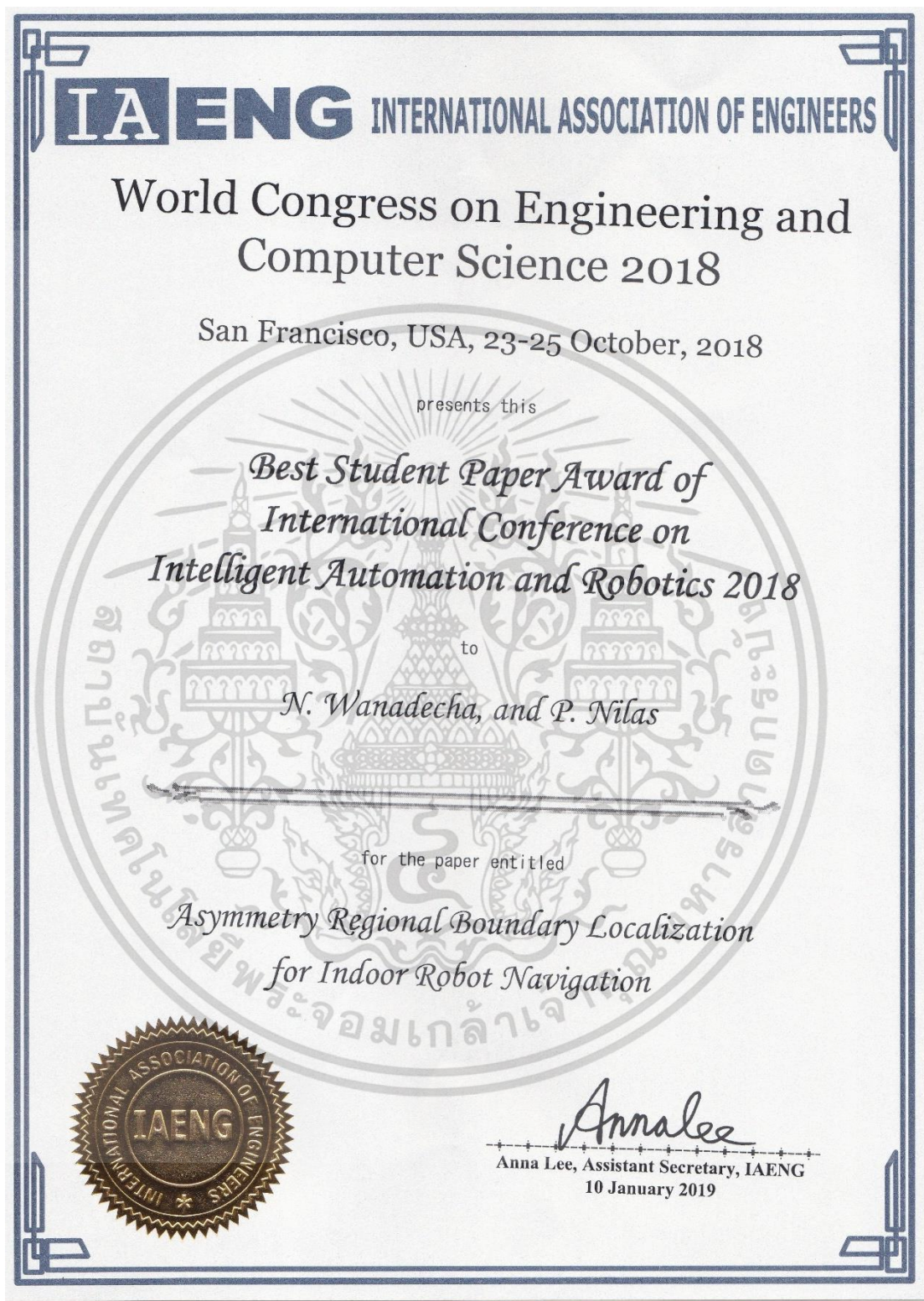


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Asymmetry Regional Boundary Localization for Indoor Robot Navigation

N. Wanadecha, and P. Nilas

Abstract—This paper presents the new methodology of indoor localization based on Received Signal Strength Indicator (RSSI) of Wireless Sensor Network (WSN) using Asymmetry Regional Boundary Localization (ARBL) which is designed to support mobile robot navigation. Since many types of obstruction such as walls, poles, and traps in the environment may confuse robot trajectory and avoidance strategy, our main objective is to analyze our desire test bed in the form of asymmetry region with the benefit of RSSI map fingerprinting technique. The regions are created from multiple anchor nodes' boundary intersection which is pre-determined by the working range that provide the finest repeatability value. Each region is classified by supervised neural network. When the robot senses the existence of obstacle, that particular region is declared as occupied zone which is not allowed to enter during path planning process. Once the entire virtual map is acquired, suitable path is obtained by Q-learning algorithm, thus the robot can travel through the field. The experiments are performed with 2 different types of test bed. Each of which shows region classification performance of 71.9% and 85.8%, respectively. Finally, to demonstrate the effective of our method, dealing with obstacles in the field, the comparison between conventional Q-learning algorithm and our proposed method is shown. The proposed method not only gives better learning progress but also spends less computational resources, reducing the complexity of the navigation problem.

Index Terms—Asymmetry Regional Boundary, indoor localization, mobile robot navigation, wireless sensor network

I. INTRODUCTION

Localization and navigation are considered to be the most important issues for mobile robot. In fact, the robot needs to know its current location respect to the world. So, it can find a way that suitable enough for reaching the goal in path planning process. One of the fundamental aspect most of the robot needed is the ability to detect obstacle and avoid undesirable collision that leads to harmful consequences. SLAM (Simultaneous Localization and Mapping) is the well-known efficient technique to locate and navigate robot that has been applied widely in real-world application. It makes use of the cooperation of multi-sensor (sensor fusion) such as wheel encoder, range finder, magnetic compass to provide the accurate information in the map. LiDAR

technology based on laser scanners has been applied as well as High-resolution cameras in order to get the real-time physical data with high quality from environment. Thus, the higher performance the sensing part, the more accuracy the map is. Unfortunately, its cost is surprisingly expensive for academic research and high computational embedded processor is required for the robot.

In indoor environment, WSN is already a basis of infrastructure. In term of target localization, it acts as the backbone of the system instead of GPS where signal propagation can't be utilized properly. Several localization schemes have been researched and proposed from many researchers. They can be broadly categorized into two classes: range-based and range-free technique. Since range-based localization relates to finding distance between target node and reference node with physical variable such as RSSI, this method provides the result as exact specific point. On the other hand, range-free method is also known as scene analysis that locates target node via other factor such as network structure, heuristic information etc. The output seems to be less specificity comparing to range-based. However, this method provides less complexity process and more robustness against the severe signal fluctuation which is the major cause of error.

SLAM requires high performance equipment that comes with high cost and computational power to finish the work. With this in mind, this paper presents the optional way to locate target using WSN incorporated with affordable price sensors called Asymmetry Regional Boundary Localization (ARBL). This proposed method aims to reduce the complexity of positioning error correction caused by the nature of radio signal. It also allows mobile robot to perceive the entire map, including obstacles and do path planning in further process. This paper is organized as follows. In section 2, literature review is briefly described including related work. Then, the proposed method is introduced thoroughly in section 3. Section 4, system overview is clarified. Section 5, experiment and result is explained, including the comparison between conventional Q-learning navigation algorithm and Q-learning with ARBL technique in simulation program. Finally, the conclusion and future work are summarized in section 6.

II. LITERATURE REVIEW

In [2], [3], [13], [14], [16], scene analysis model is used to locate the target in term of pattern recognition. It is also known as fingerprinting technique. The main idea is matching information, containing characteristic data of each particular zones between pre-measured data stored in database and current run-time data in online phase.

Manuscript received June 06, 2018.

N. Wanadecha is a master student with Instrumentation and Control Engineering Department, King Mongkut's Institute of Technology Ladkrabang, Ladkrabang, Bangkok 10520 Thailand (phone: +6685-075-0126; e-mail: 58601083@kmitl.ac.th).

P. Nilas is with Instrumentation and Control Engineering Department, King Mongkut's Institute of Technology Ladkrabang, Ladkrabang, Bangkok 10520 Thailand (e-mail: phongchai.ni@kmitl.ac.th).

ISBN: 978-988-14049-0-9

ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

WCECS 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Although, this technique required preparation phase that takes long time for initialization as well as large data storage to keep the record, its advantage is robustness against error from signal noise which mostly occurred to indoor environment. Due to the uncertainty of positioning result, some work doesn't require the specific output. The concept of geometry regional area is applied. For example, Voronoi diagram [5], [7], [8], Multi-granularity [9], Approximate Point-In-Triangle (APIT). These approach seems to be difficult in mathematical terms and algorithmic complexity is high even though some method can provide the error less than 1 meter approximately. Prior knowledge in signal processing may require in order to manage pre-processed data in proper way. It can be noticed that, machine learning is applied in [2], [4], [6], [14], and [16]. Neural Network is preferable technique in non-linear classification problem comparing to others. Furthermore, Deep Neural Network is used in [14] for function approximation, solving RSSI fingerprints in specific environment. The research is [15] predicts the future trend of localization method that hybrid technology and sensor fusion may provide the efficient result using the combination between WSN, Wi-Fi, and Bluetooth Low Energy (BLE). [2] and [13] prove this idea, reducing positioning error to 1.05 m. Additionally, [1] mentions the hardware modification in order to increase signal quality. Since this work claims that signal cannot be approximated consistently, the absorbing plate is installed underneath antenna in order to avoid interference with signal reflected by ground that caused multi-path effect.

The result of investigation can be concluded that range-free method is suitable for indoor localization with no line-of-sight of signal path. Fingerprinting with neural network is the most popular combination technique that works well and robust against signal fluctuation. Area-based localization may offer location result of the target properly instead of specific coordinates.

III. CONCEPT OVERVIEW

As mentioned earlier, our objective aims to create proper trajectory, avoiding any obstruction in the field located by RSSIs from WSN. We derive some benefits from related works above. Asymmetry Regional Boundary Localization can be divided into 4 steps

A. Regional Boundary

As introduced in [11], [12], iBeacon, an Apple trademark for its implementation of BLE beacons, uses localization technology with RSSI based ranging. Normally, using only signal strength has many chances to miscalculate the real distance. So, in iBeacon, the RSSI boundary has been considered and separated into different tiers such as immediate, near, far, and out of range (labelled as unknown). Of all the reasons, this idea leads to the appliance of Regional Boundary in this work. RSSI Boundary has been categorized into 2 tiers.

1) *In bounds*: the area between anchor's location and imagine boundary threshold.

2) *Out of bounds*: the area between imagine boundary and the rest of the field.

The boundary size of each anchors depends on signal propagation performance respect to the test bed. Plus, there are five anchors scattered in the field. So, the enclosed area,

which is occurred from the intersection among boundaries is called "Region".

B. Repeatability

The boundary threshold criterion depends on how good the radio module can perform in the field (like equipment's specifications) when it needs to provide RSSI value in the same distance over and over again. In fact, gage repeatability is a common standard that used to test the equipment stability and accuracy. This factor is concerned, in order to measure the capability of radio sources. The main objective using repeatability is to limit the imaginary line of boundary. The type of repeatability in this context can be separated into 3 parts as follows.

1) *Radius repeatability* R_r : the repetition value of reading RSSIs every 1m radius from the center of each radio source.

2) *Entire field repeatability* R_f : the value that defines repeatability of the anchor acts against the entire map.

3) *Threshold repeatability* R_T : a value of radius repeatability with less deviation from its reference.

To obtain the radius repeatability of anchor, first, rendering imaginary circles with several radius, let the anchor position as a center until the field is covered. Then, in each set of data, gathering all of the RSSIs from every coordinates lying on each imaginary circles. The radius repeatability equation can be determined as follows.

$$R_r = \sqrt{\frac{\sum_{i=1}^N (X_i - M)^2}{N - 1}} \quad (1)$$

where $M = \frac{1}{N} \sum_{i=1}^N X_i$ and $X = \frac{1}{C} \sum_{c=1}^C x_c$.

Equation (1), X_i denotes the estimated RSSI from coordinates x_i lying on imagine circumference in one radius of each dataset. C denotes the amount of interesting coordinates. M represents the average RSSI value of the radius. N represents the number of dataset. Suppose, we consider the repeatability of 10 dataset, so $N=10$. Next, the entire field repeatability comes from the average value of all radius. It means that with the condition of environment characteristic in this case, radio source generally provides this amount of error entire area. We use this feature as a reference for the final step. The entire map repeatability equation is described below in equation (2).

$$R_f = \frac{1}{j} \sum_{j=1}^j R_r \quad (2)$$

where j is number of considered radius which the origin point is the center of anchor. Finally, once the entire field repeatability is acquired, the last process is determining how close each radius repeatability compared to the reference. In equation (3), the threshold repeatability is shown as follows.

$$R_T = |R_f - R_r| \quad (3)$$

where R_T represents the absolute different between radius repeatability from i meter and the reference value R_f . The imaginary line of boundary is assigned to the radius with the finest value. Due to the different propagation pattern of each anchors, the radius repeatability of each anchor may not equally be the same. This reason leads to the conclusion that the region would be asymmetry shape.

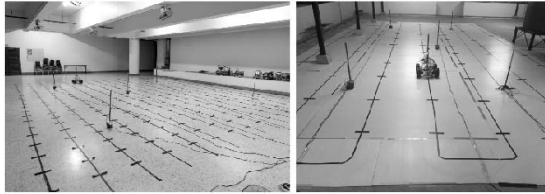


Fig. 1. Indoor test bed with 5 Xbee anchors and mobile robot. (Left) 10m x 10m size. (Right) 10m x 6m size.

C. Region Classification

In MATLAB, there are neural network tools that provides many training algorithms. However, there is no absolute solution that fits to any dataset. Trial and error method is needed to perform neural network hyper-parameter testing in order to obtain appropriate configured model that fits RSSI fingerprint dataset. Afterwards, the trained model is utilized in online mode, classifying the region respect to perceived RSSIs vector.

D. Building complete map

Once the system identifies the region from trained model classification in earlier process, the robot detects any obstacles in the field and records its discovered area into storage memory as an occupied region which is not allowed entrance. For vacant area, on the other hand, it is recorded as unoccupied region. Finally, all of "No trespassing" area and vacant entrance area are gathered and formed together into a complete map.

IV. SYSTEM DESIGN

A. Xbee network and test bed

Wireless Sensors Network in this work is composed of 6 Xbee series 2 modules. Five of them are scattered all over the test bed acting as Beacons or sometimes also known as Anchors. The sixth module is called target node installed to the mobile robot. Every module attaches the 9dBi Omnidirectional antenna, allowing the signal spread widely in horizontal axis.

As depicted in fig. 1, indoor clear space test bed is chosen with 2 different sizes: 10m x 10m and 10m x 6m. The test bed has been divided into a grid with 1 meter spacing. For test bed 1, all anchor node from number 1 to 5 are placed at coordinates (2,8), (8,8), (2,2), (8,2), (5,5), respectively. For test bed 2, all anchor nodes from number 1 to 5 are placed at coordinates (1.5, 9.5), (4.5, 9.5), (1.5, 1.5), (4.5, 1.5), (3.5, 5.5), respectively. Black line installed to the field is only for mobile robot tracking purpose in offline mode. The robot tracks the line until black cross is reached. It stops moving, collecting RSSI fingerprint data at every points. Once all location points are collected, the robot moves to home position at the start point and all RSSI fingerprints are considered as one dataset. However, the tracking line has no effect to the process in online mode. It has been removed before run-time mode starts.

B. Mobile robot

Robot is driven by 4-wheel 12VDC motor which is controlled by Arduino microcontroller. It comprises of sensors, line following infrared sensor and infrared range

TABLE I: THRESHOLD REPEATABILITY OF TEST BED 1

Anchor Number	Threshold Repeatability : test bed 1 (10 datasets)									
	1m	2m	3m	4m	5m	6m	7m	8m	9m	10m
A1	0.93	0.56	0.62	0.54	0.72	0.85	0.98	0.79	0.81	0.77
A2	0.16	0.83	0.3	0.24	0.04	0.5	0.05	0.82	0.39	0.43
A3	0.21	0.42	0.17	0.62	0.06	1.02	0.33	0.22	0.75	0.38
A4	0.39	0.29	0.66	0.29	0.62	0.12	0.35	0.83	0.53	0.59
A5	0.11	0.13	0.12	0.33	0.77	0.45	0.5	-	-	-

TABLE II: THRESHOLD REPEATABILITY OF TEST BED 2

Anchor Number	Threshold Repeatability : test bed 2 (10 datasets)									
	1m	2m	3m	4m	5m	6m	7m	8m	9m	10m
A1	-	0.48	0.07	0.42	0.77	0.17	0.17	0.97	0.22	-
A2	-	0.37	0.48	0.64	0.02	0.06	0.99	0.39	0.15	-
A3	-	0.10	0.06	0.21	0.60	0.18	1.31	0.16	0.33	-
A4	-	0.56	0.41	0.41	0.32	0.06	0.23	0.68	0.50	-
A5	0.22	0.16	0.89	0.37	0.21	1.84	-	-	-	-

finder for obstacle detection purpose. They are attached in front of the robot, providing object detection range about 0.2m. Moreover, Xbee target node is controlled by second micro-controller separately, reading RSSI from 5 anchor nodes with time interval of 500ms each. Finally, the robot communicates to Firebase, cloud database via Wi-Fi, requesting and responding all of necessary information such as RSSI values, update counts, movement control signal, obstacle detection signal etc.

C. Operator

The pre-programmed application, written for Android platform, is capable of several tasks. For instance, operating mode option (Fingerprint data collection or run-time mode), saving raw information in .xls file, controlling robot, and monitoring region output from cloud database.

D. Base station

MATLAB is responsible for training and testing classification neural network with raw data sets, collected from entire field in offline mode. Its responsible also includes data pre-processing and model performance analyzing. In online mode, m-file script is written and executed to handle classification task with current RSSIs vector, sending back the classified output to database. Also, it creates communication between the station and cloud database via RESTful API.

V. EXPERIMENT AND RESULT

A. Establishing regional boundary: Repeatability estimation

Repeatability of Xbee module assigned as anchor node is statistically tested with pre-collected 10 datasets. The range that provides good repeatability is assigned as a threshold of the boundary. The whole process is clearly depicted in concept overview. Table 1 and 2 show the calculated deviation repeatability from 10 datasets of test bed 1 and 2, respectively. The finest repeatability value is the least value in a row. For each anchor, the selected range with the best repeatability value is assigned as a boundary threshold and any enclosed shape becomes asymmetry regional area with several size. Fortunately, both test bed can be divided into 13 regions. It can be noticed that, both tables have some blank cells. For test bed 1, maximum coverage area of anchor 5 is 7m radius. Further than that, the radius is out of the test bed. Same reason happens to test bed 2. However, for 1m radius of anchor 1 to 4, there is no collected RSSI data to be calculated in that particular point. Therefore, the repeatability in this range is ignored.

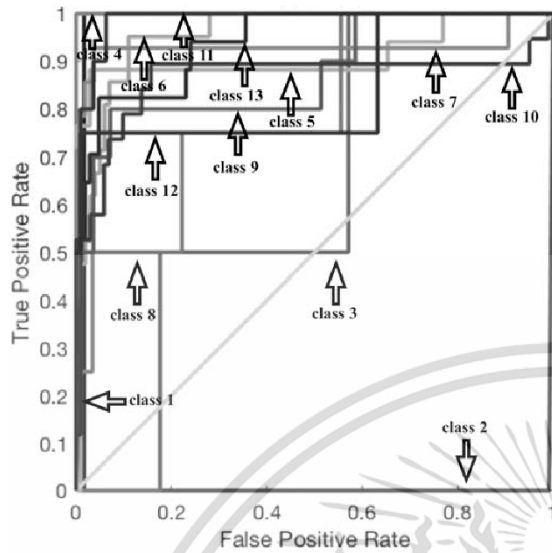


Fig. 2. Receive Operating Characteristic (ROC) graphs show classification result of test bed 1. This graph represents testing phase result.

B. Neural network hyper-parameter testing and regional classification performance

In this part of experiment, artificial neural network is involved as a regional classification model. Since there are 2 types of environment, neural network needs to be carefully design to suit its own data characteristic. The network structure for test bed 1 is configured as follows. Input layer consists 5 input nodes (5 nodes from 5 anchor raw RSSI). Hidden layer consists of 3 layers with 12 nodes each. Hyperbolic tangent Sigmoid is applied as activation function. Output layer consists of 13 nodes representing the 13 region needed to be identified in the form of one-hot encoding vector. Bayesian Regularization, training algorithm, is applied as a training algorithm. This configuration is randomly chosen based on MATLAB documentation without any experience or priori knowledge.

The result shows that training and testing performance are 83.5% and 71.8%, respectively. Also, receiver operating characteristic (ROC) graph shows the classification result from testing phase. As illustrated in fig. 2, true positive rate means model correctly classifies the data as expected and false positive rate means the model gives absolute false prediction. Obviously, the model is too complexity. It seems to be learning noise data instead of its true pattern. Although, it gives over-fitting behavior, the trained model still provides some correct region output using current RSSI data in run-time mode.

However, for test bed 2, to prevent the over-fitting model, hyper-parameter testing of neural network is performed in order to derive the best model performance that fits RSSI fingerprinting map properly. Several strategies are applied to the model. For example, L2 regularization preventing the influence from outlier data, class imbalance fix with added White Gaussian noise increasing robustness of the model. The configuration values are set up as follows. Learning rate and regularization value is chosen within the array of [0.1, 0.01, 0.001, 0.0001, 0.00001]. Learning algorithm is

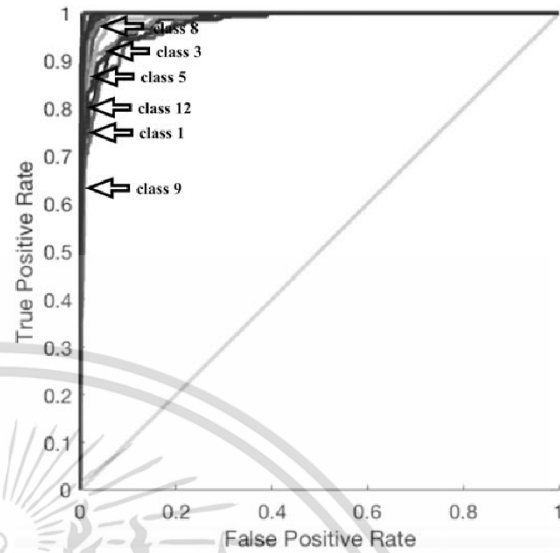


Fig. 3. Receive Operating Characteristic (ROC) graphs show classification result of test bed 2. This graph represents testing phase result.

tested between *trainsec* and *trainrp*. Activation function is tested between *tansig* and *posim*. The testing includes hidden layer number of 1 and 2 and hidden nodes from 20, 40, 60, and 80. Output layer is assigned with softmax function in order to normalize the predicted value between 0-1. Lastly, since training data affects model performance, 10-fold cross validation is utilized in order to prove the true performance by randomly split raw data into 2 partitions: 90% of entire data is assigned to be training set, the rest 10% is assigned to be testing set. This process is repeated until all of the data completely shuffled. Aftermath, the average performance is calculated. Hyper-parameter testing result shows that, the model that providing best result ends up with following configuration. Learning rate and Regularization = 0.01. Training algorithm = *trainsecg*. Activation function = *tansig*. Hidden layer = 2. Hidden node = 80. Average training and testing performance derived from confusion matrix are 87.61% and 85.76%, respectively. Furthermore, ROC graph depicted in fig. 3 shows true positive rate and false positive rate of classification result in testing phase. The entire data falls into true positive rate. In other words, this model is good fitting to RSSI-fingerprinting map of test bed 2, providing good generalization result.

C. Comparison between conventional Q-learning and Q-learning with ARBL

To demonstrate the effectiveness of the proposed algorithm, the navigation experiment is setup in simulation program written in python. In [10], the mobile robot navigation based Q-learning technique addresses the problem of robot behavior when facing different types of obstacle. For example, objects, walls, and traps. Also, another drawback is, Q-learning algorithm is a tabular method requiring large size of memory to store Q-value of relate state, action and reward. Although, [10] offers fuzzy logic implementation reducing the complexity of the navigation, the robot still needs some strategy to handle

emergency situation like U-trap. On the other hand, ARBL identifies any obstacle as a region. Therefore, the opportunity getting stuck in the trap is relieved.

The simulation of test bed is created as a grid. The robot movement restricts to only 4 actions: [up, down, left, right]. Discount factor is set as 0.9. Learning rate is 0.1. Epsilon-greedy for random exploration is initialized with 0.5. Every step the agent takes is penalized with -0.1. Reward at goal position is 10. Test bed simulation is illustrated in fig. 4.

Red triangle represents goal position. Green circle represents start position of the robot. Black square represents occupied zone which is not allowed to pass. Agent needs to find the optimal policy (represented by blue line) to get to the goal. Q-value implies how good the taken action can be valuable for the agent. It learns from experience in each episode, moving through the field. Figure 5 and 6 illustrate differentiate of Q-learning process along with episode until convergence is reached (differentiate

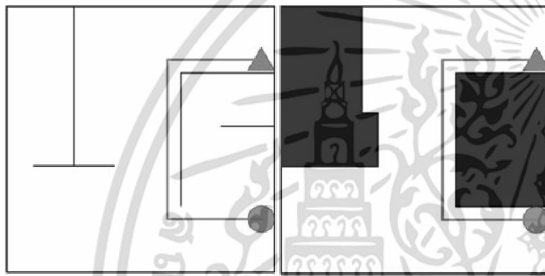


Fig. 4. Test bed 1 path planning simulation (Left) with conventional Q-learning, (Right) ARBL with Q-learning.

remains zero), meaning the optimal policy is obtained. Figure 5 shows test bed 1's learning performance of Q-learning with ARBL (red graph) against conventional method (blue graph). Figure 6 shows test bed 2's. In test bed 1, it can be noticed that, our proposed method offer better progress, reaching convergence at episode 200 earlier than the conventional one. Since the obstacle has been considered as region, the complexity of path planning problem is reduced. Our proposed method outperforms in computational time and memory size. Q-learning with ARBL spends 1.12 minutes and Q-value hash table size of 59 while conventional method spends 2.04 minutes with hash table size of 81.

For test bed 2, learning progress appeared to be the same trend. Due to the smaller size of environment, computational time, hash table size, and learning progress is less than test bed 1's situation. Both methods reach convergence at 200 episode approximately. Although our proposed method learning progress seems to be fluctuated before converged, computational time is only 41 seconds and 32 of hash table size is spent while conventional method wastes 1.17 minute and 48 hash table size to solve the problem.

VI. CONCLUSION

This paper proposes Asymmetry Regional Boundary Localization (ARBL) designed to support indoor navigation for mobile robot. Our proposal exploits the advantage of area-based range-free localization with RSSI from WSN in the field to locate mobile target and obstacles in the form of asymmetry region. Asymmetry region is established from

the intersection of each anchor's boundary which occurred from estimated communication radius with the finest

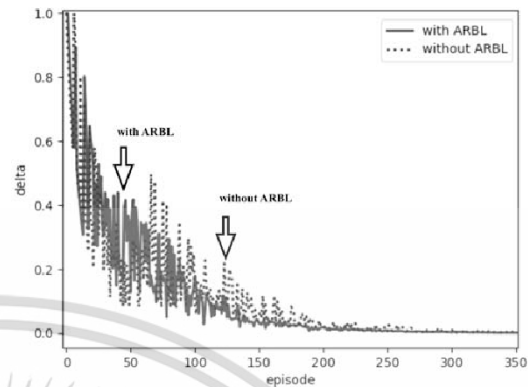


Fig. 5. Learning progress graph of test bed 1. Y axis represents deltas of Q-value. X axis represents episode.

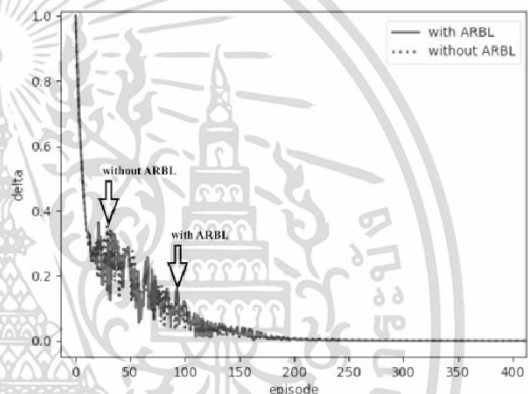


Fig. 6. Learning progress graph of test bed 2. Y axis represents deltas of Q-value. X axis represents episode.

repeatability of reading RSSI in offline phase. For test bed 1, although the model is over-fitting, trained neural network gives the classification performance of 71.9% to classify and locate region with real-time RSSI fingerprint vector, acquiring from mobile robot. However, for test bed 2, hyper-parameter test is performed in advance to fix over-fitting problem. Eventually, the model turns out to be good with classification performance of 85.8%. ROC graphs show the correspondence of both of the results. Finally, in navigation estimation task, ARBL is applied to Q-learning algorithm, comparing with conventional approach in simulation. The result shows that our method outperforms in case of computational time and memory resource reduction. Also, in test bed 1, the proposed method's learning progress reaches convergence point with less episode than the other. Due to the fact that the obstacle is considered as regional area, the complexity of navigation problem is decreased. Moreover, the complication of avoidance strategy when the robot encounters emergency situation or gets stuck within U-trap, is no longer required because the suitable path is already planned. This method accomplishes in cost and high computational resource reduction for the robot's component.

REFERENCES

- [1] J. Graefenstein and M. Essayed Bouzouraa, "Robust Method for Outdoor Localization of a Mobile Robot Using Received Signal

- Strength in Low Power Wireless Networks” in *IEEE International Conference on Robotics and Automation*, Pasadena, CA, May, 19-23, 2008, pp. 33–38.
- [2] Z. Farid, R.Nordin, M. Ismail, and N.F. Abdullah, “Hybrid Indoor-Based WLAN-WSN Localization Scheme for Improving Accuracy Based on Artificial Neural Network”. *Mobile information system*, Vol. 2016, May, 2016, pp. 123–135.
- [3] R. Yasmine and L. Pei, “Indoor Fingerprinting Algorithm for Room Level Accuracy with Dynamic Database”. *IEEE UPINLBS*, Shanghai, P.R., China, Nov, 3-4, 2016, pp. 113-121.
- [4] N.A. Sabto and K. Al Mutib, “Autonomous mobile robot localization based on RSSI measurements using RFID sensor and neural network BPANN,” *Journal of King Saud University - Computer and Information Science*. Oct, 23, 2012, pp. 137-143.
- [5] S. Cai, H. Pan, Z. Gao, N. Yao, and Z. Sun “Research of localization algorithm based on weighted Voronoi diagrams for wireless sensor network” *Journal on Wireless Communications and Networking* 2014.
- [6] L-Z. Zhao, X-b. Wen, and D. Li, “Amorphous Localization Algorithm Based on BP Artificial Neural Network” *International Journal of Distributed Sensor Networks*. Vol. 2015, Jan, 2015.
- [7] J. Qin, S. Sun, Q. Deng, L. Liu, and Y. Tian, “Indoor Trajectory Tracking Scheme Based on Delaunay Triangulation and Heuristic Information in Wireless Sensor Networks” *Sensor* 2017, June 2017.
- [8] D. Blanco, B.L. Boada, and L. Moreno, “Localization by Voronoi Diagram Correlation” *IEEE International Conference on Robotics & Automation*, Seoul, Korea, May, 21-26, 2001, pp.4232-4237.
- [9] F. Shang, Y. Jiang, A. Xiong, W. Su, and L. He, “A Node Localization Algorithm Based on Multi-Granularity Regional Division and the Lagrange Multiplier Method in Wireless Sensor Networks”. *Sensor* 2016, Nov, 2016.
- [10] L. Khrijji, F. Touati, K. Benhmed, and A. Al-Yahmedi, “Mobile robot Navigation Based on Q-learning Technique” *International Journal of Advanced Robotic Systems*, Vol. 8, No.1, 2011, pp. 45-51.
- [11] “Getting Started with iBeacon Version 1.0”, June, 2, 2014.
- [12] N. Newman, “Apple iBeacon technology briefing” *Journal of Direct, Data and Digital Marketing Practice*, vol. 15, no.3, pp. 222-225, Jan, 2014.
- [13] P. Kriz, F. Maly, and T. Kozel, “Improving Indoor Localization Using Bluetooth Low Energy Beacons”, *Mobile Information System*, Vol. 2016, March, 2016.
- [14] S. Timotheatos, G. Tsagakatakis, P. Tsakalides, and P. Trahanias, “Feature Extraction and Learning for RSSI based Indoor Device Localization”.
- [15] H. Liu, H. Darabi, P. Banerjee, and J. Liu, “Survey of Wireless Indoor Positioning Techniques and Systems”, *IEEE TRANSACTIONS ON SYSTEM, MAN, AND CYBERNATICS-PART C: APPLICATION AND REVIEWS*, Vol.37, No.6, Nov, 2008, pp.1067-1078.
- [16] C. Laoudias, D. G. Eliades, P. Kemppi, C.G. Panayiotou, and M. M. Polycarpou, “Indoor Localization Using Neural Networks with Location Fingerprints”, *ICANN 2009, Part II*, 2009, pp. 954-963

ประวัติผู้เขียน

ชื่อ-นามสกุล	นรินทร์เดช วรรณะเดชะ
วัน เดือน ปีเกิด	29 มิถุนายน 2536
ประวัติการศึกษา	2557 วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมการวัดคุม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ความสนใจเฉพาะด้าน	Robotics Artificial Intelligence Internet of Things Mobile Applications



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้