ระบบการจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุในโรงงานอุตสาหกรรม

Management of a Multi-robots System for Industrial Material Handling

พงศกร ชาญชัยชูจิต

Pongsakorn Chanchaichujit

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมเครื่องกล
มหาวิทยาลัยสงขลานครินทร์

A Thesis Submitted in Partial Fulfillment of the Requirements for the
Degree of Master of Engineering in Mechanical Engineering
Prince of Songkla University
2562

ลิขสิทธิ์ของมหาวิทยาลัยสงขลานครินทร์

ระบบการจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุในโรงงานอุตสาหกรรม

Management of a Multi-robots System for Industrial Material Handling

พงศกร ชาญชัยชูจิต

Pongsakorn Chanchaichujit

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมเครื่องกล
มหาวิทยาลัยสงขลานครินทร์

A Thesis Submitted in Partial Fulfillment of the Requirements for the
Degree of Master of Engineering in Mechanical Engineering
Prince of Songkla University
2562

ลิขสิทธิ์ของมหาวิทยาลัยสงขลานครินทร์

ชื่อวิทยานิพนธ์ ระบบการจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุในโรงงานอุตสาหกรรม
ผู้เขียน นายพงศกร ชาญชัยชูจิต
สาขาวิชา วิศวกรรมเครื่องกล

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

.....
(รองศาสตราจารย์ ดร. พฤทธิกร สมิตไมตรี)

คณะกรรมการสอบ

.....ประธานกรรมการ
(ดร. ปรมินทร์ เณรานนท์)

.....กรรมการ
(รองศาสตราจารย์ ดร. สุภาภรณ์ สุวรรณรังษี)

.....กรรมการ
(รองศาสตราจารย์ ดร. พฤทธิกร สมิตไมตรี)

บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมเครื่องกล

.....
(ศาสตราจารย์ ดร. ดำรงศักดิ์ ฟ้ารุ่งสาง)
คณบดีบัณฑิตวิทยาลัย

ขอรับรองว่า ผลงานวิจัยนี้มาจากการศึกษาวิจัยของนักศึกษาเอง และได้แสดงความขอบคุณบุคคลที่มีส่วนช่วยเหลือแล้ว

ลงชื่อ .....
(รองศาสตราจารย์ ดร. พฤทธิกร สมิตไมตรี)
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

ลงชื่อ .....
(นายพงศกร ชาญชัยชูจิต)
นักศึกษา

ข้าพเจ้าขอรับรองว่า ผลงานวิจัยนี้ไม่เคยเป็นส่วนหนึ่งในการอนุมัติปริญญาในระดับใดมาก่อน และไม่ได้ถูกใช้ในการยื่นขออนุมัติปริญญาในขณะนี้

ลงชื่อ .....
(นาย พงศกร ชาญชัยชูจิต)
นักศึกษา

ชื่อวิทยานิพนธ์ ระบบการจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุในโรงงานอุตสาหกรรม
ผู้เขียน นายพงศกร ชาญชัยชูจิต
สาขาวิชา วิศวกรรมเครื่องกล
ปีการศึกษา 2561

บทคัดย่อ

ปัจจุบันการขนถ่ายวัสดุในโรงงานอุตสาหกรรมมีการนำหุ่นยนต์ AGV มาใช้กันอย่างแพร่หลาย ซึ่งพบว่าส่วนใหญ่มักนำหุ่นยนต์ AGV หนึ่งตัว มาใช้ในการขนถ่ายวัสดุเพียงหนึ่งชนิด จากสถานีงานต้นทาง ไปยังสถานีงานปลายทาง และทำซ้ำแบบเดิมตลอดเวลา งานวิจัยนี้จึงนำเสนอการออกแบบระบบจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุในโรงงานอุตสาหกรรม โดยใช้โปรแกรม ROS (Robot Operating System) เพื่อพัฒนาศักยภาพระบบขนถ่ายวัสดุ เพิ่มความยืดหยุ่น ลดเวลาในการทำงานของหุ่นยนต์ ด้วยวิธี Market-based approach ร่วมกับ Dijkstra's algorithm สำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม ผลการจำลองการทำงานพบว่า ระบบสามารถควบคุมหุ่นยนต์ AGV ต่างชนิดกันได้ สามารถเลือกสั่งการหุ่นยนต์ขนถ่ายวัสดุได้คุ้มค่าที่สุดจากการประเมินงานของหุ่นยนต์ทั้งหมดในระบบ ณ สภาวะปัจจุบันของการทำงาน ระบบจัดการนี้สามารถลดเวลาการทำงานได้ครึ่งหนึ่งเมื่อเทียบกับวิธีแบบ FIFO นอกจากนี้ยังพบว่าการเลือกจำนวนหุ่นยนต์ให้กับระบบการจัดการหุ่นยนต์หลายตัวส่งผลต่อความเร็วในการขนถ่ายวัสดุอีกด้วย ผลการจำลองพื้นที่ทำงานขนาด 65 ตารางเมตร แสดงให้เห็นว่าหุ่นยนต์ 3 ตัว สามารถขนถ่ายวัสดุ 100 ชุดคำสั่งงานได้เร็วที่สุด งานวิจัยนี้แสดงให้เห็นว่าระบบสามารถจัดการงานสำหรับการขนถ่ายวัสดุโดยเลือกหุ่นยนต์ทำงานให้มีรอบเวลาทำงาน (cycle time) ได้เร็วที่สุด สามารถควบคุมหุ่นยนต์ที่มีลักษณะแตกต่างกันทำให้เกิดความยืดหยุ่นในการขนถ่ายวัสดุ ซึ่งสามารถนำไปใช้ขนถ่ายวัสดุในโรงงานอุตสาหกรรมที่ใช้หุ่นยนต์หลายตัวในการขนถ่ายวัสดุหลายชนิดจากสถานีเริ่มต้นไปยังสถานีปลายทางได้มากกว่าหนึ่งสถานี

คำสำคัญ: ระบบหุ่นยนต์หลายตัว, วิธีการเปรียบเทียบราคาตลาด, ขั้นตอนวิธีของไดก์สตรา, การขนถ่ายวัสดุ

Thesis Title	Management of a Multi-robots System for Industrial Material Handling
Author	Mr.Pongsakorn Chanchaichujit
Major Program	Mechanical Engineering
Academic Year	2018

ABSTRACT

Recently, material handling of manufacturing factories trend to adopt AGV robots. However, AGV robots mostly handle one-by-one material from a start station to a final station, repeatedly. This research introduces a multi-robots material handling system that can handle multiple robots and stations at the same time. The control program is based on ROS (Robot Operating System). The objective of this research is to improve effectiveness of material handling system by adding more flexibility and reducing the robot work time. The management system uses the market-based approach combined with Dijkstra's algorithm for material handling in a manufacture factory. The simulation results found that the system is able to control various types of AGV robots. This means that the system has ability to order and execute the best robot among all working robots in the system for an incoming task at the current situation. This reduces the total working time by a half comparing with FIFO method. In the case study of the 65-m^2 working area, the optimum number of the robots is 3 to execute 100 tasks. The research results shown that the system can arrange materials handling by choosing the suitable robots for overall fastest cycle time. This system increases flexibility in handling materials in various working spaces. The knowledge of this research can be applied on many factories, which use multiple robots for various tasks to hand materials from start stations to final stations.

***Keywords**:* Multi-robots system, Market-based approach, Dijkstra's algorithm, Material handling.

กิตติกรรมประกาศ

ขอขอบคุณ รศ.ดร.พฤทธิกร สมิตไมตรี อาจารย์ที่ปรึกษา ผู้ที่คอยให้คำปรึกษาและคำแนะนำต่างๆ ทั้งทางเทคนิคและการเขียนวิทยานิพนธ์ตลอดระยะเวลาของการทำวิจัยนี้ ขอขอบคุณ รศ.ดร.วนิดา อาจารย์ประจำคณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมอุตสาหการ ที่ให้คำแนะนำในช่วงเริ่มต้นการทำวิจัยเกี่ยวกับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม ขอขอบคุณ รศ.ดร.สุภาภรณ์ สุวรรณรังษีและ ดร.ปรมินทร์ เณรานนท์ ผู้ให้คำแนะนำและเพิ่มเติมความรู้ในการทำวิจัยในครั้งนี้ ขอขอบคุณอาจารย์ พี่ๆ และน้องๆ ภายในภาควิชาวิศวกรรมเครื่องกลที่ให้คำแนะนำและแนวทางการแก้ไขปัญหาในระหว่างการทำวิจัย

นอกจากนี้ขอขอบคุณคุณพ่อ คุณแม่ และญาติพี่น้องที่ให้กำลังใจและเป็นผู้อยู่เบื้องหลังในการเรียนรวมถึงคอยสนับสนุนทุกๆ เรื่องจนสามารถทำงานวิจัยฉบับนี้ได้ลุล่วง

สุดท้ายนี้งานวิจัยฉบับนี้จะสำเร็จไม่ได้ หากตนเองไม่มีความอดทนต่ออุปสรรคต่างๆ ที่พบเจอในช่วงของการทำวิจัยทั้งในเรื่องของเทคนิค การคิดค้นวิธีการแก้ปัญหาระหว่างการทำวิจัย ขอขอบคุณตนเองที่มีความพยายามอุตสาหะในการต่อสู้กับปัญหาจนสามารถทำงานวิจัยนี้ได้สำเร็จ ถึงแม้จะต้องมีการค้นคว้าในอีกหลายประเด็นเพื่อนำงานวิจัยนี้ไปสู่การใช้งานจริง แต่ก็ถือว่างานวิจัยนี้ได้ประสบความสำเร็จไปอีกก้าวหนึ่งและหวังว่าคงเป็นบันไดความรู้ให้แก่ผู้อื่นได้นำไปใช้ต่อยอดในอนาคต

พงศกร ชาญชัยชูจิต

สารบัญ

	หน้า
บทคัดย่อ	(5)
Abstract	(6)
กิตติกรรมประกาศ	(7)
สารบัญ	(8)
รายการตาราง	(9)
รายการภาพประกอบ	(10)
สัญลักษณ์และคำย่อ	(12)
รายการผลงานที่ตีพิมพ์และการประชุมวิชาการ	(14)
สำเนาต้นฉบับที่ได้รับการยินยอมจากผู้พิมพ์ผลงาน	(15)
บทนำ	1
วัตถุประสงค์	8
วิธีการวิจัย	10
ผลและการวิเคราะห์	29
เอกสารอ้างอิง	31
ภาคผนวก	
วารสารวิชาการ: ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม	33
1. บทนำ	36
2. วิธีการวิจัย	37
3. ระบบการจัดการงาน	37
3.1 การจัดการชุดคำสั่งงาน	38
3.2 การติดตามสถานะงานและหุ่นยนต์	38
3.3 การจัดการงาน	38
3.4 วิธีการเปรียบเทียบราคาตลาด	39
3.5 การควบคุมหุ่นยนต์	40
3.6 การจัดการเส้นทางของหุ่นยนต์	40
4. ผลการทดลอง	41
5. สรุป	42
เอกสารอ้างอิง	43
ประวัติผู้เขียน	45

รายการตาราง

หน้า

บทนำ

ตารางที่ 1 การแปลงเส้นทางเป็นรูปแบบการเคลื่อนที่ 23
ตารางที่ 2 ตัวอย่างการสร้างชุดคำสั่งงาน 27

วารสารวิชาการ: ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม

ตารางที่ 1 การแปลงเส้นทางเป็นรูปแบบการเคลื่อนที่ 40
ตารางที่ 2 ตัวอย่างการสร้างชุดคำสั่งงาน 42

รายการภาพประกอบ

หน้า

บทนำ

รูปที่ 1 แสดงการควบคุมหุ่นยนต์ที่สามารถประกอบตัวใหม่ได้ด้วยตัวเอง 2
รูปที่ 2 แสดงการใช้งานระบบศูนย์กลางของหุ่นยนต์หลายตัวในโรงพยาบาล 3
รูปที่ 3 แสดงขั้นตอนวิธีของไดค์สตราขั้นตอนที่ (1) และ (2) 5
รูปที่ 4 แสดงขั้นตอนวิธีของไดค์สตราขั้นตอนที่ (3) 6
รูปที่ 5 แสดงขั้นตอนวิธีของไดค์สตราในขั้นตอนขยายลิสต์ 6
รูปที่ 6 การทำงานของ ROS 7
รูปที่ 7 แสดงเวลาการจัดการงานสำหรับขนถ่ายวัสดุแบบ FIFO และ Market based approach 9
รูปที่ 8 ภาพรวมของระบบจัดการงาน 10
รูปที่ 9 แสดงรูปแบบการเพิ่มงานเข้าไปยังระบบการจัดการงาน 11
รูปที่ 10 แสดงแผนที่แบบกริดที่มีเส้นทางการเคลื่อนที่แบบ Unidirectional 12
รูปที่ 11 แสดงปัญหาการใช้เส้นทางร่วมกันของหุ่นยนต์ 13
รูปที่ 12 แสดงเส้นทางที่ถูกเพิ่มระยะให้กับหุ่นยนต์เมื่อเกิดปัญหาหุ่นยนต์ใช้เส้นทางที่สั้นที่สุดร่วมกัน 13
รูปที่ 13 แสดงแผนที่แสดงจุดโหนดต่างๆ เพื่อใช้ในการหาเส้นทางที่สั้นที่สุดด้วยขั้นตอนวิธีของไดก์สตาร์ 14
รูปที่ 14 แสดงเส้นทางที่สั้นที่สุดจากขั้นตอนวิธีของไดก์สตาร์ 15
รูปที่ 15 แสดงผลลัพธ์ที่ได้จากสถานีพิกัด (3,13) ไปยังสถานีพิกัด (5,13) 17
รูปที่ 16 แสดงผลลัพธ์ที่ได้จากสถานีพิกัด (0,0) ไปยังสถานีพิกัด (0,6) 18
รูปที่ 17 ขั้นตอนการทำงานของการจัดการงาน 19
รูปที่ 18 ตัวอย่างคำสั่งงานที่ 5 20
รูปที่ 19 ตัวอย่างการหาค่า cost ที่ได้จากระบบการจัดการ 20
รูปที่ 20 แสดงตัวอย่างกรณีงานหุ่นยนต์ 1 ตัวทำงานได้เร็วที่สุดมากกว่า 1 งาน 21
รูปที่ 21 การระบุทิศและการเคลื่อนที่ของหุ่นยนต์เทียบกับแผนที่ 24
รูปที่ 22 เหตุการณ์ deadlock ของหุ่นยนต์ 4 ตัวที่ใช้เส้นทางพร้อมกัน 24
รูปที่ 23 ขั้นตอนการทำงานของการจัดการเส้นทาง 25
รูปที่ 24 แสดงการแก้ปัญหาหุ่นยนต์ว่างงานอยู่ตรงจุดสถานีงาน 26

รูปที่ 25 แสดงการติดตามสถานะงานและหุ่นยนต์ของระบบการจัดการงาน 26
รูปที่ 26 เวลาที่ใช้ในการทำงานของระบบหุ่นยนต์หลายตัวแบบ FIFO เปรียบเทียบกับวิธี Market based approach 28
รูปที่ 27 เวลาการทำงานของระบบจัดการหุ่นยนต์ที่จำนวนชุดคำสั่งเท่ากัน แต่จำนวนหุ่นยนต์ในระบบเพิ่มขึ้น 29

วารสารวิชาการ: ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม

รูปที่ 1 แผนที่จำลองในโรงงานอุตสาหกรรมสำหรับทดสอบการจัดการงาน 37
รูปที่ 2 แผนภูมิที่สร้างจากขั้นตอนวิธีของไดค์สตรา 37
รูปที่ 3 เส้นทางที่สั้นที่สุดจากสถานีพิกัด (0,0) ไปยังสถานี (0,6) 38
รูปที่ 4 ภาพรวมของระบบจัดการงาน 38
รูปที่ 5 ขั้นตอนการทำงานของการจัดการงาน 39
รูปที่ 6 การระบุทิศและการเคลื่อนที่ของหุ่นยนต์เทียบกับแผนที่ 40
รูปที่ 7 ปัญหาจราจรเมื่อหุ่นยนต์สองตัวทำงานบนเส้นทางที่ทับกัน 41
รูปที่ 8 การแก้ปัญหาจราจรด้วยการกำหนดเส้นทางเดินทิศทางเดียว 41
รูปที่ 9 เหตุการณ์ Deadlock ของหุ่นยนต์ 4 ตัวที่ใช้เส้นทางพร้อมกัน 41
รูปที่ 10 เวลาที่ใช้ในการทำงานของระบบหุ่นยนต์หลายตัวแบบ FIFO เปรียบเทียบกับวิธี Market-based Approach 42
รูปที่ 11 เวลาการทำงานของระบบจัดการหุ่นยนต์ที่จำนวนชุดคำสั่งเท่ากัน เมื่อจำนวนหุ่นยนต์ในระบบเพิ่มขึ้น 42

สัญลักษณ์และคำย่อ

AGV	Automated Guided Vehicles
$FIFO$	First In First Out
G	กราฟที่มีค่าถ่วงน้ำหนัก
GA	Genetic Algorithm
J_m	เซตของงาน
M	จำนวนครั้งของการเคลื่อนที่แบบเชิงเส้น
N	จำนวนครั้งของการเคลื่อนที่แบบเชิงมุม
R_n	หุ่นยนต์ในระบบตัวที่ n
ROS	Robot Operating System
S	เซตของจุดซึ่งเก็บค่าจุดที่เคยเดินผ่านมาแล้ว
T	เวลาในการทำงานของหุ่นยนต์
T_m	เวลาที่ถูกประเมินสำหรับงาน
$Task_i$	งานลำดับที่ i
V	ความเร็วเชิงเส้นของหุ่นยนต์
V_n	จุดต่าง ๆ ของกราฟถ่วงน้ำหนักมีค่าถ่วงน้ำหนักเป็นบวก
WQS	Weighted Quality Score
$W(u_i, v_j)$	ค่าถ่วงน้ำหนัก
Z	จุดปลายทาง
a	จุดเริ่มต้น
i	ลำดับการมอบหมายงาน
m	งานที่ m
n	หุ่นยนต์ตัวที่ n
s	ผลรวมของระยะทางที่สั้นที่สุดของหุ่นยนต์
u_i	พิกัดปัจจุบัน
v_j	พิกัดถัดไป
ω	ความเร็วเชิงมุมของหุ่นยนต์
x_i	ตำแหน่งปัจจุบันของพิกัด x
x_j	ตำแหน่งถัดไปของพิกัด x
y_i	ตำแหน่งปัจจุบันของพิกัด y

y_j	ตำแหน่งถัดไปของพิกัด y
(x, y)	ระบบพิกัด x,y
θ	ผลรวมของมุมการเคลื่อนที่ของหุ่นยนต์

รายการผลงานที่ตีพิมพ์และการประชุมวิชาการ

1. วารสารวิชาการ

   พงศกร ชาญชัยชูจิต และ พฤทธิกร สมิตไมตรี, “ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงาน อุตสาหกรรม,” วารสารวิชาการพระจอมเกล้าพระนครเหนือ, ปีที่ 29, ฉบับที่ 3, ก.ค.–ก.ย. 2562.

สำเนาต้นฉบับที่ได้รับการยินยอมจากผู้พิมพ์ผลงาน

ที่ ศธ ๐๕๒๕.๑(๑๙)/๒๘๑

กองส่งเสริมวิชาการ สำนักงานอธิการบดี
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
๑๕๑๘ ถนนประชาราษฎร์ ๑ แขวงวงศ์สว่าง
เขตบางซื่อ กรุงเทพมหานคร ๑๐๘๐๐

๑๘ ธันวาคม ๒๕๖๑

เรื่อง รับบทความเพื่อตีพิมพ์ลงในวารสารวิชาการพระจอมเกล้าพระนครเหนือ

เรียน รองศาสตราจารย์ ดร.พฤทธิกร สมิตไมตรี

ตามที่ท่านได้ส่งบทความวิจัย เรื่อง ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม เพื่อเสนอพิจารณาลงตีพิมพ์ในวารสารวิชาการพระจอมเกล้าพระนครเหนือนั้น กองบรรณาธิการวารสารได้พิจารณายอมรับบทความดังกล่าวและให้จัดพิมพ์ลงในวารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ ๒๙ ฉบับที่ ๓ สิงหาคม – ตุลาคม ๒๕๖๒

จึงเรียนมาเพื่อโปรดทราบ และหวังเป็นอย่างยิ่งว่าจะได้รับความร่วมมือจากท่านในการส่งบทความเพื่อพิมพ์ลงในวารสารวิชาการพระจอมเกล้าพระนครเหนือต่อไป

ขอแสดงความนับถือ

(รองศาสตราจารย์ ดร.มนตรี ศิริปรัชญานันท์)
ผู้ช่วยอธิการบดีฝ่ายส่งเสริมวิชาการและกิจการพิเศษ
รองหัวหน้ากองบรรณาธิการวารสาร

กองส่งเสริมวิชาการ
โทรศัพท์ ๐ ๒๕๕๕ ๒๒๓๘
โทรสาร ๐ ๒๕๕๕ ๒๒๓๖

บทนำ

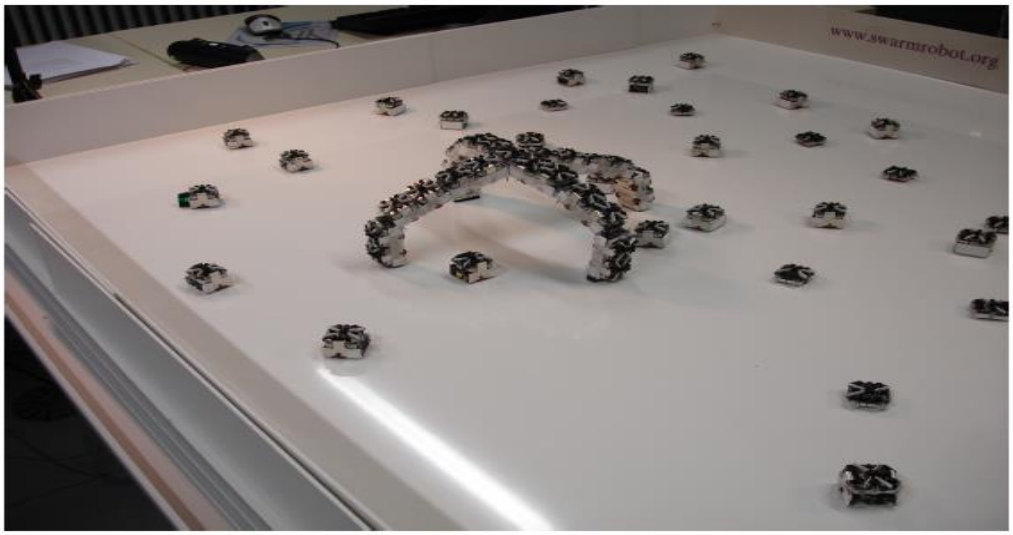
ระบบการขนถ่ายวัสดุ [1] (material handling system) เป็นลำดับขั้นตอนของกระบวนการนำปัจจัยหรือวัตถุดิบเข้ามาในกระบวนการผลิตแล้วมีการเปลี่ยนแปลงหรือแปรสภาพในด้านต่างๆ เช่น กระบวนการผลิตในโรงงาน การขนส่ง การเก็บเข้าคลังสินค้า การค้าปลีก การค้าส่ง การติดต่อสื่อสาร และสุดท้ายเกิดเป็นผลผลิต (output) ผลผลิตที่ได้จากระบบการผลิตมีมูลค่ามากกว่าปัจจัยที่ถูกนำเข้า เพราะได้ผ่านกระบวนการแปลงสภาพ ผลผลิตสินค้า (goods) และบริการ (service) โดยภายในกระบวนการแปลงสภาพปัจจัยที่ถูกนำเข้าให้กลายเป็นสินค้าและบริการดังกล่าว เมื่อพิจารณาในแง่ของการทำงานภายในโรงงานอุตสาหกรรม มักเกี่ยวข้องกับการเคลื่อนที่ การเคลื่อนย้ายวัสดุต่างๆ ที่ใช้ในกระบวนการผลิตทั้ง 3 องค์ประกอบ คือ ปัจจัยนำเข้า (input) กระบวนการแปลงสภาพ (conversion process) และผลผลิต (output) โดยการเคลื่อนย้ายดังกล่าวเรียกกันว่า การขนถ่ายวัสดุ (material handling) ที่จะลำเลียงปัจจัยการผลิตไปยังส่วนต่างๆ ทั่วทั้งกระบวนการผลิต สำหรับการพัฒนางานทางด้านการขนถ่ายวัสดุยังคงเป็นเรื่องที่น่าสนใจและมีความสำคัญต่อระบบการผลิตในอุตสาหกรรม เพราะการขนถ่ายวัสดุที่ดีนั้นนำมาซึ่งการเพิ่มกำลังการผลิตสินค้า รวมถึงสามารถลดต้นทุนให้กับโรงงานอุตสาหกรรมได้

ระบบหุ่นยนต์หลายตัว [2] เป็นการนำหุ่นยนต์มากกว่าหนึ่งตัวมาทำงานร่วมกันภายใต้สภาวะแวดล้อมต่างๆ ที่อาจจะเป็นหุ่นยนต์ที่เหมือนหรือต่างกันก็ได้ ระบบหุ่นยนต์หลายตัวเริ่มมีการวิจัยมาตั้งแต่ช่วงปี ค.ศ.1980 แต่เนื่องจากเทคโนโลยีที่ถูกจำกัดด้วยความเร็วในการประมวลผลของคอมพิวเตอร์ทำให้ระบบหุ่นยนต์หลายตัวจึงไม่เป็นที่แพร่หลายมากนักจนกระทั่งเมื่อเวลาผ่านไปคอมพิวเตอร์เริ่มทำงานได้เร็วขึ้น มีขนาดเล็กลงรวมถึงงานต่างๆ ในปัจจุบันเริ่มเข้าสู่ยุคอุตสาหกรรม 4.0 เป็นโลกที่นำข้อมูลจำนวนมากมาใช้ประโยชน์มากขึ้นในทุกด้าน อีกทั้งในบางกรณีหุ่นยนต์ที่ทำงานตัวเดียวไม่สามารถแก้ปัญหาที่ซับซ้อนได้ จึงเกิดแนวคิดในการนำหุ่นยนต์หลายตัวมาทำงานร่วมกัน มีทั้งช่วยลดเวลาการทำงานโดยการเพิ่มปริมาณหุ่นยนต์ การแชร์ข้อมูลสภาพแวดล้อมของหุ่นยนต์แต่ละตัวให้แก่กันเพื่อใช้ในการตัดสินใจทำงาน เป็นต้น

ตัวอย่างการทำงานของหุ่นยนต์หลายตัว เช่น การยกวัตถุที่หนักมากๆ โดยที่หุ่นยนต์ตัวเดียวไม่สามารถทำได้หรือทำได้แต่ต้องใช้ตัวทำงานที่มีกำลังสูงมาก ซึ่งอาจจะต้องแลกมาด้วยการลงทุนที่ค่อนข้างสูงจนทำให้คุ้มค่าต่อการลงทุน นอกจากนี้การที่หุ่นยนต์หลายตัวในลักษณะเดียวกัน ทำงานด้วยกัน หากมองดูจะพบว่ามีความซ้ำซ้อนที่มีประโยชน์ในบางเรื่อง เช่น การทดแทนซึ่งกันและกันในกรณีที่มีตัวใดตัวหนึ่งที่ไม่สามารถทำงานได้ หุ่นยนต์ตัวอื่นในระบบสามารถปรับงานของตัวเองเพื่อทำงานทดแทนหุ่นยนต์ที่เกิดความเสียหายได้ เพื่อให้งานนั้นๆ ยังคงสามารถดำเนินต่อไปได้แม้เกิดข้อผิดพลาดกับหุ่นยนต์บางตัวในระบบ

ระบบหุ่นยนต์หลายตัวในการทำงานร่วมกัน งานอาจจะแตกต่างกัน เพื่อประโยชน์ในการทำภารกิจ เช่น ในการต่อสู้หรือสงคราม อาจมีหุ่นยนต์ทางอากาศและภาคพื้นดินทำงานร่วมกัน โดยหุ่นยนต์ทางอากาศคอยสอดแนมและส่งข้อมูลไปยังหุ่นยนต์ภาคพื้นดิน หรือแม้กระทั่งในกรณีที่หุ่นยนต์เกิดความเสียหาย ก็สามารถให้หุ่นยนต์ตัวอื่นคอยซ่อมแซมระบบให้กับหุ่นยนต์ที่ได้รับความเสียหายให้สามารถกลับมาทำงานได้ปกติ

หุ่นยนต์ที่สามารถประกอบตัวใหม่ได้ด้วยตัวเอง (self-reconfigurable robots) เป็นระบบหุ่นยนต์หลายตัวแบบชนิดเดียวกันที่พิเศษแบบหนึ่ง ลักษณะการทำงานของหุ่นยนต์ชนิดนี้จะสามารถต่อกันเป็นรูปทรงหรือประกอบเป็นรูปร่างได้ มันสามารถแบ่งตัวเองออกหรือรวมตัวกันใหม่ตามงานที่ได้รับมอบหมาย ตัวอย่างหุ่นยนต์ชนิดนี้แสดงดังรูปที่ 1



รูปที่ 1 แสดงการควบคุมหุ่นยนต์ที่สามารถประกอบตัวใหม่ได้ด้วยตัวเอง
[ภาพจาก: Ondrej Stanek, David Obdrálek. Centralized multi-robot system. The Department of Software Engineering; 2012:4-12]

โดยทั่วไประบบควบคุมหุ่นยนต์หลายตัวแบ่งออกได้ 2 ประเภท คือ แบบควบคุมโดยระบบศูนย์กลาง และแบบควบคุมโดยระบบไม่มีศูนย์กลาง

ระบบศูนย์กลางหุ่นยนต์หลายตัว เป็นระบบที่รับรู้ข้อมูลในระบบทั้งหมด โดยทำการเก็บรวบรวมข้อมูลทั้งหมดจากหุ่นยนต์ในระบบ หรือจากตัวตรวจรู้ต่างๆ ที่อยู่ในระบบ จากนั้นนำมาประมวลผลเพื่อจัดการงานและสั่งให้หุ่นยนต์ทำงานตามแผนกำหนดไว้ ระบบการทำงานแบบนี้เป็นแบบตรงไปตรงมา อย่างไรก็ตามมักจะประสบปัญหาที่คาดไม่ถึงได้ง่าย เช่น จากการสื่อสาร ซึ่งโดยปกติแล้วระบบแบบนี้มักใช้กับหุ่นยนต์ที่มีจำนวนจำกัดแน่นอนและมีสภาพแวดล้อมที่เป็นปกติและไม่มีการเปลี่ยนแปลง ตัวอย่างการใช้งานระบบนี้ เช่น การใช้หุ่นยนต์ในโรงพยาบาล Nemocnice Na Homolce [3] ดังรูปที่ 2 เป็นการขนส่งอาหาร ผ้าสำหรับผู้ป่วยภายใน

โรงพยาบาล ด้วยการนำทางโดยใช้สัญลักษณ์บนพื้น ประตู และลิฟต์ เพื่อควบคุมการทำงานของหุ่นยนต์ในโรงพยาบาล

รูปที่ 2 แสดงการใช้งานระบบศูนย์กลางของหุ่นยนต์หลายตัวในโรงพยาบาล
[ภาพจาก: Ondrej Stanek, David Obdrálek. Centralized multi-robot system. The Department of Software Engineering; 2012:4-12]

ระบบไม่มีศูนย์กลางของหุ่นยนต์หลายตัว มีตัวควบคุมเพื่อสั่งงานหุ่นยนต์ แต่ละตัวจะทำงานโดยอาศัยสภาวะแวดล้อมที่เปลี่ยนไปมาเป็นประโยชน์ในการทำภารกิจ นอกจากนี้ยังใช้ข้อมูลจากหุ่นยนต์ตัวอื่นๆ ร่วมกันในกรณีที่ทำงานในสภาวะใกล้เคียงกันจะมีการแชร์ข้อมูลซึ่งกันและกัน โดยระบบนี้หุ่นยนต์ต้องมีความทนทานต่อการทำงานในสภาวะแวดล้อมต่างๆ ได้ดี ตัวอย่างของการใช้งานในระบบนี้ เช่น การนำมาใช้ในการสำรวจพื้นที่บริเวณชายแดน เป็นต้น

จากระบบควบคุมหุ่นยนต์ทั้งสองแบบที่กล่าวไปแล้วข้างต้น เมื่อนำระบบทั้งสองมารวมกันและนำไปใช้ร่วมกับการผลิตในอุตสาหกรรม รวมถึงมีการบริหารจัดการผ่านเครือข่ายอินเตอร์เน็ต เพื่อช่วยให้การผลิตเกิดความยืดหยุ่น ช่วยเพิ่มกำลังการผลิต ประสิทธิภาพและคุณภาพในการผลิต ระบบแบบนี้ถูกเรียกว่า ระบบการผลิตแบบยืดหยุ่น (Flexible manufacturing system, FMS) จึงเห็นได้ว่าการนำระบบหุ่นยนต์หลายตัวมาใช้งานในระบบการขนถ่ายวัสดุนั้นเป็นเรื่องที่ได้รับความสนใจและเป็นประโยชน์ต่อการนำไปใช้งานในภาคอุตสาหกรรมรวมถึงยังสอดคล้องกับระบบการผลิตแบบยืดหยุ่น อาจถือได้ว่างานวิจัยนี้เป็นการประยุกต์ความรู้จากสาขาวิชาต่างๆ ทางด้านวิศวกรรมหุ่นยนต์ การจัดการอุตสาหกรรมและการบริหารทรัพยากรมาบูรณาการเพื่อนำไปใช้ในการพัฒนาระบบการขนถ่ายวัสดุในอุตสาหกรรม นอกจากนี้ยังเป็นการพัฒนาเทคโนโลยีสำหรับการขนถ่ายวัสดุในประเทศให้มีขีดความสามารถในการแข่งขันกับต่างประเทศ รวมถึงเป็นการเตรียมความพร้อมการพัฒนาเพื่อเข้าสู่ยุคอุตสาหกรรม 4.0 ตามแผนการพัฒนาของประเทศได้

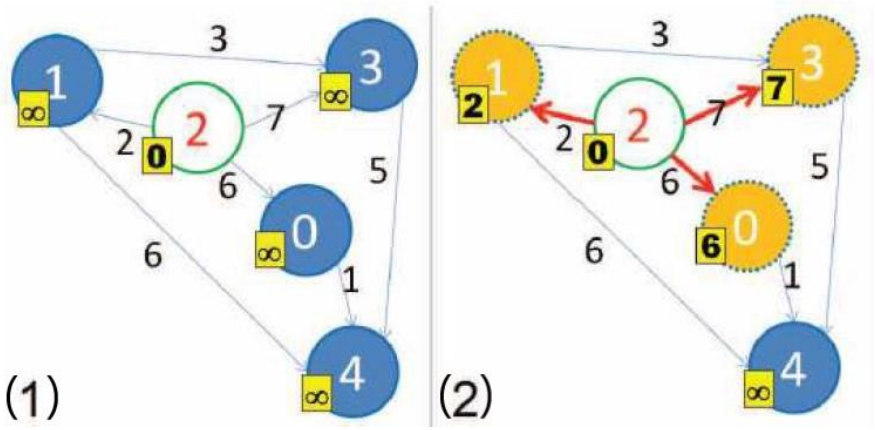
อุตสาหกรรมที่นำหุ่นยนต์ AGV (automated guided vehicles) มาใช้ทำงานแทนคน ทั้งแบบอัตโนมัติ แบบเดินตามแถบแม่เหล็ก หรือแบบใช้สัญญาณไร้สายในการระบุตำแหน่งของหุ่นยนต์ [4] ลักษณะการนำหุ่นยนต์มาทำงานร่วมในกระบวนการผลิตมักจะให้หุ่นยนต์ทำงานตามลำดับขั้นตอน FIFO (First In First Out) กล่าวคือ งานที่มาก่อนทำก่อน และงานที่มาหลังทำทีหลัง จนกระทั่งถึงงานลำดับสุดท้าย โดยไม่คำนึงถึงการเพิ่มประสิทธิภาพการขนถ่ายวัสดุ หากในระบบเพิ่มจำนวนหุ่นยนต์มาใช้ในการขนถ่ายวัสดุมากขึ้น ทำให้ประสิทธิภาพงานที่ได้ลดลง เนื่องจากขาดการวิเคราะห์เวลาที่ใช้ในการทำงานแต่ละงาน ซึ่งหากวิเคราะห์การจัดการงานของหุ่นยนต์จะพบว่า ยังสามารถเพิ่มการจัดการงานที่เหมาะสมเพื่อช่วยลดเวลาการทำงานของการขนย้ายวัสดุในภาพรวมได้อีก [5]

งานวิจัยเกี่ยวกับระบบการจัดการหุ่นยนต์หลายตัวถูกนำมาใช้เพื่อช่วยจัดการระบบให้มีประสิทธิภาพเพิ่มขึ้นในหลายด้าน เช่น การควบคุมกลยุทธ์สำหรับหุ่นยนต์หลายตัวส่งผลต่อการช่วยลดเวลาในการทำงาน [6] ปัญหาที่มักพบสำหรับวิธีการจัดเก็บและขนส่งวัตถุดิบทั่วไปอย่าง FIFO คือ ข้อจำกัดเรื่องการใช้เวลารวมมากเกินไปสำหรับการจัดการขนส่งหลายชุดคำสั่ง เพราะไม่ได้เปรียบเทียบและเลือกใช้หุ่นยนต์ที่เหมาะสมกับงานที่สุด ยกตัวอย่างเช่น หุ่นยนต์ที่ว่างงานอยู่ใกล้กับสถานีงานคำสั่งที่สองมากที่สุด แต่ต้องไปรับงานคำสั่งแรกตามวิธี FIFO ทำให้เสียเวลาในการเดินทาง เป็นต้น งานวิจัยนี้จึงมุ่งเน้นการออกแบบระบบการจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุด้วยวิธี Market-based approach โดยวิธีการนี้จะแปลงระบบเส้นทางให้อยู่ในรูปแบบตาราง จุดปลายของเส้นทางเปรียบเสมือนสินค้าที่มีการแลกเปลี่ยนอยู่ในตลาด และระยะทางระหว่างจุดเริ่มต้นไปยังจุดปลายทางเปรียบเสมือนต้นทุนที่ใช้ กำไรขึ้นอยู่กับข้อมูลต่างๆ ที่ใช้ประกอบไปจนถึงจุดปลายทาง งานวิจัยนี้ใช้เวลาในการทำงานของหุ่นยนต์เป็นตัวชี้วัดเพื่อเลือกงานให้เหมาะสมกับตำแหน่งปัจจุบันของหุ่นยนต์ โดยหุ่นยนต์ทุกตัวจะมีเส้นทางการเดินที่สั้นที่สุด ซึ่งในกรณีนี้วิธี Market-based approach ก็คือการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน

วิธีหาเส้นทางแบบกริดที่สั้นที่สุดนั้นมีอยู่หลายอัลกอริทึม เช่น ขั้นตอนวิธีของฟลอยด์-วอร์แชล (Floyd–Warshall algorithm) ใช้แก้ปัญหาวิถีสั้นสุดแบบทุกคู่ ขั้นตอนวิธีของเบลแมน-ฟอร์ด (Bellman-Ford algorithm) ใช้แก้ปัญหาวิถีสั้นสุดแบบแหล่งต้นทางเดียว โดยที่น้ำหนักของเส้นเชื่อมอาจเป็นลบได้ ขั้นตอนวิธีแบบเอสตาร์ (A* algorithm) ใช้แก้ปัญหาวิถีสั้นสุดแบบแหล่งต้นทางเดียว โดยใช้ค่าน้ำหนักของจุดปัจจุบันกับจุดเป้าหมายเพื่อเพิ่มความเร็วในการแก้ปัญหา ขั้นตอนวิธีของไดก์สตรา (Dijkstra's algorithm) ใช้แก้ปัญหาวิถีสั้นสุดแบบแหล่งต้นทางเดียว โดยที่น้ำหนักของเส้นเชื่อมต้องไม่เป็นลบเป็นต้น งานวิจัยนี้ได้เลือกใช้ขั้นตอนวิธีของ

ไดก์สตรา (Dijkstra's algorithm) โดยขั้นตอนวิธีนี้ถูกคิดค้นขึ้นโดยนักวิทยาการคอมพิวเตอร์ชาวดัตช์นามว่า แอ็ดส์เคอร์ ไดก์สตรา (Edsger Dijkstra) ในปี 1959 เพื่อแก้ไขปัญหาวิถีสั้นสุดจากจุดหนึ่งใดๆ สำหรับกราฟที่มีความยาวของเส้นเชื่อมไม่เป็นลบ ขั้นตอนวิธีนี้หาระยะทางสั้นที่สุดจากจุดหนึ่งไปยังจุดใดๆ ในกราฟจากเส้นทางที่สั้นที่สุดไปทีละจุดต่อกันไปเรื่อยๆ จนถึงจุดหมายตามที่ต้องการ ขั้นตอนวิธีของไดค์สตรามีขั้นตอนย่อยดังนี้ [7]

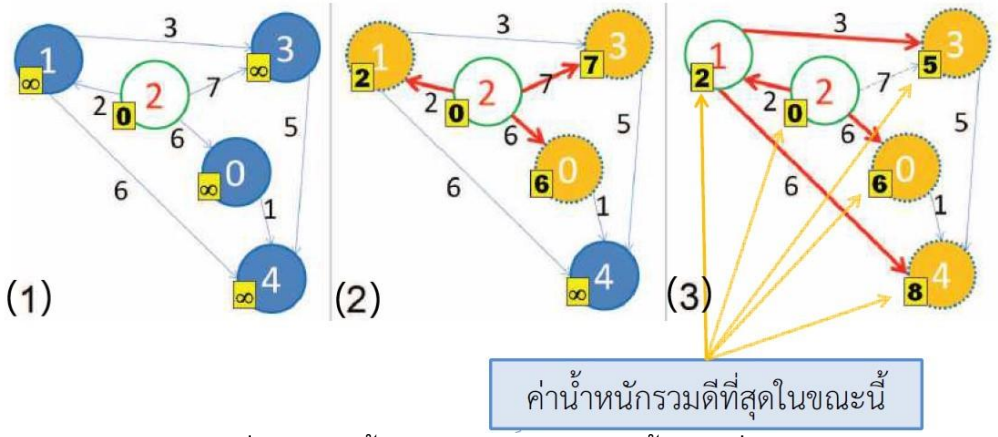
1. จากจุดเริ่มต้น (โหนด 2 ในรูปที่ 3) โดยกำหนดให้ ณ ขณะนี้มีโหนดในกลุ่มเพียงโหนดเดียวและมีค่าน้ำหนักในการไปถึงโหนดตัวเองเท่ากับศูนย์ โหนดอื่นๆ ที่ยังไม่ถูกแวะเยี่ยมถือว่ามีระยะทาง จากโหนดเริ่มต้นเป็นอนันต์ (ค่านี้จะถูกเปลี่ยนเป็นค่าจริงเมื่อโหนดถูกแวะเยี่ยม) ดังแสดงในภาพหมายเลข (1) จากรูปที่ 3
2. จากโหนดเริ่มต้นให้สำรวจเส้นเชื่อมทั้งหมดแล้วเก็บไว้ในลิสต์ จากนั้นมองหาเส้นเชื่อมที่มีค่าน้ำหนักน้อยที่สุดในลิสต์ (ได้เส้นเชื่อมที่ต่อไปโหนด 1) ดังแสดงในภาพหมายเลข (2) จากรูปที่ 3



รูปที่ 3 แสดงขั้นตอนวิธีของไดค์สตราขั้นตอนที่ (1) และ (2)
[ภาพจาก: ภิญโญ แท้ประสาทสิทธิ์. Graph Algorithm: Data Structure, Algorithms, and Applications. September 6, 2014:67-70]

3. เลือกโหนดที่ต่อกับเส้นเชื่อมนั้นเข้ามาในกลุ่ม (คือโหนดสีขาวทั้งหมด) ให้ถือว่าโหนดพวกนี้ถูกแวะเยี่ยมและได้ค่าน้ำหนักรวมเท่ากับค่าน้ำหนักจากโหนดที่มาถึงมันบวกกับเส้นเชื่อมที่ใช้ ดังนั้นค่าน้ำหนักของโหนด 1 จึงเป็น 0 + 2 = 2 เติมเส้นเชื่อมของโหนด 1 เข้าไปในลิสต์ (เส้นเชื่อมที่มีสีแดงคืออยู่ในลิสต์) ดังแสดงในภาพหมายเลข (2) จากรูปที่ 3

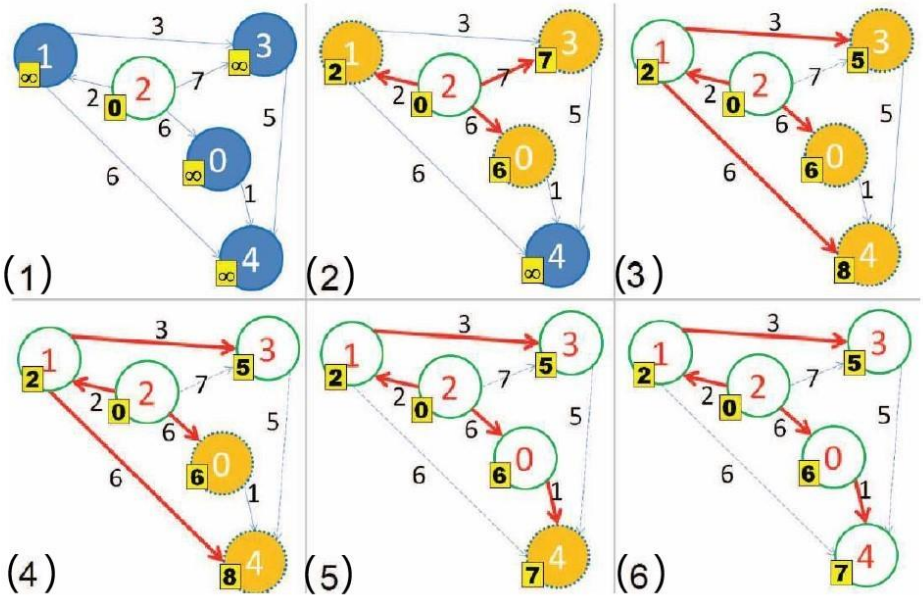
4. เลือกเส้นเชื่อมที่สร้าง**ค่าน้ำหนักรวมน้ำน้อยที่สุด** และต้องเชื่อมไปยังโหนดนอกกลุ่ม ดังแสดงในภาพหมายเลข (3) จากรูปที่ 4



รูปที่ 4 แสดงขั้นตอนวิธีของไดค์สตราขั้นตอนที่ (3)

[ภาพจาก: ภิญโญ แท้ประสาทสิทธิ์. Graph Algorithm: Data Structure, Algorithms, and Applications. September 6, 2014:67-70]

5. เลือกโหนดและขยายลิสต์ของเส้นเชื่อมต่อไปเรื่อยๆ ในทำนองเดิม จนกว่าจะไม่สามารถเพิ่มโหนดใหม่เข้ามาในกลุ่มได้อีก แสดงดังรูปที่ 5



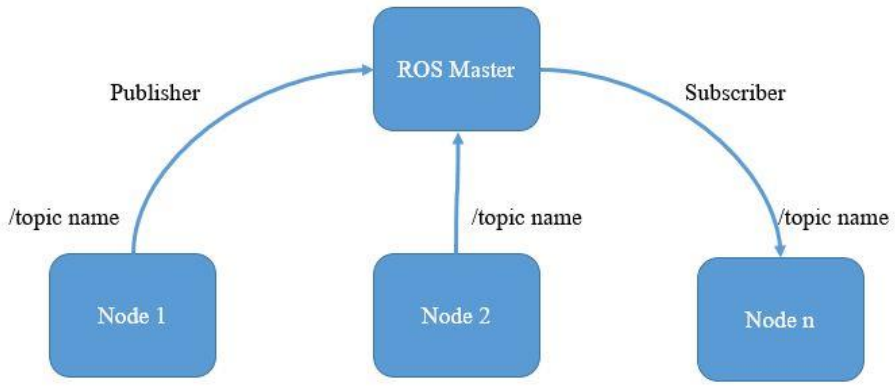
รูปที่ 5 แสดงขั้นตอนวิธีของไดค์สตราในขั้นตอนขยายลิสต์

[ภาพจาก: ภิญโญ แท้ประสาทสิทธิ์. Graph Algorithm: Data Structure, Algorithms, and Applications. September 6, 2014:67-70]

สำหรับเหตุผลที่เลือกใช้ขั้นตอนวิธีของไดก์สตราจากขั้นตอนวิธีทั้งหมด เพราะเข้ากับรูปแบบของระบบขนส่งวัสดุด้วยหุ่นยนต์หลายตัว รวมถึงเวลาที่ใช้ในการประมวลหาระยะทางที่สั้นที่สุดก็อยู่ในระดับที่ไม่นานจนเกินไป และเมื่อมีจำนวนโหนดในการประมวลผลมากขึ้น ระยะเวลาในการคำนวณก็ไม่ได้เพิ่มขึ้นมากนัก เมื่อทำการเปรียบเทียบกับอัลกอริทึมเบลแมน-ฟอร์ด [8] ผลลัพธ์จากการหาระยะทางที่สั้นที่สุดเมื่อเทียบกับ Genetic Algorithm (GA) มีค่าเท่ากัน [9] สำหรับวิธีของไดก์สตรานั้นเป็นวิธีการหาเส้นทางที่สั้นที่สุด โดยมีงานวิจัยที่นำวิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน ร่วมกับอัลกอริทึมการหาเส้นทางที่สั้นที่สุด เช่น การจัดสรรงานหุ่นยนต์หลายตัวสำหรับค้นหาและช่วยเหลือ [10] โดยใช้ค่าน้ำหนักระหว่างโหนด เมื่อมีการนำค่าน้ำหนักระหว่างโหนดแทนด้วยระยะทางระหว่างโหนด จึงสามารถนำมาใช้ในการหาระยะทางที่สั้นที่สุดของหุ่นยนต์ได้ [11] แต่ระยะทางที่ใช้ในการเดินทางที่สั้นที่สุดนั้นไม่เพียงพอต่อการนำไปตัดสินใจเลือกหุ่นยนต์ งานวิจัยนี้จึงนำเรื่องของเวลาขนถ่ายวัสดุมาช่วยในการตัดสินใจ เพราะ ทำให้เงื่อนไขการตัดสินใจนั้นขึ้นอยู่กับตัวแปรสองตัว คือ ความเร็วในการทำงานและระยะทางของหุ่นยนต์

ในการพัฒนาระบบจัดการหุ่นยนต์หลายตัวพบว่ามีความยุ่งยากในเรื่องของการสื่อสาร การจัดการทรัพยากรของระบบจัดการหุ่นยนต์ การสื่อสารผ่านระบบเครือข่าย รวมถึงหากต้องการควบคุมหุ่นยนต์ที่แต่ละตัวมีลักษณะแตกต่างกันยิ่งเพิ่มความซับซ้อนให้กับระบบควบคุมมากยิ่งขึ้น แต่ในปัจจุบันมีการพัฒนาซอฟต์แวร์ที่ใช้ในการสื่อสารกันแบบ machine-to-machine นั่นคือ ROS (Robot Operating System)

โปรแกรม ROS [12] ทำงานเสมือนตัวจัดการข้อมูลของตัวตรวจรู้ต่างๆ ในระบบที่สร้างขึ้น มันจึงถูกเรียกว่า ระบบปฏิบัติการ (Operating System) แต่ตามความจริง ROS เป็นแหล่งรวมไลบารี่และเครื่องมือต่างๆ ที่สามารถช่วยเหลือให้ผู้พัฒนาซอฟแวร์ ได้สร้างโปรแกรมสำหรับหุ่นยนต์หรือระบบหุ่นยนต์ นอกจากนี้ ROS ยังเป็นโอเพ่นซอร์ส (open source) ภายใต้เงื่อนไข BSD licenses



รูปที่ 6 หลักการทำงานของ ROS

ข้อดีของโปรแกรม ROS คือ มีความยืดหยุ่นในการควบคุมหุ่นยนต์ต่างชนิดหลายตัวพร้อมกันได้ รวมถึงเป็นโอเพ่นซอร์สที่สามารถใช้งานได้ฟรี และถูกขยายผลไปใช้งานในอุตสาหกรรมมากขึ้นในปัจจุบัน [13]

เมื่อนำองค์ประกอบของความรู้ที่กล่าวไปทั้งหมดมารวมกันจะได้เป็น ระบบการจัดการหุ่นยนต์หลายตัวโดยใช้วิธีการเลือกงานให้กับหุ่นยนต์แต่ละตัวด้วยขั้นตอนการหาระยะทางที่สั้นที่สุดของไดก์สตราร่วมกับการหาความคุ้มค่าของระยะทางในการทำงานของหุ่นยนต์แต่ละตัวด้วยวิธีการหาเวลาที่น้อยที่สุด (Market base approach) ทำให้สามารถกำหนดงานของหุ่นยนต์แต่ละตัวที่แตกต่างกันได้ ซึ่งการควบคุมหุ่นยนต์ที่แตกต่างกันหลายตัว ค่อนข้างซับซ้อนรวมถึงการสื่อสารของหุ่นยนต์กับตัวควบคุมทำได้ยาก จึงมีการนำโปรแกรมของ ROS มาช่วยในการจัดการในเรื่องการสื่อสาร ด้วยองค์ความรู้นี้จึงทำให้เกิดงานวิจัยระบบจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุในโรงงานอุตสาหกรรมนี้ขึ้นตามรายละเอียดในหัวข้อถัดไป

วัตถุประสงค์

1. ออกแบบและสร้างระบบการจัดการงานสำหรับหุ่นยนต์หลายตัวด้วยวิธีการหาเส้นทางที่สั้นที่สุดแบบ Dijkstra's algorithm ร่วมกับวิธี Market based approach
2. ทดสอบระบบการจัดการงานที่สามารถทำงานได้อย่างยืดหยุ่นในพื้นที่โรงงานอุตสาหกรรมจำลอง

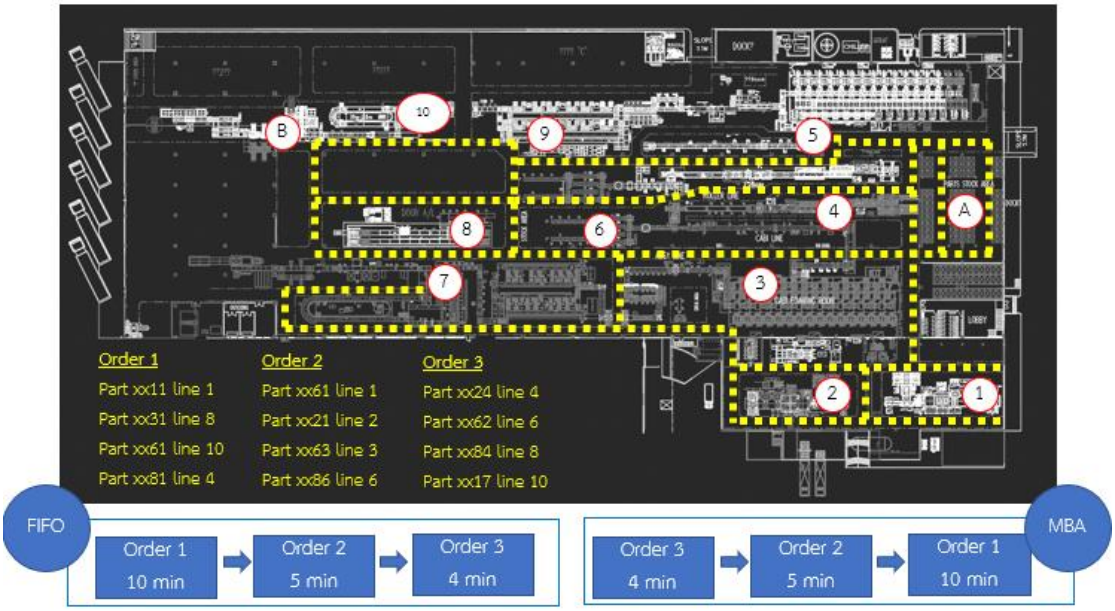
ขอบเขตของโครงงานวิจัย

1. ระบบการจัดการสามารถควบคุมหุ่นยนต์ได้มากกว่าหนึ่งตัวสำหรับขนถ่ายสินค้าในพื้นที่โรงงานอุตสาหกรรมจำลองขนาดกว้าง 5 เมตร ยาว 13 เมตร
2. ระบบมีความยืดหยุ่นในการควบคุมหุ่นยนต์ต่างชนิดกัน

ที่มาและความสำคัญของงานวิจัย

โรงงานอุตสาหกรรมการผลิตส่วนใหญ่มักจะใช้วิธีการของ FIFO สำหรับจัดการงานโดยหลักการดังกล่าวจะทำงานตามลำดับงาน คือ งานใดมาก่อนทำก่อน งานใดมาหลังทำหลัง ซึ่งหากเป็นงานที่เราทราบจำนวนของงานที่ต้องการผลิตแล้วนำวิธีของ FIFO มาจัดการงานจะพบว่าขาดประสิทธิภาพในการทำงาน จากการขาดการวิเคราะห์เวลาในแต่ละงานที่ทำ ซึ่งหากนำมาจัดการด้วยวิธีใหม่จะพบว่าสามารถจัดลำดับงานใหม่ที่ใช้เวลาทำงานจากน้อยไปมากได้ เช่น จากตัวอย่างรูปที่ 7 เป็นแผนผังโรงงานในอุตสาหกรรมหนึ่งที่มีลำดับการขนถ่ายวัสดุ 3 งาน คือ งานที่ 1 มีการขนถ่ายวัสดุ 4 ชิ้นไปยังสายงานผลิต 4 สายงาน คือ part xx11 ส่งไปยังสายการผลิตที่ 1, part xx31 ส่งไปยังสายการผลิตที่ 8, part xx61 ส่งไปยังสายการผลิตที่ 10

และ part xx81 ส่งไปยังสายการผลิตที่ 4 ใช้เวลาในการขนถ่ายวัสดุสำหรับงานที่ 1 ทั้งหมด 10 นาที งานที่ 2 part xx61 ส่งไปยังสายการผลิตที่ 1, part xx21 ส่งไปยังสายการผลิตที่ 2, part xx63 ส่งไปยังสายการผลิตที่ 3 และ part xx86 ส่งไปยังสายการผลิตที่ 6 ใช้เวลาในการขนถ่ายวัสดุสำหรับงานที่ 2 ทั้งหมด 5 นาที งานที่ 3 part xx24 ส่งไปยังสายการผลิตที่ 4, part xx62 ส่งไปยังสายการผลิตที่ 6, part xx84 ส่งไปยังสายการผลิตที่ 8 และ part xx17 ส่งไปยังสายการผลิตที่ 10 ใช้เวลาในการขนถ่ายวัสดุสำหรับงานที่ 3 ทั้งหมด 4 นาที



รูปที่ 7 แสดงเวลาการจัดการงานสำหรับขนถ่ายวัสดุแบบ FIFO และ Market based approach

จากรูปพบว่าหากใช้การจัดการงานตามลำดับด้วยวิธีการแบบ FIFO การทำงานในแต่ละงานไม่ได้ถูกจัดเรียงตามเวลาที่น้อยที่สุดไปทำก่อน แต่ให้ความสำคัญกับลำดับของงานที่มาก่อนทำก่อน แต่หากเรานำวิธีการของ Market-based approach โดยใช้เวลาของการทำงานมาเป็นเงื่อนไข สำหรับกรณีที่การผลิตรู้คำสั่งงานมากกว่าหนึ่งชุด พบว่าสามารถจัดลำดับการทำงานใหม่โดยนำงานที่ใช้เวลาน้อยที่สุดมาทำก่อน จะได้ลำดับการขนถ่ายวัสดุสินค้าใหม่จากงานที่ 1 งานที่ 2, งานที่ 3, เป็น งานที่ 3, งานที่ 2, และงานที่ 1 ตามลำดับ แต่สำหรับการจัดสมดุลการผลิตแล้วพบว่าการเรียงลำดับงานใหม่ไม่ได้ช่วยให้การทำงานทำได้เร็วขึ้น เพราะสังเกตได้ว่างานที่ใช้เวลาในการทำงานมากที่สุดคือ งานที่ 1 ใช้เวลาทำงาน 10 นาที ดังนั้นหากใช้วิธีการจัดลำดับงานจากการใช้เวลาน้อยไปมากเพียงอย่างเดียวไม่เพียงพอ จึงได้นำวิธีการหาเส้นทางที่สั้นที่สุดมาประยุกต์ร่วมกับการนำหุ่นยนต์หลายตัวทำงานร่วมกันมารวมเข้ากับวิธีการแบบ FIFO และวิธีการแบบ Market based approach เพื่อเปรียบเทียบว่าวิธีใดสามารถช่วยลดเวลาในการขนถ่ายวัสดุได้ดีกว่า

ระเบียบวิธีวิจัย

โปรแกรมในการควบคุมระบบหุ่นยนต์หลายตัวถูกเขียนใน ROS เพื่อจัดการงานและควบคุมหุ่นยนต์ทั้งหมด เมื่อโปรแกรมพัฒนาเสร็จแล้วจึงถูกทดสอบและปรับปรุงดังนี้

1. ทดสอบการทำงานของระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุซึ่งประกอบด้วย 5 ส่วน ดังนี้
   1.1. ส่วนการรับคำสั่งงาน
   1.2. ส่วนการหาเส้นทางที่สั้นที่สุดให้กับหุ่นยนต์แต่ละตัว
   1.3. ส่วนการจัดการลำดับของงาน
   1.4. ส่วนการจัดการเส้นทางของหุ่นยนต์
   1.5. ส่วนการติดตามสถานะงานและหุ่นยนต์
   โดยทั้ง 5 ส่วนได้ทำงานเชื่อมต่อและสัมพันธ์กันตามรูปที่ 8
2. เปรียบเทียบความแตกต่างของเวลาระหว่างระบบการจัดงานหุ่นยนต์หลายตัวแบบ FIFO และ Market based approach

รูปที่ 8 ภาพรวมของระบบจัดการงาน

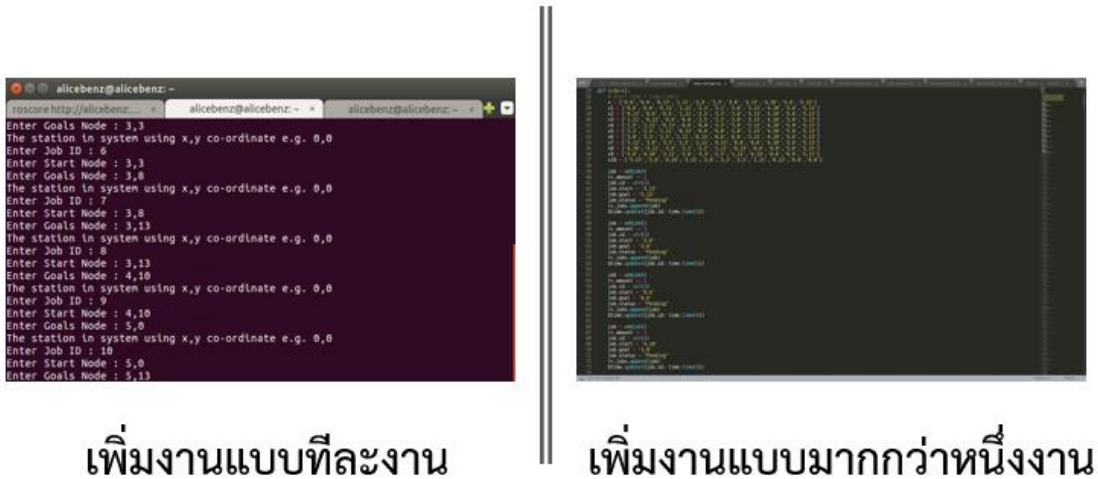
การทำงานของโปรแกรมส่วนย่อยต่างๆ มีรายละเอียดดังนี้

1.1 ส่วนการรับคำสั่งงาน

การรับคำสั่งงานสามารถทำได้ 2 รูปแบบ คือ

1. ป้อนคำสั่งงานทีละคำสั่งให้กับระบบ โดยสามารถป้อนคำสั่งงานประกอบด้วยข้อมูลดังนี้ ชื่องาน สถานีเริ่มงาน และสถานีปลายทาง
2. การเพิ่มงานแบบเป็นไฟล์ สามารถเพิ่มคำสั่งงานได้มากกว่าหนึ่งงานแต่ละงานประกอบไปด้วยข้อมูล ดังนี้ จำนวนงานทั้งหมด ชื่องาน สถานีเริ่มงานและสถานีปลายทาง

วิธีการรับคำสั่งงานของระบบสามารถแสดงได้ดังรูปที่ 9

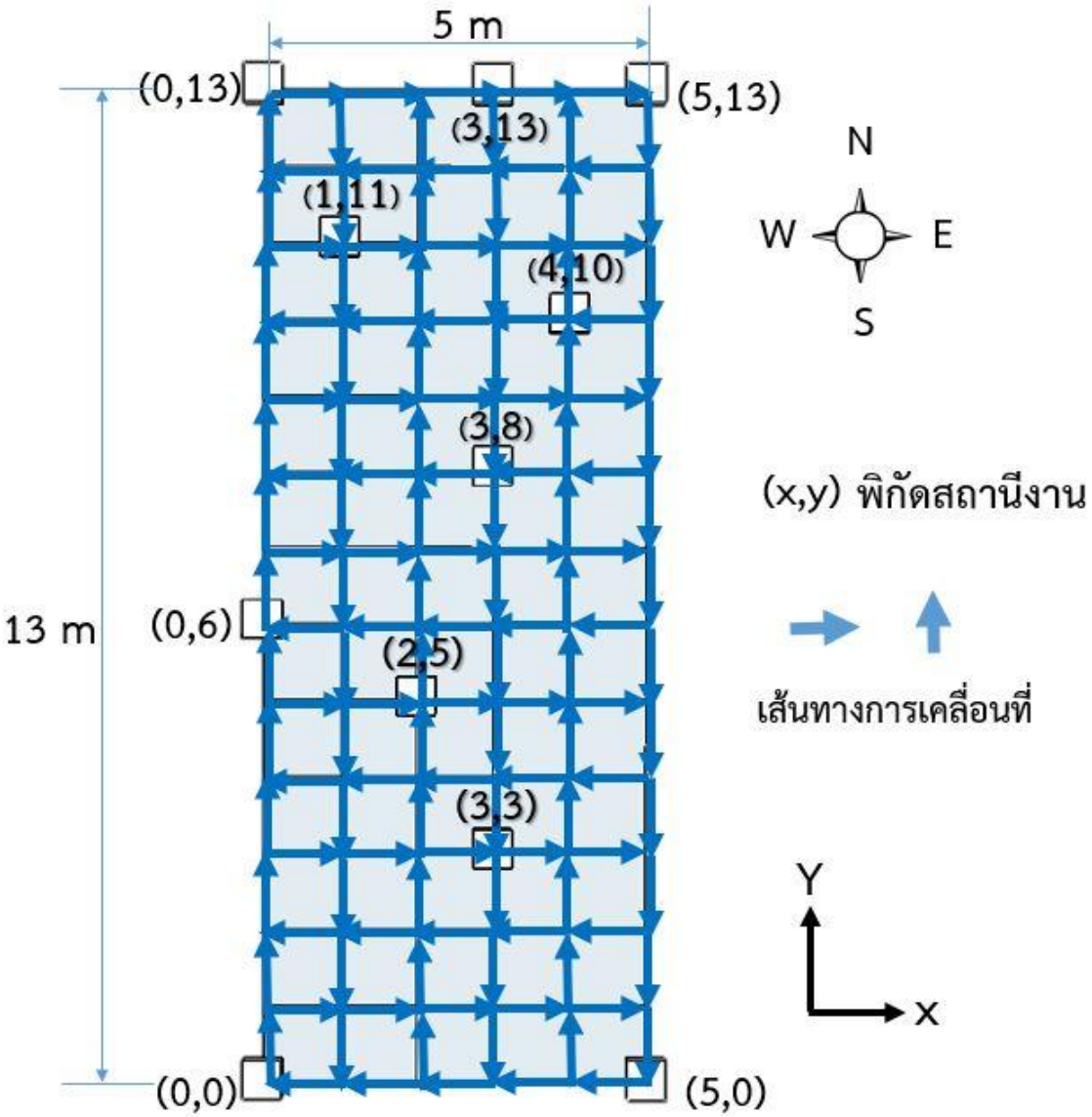


รูปที่ 9 แสดงรูปแบบการเพิ่มงานเข้าไปยังระบบการจัดการงาน

1.2 ส่วนการหาเส้นทางที่สั้นที่สุดให้กับหุ่นยนต์แต่ละตัว

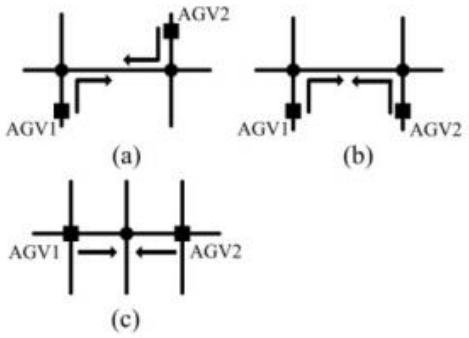
ขั้นตอนวิธีของไดก์สตรา (Dijkstra's algorithm) แก้ปัญหาวิถีสั้นสุดแบบแหล่งต้นทางเดียว มีความเสถียรและสามารถปรับเปลี่ยนโหนดเครือข่ายได้ง่าย ด้วยเหตุนี้จึงมีงานวิจัยที่ศึกษาการแก้ปัญหาเส้นทางของ AGV โดยนำวิธีการหาเส้นทางที่สั้นที่สุดแบบไดก์สตาร์มาใช้กันอย่างแพร่หลาย รวมถึงงานวิจัยนี้เช่นเดียวกัน

ในการหาเส้นทางที่สั้นที่สุดด้วยวิธีการของไดก์สตาร์ เริ่มจากการออกแบบแผนที่โรงงานจำลองขนาดกว้าง 5 เมตร ยาว 13 เมตรให้อยู่ในรูปของระบบกริดและสร้างแผนที่แบบ Unidirectional ดังรูปที่ 10



รูปที่ 10 แสดงแผนที่แบบกริดที่มีเส้นทางการเคลื่อนที่แบบ Unidirectional

การสร้างแผนที่แบบกริดที่มีเส้นทางการเคลื่อนที่แบบ Unidirectional นั้น มีส่วนช่วยในการแก้ปัญหาการจัดการเส้นทางในกรณีที่มีการใช้เส้นทางร่วมกัน เช่น การเดินสวนทางกันในเส้นทางเดียวกัน ดังรูปที่ 11

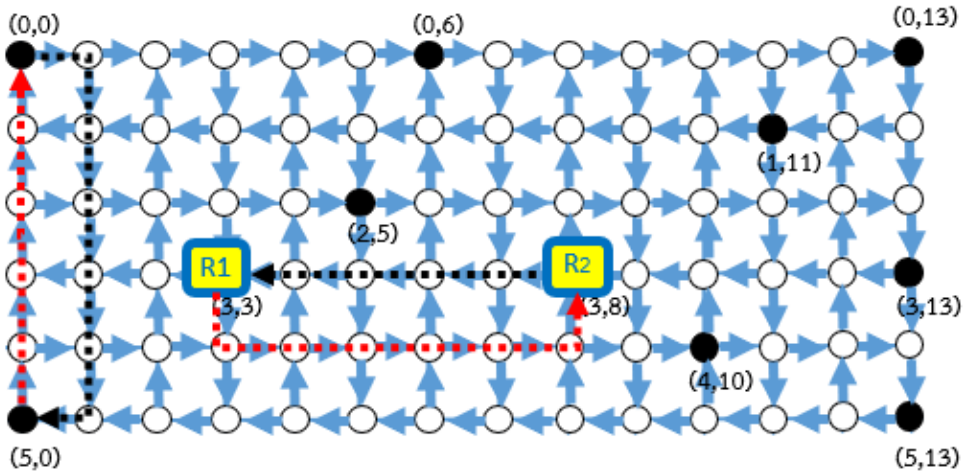


The head-on collision (a) and (b) the head-on collision happens on a path, (c) the head-on collision happens on a node.

รูปที่ 11 แสดงปัญหาการใช้เส้นทางร่วมกันของหุ่นยนต์

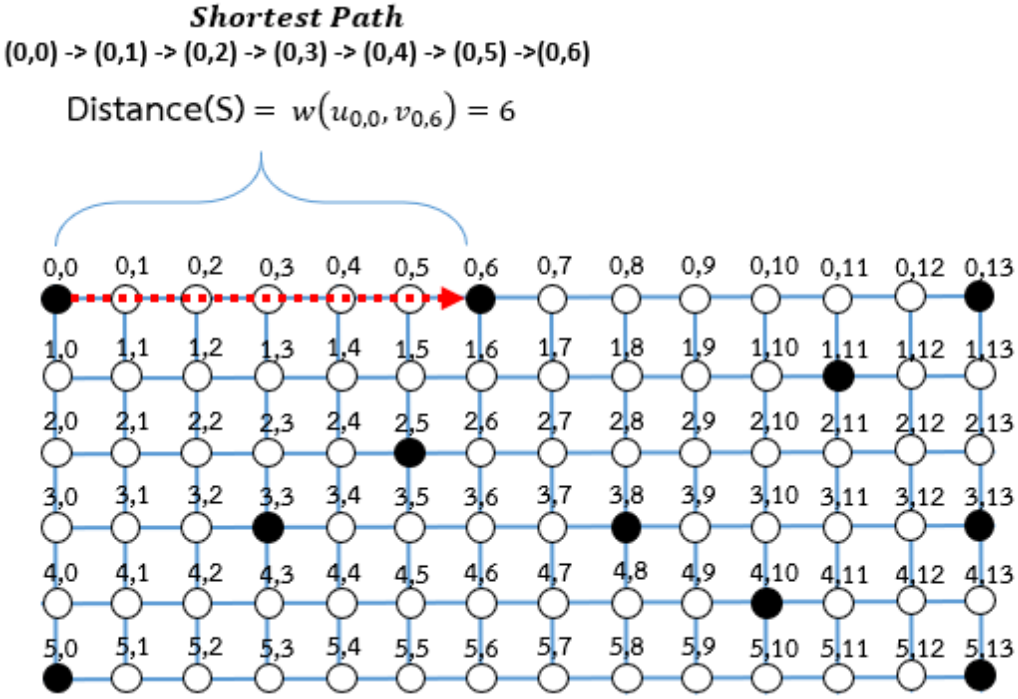
[ภาพจาก: Z. Zhang, Q. Guo, J. Chen and P. Yuan, "Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification," in IEEE Access, vol. 6, pp. 26022-26035, 2018.]

จากการนำรูปแบบเส้นทางการเคลื่อนที่แบบ Unidirectional ที่มีลักษณะการเคลื่อนที่แบบหุ่นยนต์เดินได้ทางเดียว ช่วยตัดปัญหาในเรื่องของการเกิดกรณีดังรูปที่ 10 ได้ แต่ก็มีข้อจำกัดในบางกรณีที่อาจจะเพิ่มระยะทางในการเคลื่อนที่ให้กับหุ่นยนต์ได้เช่นกัน ดังรูปที่ 12 โดยหุ่นยนต์ตัวที่ 1 (R1) หากวิเคราะห์เส้นทางที่สั้นที่สุดโดยไม่สนใจเส้นทางการเดิน พบว่าการเดินทางไปยังสถานีพิกัด (3,8) เป็นเส้นทางเดียวกับหุ่นยนต์ตัวที่ 2 (R2) คือ การเดินตามพิกัด (3,3), (3,4), (3,5), (3,6) ,(3,7) ,(3,8) แต่ด้วยการสร้างการเคลื่อนที่แบบ Unidirectional ทำให้เส้นทางที่สั้นที่สุดของหุ่นยนต์ R1 มีระยะทางเพิ่มขึ้น 2 เมตร เป็นต้น



รูปที่ 12 แสดงเส้นทางที่ถูกเพิ่มระยะให้กับหุ่นยนต์เมื่อเกิดปัญหาหุ่นยนต์ใช้เส้นทางที่สั้นที่สุดร่วมกัน

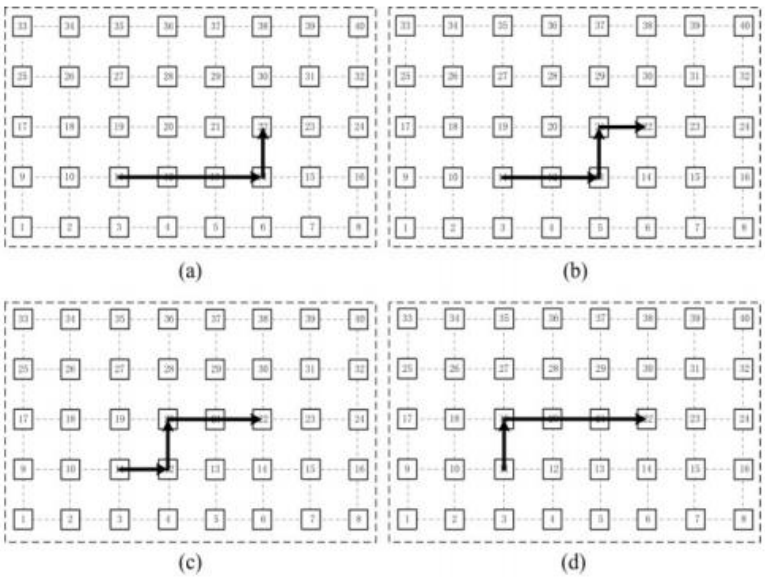
เมื่อนำแผนที่มาสร้างเป็นจุดสำหรับการสร้างเส้นทางเพื่อหาเส้นทางที่สั้นที่สุดด้วยขั้นตอนวิธีของไดค์สตาร์จะได้แผนที่ดังแสดงในรูปที่ 13



รูปที่ 13 แสดงแผนที่แสดงจุดโหนดต่าง ๆ เพื่อใช้ในการหาเส้นทางที่สั้นที่สุดด้วยขั้นตอนวิธีของไดก์สตาร์

จากแผนที่ในรูปที่ 13 พบว่าเมื่อต้องการหาเส้นทางที่สั้นที่สุดจากสถานีพิกัด (0,0) ไปยังสถานีพิกัด (0,6) ด้วยขั้นตอนวิธีของไดก์สตาร์จะได้เส้นทางที่สั้นที่สุดดังนี้ (0,0), (0,1), (0,2), (0,3), (0,4), (0,5), (0,6) โดยมีค่าน้ำหนักของเส้นทางที่ใช้หรือระยะทางรวมทั้งหมด 6 เมตร ซึ่งจากงานวิจัยของ Z. Zhang และคณะ[14] พบว่าในการสร้างเส้นทางที่สั้นที่สุดด้วขั้นตอนวิธีของไดก์สตาร์นั้นมีได้มากกว่า 1 กรณี ดังแสดงในรูปที่ 14 เนื่องจากการหาเส้นทางที่สั้นที่สุดในรูปแบบกริดนั้นมีโอกาสที่ระยะทางรวมเท่ากัน แต่เส้นทางในการเดินไม่เหมือนกัน โดยทั่วไปโปรแกรมจะทำการสุ่มเลือกมาก 1 เส้นทางแต่ในงานวิจัยนี้จะใช้สมการที่ 1 มาช่วยในการเลือกเส้นทางที่สั้นที่สุด เพราะหากลองวิเคราะห์รูปแบบทั้ง 4 จากรูปที่ 14 พบว่าเส้นทางในรูป (b) และ (c) มีการเลี้ยว 2 ครั้ง หากพิจารณาตามสมการที่ 1 พบว่าเวลาที่ใช้ในการเดินทางจะมีมากกว่าเส้นทางในแบบ (a) และ แบบ (d) ที่มีการเลี้ยวเพียงแค่ครั้งเดียว เมื่อตัดสองเส้นทางแรกออกไปแล้วขั้นตอนต่อไปในการเลือกเส้นทาง (a) หรือ (d) จะเลือกจากทิศของหุ่นยนต์ ณ เวลาขณะนั้น หากหุ่นยนต์มีทิศเริ่มต้นอยู่ในทิศเดียวกับรูปแบบการเดินแบบใดผลลัพธ์ที่ได้จากสมการที่ 1 จะทำให้เหลือเส้นทางเพียงแค่เส้นทางเดียวเท่านั้น ในกรณีสุดท้าย คือ

หากคำนวณค่าของเวลาที่ใช้ในการทำงานจากสมการที่ 1 แล้วได้ค่าเท่ากัน โปรแกรมก็จะใช้วิธีสุ่มเลือกวิธีใดวิธีหนึ่งมาเป็นเส้นทางที่สั้นที่สุดสำหรับการทำงานของหุ่นยนต์ตัวนั้น



The shortest routes between node 11 and 22 (a) the first route, (b) the second route, (c) the third route, (d) the fourth route.

รูปที่ 14 แสดงเส้นทางที่สั้นที่สุดจากขั้นตอนวิธีของไดก์สตาร์

[ภาพจาก: Z. Zhang, Q. Guo, J. Chen and P. Yuan, "Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification," in IEEE Access, vol. 6, pp. 26022-26035, 2018.]

$$T_{m,n} = \left(\frac{S_n}{V_n}\right) + \left(\frac{\theta_n}{\omega_n}\right) \qquad (1)$$

เมื่อ T คือ เวลาในการทำงานของหุ่นยนต์

S คือ ผลรวมของระยะทางที่สั้นที่สุดของหุ่นยนต์

V คือ ความเร็วเชิงเส้นของหุ่นยนต์

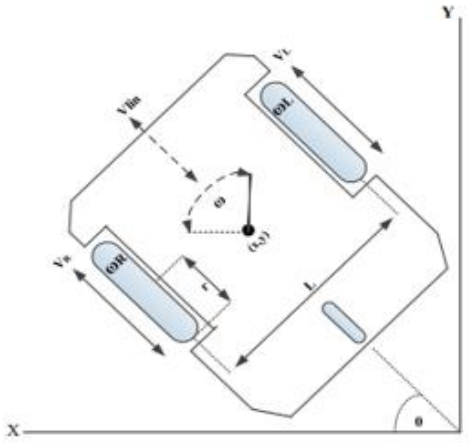
θ คือ ผลรวมของมุมการเคลื่อนที่ของหุ่นยนต์

ω คือ ความเร็วเชิงมุมของหุ่นยนต์

n คือ หุ่นยนต์ตัวที่ n

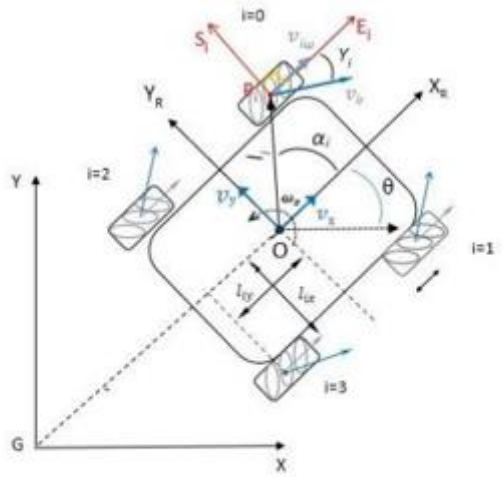
m คือ งานที่ m

ทดสอบขั้นตอนวิธีของไดค์สตาร์กับระบบการจัดการงานหุ่นยนต์ 2 ตัวที่มีลักษณะต่างกัน
หุ่นยนต์ตัวที่ 1 เป็นหุ่นยนต์ที่มีการเคลื่อนที่แบบ Differential drive มีสมการการควบคุมดังนี้



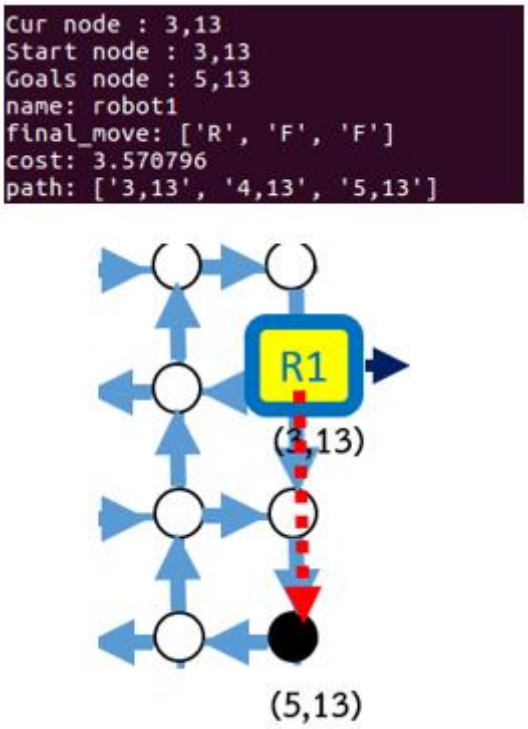
$$x_{t+\Delta} = x_t + V_{lin}\,\Delta\cos\theta_t$$
$$y_{t+\Delta} = y_t + V_{lin}\,\Delta\sin\theta_t$$
$$\theta_{t+\Delta} = \theta_t + \omega\Delta$$

หุ่นยนต์ตัวที่ 2 เป็นหุ่นยนต์ที่มีการเคลื่อนที่แบบ Macanum drive มีสมการการควบคุมดังนี้



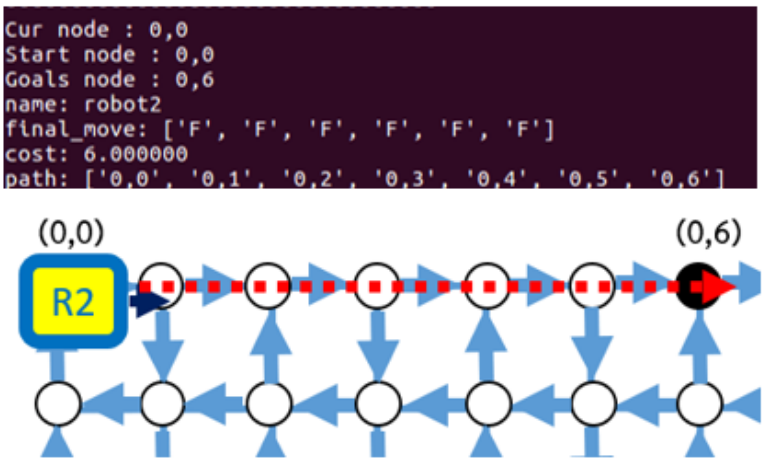
$$\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix} = \frac{r}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -\frac{1}{(l_x+l_y)} & \frac{1}{(l_x+l_y)} & -\frac{1}{(l_x+l_y)} & \frac{1}{(l_x+l_y)} \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix}$$

เมื่อทดสอบหาเส้นทางที่สั้นที่สุดของหุ่นยนต์ Differential drive ตัวที่ 1 มีพิกัดปัจจุบันอยู่สถานีพิกัด (3,13) สถานีเริ่มต้นพิกัด (3,13) และสถานีปลายทาง (5,13) ผลการหาเส้นทางที่สั้นที่สุดได้ดังรูปที่ 15



รูปที่ 15 แสดงผลลัพธ์ที่ได้จากสถานีพิกัด (3,13) ไปยังสถานีพิกัด(5,13)

เมื่อทดสอบหาเส้นทางที่สั้นที่สุดของหุ่นยนต์ Macanum drive ตัวที่ 2 มีพิกัดปัจจุบันอยู่สถานีพิกัด (0,0) สถานีเริ่มต้นพิกัด (0,0) และสถานีปลายทาง (0,6) ผลการหาเส้นทางที่สั้นที่สุดได้ดังรูปที่ 16



รูปที่ 16 แสดงผลลัพธ์ที่ได้จากสถานีพิกัด (0,0) ไปยังสถานีพิกัด(0,6)

โดยสรุปโปรแกรมที่พัฒนาขึ้นรองรับการคำนวณระยะทางที่สั้นที่สุดของหุ่นยนต์ที่มีระบบขับเคลื่อนต่างกันได้

1.3 ส่วนการจัดการลำดับของงาน

สำหรับการจัดการลำดับของงานนั้นใช้วิธี Market based approach มาช่วยในการจัดเรียงงานโดยนำระยะทาง และความเร็วของหุ่นยนต์แต่ละตัวในระบบมาวิเคราะห์เวลาที่น้อยที่สุดสำหรับงานแต่ละงาน ซึ่งหาได้จากสมการที่ 1 โดยค่าความเร็วได้หาได้จากความเร็วที่ใช้ในการทำงานของหุ่นยนต์แต่ละตัวขึ้นอยู่กับองค์ประกอบของอุปกรณ์ของหุ่นยนต์ประกอบด้วยสองค่า คือ ค่าความเร็วเชิงเส้น (V) และค่าความเร็วเชิงมุม (ω) ค่าระยะทางที่นำมาคำนวณ หาได้จากระยะทางที่สั้นที่สุดที่ใช้ในการเดินทางจากตำแหน่งปัจจุบันของหุ่นยนต์ไปยังสถานีเริ่มต้นทำงานและจากสถานีเริ่มต้นทำงานไปยังสถานีปลายทาง ตามวิธีที่กล่าวมาแล้วก่อนหน้า เมื่อโปรแกรมคำนวณจะได้เซตเวลาของหุ่นยนต์กับเวลาที่ใช้ในการทำงานแต่ละงานดังแสดงในสมการที่ 2

$$J_m = \{R_n : T_m\} \qquad (2)$$

เมื่อ J คือ เซตของงาน $\{J_1, J_2, \ldots, J_m\}$

R คือ หุ่นยนต์ในระบบ

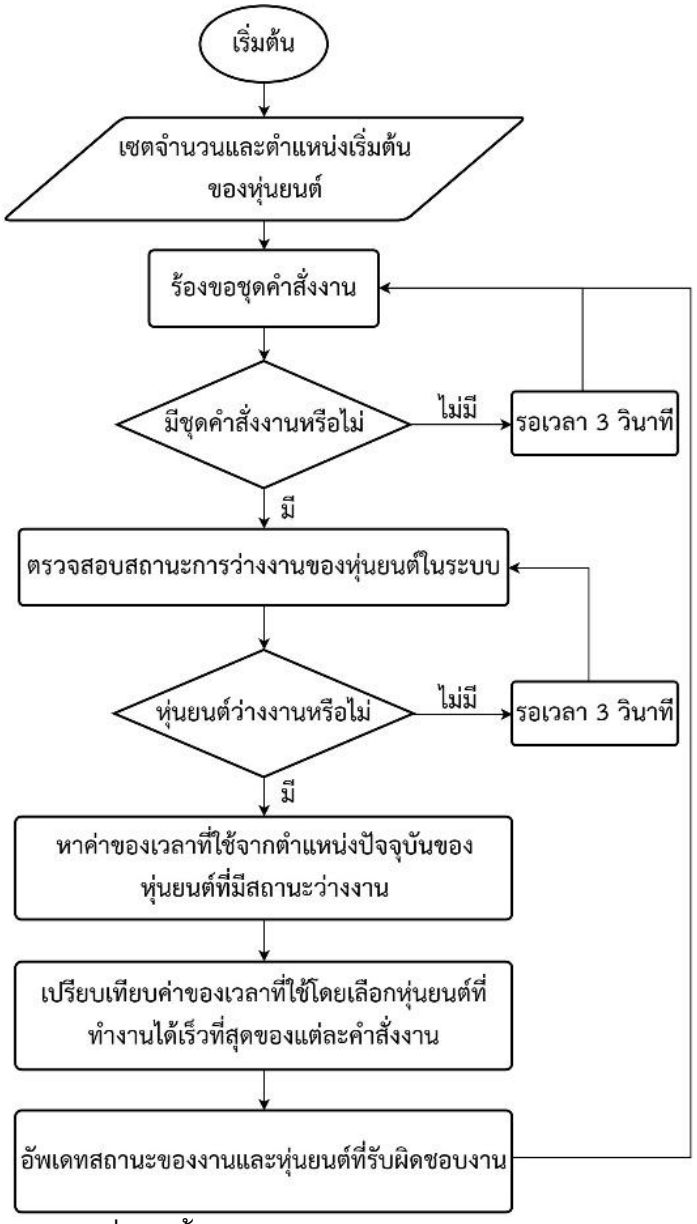
T คือ เวลาที่ถูกประเมินสำหรับงาน

จากนั้นนำเซตเวลาการทำงานของหุ่นยนต์ดังสมการที่ (2) มาเปรียบเทียบหาเวลาที่น้อยที่สุด สำหรับเลือกงานให้กับหุ่นยนต์ด้วยการนำค่าเวลาในการทำงานของหุ่นยนต์แต่ละตัวมา bidding สำหรับชุดงานนั้นๆ ดังสมการที่ (3)

$$Task_i = \min(J_m) \qquad (3)$$

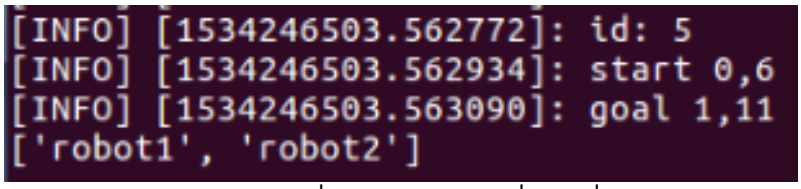
เมื่อ i คือ ลำดับการมอบหมายงาน

เมื่อสามารถทำการ bidding งานให้กับหุ่นยนต์ได้แล้ว ระบบการจัดการจะทำการมอบหมายงานนั้นๆ ให้กับหุ่นยนต์ตัวที่เหมาะสม และประเมินเวลางานใหม่ที่เข้ามาให้กับหุ่นยนต์ที่ว่างงานอยู่กระทำซ้ำเช่นนี้จนมอบหมายงานที่มีอยู่ทั้งหมดให้หุ่นยนต์ทำงานจนเสร็จสิ้น ขั้นตอนการทำงานของการจัดการงานแสดงดังรูปที่ 17

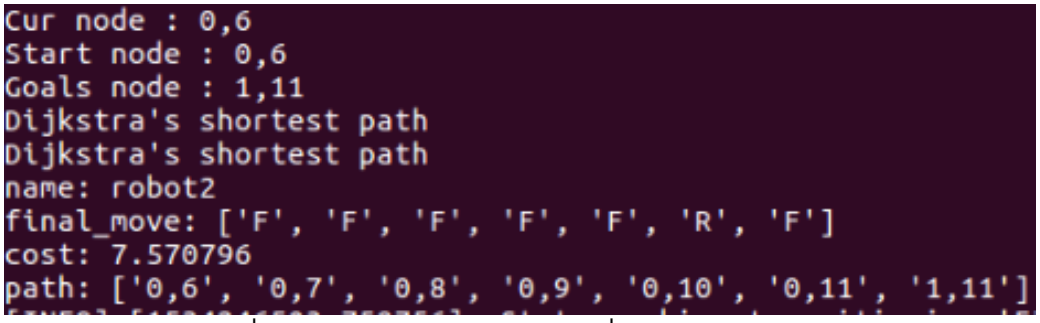


รูปที่ 17 ขั้นตอนการทำงานของการจัดการงาน

ทดสอบการหาค่า cost สำหรับ biding ด้วยวิธี Market based approach โดยมีตัวอย่างงานที่ 5 รายละเอียดแสดงดังรูปที่ 18 ตัวอย่างคำสั่งงานที่ 5 สถานีเริ่มต้น (0,6) สถานีปลายทาง (1,11) มีหุ่นยนต์ว่างงาน 2 ตัวคือ robot1 และ robot2



รูปที่ 18 ตัวอย่างคำสั่งงานที่ 5



รูปที่ 19 ตัวอย่างการหาค่า cost ที่ได้จากระบบการจัดการ

จากรูปที่ 19 พบว่าเป็นงานที่มีสถานีพิกัดเริ่มต้น คือ (0,6) สถานีปลายทาง (1,11) หุ่นยนต์ที่ถูกเลือก คือ หุ่นยนต์ robot2 มีค่า cost เท่ากับ 7.570796 ซึ่งจากการทดสอบพบว่าหุ่นยนต์ robot2 มีพิกัดปัจจุบันที่ (0,6) หุ่นยนต์ robot2 ความเร็วเชิงเส้นเท่ากับ 1 m/s ความเร็วเชิงมุมเท่ากับ 1 rad/s โดยแผนที่มีระยะห่างระหว่างจุดเท่ากับ 1 m ค่ามุมการเคลื่อนที่ของหุ่นยนต์ 90 องศา หรือ $\frac{\pi}{2}$ rad จากสมการที่ 1 คำนวณหาค่าของ cost ได้ดังนี้

$$T_{5,2} = \left(6\frac{1}{1}\right) + \left(\frac{\frac{\pi}{2}}{1}\right)$$

$T_{5,2}$ = 7.570796

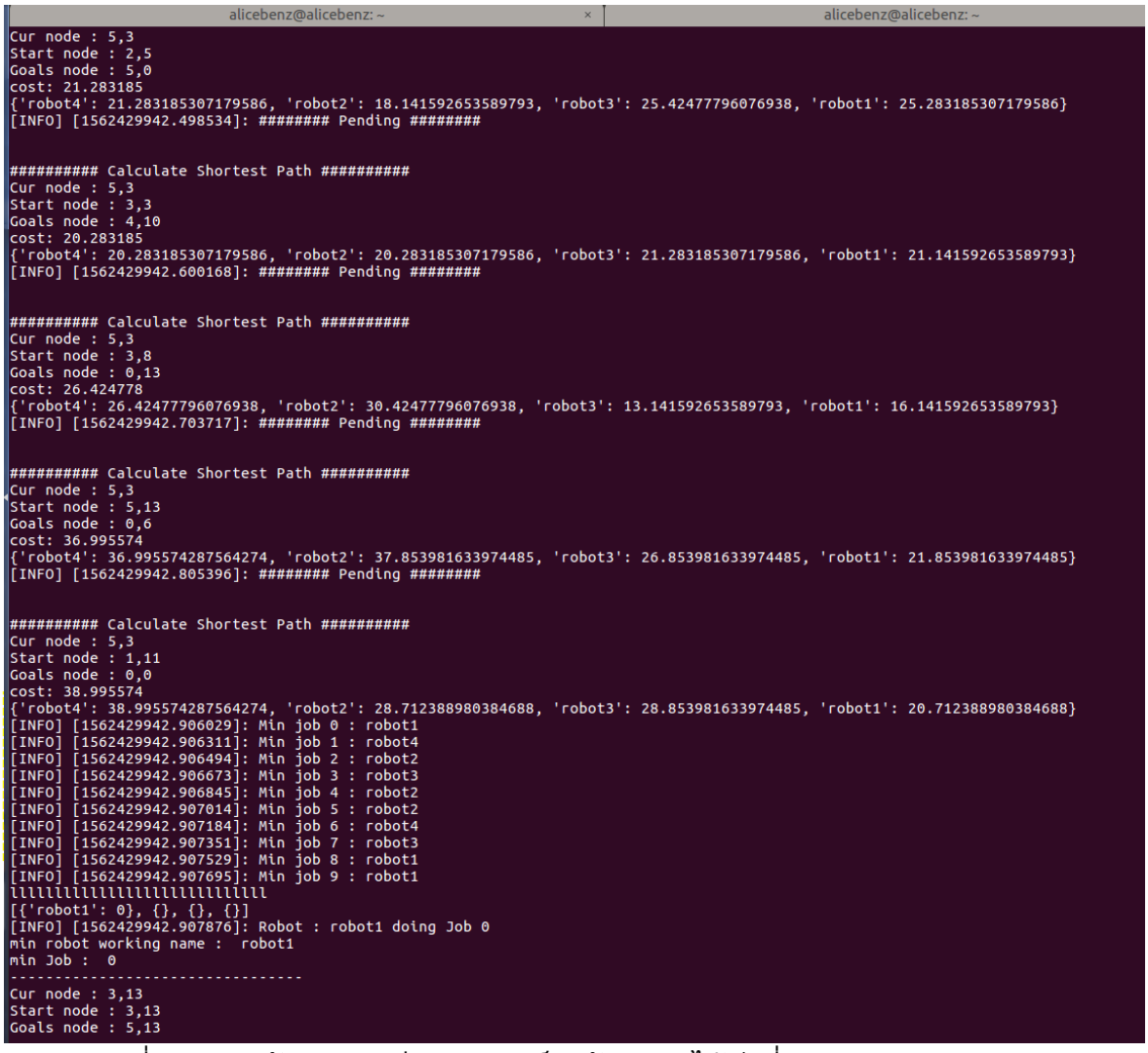
พบว่าคำตอบที่ได้เท่ากับที่ระบบการจัดการสามารถคำนวณได้อย่างถูกต้อง และหาค่าระยะทางของหุ่นยนต์ robot1 ที่มีพิกัดปัจจุบันอยู่ที่ (5,13) ทำการคำนวณเช่นเดียวกันกับหุ่นยนต์ robot2 พบว่าได้ค่า cost อยู่ที่ 27.424778 เมื่อได้ค่า cost ของหุ่นยนต์ทั้งสองแล้วสามารถเขียนเป็นเซตหุ่นยนต์ตามสมการที่ 2 ได้ดังนี้

$$J_5 = \{R_1 : 27.42477796076938\}$$
$$J_5 = \{R_2 : 7.570796326794897\}$$

นำค่าที่ได้มาทำการเปรียบเทียบเพื่อเลือกหุ่นยนต์ที่เหมาพสมกับงานที่ 5 ดังแสดงในสมการที่ 3 จะได้ว่า

$$Task_5 = \min\,(27.42477796076938,\ 7.570796326794897)$$
$$Task_5 = \ 7.570796326794897$$
$$Task_5 = R_2$$

นอกจากนี้หากมีกรณีที่หุ่นยนต์ 1 ตัวมีงานที่สามารถทำงานได้เร็วที่สุดหลายงานจากตำแหน่งปัจจุบันที่หุ่นยนต์อยู่ จะมีลักษณะการเลือกงานเช่นเดียวกับที่กล่าวไปแล้วข้างต้น ดังตัวอย่างต่อไปนี้จากรูปที่ 20



รูปที่ 20 แสดงตัวอย่างกรณีงานหุ่นยนต์ 1 ตัวทำงานได้เร็วที่สุดมากกว่า 1 งาน

จากรูปที่ 20 งาน job0 มีสถานีพิกัดเริ่มต้น (3,13) สถานีปลายทาง (5,13), งาน job8 มีสถานีพิกัดเริ่มต้น (3,8) สถานีปลายทาง (0,13) และ งาน job9 มีสถานีพิกัดเริ่มต้น (5,13) สถานีปลายทาง (0,6) จากสมการที่ 2 จะสามารถเขียนได้ว่า

$$J_0 = \{R_1: 3.570796\}$$
$$J_8 = \{R_1: 19.283185\}$$
$$J_9 = \{R_1: 18.712389\}$$

เมื่อแทนลงในสมการที่ 3 จะได้ว่างานที่มอบหมายให้กับหุ่นยนต์ robot1 คือ งานที่ 0 ซึ่งมีค่า cost น้อยที่สุด คือ 3.570796

1.4 ส่วนการจัดการเส้นทางของหุ่นยนต์

สำหรับการจัดการเส้นทางของหุ่นยนต์เพื่อให้หุ่นยนต์สามารถขนถ่ายสินค้าได้โดยไม่เกิดการชนกันในเบื้องต้นได้แก้ปัญหาด้วยการใช้การสร้างเส้นทางเดินแบบทางเดียว (Unidirectional) ซึ่งได้กล่าวไว้แล้วในหัวข้อ 1.2 การจัดการเส้นทางการเดินของหุ่นยนต์แต่ละตัวจะทำงานร่วมกับการควบคุมหุ่นยนต์ โดยมีระบบส่วนกลางเป็นตัวควบคุมการเดินของหุ่นยนต์แต่ละตัว หลังจากที่หุ่นยนต์ได้รับมอบหมายงานจากระบบการจัดการงาน หุ่นยนต์จะได้เส้นทางที่ต้องเดินไปยังจุดหมายพร้อมทั้งรูปแบบการเคลื่อนที่ สำหรับรูปแบบการเคลื่อนที่ของหุ่นยนต์นั้นจะมีการสร้างรูปแบบการเคลื่อนที่ให้กับหุ่นยนต์ซึ่งสามารถหาได้จากการนำเส้นทางที่สั้นที่สุดมาแปลงรูปแบบการเคลื่อนที่ให้กับหุ่นยนต์ดังตารางที่ 1 การแปลงรูปแบบการเคลื่อนที่นั้นทำขึ้นเพื่อลดความซับซ้อนในการควบคุมและเพื่อให้การควบคุมหุ่นยนต์นั้นเกิดความยืดหยุ่นในการนำระบบไปใช้กับหุ่นยนต์คนละชนิดกัน โดยมีลักษณะฟังก์ชันการเคลื่อนที่ของหุ่นยนต์ 3 รูปแบบ คือ Forward (เดินหน้า) Left (หุ่นยนต์เลี้ยวซ้ายเป็นมุม 90 องศา) Right (หุ่นยนต์เลี้ยวขวาเป็นมุม 90 องศา) และ Turnaround (หุ่นยนต์หมุนตามเข็มนาฬิกาเป็นมุม 180 องศา) ระบบการจัดการหุ่นยนต์จะใช้ข้อมูลตำแหน่งปัจจุบันแทนด้วย (x_i, y_i) ตำแหน่งถัดไปแทนด้วย (x_j, y_j) และทิศทางของหุ่นยนต์แทนด้วย North, East, West, South จากนั้นหาความสัมพันธ์ของตำแหน่งถัดไปและตำแหน่งปัจจุบัน โดยการหาค่า x และ y จากสมการที่ (4) และ (5) ตามลำดับ

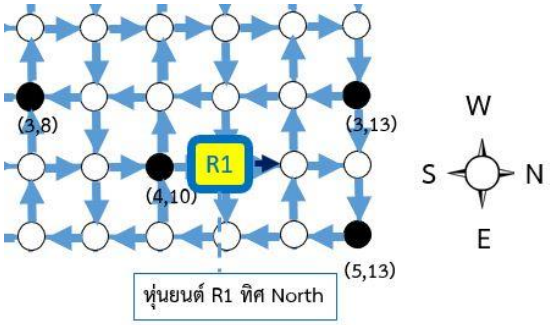
$$x = x_j - x_i \qquad (4)$$
$$y = y_j - y_i \qquad (5)$$

แล้วนำค่าที่ได้ไปเทียบกับตารางที่ 1 เพื่อแปลงเส้นทางการเคลื่อนที่ที่สั้นที่สุดจากขั้นตอนของไดก์สตรา เป็นรูปแบบการเคลื่อนที่ของหุ่นยนต์ โดยในงานวิจัยนี้ได้จัดการให้หุ่นยนต์ Mecanum drive เคลื่อนที่แบบเคลื่อนที่ไปหน้าและเลี้ยวแบบไม่สไลด์ จึงทำให้ตารางแปลงการเคลื่อนที่สามารถใช้ได้กับทั้งหุ่นยนต์แบบ Differential drive และ Mecanum drive

ตารางที่ 1 การแปลงเส้นทางเป็นรูปแบบการเคลื่อนที่

ทิศทางของหุ่นยนต์ ณ ตำแหน่งปัจจุบัน	ความสัมพันธ์ของตำแหน่งถัดไปกับตำแหน่งปัจจุบันของหุ่นยนต์		คำสั่งการเคลื่อนที่
	x	y	
North	0	1	Forward
	0	-1	Turnaround and Forward
	-1	0	Left and Forward
	1	0	Right and Forward
East	0	1	Left and Forward
	0	-1	Right and Forward
	-1	0	Turnaround and Forward
	1	0	Forward
West	0	1	Right and Forward
	0	-1	Left and Forward
	-1	0	Forward
	1	0	Turnaround and Forward
South	0	1	Turnaround and Forward
	0	-1	Forward
	-1	0	Right and Forward
	1	0	Left and Forward

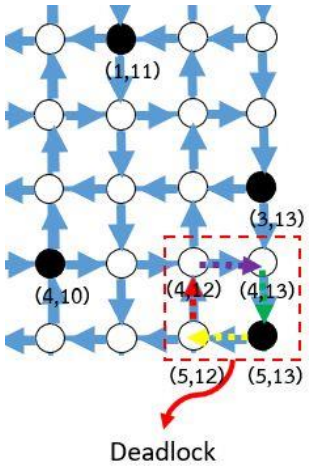
ตัวอย่างเช่น จากรูปที่ 21 หุ่นยนต์ R1 ตำแหน่งปัจจุบัน คือ (4,11) ทิศ North มีพิกัดเป้าหมายตำแหน่งถัดไป คือ (4,12) หาค่า x และ y จาก สมการที่ (4) และ (5) จะได้ว่า x = 0, y = 1 เมื่อนำค่าไปแทนในตารางที่ 1 จะได้ชุดคำสั่งการเคลื่อนที่ คือ Forward เป็นต้น



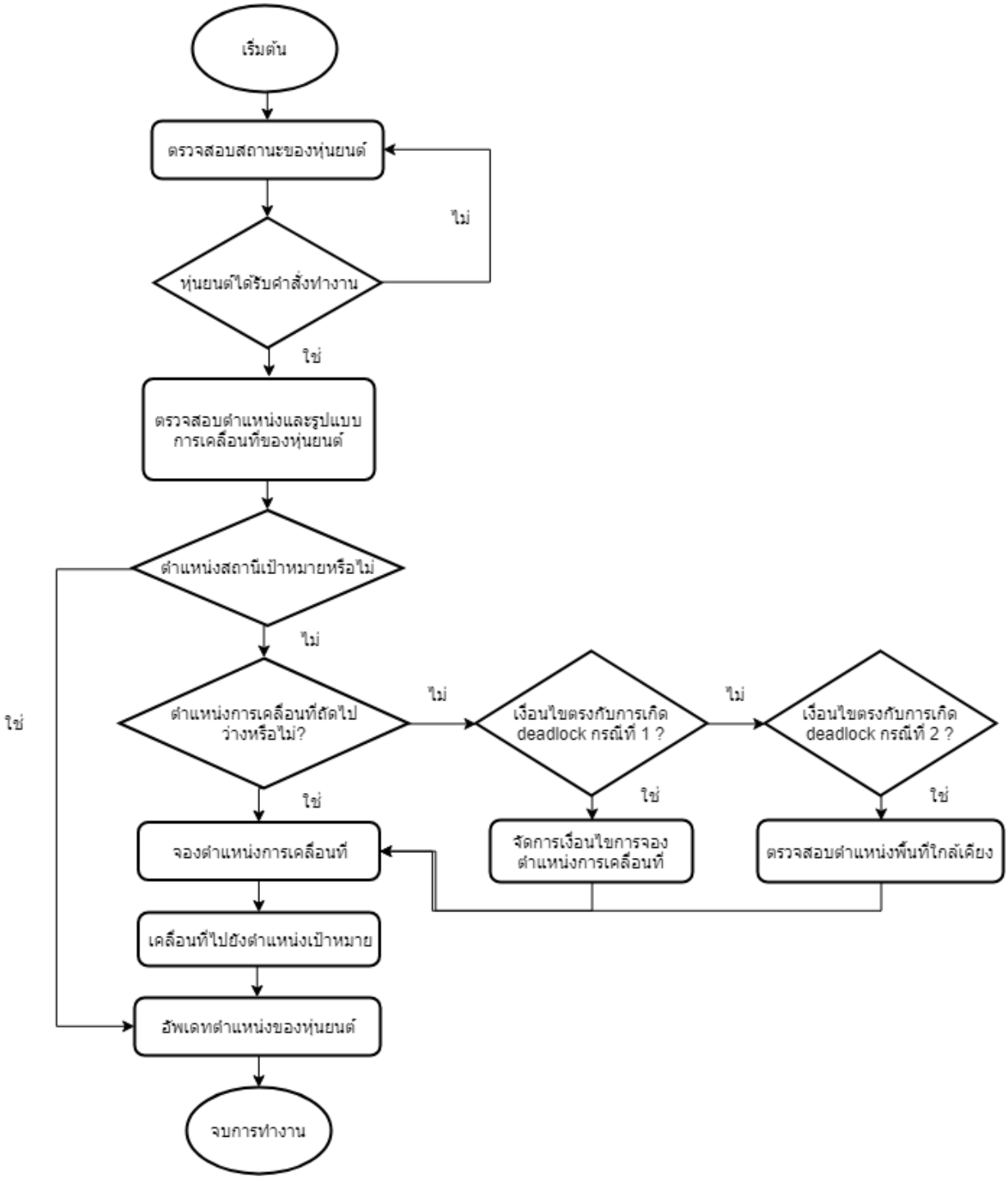
รูปที่ 21 การระบุทิศและการเคลื่อนที่ของหุ่นยนต์เทียบกับแผนที่

เมื่อได้รูปแบบการเคลื่อนที่ของหุ่นยนต์แล้ว ระบบการจัดการเส้นทางของหุ่นยนต์จะทำการควบคุมหุ่นยนต์ตามผังงานที่แสดงดังรูปที่ 23 จนกระทั่งจบการทำงาน

จากการทดสอบการควบคุมหุ่นยนต์พบว่าเกิดกรณีปัญหา deadlock ที่ทำให้หุ่นยนต์ไม่สามารถเดินต่อไปได้ดังรูปที่ 22 โดยหุ่นยนต์ 4 ตัวมีการใช้จุดจราจรร่วมกัน คือ หุ่นยนต์ตัวที่หนึ่งต้องการเคลื่อนตัวจาก (4,12) ไปยังจุด (4,13), หุ่นยนต์ตัวที่สองจากจุด (4,13) ไปยังจุด (5,13), หุ่นยนต์ตัวที่สามจาก (5,13) ไปยัง (5,12), และหุ่นยนต์ตัวที่สี่จากจุด (5,12) ไปยัง (4,12) กรณีนี้ Manca และคณะได้[15] แสดงวิธีการแก้ปัญหาไว้แล้ว กล่าวคือ ระบบจัดการเส้นทางจะทำการตรวจสอบการใช้พิกัดร่วมกันก่อนจะเกิดเหตุการณ์ดังกล่าวโดยหากมีหุ่นยนต์ใช้พิกัดที่ใช้ร่วมกันมากกว่า 2 จุด เช่น หุ่นยนต์ตัวที่หนึ่งที่ต้องการเคลื่อนตัวจากจุด (4,12) ไปยังจุด (4,13) ตรวจสอบพบว่าหุ่นยนต์ตัวที่สองต้องการใช้จุด (4,13) และหุ่นยนต์ตัวที่สี่ต้องการใช้จุด (4,12) ระบบจัดการเส้นทางจะอนุญาตให้หุ่นยนต์ทั้งสี่สามารถจองจุดถัดไปเพื่อเคลื่อนไปยังจุดถัดไปได้



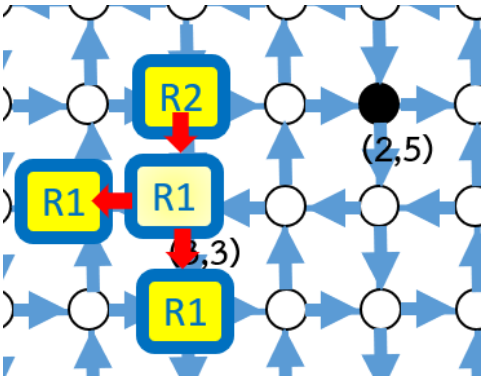
รูปที่ 22 เหตุการณ์ deadlock ของหุ่นยนต์ 4 ตัวที่ใช้เส้นทางพร้อมกัน



รูปที่ 23 ขั้นตอนการทำงานของการจัดการเส้นทาง

นอกจากนี้กรณีที่พบอีกหนึ่งกรณี คือ เมื่อหุ่นยนต์ทำงานเสร็จแล้วยังคงหยุดรองานอยู่ ณ สถานีงาน เมื่อมีหุ่นยนต์ (R2) ต้องการเข้าสถานีงานนั้นจะไม่สามารถเข้าได้

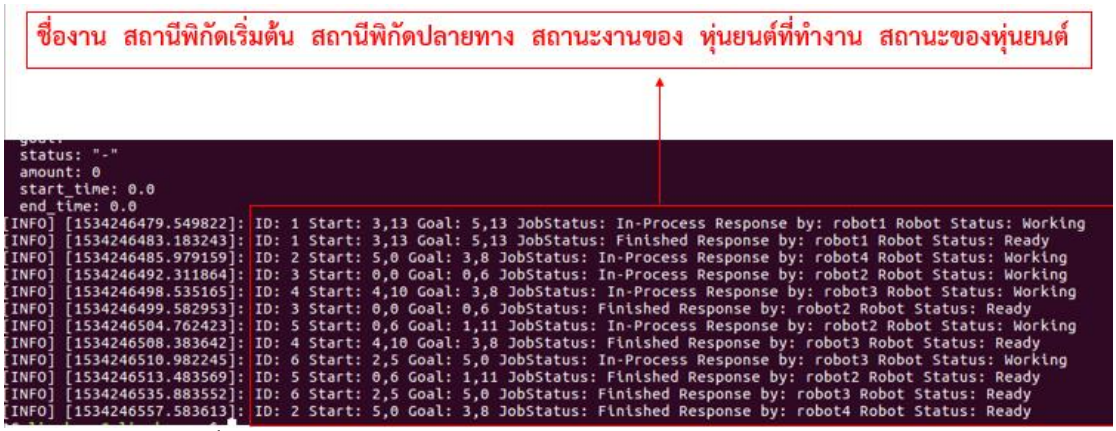
เนื่องจากมีหุ่นยนต์ (R1) ที่ว่างงานอยู่ในสถานีงาน จึงได้ทำการแก้ปัญหาโดยตรวจสอบตำแหน่งถัดของหุ่นยนต์ (R1) ของสถานีนั้น หากว่างก็ให้เคลื่อนย้ายไปยังตำแหน่งที่ว่างและให้หุ่นยนต์ (R2) ที่มาส่งสินค้าได้เข้าไปยังสถานีงานนั้น ดังแสดงในรูปที่ 24



รูปที่ 24 แสดงการแก้ปัญหาหุ่นยนต์ว่างงานอยู่ตรงจุดสถานีงาน

1.5 ส่วนการติดตามสถานะงานและหุ่นยนต์

การติดตามสถานะงานและหุ่นยนต์เป็นส่วนที่ใช้ในการติดตามสถานะงานต่าง ๆ ระบบ โดยโปรแกรมจะทำการอัพเดทรายละเอียดต่างๆ ในขณะที่ระบบทำงาน ดังแสดงในรูปที่ 25 รายละเอียดในการแสดงสถานะของระบบจะประกอบไปด้วย ชื่องาน สถานีพิกัดเริ่มต้น สถานีพิกัดปลายทาง สถานะของงาน หุ่นยนต์ที่ทำงาน สถานะของหุ่นยนต์



รูปที่ 25 แสดงการติดตามสถานะงานและหุ่นยนต์ของระบบการจัดการงาน

2. การเปรียบเทียบความแตกต่างของเวลาระหว่างระบบการจัดงานหุ่นยนต์หลายตัวแบบ FIFO และ Market based approach

หลังจากได้สร้างระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรมแล้ว ต่อไปจะเป็นการทดสอบเวลาที่ใช้ในการจัดการงานระหว่างการจัดการงานแบบ FIFO ที่ทำงานตามลำดับของงานที่มาก่อนจัดการก่อนและงานที่มาหลังจัดการหลัง เปรียบเทียบกับการจัดการงานแบบ Market based approach ที่อาศัยการ bidding ค่าของเวลาที่ใช้น้อยที่สุดมาทำงานก่อนโดยพิจารณาจากคุณสมบัติของหุ่นยนต์แต่ละตัว ณ ตำแหน่งปัจจุบัน โดยทั้งการจัดการงานแบบ FIFO และ Market based approach จะใช้วิธีหาระยะทางที่สั้นที่สุดด้วยวิธีของไดค์สตราเหมือนกัน

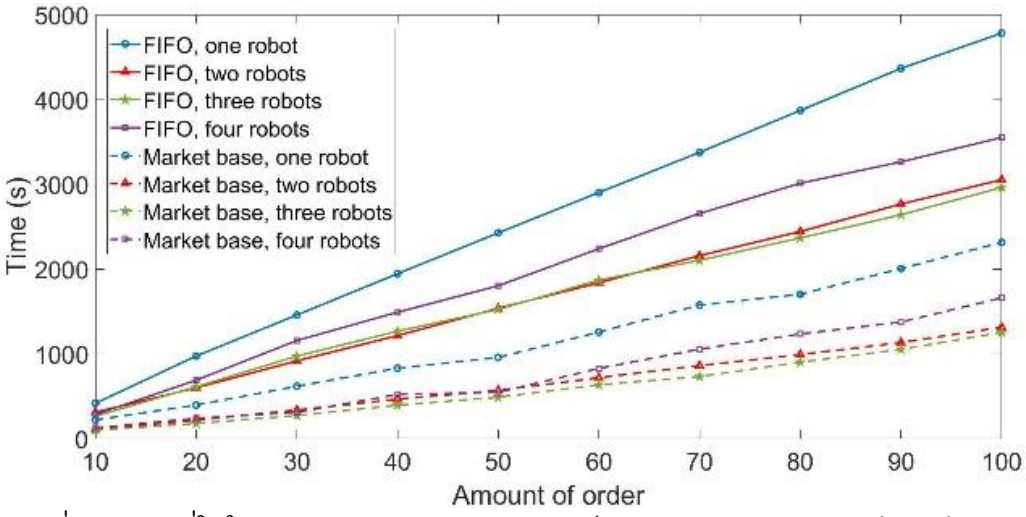
ในการทดสอบจะสอบเปรียบเทียบจากงานที่ออกแบบไว้โดยชุดคำสั่งงานที่ทำเหมือนกัน หุ่นยนต์จำนวนเท่ากัน และคุณสมบัติเหมือนกัน แต่การตัดสินใจเลือกชุดคำสั่งงานแตกต่างกัน เพื่อเปรียบเทียบค่าของเวลาที่ใช้ในการจัดการงานทั้งหมด โดยการทดลองจะทดสอบชุดคำสั่งงานเริ่มต้นจากการทดลองเปรียบเทียบจำนวนหุ่นยนต์ที่เหมาะสมในระบบโดยการคำนวณเวลาที่ใช้ทำงานของระบบด้วยจำนวนหุ่นยนต์หนึ่งตัวถึงสี่ตัวทำงานแบบ FIFO เปรียบเทียบกับวิธี Market based โดยเพิ่มจำนวนชุดคำสั่งงานครั้งละ 10 งาน จนถึง 100 งาน โดยงานที่ใช้ในการจำลองอ้างอิงจากสถานีงานทั้งหมด 11 สถานี(จุด) ยกตัวอย่างการสร้างชุดคำสั่งงานดังตารางที่ 2 โดยมีสถานีเริ่มต้นและสิ้นสุดที่ได้มาจากการสุ่มกระจายทั่วบริเวณโรงงานจำลอง

ตารางที่ 2 ตัวอย่างการสร้างชุดคำสั่งงาน

คำสั่งงาน	สถานีเริ่มต้น	สถานีปลายทาง
01	0,0	0,6
02	0,6	0,13
03	0,13	1,11
04	1,11	2,5
05	2,5	3,3
06	3,3	3,8
07	3,8	3,13
08	3,13	4,10
09	4,10	5,0
10	5,0	5,13

กำหนดให้หุ่นยนต์มีความเร็วเชิงเส้น 1 m/s ความเร็วเชิงมุม 1.57 rad/s สำหรับตำแหน่งเริ่มต้นนั้นจะกำหนดตำแหน่งหุ่นยนต์ให้กระจายทั่วแผนที่ โดยให้

ตำแหน่งเริ่มต้นของหุ่นยนต์ตัวที่หนึ่งอยู่ที่พิกัด 3,13 ลักษณะของหุ่นยนต์ตัวที่หนึ่งเป็นแบบ Differential wheel หุ่นยนต์ตัวที่สองอยู่ที่พิกัด (0,0) ลักษณะของหุ่นยนต์ตัวที่สองเป็นแบบ Mecanum wheel หุ่นยนต์ตัวที่สามอยู่พิกัด (4,7) ลักษณะของหุ่นยนต์ตัวที่สามเป็นแบบ Differential wheel หุ่นยนต์ตัวที่สี่อยู่พิกัด (5,3) ลักษณะของหุ่นยนต์ตัวที่สี่เป็นแบบ Mecanum wheel ทำงานบนแผนที่จำลองโรงงานอุตสาหกรรมตามรูปที่ 10 ผลการจำลองการทำงานของโปรแกรมระบบควบคุมด้วย ROS แสดงให้เห็นความสัมพันธ์ของจำนวนงานและเวลาที่ใช้ในการจัดการงานทั้งหมดของทั้งสองวิธีเปรียบเทียบกันดังรูปที่ 26

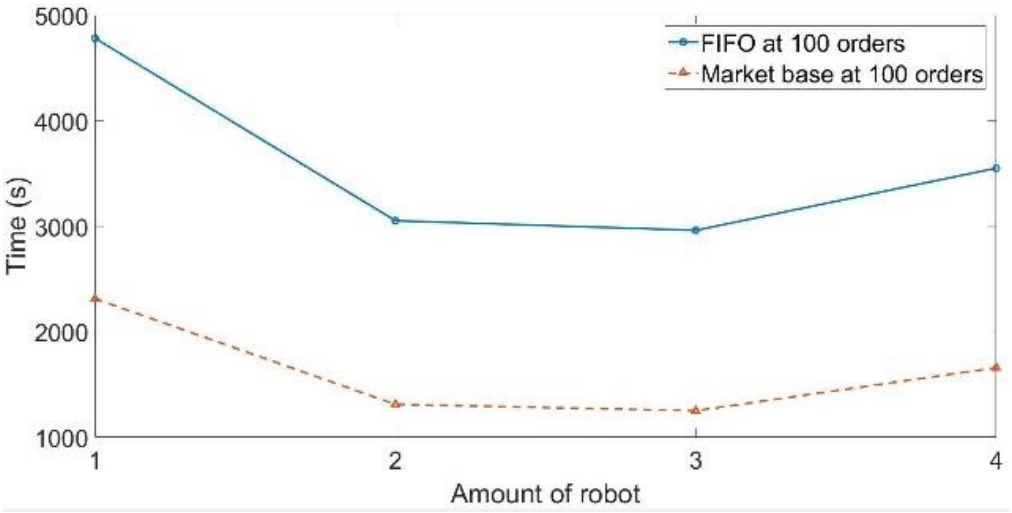


รูปที่ 26 เวลาที่ใช้ในการทำงานของระบบหุ่นยนต์หลายตัวแบบ FIFO เปรียบเทียบกับวิธี Market based approach

จากผลการทดลองพบว่าวิธีการ Market based ใช้เวลาในการจัดการงานทั้งหมดเสร็จสิ้นได้เร็วกว่าวิธีแบบ FIFO ถึง 1 เท่าตัว เมื่อเปรียบเทียบด้วยกรณีที่มีหุ่นยนต์จำนวนทำงานเท่ากันแต่วิธีจัดการงานต่างกัน และเวลาที่ใช้ในการทำภารกิจจนเสร็จสิ้น จะเพิ่มขึ้นเป็นเชิงเส้นเมื่อจำนวนชุดคำสั่งเพิ่มขึ้น

นอกจากนี้ผลการจำลองยังพบว่า หากพิจารณาที่จำนวนชุดคำสั่งเท่ากันแล้ว รูปที่ 27 แสดงให้เห็นว่าเมื่อจำนวนหุ่นยนต์แบบเดียวกันในระบบมีจำนวนเพิ่มขึ้น เวลาที่ใช้ในการจัดการงานทั้งหมดจนเสร็จสิ้นจะลดลงจนถึงจุดหนึ่งเท่านั้น เพราะหากมีจำนวนหุ่นยนต์เยอะจนเกินไปจะทำให้เกิดการจราจรแออัด หุ่นยนต์ต้องเสียเวลารอเส้นทางว่างเกิดขึ้น เป็นผลให้ทั้งระบบใช้เวลามากกว่าที่ควรจะเป็น จากการทดลองการจัดการงานที่ 100 ชุดคำสั่งงาน โดยเพิ่มจำนวนหุ่นยนต์ในระบบจาก 1 ถึง 4 ตัว พบว่าสำหรับพื้นที่การทำงานที่กว้าง 5 เมตร ยาว 13 เมตร สำหรับการจัดการชุดคำสั่ง 100

งาน การเลือกใช้หุ่นยนต์ทำงาน 3 ตัว ด้วยวิธี Market base สามารถจัดการงานได้เร็วที่สุดด้วยเวลา 1,252 วินาที ใกล้เคียงกับหุ่นยนต์จำนวน 2 ตัว



รูปที่ 27 เวลาการทำงานของระบบจัดการหุ่นยนต์ที่จำนวนชุดคำสั่งเท่ากัน แต่จำนวนหุ่นยนต์ในระบบเพิ่มขึ้น

สรุปผลการศึกษา

จากผลการจำลองการจัดการงานด้วยวิธีแบบ FIFO เปรียบเทียบกับวิธี Market based พบว่า วิธีการของ Market base approach ทำงานได้เร็วกว่า รวมถึงจำนวนหุ่นยนต์ที่เพิ่มขึ้น สามารถช่วยลดเวลาในการทำงานได้ ซึ่งจากการทดลองสำหรับจำนวนงานทั้งหมด 100 งาน ด้วยวิธีของ Market based ระบบสามารถลดเวลาการทำงานลงครึ่งหนึ่งเมื่อเทียบกับระบบ FIFO นอกจากนี้ยังพบว่าการนำหุ่นยนต์หลายตัวมาใช้ในการทำงาน จำนวนหุ่นยนต์ที่มากขึ้นไม่ได้ส่งผลให้เวลาในการทำงานลดลงเสมอไป เมื่อมีหุ่นยนต์มากถึงจำนวนหนึ่งก็จะส่งผลให้ระบบการทำงานช้าลงได้ด้วย ยกตัวอย่างเช่น เวลาที่ใช้ในการคำนวณเส้นทางมากขึ้น ปัญหาการจัดการจราจรที่พบบ่อยขึ้น ดังนั้นควรให้ความสำคัญกับจำนวนหุ่นยนต์ที่เลือกใช้ในการขนถ่ายวัสดุในอุตสาหกรรมด้วยเช่นกัน

นอกจากนี้หากสังเกตรูปที่ 27 การทำงานหุ่นยนต์ 1 ตัวเท่ากันจะพบว่าการใช้วิธีของ Market based approach ใช้เวลาการทำงานน้อยกว่า เนื่องจากในการเลือกงานนั้น การจัดการงานของหุ่นยนต์จะให้ความสำคัญกับตำแหน่งปัจจุบันของหุ่นยนต์ เมื่อระบบทำการเลือกงานให้กับหุ่นยนต์ด้วยวิธีการของ Market based approach ที่นำค่าของเวลาและตำแหน่งปัจจุบันของหุ่นยนต์แต่ละตัวไปคำนวณหาค่า cost ตามสมการที่ 1 เพื่อนำมาตัดสินใจเลือกงานให้กับหุ่นยนต์จะพบว่ามีบางกรณีที่หุ่นยนต์ที่เพิ่งขนถ่ายวัสดุเสร็จจากชุดงานก่อนหน้า มีตำแหน่งที่จะขนถ่ายวัสดุสำหรับชุดงานถัดไปใกล้เคียงมากกว่าการจัดการ

งานแบบ FIFO ที่ถูกบังคับให้ต้องทำงานตามลำดับงาน ทำให้ต้องใช้เวลาในการทำงานมากกว่าวิธีการแบบ Market based approach

ถึงแม้ข้อดีของการจัดการงานแบบ Market based approach จะสามารถช่วยลดเวลาในการทำงานได้เร็วกว่าวิธีการจัดการงานแบบ FIFO แต่ก็ยังคงมีข้อจำกัดในเรื่องของการนำไปประยุกต์ใช้งานสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม เนื่องจากการจัดการงานแบบ Market based approach จะสามารถจัดการงานได้ดีก็ต่อเมื่อรู้จำนวนชุดคำสั่งงานล่วงหน้ามากกว่าหนึ่งชุดคำสั่ง และจะสามารถจัดการชุดคำสั่งงานได้เมื่อมีจำนวนชุดคำสั่งงานจำนวนหนึ่งแต่หากเป็นงานขนถ่ายวัสดุที่ไม่สามาถระบุชุดคำสั่งงานได้ เช่น ระบบการรับงานแบบ Just In Time การเลือกใช้งานแบบ FIFO จะเหมาะสำหรับนำมาจัดการงานได้ดีกว่า เพราะ ไม่ต้องรอรับคำสั่งงานจนถึงจำนวนหนึ่งแล้วค่อยจัดการงานเหมือนดังเช่นการจัดการงานแบบ Market based approach

สำหรับวิธี Market based เป็นอีกหนึ่งวิธีที่น่าสนใจสำหรับนำมาใช้จัดการชุดคำสั่งงานสำหรับระบบหุ่นยนต์หลายตัว หลายสถานี เพื่อการขนถ่ายวัสดุในโรงงานอุตสาหกรรม เพราะช่วยลดเวลาในการทำงานถึงครึ่งหนึ่งเมื่อเปรียบเทียบกับวิธีแบบ FIFO อีกทั้งวิธี Market based ยังนำข้อมูลของหุ่นยนต์แต่ละแบบมาเปรียบเทียบและใช้ตัดสินในการเลือกงานที่เหมาะสมที่สุดให้กับหุ่นยนต์แต่ละประเภทในระบบได้ โดยคำนวณจากคุณสมบัติของตัวหุ่นยนต์แต่ละตัวและตำแหน่งของหุ่นยนต์มาเปรียบเทียบกัน เพื่อส่งมอบงานให้กับหุ่นยนต์ที่เหมาะสมที่สุดที่ว่างงานอยู่ ณ เวลานั้นแตกต่างกับการทำงานแบบ FIFO ที่ไม่ได้นำเวลาการทำงานของหุ่นยนต์แต่ละตัวมาทำการเปรียบเทียบเพื่อเลือกใช้หุ่นยนต์ให้เหมาะสมกับงาน งานวิจัยนี้สามารถนำไปใช้ขนถ่ายวัสดุในโรงงานอุตสาหกรรมที่ใช้หุ่นยนต์หลายตัวในการขนถ่ายวัสดุหลายชนิดจากสถานีเริ่มต้นไปยังสถานีปลายทางได้มากกว่าหนึ่งสถานี และเหมาะกับโรงงานอุตสาหกรรมที่มีการขนถ่ายวัสดุแบบมีชุดคำสั่งงานล่วงหน้าหลายชุดที่มีผังงานและสถานีคงที่สำหรับการผลิตชิ้นงานในแต่ละรุ่น เช่น โรงงานอุตสาหกรรมการผลิตเครื่องใช้ไฟฟ้า เป็นต้น

เอกสารอ้างอิง

[1] ระบบการขนถ่ายวัสดุ. Thailand industry Web site. 2013. Available at: thailandindustry .com/indust_newweb/articles_preview.php?cid=19078. Accessed April 23, 2016.

[2] Ondrej Stanek, David Obdrálek. Centralized multi-robot system. The Department of Software Engineering; 2012:4-12

[3] Nohýnková St¥pánka. Za roboty Na Homolku. RobotRevue 2010/11. RCR. ISSN 1804-056X

[4] C. J. Malmborg, “A model for the design of zone control automated guided vehicle systems,” International Journal of Production Research, vol. 28, no. 10, pp. 1741–1758, 1990.

[5] F. Tang and L. E. Parker, “A complete methodology for generating multi-robot task solutions using ASyMTRe-D and market-based task allocation,” in 2007 IEEE International Conference on Robotics and Automation, 2007, pp. 3351–3358.

[6] P. K. Das, H. S. Behera, and B. K. Panigrahi, “Intelligent-based multi-robot path planning inspired by improved classical Q-learning and improved particle swarm optimization with perturbed velocity,” International Journal Engineering Science and Technology, vol. 19, no. 1, pp. 651–669, 2016.

[7] ภิญโญ แท้ประสาทสิทธิ์. Graph Algorithm: Data Structure, Algorithms, and Applications. September 6, 2014:67-70

[8] K. Thippeswamy, J. Hanumanthappa, and D. H. Manjaiah, “A study on contrast and comparison between Bellman-Ford algorithm and Dijkstra’s algorithm,” presented at the National Conference on Wireless Networks-09 (NCOWN-2010), 2014, pp. 1–6.

[9] Y. Sharma, S. C. Saini, and M. Bhandhari, “Comparison of Dijkstra’s shortest path algorithm with genetic algorithm for static and dynamic routing network,” International Journal of Electronics and Computer Science Engineering, vol. 1, no. 2, pp. 416–425, 2012.

[10] A. Hussein, M. Adel, M. Bakr, O. M. Shehata, and A. Khamis, “Multi-robot task allocation for search and rescue missions,” in Journal of Physics: Conference Series, 2014, vol. 570, p. 052006.

[11] S. A. Fadzli, S. I. Abdulkadir, M. Makhtar, and A. A. Jamal, “Robotic indoor path planning using Dijkstra,” in 2015 2nd International Conference on Information Science and Security (ICISS), 2015, pp. 1–4.

[12] ROS Introduction. Erle Robotics Web site. 2016. Available at:http://erlerobotics.com/blog/ros-introduction/#What_is_ros. Accessed September 23, 2015.

[13] M. Quigley et al., “ROS: an open-source robot operating system,” in ICRA workshop on open source software, 2009, vol. 3, p. 5.

[14] Z. Zhang, Q. Guo, J. Chen and P. Yuan, "Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification," in IEEE Access, vol. 6, pp. 26022-26035, 2018.

[15] S. Manca, A. Fagiolini, and L. Pallottino, “Decentralized coordination system for multiple agvs in a structured environment,” IFAC Proc. Vol., vol. 44, no. 1, pp. 6005–6010, 2011.

วารสารวิชาการ:

พงศกร ชาญชัยชูจิต และ พฤทธิกร สมิตไมตรี, “ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม,” วารสารวิชาการพระจอมเกล้าพระนครเหนือ, ปีที่ 29, ฉบับที่ 3, ก.ค.–ก.ย. 2562.

วารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ 29 ฉบับที่ 3 ก.ค.–ก.ย. 2562
The Journal of KMUTNB., Vol. 29, No. 3, Jul.–Sep. 2019

บทความวิจัย

ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม

พงศกร ชาญชัยชูจิต และ พฤทธิกร สมิตไมตรี*
ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์

* ผู้นิพนธ์ประสานงาน โทรศัพท์ 0 7428 7214 อีเมล: pruittikorn.s@psu.ac.th DOI: 10.14416/j.kmutnb.2019.xx.xxx
รับเมื่อ 9 กันยายน 2561 แก้ไขเมื่อ 14 พฤศจิกายน 2561 ตอบรับเมื่อ 18 ธันวาคม 2561 เผยแพร่ออนไลน์ x 2562
© 2019 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

บทคัดย่อ

ปัจจุบันการขนถ่ายวัสดุในโรงงานอุตสาหกรรมมีการนำหุ่นยนต์ AGV มาใช้กันอย่างแพร่หลาย ซึ่งพบว่าส่วนใหญ่มักนำหุ่นยนต์ AGV หนึ่งตัวมาใช้ในการขนถ่ายวัสดุเพียงหนึ่งชนิด จากสถานีงานต้นทางไปยังสถานีงานปลายทาง และทำซ้ำแบบเดิมตลอดเวลา งานวิจัยนี้จึงนำเสนอการออกแบบระบบจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุในโรงงานอุตสาหกรรม โดยใช้โปรแกรม ROS (Robot Operating System) เพื่อพัฒนาศักยภาพระบบขนถ่ายวัสดุ เพิ่มความยืดหยุ่น ลดเวลาในการทำงานของหุ่นยนต์ ด้วยวิธี Market-based Approach ร่วมกับ Dijkstra's Algorithm สำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม ผลการจำลองการทำงานพบว่า ระบบสามารถควบคุมหุ่นยนต์ AGV ต่างชนิดกันได้ สามารถเลือกสั่งการหุ่นยนต์ขนถ่ายวัสดุได้คุ้มค่าที่สุดจากการประเมินงานของหุ่นยนต์ทั้งหมดในระบบ ณ สภาวะปัจจุบันของการทำงาน ระบบจัดการนี้สามารถลดเวลาการทำงานได้ครึ่งหนึ่งเมื่อเทียบกับวิธีแบบ FIFO นอกจากนี้ยังพบว่าการเลือกจำนวนหุ่นยนต์ให้กับระบบการจัดการหุ่นยนต์หลายตัวส่งผลต่อความเร็วในการขนถ่ายวัสดุอีกด้วย ผลการจำลองพื้นที่ทำงานขนาด 65 ตารางเมตร แสดงให้เห็นว่าหุ่นยนต์ 3 ตัว สามารถขนถ่ายวัสดุ 100 ชุดคำสั่งงานได้เร็วที่สุด งานวิจัยนี้แสดงให้เห็นว่าระบบสามารถจัดการงานสำหรับการขนถ่ายวัสดุโดยเลือกหุ่นยนต์ทำงานให้มีรอบเวลาทำงาน (Cycle Time) ได้เร็วที่สุด สามารถควบคุมหุ่นยนต์ที่มีลักษณะแตกต่างกันทำให้เกิดความยืดหยุ่นในการขนถ่ายวัสดุ ซึ่งสามารถนำไปใช้ขนถ่ายวัสดุในโรงงานอุตสาหกรรมที่ใช้หุ่นยนต์หลายตัวในการขนถ่ายวัสดุหลายชนิดจากสถานีเริ่มต้นไปยังสถานีปลายทางได้มากกว่าหนึ่งสถานี

คำสำคัญ: ระบบหุ่นยนต์หลายตัว, วิธีการเปรียบเทียบราคาตลาด, ขั้นตอนวิธีของไดก์สตรา, การขนถ่ายวัสดุ

การอ้างอิงบทความ: พงศกร ชาญชัยชูจิต และ พฤทธิกร สมิตไมตรี, "ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงานอุตสาหกรรม," *วารสารวิชาการพระจอมเกล้าพระนครเหนือ*, ปีที่ 29, ฉบับที่ 3, หน้า x–x, ก.ค.–ก.ย. 2562.

วารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ 29 ฉบับที่ 3 ก.ค.–ก.ย. 2562
The Journal of KMUTNB., Vol. 29, No. 3, Jul.–Sep. 2019

Research Article

Management of a Multi-robots System for Industrial Material Handling

Pongsakorn Chanchaichujit and Pruittikorn Smithmaitrie*
Department of Mechanical Engineering, Faculty of Engineering, Prince of Songkla University, Songkhla, Thailand

* Corresponding Author, Tel. 0 7428 7214, E–mail: pruittikorn.s@psu.ac.th
DOI: 10.14416/j.kmutnb.2019.xx.xxx
Received 9 September 2018; Revised 14 November 2018; Accepted 18 December 2018; Published online: x 2019
© 2019 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

Abstract

Recently, material handling of manufacturing factories tends to widely adopt AGV robots. However, AGV robots mostly handle one-by-one material from a start station to a final station, repeatedly. This research introduces a multi-robots material handling system that can handle multiple robots and stations at the same time. The control program is based on ROS (Robot Operating System). The objective of this research is to improve the effectiveness of material handling system by adding more flexibility and reducing the robot work time. The management system uses the market-based approach combined with Dijkstra's algorithm for material handling in a manufacture factory. The simulation results found that the system is able to control various types of AGV robots. This means that the system has ability to order and execute the best robot among all working robots in the system for an incoming task at the current situation. This reduces the total working time by a half comparing with FIFO method. In the case study of the 65-m^2 working area, the optimum number of the robots is 3 to execute 100 tasks. The research results shown that the system can arrange materials handling by choosing the suitable robots for overall fastest cycle time. This system increases flexibility in handling materials in various working spaces. The knowledge of this research can be applied on many factories, which use multiple robots for various tasks to hand materials from start stations to final stations.

Keywords: Multi-robots System, Market-based Approach, Dijkstra's Algorithm, Material Handling

Please cite this article in press as: P. Chanchaichujit and P. Smithmaitrie, "Management of a multi-robots system for industrial material handling," *The Journal of KMUTNB*, vol. 29, no. 3, pp. x–x, Jul.–Sep. 2019 (in Thai).

วารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ 29 ฉบับที่ 3 ก.ค.–ก.ย. 2562
The Journal of KMUTNB., Vol. 29, No. 3, Jul.–Sep. 2019

1. บทนำ

หุ่นยนต์ AGV (Automated Guided Vehicles) ถูกนำมาใช้ช่วยทำงานแทนคน ทั้งแบบอัตโนมัติ แบบเดินตามแถบแม่เหล็ก หรือแบบใช้สัญญาณไร้สายในการระบุตำแหน่งของหุ่นยนต์ [1] ลักษณะการนำหุ่นยนต์มาทำงานร่วมในกระบวนการผลิตมักจะให้หุ่นยนต์ทำงานตามลำดับขั้นตอน FIFO (First In First Out) กล่าวคือ งานที่มาก่อนทำก่อน และงานที่มาหลังทำทีหลัง จนกระทั่งถึงงานลำดับสุดท้าย ซึ่งไม่ได้คำนึงถึงการเพิ่มประสิทธิภาพการขนถ่ายวัสดุ หากในระบบเพิ่มจำนวนหุ่นยนต์มาใช้ในการขนถ่ายวัสดุมากขึ้น จะทำให้ประสิทธิภาพที่ได้ลดลง เนื่องจากขาดการวิเคราะห์เวลาที่ใช้ในการทำงานแต่ละงาน ซึ่งหากวิเคราะห์การจัดการงานของหุ่นยนต์จะพบว่า ยังสามารถเพิ่มการจัดการงานที่เหมาะสมเพื่อช่วยลดเวลาการทำงานของการขนย้ายวัสดุในภาพรวมได้อีก [2]

งานวิจัยเกี่ยวกับระบบการจัดการหุ่นยนต์หลายตัวถูกนำมาใช้เพื่อช่วยจัดการระบบให้มีประสิทธิภาพเพิ่มขึ้นในหลายๆ ด้าน เช่น การควบคุมกลยุทธ์สำหรับหุ่นยนต์หลายตัวส่งผลต่อการช่วยลดเวลาในการทำงาน [3] ปัญหาที่มักพบสำหรับวิธีการจัดเก็บและขนส่งวัตถุดิบทั่วไปอย่าง FIFO คือ ข้อจำกัดเรื่องการใช้เวลารวมมากเกินไปสำหรับการจัดการขนส่งหลายชุดคำสั่ง เพราะไม่ได้เปรียบเทียบและเลือกใช้หุ่นยนต์ที่เหมาะสมกับงานที่สุด ยกตัวอย่างเช่น หุ่นยนต์ที่ว่างงานอยู่ใกล้กับสถานีงานคำสั่งที่สองมากที่สุด แต่ต้องไปรับงานคำสั่งแรกตามวิธี FIFO ทำให้เสียเวลาในการเดินทาง เป็นต้น งานวิจัยนี้จึงมุ่งเน้นการออกแบบระบบการจัดการหุ่นยนต์หลายตัวสำหรับขนถ่ายวัสดุด้วยวิธี Market-based Approach โดยวิธีการนี้จะแปลงระบบเส้นทางให้อยู่ในรูปแบบกริด จุดปลายของเส้นทางเปรียบเสมือนสินค้าที่มีการแลกเปลี่ยนอยู่ในตลาด และระยะทางระหว่างจุดเริ่มต้นไปยังจุดปลายทางเปรียบเสมือนต้นทุนที่ใช้ กำไรจะขึ้นอยู่กับข้อมูลต่างๆ ที่ใช้ประกอบไปจนถึงจุดปลายทาง งานวิจัยนี้ใช้เวลาในการทำงานของหุ่นยนต์เป็นตัวชี้วัดเพื่อเลือกงานให้เหมาะสมกับตำแหน่งปัจจุบันของหุ่นยนต์ โดยหุ่นยนต์ทุกตัวจะมีเส้นทางการเดินที่สั้นที่สุดซึ่งในกรณีนี้วิธี Market-based Approach ก็คือการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน

วิธีหาเส้นทางแบบกริดที่สั้นที่สุดนั้นมีอยู่หลายอัลกอรึทึม งานวิจัยนี้ได้เลือกใช้ขั้นตอนวิธีของไดก์สตรา (Dijkstra's Algorithm) เพราะเข้ากับรูปแบบของระบบขนส่งวัสดุด้วยหุ่นยนต์หลายตัว รวมถึงเวลาที่ใช้ในการประมวลหาระยะทางที่สั้นที่สุดก็อยู่ในระดับที่ไม่นานจนเกินไป และเมื่อมีจำนวนโหนดในการประมวลผลมากขึ้น ระยะเวลาในการคำนวณก็ไม่ได้เพิ่มขึ้นมากนัก เมื่อทำการเปรียบเทียบกับอัลกอรึทึมเบลแมน-ฟอร์ด [4] ผลลัพธ์จากการหาระยะทางที่สั้นที่สุดเมื่อเทียบกับ Genetic Algorithm (GA) มีค่าเท่ากัน [5] สำหรับวิธีของไดก์สตรานั้นเป็นวิธีการหาเส้นทางที่สั้นที่สุด โดยมีงานวิจัยที่นำวิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน ร่วมกับอัลกอริทึมการหาเส้นทางที่สั้นที่สุด เช่น การจัดสรรงานหุ่นยนต์หลายตัวสำหรับค้นหาและช่วยเหลือ [6] โดยใช้ค่าน้ำหนักระหว่างโหนด เมื่อมีการนำค่าน้ำหนักระหว่างโหนดแทนด้วยระยะทางระหว่างโหนด จึงสามารถนำมาใช้ในการหาระยะทางที่สั้นที่สุดของหุ่นยนต์ได้ [7] แต่ระยะทางที่ใช้ในการเดินทางที่สั้นที่สุดนั้นไม่เพียงพอต่อการนำไปตัดสินใจเลือกหุ่นยนต์ งานวิจัยนี้จึงนำเรื่องของเวลาขนถ่ายวัสดุมาช่วยในการตัดสินใจ เพราะทำให้เงื่อนไขการตัดสินใจนั้นขึ้นอยู่กับตัวแปรสองตัวคือ ความเร็วในการทำงาน และระยะทางของหุ่นยนต์

ในการพัฒนาระบบจัดการหุ่นยนต์หลายตัวพบว่ามีความยุ่งยากในเรื่องของการสื่อสาร การจัดการทรัพยากรของระบบจัดการหุ่นยนต์ การสื่อสารผ่านระบบเครือข่าย รวมถึงหากต้องการควบคุมหุ่นยนต์ที่แต่ละตัวมีลักษณะแตกต่างกันยิ่งเพิ่มความซับซ้อนในระบบควบคุมมากยิ่งขึ้น แต่ในปัจจุบันมีการพัฒนาซอฟต์แวร์ที่ใช้ในการสื่อสารกันแบบ Machine-to-machine นั่นคือ ROS (Robot Operating System) ข้อดีของโปรแกรม ROS คือ มีความยืดหยุ่นในการควบคุมหุ่นยนต์ต่างชนิดหลายตัวพร้อมกันได้ รวมถึงเป็นโอเพนซอร์สที่สามารถใช้งานได้ฟรี และถูกขยายผลไปใช้งานในอุตสาหกรรมมากขึ้นในปัจจุบัน [8]

ขั้นตอนวิธีของไดก์สตรากำหนดให้ G คือ กราฟที่มีค่าถ่วงน้ำหนัก โดยกำหนดให้ทุกทางมีค่าถ่วงน้ำหนักเป็นบวก G ประกอบด้วยจุดต่างๆ คือ $v_0, v_1, \ldots, v_n$ และมีค่าถ่วงน้ำหนักเท่ากับ $w(u_i, v_j)$ เมื่อ u_i คือ พิกัดปัจจุบัน และ V_j คือ พิกัดถัดไป โดยให้ a เป็นจุดเริ่มต้น มี z เป็นจุดปลายทาง กำหนดให้ S คือ เซตของจุดซึ่งเก็บค่าจุดที่เคยเดินผ่านมาแล้ว [9]

ขั้นตอนของ Dijkstra's Algorithm

For $i = 1$ to n

$L(v_1) = \infty$

$L(a) = 0$

$S = \varnothing$

While $z \notin S$

Begin

U = จุดที่ไม่ได้อยู่ใน S และมีระยะทางน้อยที่สุดจาก $L(u)$

$S = S \cup \{u\}$

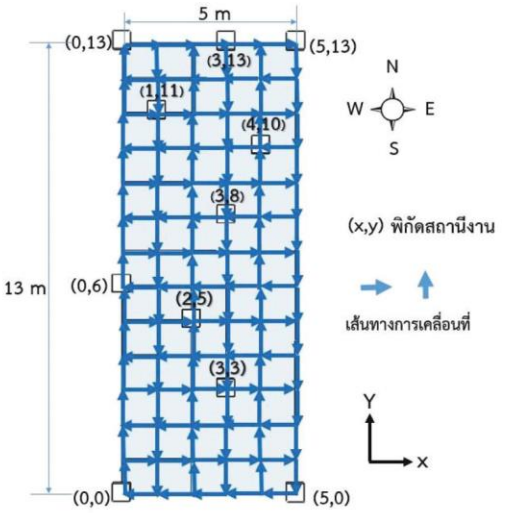
For ทุกจุดของ v ที่ไม่ได้อยู่ใน S

If $L(u) + W(u,v) < L(v)$ *Then*

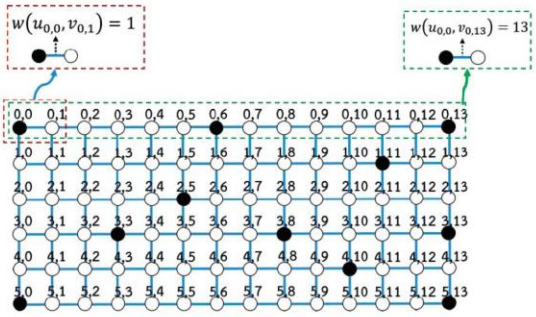
$L(v) = L(u) + W(u,v)$

Return $L(z)$

END



รูปที่ 1 แผนที่จำลองในโรงงานอุตสาหกรรมสำหรับทดสอบการจัดการงาน



รูปที่ 2 แผนภูมิที่สร้างจากขั้นตอนวิธีของไดก์สตรา

2. วิธีการวิจัย

สำหรับขั้นตอนวิธีของไดก์สตรานั้นถูกนำมาใช้ในการหาผลรวมของเส้นทางที่สั้นที่สุดของหุ่นยนต์จากสถานีเริ่มต้นไปยังสถานีปลายทาง วิธีการหาเส้นทางที่สั้นที่สุดหาจากแผนที่แบบตารางแสดงดังรูปที่ 1

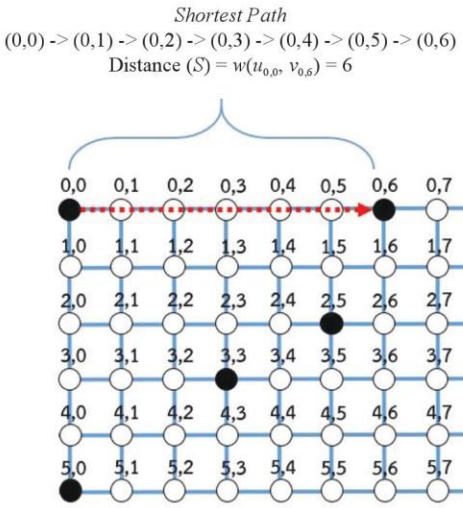
งานวิจัยนี้จะใช้แผนที่ดังรูปที่ 1 ในการทดสอบระบบจัดการงานให้กับหุ่นยนต์ ซึ่งขนาดของแผนที่มีขนาดกว้าง 5 เมตร ยาว 13 เมตร ประกอบด้วยสถานีงานทั้งหมด 11 สถานี สำหรับค่าถ่วงน้ำหนักที่ใช้แทนในการขั้นตอนวิธีของไดก์สตรา แทนด้วยค่าระยะห่างของจุดแต่ละจุดเท่ากับ 1 เมตร และแทนจุดแต่ละจุดบนเส้นทางด้วยพิกัด (x, y)

เมื่อใช้ขั้นตอนวิธีของไดก์สตรา จะได้แผนผังสำหรับการหาเส้นทางที่สั้นที่สุดดังแสดงในรูปที่ 2 ทดสอบหาระยะทางที่สั้นที่สุดจากสถานีพิกัด (0, 0) ไปยังพิกัด (0, 6) เมื่อจุดดำแทนสถานีงาน และจุดขาวแทนตำแหน่งเส้นทางเดิน ผลจากทดสอบแสดงได้ดังรูปที่ 3 จะได้ระยะทางในการเดินทางจากสถานีพิกัด (0, 0) ไปสถานีพิกัด (0, 6) เป็นระยะทาง 6 เมตร โดยผ่านเส้นทางพิกัด (0, 1), (0, 2), (0, 3), (0, 4), (0, 5) ตามลำดับ

3. ระบบการจัดการงาน

ระบบการจัดการงานสำหรับหุ่นยนต์หลายตัวแบ่งออกเป็นสองประเภท คือ ระบบศูนย์กลางหุ่นยนต์หลายตัว

วารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ 29 ฉบับที่ 3 ก.ค.–ก.ย. 2562
The Journal of KMUTNB., Vol. 29, No. 3, Jul.–Sep. 2019



รูปที่ 3 เส้นทางที่สั้นที่สุดจากสถานีพิกัด (0, 0) ไปยังสถานีพิกัด (0, 6)

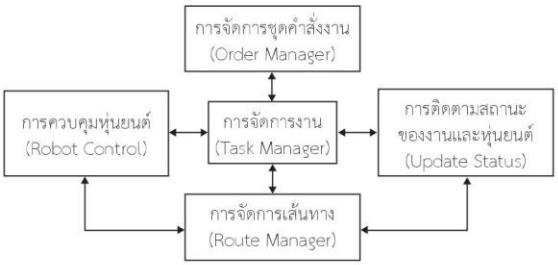
(Centralized Multi-robot System) และระบบไม่มีศูนย์กลางหุ่นยนต์หลายตัว (Decenrtalized Multi-robot System) งานวิจัยนี้เลือกใช้แบบระบบศูนย์กลางหุ่นยนต์หลายตัว เพราะสามารถควบคุมและจัดการได้ง่ายเหมาะสำหรับสภาวะแวดล้อมเป็นปกติไม่เปลี่ยนแปลง จำนวนหุ่นยนต์คงที่ สำหรับระบบการจัดการงานของงานวิจัยนี้แบ่งการจัดการงานออกเป็น 5 ส่วน คือ การจัดการชุดคำสั่งงาน การติดตามสถานะของงานและหุ่นยนต์ การจัดการงาน การควบคุมหุ่นยนต์ และการจัดการเส้นทาง ดังรูปที่ 4

3.1 การจัดการชุดคำสั่งงาน (Order Manager)

เป็นส่วนที่ให้ผู้ใช้ป้อนชุดคำสั่งงานเข้าสู่ระบบ เพื่อให้ระบบทำหน้าที่จัดการงาน

3.2 การติดตามสถานะของงานและหุ่นยนต์ (Update Status)

เป็นส่วนที่จัดการในเรื่องของการอัปเดทสถานะของหุ่นยนต์และชุดคำสั่งงานที่หุ่นยนต์แต่ละตัวกำลังทำงานอยู่ เพื่อให้ส่วนของการจัดการงานสามารถตรวจสอบสถานะของงานและสถานะของหุ่นยนต์ ณ เวลาปัจจุบันได้



รูปที่ 4 ภาพรวมของระบบจัดการงาน

3.3 การจัดการงาน (Task Manager)

เป็นส่วนหลักที่ใช้ในการคำนวณเวลาสำหรับชุดคำสั่งงานและตัดสินใจเลือกหุ่นยนต์ที่ทำงานได้รวดเร็วที่สุด สำหรับการประเมินค่าเวลาการทำงานของหุ่นยนต์จะเริ่มคำนวณหลังจากที่ได้เส้นทางที่สั้นที่สุดในการเคลื่อนที่ของหุ่นยนต์แต่ละงานจากขั้นตอนของไดก์สตรา จากนั้นนำมาคำนวณหาเวลาการทำงานของหุ่นยนต์ในแต่ละงานได้จากสมการที่ (1)

$$T_{m,n} = \left(M \frac{S_n}{V_n} \right) + \left(N \frac{\theta_n}{\omega_n} \right) \quad (1)$$

เมื่อ T คือ เวลาในการทำงานของหุ่นยนต์
S คือ ผลรวมของระยะทางที่สั้นที่สุดของหุ่นยนต์
V คือ ความเร็วเชิงเส้นของหุ่นยนต์
θ คือ ผลรวมของมุมการเคลื่อนที่ของหุ่นยนต์
ω คือ ความเร็วเชิงมุมของหุ่นยนต์
n คือ หุ่นยนต์ตัวที่ n
m คือ งานที่ m
M คือ จำนวนครั้งของการเคลื่อนที่แบบเชิงเส้น
N คือ จำนวนครั้งของการเคลื่อนที่แบบเชิงมุม

สมการที่ (1) นำความเร็วของหุ่นยนต์ทั้งความเร็วเชิงเส้นและเชิงมุมที่ใช้ในการเคลื่อนที่มาแปลงให้อยู่ในรูปของเวลาเพื่อนำมาเลือกหุ่นยนต์ที่ใช้เวลาทำงานน้อยที่สุด

อย่างไรก็ตาม แผนที่ที่ใช้ในการทดลองนี้มีระยะห่างระหว่างจุดพิกัดเป็นระยะคงที่ ดังนั้นหากนำไปใช้ในการคำนวณเวลาสำหรับแผนผังที่มีระยะห่างของจุดพิกัดไม่คงที่จะต้องมีการปรับสมการให้ถูกต้องตามระยะห่างของจุดพิกัดด้วย

3.4 วิธีการเปรียบเทียบราคาตลาด (Market-based Approach)

วิธีการนี้เป็นหนึ่งในการประเมินมูลค่าของสินค้าโดยอาศัยการเปรียบเทียบข้อมูลต่างๆ เช่น ประเภทของสินค้า ขนาด ตำแหน่งที่ตั้ง ราคา ที่มีลักษณะคล้ายคลึงกันในด้านต่างๆ แล้วทำการวิเคราะห์เพื่อสรุปผลเป็นมูลค่าของสินค้านั้น การเปรียบเทียบราคาตลาดมีวิธีย่อยหลายวิธี เช่น วิธีการเปรียบเทียบโดยตรง (Direct Comparison) วิธีปรับเปลี่ยนราคาซื้อขาย (Grid Adjustment) วิธีคะแนนคุณภาพถ่วงน้ำหนัก (Weighted Quality Score; WQS) และวิธีสมการถดถอย (Regression Analysis) [10] มีหลายงานวิจัยนำวิธีการเหล่านี้มาใช้ในการจัดการหุ่นยนต์หลายตัว โดยการนำค่าข้อมูลตำแหน่ง ระยะห่างของหุ่นยนต์ และเวลาการทำงานของหุ่นยนต์แต่ละตัวมาเปรียบเทียบกันด้วยการ Bidding ระหว่างหุ่นยนต์ แล้วใช้วิธีเลือกค่าดัชนีกำหนดแบบน้อยที่สุดหรือมากที่สุดอย่างใดอย่างหนึ่ง เพื่อเลือกหุ่นยนต์ที่เหมาะสมกับงาน [11]–[13]

เวลาทำงานของหุ่นยนต์ที่หาได้จากสมการที่ (1) ถูกนำมาใช้ในการ Bidding เพื่อเลือกงานที่เหมาะสมกับหุ่นยนต์ด้วยระยะทางที่สั้นที่สุดและการทำงานเร็วที่สุด โดยพิจารณาตัวแปร 3 ตัว คือ ระยะทางจากตำแหน่งปัจจุบันของหุ่นยนต์ ความเร็วเชิงเส้น และความเร็วเชิงมุมของหุ่นยนต์ มาประเมินเป็นเวลาที่ใช้ในการทำงานของหุ่นยนต์ตามสมการที่ (1) จะได้เซตเวลาของหุ่นยนต์แต่ละตัวในระบบที่ใช้ในแต่ละงานตามสมการที่ (2) ซึ่งเป็นการแสดงเวลาที่หุ่นยนต์แต่ละตัวต้องใช้ทำงานในแต่ละงานจากตำแหน่งปัจจุบันที่หุ่นยนต์อยู่

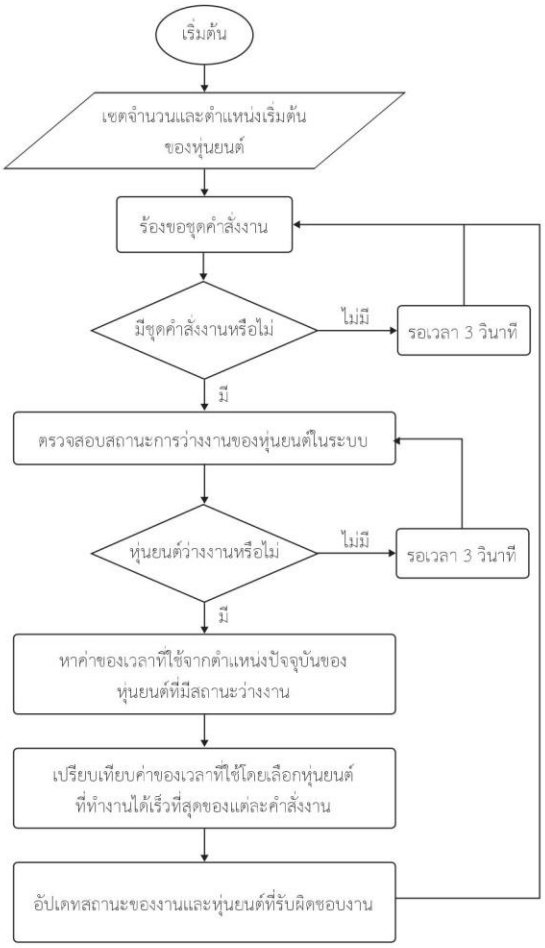
$$J_m = \{R_n : T_m\} \tag{2}$$

เมื่อ J คือ เซตของงาน $\{J_1, J_2, \ldots, J_m\}$

R คือ หุ่นยนต์ในระบบ

T คือ เวลาที่ถูกประเมินสำหรับงาน

จากนั้นนำเซตเวลาการทำงานของหุ่นยนต์ดังสมการที่ (2) มาเปรียบเทียบหาเวลาที่น้อยที่สุด สำหรับเลือกงานให้กับหุ่นยนต์ด้วยการนำค่าเวลาในการทำงานของหุ่นยนต์แต่ละตัวมา Bidding สำหรับชุดงานนั้นๆ ดังสมการที่ (3)



รูปที่ 5 ขั้นตอนการทำงานของการจัดการงาน

$$Task_i = \min(J_m) \tag{3}$$

เมื่อ i คือ ลำดับการมอบหมายงาน

เมื่อสามารถทำการ Bidding งานให้กับหุ่นยนต์ได้แล้ว ระบบการจัดการจะทำการมอบหมายงานนั้นๆ ให้กับหุ่นยนต์ตัวที่เหมาะสม และประเมินเวลางานใหม่ที่เข้ามาให้กับหุ่นยนต์ที่ว่างงานอยู่กระทำซ้ำเช่นนี้จนมอบหมายงานที่มีอยู่ทั้งหมดให้หุ่นยนต์ทำงานจนเสร็จสิ้น ขั้นตอนการทำงานของการจัดการงานแสดงดังรูปที่ 5

วารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ 29 ฉบับที่ 3 ก.ค.–ก.ย. 2562
The Journal of KMUTNB., Vol. 29, No. 3, Jul.–Sep. 2019

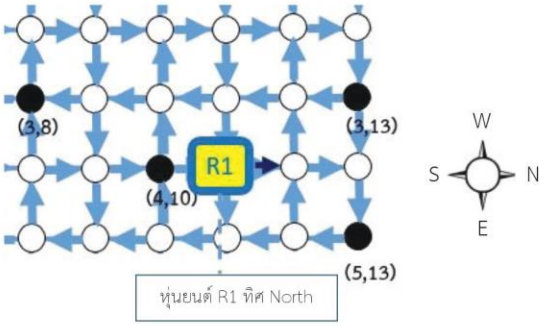
3.5 การควบคุมหุ่นยนต์ (Robot Control)

เมื่อระบบควบคุมและจัดการงานมีหุ่นยนต์ที่แตกต่างกันทำงานร่วมกัน จะส่งผลถึงการควบคุมการเคลื่อนที่ของหุ่นยนต์ที่มีความแตกต่างกัน เช่น ระบบของล้อหุ่นยนต์เป็นแบบ Differential Drive [14] หรือแบบ Mecanum Drive [15] ทำให้ระบบสั่งงานการเคลื่อนที่ของหุ่นยนต์มีตัวแปรที่ต่างกันและมีความซับซ้อนมากขึ้น เพื่อให้ง่ายต่อการสั่งการของระบบการจัดการ จึงอาศัยการส่งคำสั่งรูปแบบการเคลื่อนที่แทนการส่งตำแหน่งปลายทางให้กับหุ่นยนต์ โดยส่วนการควบคุมการเคลื่อนที่จะทำหน้าที่บังคับการเคลื่อนที่ของหุ่นยนต์ตัวนั้นแทน เพื่อให้ตัวระบบการจัดการงานกับการควบคุมการเคลื่อนที่ของหุ่นยนต์ถูกแยกออกจากกันแต่ยังสามารถทำงานร่วมกันได้ และสร้างการควบคุมหุ่นยนต์แต่ละตัวให้มีฟังก์ชันการเคลื่อนที่ตามระบบขับเคลื่อนล้อของตนเอง โดยมีลักษณะฟังก์ชันการเคลื่อนที่ของหุ่นยนต์ทั้งหมด 3 รูปแบบ คือ Forward (เดินหน้า) Left (หุ่นยนต์เลี้ยวซ้ายเป็นมุม 90 องศา) Right (หุ่นยนต์เลี้ยวขวาเป็นมุม 90 องศา) และ Turnaround (หุ่นยนต์หมุนตามเข็มนาฬิกาเป็นมุม 180 องศา) ระบบการจัดการหุ่นยนต์จะใช้ข้อมูลตำแหน่งปัจจุบันแทนด้วย (x_i, y_i) ตำแหน่งถัดไปแทนด้วย (x_j, y_j) และทิศทางของหุ่นยนต์แทนด้วย North, East, West, South จากนั้นหาความสัมพันธ์ของตำแหน่งถัดไปและตำแหน่งปัจจุบัน โดยการหาค่า x และ y จากสมการที่ (4) และ (5) ตามลำดับ

$$x = x_j - x_i \tag{4}$$

$$y = y_j - y_i \tag{5}$$

แล้วนำค่าที่ได้ไปเทียบกับตารางที่ 1 เพื่อแปลงเส้นทางการเคลื่อนที่ที่สั้นที่สุดจากขั้นตอนของไดก์สตรา เป็นรูปแบบการเคลื่อนที่ของหุ่นยนต์ ตัวอย่างเช่น จากรูปที่ 6 หุ่นยนต์ R1 ตำแหน่งปัจจุบัน คือ (4,11) ทิศ North มีพิกัดเป้าหมายตำแหน่งถัดไป คือ (4,12) หาค่า x และ y จากสมการที่ (4) และ (5) จะได้ว่า x = 0, y = 1 เมื่อนำค่าไปแทนในตารางที่ 1 จะได้ชุดคำสั่งการเคลื่อนที่ คือ Forward เป็นต้น



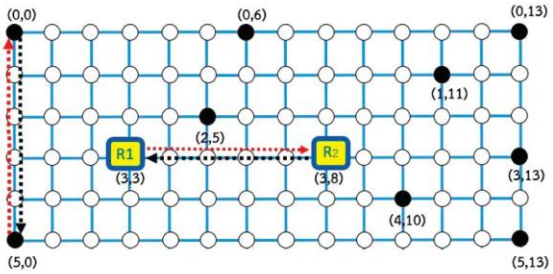
รูปที่ 6 การระบุทิศและการเคลื่อนที่ของหุ่นยนต์เทียบกับแผนที่

ตารางที่ 1 การแปลงเส้นทางเป็นรูปแบบการเคลื่อนที่

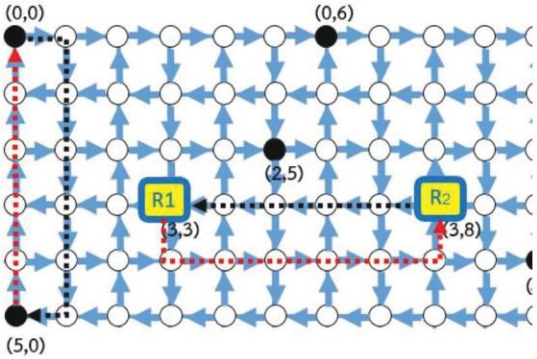
ทิศทางของหุ่นยนต์ ณ ตำแหน่งปัจจุบัน	ความสัมพันธ์ของตำแหน่งถัดไปกับตำแหน่งปัจจุบันของหุ่นยนต์		คำสั่งการเคลื่อนที่
	x	y	
North	0	1	Forward
	0	−1	Turnaround and Forward
	−1	0	Left and Forward
	1	0	Right and Forward
East	0	1	Left and Forward
	0	−1	Right and Forward
	−1	0	Turnaround and Forward
	1	0	Forward
West	0	1	Right and Forward
	0	−1	Left and Forward
	−1	0	Forward
	1	0	Turnaround and Forward
South	0	1	Turnaround and Forward
	0	−1	Forward
	−1	0	Right and Forward
	1	0	Left and Forward

3.6 การจัดการเส้นทางของหุ่นยนต์ (Route Manager)

เมื่อหุ่นยนต์ได้รับงานที่มอบหมายแล้ว อีกเหตุการณ์หนึ่งที่จะเกิดขึ้น คือ ระบบจัดการงานหาเส้นทางที่สั้นที่สุดได้ผลเป็นเส้นทางที่ทับซ้อนกันของหุ่นยนต์สองตัว ทำให้เกิดปัญหาการจราจรขณะทำงาน ตัวอย่างเช่นในรูปที่ 7 เมื่อหุ่นยนต์ตัวที่หนึ่งต้องการนำสินค้าจากสถานีพิกัด (3, 3) ไปยังสถานี



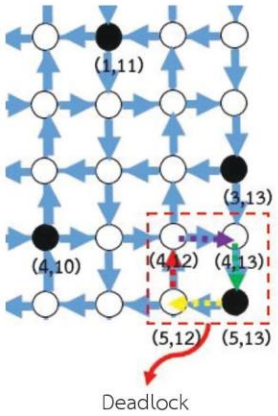
รูปที่ 7 ปัญหาจราจรเมื่อหุ่นยนต์สองตัวทำงานบนเส้นทางที่ทับกัน



รูปที่ 8 การแก้ปัญหาการจราจรด้วยการกำหนดเส้นทางเดินทิศทางเดียว

พิกัด (3, 8) และหุ่นยนต์ตัวที่สองต้องการนำสินค้าจากพิกัด (3, 8) ไปส่งยังพิกัด (3, 3) ผลคำนวณเส้นทางที่สั้นที่สุดของหุ่นยนต์ทั้งสองตัวพบว่าเกิดเส้นทางทับซ้อนกันบนเส้นทางเดียวกัน จึงทำให้เกิดปัญหาการจราจรขึ้น

งานวิจัยนี้ใช้วิธีการแก้ปัญหาด้วยการสร้างเส้นทางเดินแบบทางเดียว (Unidirectional) [16] โดยทุกจุดบนแผนที่แบบตารางจะต้องมีทางเข้าและออกอย่างน้อย 1 ทาง และมีจำนวนทางเดินในแนวกว้างและยาวเป็นจำนวนคู่เสมอ หากเกิดกรณีหุ่นยนต์ต้องใช้จุดถัดไปร่วมกัน หุ่นยนต์ที่ได้รับคำสั่งเข้าถึงการจองจุดถัดไปก่อน จะมีสิทธิ์จองพื้นที่นั้นเพื่อใช้เส้นทาง หุ่นยนต์อีกตัวจะต้องหยุดรอจนกว่าจุดถัดไปนั้นจะว่าง แล้วจึงทำการจองเพื่อใช้เส้นทางนั้นในลำดับถัดไป ด้วยเงื่อนไขนี้เส้นทางใหม่ที่ได้จากการกำหนดเส้นทางแบบเดินทางเดียวจะได้ผลการคำนวณเส้นทางดังรูปที่ 8



รูปที่ 9 เหตุการณ์ Deadlock ของหุ่นยนต์ 4 ตัวที่ใช้เส้นทางพร้อมกัน

นอกจากนี้ระหว่างทำการทดลองได้พบกรณีของ Deadlock เมื่อหุ่นยนต์สี่ตัวจองใช้เส้นทางพร้อมกัน ดังรูปที่ 9 โดยหุ่นยนต์ 4 ตัว มีการใช้จุดจราจรร่วมกัน คือ หุ่นยนต์ตัวที่หนึ่งต้องการเคลื่อนตัวจาก (4, 12) ไปยังจุด (4, 13) หุ่นยนต์ตัวที่สองจากจุด (4, 13) ไปยังจุด (5, 13) หุ่นยนต์ตัวที่สามจาก (5, 13) ไปยัง (5, 12) และหุ่นยนต์ตัวที่สี่จากจุด (5, 12) ไปยัง (4, 12) กรณีนี้ Manca *et al*. [17] ได้แสดงวิธีการแก้ปัญหาไว้แล้ว กล่าวคือ ระบบจัดการเส้นทางจะทำการตรวจสอบการใช้พิกัดร่วมกันก่อนจะเกิดเหตุการณ์ดังกล่าวโดยหากมีหุ่นยนต์ใช้พิกัดที่ใช้ร่วมกันมากกว่า 2 จุด เช่น หุ่นยนต์ตัวที่หนึ่งที่ต้องการเคลื่อนตัวจากจุด (4, 12) ไปยังจุด (4, 13) ตรวจสอบพบว่าหุ่นยนต์ตัวที่สองต้องการใช้จุด (4,13) และหุ่นยนต์ตัวที่สี่ต้องการใช้จุด (4, 12) ระบบจัดการเส้นทางจะรอจนกว่าหุ่นยนต์ตัวที่หนึ่งเคลื่อนที่ไปยังจุด (4, 13) จากนั้นจึงจะอนุญาตให้หุ่นยนต์ตัวที่สี่หรือสองสามารถทำงานต่อไปได้

4. ผลการทดลอง

เริ่มต้นจากการทดลองเปรียบเทียบจำนวนหุ่นยนต์ที่เหมาะสมในระบบโดยการคำนวณเวลาที่ใช้ทำงานของระบบด้วยจำนวนหุ่นยนต์หนึ่งตัวถึงสี่ตัวทำงานแบบ FIFO เปรียบเทียบกับวิธี Market-based Approach โดยเพิ่ม

วารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ 29 ฉบับที่ 3 ก.ค.–ก.ย. 2562
The Journal of KMUTNB., Vol. 29, No. 3, Jul.–Sep. 2019

จำนวนชุดคำสั่งงานครั้งละ 10 งาน จนถึง 100 งาน โดยงานที่ใช้ในการจำลองอ้างอิงจากสถานีงานทั้งหมด 11 สถานี (จุด) ยกตัวอย่างการสร้างชุดคำสั่งงานดังตารางที่ 2 โดยมีสถานีเริ่มต้นและสิ้นสุดที่ได้มาจากการสุ่ม กำหนดให้หุ่นยนต์มีความเร็วเชิงเส้น 1 เมตร/วินาที ความเร็วเชิงมุม 1.57 เรเดียน/วินาที สำหรับตำแหน่งเริ่มต้นนั้นจะกำหนดตำแหน่งหุ่นยนต์ให้กระจายทั่วแผนที่ โดยให้ตำแหน่งเริ่มต้นของหุ่นยนต์ตัวที่หนึ่งอยู่ที่พิกัด 3, 13 หุ่นยนต์ตัวที่สองอยู่ที่พิกัด (0, 0) หุ่นยนต์ตัวที่สามอยู่พิกัด (4, 7) หุ่นยนต์ตัวที่สี่อยู่พิกัด (5, 3) ทำงานบนแผนที่จำลองโรงงานอุตสาหกรรมตามรูปที่ 1

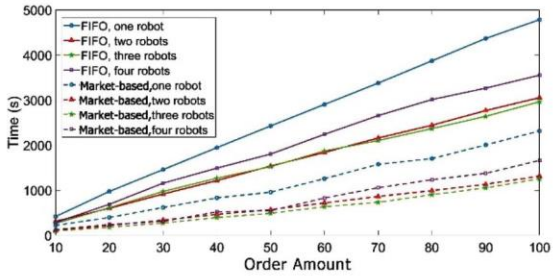
ตารางที่ 2 ตัวอย่างการสร้างชุดคำสั่งงาน

คำสั่งงาน	สถานีเริ่มต้น	สถานีปลายทาง
01	0,0	0,6
02	0,6	0,13
03	0,13	1,11
04	1,11	2,5
05	2,5	3,3
06	3,3	3,8
07	3,8	3,13
08	3,13	4,10
09	4,10	5,0
10	5,0	5,13

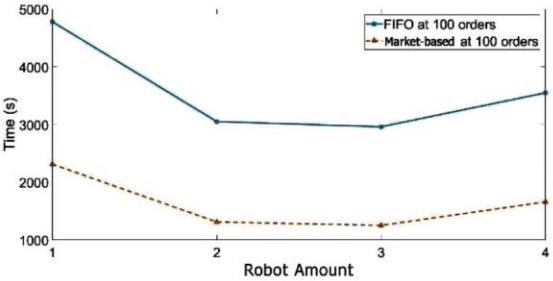
ผลการจำลองการทำงานของโปรแกรมระบบควบคุมด้วย ROS แสดงให้เห็นความสัมพันธ์ของจำนวนงานและเวลาที่ใช้ในการจัดการงานทั้งหมดของทั้งสองวิธีเปรียบเทียบกันดังรูปที่ 10

จากผลการทดลองพบว่าวิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน ใช้เวลาในการจัดการงานทั้งหมดเสร็จสิ้นได้เร็วกว่าวิธีแบบ FIFO ถึง 1 เท่าตัว เมื่อเปรียบเทียบด้วยกรณีที่มีหุ่นยนต์จำนวนทำงานเท่ากันแต่วิธีจัดการงานต่างกัน และเวลาที่ใช้ในการทำภารกิจจนเสร็จสิ้น จะเพิ่มขึ้นเป็นเชิงเส้นเมื่อจำนวนชุดคำสั่งเพิ่มขึ้น

นอกจากนี้เมื่อพิจารณาที่จำนวนชุดคำสั่งเท่ากัน รูปที่ 11 แสดงให้เห็นว่าเมื่อหุ่นยนต์ในระบบมีจำนวนเพิ่มขึ้น เวลาที่



รูปที่ 10 เวลาที่ใช้ในการทำงานของระบบหุ่นยนต์หลายตัวแบบ FIFO เปรียบเทียบกับวิธี Market-based Approach



รูปที่ 11 เวลาการทำงานของระบบจัดการหุ่นยนต์ที่จำนวนชุดคำสั่งเท่ากัน เมื่อจำนวนหุ่นยนต์ในระบบเพิ่มขึ้น

ใช้ในการจัดการงานทั้งหมดจนเสร็จสิ้นจะลดลงจนถึงจุดหนึ่งเท่านั้น เพราะหากมีจำนวนหุ่นยนต์เยอะจนเกินไปจะทำให้เกิดการจราจรแออัด หุ่นยนต์ต้องเสียเวลารอเส้นทางว่าง เป็นผลให้ทั้งระบบใช้เวลามากกว่าที่ควรจะเป็น จากการทดลองการจัดการงานที่ 100 ชุดคำสั่งงาน โดยเพิ่มจำนวนหุ่นยนต์ในระบบจาก 1 ถึง 4 ตัว พบว่าสำหรับพื้นที่การทำงานที่กว้าง 5 เมตร ยาว 13 เมตร สำหรับการจัดการชุดคำสั่ง 100 งาน การเลือกใช้หุ่นยนต์ทำงาน 3 ตัว ด้วยวิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงานสามารถจัดการงานได้เร็วที่สุดด้วยเวลา 1,252 วินาที ใกล้เคียงกับหุ่นยนต์จำนวน 2 ตัว

5. สรุป

จากผลการจำลองการจัดการงานด้วยวิธีแบบ FIFO เปรียบเทียบกับวิธี Market-based Approach หรือการ

เลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน พบว่า วิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน ทำงานสำเร็จได้เร็วกว่า รวมถึงจำนวนหุ่นยนต์ที่เพิ่มขึ้น สามารถช่วยลดเวลาในการทำงานได้ ซึ่งจากการทดลองสำหรับจำนวนงานทั้งหมด 100 งาน ด้วยวิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน ระบบสามารถลดเวลาการทำงานลงครึ่งหนึ่งเมื่อเทียบกับระบบ FIFO นอกจากนี้ยังพบว่าการนำหุ่นยนต์หลายตัวมาใช้ในการทำงาน จำนวนหุ่นยนต์ที่มากขึ้นไม่ได้ส่งผลให้เวลาในการทำงานลดลงเสมอไป เมื่อมีหุ่นยนต์มากถึงจำนวนหนึ่งก็จะส่งผลให้ระบบการทำงานช้าลงได้ด้วย ยกตัวอย่างเช่น เวลาที่ใช้ในการคำนวณเส้นทางมากขึ้น ปัญหาการจัดการจราจรที่พบบ่อยขึ้น ดังนั้นควรให้ความสำคัญกับจำนวนหุ่นยนต์ที่เลือกใช้ในการขนถ่ายวัสดุในอุตสาหกรรมด้วยเช่นกัน

สำหรับวิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน เป็นอีกหนึ่งวิธีที่น่าสนใจสำหรับนำมาใช้จัดการชุดคำสั่งงานสำหรับระบบหุ่นยนต์หลายตัว หลายสถานี เพื่อการขนถ่ายวัสดุในโรงงานอุตสาหกรรม เพราะช่วยลดเวลาในการทำงานถึงครึ่งหนึ่งเมื่อเปรียบเทียบกับวิธีแบบ FIFO อีกทั้งวิธีการเลือกหุ่นยนต์ที่ใช้เวลาน้อยที่สุดไปทำงาน ยังนำข้อมูลของหุ่นยนต์แต่ละแบบมาเปรียบเทียบและใช้ตัดสินในการเลือกงานที่เหมาะสมที่สุดให้กับหุ่นยนต์แต่ละประเภทในระบบได้ โดยคำนวณจากคุณสมบัติของตัวหุ่นยนต์แต่ละตัวและตำแหน่งของหุ่นยนต์มาเปรียบเทียบกัน เพื่อส่งมอบงานให้กับหุ่นยนต์ที่เหมาะสมที่สุดที่ว่างงานอยู่ ณ เวลานั้น ซึ่งแก้ไขข้อบกพร่องของการสั่งงานแบบ FIFO ที่ไม่ได้นำเวลาการทำงานของหุ่นยนต์แต่ละตัวมาเปรียบเทียบเพื่อเลือกใช้หุ่นยนต์ให้เหมาะสมกับงาน

งานวิจัยนี้สามารถนำไปใช้กับการขนถ่ายวัสดุในโรงงานอุตสาหกรรมที่มีหุ่นยนต์หลายตัวในการขนถ่ายวัสดุหลายชนิด จากสถานีเริ่มต้นไปยังสถานีปลายทางได้มากกว่าหนึ่งสถานี

เอกสารอ้างอิง

[1] C. J. Malmborg, “A model for the design of zone control automated guided vehicle systems,” *International Journal of Production Research*, vol. 28, no. 10, pp. 1741–1758, 1990.

[2] F. Tang and L. E. Parker, “A complete methodology for generating multi-robot task solutions using ASyMTRe-D and market-based task allocation,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 3351–3358.

[3] P. K. Das, H. S. Behera, and B. K. Panigrahi, “Intelligent-based multi-robot path planning inspired by improved classical Q-learning and improved particle swarm optimization with perturbed velocity,” *International Journal Engineering Science and Technology*, vol. 19, no. 1, pp. 651–669, 2016.

[4] K. Thippeswamy, J. Hanumanthappa, and D. H. Manjaiah, “A study on contrast and comparison between Bellman-Ford algorithm and Dijkstra’s algorithm,” presented at the National Conference on Wireless Networks-09 (NCOWN-2010), 2010.

[5] Y. Sharma, S. C. Saini, and M. Bhandhari, “Comparison of Dijkstra’s shortest path algorithm with genetic algorithm for static and dynamic routing network,” *International Journal of Electronics and Computer Science Engineering*, vol. 1, no. 2, pp. 416–425, 2012.

[6] A. Hussein, M. Adel, M. Bakr, O. M. Shehata, and A. Khamis, “Multi-robot task allocation for search and rescue missions,” *Journal of Physics: Conference Series*, vol. 570, pp. 1–10, 2014.

[7] S. A. Fadzli, S. I. Abdulkadir, M. Makhtar, and A. A. Jamal, “Robotic indoor path planning using Dijkstra’s algorithm with multi-layer dictionaries,” in *Proceedings 2015 2nd*

วารสารวิชาการพระจอมเกล้าพระนครเหนือ ปีที่ 29 ฉบับที่ 3 ก.ค.–ก.ย. 2562
The Journal of KMUTNB., Vol. 29, No. 3, Jul.–Sep. 2019

International Conference on Information Science and Security (ICISS), 2015, pp. 1–4.

[8] M. Quigley, K. Conley, B. Gerkey, and J. Faust, "ROS: An open-source robot operating system," presented at the ICRA workshop on open source software, 2009.

[9] M. Ben-Ari and F. Mondada, "Mapping-based navigation," in *Proceedings Elements of Robotics*, 2018, pp. 165–178.

[10] M. Karlsson, F. Ygge, and A. Andersson, "Market-based approaches to optimization," *Computational Intelligence*, vol. 23, no. 1, pp. 92–109, 2007.

[11] M. B. Dias, R. Zlot, N. Kalra, and A. Stentz, "Market-based multirobot coordination: A survey and analysis," in *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1257–1270, 2006.

[12] M. B. Dias and A. Stentz, "Opportunistic optimization for market-based multirobot control," presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 2002.

[13] R. Zlot and A. Stentz, "Market-based multirobot coordination for complex tasks," *International Journal of Robotics Research*, vol. 25, no. 1, pp. 73–101, 2006.

[14] C. Myint and N. N. Win, "Position and velocity control for two-wheel differential drive mobile robot," *International Journal of Science Engineering and Technology Research (IJSETR)*, vol. 5, no. 9, pp. 2849–2855, 2016.

[15] H. Taheri, B. Qiao, and N. Ghaeminezhad, "Kinematic model of a four mecanum wheeled mobile robot," *International Journal of Computer Applications*, vol. 113, no. 3, pp. 6–9, 2015.

[16] J. Rubaszewski, A. Yalaoui, L. Amodeo, and S. Fuchs, "Extensions of the unidirectional flow path design problem solved by efficient metaheuristics," *IFAC Proceedings Volumes*, vol. 46, no. 9, pp. 784–789, 2013.

[17] S. Manca, A. Fagiolini, and L. Pallottino, "Decentralized coordination system for multiple AGVs in a structured environment," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 6005–6010, 2011.

ประวัติผู้เขียน

ชื่อ สกุล นายพงศกร ชาญชัยชูจิต

รหัสประจำตัวนักศึกษา 5810120036

วุฒิการศึกษา

วุฒิ	ชื่อสถาบัน	ปีที่สำเร็จการศึกษา
วิศวกรรมศาสตร์บัณฑิต (วิศวกรรมเมคาทรอนิกส์)	มหาวิทยาลัยสงขลานครินทร์	2011

ทุนการศึกษา

ทุนอุดหนุนการวิจัยเพื่อวิทยานิพนธ์ ปี 2559

ตำแหน่งและสถานที่ทำงาน

วิศวกรผู้ช่วยโครงการโรงประลองต้นแบบทางวิศวกรรมโรงเรียนหาดใหญ่วิทยาลัยสมบูรณ์กุลกันยา

การตีพิมพ์เผยแพร่ผลงาน

พงศกร ชาญชัยชูจิต และ พฤทธิกร สมิตไมตรี, “ระบบการจัดการหุ่นยนต์หลายตัวสำหรับการขนถ่ายวัสดุในโรงงาน อุตสาหกรรม,” วารสารวิชาการพระจอมเกล้าพระนครเหนือ, ปีที่ 29, ฉบับที่ 3, ก.ค.–ก.ย. 2562.