สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ขั้นตอนวิธีอาณานิคมมดแบบดัดแปลงสำหรับปัญหาการจัดตารางการผลิตแบบตามสั่ง

A MODIFIED ANT COLONY ALGORITHM FOR JOB SHOP SCHEDULING PROBLEM

T123038

สรรเพ็ชญ์ ไชยศิลปสังข์
SANPET CHAISINLAPASANG

วิทย
ส33ช
2555
เลขหมู่ 123038
เลขทะเบียน
วัน,เดือน,ปี 18 ต.ค. 2559
b.12434519
i.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.2555
KMITL-2012-EN-M-217-127

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A MODIFIED ANT COLONY ALGORITHM FOR JOB SHOP SCHEDULING PROBLEM

SANPET CHAISINLAPASANG

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN INDUSTRIAL ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2012

KMITL-2012-EN-M-217-127

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2012
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ ขั้นตอนวิธีอาณานิคมมดแบบดัดแปลงสำหรับปัญหาการจัดตารางการผลิตแบบตามสั่ง
Thesis Title A Modified Ant Colony Algorithm for Job Shop Scheduling Problem
นักศึกษา นายสรรเพ็ชญ์ ไชยศิลปสังข์
รหัสประจำตัว 53612410
ปริญญา วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา วิศวกรรมอุตสาหการ
อาจารย์ที่ปรึกษาวิทยานิพนธ์ ดร.อุดม จันทร์จรัสสุข
หมายเลขวิทยานิพนธ์ KMITL-2012-EN-M-217-127

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
รศ.ดร.ฤดี	มาสุจันท์	
ผศ.ดร.สิทธิพร	พิมพ์สกุล	
รศ.ดร.เดชา	พวงดาวเรือง	
ดร.ชุมพล	ยวงใย	
ดร.อุดม	จันทร์จรัสสุข	

วัน / เดือน/ ปี ที่สอบ วันพฤหัสบดีที่ 31 พฤษภาคม พ.ศ. 2555 เวลา 09.00-11.00 น.
สถานที่สอบ ณ อาคาร A ชั้น 5 ห้องประชุม 3

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

คณะวิศวกรรมศาสตร์ รับรองแล้ว

(รองศาสตราจารย์ ดร.สุชัชวีร์ สุวรรณสวัสดิ์)
คณบดี คณะวิศวกรรมศาสตร์
วันที่ 31 พฤษภาคม พ.ศ. 2555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	**ขั้นตอนวิธีอาณานิคมมดแบบดัดแปลงสำหรับปัญหาการจัดตารางการผลิตแบบตามสั่ง**
นักศึกษา	**นายสรรเพ็ชญ์ ไชยศิลปสังข์**
รหัสประจำตัว	53612410
ปริญญา	**วิศวกรรมศาสตรมหาบัณฑิต**
สาขาวิชา	**วิศวกรรมอุตสาหการ**
พ.ศ.	2555
อาจารย์ที่ปรึกษาวิทยานิพนธ์	**ดร.อุดม จันทร์จรัสสุข**

บทคัดย่อ

งานวิจัยนี้นำเสนอขั้นตอนวิธีอาณานิคมมด (Ant Colony Optimization, ACO) สำหรับการค้นหาคำตอบที่ดีที่สุดของปัญหาการจัดตารางการผลิตแบบตามสั่ง โดยผู้วิจัยมุ่งประเด็นไปในเรื่อง การเลือกใช้ค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่เหมาะสมสำหรับแต่ละปัญหา จากการทดสอบประสิทธิภาพของโปรแกรมคอมพิวเตอร์ที่พัฒนาขึ้นด้วยภาษา Visual C# กับชุดปัญหามาตรฐานจาก OR-Library จำนวน 20 ปัญหา พบว่า คำตอบที่ได้มีประสิทธิภาพสูง โดยมีเปอร์เซ็นต์ความผิดพลาดอยู่ในช่วงที่ยอมรับได้ (MRE%<10%) เมื่อเทียบกับคำตอบที่ดีที่สุด (Optimal Solution) และได้ผลคำตอบที่ดีที่สุดจำนวน 4 ชุดปัญหาคิดเป็นร้อยละ 26 ของปัญหาทั้งหมด จากการวิเคราะห์ความสัมพันธ์ของค่า α และ β ที่ได้จากการทดลองซึ่งนำไปสู่คำตอบที่ดีที่สุด ทำให้ผู้วิจัยสามารถหาแนวโน้มของผลคำตอบที่ดีที่สุด ที่เกิดจากการเปลี่ยนแปลงค่า α และ β ได้ และนำไปพัฒนาเป็นขั้นตอนวิธีในการเปลี่ยนแปลงค่า α และ β อย่างอัตโนมัติ ผลการทดลองคำตอบที่ได้อยู่ในช่วงที่ยอมรับได้ (MRE%<10%) คิดเป็นร้อยละ 60 ของปัญหาทั้งหมดที่ทำการทดลอง และให้ผลคำตอบเท่ากับคำตอบที่ดีที่สุดคิดเป็นร้อยละ 15 ของปัญหาทั้งหมดที่ทำการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis	A Modified Ant Colony Algorithm for Job Shop Scheduling Problem
Student	Sanpet Chaisinlapasang
Student ID.	53612410
Degree	Master of Engineering
Program	Industrial Engineering
Year	2012
Thesis Advisor	Dr. Udom Janjarasuk

ABSTRACT

This thesis presents an ant colony optimization (ACO) algorithm for solving the job shop scheduling problem (JSSP). The purpose of this study is to seek for the appropriate values of pheromone influence (α) and heuristic influence (β) parameters used in the ACO algorithm. A computer program is implemented in Visual C# to test the effectiveness of the algorithm using standard benchmark problems from the OR-Library (20 benchmarks). The results showed that the mean relative error compared to the optimal solution is within the acceptable range (MRE% < 10%). Optimal solutions were found in four problems, which constitute 26% of all tested problems. From the experiment, the appropriate values of α and β were obtained for each problem, and the relationship of α and β were analyzed by considering the responses to the changes of α and β which leads to the best result. An algorithm for automatically adjust the values of α and β is further proposed and tested. Computation results showed that 60 percent of the answer from all tested of the answer has the optimal solution.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ดร. อุดม จันทร์จรัสสุข ที่ให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้และประการณ์ที่ดีแก่ข้าพเจ้า

ขอขอบพระคุณกรรมการสอบหัวข้อและโครงร่างวิทยานิพนธ์ที่ได้กรุณาให้คำแนะนำตลอดจนข้อชี้แนะ จนในที่สุดทำให้วิทยานิพนธ์ฉบับนี้สำเร็จลงได้

ขอขอบคุณเพื่อนๆและพี่ๆทุกคน ที่ให้ความช่วยเหลือและเป็นกำลังในยามที่ท้อใจ

ขอขอบคุณ สถานที่ท่องเที่ยวพักผ่อนหย่อนใจทุกที่โดยเฉพาะโต๊ะปิงปองในภาควิชาอุตสาหการ ที่ผมใช้เป็นที่ผ่อนคลายและใช้คิดแก้ไขปัญหาจนลุล่วงงานวิจัยนี้

วิทยานิพนธ์เล่มนี้ได้รับทุนสนับสนุนการทำวิทยานิพนธ์จากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สุดท้ายต้องขอขอบคุณเพื่อนๆและน้องๆทุกคนที่เป็นกำเป็นลังใจที่ดีตลอดมา

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า

สรรเพ็ชญ์ ไชยศิลปสังข์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญตาราง VI
สารบัญรูป VIII
บทที่ 1 บทนำ 1
1.1 ความเป็นมาและความสำคัญของปัญหา 1
1.2 วัตถุประสงค์ของการศึกษา 2
1.3 ขอบเขตการวิจัย 3
1.4 ขั้นตอนของการศึกษา 3
1.5 ประโยชน์ที่คาดว่าจะได้รับ 3
บทที่ 2 งานวิจัยที่เกี่ยวข้อง 4
2.1 การจัดตารางการผลิต 4
2.2 แบบจำลองการจัดตารางการผลิต (Scheduling Models) 5
2.2.1 แบบจำลองการผลิตแบบไหล (Flow Shop) 5
2.2.2 แบบจำลองการผลิตแบบตามสั่ง (Job Shop) 5
2.3 การผลิตแบบตามสั่ง (Job shop) 5
2.3.1 ข้อดีของการผลิตแบบตามสั่ง 6
2.3.2 ข้อเสียของการผลิตแบบตามสั่ง 6
2.4 ตัวอย่างและวิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง 7
2.4.1 ตัวอย่างการผลิตแบบตามสั่ง 7
2.4.2 วิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง 7
2.5 มาตรฐานที่ใช้วัดสมรรถนะการจัดตารางการผลิต 10
2.6 ข้อจำกัดที่เกี่ยวข้องกับการจัดตารางการผลิต 10
2.7 การแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง 11
2.7.1 วิธีดีเทอร์มินิสติค (Deterministic Scheduling) 11
2.7.2 วิธีฮิวริสติค (Heuristic Scheduling) 11
2.8 วิธีอาณานิคมมด (Ant Colony Optimization, ACO) 13
2.9 ขั้นตอนของวิธีอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง 19
2.9.1 การส่งมดออกไปหาเส้นทาง 20
2.9.2 การปรับปรุงคุณภาพคำตอบเฉพาะที่ (Local improvement) 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.9.3 การปรับปรุงค่าความเข้มข้นของฟีโรโมน (Pheromone update) 22

2.9.4 การกำหนดค่าพารามิเตอร์ 23

บทที่ 3 วิธีการดำเนินการวิจัย 27

3.1 การศึกษาและวิเคราะห์ชุดปัญหาของการผลิตแบบตามสั่ง 27

3.1.1 วิเคราะห์ลักษณะชุดปัญหามาตรฐาน Lawrence จาก OR-Library 28

3.1.2 การกำหนดปัญหาโดยใช้แบบจำลองทางคณิตศาสตร์ 29

3.2 การปรับปรุงและพัฒนาวิธีอาณานิคมมดในการแก้ปัญหาจัดตารางการผลิตแบบตามสั่ง 30

3.2.1 การจัดลำดับการผลิต 32

3.2.2 การกำหนดค่าข้อมูลทางฮิวริสติค (η) 35

3.2.3 การปรับค่าฟีโรโมน (Pheromone update) 36

3.2.4 การกำหนดค่าและการเปลี่ยนค่าน้ำหนักของฟีโรโมนและค่าน้ำหนักของฮิวริสติค 36

3.3.5 การปรับปรุงคุณภาพคำตอบเฉพาะที่ (Local improvement) 38

3.3 การพัฒนาโปรแกรมคอมพิวเตอร์ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง 39

3.3.1 การออกแบบโปรแกรมคอมพิวเตอร์ 39

บทที่ 4 ผลการดำเนินงาน 41

4.1 การตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์ 41

4.2 การทดลองและเปรียบเทียบผลคำตอบ 43

4.2.1 การทดลองและเปรียบเทียบผลคำตอบกับชุดปัญหามาตรฐาน 43

4.2.2 การทดลองและเปรียบเทียบผลคำตอบในปัญหาการผลิตแบบตามสั่งในปัจจุบัน 51

บทที่ 5 สรุปผลกการดำเนินงาน 56

5.1 สรุปและอภิปรายผลการดำเนินงาน 56

5.2 ข้อเสนอแนะ 57

บรรณานุกรม 58

ภาคผนวก 61

ภาคผนวก ก ปัญหาที่ใช้ในการทดลอง 62

ภาคผนวก ข โปรแกรมคอมพิวเตอร์ 71

ภาคผนวก ค บทความวิชาการที่ได้รับการตีพิมพ์ 96

ประวัติผู้เขียน 104

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

2.1 แสดงตัวอย่างปัญหาการผลิตแบบตามสั่ง 7

2.2 ตัวอย่างการจัดลำดับของปัญหาที่มี 3 งานและ 3 เครื่องจักร 22

2.3 แสดงค่าความสัมพันธ์ของค่า α และ β ที่จะให้คำตอบที่ดีที่สุดของปัญหาการเดินทางของพนักงานขาย 25

3.1 แสดงข้อมูลที่ได้จากตัวอย่างปัญหา La01 27

4.1 รายละเอียดของปัญหามาตรฐาน Lawrence ใช้ในการทดลอง 44

4.2. ค่า α และ β ที่ใช้ในการทดลองในการแก้ปัญหาแต่ละปัญหา 45

4.3 แสดงค่าเวลาที่ใช้ในการดำเนินการ (Makespan) ของการวิจัยนี้เปรียบเทียบกับวิธีอื่นและค่า α และ β ที่เหมาะสมแต่ละปัญหา 46

4.4 แสดงค่าเฉลี่ย α , β และเวลาในการหาคำตอบของการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ 50

4.5 แสดงค่าเฉลี่ยของ α , β และเวลาในการหาคำตอบของการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ ของปัญหา la04 ในจำนวนรอบ (Iteration)ที่แตกต่างกัน 51

4.6 ตัวอย่างปัญหาระบบการผลิตแบบตามสั่งของบริษัท Shanghai Volkswagen Automobile Co.Ltd. (Xie and Huo. 2010 : 942) 52

4.7 ผลการทดลองแก้ปัญหา บริษัท Shanghai Volkswagen Automobile Co.Ltd. 53

ก.1 รายละเอียดของปัญหามาตรฐาน Lawrence ใช้ในการทดลอง 62

ก.2 ปัญหา la01 63

ก.3 ปัญหา la02 63

ก.4 ปัญหา la03 63

ก.5 ปัญหา la04 63

ก.6 ปัญหา la05 64

ก.7 ปัญหา la06 64

ก.8 ปัญหา la07 64

ก.9 ปัญหา la08 64

ก.10 ปัญหา la09 65

ก.11 ปัญหา la10 65

ก.12 ปัญหา la11 65

ก.13 ปัญหา la12 66

ก.14 ปัญหา la13 66

ก.15 ปัญหา la14 67

ก.16 ปัญหา la15 67

ก.17 ปัญหา la16 68

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

ตารางที่ หน้า
ก.18 ปัญหา la17 68
ก.19 ปัญหา la18 69
ก.20 ปัญหา la19 69
ก.21 ปัญหา la20 70
ข.1 การกำหนดค่าตัวแปรสำหรับการรับค่าต่างๆ 71
ข.2 การกำหนดค่าตัวแปรสำหรับการหาเส้นทาง 71
ข.3 คำอธิบายความหมายของป้ายชื่อ (Label) ของส่วนการรับค่าค่าคงที่ที่ใช้ในการหาคำตอบ จากรูปที่ ข.2 74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

1.1 จำนวนงานวิจัยที่เกี่ยวกับการแก้ปัญหาการผลิตแบบตามสั่ง ตั้งแต่ปี 2000-2009 2

2.1 ความสัมพันธ์ของการจัดตารางการผลิต 4

2.2 แสดงลักษณะการผลิตแบบตามสั่งและแบบตามการไหล 6

2.3 ขั้นตอนการจัดตารางการผลิตแบบตามสั่ง 8

2.4 (ก) Disjunctive graph, (ข) การแสดงลำดับการผลิตลงบน disjunctive graph, (ค) Gantt diagram ของลำดับการผลิต 9

2.5 การเดินทางออกจากรังไปหาอาหารของมดชุดแรก 14

2.6 หลังจากมดชุดแรกออกเดินทางไปถึงแหล่งอาหาร 14

2.7 มดชุดที่สองออกเดินทางไปยังแหล่งอาหาร 15

2.8 มดชุดที่สองเลือกเส้นทางในการเดินทางกลับมารัง 15

2.9 มดชุดที่สองเดินทางกลับมาถึงรัง 16

2.10 การเลือกเส้นทางของมดตัวอื่นๆ 16

2.11 แสดงการหาเส้นทางของมดในธรรมชาติ 17

2.12 ขั้นตอนของวิธีอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง 20

2.13 Disjunctive graph ของปัญหาที่มี 3 งานและ 3 เครื่องจักร 21

3.1 Disjunctive graph ของปัญหาที่มี 10 งานและ 5 เครื่องจักร 29

3.2 ผังการไหลการทำงานของโปรแกรมคอมพิวเตอร์ 31

3.3 ทิศทางที่มดจะเลือกการดำเนินการของการส่งมดออกไปหาเส้นทางแบบไปข้างหน้า 33

3.4 ทิศทางที่มดจะเลือกการดำเนินการของการส่งมดออกไปหาเส้นทางแบบกลับหลัง 34

3.5 แสดงค่า α และ β ทั้ง 5 คู่แรกที่มดนำไปจัดตารางการผลิตสำหรับการดำเนินการรอบแรก 37

3.6 (a) แสดงการปรับค่าแบบก้าวกระโดดและการลดค่า $\Delta\alpha$ และ $\Delta\beta$ เมื่อได้ค่า α และ β ใหม่เท่ากับค่าเดิม, (b) แสดงกระบวนการในการปรับค่า α และ β หลังการลดค่า $\Delta\alpha$ และ $\Delta\beta$ 38

3.7 แสดงวิธีการปรับปรุงคุณภาพคำตอบของ E.Nowicki 38

3.8 ขั้นตอนการพัฒนาโปรแกรมคอมพิวเตอร์ 39

4.1 แสดงเมตริกซ์ปัญหาที่ป้อนให้กับโปรแกรมคอมพิวเตอร์ 41

4.2 แสดงไฟล์บันทึกคำตอบที่ดีที่สุดที่โปรแกรมคอมพิวเตอร์หาได้และค่าเวลาเริ่มต้นและเวลาสิ้นสุดของแต่ละการดำเนินการ 42

4.3 แสดงแผนภูมิแกนต์ของการหาคำตอบโดยโปรแกรมคอมพิวเตอร์ 42

4.5 แสดงความสัมพันธ์ระหว่างค่า α และ β เปรียบเทียบกับค่าเวลาในการดำเนินการทั้งหมดในชุดปัญหา La09 47

4.6 แสดงค่าเริ่มต้นของ α และ β ทั้ง 5 คู่แรกที่มดนำไปจัดตารางการผลิตสำหรับการดำเนินการรอบแรก 48

4.7 แผนภูมิแกนต์แสดงตารางการผลิตของบริษัท Shanghai Volkswagen Automobile Co.Ltd. 54
(ก) ตารางการผลิตจากการแก้ปัญหาด้วยงานวิจัยนี้ (ข) ตารางการผลิตจากการแก้ปัญหาด้วยงานวิจัย Xie และ Huo. 2010 54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

ข.1 แผนผังส่วนประกอบของโปรแกรมคอมพิวเตอร์ 72

ข.2 หน้าต่างส่วนของการรับข้อมูล 73

ข.3 หน้าต่างของโปรแกรมคอมพิวเตอร์ระหว่างการจัดตารางการผลิต 74

ข.4 คำอธิบายความหมายของค่าต่างๆที่มีการแสดงผล 75

ข.5 อธิบายไฟล์การบันทึกข้อมูลของการจัดตารางการผลิต 77

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันในทุกอุตสาหกรรมมีการแข่งขันสูงและการจัดตารางการผลิตเป็นส่วนสำคัญ การจัดตารางการผลิตที่มีประสิทธิภาพจะทำให้ลดค่าใช้จ่าย ลดเวลาในการผลิตและเกิดความได้เปรียบในการแข่งขันทางธุรกิจ อีกทั้งการจัดตารางการผลิตที่ดีนั้นเป็นประเด็นหลักในการประเมินประสิทธิภาพขององค์กร บ่อยครั้งปัญหาของระบบการผลิตจะเกี่ยวข้องกับรูปแบบและประสิทธิภาพของการจัดตารางการผลิต ตัวอย่างเช่น ถ้าสินค้าผลิตเสร็จก่อนเวลาส่งมอบ ทำให้เสียค่าใช้จ่ายเพิ่มในการเก็บสินค้านั้นและยังส่งผลถึงค่าใช้จ่ายในการจ้างคนงานแบบล่วงเวลาของกระบวนผลิตถัดไปที่นำส่วนประกอบนั้นไปใช้ โดยเฉพาะกับส่วนประกอบที่มีอายุจำกัด ในทางกลับกันถ้าการผลิตสินค้าเสร็จล่าช้า กระบวนการผลิตถัดไปต้องหยุดหรือต้องเสียค่าใช้จ่ายเพิ่มในการสั่งซื้อส่วนประกอบจากที่อื่นมาทดแทน ซึ่งอาจจะส่งผลต่อความน่าเชื่อถือและความพึงพอใจของลูกค้าต่อสินค้าและบริการของธุรกิจนั้น (เศรษฐา และ ธราธร. ม.ป.ป.)

ในระบบการผลิตในอุตสาหกรรมทั่วไปมี 2 ลักษณะคือ การผลิตแบบตามการไหล (Flow Shop) และการผลิตแบบตามสั่ง (Job shop) การผลิตทั้งสองชนิดนี้แตกต่างกันตามลักษณะและทิศทางการผลิต ซึ่งวิธีการผลิตแบบตามสั่งมีแนวโน้มขยายตัวมากขึ้น เนื่องจากการผลิตแบบตามสั่งจะก่อให้เกิดความหลากหลายของผลิตภัณฑ์มากกว่าการผลิตแบบตามการไหล แต่มีความยุ่งยากและซับซ้อนมากกว่า เนื่องจากมีทิศทางการไหลของงานไม่แน่นอน ประกอบด้วยกระบวนการหลายๆกระบวนการ

จากที่กล่าวมาข้างต้นจะเห็นได้ว่าปัญหาในการจัดตารางการผลิตมีความสำคัญเป็นอย่างมาก สามารถแก้ปัญหาโดยการประยุกต์ใช้หลักของการวิจัยการดำเนินงาน (Operations Research) ซึ่งปัญหานั้นมีความซับซ้อนในการหาคำตอบที่ดีที่สุด (Optimal solution) ถ้าแก้ปัญหาด้วยวิธีดีเทอร์มินิสติกส์ (Deterministic) จะใช้เวลานานในการหาคำตอบ ถึงแม้คำตอบที่ได้จะเป็นคำตอบที่ดีที่สุดจริงๆ แต่ในสภาวะในปัจจุบัน งานมีจำนวนมากและต้องการคำตอบภายในเวลาที่ยอมรับได้เพื่อเพิ่มความคล่องตัวในการผลิต จึงจำเป็นต้องใช้วิธีการหาคำตอบที่ใช้เวลาน้อยกว่า นั่นก็คือวิธีฮิวริสติค (Heuristic) แม้วิธีฮิวริสติคอาจจะไม่ได้คำตอบที่ดีที่สุดแต่คำตอบที่ได้ถือว่ายอมรับได้หรือใกล้เคียงคำตอบที่ดีที่สุด

วิธีฮิวริสติคมีอยู่หลายวิธี โดยวิธีที่เลือกมาใช้ในการวิจัยนี้คือขั้นตอนวิธีอาณานิคมมด (Ant Colony Algorithm) ซึ่งจัดอยู่ในกลุ่มวิธีการทางปัญญาประดิษฐ์ (Artificial Intelligent) โดยจะใช้หลักความน่าจะเป็นมาช่วยในการหาคำตอบ โดยวิธีนี้ได้มีการนำมาประยุกต์ในการแก้ปัญหาต่างๆมากมาย รวมไปถึงการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง ซึ่งคำตอบที่ได้ก็มีประสิทธิภาพสูง แต่วิธีนี้มีข้อด้อยในเรื่องของเวลาที่ใช้ในการหาคำตอบค่อนข้างมาก และถ้าเปลี่ยนรูปแบบของปัญหาก็ต้องมีการเปลี่ยนแปลงค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ซึ่งค่าคงที่ทั้งสองนี้ต้องใช้วิธีลองผิดลองถูก นั่นคือต้องใช้เวลามากในการหาค่าที่เหมาะสม ส่งผลให้เวลาในการหาคำตอบมากขึ้นเช่นกัน โดยในรอบ 10 ปีที่ผ่านมามีการวิจัยและการพัฒนาวิธีการหาคำตอบแบบอาณานิคมมดเพียง 10% เมื่อเทียบกับวิธีอื่น) ดังแสดงในรูปที่ 1.1 เช่นวิธีการค้นหาแบบทาบู (Tabu

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Search) และขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm) จึงทำให้วิธีการนี้ไม่ได้รับการพัฒนาเท่าที่ควร (Fan. and Zhang. 2010 : 285)

Algorithms	00	01	02	03	04	05	06	07	08	09	Total
1. Accurate algorithm											
1) Analytical method											
-Johnson algorithm	1	-	2	1	1	3	2	1	2	1	14
-Moore algorithm	-	-	-	-	1	-	-	-	-	-	1
2) Enumerative method											
-Bound and branch	6	6	4	2	5	11	9	11	13	18	85
-Mathematical method											
-*Mixed integer programming*	5	-	2	2	1	6	6	1	7	9	39
-*Integer programming*	9	2	6	3	2	9	7	3	8	13	62
-*Lagrangian relaxation*	2	1	1	2	2	3	2	1	1	2	17
-*Decomposition approach*	1	2	3	1	1	-	3	4	2	5	22
2. Approximate algorithm											
3) Constructive method											
-Dispatching rules	15	5	10	9	11	10	14	11	19	12	116
-Bottleneck based heuristics											
-*Shifting bottleneck procedure*	2	2	1	2	8	1	8	5	6	2	37
-*Beam search*	1	-	2	-	1	-	-	2	3	1	10
-Insertion algorithm	-	-	2	4	2	2	1	3	3	4	21
4) Local search method											
-Tabu Search	3	3	6	10	10	18	12	22	30	26	140
-Simulated Annealing	3	6	3	6	9	9	6	6	13	17	78
-Threshold accepting	-	-	3	-	1	-	-	-	-	-	4
-Variable Neighborhood Search	-	1	-	-	-	-	2	1	3	2	9
-Evolutionary Computation											
-*Genetic Algorithm*	16	14	11	21	17	28	31	22	52	62	274
-*Evolutionary Strategy*	-	-	-	1	-	1	-	1	-	2	5
-*Evolutionary programming*	1	1	-	-	-	-	2	1	1	1	7
-*Genetic Programming*	2	3	3	-	1	2	4	2	6	8	31
-*Particle Swarm Optimization*	-	-	-	-	-	1	5	2	16	11	35
5) Artificial intelligence											
-Neural Network	4	3	3	3	2	9	3	-	4	3	34
-Ant Colony Optimization	-	-	-	-	1	2	4	5	8	13	33
-Immune Algorithm	-	-	-	1	1	3	4	2	1	11	23
-Expert System	1	1	1	-	1	-	3	-	1	-	8

รูปที่ 1.1 จำนวนงานวิจัยที่เกี่ยวกับการแก้ปัญหาการผลิตแบบตามสั่ง ตั้งแต่ปี 2000-2009
(Fan. and Zhang. 2010 : 285)

ดังนั้นในงานวิจัยนี้จึงเน้นในการพัฒนาสมรรถนะของการหาคำตอบจากวิธีอาณานิคมมด โดยการหาค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่เหมาะสมกับปัญหา และพัฒนาการจัดตารางการผลิตด้วยวิธีอาณานิคมมดให้มีการปรับเปลี่ยนค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) อย่างอัตโนมัติ

1.2 วัตถุประสงค์ของการศึกษา

การศึกษาและการทำวิจัยนี้มุ่งเน้นไปถึง

1.2.1 เพื่อประยุกต์ใช้ขั้นตอนวิธีอาณานิคมมด (Ant Colony Algorithm) ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง (Job-shop Scheduling)

1.2.1 เพื่อปรับปรุงและพัฒนาขั้นตอนวิธีอาณานิคมมด ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2.2 เพื่อศึกษาและเปรียบเทียบสมรรถนะของการจัดตารางการผลิตแบบตามสั่งระหว่างวิธีที่พัฒนาขึ้นมาโดยอาศัยขั้นตอนวิธีอาณานิคมมดกับวิธีอื่นๆ ในการแก้ปัญหามาตรฐานของการจัดตารางการผลิตแบบตามสั่ง

1.2.1 เพื่อเปรียบเทียบสมรรถนะของการจัดตารางการผลิตแบบตามสั่ง ระหว่างวิธีที่พัฒนาขึ้นมาโดยอาศัยขั้นตอนวิธีอาณานิคมมดกับวิธีอื่น ในการแก้ปัญหาระบบการผลิตจริงในปัจจุบัน

1.3 ขอบเขตการวิจัย

1.3.1 ทำการทดลองเปรียบเทียบกับวิธีการอื่นๆบนปัญหามาตรฐานที่มีคำตอบที่ดีที่สุด (Optimal Solution) จากวิธีดีเทอร์มินิสติค ในด้านของประสิทธิภาพในการหาคำตอบ

1.3.2 ทำการทดลองเปรียบเทียบเวลาในการหาคำตอบ ระหว่างวิธีที่มีและไม่มีการเปลี่ยนแปลงค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β)

1.3.3 ทดสอบคำตอบกับชุดปัญหามาตรฐานของ Lawrence (Lawrence . 1984) และทดสอบกับชุดปัญหาการผลิตแบบตามสั่งของบริษัทผลิตรถยนต์ในประเทศจีน (Xie and Huo. 2010 : 942) ในงานวิจัยนี้ไม่พิจารณาในประเด็นระบบการผลิตที่มีการระบุลำดับของงาน

1.3.4 งานวิจัยนี้ไม่พิจารณาระบบการผลิตที่มีการใช้เครื่องมือ (Tool) ร่วมกันของแต่ละการดำเนินการ

1.4 ขั้นตอนของการศึกษา

1.4.1 ศึกษารูปแบบปัญหาการจัดตารางการผลิตแบบตามสั่ง

1.4.2 ศึกษาวิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งด้วยขั้นตอนวิธีอาณานิคมมด

1.4.3 ปรับปรุงวิธีการหาคำตอบการจัดตารางการผลิตแบบตามสั่งด้วยขั้นตอนวิธีอาณานิคม

1.4.4 ออกแบบและพัฒนาวิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งด้วยขั้นตอนวิธีอาณานิคมมดโดยใช้โปรแกรมคอมพิวเตอร์

1.4.5 ทำการทดลองเปรียบเทียบสมรรถนะในการหาคำตอบของโปรแกรมคอมพิวเตอร์ที่จัดทำขึ้นกับวิธีอื่นๆ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 สามารถนำงานวิจัยที่จัดทำขึ้นเป็นตัวแบบ (Model) ในการปรับปรุงและพัฒนาวิธีการจัดตารางการผลิตแบบตามสั่งด้วยการประยุกต์ใช้ขั้นตอนวิธีอาณานิคมมด ในเรื่องของการปรับปรุงค่าคงที่ในการหาคำตอบได้

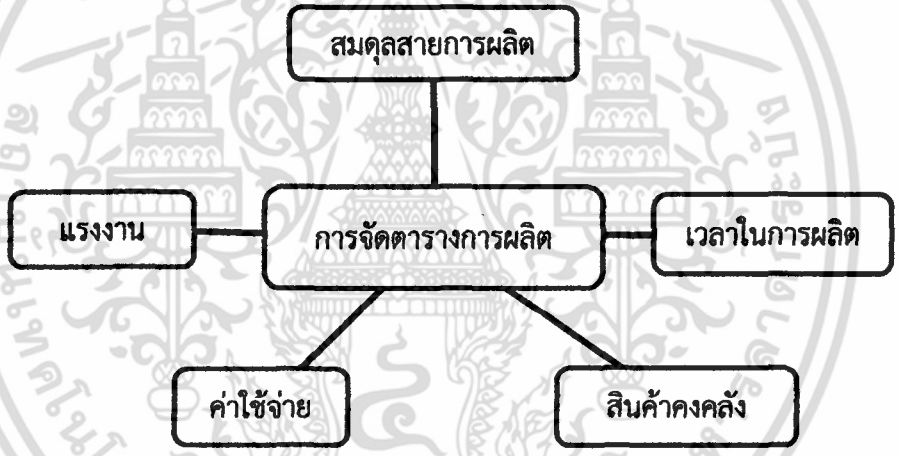
1.5.2 สามารถนำโปรแกรมคอมพิวเตอร์ที่จัดทำขึ้นไปประยุกต์หรือพัฒนาให้สอดคล้องกับสภาพความเป็นจริงมากขึ้น ซึ่งทำให้การจัดตารางการผลิตมีประสิทธิภาพมากขึ้น สามารถลดเวลาในการทำงาน และลดค่าใช้จ่ายระหว่างการผลิตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
งานวิจัยที่เกี่ยวข้อง

2.1 การจัดตารางการผลิต

การจัดลำดับการผลิต (Scheduling) เป็นส่วนหนึ่งของการวางแผน (Planning) ในการทำงานและการผลิตในอุตสาหกรรม เมื่อพิจารณาในการทำแผนการผลิตจะเกี่ยวข้องกับการจัดตัวแบบ model ของการดำเนินงานในกระบวนการผลิตทั้งหมดให้สอดคล้องกับเป้าหมาย (Objective) และข้อจำกัด (Constraint) ที่เกิดขึ้นของโครงการหรือสายการผลิต (Production line) ส่วนกรณีของการทำตารางการผลิตนั้น จะเกี่ยวข้องกับการกำหนดทรัพยากร (Resource) ซึ่งในที่นี้ก็คือ แรงงานคน เครื่องจักร อุปกรณ์ เครื่องมือ วัตถุดิบ วิธีการดำเนินงานและปัจจัยอื่นๆ ที่เกี่ยวข้องกับกระบวนการผลิต ให้มีความสอดคล้องและเหมาะสมกับกิจกรรม (Activity) ของแต่ละงานในสายการผลิต (เอกชาติ หัตถา. 2543 : 28)



รูปที่ 2.1 ความสัมพันธ์ของการจัดตารางการผลิต

จากรูปที่ 2.1 จะเห็นได้ว่าการจัดตารางการผลิตมีความสำคัญกับทุกส่วนของระบบการผลิต โดยการจัดตารางการผลิตเป็นการจัดสรรทรัพยากรที่มีอยู่อย่างจำกัดให้กับงาน ภายใต้เงื่อนไขของเวลา โดยที่ พิภพ ลลิตาภรณ์ (2545:392) ได้กล่าวถึงจุดประสงค์ของการจัดตารางการผลิต 3 วัตถุประสงค์ดังนี้

1. เพิ่มประโยชน์การใช้หน่วยงานคือ การลดช่องว่างงานของหน่วยงาน
2. ลดจำนวนงานโดยเฉลี่ยที่คอยอยู่ในแถวคอย
3. ลดจำนวนงานที่เสร็จช้ากว่ากำหนดหรือพยายามทำใบสั่งงานทุกงานให้เสร็จในระยะเวลาที่กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 แบบจำลองการจัดตารางการผลิต (Scheduling Models)

สามารถจำแนกแบบจำลองการจัดตารางการผลิตตามลักษณะการจัดเรียงของเครื่องจักรและการไหลของชิ้นงานในระบบออกเป็นประเภทต่างๆดังนี้

2.2.1 แบบจำลองการผลิตแบบไหล (Flow Shop)

ระบบประกอบไปด้วยเครื่องจักรจำนวน m เครื่องที่แตกต่างกันวางต่อกันแบบอนุกรม งานทั้งหมดจะมีเส้นทางการไหลของงานไปในทิศทางเดียวกัน (Unidirectional Flow) ลักษณะการผลิตแบบการไหลตามสายงาน (Flow Shop) จะมีสายการผลิตที่ค่อนข้างลงตัว การไหลของงานจะเป็นไปในทิศทางเดียวตั้งแต่ต้นจนกระทั่งเสร็จสิ้นงาน และมีการจัดตารางเครื่องจักรเรียงกันตามกระบวนการต่อเนื่องกัน โดยส่วนมากจะเป็นการผลิตเก็บไว้เพื่อรอคำสั่งซื้อจากลูกค้า (Make to stock) (ปัญจพร แพใหญ่. 2549 : 9-10)

2.2.2 แบบจำลองการผลิตแบบตามสั่ง (Job Shop)

ระบบประกอบด้วยเครื่องจักร m เครื่อง ในแต่ละหน่วยงานจะมีเส้นทางการไหลของงานเฉพาะของตนเอง ทิศทางการไหลของงานมีได้หลายทิศทาง (Non-Unidirection Flow) แต่ละงานสามารถที่จะดำเนินงานบนเครื่องจักรใดๆ ก็ตามที่อยู่บนเส้นทางงานของตนได้เพียง 1 ครั้งเท่านั้น ซึ่งจะอธิบายโดยละเอียดในหัวข้อถัดไป

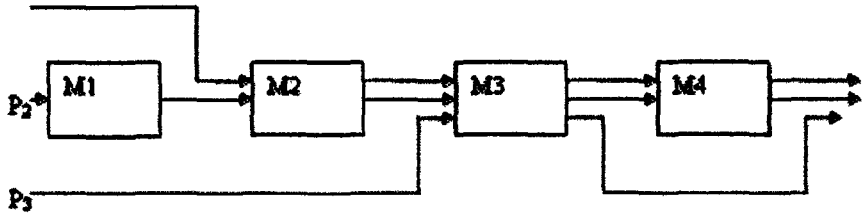
2.3 การผลิตแบบตามสั่ง (Job shop)

การผลิตแบบตามสั่ง ให้ความสำคัญกับตัวผลิตภัณฑ์เป็นสำคัญ เนื่องจากการผลิตแบบตามสั่งจะมีขั้นตอนหรือลำดับการดำเนินงานที่แตกต่างกันตามชนิดของผลิตภัณฑ์ โดยที่การผลิตจะต้องตอบสนองความต้องการของตลาดและลูกค้าเป็นหลัก โดยทั่วไปการผลิตแบบตามสั่งนี้ จะมีลำดับการผลิตที่ซับซ้อนกว่าการผลิตแบบตามการไหลและยังต้องพิจารณาขั้นตอนในระหว่างการผลิต เช่นการลำเลียง และขนถ่ายสินค้าระหว่างการผลิต (Intermediate product) จากอุปกรณ์หนึ่งไปยังอีกอุปกรณ์หนึ่ง ลักษณะของการผลิตแบบตามสั่งนั้น เป็นการผลิตแบบ make to order ที่มีปริมาณการผลิตไม่มาก เมื่อเทียบกับการผลิตแบบ make to stock เพราะว่ามีการผลิตที่ไม่สม่ำเสมอ ซึ่งประสิทธิภาพในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง (Job shop scheduling problem, JSSP) มีผลต่อประสิทธิภาพของการผลิตและคุณภาพของผลิตภัณฑ์ รูปแบบหรือสมมุติฐานที่ใช้กันทั่วไปของปัญหาการผลิตแบบตามสั่ง จะมีงานที่แตกต่างกัน n งาน จัดสรรลงใน m เครื่องจักรที่แตกต่างกัน โดยมีข้อกำหนดคือ

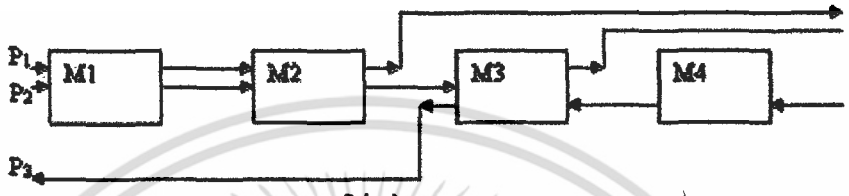
ก. งานแต่ละงานจะทำบนเครื่องจักรทีละ 1 เครื่อง
ข. ไม่มีงานไหนมีความสำคัญกว่างานอื่น
ค. ทุกการดำเนินการไม่สามารถขาดตอน (Interrupt)
ง. แต่ละเครื่องจักรสามารถทำงานเพียง 1 งานในเวลานั้น
จ. ไม่มีการกำหนดเวลาเริ่มต้นและเวลาส่งงาน (เอกชาติ หัตถา. 2543 : 37-38)

รูปที่ 2.2 แสดงถึงความแตกต่างระหว่างระบบการผลิตแบบไหลและการผลิตแบบตามสั่ง ซึ่งการผลิตแบบตามสั่งจะมีลักษณะที่คล้ายกับการผลิตแบบตามการไหล แต่ข้อแตกต่างที่ชัดเจนระหว่างการผลิตทั้ง 2 ชนิดนี้คือ ลำดับขั้นตอนในการผลิตย่อย (Task) เพราะ ระบบการผลิตแบบตามสั่ง จะมีลำดับการผลิตที่แตกต่างกันในแต่ละผลิตภัณฑ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Flow shop



Job shop

รูปที่ 2.2 แสดงลักษณะการผลิตแบบตามสั่งและแบบตามการไหล

เครื่องจักรอุปกรณ์ต่างๆ จะถูกรวมกันตามหน้าที่การใช้งานไว้ในสถานีการผลิต (Work Station) แยกเป็นหมวดหมู่ การเดินเครื่องจักรผลิตสินค้า จะผลิตสินค้าชนิดหนึ่งจนได้ปริมาณตามที่ต้องการ แล้วจึงเปลี่ยนไปผลิตสินค้าชนิดอื่นโดยใช้เครื่องจักรชุดเดิม

2.3.1 ข้อดีของการผลิตแบบตามสั่ง

ก. มีความยืดหยุ่นในการดำเนินงานสูงเมื่อเทียบกับการผลิตชนิดแบบไหล

ข. สามารถปรับเปลี่ยนแผนการดำเนินการผลิตได้อย่างรวดเร็วตามความต้องการของตลาดนั่นคือ มีความยืดหยุ่น (Flexible)

ค. สามารถผลิตผลิตภัณฑ์ให้มีความหลากหลายสูง เช่น การผลิตสินค้าที่มีสีต่างกัน การผลิตโต๊ะที่มีขาที่แตกต่างกัน

ง. เนื่องจากอุปกรณ์ที่ใช้เป็นเครื่องมือชนิดไม่เฉพาะทาง (General Purpose) ซึ่งสามารถนำไปใช้ในการดำเนินงานชนิดอื่นได้ เมื่อมีการปรับเปลี่ยนการใช้งานในบางส่วน

จ. เนื่องจากลักษณะงานและลำดับการดำเนินงานที่ไม่ซ้ำกันในแต่ละผลิตภัณฑ์ เป็นผลให้แรงงานเกิด การตื่นตัวในการทำงานและมีประสิทธิภาพในการทำงานเพิ่มขึ้นเมื่อเทียบกับการผลิตแบบไหล ที่มีความจำเจของการทำงาน

2.3.2 ข้อเสียของการผลิตแบบตามสั่ง

ก. เนื่องจากอุปกรณ์ที่ใช้เป็นชนิดไม่เฉพาะทาง ดังนั้นขั้นตอนการดำเนินงานในแต่ละอุปกรณ์ รวมทั้งค่าใช้จ่ายในแต่ละอุปกรณ์เมื่อเทียบกับการผลิตชนิด Flow shop จะมีค่าใช้จ่ายที่สูงกว่าเนื่องจากจะต้องมีบุคลากรหรือแรงงานที่มีการฝึกหัดในการใช้อุปกรณ์อย่างถูกต้อง

ข. เนื่องจากการดำเนินงานแบบตามสั่ง เป็นการดำเนินงานที่มีความซับซ้อนกว่าการดำเนินงานแบบไหล เป็นผลให้การดูแลรักษาความปลอดภัยขณะดำเนินงานจึงเป็นเรื่องที่สำคัญ และการบริหารวัสดุคงคลังยังเป็นเรื่องที่ต้องคำนึงถึงเป็นอย่างมากอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ตัวอย่างและวิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

2.4.1 ตัวอย่างการผลิตแบบตามสั่ง

การผลิตแบบตามสั่งจะผลิตตามความต้องการของลูกค้า ปริมาณไม่มาก แต่หลากหลาย โดยมีตัวอย่างผลิตภัณฑ์ดังเช่น การบริการคนไข้ที่เข้ารับการรักษาในโรงพยาบาล การผลิตเครื่องประดับแบบสั่งทำ การกลึงผลิตภัณฑ์ที่ต้องใช้หลายเครื่องกลึงที่มีลักษณะเฉพาะแตกต่างกัน รวมไปถึงการผลิตเครื่องปั้นดินเผาและระบบโรงงานประกอบชิ้นส่วนรถยนต์ เป็นต้น ตัวอย่างลักษณะรูปแบบของปัญหาการจัดตารางการผลิตแบบตามสั่ง เป็นดังตารางที่ 2.1

ตารางที่ 2.1 แสดงตัวอย่างปัญหาการผลิตแบบตามสั่ง

	ลำดับ 1		ลำดับ 2		ลำดับ 3	
	เครื่องจักร	เวลาดำเนินการ	เครื่องจักร	เวลาดำเนินการ	เครื่องจักร	เวลาดำเนินการ
งาน 1	1	71	2	46	3	82
งาน 2	3	66	1	50	2	94
งาน 3	1	39	2	91	3	57

จากตารางที่ 2.1 สามารถวิเคราะห์ได้ดังนี้

จำนวนงาน = 3 งาน

จำนวนเครื่องจักร = 3 เครื่องจักร

จำนวนการดำเนินการ 3 x 3 = 9 การดำเนินการ

แต่ละงานมี 3 การดำเนินการ โดยมีลำดับการดำเนินการดังเช่น งานที่ 1 คือ

การดำเนินการที่ 1 ทำบนเครื่องจักร 1 เวลาในการดำเนินการ 71 หน่วยเวลา

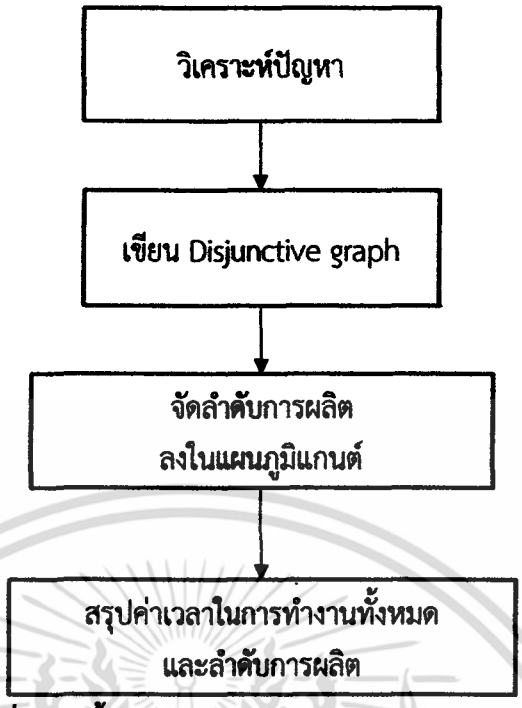
การดำเนินการที่ 2 ทำบนเครื่องจักร 2 เวลาในการดำเนินการ 46 หน่วยเวลา

การดำเนินการที่ 3 ทำบนเครื่องจักร 3 เวลาในการดำเนินการ 82 หน่วยเวลา

2.4.2 วิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

ขั้นตอนในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งมีหลากหลายวิธี แต่ผู้วิจัยยกตัวอย่างเพียง 1 วิธีคือการหาเส้นทางโดย Disjunctive graph โดยมีขั้นตอนดังรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ขั้นตอนการจัดตารางการผลิตแบบตามสั่ง

จากรูปที่ 2.3 เป็นการแสดงขั้นตอนการจัดตารางการผลิตแบบตามสั่ง วิธีหนึ่ง โดยใช้ Disjunctive graph มีขั้นตอนทั้งหมด 4 ขั้นตอน โดยขั้นตอนจะเริ่มจากการวิเคราะห์ปัญหา การเขียน Disjunctive graph หลังจากนั้นทำการจัดลำดับการผลิต และสุดท้ายสรุปค่าเวลาในการทำงานทั้งหมดและลำดับการผลิต โดยสมารถอธิบายแต่ละขั้นตอนได้ดังนี้

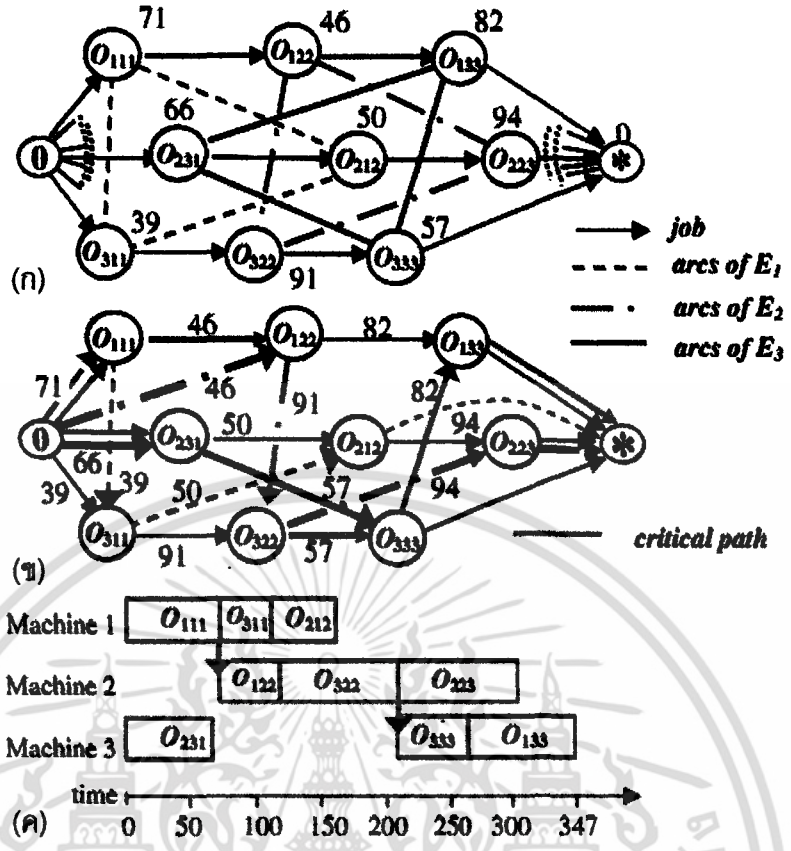
การวิเคราะห์ปัญหา

การวิเคราะห์ปัญหาเป็นขั้นตอนที่ต้องการหาจำนวนของงาน (Job) จำนวนของเครื่องจักรทั้งหมด (Machine) รวมไปถึงลำดับในการผลิตและเวลาในการผลิตแต่ละการดำเนินการ (Processing time) ดังได้วิเคราะห์ไปแล้วในหัวข้อ 2.4.1

การเขียน Disjunctive graph

ปัญหาของการจัดตารางการผลิตแบบตามสั่ง (JSP) สามารถเขียนให้อยู่ในรูปของ Disjunctive graph โดยมีรูปแบบ DG= (N,A,E_{jh}) เมื่อ N เป็นเซตของการดำเนินการจากจุด dummy start ไปจนถึงการดำเนินการสุดท้าย A คือเซตของเส้นที่เชื่อมต่อระหว่างการดำเนินการ E_{jh} คือเซตของเส้นที่ไม่เชื่อมต่อระหว่างการดำเนินการของงานใดๆ ที่กระทำบนเครื่องจักรเดียวกัน (Disjunctive arc) ดังแสดงในรูป 2.4 (Andrea. and Gino. 2007 : 505-506)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 (ก) Disjunctive graph, (ข) การแสดงลำดับการผลิตลงบน disjunctive graph, (ค) Gantt diagram ของลำดับการผลิต (Andrea. and Gino. 2007 : 506)

จากรูปที่ 2.4 ก. แสดง Disjunctive graph โดยในกราฟจะแสดงข้อมูล ค่าเวลาการดำเนินการ (Processing time) และลำดับการดำเนินการบนแต่ละเครื่องจักร โดยใช้สัญลักษณ์การดำเนินการในการบอกเอกลักษณ์คือ $O_{งาน,เครื่องจักร,ลำดับ}$ ส่วนรูปที่ 2.4 ข เป็นการแสดง โดย disjunctive graph ที่การลำดับของการดำเนินการ (Operation sequence) ของแต่ละเครื่องจักรแล้ว หลังจากนั้นนำมาเขียนให้อยู่ในรูปของแผนภูมิแกนต์ (Gantt diagram) ดังรูปที่ 2.4 ค

การจัดลำดับการผลิตลงในแผนภูมิแกนต์และการสรุปค่าเวลาในการทำงานทั้งหมดและลำดับการผลิต

แผนภูมิแกนต์ (Gantt diagram) ถูกพัฒนาขึ้นโดยเฮนรี แกนต์ (Henry Gantt) เพื่อช่วยแก้ปัญหาเกี่ยวกับการจัดลำดับงาน และใช้ในการแสดงผลของการจัดตารางการผลิต ซึ่งจะเป็นการแสดงความสัมพันธ์ระหว่างทรัพยากร เช่น เครื่องจักร กับเวลา เป็นต้น เมื่อเขียน Disjunctive graph ใหม่ในรูปของแผนภูมิแกนต์ก็สามารถทราบถึงเส้นทางวิกฤต (Critical path) คือเส้นทางที่เชื่อมต่อจุดเริ่มต้นไปถึงจุดสุดท้าย เมื่อทราบเส้นทางวิกฤตจะทำให้ทราบช่วงกว้างของเวลาการทำงาน สิ้นที่สุดหรือค่าเวลาในการผลิตทั้งหมด (Makespan) ซึ่งก็คือ เวลาแล้วเสร็จของงานที่มีเวลาแล้วเสร็จช้าที่สุดในรอบการจัดตารางการผลิตแต่ละรอบ (Maximum Completion Time) (ปัญจพร แพใหญ่. 2549 : 7) จากรูปที่ 2.4 ค. สามารถสรุป ลำดับของการดำเนินการ (Operation sequence) เส้นทางวิกฤตและค่าเวลาในการทำงานทั้งหมด ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับของการดำเนินการ (Operation sequence)

$$O_{111} \rightarrow O_{231} \rightarrow O_{311} \rightarrow O_{122} \rightarrow O_{322} \rightarrow O_{212} \rightarrow O_{333} \rightarrow O_{223} \rightarrow O_{133}$$

เส้นทางวิกฤต (Critical path)

$$O_{111} \rightarrow O_{122} \rightarrow O_{322} \rightarrow O_{333} \rightarrow O_{133}$$

ค่าเวลาในการทำงานทั้งหมดซึ่งจากแผนภูมิแกนต์ในรูปที่ 2.4 ค จะได้ค่าเวลาในการทำงานทั้งหมดเท่ากับ 347 หน่วยเวลา

2.5 มาตรฐานที่ใช้วัดสมรรถนะการจัดตารางการผลิต

ดัชนีที่ใช้ในการวัดสมรรถนะของการจัดตารางการผลิตคือ ระยะเวลาที่ใช้ในการดำเนินการผลิตทุกผลิตภัณฑ์เสร็จหรือค่าเวลาในการทำงานทั้งหมด (Makespan) ซึ่งเป็นค่าที่แสดงถึงการตอบสนองของกระบวนการที่มีต่อปริมาณผลิตภัณฑ์ที่ต้องผลิตตามจำนวนวัตถุดิบภายในคลัง (Stock) ว่ามีความเปลี่ยนแปลงมากหรือน้อยอย่างไร

ถ้าค่าเวลาในการทำงานทั้งหมด ที่คำนวณได้มีค่าน้อย ย่อมจะแสดงว่าการผลิตมีความคล่องตัวและสามารถดำเนินงานตอบสนองกับการวางแผนในขั้นต้นได้เป็นอย่างดี

ถ้าค่าเวลาในการทำงานทั้งหมด ที่คำนวณได้มีค่ามาก ย่อมจะแสดงว่ามีความสามารถในการดำเนินการผลิตรวมทั้งจัดสรรทรัพยากรการผลิตไม่ดีเท่าที่ควร ซึ่งอาจจะกระทบโดยตรงต่อปริมาณที่ตลาดต้องการได้ในช่วงเวลานั้นๆ รวมไปถึงปริมาณคงคลัง ที่ไม่สามารถระบายออกได้อย่างเหมาะสมในช่วงเวลาที่กำหนด ซึ่งอาจเป็นผลให้เกิดความเสียหายขึ้นกับวัตถุดิบได้สำหรับกรณีที่วัตถุดิบมีการเสื่อมสภาพหรือหมดอายุได้ และถ้ามองในด้านการเงิน จะส่งผลถึงต้นทุนในการผลิตที่เพิ่มขึ้น ในการจ้างงานแบบล่วงเวลา (Over time)

ในกรณีของระบบการผลิตแบบตามสั่ง จะเน้นในเรื่องการตอบสนองต่อความต้องการของลูกค้าโดยตรง (Make to order) ซึ่งในที่นี้คือ ระยะเวลาที่ใช้ในการผลิตผลิตภัณฑ์ทุกชนิด จะต้องสอดคล้องกับระยะเวลาที่ลูกค้ากำหนด (Due date) โดยระยะเวลาดังกล่าวจะต้องมีค่าที่น้อยกว่าค่าระยะเวลาที่ลูกค้ากำหนด

2.6 ข้อจำกัดที่เกี่ยวข้องกับการจัดตารางการผลิต

เอกชาติ หัตถา (2543 : 47-48) ได้กล่าวถึงข้อจำกัดที่เกี่ยวข้องกับการจัดตารางการผลิต ไว้ดังต่อไปนี้

- **ข้อจำกัดที่เกี่ยวกับลำดับก่อนหลัง (Precedence Constraint)**

ข้อจำกัดที่เกี่ยวกับลำดับก่อนหลัง คือ ข้อจำกัดที่เกี่ยวกับลำดับก่อนหลังที่ดำเนินการผลิต Intermediate produce อย่างใดอย่างหนึ่งเสร็จสิ้นก่อนที่ Intermediate produce อื่นจะถูกดำเนินการผลิตในหน่วยการผลิตถัดไป โดยขั้นตอนดังกล่าวจะไม่สามารถสลับที่กันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ข้อจำกัดที่เกี่ยวกับทรัพยากร (Resource Constraint)**

ข้อจำกัดที่เกี่ยวกับทรัพยากร คือ ข้อจำกัดที่เกี่ยวกับทรัพยากรทั้งหมดที่เกี่ยวข้องกับการผลิตในทุกช่วงเวลาในการดำเนินงาน ทรัพยากรต่างๆ คือ แรงงานคน, อุปกรณ์หรือเครื่องจักรที่ใช้, วัตถุดิบ และสาธารณูปโภคต่างๆ

- **เวลาการส่งมอบ (Deadline หรือ Due date)**

ข้อจำกัดนี้จะเกี่ยวข้องกับระยะเวลาที่ใช้ในการผลิตโดยตรง ซึ่งในที่นี้จะต้องตอบสนองความต้องการของลูกค้าในระยะเวลาที่กำหนด

2.7 การแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

การแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง (Job shop scheduling, JSP) มีการวิจัยมาตั้งแต่ปี ค.ศ. 1963 โดย Muth และ Thompson หลังจากนั้นก็มีการพัฒนาวิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งมาโดยตลอด (Fan. and Zhang. 2010 : 282)

โดยอนิรุจน์ ตันเทิดทิตย์ (2546 : 7-9) แบ่งวิธีการจัดลำดับงานออกเป็น 2 ประเภทคือ

2.7.1 วิธีดีเทอร์มินิสติค (Deterministic Scheduling)

วิธีดีเทอร์มินิสติค คือการหาคำตอบที่ดีที่สุด โดยในการแก้ปัญหาจำเป็นต้องใช้เวลามากในการคำนวณเพื่อหาคำตอบ โดยเฉพาะปัญหาที่มีลักษณะของข้อมูลและตัวแปรมาก แต่คำตอบที่ได้จะเป็นคำตอบที่เหมาะสมที่สุด (Optimal Solution) โดยใช้เทคนิคเชิงคณิตศาสตร์ (Mathematical modeling) ในการหาคำตอบดังนี้

- **กำหนดการเชิงเส้น (Linear- programing)**

กำหนดการเชิงเส้นเป็นตัวแบบทางคณิตศาสตร์ที่เกี่ยวกับการจัดสรรทรัพยากรที่มีอยู่จำกัดให้เกิดประโยชน์มากที่สุด ลักษณะของกำหนดการเชิงเส้นคือ ฟังก์ชันจุดประสงค์ (Objective function) ที่แทนเป้าหมายและข้อจำกัดต่างๆ ต้องเป็นเชิงเส้นทั้งหมด โดยมีการกำหนดตัวแปรการตัดสินใจ (Decision variable) และข้อจำกัดต่างๆให้สอดคล้องกับระบบการผลิตนั้นๆ (วิภาวรรณ. 2543 : 11)

- **กำหนดการเชิงจำนวนเต็ม,กำหนดการจำนวนเต็มผสม (Integer-programming , Mixed-integer Programming)**

กำหนดการเชิงจำนวนเต็มเป็นวิธีทางตัวเลขจำนวนเต็มเพื่อหาผลลัพธ์ที่ดีที่สุดการหาคำตอบโดยวิธีนี้จะใช้การคำนวณเป็นอย่างมากและยังพบว่ายังมีข้อจำกัดในการสร้างสมการทางคณิตศาสตร์สำหรับบางปัญหา (Baker. 1974 : 206)

- **วิธีบรานช์แอนด์บาวด์ (Branch and Bound)**

วิธีบรานช์แอนด์บาวด์ประกอบด้วย 2 ขั้นตอน คือ ขั้นตอนบรานช์และบาวด์ ขั้นตอนบรานช์เป็นกระบวนการแบ่งส่วนของปัญหาที่มีขนาดใหญ่ออกเป็นปัญหาย่อย ขั้นตอนบาวด์เป็นกระบวนการของการคำนวณค่าขอบเขตล่าง (Lower Bound) ที่ดีที่สุดสำหรับปัญหาย่อยนั้น ประสิทธิผลจะขึ้นอยู่กับขอบเขตล่างที่ดีซึ่งจะทำให้ได้ผลลัพธ์ที่ดีที่สุด (ปัญจพร แพใหญ่. 2549 : 15)

2.7.2 วิธีฮิวริสติค (Heuristic Scheduling)

วิธีฮิวริสติค คือวิธีที่พัฒนามาจาก วิธีแบบดีเทอร์มินิสติค แต่เวลาที่ใช้ในการคำนวณลดลงอยู่ในช่วงที่ยอมรับได้ แต่อย่างไรก็ตามคำตอบที่ได้รับจะไม่ใช่คำตอบที่ดีที่สุดจริงๆเหมือนวิธีดีเทอร์มินิสติค แต่จะเป็นคำตอบที่มีค่าใกล้เคียงกับคำตอบที่ดีที่สุดในช่วงเวลานั้น การหาคำตอบแบบฮิวริสติค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นที่ได้รับความนิยมในการใช้งานอย่างมาก (Fan. 2010 : 285-286) เทคนิคของการหาคำตอบแบบฮิวริสติกที่มีผู้คิดค้น และนิยมใช้งานในปัจจุบันมีดังนี้

- **วิธี Tabu search**

วิธี Tabu search เป็นวิธีหาคำตอบโดยการค้นหาคำตอบด้วยการเปรียบเทียบข้อมูลที่ติดกัน (Local Search) ในชุดข้อมูลทั้งหมดและจะหาคำตอบไปเรื่อยๆ จนได้ค่าคำตอบที่ดีที่สุดหรือจนกระทั่ง ไม่สามารถหาคำตอบที่ดีกว่าได้ โดยเมื่อได้คำตอบที่ดีกว่าจะทำการบันทึกลงในฐานข้อมูล (Tabu list) จากนั้นจะทำการค้นหาคำตอบใหม่ไปเรื่อยๆ หากค่าคำตอบที่ได้ใหม่มีทิศทางที่ไม่ดีไปกว่าเดิมก็จะทำการเปลี่ยนเส้นทางในการคำนวณ (Ching. and Fang. 1999 : 112-113)

- **วิธี Simulated Annealing**

วิธี Simulated Annealing เป็นการหาคำตอบโดยค้นหาด้วยการเปรียบเทียบข้อมูลที่ติดกันในชุดข้อมูลทั้งหมดเช่นกัน แต่ต่างจากวิธี Tabu search คือจะใช้การสุ่มจุดที่มีแนวโน้มจะก่อให้เกิดคำตอบที่ดีที่สุด และใช้สมการจุดมุ่งหมายเคลื่อนที่หาคำตอบไปตลอดทุกชุดข้อมูลโดยเปรียบเทียบกับค่าคำตอบที่ได้ในปัจจุบัน (Mamalis and Malagardis. 1996 : 66) เทคนิคการจำลองสถานการณ์ (Simulation) มีการนำเทคนิคการจำลองสถานการณ์มาใช้แก้ปัญหาการจัดตารางการผลิตเนื่องจากการจำลองสถานการณ์มีความยืดหยุ่นในการสร้างแบบจำลองระบบการผลิตได้หลายรูปแบบและคำนึงถึงข้อจำกัดต่างๆ ได้มากมาย นอกจากนี้ยังสามารถทดสอบสถานการณ์ต่างๆ ที่เกิดขึ้นได้ทั้งสถานการณ์ที่แน่นอนและไม่แน่นอน และยังรองรับกับข้อมูลที่มีการแจกแจงทางสถิติได้ด้วย มีการนำการจำลองสถานการณ์มาใช้ทดสอบสมมุติฐานด้านต่างๆ เพื่อหาผลการจัดตารางการผลิตที่เหมาะสมที่สุด (ยงยุทธ ลิขิตพัฒนะกุล. 2546 : 12)

- **วิธีการเชิงพันธุศาสตร์** (Genetics Algorithm)

วิธีการเชิงพันธุศาสตร์ เป็นการหาคำตอบด้วยการแตกสายพันธุ์ของข้อมูล จากรุ่นหนึ่ง (Parent) สู่อีกรุ่นหนึ่ง (Offspring) โดยการแตกสายพันธุ์นั้นอาจทำให้เกิดการกลายพันธุ์ของข้อมูล ข้อมูลที่เกิดขึ้นใหม่นั้นถ้ามีค่าที่เลวลงจะถูกกำจัดทิ้งในขณะที่ข้อมูลที่มีค่าที่ดีขึ้นจะแตกสายพันธ์ต่อไป แต่กระบวนการทั้งหมดที่ใช้หาคำตอบจะใช้เวลานาน

- **วิธี Beam Search**

วิธี Beam Search จะมีลักษณะคล้ายกับวิธีแบบดีเทอร์มินิสติคแบบบรานช์แอนด์บาวด์ Branch-and-Bound แต่จะตัดการพิจารณาในเส้นทางที่เหมือน หรือคล้ายจะเป็นเส้นทางที่ไม่ก่อให้เกิดคำตอบออกไป การตัดการพิจารณาเช่นนี้จะสามารถลดเวลาการคำนวณลงได้ (Sabuncuoglu. and Bayiz. 1998 : 392)

- **ระบบโครงข่ายประสาทเทียม (Neural Network)**

ระบบเครือข่ายประสาทเทียม เป็นการเลียนแบบระบบประสาทในสมอง โดยการแก้ปัญหาจะสร้างส่วนรับข้อมูลและทำการศึกษาหาคำตอบจากข้อมูลที่รับเข้ามา ซึ่งส่วนรับข้อมูลจะสร้างรูปแบบในการคำนวณ เพื่อใช้ในการหาคำตอบสำหรับชุดข้อมูลใหม่

- **เทคนิคกฎเกณฑ์การจ่ายงาน (Dispatching Rules)**

เทคนิคนี้ออกแบบมาเพื่อหาวิธีการที่ดี (Good Solution) และสามารถแก้ปัญหาที่มีความซับซ้อนได้ เทคนิคกฎเกณฑ์การจ่ายงานมีกฎเกณฑ์การจ่ายงานให้เลือกใช้เป็นจำนวนมาก กฎเกณฑ์การจัดตารางการผลิตแบ่งออกเป็นหลายๆระดับ คือ กฎเกณฑ์ตามลำดับความสำคัญอย่างง่าย เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPT , EDD , FIFO , และ MINSLACK กฎเกณฑ์ที่มีการรวมกัน (Combination) ระหว่างกฎเกณฑ์ในระดับแรกและกฎเกณฑ์ที่อ้างอิงกับดัชนีความสำคัญ (ยงยุทธ ลิขิตพัฒนะกุล. 2546 : 11)

- **เทคนิคค้นหาใกล้เคียง (Neighborhood Search)**

เป็นวิธีการค้นหาคำตอบเพื่อค้นหาคำตอบที่ดีที่สุดได้อย่างมีประสิทธิภาพและรวดเร็วโดยใช้เทคนิคหลายๆอย่าง เช่น Tabu Search , Simulated Annealing และ Genetic Algorithms (ยงยุทธ ลิขิตพัฒนะกุล. 2546 : 11)

- **เทคนิค Fuzzy Logic**

มีการใช้เทคนิค Fuzzy Logic ในการแก้ปัญหาการจัดตารางการผลิตที่มีพารามิเตอร์ไม่แน่นอน (Uncertainly) ได้แก่เวลาทำงาน (Process Time) ข้อจำกัดต่างๆ เวลาเตรียมเครื่องจักร แล้วนำความไม่แน่นอนเหล่านั้นนำมาแทนด้วย Fuzzy Number มีการใช้หลักการของ Fuzzy Logic เพื่อรวมกฎเกณฑ์การจ่ายงานสำหรับปัญหาที่มีเกณฑ์มากกว่า 1 อย่าง หรือปัญหาการใช้เวลาเตรียมเครื่องจักรน้อยที่สุด (ยงยุทธ ลิขิตพัฒนะกุล. 2546 : 12)

- **ทฤษฎี Bottleneck Schedule**

ทฤษฎี Bottleneck Schedule คือ การพิจารณาจัดลำดับการทำงานสำหรับการผลิตที่มีการใช้เครื่องจักร m เครื่อง (Machines) และมีการรับงานมาทำ n งาน (Jobs) ให้สำเร็จลุล่วงไปโดยเร็วที่สุด โดยจะพิจารณาจากเครื่องจักรคอขวด (เครื่องจักรคอขวด คือ เครื่องจักรที่มีเวลารวมของการทำงานมากที่สุด) เป็นหลัก หลังจากที่เรากำหนดเครื่องจักรคอขวดแล้ว ก็จะทำการจัดลำดับงานของทุกงานที่จะต้องทำบนเครื่องจักรคอขวดหลังจากนั้นก็จะจัดลำดับงานการทำงานของเครื่องจักรที่มีเวลารวมของการทำงานที่มีอันดับถัดไป (วุฒิพงศ์ วุฒิจริยากุล. 2545:2)

- **วิธีอาณานิคมมด (Ant Colony Optimization)**

Dorigo et.al. (1996 : 29-31) ได้เสนอแนวคิดวิธีการหาคำตอบโดยใช้วิธีอาณานิคมมด ซึ่งการค้นหาอาศัยการเลียนแบบพฤติกรรมการหาอาหารของมดจริง เทียบได้กับการหาเส้นทางจากรังมดไปยังแหล่งอาหาร โดยมดจะอาศัยสารเคมีที่เรียกว่า ฟีโรโมน (Pheromone) ที่ปล่อยไว้บนเส้นทาง ใช้ในการจดจำเส้นทางและใช้ในการสื่อสารกับมดตัวอื่น โดยจะอธิบายเพิ่มเติมในหัวข้อถัดไป

ปัญหาการจัดตารางการผลิตเป็นปัญหาที่แปรผันตามขนาดของปัญหา (จำนวนงานและเครื่องจักร) ซึ่งจัดอยู่ในประเภทปัญหา NP (Nondeterministic Polynomial) เพราะฉะนั้นถ้าปัญหาที่มีขนาดของปัญหาใหญ่ก็จะใช้เวลามาก (Fan. and Zhang. 2010 : 282)

2.8 วิธีอาณานิคมมด (Ant Colony Optimization, ACO)

จากที่กล่าวไปแล้วข้างต้นว่าวิธีอาณานิคมมด อาศัยการเลียนแบบพฤติกรรมการหาอาหารของมดจริง โดยมดจะอาศัยสารเคมีที่เรียกว่า ฟีโรโมน ปล่อยไว้บนเส้นทาง เพื่อจดจำเส้นทางและใช้ในการสื่อสารกับมดตัวอื่น วิธีนี้มีพื้นฐานมาจากการทดลองแก้ไขปัญหาเส้นทางสองเส้นทาง (Double bridge) ของ Deneubourg ในปี 1990 โดยทำการทดลองกับมดประเทศอาเจนตินา เพื่อศึกษาพฤติกรรมการเลือกเส้นทางของมด ในการปล่อยฟีโรโมนบนแต่ละเส้นทาง เหตุผลที่เลือกมดจากอาเจนตินา เนื่องจากเป็นมดที่มีประสาทสัมผัสทางการมองเห็นไม่ดีนัก ผลการศึกษาได้สรุปไว้ในบทความของ Dorigo and Gambarella (1997: 2-3) ซึ่งได้แสดงวิธีการหาเส้นทางของมด โดยมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะใช้การดมกลิ่นของฟีโรโมนที่มีการปล่อยไว้บนเส้นทาง มดตัวต่อมาจะพิจารณาจากความเข้มข้นของกลิ่นฟีโรโมนเพื่อช่วยในการตัดสินใจเลือกเส้นทางเดินต่อไป โดยมี 10 ขั้นตอนดังนี้

1. มดในธรรมชาติ จะออกจากรังไปหาอาหาร โดยในการเดินทางครั้งแรก ยังไม่มีฟีโรโมนอยู่บนเส้นทางเพราะฉะนั้นมดชุดแรกจะเดินทางในรูปแบบของการสำรวจดังรูปที่ 2.5

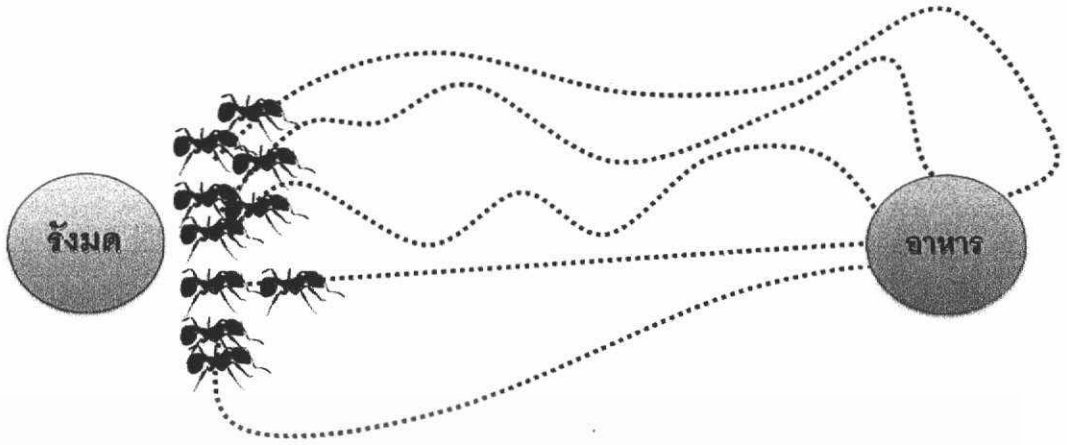
รูปที่ 2.5 การเดินทางออกจากรังไปหาอาหารของมดชุดแรก

2. มดชุดแรกทำการสำรวจหาอาหารและได้ไปถึงแหล่งอาหาร โดยมดแต่ละตัวได้ปล่อยฟีโรโมนไว้บนเส้นทางที่แตกต่างกันดังรูปที่ 2.6

รูปที่ 2.6 หลังจากมดชุดแรกออกเดินทางไปถึงแหล่งอาหาร

3. เมื่อมดสำรวจชุดแรกเดินกลับมาถึงรัง มดชุดที่สองซึ่งจะเป็นมดงานจะทำหน้าที่ขนอาหาร มดชุดนี้ จะเลือกเส้นทางจากการสุ่มและความเข้มข้นของฟีโรโมน แต่เนื่องจากความเข้มข้นยังไม่เด่นชัด เพราะฉะนั้นการเลือกเส้นทางจะมีแนวโน้มเป็นการสุ่ม โดยในรูปที่ 2.7 จะยกตัวอย่าง เมื่อมีมด 10 ตัว เลือกไปใน 5 เส้นทาง เส้นทางละ 2 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



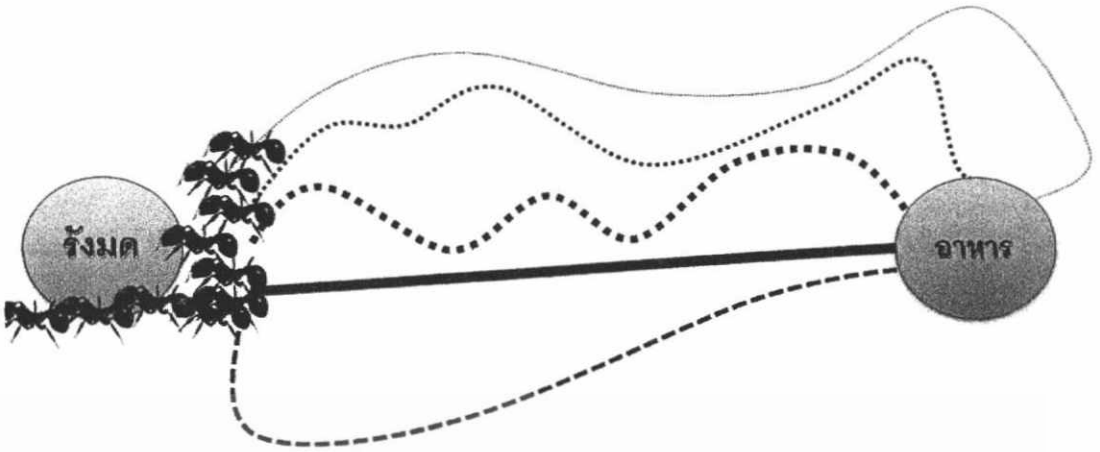
รูปที่ 2.7 มดชุดที่สองออกเดินทางไปยังแหล่งอาหาร

4. เมื่อมดชุดที่สองเดินทางถึงแหล่งอาหาร หลังจากนั้นจะเดินทางกลับ มดจะมีการสื่อสารกันระหว่างมด หลังจากนั้นมดจะเลือกเส้นทางที่ระยะทางสั้นกว่า (เลือกเส้นทางของมดที่มาถึงเป็นตัวแรก)

รูปที่ 2.8 มดชุดที่สองเลือกเส้นทางในการเดินทางกลับมารัง

5. เมื่อมดชุดที่สองเดินทางกลับมาถึงรัง เส้นทางที่มดเลือกเดินมากที่สุด และสั้นที่สุด จะมีความเข้มข้นของฟีโรโมนมากที่สุด เพราะมีสารฟีโรโมนของมดทุกตัวที่เดินผ่าน และใช้เวลาในการเดินทางน้อย ทำให้การระเหยมีน้อย จากรูปที่ 2.9 จะพบว่าเส้นทางที่มดเลือกเดินมากจะมีความเข้มข้นของฟีโรโมนสูง แสดงเป็นเส้นทึมหนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 มดชุดที่สองเดินทางกลับมาถึงรัง

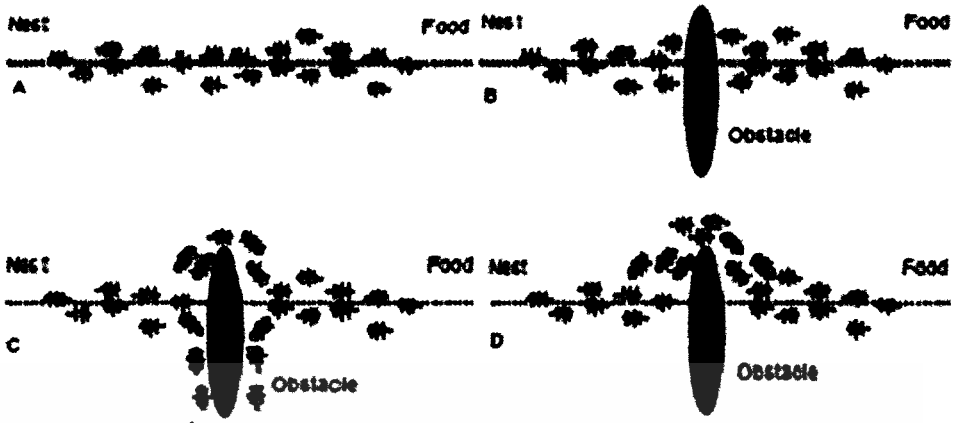
6. เมื่อมดชุดอื่นๆ จะใช้ความเข้มข้นของฟีโรโมนที่มีอยู่บนเส้นทาง ในการเลือกเส้นทางที่สั้นที่สุด คือเลือกจากเส้นทางที่มีความเข้นข้นของฟีโรโมนสูงที่สุด

รูปที่ 2.10 การเลือกเส้นทางของมดตัวอื่นๆ

โดยปัจจัยที่มีผลต่อความเข้มข้นคือ การระเหย (Evaporation) ของฟีโรโมน ซึ่งมีอัตราการระเหยนั้นขึ้นอยู่กับเวลา ถ้าความเร็วในการเดินของมดคงที่ และระยะทางของเส้นทางแต่ละเส้นทางแตกต่างกัน ระยะทางที่ยาวกว่าต้องใช้เวลาในการเดินทางนานกว่า ซึ่งจะทำให้ความเข้มข้นของฟีโรโมนน้อยกว่าระยะทางสั้น ทำให้มดตัวอื่นจะมีความน่าจะเป็นในการเลือกเส้นทางที่มีความเข้มข้นของฟีโรโมนมากกว่าซึ่งก็คือเส้นทางที่สั้นกว่านั่นเอง ดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 2.11 แสดงการหาเส้นทางของมดในธรรมชาติ
(Dorigo. and Gambardella. 1997 : 3)

Stützle. and Dorigo. (1999 : 5-6) ได้กล่าวถึงอัลกอริทึมระบบมด (Ant System) ของการแก้ปัญหาคนเดินทางขายของ (Travelling Salesman Problem ,TSP) สามารถนำมาประยุกต์ใช้ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง (Job-shop scheduling) โดยจะมีกฎการเปลี่ยนสถานะ (State transition rule) ดังนี้

$$O_n = \begin{cases} max\left\{\tau_p(O_p, O_{nt'})^{\alpha}\eta(O_p, O_{nt'})^{\beta}\right\} & \text{,ถ้า } q \le q_0 \\ P & \text{,ถ้า } q > q_0 \end{cases} \qquad (2.1)$$

โดย O_n = การดำเนินการถัดไปที่มดจะเลือก
τ_p = ความเข้มของฟีโรโมนตำแหน่งปัจจุบัน
η = ค่าสัมประสิทธิ์ทางฮิวริสติค
O_p = การดำเนินการในขณะนั้น
$O_{nt'}$ = การดำเนินการถัดไป
q = จำนวนจริงที่ได้จากการสุ่มตัวเลขที่มีการแจกแจงแบบยูนิฟอร์ม $(0 \le q \le 1)$
q_0 = พารามิเตอร์การสำรวจ $(0 \le q_0 \le 1)$
α = ค่าน้ำหนักของฟีโรโมน
β = ค่าน้ำหนักของฮิวริสติค
P = การดำเนินการที่สามารถกระทำได้ (Feasible move) โดยเลือกจากกฎสัดส่วนของการสุ่ม (Random proportional rule) ดังนี้

$$P_a = \frac{\tau_p(O_p,O_{nt'})^{\alpha}\eta(O_p,O_{nt'})^{\beta}}{\sum_{O_{nt'}\in C_L}\left\{\tau_p(O_p,O_{nt'})^{\alpha}\eta(O_p,O_{nt'})^{\beta}\right\}} \qquad (2.2)$$

โดย P_a = ความน่าจะเป็นของแต่ละการดำเนินการ
C_L = รายชื่อการดำเนินการที่สามารถทำต่อไป (Candidate list)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
123038

จากสมการที่ 2.1 ค่า q_0ที่กำหนดขึ้นมาจะนำมาพิจารณากับค่า q ที่สุ่มได้ เมื่อค่า q น้อยกว่าหรือเท่ากับ q_0 มดจะเลือกจากการดำเนินการที่มีค่าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคมากที่สุดจากรายชื่อการดำเนินการที่สามารถทำต่อไป แต่ถ้าค่า q มากกว่า q_0 (ตั้งค่า q_0 เข้าใกล้ 0 จะทำให้ระบบให้ความสำคัญกับการสำรวจ) มดจะเลือกจากสมการที่ 2.2 โดยจะเลือกจากความน่าจะเป็นของค่าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคจากรายชื่อการดำเนินการที่สามารถทำต่อไปถ้าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคต่ำ ความน่าจะเป็นที่มดจะเลือกการดำเนินการนั้นจะมีค่าต่ำ ในทางกลับกันถ้าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคสูง ความน่าจะเป็นที่มดจะเลือกการดำเนินการนั้นจะมีค่าสูง โดยผลรวมของความน่าจะเป็นทั้งหมดของการดำเนินการที่ยังไม่ได้ทำมีค่าเท่ากับ 1 และด้วยเหตุนี้ทำให้การหาคำตอบด้วยวิธีอาณานิคมมดสามารถเลือกการดำเนินการถัดไปที่หลากหลาย ซึ่งอาจจะนำไปสู่เส้นทางอื่นที่ดีกว่าเส้นทางเดิมที่มีอยู่

วิธีอาณานิคมมดได้มีการพัฒนามาอย่างต่อเนื่อง เริ่มจากการค้นพบระบบมด (Ant system) ใน ค.ศ. 1991 โดยได้ถูกคิดค้นและพัฒนาขึ้นโดย Marco Dorigo และคณะ

ต่อมา Dorigo (1992) พัฒนา Ant system ให้มีประสิทธิภาพดีขึ้น โดยเรียกการพัฒนานี้ว่า Elitist strategy โดยที่ส่วนการทำงานเริ่มต้นจะเหมือนกับ Ant system แต่จะเพิ่มในส่วนของการเก็บค่าคำตอบที่ดีที่สุดของแต่ละรอบ (Best so far tour) เพื่อการเพิ่มร่องรอยฟีโรโมน (Update pheromone trail) โดยจะเก็บค่าที่ดีที่สุดของรอบที่หนึ่ง แล้ว Update pheromone ซึ่งเส้นทางที่เป็นค่าคำตอบที่ดีที่สุดของแต่ละรอบ จะมีพจน์ที่เพิ่มขึ้นมาเพื่อทำให้เส้นทางที่เป็นค่าคำตอบที่ดีที่สุดของแต่ละรอบ จะมีปริมาณฟีโรโมนมากกว่าเส้นทางที่มดผ่านปกติทั่วไป การพัฒนาดังกล่าวทำให้ระบบมด (Ant system) เปลี่ยนเป็น Elitist ant system

หลังจากนั้น Stutzle กับ Hoos (2000) ได้พัฒนา Max-Min Ant System มาจากระบบมด โดยพัฒนาจากระบบมด 4 อย่างด้วยกัน คือ 1) ค่าของพจน์ที่เพิ่มขึ้นมาจะมีค่าเป็น หนึ่งส่วนระยะทางก็ต่อเมื่อเป็นรอบที่ดีที่สุดของรอบการคำนวณนั้น 2) การกำหนดค่าของฟีโรโมนให้อยู่ในช่วงที่สมการกำหนด เพื่อจำกัดขอบเขตของเส้นทางที่ดีที่สุดเพียงหนึ่งช่วงเท่านั่น ทำให้หาเส้นทางที่ดีที่สุดได้อย่างรวดเร็ว 3) ค่าฟีโรโมนเริ่มต้นจะมีค่าตัวแปรการระเหยของปริมาณฟีโรโมนไว้ในตอนเริ่มต้น 4) ถ้าปริมาณฟีโรโมนเริ่มต้น เริ่มมีค่าคงที่หรือไม่มีการเพิ่มขึ้นแล้วก็จะสร้างจำนวนรอบที่แน่นอนสำหรับการคำนวณครั้งต่อไป

วิธีที่นำมาใช้ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งคือ วิธีอาณานิคมมด (Ant Colony Optimization, ACO) หรือระบบอาณานิคมมด (Ant colony system, ACS) พัฒนาโดย Dorigo และ Gambardella มีความต่างจากระบบมดคือ 1) ACS จะพัฒนาในส่วนของการจำเส้นทางในการเดินของมด โดยจะทำให้มดมีประสบการณ์ในการจำเส้นทางมากขึ้นและมีผลต่อการตัดสินใจในการเลือกเส้นทางมากขึ้นด้วย 2) การระเหยของฟีโรโมนและการเพิ่มค่าฟีโรโมนจะทำในส่วนที่เป็นเส้นทางที่ดีที่สุดเท่านั้น 3) ในแต่ละเส้นทางที่มดเดินผ่านไปนั่น ฟีโรโมนจะมีการระเหยเพื่อที่จะทำให้เกิดการเพิ่มเส้นทางหรือโอกาสในการเลือกเส้นทางอื่นๆ ซึ่ง Dorigo ได้สรุปคุณลักษณะของวิธีการหาคำตอบด้วยวิธีอาณานิคมมดดังนี้

- เป็นอัลกอริทึมที่เอนกประสงค์ (Versatile) คือสามารถใช้ได้กับปัญหาได้หลากหลายลักษณะ
- มีความคงทน (Robust) คือสามารถใช้งานได้ในปัญหาที่มีการเปลี่ยนแปลงค่าพารามิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อาศัยพื้นฐานประชากร (Population Based) คือใช้การป้อนกลับที่เป็นประโยชน์ (Positive Feedback) ช่วยในการหาคำตอบ ซึ่งจะทำให้ระบบมีความยืดหยุ่นที่จะดำเนินการค้นหาแบบขนาน (Parallel implementation)

จากคุณลักษณะเด่นข้างต้นของวิธีอาณานิคมมดทำให้มีการนำมาใช้แก้ปัญหาได้หลากหลายรูปแบบ ดังเช่น

- ปัญหาการเดินทางของพนักงานขาย (Travelling salesman problem, TSP) เป็นปัญหาทางด้านเครือข่ายที่หาเส้นทางที่สั้นที่สุดของพนักงานขาย ในปัญหานี้พนักงานขายจะต้องเดินทางไปในทุกๆเมือง แต่ละเมืองจะมีระยะห่างที่แตกต่างกัน คำตอบของปัญหานี้ก็คือลำดับของการเดินทางที่ให้ระยะทางที่สั้นที่สุด (Dorigo et. al. 1996 : 31)

- ปัญหาการจัดสรรงานกำลังสอง (Quadratic assignment Problem, QAP) ปัญหาการจัดสรรงานกำลงสอง เป็นปัญหาการจัดสรรงาน n งานให้กับทรัพยากร n ทรัพยากรโดยมีเป้าหมายเพื่อให้ได้ค่าใช้จ่ายรวมของความสัมพันธ์กำลังสอง (Total quadratic interactive cost) ต่ำที่สุด (Dorigo et. al. 1996 : 30, วนิดา และคณะ 2550 : 327)

- ปัญหา Constraint Satisfaction Problems (CSP) คือ การกำหนดเงื่อนไขเพื่อช่วยในการแก้ปัญหา โดยการกำหนดตัวแปร (Variables) โดเมน (Domains) และข้อจำกัด (Constraint) เป็นปัญหาประเภทที่ต้องทดลองไปเรื่อย ๆ เพื่อค้นหาคำตอบ (Pimount, S. and Solnon C. 2000 : 100)

- ปัญหาการจัดตารางสอนในสถาบันอุดมศึกษา (University course timetabling problem) เป็นการจัดสรรวิชาเรียนลงตารางเวลา ตามข้อกำหนดและเงื่อนไงต่างๆ (Socha, K. Knowles, J. and Sampels, M. 2002. : 1)

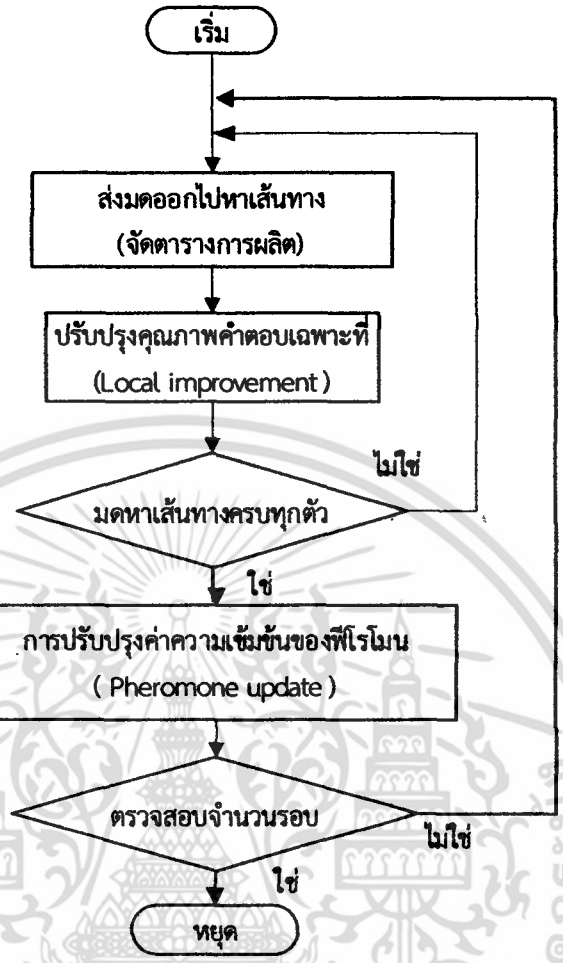
- ปัญหาการตัดและบรรจุ (Cutting and packing problem) เป็นปัญหาที่หารูปแบบในการตัดบรรจุภัณฑ์ เพื่อจะลดของเสียที่เกิดจากการตัด ลดเวลาในการตัด โดยมีเงื่อนไขในเรื่องของรูปร่างและวัสดุที่จะตัด (Burke, E. and Kendall, G. 1999 : 453)

จะเห็นได้ว่าวิธีอาณานิคมมดนั้นสามารถแก้ปัญหาต่างๆได้มากมายและหลากหลายรูปแบบ แสดงถึงประสิทธิภาพและสมรรถนะของวิธีนี้ ซึ่งเป็นการสนับสนุนแนวคิดของผู้วิจัยที่จะพัฒนาวิธีอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

2.9 ขั้นตอนของวิธีอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

การจัดตารางการผลิตแบบตามสั่งด้วยวิธีอาณานิคมมด จะมีกระบวนการที่ซับซ้อนหลายกระบวนการ แต่ละกระบวนการก็จะมีวิธีการและจุดประสงค์แตกต่างกัน ซึ่งแนวคิดของวิธีอาณานิคมมดสามารถแสดงได้ดังรูปที่ 2.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 ขั้นตอนของวิธีอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

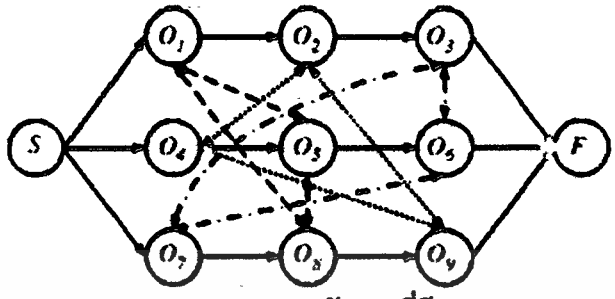
จากรูปที่ 2.12 จะเห็นได้ว่าขั้นตอนของวิธีอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งมีหลายขั้นตอน แต่ละขั้นตอนมีกระบวนการแตกต่างกัน เริ่มจากขั้นตอน การส่งมดออกไปหาเส้นทาง หลังจากนั้นเมื่อมดหาเส้นทางครบแล้ว นำเส้นทางนั้นมาคำนวณหาระยะทาง เมื่อคำนวณได้แล้ว จะปล่อยให้มดตัวถัดมาหาเส้นทาง ทำเช่นนี้จนกว่ามดทุกตัวที่กำหนดไว้จะหาเส้นทางครบทุกตัว หลังจากนั้น เลือกคำตอบที่ดีที่สุดจากมดที่หาเส้นทางทั้งหมด นำมาทำการปรับปรุงคุณภาพคำตอบ หลังจากนั้นนำคำตอบที่ได้จากการปรับปรุง ไปทำการปรับปรุงค่าความเข้มข้นของฟีโรโมน โดยทำการปรับปรุงค่า (เพิ่มค่าความเข้มข้นของฟีโรโมน) ฟีโรโมนของเส้นทางที่ดีที่สุดให้มากขึ้น หลังจากนั้นเริ่มปล่อยมดรอบใหม่ ทำเช่นนี้ไปจนกว่าครบจำนวนรอบที่กำหนด สุดท้ายจะได้เส้นทางที่เป็นคำตอบที่ดีที่สุดออกมา โดยสามารถอธิบายขั้นตอนต่างๆได้ดังนี้

2.9.1 การส่งมดออกไปหาเส้นทาง

การส่งมดออกไปหาเส้นทางในที่นี้หมายถึงการจัดตารางการผลิต โดยมดจะทำการเลือก node ของการดำเนินการ ตามเงื่อนไขของแต่ละปัญหา โดยมดจะต้องเลือกการดำเนินการทั้งหมดของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหานั้น เปรียบได้กับการที่มดเลือกเดินไปยังทุกๆแหล่งอาหาร แล้วกลับมายังรังมด เช่นตัวอย่างดังต่อไปนี้



รูปที่ 2.13 Disjunctive graph ของปัญหาที่มี 3 งานและ 3 เครื่องจักร

รูปที่ 2.13 แสดง Disjunctive graph ของตัวอย่างปัญหาการจัดตารางการผลิตแบบตามสั่ง โดยกำหนด S และ F คือการดำเนินการเริ่มต้นและสุดท้ายตามลำดับ และ O แสดงถึงการดำเนินการ ซึ่งในปัญหาข้างต้นมีจำนวนงาน 3 งาน และ 3 เครื่องจักร โดยลำดับของแต่งานเป็นไปดังแสดงในรูปดังนี้

งานที่ 1 มีลำดับคือ $O_1 \rightarrow O_2 \rightarrow O_3$
งานที่ 2 มีลำดับคือ $O_4 \rightarrow O_5 \rightarrow O_6$
งานที่ 3 มีลำดับคือ $O_7 \rightarrow O_8 \rightarrow O_9$

การเลือก node ของมดต้องอยู่ภายใต้ข้อจำกัดข้างต้นที่กล่าวมา จำนวน node ทั้งหมดที่ต้องดำเนินการเท่ากับ จำนวนงาน x จำนวนเครื่องจักร ในตัวอย่างนี้เท่ากับ 3 x 3 = 9 nodes เพราะฉะนั้นลำดับการผลิตจะมีทั้งหมด 9 การดำเนินการ และมีการเลือกเส้นทาง 9 ครั้งเช่นกัน

ตัวอย่างการหาลำดับการผลิต

มดเลือก node จากทางเลือกทั้งหมด 3 node ที่อยู่ในรายชื่อการดำเนินการที่สามารถทำต่อไป (Candidate list) ซึ่งเป็น node แรกของทุกๆ งาน (คอลัมน์ซ้ายสุด) และไม่ได้อยู่ในรายชื่อการดำเนินการที่จัดลำดับแล้ว (Visited list) ส่วน node ที่ยังไม่ได้มีการจัดลำดับจะเก็บอยู่ใน รายชื่อการดำเนินที่ยังไม่ได้จัดลำดับการผลิต (Unvisited list) ดังแสดงในตารางที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 ตัวอย่างการจัดลำดับของปัญหาที่มี 3 งานและ 3 เครื่องจักร

ครั้งที่	Candidate list	เลือก	Visited list	Unvisited list
1	O_1, O_4, O_7	O_1	O_1	$O_2, O_3, O_4, O_5, O_6, O_7, O_8, O_9$
2	O_2, O_4, O_7	O_4	O_1, O_4	$O_2, O_3, O_5, O_6, O_7, O_8, O_9$
3	O_2, O_5, O_7	O_5	O_1, O_4, O_5	$O_2, O_3, , O_6, O_7, O_8, O_9$
4	O_2, O_6, O_7	O_2	O_1, O_4, O_5, O_2	$O_3, , O_6, O_7, O_8, O_9$
5	O_3, O_6, O_7	O_6	O_1, O_4, O_5, O_2, O_6	O_3, O_7, O_8, O_9
6	O_3, O_7	O_3	$O_1, O_4, O_5, O_2, O_6, O_3$	O_7, O_8, O_9
7	O_7	O_7	$O_1, O_4, O_5, O_2, O_6, O_3, O_7$	O_8, O_9
8	O_8	O_8	$O_1, O_4, O_5, O_2, O_6, O_3, O_7, O_8$	O_9
9	O_9	O_9	$O_1, O_4, O_5, O_2, O_6, O_3, O_7, O_8, O_9$	-

เมื่อมด 1 ตัวหาเส้นทางหรือในที่นี้คือลำดับการผลิตครบทั้ง 9 nodes แล้ว สามารถนำลำดับที่ได้มาเขียนเป็นแผนภูมิแกนต์ เพื่อหาเวลาในกาทำงานทั้งหมดต่อไป

2.9.2 การปรับปรุงคุณภาพคำตอบเฉพาะที่ (Local improvement)

การปรับปรุงคุณภาพคำตอบในวิธีการหาคำตอบด้วยวิธีอาณานิคมมดเป็นการนำผลคำตอบ ซึ่งผลคำตอบของแก้ปัญหาการจัดตารางการผลิตคือ ลำดับการทำงาน มาปรับปรุงทำให้ได้ค่าเวลาในการดำเนินการทั้งหมด (Makespan) น้อยลง โดยการปรับปรุงคุณภาพคำตอบของวิธีการหาคำตอบด้วยวิธีอาณานิคมมดนั้นจะทำหลังการส่งมดออกไปหาเส้นทาง เพราะหลังจากกระบวนการนี้ จะได้ผลลัพธ์ออกมาเป็นลำดับการผลิตและค่าเวลาในการดำเนินการทั้งหมด

หลังจากการปรับปรุงค่าผลคำตอบ จะนำค่าเวลาในการดำเนินการทั้งหมดหลังการปรับปรุงเปรียบเทียบกับค่าเวลาในการดำเนินการทั้งหมดก่อนปรับปรุง ถ้าค่าเวลาในการดำเนินการทั้งหมดหลังปรับปรุงน้อยกว่าค่าเวลาในการดำเนินการทั้งหมดก่อนปรับปรุง ก็จะใช้ผลคำตอบหลังปรับปรุง แต่ถ้าค่าเวลาในการดำเนินการทั้งหมดหลังปรับปรุงมากกว่าค่าเวลาในการดำเนินการทั้งหมดก่อนปรับปรุง ก็จะใช้ผลคำตอบก่อนปรับปรุงเช่นเดิม ซึ่งการปรับปรุงคุณภาพคำตอบจะช่วยให้หาคำตอบได้เร็วขึ้น นั่นก็คือ สามารถลดจำนวนมดและจำนวนรอบให้น้อยลงได้

2.9.3 การปรับปรุงค่าความเข้มข้นของฟีโรโมน (Pheromone update)

ค่าความเข้มข้นของฟีโรโมนได้มีการกำหนดค่าเริ่มต้นให้เท่ากันทั้งหมดในทุกๆเส้นทาง โดยค่าเริ่มต้นนั้นควรกำหนดให้เหมาะสมกับ จำนวนรอบและอัตราการระเหย จากรูปที่ 2.12 การปรับปรุงค่าความเข้มข้นของฟีโรโมนจะมีการทำหลังจากมีการหาคำตอบของมดทุกๆรอบ นั่นก็คือหลังจากมดทุกตัวหาคำตอบหมดครบแล้ว ในจำนวนมดทุกตัวในรอบนั้น จะมีการเปรียบเทียบคำตอบ (ลำดับหรือตารางการผลิต) ซึ่งจะนำไปสู่การปรับปรุงค่าฟีโรโมนดังนี้

- ถ้าลำดับการทำงานนั้นเป็นลำดับการทำงานที่ดีที่สุด จะเพิ่มค่าความเข้มข้นของฟีโรโมนให้กับลำดับการทำงานนั้น (เส้นทางที่ดีที่สุด)
- ถ้าลำดับการทำงานนั้นไม่เป็นลำดับการทำงานที่ดีที่สุด จะลดค่าความเข้มข้นของฟีโรโมนให้กับลำดับการทำงานนั้น (เส้นทางอื่นๆ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปรับปรุงค่าฟีโรโมนข้างต้น สามารถเขียนเป็นสมาการดังการวิจัยของ Dorigo et. al. (1996 : 31) ดังนี้

$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij} \quad (2.3)$$

$$\Delta\tau_{ij} = \begin{cases} \frac{Q}{L} & \text{,ถ้าอยู่ในเส้นทางที่ดีที่สุด} \\ 0 & \text{,อื่นๆ} \end{cases} \quad (2.4)$$

โดย $\tau_{ij}(t+n)$ = ค่าความเข้มข้นของฟีโรโมนของการดำเนินการ i ไป j ในรอบถัดไป

$\tau_{ij}(t)$ = ค่าความเข้มข้นของฟีโรโมนของการดำเนินการ i ไป j ในรอบนั้น

ρ = ค่าอัตราการระเหยของค่าฟีโรโมน

Q = ค่าคงที่

L = ระยะทางในการเดินทางซึ่งในปัญหาการจัดตารางการผลิตคือค่าเวลาในการทำงานทั้งหมด (Makespan)

เมื่อทราบค่าลำดับการทำงานที่ดีที่สุด ขั้นตอนต่อไปคือหาค่า เส้นทางวิกฤต (Critical Path) เพื่อนำไปใช้ในการปรับปรุงค่าความเข้มขึ้นของฟีโรโมนตามเส้นทางวิกฤต หลังการปรับปรุงตามสมการที่ 2.3 และ 2.4 จะส่งผลให้ลำดับการทำงานที่ดี มีโอกาสที่จะถูกเลือกโดยมดในรอบถัดไปมากกว่าลำดับการทำงานอื่นๆ ซึ่งการปรับปรุงค่าความเข้มข้นของฟีโรโมนจะทำให้ มดสามารถหาลำดับการผลิตที่ลู่เข้าไปสู่ลำดับการผลิตที่ดีที่สุด

2.9.4 การกำหนดค่าพารามิเตอร์

ในกระบวนการหาคำตอบของวิธีอาณานิคมมด พารามิเตอร์หลายตัวที่มีความสำคัญ ตามสมการที่ 2.1, 2.2, 2.3 มีดังต่อไปนี้

- **ข้อมูลทางฮิวริสติค (η)**

ข้อมูลทางฮิวริสติคเป็นพารามิเตอร์ที่สำคัญในการเลือกเส้นทางของสมการที่ 2.1และ 2.2 คือ η ซึ่งเป็นข้อมูลทางฮิวริสติค มีการวิจัยมากมายที่แสดงถึงการได้มาของ η เช่น Dorigo et. al. (1996 : 31) ได้ทำการวิจัยเกี่ยวกับ ปัญหาการหาลำดับการเดินขายของ (Travel Salesman Problem, TSP) กำหนดค่า η เท่ากับ $\frac{1}{d}$ เมื่อ d เป็นระยะทางระหว่างเมือง ในปัญหาการจัดตารางการผลิต η สามารถกำหนดค่าจากหนึ่งส่วนเวลาดำเนินการตามที่ Heinonen and Pettersson (2007 : 992) ได้กำหนดไว้ดังนี้

$$\eta = \frac{C}{d_{O_p.O_{nt'}}} \quad (2.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย $d_{O_p,O_{nt'}}$ = ระยะทางระหว่าง node O_pไป $O_{nt'}$ (เวลาในการดำเนินการ)
C = ค่าคงที่

จากสมการที่ 2.5 จะพบว่าถ้าค่าเวลาในการดำเนินการ (Processing time) ในการดำเนินการมีค่ามาก ข้อมูลทางฮิวริสติคจะมีค่าน้อย จะทำให้เส้นทางนั้นมีความน่าจะเป็นที่จะถูกเลือกน้อย แต่ถ้าค่าเวลาในการดำเนินกาในการดำเนินการมีค่าน้อย ข้อมูลทางฮิวริสติคจะมีค่ามาก จะทำให้เส้นทางนั้นมีความน่าจะเป็นที่จะถูกเลือกมากกว่า

- **พารามิเตอร์การสำรวจ (q_0)**

ค่า q_0 กำหนดขึ้นเพื่อนำมาพิจารณากับค่า q ที่สุ่มได้ ว่าจะใช้วิธีการใดในการหาเส้นทางจากสมการที่ 2.1 ระหว่างเลือกจากค่าฟีโรโมนและอิวริสติคมากที่สุด หรือเลือกจากค่าความน่าจะเป็น ค่า q_0 ควรอยู่ระหว่าง 0 และ1 ถ้าตั้งค่า q_0 เข้าใกล้ 1 จะทำให้ระบบให้ความสำคัญกับการหาเส้นทางที่ดีที่สุด (Exploitation Activity) โดยรูปแบบของการค้นหาจะเป็นแบบวงแคบ (Local search) ในทางกลับกันถ้า ตั้งค่า q_0 เข้าใกล้ 0 จะทำให้ระบบให้ความสำคัญกับการหาเส้นทางใหม่ๆ หรือการสำรวจนั่นเอง ซึ่งจะทำให้เวลาในการคนหาเส้นทางมากขึ้น

โดยทั่วไปแล้วเราจะใช้ค่า q_0 ที่มีค่ามากก็ต่อเมื่อเป็นปัญหาขนาดเล็ก เพราะจะไม่สิ้นเปลืองเวลาและจะทำให้รูปแบบในการหาคำตอบเป็นการค้นหาแบบวงกว้าง (Global search)(Keqi Wang, Li Li. 2009 : 1185)

- **จำนวนมด**

จำนวนมดที่ใช้ในการหาเส้นทางมีผลต่อผลคำตอบและเวลาในการหาคำตอบ คือถ้าจำนวนมดมาก จะส่งผลให้ในแต่ละรอบนั้น มดสามารถสำรวจหาเส้นทางได้หลายเส้นทาง จะทำให้สามารถหาคำตอบโดยใช้จำนวนรอบน้อยลง แต่อย่างไรก็ตามจำนวนมดจะแปรผันตรงกับเวลาที่ใช้ในการหาคำตอบต่อ 1 รอบ (Iteration) คือ

ถ้าจำนวนมดมากเวลาในการหาคำตอบต่อ 1 รอบก็จะมาก ความสามารถในการหาคำตอบก็จะมีประสิทธิภาพสูง เสถียรภาพในการหาคำตอบสูง แต่ถ้าจำนวนมดมีมากเกินไป ค่าฟีโรโมนในเส้นทางต่างๆ จะมีค่าไม่ต่างกันหรือต่างกันเล็กน้อย การหาเส้นทางอาจจะเป็นการสุ่ม (Random)

ในทางกลับกันถ้าจำนวนมดน้อยค่าเวลาในการหาคำตอบต่อ 1 รอบจะน้อย ความสามารถในการหาคำตอบก็จะมีประสิทธิภาพน้อยเช่นกัน การค้นหาจะเป็นการเลือกเส้นทางเดิมๆ (Stagnation stage) (Keqi Wang, Li Li. 2009 : 1185)

การวิจัยของ Keqi Wang ในปี 2009 ได้สรุปไว้ว่า การกำหนดค่าจำนวนมด ต้องกำหนดให้เหมาะสมกับขนาดของปัญหา และสัมพันธ์กับค่าคงที่อื่นๆด้วย โดยถ้าเป็นปัญหาที่มีขนาดใหญ่ คือ ปัญหาที่มีจำนวนงานและเครื่องจักรมาก จำนวนมดที่ใช้จะมากเช่นกัน เพราะปัญหาที่มีขนาดใหญ่จะมีความหลากหลายของเส้นทาง จึงต้องเพิ่มจำนวนมดเพื่อทำให้การค้นหาครอบคลุมและมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ค่าคงที่การระเหย (ρ)**

ค่าคงที่การระเหย (ρ) เป็นค่าที่มีความสัมพันธ์กับรูปแบบการปรับปรุงค่าฟีโรโมนจากสมการที่ 2.3 สามารถปรับเปลี่ยนได้ตามรูปแบบสมการปรับปรุงค่าฟีโรโมน โดยค่าคงที่การระเหยมีค่าระหว่าง 0 ถึง 1 ถ้าค่าคงที่การระเหยมีค่ามาก การเดินทางของมดจะมีแนวโน้มไปเส้นทางเดิมๆ แต่ถ้าค่าคงที่การระเหยมีค่าน้อย การเดินทางของมดจะมีแนวโน้มหาเส้นทางใหม่ๆมากขึ้น

- **ค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β)**

กฎการเปลี่ยนสถานะ (State transition rule) ในสมการที่ 2.1และ 2.2 พารามิเตอร์สองตัวที่มีความสำคัญมากคือ ค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) จะบอกความสัมพันธ์เป็นสัดส่วนระหว่างความเข้มข้นของฟีโรโมนกับค่าข้อมูลทางฮิวริสติค (η) Dorigo et.al. (1996 : 32) ได้แสดงค่าความสัมพันธ์ของ α และ β ของปัญหาการเดินทางของพนักงานขาย (Travelling salesman problem, TSP) ดังตารางที่ 2.3

ตารางที่ 2.3 แสดงค่าความสัมพันธ์ของค่า α และ β ที่จะให้คำตอบที่ดีที่สุดของปัญหาการเดินทางของพนักงานขาย

α	β
0.5	5.0
1.0	1.0
1.0	2.0
1.0	5.0

ค่า α และ β ตามตารางที่ 2.3 เป็นค่าที่เหมาะสมในการใช้งาน ถ้าวิเคราะห์เรื่องความสัมพันธ์ผลก็คือ ถ้า α มีค่ามากมดจะเลือกเส้นทางจากความเข้มข้นของฟีโรโมนเป็นหลัก จะทำให้ประสิทธิภาพในการค้นหาแบบวงกว้าง (Global search) ลดลง แต่ถ้า β มีค่ามากมดจะเลือกเส้นทางที่ระยะทางสั้น (Local area) แต่ก็จะทำให้ประสิทธิภาพในการค้นหาแบบวงกว้างลดลงเช่นกัน จะทำให้ระบบค้นหาเป็นการค้นหาคำตอบในวงแคบ (Local search)

ถ้าทั้ง α และ β มีค่ามาก forward feedback จะส่งผลถึงคำตอบ ซึ่งถ้าปัญหามีขนาดใหญ่ ผลคำตอบอาจจะเป็นคำตอบที่ดีที่สุดในวงแคบ (local optimum)

ถ้าทั้ง α และ β มีค่าน้อย มดจะไม่สนใจความสำคัญของข้อมูลใดๆ การหาเส้นทางก็จะเป็นการสุ่ม (Keqi Wang. 2009 : 1185-1186)

ถ้า $\alpha = 0$ การค้นหาเส้นทางจะไม่นำความเข้มข้นของฟีโรโมนมาพิจารณา ทำให้จะเลือกการดำเนินการที่ใกล้หรือเวลาดำเนินการน้อย ในทางกลับกัน ถ้า $\beta = 0$ การค้นหาจะพิจารณาเฉพาะความเข้มข้นของฟีโรโมนเท่านั้น ซึ่งโดยทั่วไปแล้วจะนำไปสู่สถานการณ์เมื่อยล้า (Stagnation situation) เพราะมดทุกตัวจะเดินตามเส้นทางเดิมๆ (Heinonen. and Pettersson. 2007 : 992)

จะเห็นได้ว่าค่า α และ β นั้นมีความสำคัญมากในการหาการดำเนินการถัดไปในกระบวนการเปลี่ยนสถานะ ในการวิจัยที่ผ่านมาการหาค่า α และ β จะใช้วิธีลองผิดลองถูกและทดลองหาคำตอบออกมา ซึ่งต้องลองผิดลองถูกไปจนกว่าจะได้คำตอบเป็นที่น่าพอใจ ซึ่งการหาค่า α และ β ด้วยวิธีการนี้ต้องใช้เวลานาน ยิ่งไปกว่านั้นค่า α และ β ที่ได้อาจจะไม่ใช่ค่าที่เหมาะสมที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นในการวิจัยนี้จะทำการหาค่า α และ β ที่เหมาะสมกับแต่ละรูปแบบของปัญหา เพื่อเพิ่มประสิทธิภาพในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีการดำเนินการวิจัย

การดำเนินการวิจัยสำหรับแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง โดยใช้วิธีขั้นตอนอาณานิคมมดเป็นแนวคิดในการแก้ปัญหา มีขั้นตอนการดำเนินการคือ

1. การศึกษาและวิเคราะห์ชุดปัญหาของการผลิตแบบตามสั่ง
2. การปรับปรุงและพัฒนาวิธีที่ใช้ในการแก้ปัญหา
3. การพัฒนาโปรแกรมคอมพิวเตอร์ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

3.1 การศึกษาและวิเคราะห์ชุดปัญหาของการผลิตแบบตามสั่ง

งานวิจัยนี้มีจุดประสงค์เพื่อเปรียบเทียบประสิทธิภาพในการหาคำตอบ เพราะฉะนั้นปัญหาที่จะนำมาทดสอบกับโปรแกรมคอมพิวเตอร์ที่จัดทำขึ้นต้องเป็นปัญหาที่มีการยอมรับในวงกว้างและต้องเป็นปัญหาที่ใกล้เคียงกับระบบการผลิตจริงในปัจจุบัน ผู้วิจัยจึงเลือกชุดปัญหามาตรฐานของ Lawrence เพื่อใช้ในการทดลองและเปรียบเทียบผลคำตอบระหว่างวิธีการแก้ปัญหาที่จัดทำขึ้นกับวิธีการแก้ปัญหาอื่นๆ โดยชุดปัญหานี้นำมาจาก OR-Library ซึ่งจัดเก็บในรูปแบบของตาราง โดยแต่ละแถวแทนรายละเอียดของงานแต่ละงาน และแต่ละคอลัมน์แทนเครื่องจักรและเวลาในการดำเนินการ ตัวอย่างปัญหาชุดปัญหามาตรฐานของ Lawrence แสดงดังตารางที่ 3.1

ตารางที่ 3.1 แสดงข้อมูลที่ได้จากตัวอย่างปัญหา La01

	ลำดับ 1		ลำดับ 2		ลำดับ 3		ลำดับ 4		ลำดับ 5	
	เครื่องจักร	เวลาดำเนินการ	เครื่องจักร	เวลาดำเนินการ	เครื่องจักร	เวลาดำเนินการ	เครื่องจักร	เวลาดำเนินการ	เครื่องจักร	เวลาดำเนินการ
งาน 1	1	21	0	53	4	95	3	55	2	34
งาน 2	0	21	3	52	4	16	2	26	1	71
งาน 3	3	39	4	98	1	42	2	31	0	12
งาน 4	1	77	0	55	4	79	2	66	3	77
งาน 5	0	83	3	34	2	64	1	19	4	37
งาน 6	1	54	2	43	4	79	0	92	3	62
งาน 7	3	69	4	77	1	87	2	87	0	93
งาน 8	2	38	0	60	1	41	3	24	4	83
งาน 9	3	17	1	49	4	25	0	44	2	98
งาน 10	4	77	3	79	2	43	1	75	0	96

หมายเหตุ ลำดับตัวเลขเครื่องจักรของชุดปัญหามาตรฐานที่นำมาจาก OR-Library จะเริ่มจาก 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 วิเคราะห์ลักษณะชุดปัญหามาตรฐาน Lawrence จาก OR-Library

จากตารางที่ 3.1 สามารถสรุปลักษณะรูปแบบของชุดปัญหามาตรฐานที่ได้มาจาก OR-Library ได้ว่าจำนวนแถวจะแสดงถึงจำนวนงาน ซึ่งแต่ละแถวมีการแสดงถึงลำดับการดำเนินการและเวลาในการผลิตของแต่ละการดำเนินการดังนี้

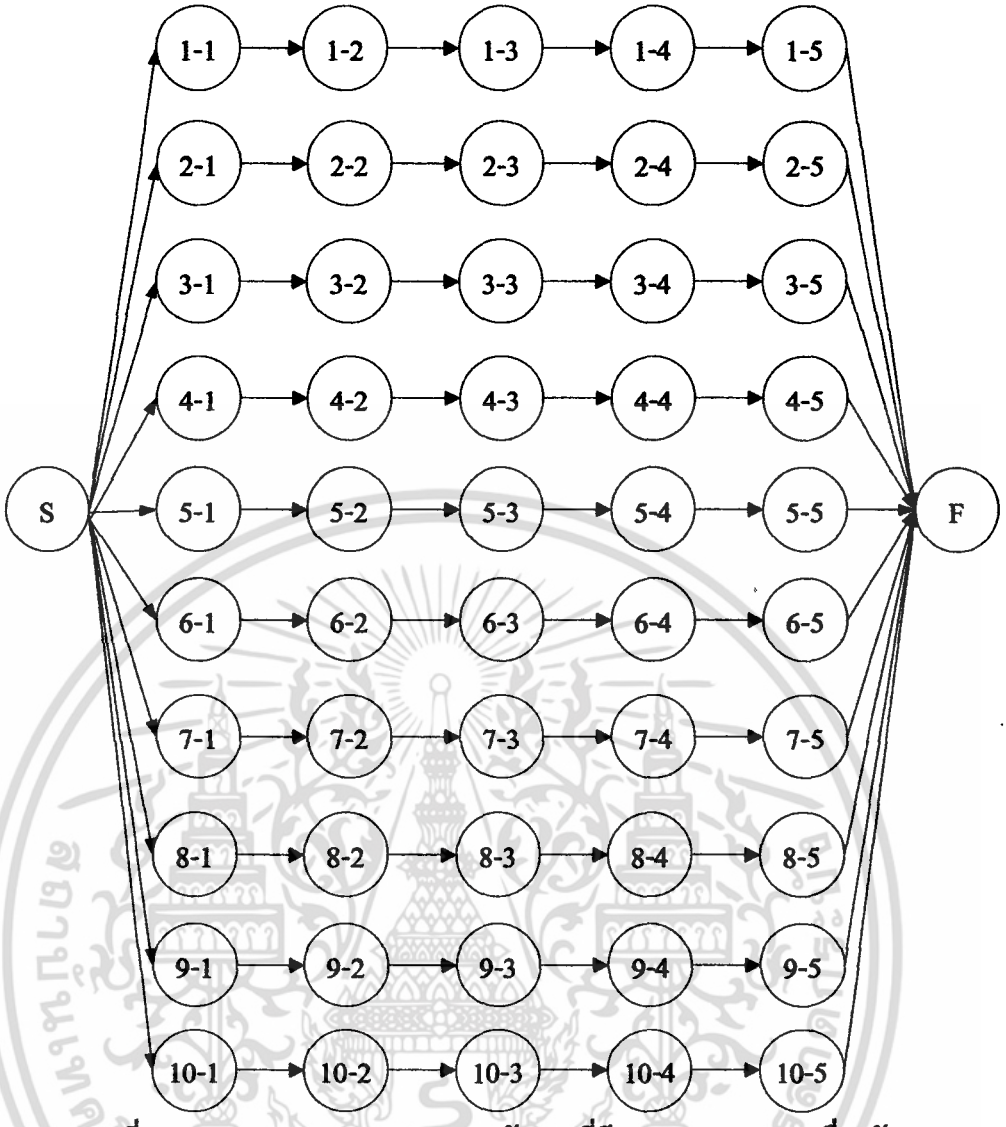
จำนวนงาน	=	10	งาน
จำนวนเครื่องจักร	=	5	เครื่องจักร
จำนวนการดำเนินการ 10 x 5	=	50	การดำเนินการ

แต่ละงานมี 5 การดำเนินการ โดยมีลำดับการดำเนินการดังเช่น งานที่ 1 คือ

การดำเนินการที่ 1	ทำบนเครื่องจักร 1	เวลาในการดำเนินการ 21 หน่วยเวลา
การดำเนินการที่ 2	ทำบนเครื่องจักร 0	เวลาในการดำเนินการ 53 หน่วยเวลา
การดำเนินการที่ 3	ทำบนเครื่องจักร 4	เวลาในการดำเนินการ 95 หน่วยเวลา
การดำเนินการที่ 4	ทำบนเครื่องจักร 3	เวลาในการดำเนินการ 55 หน่วยเวลา
การดำเนินการที่ 5	ทำบนเครื่องจักร 2	เวลาในการดำเนินการ 34 หน่วยเวลา

ข้อมูลการผลิตในงานอื่นๆ สามารถตีความหมายได้ดังตัวอย่างข้างต้น และสามารถนำมาเขียนเป็น Disjunctive graph ได้ดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 Disjunctive graph ของปัญหาที่มี 10 งานและ 5 เครื่องจักร

จากรูปที่ 3.1 การดำเนินการแต่ละการดำเนินการแสดงด้วย สัญลักษณ์วงกลม และมีตัวเลขการดำเนินการอยู่ภายในวงกลม เลขตัวแรกแสดงถึง งาน และตัวที่สองแสดงถึง ลำดับการดำเนินการของงานนั้น และเพื่อการหาคำตอบผู้วิจัยจึงเพิ่มการดำเนินการ (Dummy operation) เริ่มต้น S (Start) และการดำเนินการสุดท้าย F (Final) เข้าไปเพื่อแสดงถึงทิศทางของลำดับการทำงาน ซึ่งสามารถดูจากหัวลูกศรที่ชี้ไปยังการดำเนินการถัดไป

3.1.2 การกำหนดปัญหาโดยใช้แบบจำลองทางคณิตศาสตร์

การแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งนั้น จะต้องคำนึงถึงข้อจำกัดและเงื่อนไขต่างๆ ของปัญหา จากการวิเคราะห์ปัญหาการจัดตารางการผลิตแบบตามสั่ง ที่ได้อธิบายไปแล้วในหัวข้อ 3.1 สามารถนำปัญหาการจัดตารางการผลิตแบบตามสั่งมาเขียนให้อยู่ในรูปของตัวแบบทางคณิตศาสตร์ (Mathematical Model) ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Minimize	C_{max}		
s.t.	$C_{max} - S_{ij} \geq P_{ij}$	$\forall(i,j)$	(3.1)
	$S_{kj} - S_{ij} \geq P_{ij}$	$\forall(i,j) \rightarrow (k,j)$	(3.2)
	$S_{ij} - S_{il} \geq P_{ij}$ หรือ $S_{kj} - S_{ij} \geq P_{ij}$	$\forall(i,l),(i,j)$	(3.3)
	$P_{ij} > 0$, $S_{ij} > 0$	$\forall(i,j)$	(3.4)

เมื่อ C_{max} = เวลาในการดำเนินการทั้งหมดหรือเวลาที่การดำเนินการสุดท้ายเสร็จสิ้น (Makespan)

S_{ij} = เวลาเริ่มการดำเนินการ (Starting time of operation) งาน j ที่ทำบนเครื่องจักร i

P_{ij} = เวลาในการดำเนินการ (Processing time of operation) งาน j ที่ทำบนเครื่องจักร i

i,k = เครื่องจักรใดๆ

j,l = ลำดับงานใดๆ

สมการตัวแบบทางคณิตศาสตร์ดังกล่าวเป็นการปรับปรุงมากจากการวิจัยของ Ramezanali (2007 : 80-81) ซึ่งสามารถอธิบายได้ว่า จุดประสงค์ของการแก้ปัญหาการจัดตารางการผลิตคือ เพื่อหาค่าเวลาในการผลิตทั้งหมดที่น้อยที่สุด (Minimize makespan) โดยใช้สมการที่ 3.1 ถึง 3.5 ในการกำหนดเงื่อนไขดังนี้

เงื่อนไขในสมการที่ 3.1 เป็นเงื่อนไขของค่า makespan ต้องมีค่ามากกว่าเวลาเริ่มต้นของการดำเนินการใดๆ มากกว่าหรือเท่ากับเวลาที่ใช้ในการดำเนินการนั้นๆ นั่นคือ Makspan ได้มาจากเวลาในการดำเนินการทั้งหมด ไม่มีการดำเนินการใดจะใช้เวลามากกว่า Makspan

เงื่อนไขในสมการที่ 3.2 เป็นเงื่อนไขของการดำเนินการที่อยู่ในงานเดียวกัน การดำเนินการถัดไปต้องรอจนกว่าการดำเนินการก่อนหน้าที่อยู่ในงานเดียวกันทำเสร็จก่อน

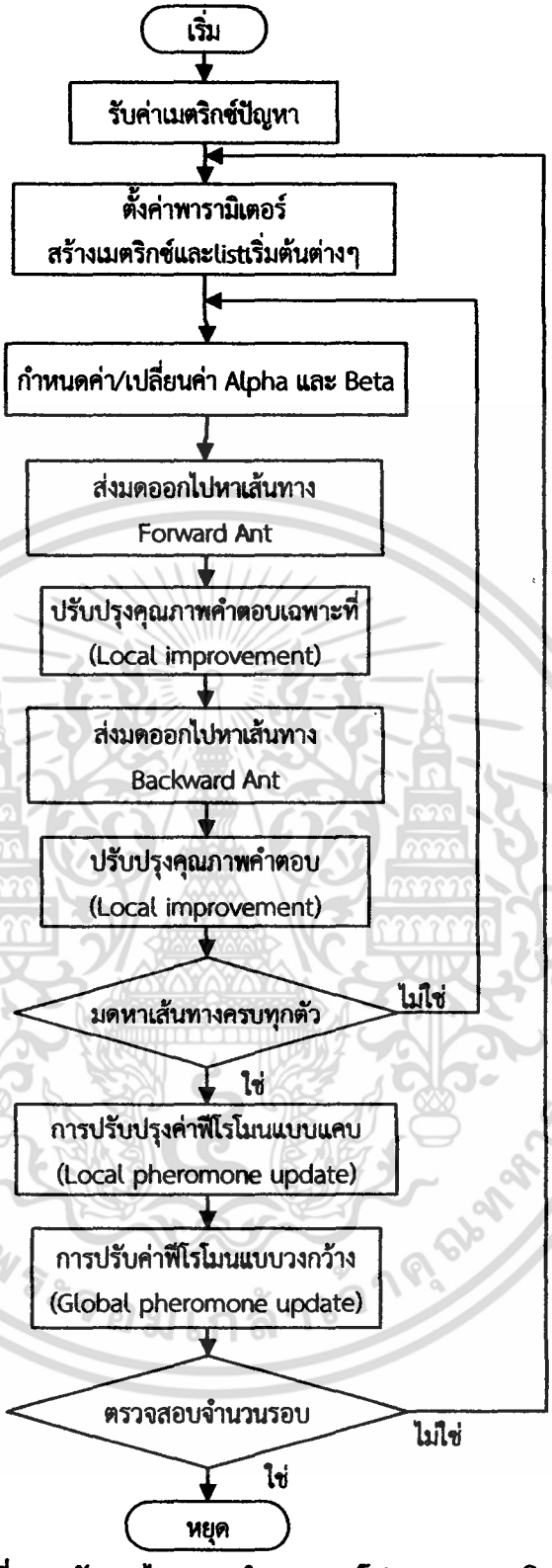
เงื่อนไขในสมการที่ 3.3 เป็นเงื่อนไขของการดำเนินการที่ทำบนเครื่องจักรเดียวกัน ต้องรอจนกว่าการดำเนินการก่อนหน้าทำเสร็จก่อน

เงื่อนไขในสมการที่ 3.4 เป็นเงื่อนไข ทุกการดำเนินการต้องมีค่าเวลาในการดำเนินการ และเวลาในการเริ่มการดำเนินการต้องมีค่ามากกว่าศูนย์

3.2 การปรับปรุงและพัฒนาวิธีอาณานิคมมดในการแก้ปัญหาจัดตารางการผลิตแบบตามสั่ง

ในขั้นตอนนี้เป็นปรับปรุงวิธีการแก้ปัญหาขั้นตอนวิธีอาณานิคมมด เพื่อให้ได้คำตอบที่ดีขึ้น โดยการพัฒนาและปรับปรุงนั้นจะมุ่งเน้นไปในด้านการหาค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่เหมาะสมในแต่ละปัญหา และพัฒนาให้โปรแกรมคอมพิวเตอร์มีการปรับเปลี่ยนค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) อย่างอัตโนมัติ โดยมีขั้นตอนดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 ผังการไหลการทำงานของโปรแกรมคอมพิวเตอร์

ขั้นตอนการทำงานของวิธีการจัดตารางการผลิตแบบตามสั่งด้วยขั้นตอนวิธีอาณานิคมมด (Ant colony optimization) **สามารถเขียนอยู่ในรูปแบบของ** pseudo code **ดังนี้**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Algorithm JSP-ACO
กำหนดค่าเริ่มต้นของฟีโรโมน และค่าเริ่มต้นต่างๆ
repeat
    for มด $k \in \{1, \ldots, m\}$ [สร้างคำตอบที่เป็นไปได้]
        $S := \{1, \ldots, n\}$ [เซตของการดำเนินการที่ยังไม่ได้เลือก]
        เลือกการดำเนินการ i ด้วยความน่าจะเป็น p_{0i}
        repeat
            เลือกการดำเนินการ j ด้วยความน่าจะเป็น p_{ij}
            $S := S - \{j\}$
            $i := j$
        until $S = \emptyset$
    endfor
    การปรับปรุงคุณภาพคำตอบ (Local improvement)
    การปรับค่าฟีโรโมน (Pheromone update)
        forall i, j do
            ลดฟีโรโมน
        endfor
        forall i, j ในขั้นตอนวนซ้ำที่ได้คำตอบที่ดีที่สุด (Critical path) do
            เพิ่มฟีโรโมน
        endfor
until บรรลุเกณฑ์สิ้นสุดการทำงาน

โปรแกรมคอมพิวเตอร์จัดตารางการผลิตประกอบไปด้วยการรับค่าเมตริกซ์ปัญหา หลังจากนั้นทำการสร้างเมตริกซ์และ List ต่างๆ เพื่อเป็นการจองหน่วยความจำ กระบวนการต่อไปคือการการจัดตารางการผลิตของมดทีละตัว หลังจากมดทุกตัวจัดตารางการผลิตเสร็จแล้ว ทำการปรับปรุงค่าฟีโรโมนแบบแคบและแบบวงกว้าง หลังจากนั้นจะเริ่มการหาเส้นทางในรอบใหม่ (Next iteration) ซึ่งจะมีการปรับเปลี่ยนค่า α และ β จากค่า α และ β ที่ให้ผลตอบที่ดีที่สุดในรอบก่อนหน้า
โดยสามารถอธิบายขั้นตอนต่างๆ อย่างละเอียดดังนี้

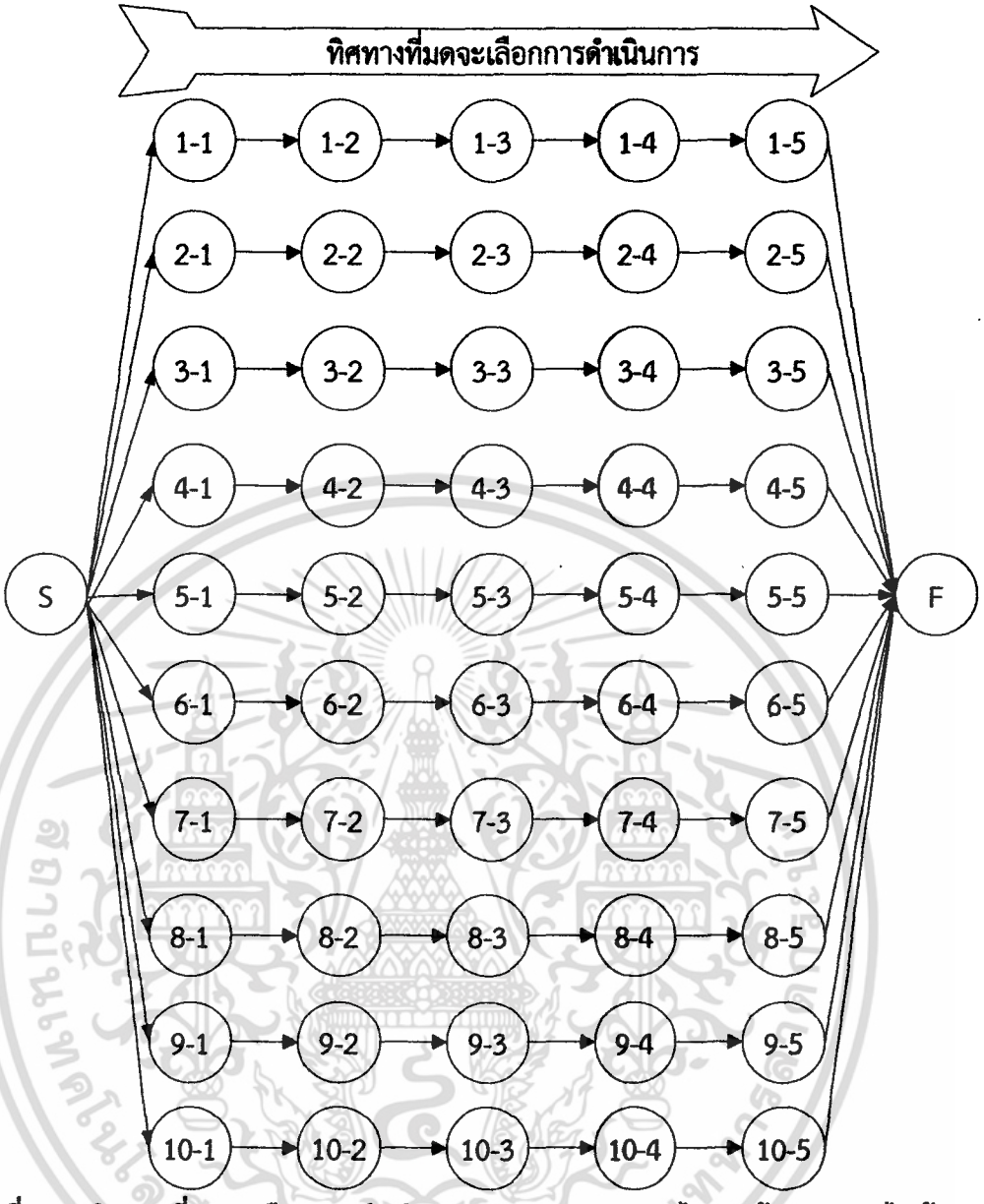
3.2.1 การจัดลำดับการผลิต

การจัดตารางการผลิตหรือจัดลำดับการผลิต ในวิธีอาณานิคมมดนั้นก็คือการส่งมดออกเดินหาเส้นทางนั่นเอง ซึ่งในวิทยานิพนธ์ฉบับนี้มีการส่งมดออกหาเส้นทาง 2 รูปแบบ คือ

- **การหาเส้นทางแบบไปข้างหน้า (Forward)**

การส่งมดออกไปหาเส้นทางแบบไปข้างหน้า (Forward) เป็นการให้มดเลือกการดำเนินการ (Node) จาก S ไป F ดังทิศทางที่มดจะเลือกการดำเนินการที่แสดงในรูปที่ 3.3 โดย S และ F เป็นการดำเนินการที่กำหนดขึ้นเพื่อใช้ในการหาคำตอบ (Dummy node)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

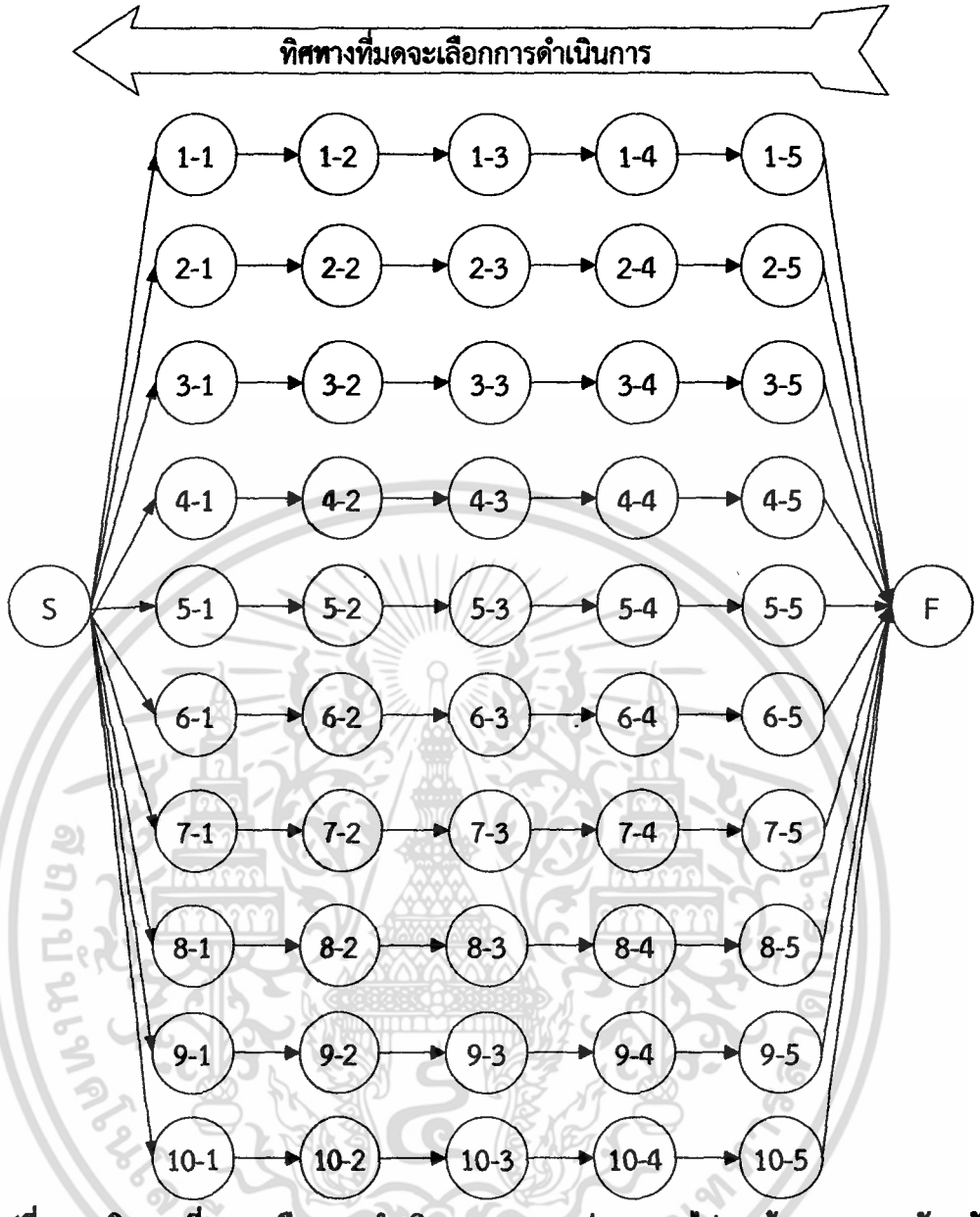


รูปที่ 3.3 ทิศทางที่มดจะเลือกการดำเนินการของการส่งมดออกไปหาเส้นทางแบบไปข้างหน้า

- **ส่งมดออกไปหาเส้นทางแบบกลับหลัง (Backward)**

การส่งมดออกไปหาเส้นทางแบบกลับหลัง (Backward) เป็นการให้มดเลือกการดำเนินการ (Node)) จาก F ไป S ดังทิศทางที่มดจะเลือกการดำเนินการที่แสดงในรูปที่ 3.4 ซึ่งการส่งส่งมดออกไปหาเส้นทางในรูปแบบนี้ เมื่อได้ลำดับการผลิตทั้งหมดแล้ว จะต้องนำมาเรียงลำดับใหม่ ให้เป็นไปตามลำดับจาก S ไปยัง F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ทิศทางที่มดจะเลือกการดำเนินการของการส่งมดออกไปหาเส้นทางแบบกลับหลัง

การมีรูปแบบการส่งมดออกหาเส้นทาง 2 รูปแบบนั้น ก็เพื่อทำให้การแก้ปัญหา ให้ผลคำตอบที่ลู่เข้าหาคำตอบที่ดีเร็วขึ้น และใช้ผลของการปรับปรุงค่าความเข้มข้นของฟีโรโมนอย่างมีประสิทธิภาพมากที่สุด

การเลือกเส้นทางของมดนั้น มดตัดสินใจในการเลือกการดำเนินการจากสมการ 2.1 และ 2.2 โดยในการเลือกการดำเนินการแรกนั้นจะเลือกจากความน่าจะเป็นของการดำเนินการทั้งหมดที่ทำได้ (สมการ 2.2) ส่วนในการเลือกการดำเนินการอื่นๆ จะใช้กฎการเปลี่ยนสถานะ (State transition rule, สมการ 2.1 และสมการ 2.2) เพื่อจะให้เกิดความหลากหลายในการหาเส้นทาง

การหาเส้นทางของมดแต่ละตัวในแต่ละรอบนั้น จะมีฟังก์ชันจุดประสงค์ (Objective Function) เพื่อบอกว่าเส้นทางที่ดีที่สุดเป็นของมดตัวไหนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$opt = min(T) = min\{max[T_1, T_2, T_3, \dots, T_m]\} \quad (3.5)$$

โดย T_i, = เวลาที่ใช้ในการดำเนินการของเส้นทางที่ i

i = 1,2,3 ... m เมื่อ m คือจำนวนมด

เมื่อได้ค่าฟังก์ชันจุดประสงค์มาแล้วก็นำมาหาค่า Fitness เพื่อใช้การเปรียบเทียบคำตอบในแต่ละรอบของการหาเส้นทาง ซึ่ง fitness function จะปรับค่าจากฟังก์ชันจุดประสงค์มาให้อยู่ในช่วง [0,100] โดยมีสมการดังนี้

$$f_i = 100 \times \frac{opt}{T_{Ant}} \quad (3.6)$$

โดย opt = คำตอบที่ดีที่สุดในขณะนั้น

T_{Ant} = คำตอบจากมดใดๆ

จากสมการ 3.6 จะเห็นได้ว่าเมื่อเส้นทางใดมีค่าเวลาในการดำเนินการทั้งหมด (Makespan) น้อยที่สุดหรือเป็นค่าที่ดีที่สุดของรอบนั้น จะมีค่า fitness function เท่ากับ 100 โดยเส้นทางอื่นๆที่หาจากมดตัวอื่นๆ จะมีค่าน้อยกว่าร้อย การปรับค่าผลคำตอบจากในการดำเนินการทั้งหมดมาเป็น fitness function จะทำให้การเปรียบเทียบมีประสิทธิภาพมากขึ้นเพราะค่าของตัวเลขมีขอบเขตเดียวกัน

3.2.2 การกำหนดค่าข้อมูลทางฮิวริสติค (η)

ในการวิจัยนี้จะใช้ข้อมูลทางฮิวริสติค (η) ของ Zhong (2008 : 1016) โดยกำหนดข้อมูลทางฮิวริสติคให้ซับซ้อนขึ้น โดยคำนึงถึงความสมดุลของภาระงาน (Balance of Workload) บนเครื่องจักรแต่ละเครื่อง ให้ค่า η เกิดจากการถ่วงน้ำหนักสองส่วน ในส่วนแรกเป็นข้อมูลทาง ฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่มีเวลาในการดำเนินการ (Processing time) น้อย ส่วนที่สองเป็นข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่ทำอยู่บนเครื่องจักรที่มีภาระงานในเวลานั้นน้อยกว่าเครื่องจักรอื่น เพื่อเป็นการรักษาสมดุลของภาระงานในทุกเครื่องจักร

$$\eta = \eta_H \times \eta_{workload} \quad (3.7)$$

$$\eta_{workload} = \left(\frac{t_{max} - t_{next'}}{t_{max} - t_{min}}\right) x_1 + x_2 \quad (3.8)$$

โดย η_H = ข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่มีเวลาในการดำเนินการ (Processing time) น้อยมีค่าเท่ากับ $\frac{C}{d_{O_p, O_{nt'}}}$

$\eta_{workload}$ = ข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่ทำอยู่บนเครื่องจักรที่มีภาระงานในเวลานั้นน้อยกว่าเครื่องจักรอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

x_1, x_2 = เป็นค่าคงที่สัดส่วนของน้ำหนักของค่าภาระงานระหว่างข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่มีเวลาในการดำเนินการและข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่ทำอยู่บนเครื่องจักรที่มีภาระงานในเวลานั้นน้อยกว่าเครื่องจักรอื่น โดย $x_1 + x_2 = 1$

3.2.3 การปรับค่าฟีโรโมน (Pheromone update)

ในการวิจัยนี้ใช้การปรับค่าฟีโรโมน (Pheromone update) ใน 2 กรณีคือการปรับปรุงค่าฟีโรโมนแบบแคบ (Local pheromone update) และอีกกรณีหนึ่งคือการปรับค่าแบบฟีโรโมนแบบวงกว้าง (Global pheromone update) โดยจะปรับค่าฟีโรโมนใหม่ เมื่อครบ 1 รอบ (มดทุกตัวหาเส้นทางหมดแล้ว)

การปรับปรุงค่าฟีโรโมนแบบแคบ

$$\tau_p(O_p, O_{nt'}) = \begin{cases} \tau_{old}(O_p, O_{nt'}) + \frac{y}{C_{max}} & \text{, ถ้า } O_p, O_{nt'} \epsilon\ tour\ S_{tl} \\ \tau_{old}(O_p, O_{nt'}) - \frac{y}{C_{max}} & \text{, อื่นๆ} \end{cases} \quad (3.9)$$

การปรับค่าแบบฟีโรโมนแบบวงกว้าง

$$\tau_p(O_p, O_{nt'}) = \begin{cases} \tau_{old}(O_p, O_{nt'}) + z & \text{, ถ้า } O_p, O_{nt'} \epsilon \text{ tour } S_{tg} \\ \tau_{old}(O_p, O_{nt'}) \times (1-\rho) & \text{, อื่นๆ} \end{cases} \quad (3.10)$$

โดย S_{tl} = ลำดับของการดำเนินการที่ให้ค่าเวลาในการดำเนินการทั้งหมด (Makespan) น้อยที่สุดของรอบนั้นๆ (เส้นทางวิกฤตใช้ในการระบุเส้นทางที่จะปรับปรุงค่าฟีโรโมน)

S_{tg} = ลำดับของการดำเนินการที่ให้ค่าเวลาในการดำเนินการทั้งหมดน้อยสุดที่หาได้ในขณะนั้น

τ_{old} = ค่าความเข้มข้นของฟีโรโมนก่อนการปรับปรุงค่า

ρ = ค่าคงที่การระเหย ($\rho \in [0,1)$)

y, z = ค่าคงที่

C_{max} = ค่าเวลาในการดำเนินการทั้งหมด (Makespan)

กำหนดค่าความเข้มข้นของฟีโรโมนที่น้อยที่สุดและมากที่สุด (τ_{min}, τ_{max}) ซึ่งถ้าค่าระดับของฟีโรโมนต่ำกว่าค่าต่ำสุดหรือสูงกว่าค่าสูงสุดที่ตั้งไว้ จะปรับค่าให้เท่ากับค่าต่ำที่สุดและสูงที่สุดที่ตั้งไว้ตามลำดับ Keqi Wang. (2009 : 1185)

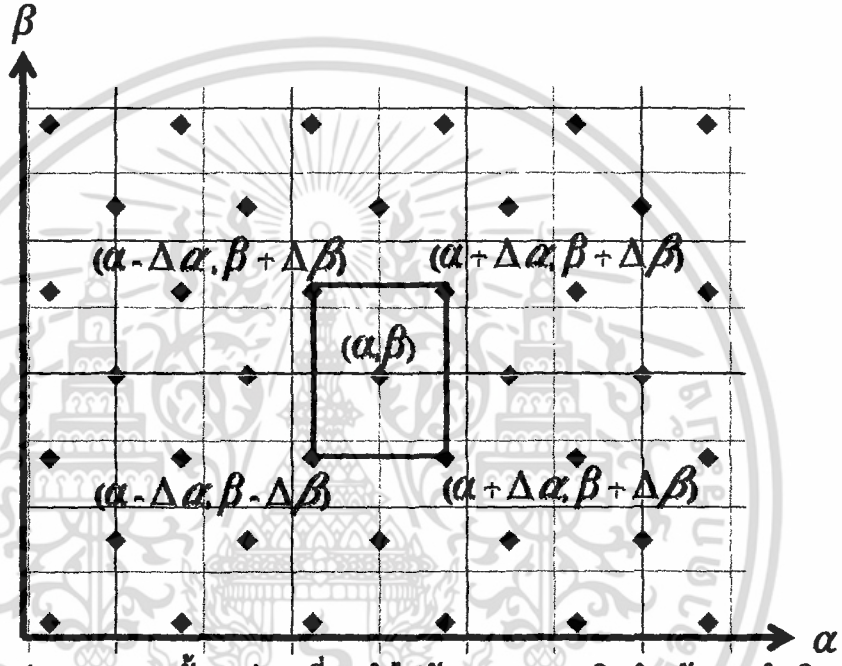
3.2.4 การกำหนดค่าและการเปลี่ยนค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β)

ในขั้นตอนนี้เป็นการปรับปรุงวิธีการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งด้วยวิธีอาณานิคมมด ซึ่งปรับปรุงให้โปรแกรมคอมพิวเตอร์สามารถเปลี่ยนค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ได้อย่างอัตโนมัติ ทำให้ผลการหาคำตอบมีประสิทธิภาพมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยก่อนที่โปรแกรมคอมพิวเตอร์จะสามารถเปลี่ยนค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ได้อย่างอัตโนมัติได้นั้น ผู้วิจัยจะต้องทำการทดลองหาแนวโน้มของค่าทั้งสองที่จะทำให้ได้คำตอบที่ดีที่สุดเสียก่อน หลังจากนั้นจะนำแนวโน้มที่ได้มาวิเคราะห์เป็นตัวเลขเริ่มต้นในการกำหนดค่าดังนี้

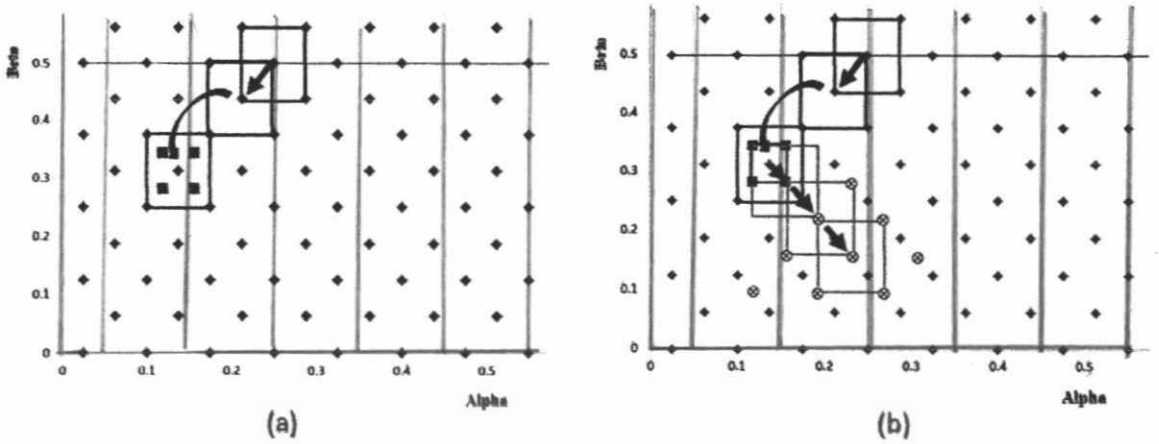
1. นำค่า α และ β ที่เหมาะสมที่สุดจากการทดลองหาแนวโน้มมาเป็นจุดเริ่มต้น
2. จากจุดเริ่มต้นดังกล่าว ทำการเพิ่มและลดค่า α และ β ด้วยค่าคงที่ความต่างค่าหนึ่ง ($\Delta\alpha$ และ $\Delta\beta$) 4 ทิศทางดังรูปที่ 3.5 จะได้ค่า α และ β รวมทั้งสิ้น 5 คู่เพื่อทำการทดลอง
3. ทำการทดลองหาคำตอบ เมื่อทำการทดลองรอบใหม่ ค่า α และ β คู่ใดให้คำตอบดีที่สุด นำจุดนั้นเป็นจุดกึ่งกลางดังรูปที่ 3.6 a แล้วทำตามขั้นตอนที่ 2 จนหมดรอบของการหาคำตอบ



รูปที่ 3.5 แสดงค่า α และ β ทั้ง 5 คู่แรกที่มดนำไปจัดตารางการผลิตสำหรับการดำเนินการรอบแรก

- ถ้าค่า α และ β ที่ดีที่สุดนั้นเป็นค่าที่อยู่จุดกึ่งกลาง ในรอบถัดไปจะใช้จุดเดิมในการทดลอง แต่ลดค่า $\Delta\alpha$ และ $\Delta\beta$ หลังจากนั้นจะเพิ่มค่า $\Delta\alpha$ และ $\Delta\beta$ ดังเดิม ดังรูปที่ 3.6 a และ 3.6 b
- ถ้าผลคำตอบในรอบนั้นที่ได้มีค่าดีกว่าในรอบก่อนหน้าจนถึงเกณฑ์กำหนดไว้ จะทำการปรับเปลี่ยนค่าแบบก้าวกระโดด เพื่อทำให้ลดเวลาในการหาคำตอบ ดังรูปที่ 3.6 a

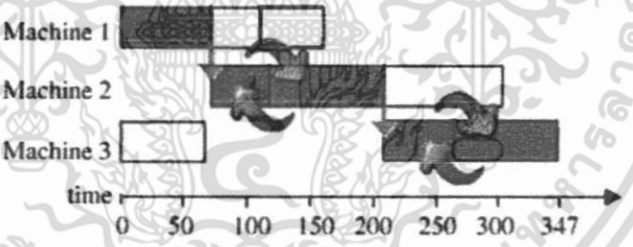
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 (a) แสดงการปรับค่าแบบก้าวกระโดดและการลดค่า $\Delta\alpha$ และ $\Delta\beta$ เมื่อได้ค่า α และ β ใหม่เท่ากับค่าเดิม, (b) แสดงกระบวนการในการปรับค่า α และ β หลังการลดค่า $\Delta\alpha$ และ $\Delta\beta$

3.3.5 การปรับปรุงคุณภาพคำตอบเฉพาะที่ (Local improvement)

หลังจากได้คำตอบจากการวนครบ 1 รอบแล้ว จะทำการปรับปรุงคุณภาพของตารางการผลิต โดยในการวิจัยนี้นำวิธีการปรับปรุงคุณภาพคำตอบของ Nowicki and Smutinicki (1996 : 799) มาประยุกต์ใช้งาน หลักการปรับปรุงคือ ถ้าหลังจากการปรับปรุงแล้วได้คำตอบที่ดีขึ้น ก็จะใช้คำตอบนั้น แต่ถ้าคำตอบที่ปรับปรุงไม่ดีขึ้น ก็จะใช้คำตอบเดิมก่อนปรับปรุง โดยวิธีการปรับปรุงจะเป็นการสลับกับ block ข้างเคียง (Neighborhood solution) ดังรูปที่ 3.7



รูปที่ 3.7 แสดงวิธีการปรับปรุงคุณภาพคำตอบของ E.Nowicki

จากรูปที่ 3.7 สามารถอธิบายขั้นตอนการปรับปรุงคุณภาพคำตอบเฉพาะที่คือ

- วิเคราะห์ เส้นทางวิกฤต โดยแบ่งเป็น block ตามเครื่องจักร
- ทำการปรับปรุงเส้นทางวิกฤตดังนี้

ใน block ที่ 1: สลับสองการดำเนินการสุดท้าย

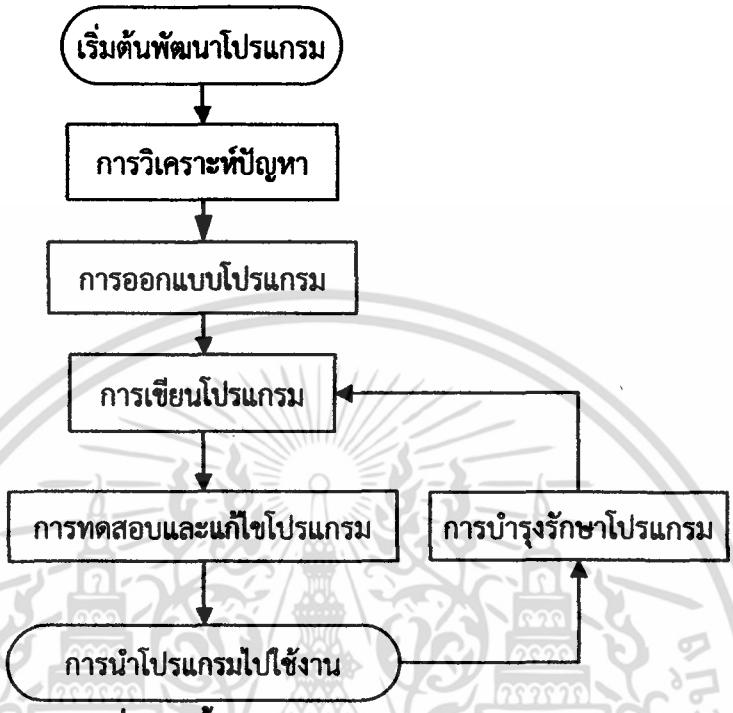
ใน block อื่นๆ : สลับสองการดำเนินการแรกและสุดท้าย เฉพาะ block ที่มีมากกว่า 2 การดำเนินการ

ใน block สุดท้าย : สลับสองการดำเนินการแรกแต่ที่สำคัญ การสลับกันของการดำเนินการ ต้องสอดคล้องกับลำดับการผลิตด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การพัฒนาโปรแกรมคอมพิวเตอร์ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

โปรแกรมคอมพิวเตอร์เป็นเครื่องมือในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง โดยมีขั้นตอนการพัฒนาโปรแกรมคอมพิวเตอร์ที่สำคัญ ดังรูปที่ 3.8



รูปที่ 3.8 ขั้นตอนการพัฒนาโปรแกรมคอมพิวเตอร์

ขั้นตอนการพัฒนาโปรแกรมคอมพิวเตอร์ทั้ง 6 ขั้นตอน ดังแสดงในรูปที่ 3.8 นั้น ในส่วนของการวิเคราะห์ปัญหาได้อธิบายไปแล้วข้างต้น ส่วนขั้นตอนอื่นๆสามารถอธิบายได้ดังนี้

3.3.1 การออกแบบโปรแกรมคอมพิวเตอร์

การออกแบบโปรแกรมคอมพิวเตอร์เป็นการประยุกต์ใช้ขั้นตอนวิธีอาณานิคมมดร่วมกับความสามารถในการประมวลผลด้วยคอมพิวเตอร์ และต้องสอดคล้องกับลักษณะของปัญหาการจัดตารางการผลิตแบบตามสั่ง ซึ่งในขั้นตอนนี้จะรวมถึงการปรับปรุงวิธีการแก้ปัญหาด้วยวิธีขั้นตอนอาณานิคมมด โดยมีขั้นตอนทั้งหมดดังนี้

3.3.1.1 การแสดงผลลัพธ์

ในขั้นตอนการแสดงผลลัพธ์เป็นขั้นตอนที่แสดงผลของการหาคำตอบระหว่างและหลังจากโปรแกรมคอมพิวเตอร์หาคำตอบเสร็จแล้ว โดยออกแบบให้โปรแกรมคอมพิวเตอร์แสดงผลลัพธ์อยู่รูปแบบของไฟล์ Text Document (.txt) โดยมีข้อมูลที่แสดงผลลัพธ์ดังนี้

- ค่าเวลาในการทำงานทั้งหมด (Makespan)
- เส้นทางวิกฤต (Critical path)
- ลำดับของการดำเนินการ (Operation sequence)
- เวลาที่ใช้ในการหาคำตอบ (Elapsed times)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1.2 การเขียนโปรแกรมคอมพิวเตอร์

การเขียนโปรแกรมคอมพิวเตอร์เป็นการนำผลลัพธ์ของการออกแบบโปรแกรมคอมพิวเตอร์เปลี่ยนมาเป็นภาษาคอมพิวเตอร์ ในวิทยานิพนธ์ฉบับนี้ใช้ C# ในการเขียนโปรแกรมคอมพิวเตอร์เพราะเป็นโปรแกรมคอมพิวเตอร์ที่มีความหลากหลายของการใช้ภาษา ทำให้ในการประมวลผลมีความยืดหยุ่นมากขึ้นและโปรแกรมคอมพิวเตอร์ Visual C# นั้น สามารถสร้างหน้าต่าง (Window form) ที่สะดวกต่อผู้ใช้งานและเป็นโปรแกรมคอมพิวเตอร์ที่มีความยืดหยุ่นมากในการเชื่อมต่อกับโปรแกรมคอมพิวเตอร์อื่นๆ

3.3.1.3 ข้อจำกัดของโปรแกรมคอมพิวเตอร์

โปรแกรมคอมพิวเตอร์ที่จัดทำขึ้นนั้น เป็นโปรแกรมคอมพิวเตอร์ที่รองรับกับรูปแบบการผลิตแบบตามสั่งของ Lawrence หรือระบบการผลิตอื่นๆที่สามารถจัดรูปแบบการผลิตให้ตรงกับรูปแบบการผลิตแบบตามสั่งของ Lawrence โดยโปรแกรมคอมพิวเตอร์มีข้อจำกัดดังนี้

- โปรแกรมคอมพิวเตอร์ไม่รองรับระบบการผลิตที่มีการระบุลำดับของงาน (มีการระบุว่าต้องทำงานใดก่อนงานใด เช่น เริ่ม งาน1 หรือ งาน2 เมื่อทำเสร็จแล้วทำ งาน 3, งาน 4 หรือ งาน 5 เมื่อทำเสร็จแล้วทำ งาน 6 หรือ งาน 7)
- โปรแกรมคอมพิวเตอร์ไม่รองรับระบบการผลิตที่มีการใช้เครื่องมือ (Tool) ร่วมกันของแต่ละการดำเนินการ เช่นการผลิตล้ออัลลอย โดยแต่ละงาน และแต่ละการดำเนินการมีการใช้เครื่องมือในการจับชิ้นงาน ในที่นี้คือ แคลมป์ (Clamp) ซึ่งจะมีจำนวนจำกัด และแคลมป์แต่ละชนิดสามารถใช้กับเครื่องจักรที่แตกต่างกัน เป็นต้น

ความแตกต่างของงานวิจัยนี้กับงานวิจัยอื่นๆที่ได้ศึกษามาคือ มีการพัฒนารูปแบบวิธีการจัดลำดับการผลิต ซึ่งจะทำให้ผลคำตอบที่ได้ลู่เข้าหาคำตอบที่ดีเร็วขึ้นเพราะมีการพัฒนาทิศทางการจัดลำดับให้มีสองทิศทางดังนี้ ทิศทางแรกเป็นการหาเส้นทางแบบไปข้างหน้า (Forward) ทิศทางที่สองเป็นการหาเส้นทางแบบกลับหลัง (Backward) การพัฒนาอย่างที่สองคือการพัฒนาการกำหนดค่าข้อมูลทางฮิวริสติค (η) ในด้านการให้ความสำคัญของการรักษาสมดุลของภาระงานของทุกๆ เครื่องจักร และสุดท้ายเป็นการพัฒนาวิธีการกำหนดค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่เหมาะสมของแต่ละปัญหา โดยมีการเพิ่มวิธีการปรับเปลี่ยนค่าทั้งสองอย่างอัตโนมัติพร้อมๆ กับการหาคำตอบ ซึ่งจะทำให้ขั้นตอนในการหาคำตอบลดลงเพราะไม่ต้องใช้วิธีลองผิดลองถูกในการหาค่าทั้งสองที่เหมาะสมและเวลาในการหาคำตอบลดลงเช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

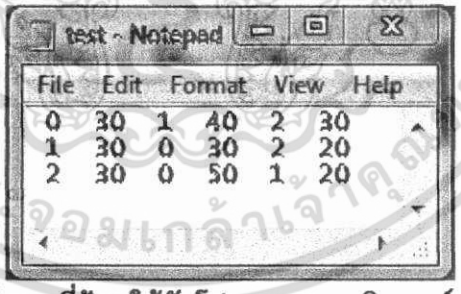
บทที่ 4
ผลการดำเนินงาน

จากการศึกษาการแก้ปัญหาการจัดตารางการผลิตด้วยขั้นตอนวิธีอาณานิคมมดและการออกแบบโปรแกรมคอมพิวเตอร์ในบทที่ 3 โดยลักษณะของโปรแกรมคอมพิวเตอร์อธิบายอยู่ในภาคผนวก ข ผู้วิจัยได้ดำเนินการทดลองและเปรียบเทียบผลคำตอบดังนี้

1. การตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์
2. การทดลองและเปรียบเทียบผลคำตอบ
    - การทดลองหาค่า α และ β ที่เหมาะสมโดยการกำหนดค่า
    - การทดลองหาค่าคำตอบที่ดีที่สุดโดยการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ
    - การทดลองและเปรียบเทียบผลคำตอบในปัญหาการผลิตแบบตามสั่งในปัจจุบัน

4.1 การตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์

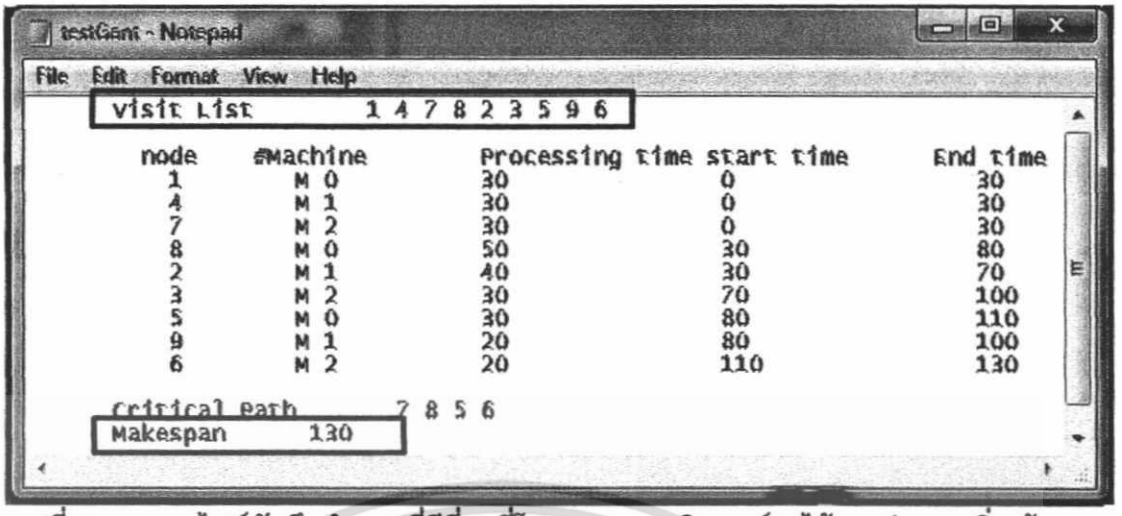
การตรวจสอบความถูกต้องของโปรแกรมคอมพิวเตอร์ทดสอบโดยการหาค่าเวลาในการดำเนินการทั้งหมด (Makespan) ของลำดับการผลิต จุดประสงค์ของการทดสอบนี้เพื่อตรวจสอบความถูกต้องของการจัดลำดับการผลิต และการนำลำดับการผลิต (Operation sequence) ที่ได้จากการจัดลำดับการผลิตมาคำนวณหาค่าเวลาการดำเนินการทั้งหมด (Makespan) ทดสอบโดยการนำปัญหาขนาดเล็กของ (Udomsakdigool and Kachitvichyanukul. 2008 : 4158) ให้โปรแกรมคอมพิวเตอร์จัดลำดับการผลิตเพื่อคำนวณหาค่าเวลาการดำเนินการทั้งหมด หลังจากนั้นนำมาเขียนแผนภูมิแกนต์ (Gant chart) โดยจะเปรียบเทียบคำตอบกับงานวิจัยของ Udomsakdigool (2008 : 4158)



รูปที่ 4.1 แสดงเมตริกซ์ปัญหาที่ป้อนให้กับโปรแกรมคอมพิวเตอร์ (Udomsakdigool and Kachitvichyanukul. 2008 : 4158)

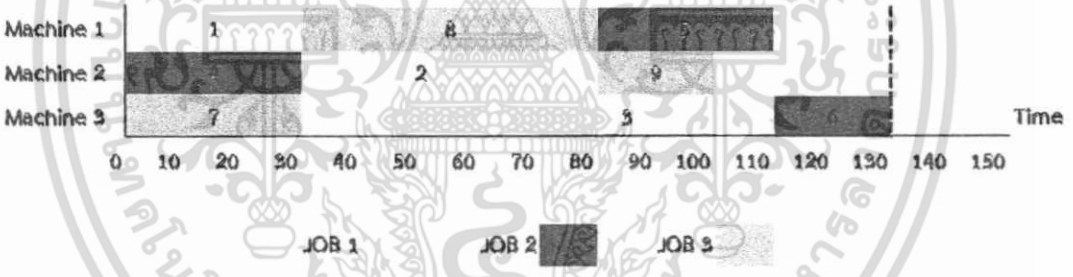
ทดสอบโดยการป้อนปัญหาขนาดเล็ก โดยลักษณะปัญหาที่ใช้ในการทดสอบเป็นตามรูปที่ 4.1 ซึ่งเป็นปัญหาการจัดตารางการผลิตแบบตามสั่งที่มี 3 งาน และ 3 เครื่องจักร หลังจากนั้นให้โปรแกรมคอมพิวเตอร์หาคำตอบซึ่งได้ผลการหาคำตอบดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงไฟล์บันทึกคำตอบที่ดีที่สุดที่โปรแกรมคอมพิวเตอร์หาได้และค่าเวลาเริ่มต้นและเวลาสิ้นสุดของแต่ละการดำเนินการ

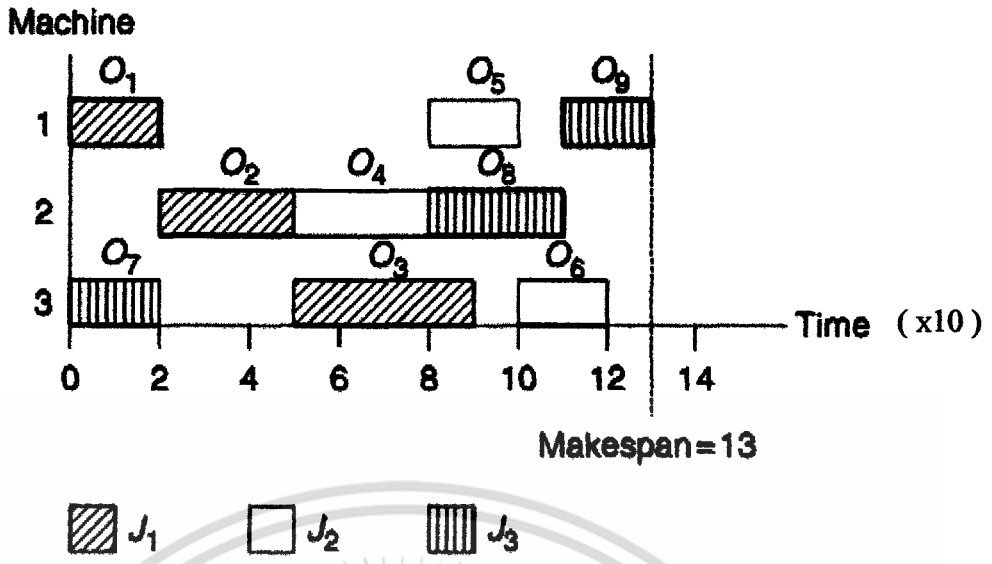
จากรูปที่ 4.2 จะได้ค่าลำดับการผลิตคือ 1 4 7 8 2 3 5 9 6 และได้ค่าเวลาในการดำเนินการทั้งหมด 130 หน่วยเวลา ผู้วิจัยนำค่าลำดับการผลิตนี้ ไปเขียนให้อยู่ในรูปของแผนภูมิแกนต์ ได้ดังรูปที่ 4.3



รูปที่ 4.3 แสดงแผนภูมิแกนต์ของการหาคำตอบโดยโปรแกรมคอมพิวเตอร์

จากรูปที่ 4.3 แสดงแผนภูมิแกนต์ของลำดับการผลิตซึ่งสามารถหาค่าเวลาการดำเนินการทั้งหมดได้ 130 หน่วยเวลา เพื่อเป็นการตรวจสอบความถูกต้องของการทำงานของโปรแกรมคอมพิวเตอร์ จึงนำมาคำตอบเปรียบเทียบกับงานวิจัยของ Udomsakdigool (2008 : 4158) โดยผลคำตอบของ Udomsakdigool ตามรูปที่ 4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แผนภูมิแกนต์งานวิจัยของ Udomsakdigool (2008 : 4158)

จากรูปที่ 4.4 ค่าเวลาในการดำเนินการทั้งหมดของ Udomsakdigool มีค่าเท่ากับ 130 หน่วยเวลาซึ่งมีค่าเท่ากับค่าที่โปรแกรมคอมพิวเตอร์หาได้ (ลำดับการผลิตจะแตกต่างกันเนื่องจากปัญหาดังกล่าว สมารถจัดลำดับการผลิตได้หลายรูปแบบ แต่มีค่าเวลาในการดำเนินการทั้งหมดที่ดีที่สุดเท่ากัน คือ 130 หน่วยเวลาเท่ากัน)

จากผลการทดสอบข้างต้นพบว่าค่าเวลาการดำเนินการทั้งหมดที่ได้จากโปรแกรมคอมพิวเตอร์มีค่าเท่ากันกับงานวิจัยของ Udomsakdigool สรุปได้ว่าโปรแกรมคอมพิวเตอร์ที่พัฒนาขึ้นสามารถหาลำดับการผลิตและค่าค่าเวลาในการดำเนินการทั้งหมดได้อย่างถูกต้อง

4.2 การทดลองและเปรียบเทียบผลคำตอบ

ในการทดลองและเปรียบเทียบผลคำตอบ ผู้วิจัยได้ทำการทดลองใน 2 ประเภทของปัญหาเพื่อให้ครอบคลุมในด้านของประสิทธิภาพในการหาคำตอบ โดยมีการทดลองดังนี้

1.การทดลองและเปรียบเทียบผลคำตอบกับชุดปัญหามาตรฐาน

2.การทดลองและเปรียบเทียบผลคำตอบกับปัญหาการผลิตแบบตามสั่งในปัจจุบัน

4.2.1 การทดลองและเปรียบเทียบผลคำตอบกับชุดปัญหามาตรฐาน

โปรแกรมคอมพิวเตอร์การจัดตารางการผลิตแบบตามสั่งด้วยวิธีอาณานิคมมดจะทำการทดลองหาคำตอบกับชุดปัญหามาตรฐาน Lawrence โดยเมตริกซ์ของปัญหานำมาจาก OR-Library ซึ่งเป็นชุดปัญหาที่ได้รับการยอมรับจากการวิจัยต่างๆในเรื่องของความใกล้เคียงกับสภาวะงานอุตสาหกรรมจริง โดยจะทำการทดลองในปัญหา La01 จนถึง La15 ซึ่งจะครอบคลุมปัญหาที่มี 10 งานและ 5 เครื่องจักร, 15 งานและ 5 เครื่องจักรและปัญหาที่มี 20 งานและ 5 เครื่องจักรดังแสดงในตารางที่ 4.1 ซึ่งขนาดปัญหาดังกล่าวข้างต้นถือว่าครอบคลุมกับระบบการผลิตจริงในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 รายละเอียดของปัญหามาตรฐาน Lawrence **ใช้ในการทดลอง**

ชื่อปัญหา	จำนวนงาน	จำนวนเครื่องจักร	จำนวนการดำเนินการ
LA01	10	5	50
LA02	10	5	50
LA03	10	5	50
LA04	10	5	50
LA05	15	5	75
LA06	15	5	75
LA07	15	5	75
LA08	15	5	75
LA09	15	5	75
LA10	15	5	75
LA11	20	5	100
LA12	20	5	100
LA13	20	5	100
LA14	20	5	100
LA15	20	5	100
LA16	10	10	100
LA17	10	10	100
LA18	10	10	100
LA19	10	10	100
LA20	10	10	100

คำตอบที่ได้จากการทดลองคือค่าเวลาในการดำเนินการทั้งหมด (Makespan) ซึ่งถ้าคำตอบมีค่าน้อยแสดงว่าตารางการผลิตนั้นมีประสิทธิภาพสูง โดยการทดลองหาค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่เหมาะสมที่จะใช้การปรับเปลี่ยนค่า α และ β และทดลองหาคำตอบที่ดีที่สุด หลังจากนั้นนำผลคำตอบที่ได้เปรียบเทียบกับคำตอบที่ดีที่สุด (Optimal solution) และหาค่าเปอร์เซ็นต์ความผิดพลาด (MRE%, Mean relative error)

การทดสอบประสิทธิภาพของโปรแกรมคอมพิวเตอร์ ได้จัดทำการเขียนชุดคำสั่งด้วยโปรแกรมคอมพิวเตอร์ Visual C# และทดสอบอัลกอริทึมกับคอมพิวเตอร์รุ่น Pentium Core Quad ที่ความถี่ 2.5 GHz. และในหน่วยความจำ 2 GB บนระบบปฏิบัติการ Window 7 Professional โดยมีการทดลองดังนี้

4.2.1.1 การทดลองหาค่า α และ β ที่เหมาะสมโดยการกำหนดค่า

การทดลองหาค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่เหมาะสมของแต่ละปัญหา ผู้วิจัยต้องทำการคัดเลือกค่า α และ β ที่ใช้ในการทดลองหาคำตอบ ซึ่งจากการที่ผู้วิจัยได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ศึกษาจากงานวิจัยอื่นๆ ที่นำปัญหาของ Lawrence มาทดลองแก้ปัญหาเช่นเดียวกับงานวิจัยนี้ และจากการทดลองลองผิดลองถูกปรับเปลี่ยนค่า α และ β ผู้วิจัยจึงได้มาซึ่งค่า α และ β ดังต่อไปนี้

$$\alpha = \{0.1, 0.25, 0.5, 1\} \text{ และ } \beta = \{0.25, 0.5, 0.75, 1, 2\}$$

จากค่า α และ β อย่างละสี่ค่าข้างต้น จะได้คู่ลำดับทั้งหมด 16 คู่ ดังแสดงในตารางที่ 4.2

ตารางที่ 4.2. ค่า α และ β ที่ใช้ในการทดลองในการแก้ปัญหาแต่ละปัญหา

ลำดับ	α	β	ลำดับ	α	β
1	0.10	0.25	8	0.50	0.25
2	0.10	0.50	9	0.50	0.50
3	0.10	0.75	11	0.50	0.75
4	0.10	1.20	12	0.50	1.20
5	0.25	0.25	13	1.00	0.25
6	0.25	0.50	14	1.00	0.50
7	0.25	0.75	15	1.00	0.75
8	0.25	1.20	16	1.00	1.20

การทดลองหาค่า α และ β ที่เหมาะสมโดยการกำหนดค่า ทำการทดลองซ้ำ 3 ครั้ง สำหรับแต่ละคู่ α และ β (คู่ลำดับ (α, β) จำนวน 16 คู่ ทดลองซ้ำ 3 ครั้ง จะได้จำนวนการทดลองทั้งหมด 48 การทดลองต่อ 1 ปัญหา) และเปรียบเทียบกับผลที่ได้จากวิธีฮิวริสติคอื่นได้แก่

- RD-ACS: เป็นวิธีอาณานิคมมด ของการจัดตารางการผลิตแบบตามสั่งด้วยเส้นทางที่ยืดหยุ่นและสามารถแยกเวลาติดตั้ง (Andrea.and Gino. 2007 : 512)
- TSSB: เป็นวิธีการค้นหาแบบทาบู (Tabu search) โดยอาศัยการ Shifting bottleneck (Ferdinando. and Emanuela. 2000 : 305)
- HGA: เป็นวิธีการทางพันธุกรรมในการจัดตารางการผลิตแบบตามสั่ง (Goncalves. et.al. 2005 : 90)
- Beam search (Sabuncuoglu. and Bayiz. 1999 : 402)

ในการการทดลองหาค่า α และ β ที่เหมาะสมโดยการกำหนดค่ามีค่าคงที่ในการทดลองดังต่อไปนี้

จำนวนมด 200 ตัว/รอบ, ดำเนินการวนซ้ำ 200 รอบ, q_0 = 0.0001, ρ = 0.7, C = 50, x_1 = 0.8, x_2 = 0.2, y = 10 และ z = 0.3

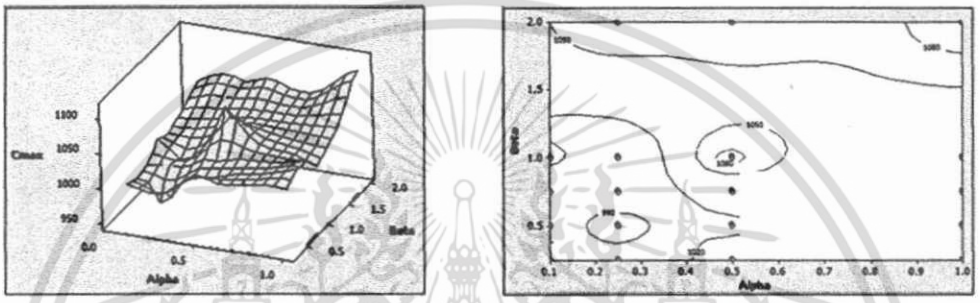
ผลคำตอบที่ได้เปรียบเทียบกับคำตอบที่ดีที่สุด (Optimal solution) และหาค่าเปอร์เซ็นต์ความผิดพลาด (MRE%, Mean relative error) ผลการทดลองดังรูปที่ 4.4 และดังตารางที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 แสดงค่าเวลาที่ใช้ในการดำเนินการ (Makespan) ของการวิจัยนี้เปรียบเทียบกับวิธีอื่นและค่า α และ β ที่เหมาะสมแต่ละปัญหา

ปัญหา	จำนวน งาน x เครื่องจักร	คำตอบที่ดีที่สุด (Optimal)	HGA (2005)	TSSB (2000)	RD-ACS (2007)	Beam search (1999)	คำตอบจากการวิจัย					
							ค่าเฉลี่ย	สูงสุด	ต่ำสุด	MRE%	α	β
la01	10x5	**666**	666	666	666	666	698	721	681	2.25%	0.25	0.75
la02	10x5	**655**	655	655	665	704	726	734	716	9.31%	0.10	0.75
la03	10x5	**597**	603	597	604	650	686	737	657	10.05%	0.25	0.50
la04	10x5	**590**	598	590	611	620	649	653	643	8.98%	0.10	0.50
la05	10x5	**593**	593	593	593	593	**593**	**593**	**593**	**0.00%**	0.10	0.50
la06	15x5	**926**	926	926	926	926	949	962	933	0.76%	0.10	0.50
la07	15x5	**890**	890	890	890	890	959	982	941	5.73%	0.10	0.75
la08	15x5	**863**	863	863	863	863	941	951	927	7.42%	0.10	0.50
la09	15x5	**951**	951	951	951	951	963	974	**951**	**0.00%**	0.25	0.50
la10	15x5	**958**	958	958	958	958	963	973	**958**	**0.00%**	0.10	0.25
la11	**20x5**	**1222**	1222	1222	1222	1222	1294	1299	1292	**5.73%**	0.25	0.50
la12	20x5	**1039**	1039	1039	1039	1039	1099	1091	1098	5.68%	0.10	0.25
la13	20x5	**1150**	1150	1150	1150	1150	1217	1228	1185	3.04%	0.10	0.50
la14	20x5	**1292**	1292	1292	1292	1292	**1292**	**1292**	**1292**	**0.00%**	0.10	0.25
la15	20x5	**1207**	1207	1207	1212	1207	1375	1408	1338	10.85%	0.25	0.50

จากรูปที่ 4.5 สามารถวิเคราะห์แนวโน้มของผลคำตอบที่เกิดจากการเปลี่ยนค่า α และ β คือ ถ้าค่า α และ β มากขึ้นจะส่งผลให้ค่าเวลาในการดำเนินการทั้งหมด (Makespan) มีแนวโน้มมากขึ้น และสามารถวิเคราะห์ได้ว่า ค่า α ไม่มีผลต่อค่าเวลาในการดำเนินการทั้งหมด และถ้านำผลการทดลองจากตารางที่ 4.3 มาวิเคราะห์ร่วมกันจะสรุปได้ว่า ค่า β ต้องมีค่ามากกว่า α จึงจะทำให้เกิดคำตอบที่ดีที่สุด และผลคำตอบของการวิจัยนี้เปรียบเทียบกับผลคำตอบที่ดีที่สุดและการวิจัยอื่น ให้ผลที่ยอมรับได้ โดยให้ค่าเปอร์เซ็นต์ความผิดพลาด (MRE%) ไม่เกิน 10% ในปัญหา la01-la14 ส่วนในปัญหาที่มีค่าเปอร์เซ็นต์ความผิดพลาดมากกว่า 10% เนื่องจากจำนวนมดและจำนวนรอบมีค่าน้อยสำหรับการแก้ปัญหานั้น และจากการทดลองหาคำตอบที่ดีที่สุดการวิจัยนี้ได้ผลคำตอบที่ดีที่สุดในชุดปัญหา la05, la09, la10, la14 จากปัญหาทั้งหมดที่ทำการทดลอง



(a) Surface plot (b) Contour plot

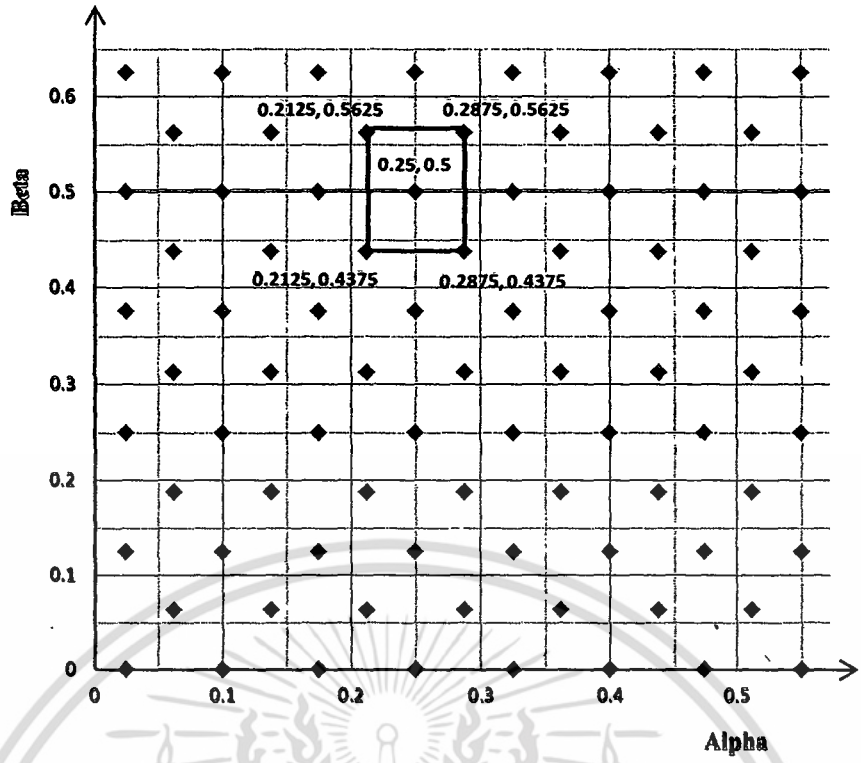
รูปที่ 4.5 แสดงความสัมพันธ์ระหว่างค่า α และ β เปรียบเทียบกับค่าเวลาในการดำเนินการทั้งหมดในชุดปัญหา La09

4.2.1.2 การทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ

หลังจากทดลองหาค่า α และ β ที่เหมาะสมของแต่ละปัญหา ทำให้ทราบแนวโน้มของผลคำตอบที่เกิดจากการเปลี่ยนค่า α และ β ต่างๆ หลังจากนั้นทางผู้วิจัยได้ออกแบบการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ ตามที่ได้นำเสนอไปในหัวข้อ 3.2.4 โดยมีวิธีการปรับเปลี่ยนค่า α และ β ดังนี้

— มดทุกตัวในแต่ละรอบ ต้องจัดตารางการผลิตทั้งหมด 5 ครั้ง (คู่ α และ β จำนวน 5 คู่) โดยให้ค่า (α , β) ในรอบแรกจะมีคู่ลำดับดังต่อไปนี้ (0.25,0.5), (0.2125,0.5625), (0.212,0.4375), (0.2875,0.5625), (0.2875,0.4375)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงค่าเริ่มต้นของ α และ β ทั้ง 5 คู่แรกที่มดนำไปจัดตารางการผลิตสำหรับการดำเนินการรอบแรก

ค่าเริ่มต้นของ α และ β ทั้ง 5 คู่แรกที่มดนำไปจัดตารางการผลิตสำหรับการดำเนินการรอบแรกที่ได้แสดงไว้ในรูปที่ 4.6 เกิดจากการวิเคราะห์การทดลองค่า α และ β ที่เหมาะสมของแต่ละปัญหา โดยพบว่าในปัญหาของ Lawrence ค่า α และ β ที่เหมาะสมของแต่ละปัญหา จะมีค่าใกล้เคียงกับ 0.25 และ 0.5 ตามลำดับ จึงกำหนดค่านี้เป็นจุดกึ่งกลาง ส่วนอีก 4 จุดรอบข้างเกิดจากการบวกและลบด้วยค่าคงที่ โดยถ้าเป็นค่า α จะบวกและลบด้วย 0.0375 ส่วนค่า β จะบวกและลบด้วย 0.0625 โดยค่าคงที่ทั้งสองค่านี้ผู้วิจัยกำหนดจากความละเอียดของการค้นหา

– รอบอื่นๆที่ไม่ใช่รอบแรก ค่า α และ β จะเปลี่ยนไปในทิศทางของค่า α และ β ที่ให้คำตอบดีที่สุด (Makespan น้อยที่สุด) โดยค่า α และ β ใหม่จะได้จากสมการที่ 4.1, 4.2, 4.3, 4.4 และ 4.5

$$(\alpha_{jump}, \beta_{jump}) = \begin{cases} (\alpha_{best} \cdot j_\alpha \cdot 3,\ \beta_{best} \cdot j_\beta \cdot 3) & ,\text{ถ้า } decrease \geq 0.10 \\ (\alpha_{best} \cdot j_\alpha \cdot 2,\ \beta_{best} \cdot j_\beta \cdot 2) & ,\text{ถ้า } decrease \geq 0.05 \\ (\alpha_{best},\ \beta_{best}) & ,\text{ถ้า } 0 \leq decrease < 0.10 \end{cases} \quad (4.1)$$

โดย $\alpha_{jump}, \beta_{jump}$ = ค่า α และ β ในตำแหน่งที่ทิศทางนั้นมีแนวโน้มให้ผลคำตอบดีขึ้นซึ่งค่านี้จะเลื่อนไปอย่างก้าวกระโดด ตามสัดส่วนการลดลงของผลคำตอบ

$\alpha_{best}, \beta_{best}$ = ค่า α ที่ดีที่สุดและค่า β ที่ดีที่สุดของรอบนี้ ตามลำดับ

j_α = ± 0.0375 (มีทิศทางตามค่า α_{best})

j_β = ± 0.0625 (มีทิศทางตามค่า β_{best})

$decrease$ = ค่าสัมบูรณ์สัดส่วนการลดลงของผลคำตอบหาได้จากสมการที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$decrease = \left|\frac{makespan_{\text{รอบก่อนหน้า}} - makespan_{\text{รอบนี้}}}{makespan_{\text{รอบก่อนหน้า}}}\right| \quad (4.2)$$

สมการที่ 4.1 แสดงวิธีการค้นหาค่า α และ β โดยจะมีการค้นหาแบบก้าวกระโดดตามสัดส่วนการลดลงของผลคำตอบของรอบก่อนหน้า ถ้าค่าสัดส่วนการลดลงของผลคำตอบมีมาก การเลื่อนค่า α และ βก็จะมีการก้าวกระโดด ถ้าสัดส่วนการลดลงของคำตอบมากการก้าวกระโดดก็จะมากเช่นกันตามสัดส่วนในสมการที่ 4.1 โดยค่าสัดส่วนการลดลงของผลคำตอบหาได้จากสมการที่ 4.2 ซึ่งจะเป็นค่าสัดส่วนการเปลี่ยนแปลงของผลคำตอบ การก้าวกระโดดของการค้นหา α และ β จะทำให้ลดเวลาและจำนวนรอบของการค้นหาคำตอบได้

หลังจากการปรับค่า α และ β ไปในทิศทางที่มีค่า Makespane น้อยที่สุด และใช้หลักการก้าวกระโดดตามสมการ 4.1 และ 4.2 หลังจากที่ได้คู่ลำดับ α และ β ถัดไปแล้ว นำจุดนั้นมาหาค่า α และ β อีก 4 จุดตามสมการ 4.3 และถ้าคำตอบในรอบใด ค่า Makespan ที่ดีที่สุดเกิดขึ้นจากค่า α และ β จุดกึ่งกลาง นั่นคือค่า Makespan มีค่าเท่าเดิม ส่งผลให้ค่า $decrease$ เท่ากับ 0 ในรอบต่อไปจะมีการลดค่า $\Delta\alpha$ และ $\Delta\beta$ตามสมการที่ 4.4 และ 4.5

$$(\alpha_{new}, \beta_{new}) = \begin{cases} (\alpha_{jump}, \beta_{jump}) \\ (\alpha_{jump} + \Delta\alpha, \beta_{jump} + \Delta\beta) \\ (\alpha_{jump} + \Delta\alpha, \beta_{jump} - \Delta\beta) \\ (\alpha_{jump} - \Delta\alpha, \beta_{jump} + \Delta\beta) \\ (\alpha_{jump} - \Delta\alpha, \beta_{jump} - \Delta\beta) \end{cases} \quad (4.3)$$

$$\Delta\alpha = \begin{cases} 0.03750 & ,\text{ถ้า } decrease > 0 \\ 0.01875 & ,\text{ถ้า } decrease = 0 \end{cases} \quad (4.4)$$

$$\Delta\beta = \begin{cases} 0.06250 & ,\text{ถ้า } decrease > 0 \\ 0.03125 & ,\text{ถ้า } decrease = 0 \end{cases} \quad (4.5)$$

โดย $\alpha_{new}, \beta_{new}$ = ค่า α และ β ใหม่ทั้ง 5 คู่ในรอบถัดไปที่มดนำไปจัดตารางการผลิตและกำหนดขอบเขต $0.025 \leq \alpha_{new} \leq 1$ และ $0 \leq \beta_{new} \leq 1$

จากสมการที่ 4.3 4.4 4.5 จะเห็นได้ว่าการวิธีการปรับเปลี่ยนค่า α และ β เป็นไปอย่างก้าวกระโดดตามสัดส่วนของการลดลงของผลคำตอบ ซึ่งจะทำให้ลดเวลาและจำนวนครั้งในการหาคำตอบที่ดีที่สุด โดยมีกระบวนการดังรูปที่ 3.6

ผู้วิจัยทำการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติโดยกำหนดตัวแปรต่างๆดังนี้ จำนวนมด 50 ตัว/รอบ, ดำเนินการวนซ้ำ 200 รอบ, q_0 = 0.0001, ρ = 0.7, C = 50, x_1 = 0.8, x_2 = 0.2, y = 10 และ z = 0.3 ทำการทดลองซ้ำ 5 ครั้งในแต่ละปัญหามาตรฐาน ตั้งแต่ปัญหา la01 ถึง la20 โดยได้ผลการทดลองดังตารางที่ 4.4 และการศึกษาผลกระทบของจำนวนรอบ (Iteration) ต่อผลคำตอบและเวลาในการหาคำตอบได้ผลดังตารางที่ 4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 แสดงค่าเฉลี่ย α , β และเวลาในการหาคำตอบของการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ

ปัญหา	จำนวนงาน	จำนวนเครื่องจักร	คำตอบที่ดีที่สุด	α เฉลี่ย	β เฉลี่ย	Elapse time เฉลี่ย(s)	Makespan								
							ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	SD	ค่าเฉลี่ย		ค่าต่ำสุด	
												คำตอบ	MRE%	คำตอบ	MRE%
la01	10	5	666	0.141	0.365	189.039	667	714	680	816	58.55	717.60	7.75%	667	0.15%
la02	10	5	655	0.150	0.677	193.614	736	743	741	755	8.73	741.40	13.19%	732	11.76%
la03	10	5	597	0.103	0.417	194.665	679	686	672	689	8.35	679.20	13.77%	670	12.23%
la04	10	5	590	0.194	0.312	191.400	665	637	652	639	15.48	653.00	10.68%	637	7.97%
la05	10	5	593	0.350	0.625	191.243	593	593	593	593	0.00	**593.00**	**0.00%**	**593**	**0.00%**
la06	15	5	926	0.113	0.542	393.586	967	977	955	995	18.30	968.60	4.60%	949	2.48%
la07	15	5	890	0.228	0.698	404.320	998	962	966	980	14.10	976.40	9.71%	962	8.09%
la08	15	5	863	0.059	0.594	402.658	957	950	937	971	13.27	956.00	10.78%	937	8.57%
la09	15	5	951	0.144	0.411	398.309	1049	990	1012	1005	21.94	1015.40	6.77%	990	4.10%
la10	15	5	958	0.063	0.167	396.816	958	987	958	973	12.86	971.00	1.36%	**958**	**0.00%**
la11	20	5	1222	0.203	0.391	694.911	1310	1287	1277	1269	16.65	1288.60	5.45%	1269	3.85%
la12	20	5	1039	0.125	0.168	693.518	1102	1143	1121	1132	16.47	1121.60	7.95%	1102	6.06%
la13	20	5	1150	0.134	0.401	697.614	1263	1244	1217	1259	21.17	1240.80	7.90%	1217	5.83%
la14	20	5	1292	0.141	0.432	695.042	1325	1294	1292	1292	14.56	1299.00	0.54%	**1292**	**0.00%**
la15	20	5	1207	0.106	0.313	696.237	1395	1396	1406	1417	10.76	1400.80	16.06%	1390	15.16%
la16	10	10	945	0.081	0.694	530.604	1121	1162	1127	1073	32.68	1117.20	18.22%	1073	13.54%
la17	10	10	784	0.306	0.600	527.752	923	994	996	914	38.99	959.80	22.42%	914	16.58%
la18	10	10	848	0.115	0.219	541.683	1062	1034	1049	1013	24.77	1047.00	23.47%	1013	19.46%
la19	10	10	842	0.078	0.644	534.978	1052	1053	1056	1051	4.44	1051.20	24.85%	1044	23.99%
la20	10	10	902	0.123	0.563	534.392	1139	1081	1164	1145	31.00	1132.40	25.54%	1081	19.84%

หมายเหตุ : Elapse time คือ ค่าเวลาที่ใช้ในการหาคำตอบ มีหน่วยเป็นวินาที
SD คือ ค่าการเบี่ยงเบนมาตาฐานของคำตอบ

ตารางที่ 4.5 แสดงค่าเฉลี่ยของ α , β และเวลาในการหาคำตอบของการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ ของปัญหา la04 ในจำนวนรอบ (Iteration)ที่แตกต่างกัน

จำนวนรอบ (Iteration)	α เฉลี่ย	β เฉลี่ย	Elapse time เฉลี่ย(s)	คำตอบที่ดีที่สุด	Makespan				
					SD	ค่าเฉลี่ย		ค่าต่ำสุด	
						คำตอบ	MRE%	คำตอบ	MRE%
200	0.213	0.275	191.400	590	15.48	653.00	10.68%	637	7.97%
400	0.059	0.063	371.021	590	10.08	659.20	11.73%	643	8.98%
600	0.070	0.250	548.553	590	10.45	656.40	11.25%	642	8.81%
800	0.287	0.125	749.598	590	10.01	644.60	9.25%	634	7.46%
1000	0.070	0.444	929.732	590	6.91	644.40	9.22%	634	7.46%
1400	0.213	0.069	1348.531	590	13.33	640.40	8.54%	629	6.61%
1800	0.048	0.106	1689.541	590	4.36	639.00	8.31%	634	7.46%
3000	0.422	0.275	2880.463	590	5.76	638.80	8.27%	635	7.63%
5000	0.100	0.113	4830.647	590	4.55	629.20	6.64%	623	5.59%
10000	0.025	0.063	10021.890	590	3.36	633.60	7.39%	630	6.78%

จากผลการทดลองในตารางที่ 4.4 จะพบว่าการหาคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ ให้ผลคำตอบอยู่ในช่วงที่ยอมรับได้ (MRE%<10%) คิดเป็นร้อยละ 60 ของปัญหาทั้งหมดที่ทำการทดลอง และให้ผลคำตอบเท่ากับคำตอบที่ดีที่สุดจำนวน 3 ปัญหา คือปัญหา la05, la10 และ la14 คิดเป็นร้อยละ 15 ของปัญหาทั้งหมดที่ทำการทดลอง ซึ่งค่า MRE% และเวลาในการหาคำตอบ (Elapse time) มีแนวโน้มไปในทิศทางเดียวกันกับขนาดของปัญหาคือจะเพิ่มขึ้นเมื่อขนาดของปัญหามากขึ้น โดยจากผลการทดลองพบว่าจำนวนงานมีผลมากกว่าจำนวนของเครื่องจักร และจากผลการศึกษาผลกระทบของจำนวนรอบต่อผลคำตอบและเวลาในการหาคำตอบในตารางที่ 4.5 พบว่าจำนวนรอบที่มากขึ้นทำให้ได้ผลคำตอบมีแนวโน้มที่ดีขึ้นและการกระจายตัวของผลคำตอบ (Standard deviation, SD) มีแนวโน้มลดน้อยลง

4.2.2 การทดลองและเปรียบเทียบผลคำตอบในปัญหาการผลิตแบบตามสั่งในปัจจุบัน

เพื่อเป็นการทดสอบสมรรถนะของโปรแกรมคอมพิวเตอร์จัดตารางการผลิตแบบตามสั่ง ให้ครอบคลุมถึงระบบการผลิตจริงในปัจจุบัน ผู้วิจัยจึงนำตัวอย่างปัญหาระบบการผลิตแบบตามสั่งของบริษัท Shanghai Volkswagen Automobile Co.Ltd. มาเป็นปัญหาตัวอย่างในการทดสอบและเปรียบเทียบสมรรถนะของโปรแกรมคอมพิวเตอร์ที่จัดทำขึ้น ซึ่งได้นำมาจากงานวิจัยของ Xie และ Huo. 2010 โดยลักษณะของปัญหาดังตารางที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 ตัวอย่างปัญหาระบบการผลิตแบบตามสั่งของบริษัท Shanghai Volkswagen Automobile Co.Ltd. (Xie and Huo. 2010 : 942)

Process number	1-1	1-2	1-3	1-4						
Required time	10	30	20	10						
Machine code	6	3	2	2						
Process number	2-1	2-2	2-3	2-4	2-5					
Required time	10	20	20	20	10					
Machine code	6	1	2	1	2					
Process number	3-1	3-2	3-3	3-4						
Required time	10	20	30	10						
Machine code	7	4	5	2						
Process number	4-1	4-2	4-3	4-4	4-5					
Required time	10	20	20	30	10					
Machine code	6	2	1	5	1					
Process number	5-1	5-2	5-3	5-4	5-5	5-6	5-7	5-8	5-9	5-10
Required time	10	20	20	10	20	20	30	10	30	10
Machine code	7	4	4	2	3	1	5	6	5	1

หมายเหตุ: Process number เป็นตัวเลขที่แสดงถึง ลำดับของการดำเนินการในแต่ละงาน เลขตัวหน้า คือ ลำดับของงาน เลขตัวหลัง คือ ลำดับของการดำเนินการ Required time เป็นตัวเลขที่แสดงถึงเวลาในการผลิตของแต่ละการดำเนินการ Machine code เป็นตัวเลขที่แสดงถึง เครื่องจักรที่การดำเนินการนั้นใช้ในการผลิต

ลักษณะของปัญหาของบริษัท Shanghai Volkswagen Automobile Co.Ltd. สามารถวิเคราะห์ได้คือ มีจำนวนงานทั้งหมด 5 งาน มีการดำเนินการทั้งหมด 28 การดำเนินการและมีเครื่องจักร 7 เครื่องจักร ลักษณะที่แตกต่างจากปัญหาของ Lawrence มีดังนี้

- **แต่ละงาน จำนวนการดำเนินการไม่เท่ากัน** ยกตัวอย่างเช่น ในงานที่ 1 มีการดำเนินการ 4 การดำเนินการ แต่ในงานที่ 5 มีการดำเนินการ 10 การดำเนินการ
- ***ภายในงานเดียวกัน การดำเนินการสามารถใช้เครื่องจักรซ้ำได้*** ยกตัวอย่างเช่น ในงานที่ 5

  เครื่องจักรที่ 1 จำนวนสองการดำเนินการ คือ การดำเนินการที่ 5-6 และ 5-10
  เครื่องจักรที่ 4 จำนวนสองการดำเนินการ คือ การดำเนินการที่ 5-2 และ 5-3
  เครื่องจักรที่ 5 จำนวนสองการดำเนินการ คือ การดำเนินการที่ 5-7 และ 5-9

4.2.2.2 การทดลองแก้ปัญหา

ทำการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ ซึ่งกำหนดตัวแปรต่างๆดังนี้ จำนวนมด 50 ตัว/รอบ, ดำเนินการวนซ้ำ 200 รอบ, q_0 = 0.0001, ρ = 0.7, C = 50, x_1 = 0.8, x_2 = 0.2, y = 10 และ z = 0.3 ทำการทดลองซ้ำ 5 ครั้ง โดยได้ผลการทดลองดังตารางที่ 4.7

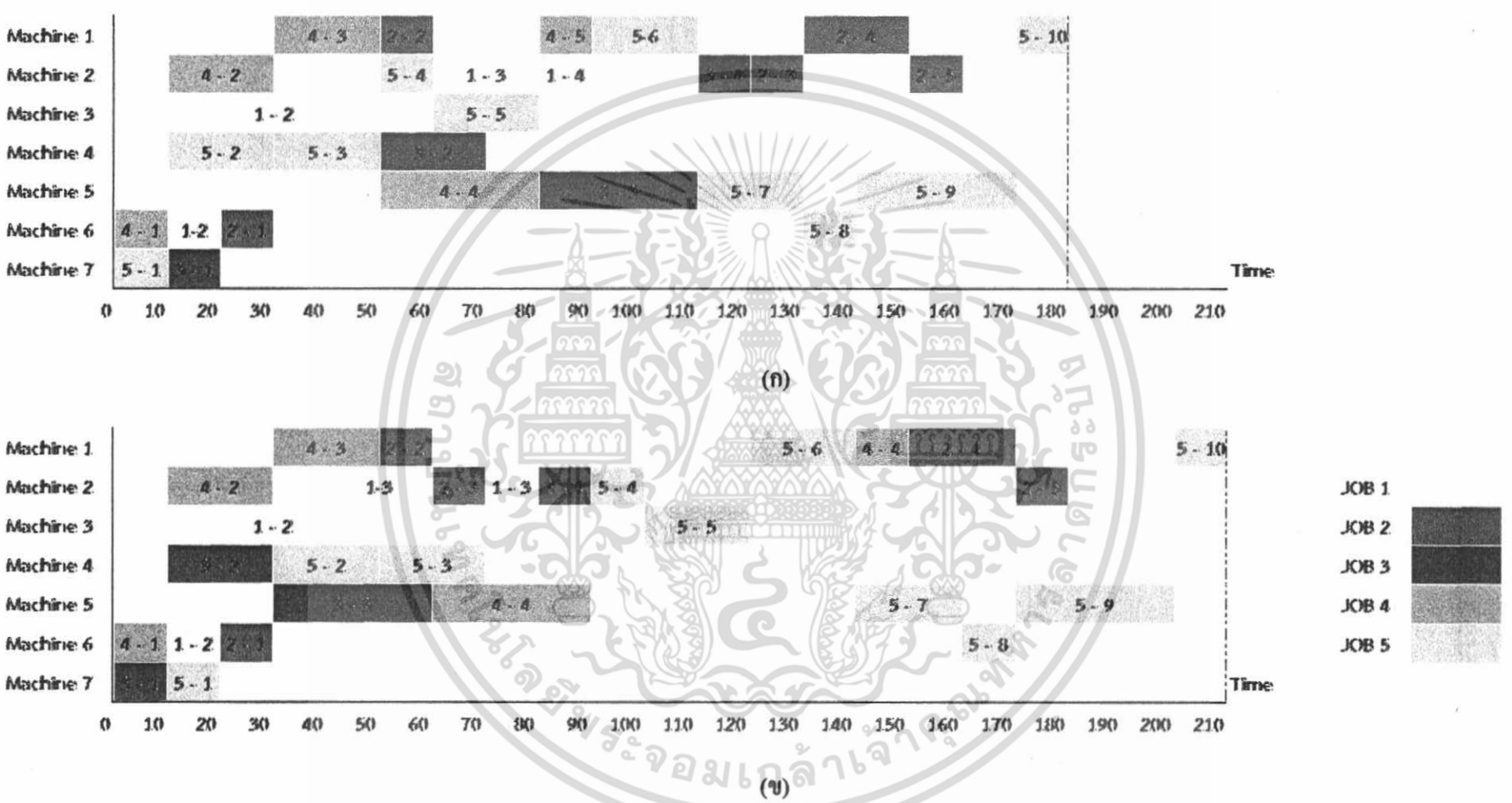
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 ผลการทดลองแก้ปัญหา บริษัท Shanghai Volkswagen Automobile Co.Ltd.

ครั้งที่	α	β	Elapse time (s)	Makespane
1	0.2875	0.5625	123.331	180
2	0.2875	0.5625	129.714	180
3	0.2875	0.4375	129.917	180
4	0.2125	0.5625	128.747	180
5	0.2125	0.4375	127.670	180

จากตารางที่ 4.7 พบว่าการทดลองทั้ง 5 ครั้งได้ผลคำตอบเท่ากันคือใช้เวลาในการดำเนินการทั้งหมด 180 หน่วยเวลา และค่า α และ β ที่ได้มีค่าใกล้เคียงกัน เมื่อนำผลการทดลองนี้เปรียบเทียบกับผลคำตอบของการวิจัยของ Xie ที่ผู้วิจัยได้นำตัวอย่างปัญหานี้มาทดลองหาคำตอบ ซึ่งเป็นงานวิจัยที่แก้ปัญหาการจัดตารางการผลิตแบบตามสั่งด้วยวิธีการเชิงพันธุศาสตร์ (Genetics Algorithm) โดยการวิจัยของ Xie ได้ผลคำตอบ 210 หน่วยเวลา นั่นก็คืองานวิจัยนี้มีสมรรถนะในแก้ปัญหาการจัดตารางการผลิตของบริษัท Shanghai Volkswagen Automobile Co.Ltd. ได้ดีกว่า 30 หน่วยเวลา โดยสามารถนำผลการหาคำตอบมาเปรียบเทียบดังแผนภูมิแกนต์ดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แผนภูมิแกนต์แสดงตารางการผลิตของบริษัท Shanghai Volkswagen Automobile Co.Ltd.

(ก) ตารางการผลิตจากการแก้ปัญหาด้วยงานวิจัยนี้ (ข) ตารางการผลิตจากการแก้ปัญหาด้วยงานวิจัย Xie และ Huo. 2010

จากแผนภูมิแกนต์ในรูปที่ 4.7 ก เป็นตารางการผลิตจากการแก้ปัญหาด้วยงานวิจัยนี้ ส่วนรูปที่ 4.7 ข เป็นตารางการผลิตจากการแก้ปัญหาด้วยงานวิจัย Xie และ Huo จะเห็นได้ว่าการวิจัยนี้สามารถลดค่าเวลาในการดำเนินการลงได้ 30 หน่วยเวลา

จากผลการทดลองทั้งสองส่วนคือ การทดลองและเปรียบเทียบผลคำตอบกับชุดปัญหามาตรฐาน และการทดลองและเปรียบเทียบผลคำตอบในปัญหาการผลิตแบบตามสั่งในปัจจุบัน สามารถสรุปได้ว่าการหาคำตอบการจัดตารางการผลิตแบบสั่งของการวิจัยนี้ มีประสิทธิภาพ เพราะคำตอบที่หาได้อยู่ในช่วงที่ยอมรับได้และสามารถลดเวลาในการหาคำตอบได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินงาน

5.1 สรุปและอภิปรายผลการดำเนินงาน

งานวิจัยนี้นำเสนอขั้นตอนวิธีอาณานิคมมด สำหรับการค้นหาคำตอบที่ดีที่สุดของปัญหาการจัดตารางการผลิตแบบตามสั่ง และในงานวิจัยนี้ได้แก้ไขและปรับปรุงขั้นตอนวิธีอาณานิคมมดที่ใช้ในการแก้ปัญหาการผลิตแบบตามสั่ง ในเรื่องของรูปแบบการส่งมดออกไปหาเส้นทางรวมไปถึงรูปแบบการปรับปรุงค่าฟีโรโมน วิธีการหาค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่เหมาะสม โดยทำการทดสอบกับชุดปัญหามาตรฐานของ Lawrence จาก OR-Library ทั้งหมด 20 ปัญหา และปัญหาการจัดตารางการผลิตของบริษัทตัวอย่าง ในส่วนของการทดสอบในชุดปัญหามาตรฐาน Lawrence ได้แบ่งการทดลองออกเป็น 2 ส่วน คือการทดลองหาค่า α และ β ที่เหมาะสมโดยการกำหนดค่า และการทดลองหาค่าคำตอบที่ดีที่สุดโดยการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ ผลจากการทดลองหาค่า α และ β ที่เหมาะสมโดยการกำหนดค่า พบว่าคำตอบที่ได้มีประสิทธิภาพสูง คือผลคำตอบมีเปอร์เซ็นต์ความผิดพลาดอยู่ในช่วงที่ยอมรับได้ (MRE%<10%) เมื่อเทียบกับคำตอบที่ดีที่สุด (Optimal solution) และได้ผลคำตอบที่ดีที่สุดจำนวน 4 ชุดปัญหาซึ่งคิดเป็นร้อยละ 26 ของปัญหาทั้งหมดที่ทำการทดลอง โดยการทดลองหาคำตอบนั้นทำให้ทราบค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β) ที่มีความเหมาะสมของแต่ละชุดปัญหา ซึ่งผลการทดลองหาค่า α และ β ดังกล่าวสอดคล้องกับการกำหนดค่า α และ β ของการวิจัยอื่น ผลการทดลองนี้สามารถนำไปใช้สำหรับการวิจัยอื่น ๆ ที่มีการประยุกต์ใช้ขั้นตอนวิธีอาณานิคมมดในการแก้ปัญหาโดยเฉพาะปัญหาของ Lawrence ซึ่งจะเป็นการลดเวลาในการวิจัย และสามารถวิเคราะห์ความสัมพันธ์ของค่า α และ β ที่จะนำไปสู่คำตอบที่ดีที่สุด จากแนวโน้มของผลคำตอบที่เกิดจากการเปลี่ยนแปลงค่า α และ β

สำหรับการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ ให้ผลคำตอบอยู่ในช่วงที่ยอมรับได้ (MRE%<10%) คิดเป็นร้อยละ 60 ของปัญหาทั้งหมดที่ทำการทดลอง และให้ผลคำตอบเท่ากับคำตอบที่ดีที่สุดจำนวน 3 ปัญหา คือปัญหา la05, la10 และ la14 คิดเป็นร้อยละ 15 ของปัญหาทั้งหมดที่ทำการทดลอง เวลาที่ใช้ในการหาคำตอบอยู่ในช่วงที่ยอมรับได้และเหมาะกับการนำไปใช้แก้ปัญหาในระบบอุตสาหกรรมในปัจจุบัน แม้ว่าสามารถหาคำตอบได้ดีกว่าวิธีกำหนดค่า α และ β เพียง 3 ปัญหา แต่ใช้เวลาในการหาคำตอบน้อยกว่ามาก เพราะการทดลองโดยการกำหนดค่า α และ β นั้น มีจำนวนการทดลอง 48 ครั้ง (α และ β จำนวน 16 คู่ ทำการทดลองซ้ำ 3 ครั้ง) จึงจะได้ผลคำตอบของแต่ละปัญหา ซึ่งการทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติมีผลคำตอบที่มีประสิทธิภาพน้อยกว่าแบบกำหนดค่า α และ β เนื่องจากจำนวนมดและจำนวนรอบยังไม่เหมาะสมกับขนาดและความยากของแต่ละปัญหา

สำหรับการทดลองหาคำตอบในระบบการผลิตในบริษัทตัวอย่าง ผลคำตอบที่ได้มีค่าดีกว่าผลงานวิจัยที่เคยได้แก้ปัญหานี้มา ซึ่งลดเวลาในการดำเนินการทั้งหมดลงร้อยละ 16 นั่นคือถ้านำตารางการผลิตที่ได้จากการวิจัยนี้ไปปฏิบัติจริง และทำให้ประสิทธิภาพการผลิตของบริษัทตัวอย่างมากขึ้น

สรุปงานวิจัยนี้สามารถปรับปรุงข้อด้อยของวิธีการหาคำตอบด้วยขั้นตอนวิธีอาณานิคมมดในเรื่องของการหาค่า α และ β ที่เหมาะสมในแต่ละปัญหาได้อย่างมีประสิทธิภาพ ซึ่งจะทำให้การแก้ปัญหาการจัดตารางการผลิตแบบตามสั่งด้วยขั้นตอนวิธีอาณานิคมมดมีความยืดหยุ่น (Flexible) มากขึ้น สามารถใช้หาคำตอบในปัญหาการจัดตารางการผลิตแบบตามสั่งที่หลากหลายมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ข้อเสนอแนะ

ผู้วิจัยมีแนวคิดในการต้องพัฒนาโปรแกรมคอมพิวเตอร์การจัดตารางการผลิตแบบตามสั่งโดยใช้ขั้นตอนวิธีอาณานิคมมดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติให้มีประสิทธิภาพมากขึ้น ในเรื่องการกำหนดจำนวนมดและจำนวนรอบให้เหมาะสมกับขนาดและความยากของแต่ละปัญหาซึ่งสามารถแก้ไขได้ดังนี้

- สำหรับปัญหาที่มีจำนวนงานและเครื่องจักรมาก ควรให้ความสำคัญกับการเพิ่มจำนวนมด มากกว่าการเพิ่มจำนวนรอบเนื่องจากปัญหามีเส้นทางหลากหลายมาก เพราะจะทำให้การค้นหาเส้นทางในแต่ละรอบครอบคลุมมากขึ้น)

- สำหรับปัญหามีจำนวนการดำเนินการหลากหลาย ควรให้ความสำคัญกับการเพิ่มจำนวนรอบ มากกว่าการเพิ่มจำนวนมด เพราะจะทำให้จำนวนครั้งของการปรับปรุงค่าความเข้มข้นของฟีโรโมนเพิ่มขึ้น (Pheromone update) ทำให้คำตอบที่ได้ใกล้เคียงกับคำตอบที่ดีที่สุดมากที่สุด

อีกส่วนหนึ่งที่จะพัฒนาต่อไปคือ วิธีการการปรับเปลี่ยนค่าค่า α และ β อย่างอัตโนมัติ (ตามรูปที่ 3.5 และ 3.6) เนื่องจากในการวิจัยนี้เกิดปัญหาการปรับเปลี่ยนค่าแบบเฉพาะที่ (Local search) เพราะการขยายจากจุดกลางออกเป็น 4 จุดรอบข้าง ค่าคงที่การขยายเป็นค่าคงที่ซึ่งบางครั้งการค้นหาอาจจะตกอยู่ในพื้นที่ที่ให้ผลคำตอบดีแต่ไม่ใช่พื้นที่ที่ดีที่สุด ซึ่งอาจจะมีพื้นที่อื่นที่ให้ผลคำตอบดีกว่า แก้ไขปัญหานี้โดยการพัฒนาการขยายจากจุดกลางออกเป็น 4 จุดรอบข้าง ค่าการขยายดังกล่าวจะต้องขึ้นอยู่กับการเปลี่ยนแปลงของผลคำตอบ โดยถ้าความแตกต่างผลคำตอบในแต่ละรอบน้อย ให้มีการเพิ่มค่าการขยายมากขึ้น ซึ่งจะทำให้การค้นหาคำตอบไม่เป็นแบบเฉพาะที่

โดยในส่วนการนำไปประยุกต์ใช้งานจริงนั้น ควรมีการปรับปรุงให้เหมาะสมกับรูปแบบการผลิตของแต่ละโรงงานและความสะดวกของผู้ใช้ ซึ่งจากข้อจำกัดของโปรแกรมคอมพิวเตอร์ที่จัดทำขึ้นนั้นควรมีการพัฒนาและปรับปรุงดังต่อไปนี้

- การออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ให้สามารถแสดงผลของการจัดตารางการผลิตเป็นรูปแบบของแผนภูมิแกนต์ (Gantt Chart) เพื่อจะทำให้เกิดความสะดวกกับผู้ใช้เมื่อนำผลการหาคำตอบไปใช้งาน

- ออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ให้สามารถรองรับระบบการผลิตแบบตามสั่ง ซึ่งมีเครื่องจักรขนานได้ (Parallel Machine) เพราะจะทำให้โปรแกรมคอมพิวเตอร์นี้นำไปใช้งานได้อย่างหลากหลายมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ปัญจพร แพใหญ่. 2549. "การจัดตารางการผลิตแบบตามสั่งโดยใช้ระบบฮิวริสติกแบบผสมผสาน กรณีศึกษา : โรงงานผลิตเฟือง." วิทยานิพนธ์ วิศวกรรมศาสตร์มหาบัณฑิต สาขาวิชาวิศวกรรมมหาบัณฑิต ภาควิชาวิศวกรรมการผลิต สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

พิภพ ลลิตาภรณ์. 2545. **ระบบการวางแผนและควบคุมการผลิต.** กรุงเทพฯ : สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น).

ยงยุทธ ลิขิตพัฒนะกุล. 2546. "ระบบการจัดตารางการผลิตตามขีดจำกัดในการนผลิตแบบตามงาน : กรณีศึกษาการสร้างและซ่อมชิ้นส่วนทางเครื่องกล การไฟฟ้าฝ่ายผลิตแห่งประเทศไทย." วิทยานิพนธ์ วิศวกรรมมหาบัณฑิต สาขาวิชาวิศวกรรมการจัดการอุตสาหกรรม ภาควิชาอุตสาหกรรมอุตสาหการ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

วนิดา เลิศพิพัฒนานนท์. 2550. "การเปรียบเทียบการจัดสรรงานกำลังสองที่มเป้าหมายเป็นการหาค่าต่ำสุดสำหรับความแปรปรวนร่วมของสัมประสิทธิ์ค่าใช่จ่าย" 325-337. **การประชุมวิชาการด้านการวิจัยดำเนินงานแห่งชาติ ประจำปี 2550.**

วิภาวรรณ สิงห์พริ้ง. 2543. **การวิจัยการดำเนินการ.** กรุงเทพฯ : หนังสือในโครงการส่งเสริมการสร้างตำรา มหาวิทยาเทคโนโลยีพระจอมเกล้าธนบุรี.

วุฒิพงศ์ วุฒิจริยากุล. 2545. "โปรแกรมจัดลำดับการทำงาน." ปริญญานิพนธ์ คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมอุตสาหการ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

เศรษฐา เพชรอำไพ และ ธราธร กูลภัทรนิรันดร์ ม.ป.ก. "การประยุกต์วิธีเชิงพันธุกรรมในการจัดตารางการทำงาน." วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต คณะบริหารธุรกิจ สาขาวิชาเทคโนโลยีโลจิสติกส์ มหาวิทยาลัยเทคโนโลยีมหานคร.

อนิรุจน์ ตันเทิดทิตย์. 2546. "การศึกษาถึงวิธีการแก้ปัญหาการจัดเรียงลำดับงานที่เป็นแบบ Multiple Machine-Multiple Job โดยใช้วิธี Genetic Algorithm." วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธรรมศาสตร์.

เอกชาติ หัตถา. 2543. "การประยุกต์เจเนติกส์อัลกอริธึมส์สำหรับการจัดลำดับการผลิตชนิดตามงานใน กระบวนการเคมีแบบกะ." วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต ภาควิชาวิศวกรรมเคมี มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี.

Andrea, R. and Gino, D. 2007. "Flexible Job-shop Scheduling with Routing flexibility and Separable Setup Times Using Ant Colony Optimisation Method." 503–516. in **Robotics and Computer-Integrated Manufacturing.** Vol.23. Elsevier Science

Baker, K.R. 1974. **"Introduction to Sequencing and Scheduling."** John Wiley &Sons Inc. New York.

Cheng, T. and Feng, L. 2003. "A New Hybrid Heuristic Technique for Solving Job-shop Scheduling Problem." 53 – 58. in **IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing System** : Technology and Applications. Lviv.

Ching, L. and Fang, L. 1999. "A Tabu Search Algorithm for The Open Shop Scheduling Problem." **Computers & Operations Research.** Vol.26. Elsevier Science.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dorigo, M. 1992. "Optimization, Leaning and Natural Algorithms." Ph.D. Thesis. Dipartemento di Elettronica. Politecnico di Milano.

Dorigo, M. Maniezzo, V. and Colorni, A. 1996 "Ant System: Optimization by a Colony of Cooperating Agents." 29 – 41. in **IEEE transactions on systems, man, and cybernetics.** Vol 26.

Dorigo, M. and Gambarella, L.M. 1997. "Ant Colony for the Traveling Salesman Problem." 73-81. in **Accepted for publication in BioSystems.** Vol.43.

Burke, E. and Kendall, G. 1999. "Apply Ant Algorithms and the No Fit Polygon to the Nest Problem." 453-464 . **Proceeding of 12th Australian Joint Conference on Artificial Intelligent** . vol.1747.

Fan, K. and Zhang, R. 2010. "An Analysis of Research in Job Shop Scheduling Problem (2000-2009)." 282-288. in **IEEE international conference on advanced management science(ICAMS).**

Ferdinando, P. and Emanuela, M. 2000. "A Tabu Search Method Guided by Shifting Bottleneck for the Job Shop Scheduling Problem." 297-310. in **European Journal of Operational Research.** Elsevier Science.

Goncalves, J.F. ,Mendes, J. J. D. M. and Resende, M. G. C. 2005. "A Hybrid Genetic Algorithm for the Job Shop Scheduling Problem." 77-95.

Heinonen, J. and Pettersson, F. 2007 "Hybrid Ant Colony Optimization and Visibility Studies Applied to a Job-shop Scheduling Problem." 989-998. in **Applied Mathematics and Computation.** Elsevier Science.

Keqi Wang, Li Li. 2009. "Multi-objective Flexible Job Shop Schedule Based on Improved Ant Colony Algorithm." 1183-1187. in **International Conference on Information and Automation.** Zhuhai.

Lawrence, S. 1984. " Resource Constrained Project Scheduling: an Experimental Investigation of Heuristic Scheduling Techniques." (Supplement),Graduate School of Industrial Administration. Carnegie-Mellon University.

Mamalis, A G. and Malagardis, I. 1996. "Determination of Due Dates in Job Shop Scheduling by Simulated Annealing." 65-72. **Computer Integrated Manufacturing System.** Vol.9. Elsevier Science.

Nowicki, E. and Smutinicki, C. 1996. "A fast Taboo Search Algorithm for the Job-shop Problem." 97-813. **Management Science.** Vol.42.

Pimount, S. and Solnon C. 2000. "A Generic Ant Algorithm for Solving Constraint Satisfaction Problem." 100 – 108. **Second International Workshop on Ant Algorithm.**

Ramezanali, M. 2007. "A New Approach to Job Shop-Scheduling Problem." 79-86. **International Journal of Quality research.** Vol. 1.

Sabuncuoglu, I. and Bayiz, M. 1999. "Theory and Methodology Job Shop Scheduling with Beam Search." 390-412. **European Journal of Operational Research.** Vol.118.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Stutzle, T. and Hoos, H.H. 2000. "MAX-MIN ant system." 889-914. Future Generat. Comput. Syst.16

Socha, K. Knowles, J. and Sampels, M. 2002. "A MAX-MIN Ant System For University Course Timetabling Problem" 1-13. **Proceeding of the third International Workshop on Ant Algorithms**. Vol 2463.

Stützle, T. and Dorigo, M. 1999. "ACO Algorithms for the Traveling Salesman Problem." 163-183. **Evolutionary Algorithms in Engineering and Computer Science**. New York.

Udomsakdigool A. and Kachitvichyanukul V. 2008. "Multiple colony ant algorithm for job-shop scheduling problem" 4155 – 4175. **International Journal of Production Research**. Vol. 48.

Xie, H. and Huo, H. 2010. "Reseach on Job-Shop Scheduling Problem Based On Self-Adaptation Genetic Algorithm." 940-963. **IEEE.**

Zhong, Y. 2008. "Ant Colony Optimization with Local Search Applied to the Flexible Job Shop Scheduling Problems." 1015-1020. **IEEE.**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ปัญหาที่ใช้ในการทดลอง

ตารางที่ ก.1 รายละเอียดของปัญหามาตรฐาน Lawrence ใช้ในการทดลอง

ชื่อปัญหา	จำนวนงาน	จำนวนเครื่องจักร	จำนวนการดำเนินการ	คำตอบที่ดีที่สุด
LA01	10	5	50	666
LA02	10	5	50	655
LA03	10	5	50	597
LA04	10	5	50	590
LA05	15	5	75	593
LA06	15	5	75	926
LA07	15	5	75	890
LA08	15	5	75	863
LA09	15	5	75	951
LA10	15	5	75	958
LA11	20	5	100	1222
LA12	20	5	100	1039
LA13	20	5	100	1150
LA14	20	5	100	1292
LA15	20	5	100	1207
LA16	10	10	100	945
LA17	10	10	100	784
LA18	10	10	100	848
LA19	10	10	100	842
LA20	10	10	100	902

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางปัญหาที่ใช้ในการทดลองหาคำตอบ ของชุดปัญหามาตรฐาน Lawrence โดยเมตริกซ์ของปัญหานำมาจาก OR-Library

ตารางที่ ก.2 ปัญหา la01

Lawrence 10x5 instance
10 5
1 21 0 53 4 95 3 55 2 34
0 21 3 52 4 16 2 26 1 71
3 39 4 98 1 42 2 31 0 12
1 77 0 55 4 79 2 66 3 77
0 83 3 34 2 64 1 19 4 37
1 54 2 43 4 79 0 92 3 62
3 69 4 77 1 87 2 87 0 93
2 38 0 60 1 41 3 24 4 83
3 17 1 49 4 25 0 44 2 98
4 77 3 79 2 43 1 75 0 96

ตารางที่ ก.3 ปัญหา la02

Lawrence 10x5 instance
10 5
0 20 3 87 1 31 4 76 2 17
4 25 2 32 0 24 1 18 3 81
1 72 2 23 4 28 0 58 3 99
2 86 1 76 4 97 0 45 3 90
4 27 0 42 3 48 2 17 1 46
1 67 0 98 4 48 3 27 2 62
4 28 1 12 3 19 0 80 2 50
1 63 0 94 2 98 3 50 4 80
4 14 0 75 2 50 1 41 3 55
4 72 2 18 1 37 3 79 0 61

ตารางที่ ก.4 ปัญหา la03

Lawrence 10x5 instance
10 5
1 23 2 45 0 82 4 84 3 38
2 21 1 29 0 18 4 41 3 50
2 38 3 54 4 16 0 52 1 52
4 37 0 54 2 74 1 62 3 57
4 57 0 81 1 61 3 68 2 30
4 81 0 79 1 89 2 89 3 11
3 33 2 20 0 91 4 20 1 66
4 24 1 84 0 32 2 55 3 8
4 56 0 7 3 54 2 64 1 39
4 40 1 83 0 19 2 8 3 7

ตารางที่ ก.5 ปัญหา la04

Lawrence 10x5 instance
10 5
0 12 2 94 3 92 4 91 1 7
1 19 3 11 4 66 2 21 0 87
1 14 0 75 3 13 4 16 2 20
2 95 4 66 0 7 3 7 1 77
1 45 3 6 4 89 0 15 2 34
3 77 2 20 0 76 4 88 1 53
2 74 1 88 0 52 3 27 4 9
1 88 3 69 0 62 4 98 2 52
2 61 4 9 0 62 1 52 3 90
2 54 4 5 3 59 1 15 0 88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.6 ปัญหา la05

Lawrence 10x5 instance

10 5
1 72 0 87 4 95 2 66 3 60
4 5 3 35 0 48 2 39 1 54
1 46 3 20 2 21 0 97 4 55
0 59 3 19 4 46 1 34 2 37
4 23 2 73 3 25 1 24 0 28
3 28 0 45 4 5 1 78 2 83
0 53 3 71 1 37 4 29 2 12
4 12 2 87 3 33 1 55 0 38
2 49 3 83 1 40 0 48 4 7
2 65 3 17 0 90 4 27 1 23

ตารางที่ ก.7 ปัญหา la06

Lawrence 15x5 instance

15 5
1 21 2 34 4 95 0 53 3 55
3 52 4 16 1 71 2 26 0 21
2 31 0 12 1 42 3 39 4 98
3 77 1 77 4 79 0 55 2 66
4 37 3 34 2 64 1 19 0 83
2 43 1 54 0 92 3 62 4 79
0 93 3 69 1 87 4 77 2 87
0 60 1 41 2 38 4 83 3 24
2 98 3 17 4 25 0 44 1 49
0 96 4 77 3 79 1 75 2 43
4 28 2 35 0 95 3 76 1 7
0 61 4 10 2 95 1 9 3 35
4 59 3 16 1 91 2 59 0 46
4 43 1 52 0 28 2 27 3 50
0 87 1 45 2 39 4 9 3 41

ตารางที่ ก.8 ปัญหา la07

Lawrence 15x5 instance

15 5
0 47 4 57 1 71 3 96 2 14
0 75 1 60 4 22 3 79 2 65
3 32 0 33 2 69 1 31 4 58
0 44 1 34 4 51 3 58 2 47
3 29 1 44 0 62 2 17 4 8
1 15 2 40 0 97 4 38 3 66
2 58 1 39 0 57 4 20 3 50
2 57 3 32 4 87 0 63 1 21
4 56 0 84 2 90 1 85 3 61
4 15 0 20 1 67 3 30 2 70
4 84 0 82 1 23 2 45 3 38
3 50 2 21 0 18 4 41 1 29
4 16 1 52 0 52 2 38 3 54
4 37 0 54 3 57 2 74 1 62
4 57 1 61 0 81 2 30 3 68

ตารางที่ ก.9 ปัญหา la08

Lawrence 15x5 instance

15 5
3 92 2 94 0 12 4 91 1 7
2 21 1 19 0 87 3 11 4 66
1 14 3 13 0 75 4 16 2 20
2 95 4 66 0 7 1 77 3 7
2 34 4 89 3 6 1 45 0 15
4 88 3 77 2 20 1 53 0 76
4 9 3 27 0 52 1 88 2 74
3 69 2 52 0 62 1 88 4 98
3 90 0 62 4 9 2 61 1 52
4 5 2 54 3 59 0 88 1 15
0 41 1 50 4 78 3 53 2 23
0 38 4 72 2 91 3 68 1 71
0 45 3 95 4 52 2 25 1 6
3 30 1 66 0 23 4 36 2 17
2 95 0 71 3 76 1 8 4 88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.10 ปัญหา la09

Lawrence 15x5 instance

15 5

1 66 3 85 2 84 0 62 4 19

3 59 1 64 2 46 4 13 0 25

4 88 3 80 1 73 2 53 0 41

0 14 1 67 2 57 3 74 4 47

0 84 4 64 2 41 3 84 1 78

0 63 3 28 1 46 2 26 4 52

3 10 2 17 4 73 1 11 0 64

2 67 1 97 3 95 4 38 0 85

2 95 4 46 0 59 1 65 3 93

2 43 4 85 3 32 1 85 0 60

4 49 3 41 2 61 0 66 1 90

1 17 0 23 3 70 4 99 2 49

4 40 3 73 0 73 1 98 2 68

3 57 1 9 2 7 0 13 4 98

0 37 1 85 2 17 4 79 3 41

ตารางที่ ก.11 ปัญหา la10

Lawrence 15x5 instance

15 5

1 58 2 44 3 5 0 9 4 58

1 89 0 97 4 96 3 77 2 84

0 77 1 87 2 81 4 39 3 85

3 57 1 21 2 31 0 15 4 73

2 48 0 40 1 49 3 70 4 71

3 34 4 82 2 80 0 10 1 22

1 91 4 75 0 55 2 17 3 7

2 62 3 47 1 72 4 35 0 11

0 64 3 75 4 50 1 90 2 94

2 67 4 20 3 15 0 12 1 71

0 52 4 93 3 68 2 29 1 57

2 70 0 58 1 93 4 7 3 77

3 27 2 82 1 63 4 6 0 95

1 87 2 56 4 36 0 26 3 48

3 76 2 36 0 36 4 15 1 8

ตารางที่ ก.12 ปัญหา la11

Lawrence 20x5 instance

20 5

2 34 1 21 0 53 3 55 4 95

0 21 3 52 1 71 4 16 2 26

0 12 1 42 2 31 4 98 3 39

2 66 3 77 4 79 0 55 1 77

0 83 4 37 3 34 1 19 2 64

4 79 2 43 0 92 3 62 1 54

0 93 4 77 2 87 1 87 3 69

4 83 3 24 1 41 2 38 0 60

4 25 1 49 0 44 2 98 3 17

0 96 1 75 2 43 4 77 3 79

0 95 3 76 1 7 4 28 2 35

4 10 2 95 0 61 1 9 3 35

1 91 2 59 4 59 0 46 3 16

2 27 1 52 4 43 0 28 3 50

4 9 0 87 3 41 2 39 1 45

1 54 0 20 4 43 3 14 2 71

4 33 1 28 3 26 0 78 2 37

1 89 0 33 2 8 3 66 4 42

4 84 0 69 2 94 1 74 3 27

4 81 2 45 1 78 3 69 0 96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.13 ปัญหา la12

Lawrence 20x5 instance

20 5

1 23 0 82 4 84 2 45 3 38

3 50 4 41 1 29 0 18 2 21

4 16 3 54 1 52 2 38 0 52

1 62 3 57 4 37 2 74 0 54

3 68 1 61 2 30 0 81 4 57

1 89 2 89 3 11 0 79 4 81

1 66 0 91 3 33 4 20 2 20

3 8 4 24 2 55 0 32 1 84

0 7 2 64 1 39 4 56 3 54

0 19 4 40 3 7 2 8 1 83

0 63 2 64 3 91 4 40 1 6

1 42 3 61 4 15 2 98 0 74

1 80 0 26 3 75 4 6 2 87

2 39 4 22 0 75 3 24 1 44

1 15 3 79 4 8 0 12 2 20

3 26 2 43 0 80 4 22 1 61

2 62 1 36 0 63 3 96 4 40

1 33 3 18 0 22 4 5 2 10

2 64 4 64 0 89 1 96 3 95

2 18 4 23 3 15 1 38 0 8

ตารางที่ ก.14 ปัญหา la13

Lawrence 20x5 instance

20 5

3 60 0 87 1 72 4 95 2 66

1 54 0 48 2 39 3 35 4 5

3 20 1 46 0 97 2 21 4 55

2 37 0 59 3 19 1 34 4 46

2 73 3 25 1 24 0 28 4 23

1 78 3 28 2 83 0 45 4 5

3 71 1 37 2 12 4 29 0 53

4 12 3 33 1 55 2 87 0 38

0 48 1 40 2 49 3 83 4 7

0 90 4 27 2 65 3 17 1 23

0 62 3 85 1 66 2 84 4 19

3 59 2 46 4 13 1 64 0 25

2 53 1 73 3 80 4 88 0 41

2 57 4 47 0 14 1 67 3 74

2 41 4 64 3 84 1 78 0 84

4 52 3 28 2 26 0 63 1 46

1 11 0 64 3 10 4 73 2 17

4 38 3 95 0 85 1 97 2 67

3 93 1 65 2 95 0 59 4 46

0 60 1 85 2 43 4 85 3 32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.15 ปัญหา la14

Lawrence 20x5 instance

20 5

3 5 4 58 2 44 0 9 1 58

1 89 4 96 0 97 2 84 3 77

2 81 3 85 1 87 4 39 0 77

0 15 3 57 4 73 1 21 2 31

2 48 4 71 3 70 0 40 1 49

0 10 4 82 3 34 2 80 1 22

2 17 0 55 1 91 4 75 3 7

3 47 2 62 1 72 4 35 0 11

1 90 2 94 4 50 0 64 3 75

3 15 2 67 0 12 4 20 1 71

4 93 2 29 0 52 1 57 3 68

3 77 1 93 0 58 2 70 4 7

1 63 3 27 0 95 4 6 2 82

4 36 0 26 3 48 2 56 1 87

2 36 1 8 4 15 3 76 0 36

4 78 1 84 3 41 0 30 2 76

1 78 0 75 4 88 3 13 2 81

0 54 4 40 2 13 1 82 3 29

1 26 4 82 0 52 3 6 2 6

3 54 1 64 0 54 2 32 4 88

ตารางที่ ก.16 ปัญหา la15

Lawrence 20x5 instance

20 5

0 6 2 40 1 81 3 37 4 19

2 40 3 32 0 55 4 81 1 9

1 46 4 65 2 70 3 55 0 77

2 21 4 65 0 64 3 25 1 15

2 85 0 40 1 44 3 24 4 37

0 89 4 29 1 83 3 31 2 84

4 59 3 38 1 80 2 30 0 8

0 80 2 56 1 77 4 41 3 97

4 56 0 91 3 50 2 71 1 17

1 40 0 88 4 59 2 7 3 80

0 45 1 29 2 8 4 77 3 58

2 36 0 54 3 96 1 9 4 10

0 28 2 73 1 98 3 92 4 87

0 70 3 86 2 27 1 99 4 96

1 95 0 59 4 56 3 85 2 41

1 81 2 92 4 32 0 52 3 39

1 7 4 22 2 12 0 88 3 60

3 45 0 93 2 69 4 49 1 27

0 21 1 84 2 61 3 68 4 26

1 82 2 33 4 71 0 99 3 44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.17 ปัญหา la16

Lawrence 10x10 instance
10 10
1 21 6 71 9 16 8 52 7 26 2 34 0 53 4 21 3 55 5 95
4 55 2 31 5 98 9 79 0 12 7 66 1 42 8 77 6 77 3 39
3 34 2 64 8 62 1 19 4 92 9 79 7 43 6 54 0 83 5 37
1 87 3 69 2 87 7 38 8 24 9 83 6 41 0 93 5 77 4 60
2 98 0 44 5 25 6 75 7 43 1 49 4 96 9 77 3 17 8 79
2 35 3 76 5 28 9 10 4 61 6 9 0 95 8 35 1 7 7 95
3 16 2 59 0 46 1 91 9 43 8 50 6 52 5 59 4 28 7 27
1 45 0 87 3 41 4 20 6 54 9 43 8 14 5 9 2 39 7 71
4 33 2 37 8 66 5 33 3 26 7 8 1 28 6 89 9 42 0 78

ตารางที่ ก.18 ปัญหา la17

Lawrence 10x10 instance
10 10
4 18 7 21 9 41 2 45 3 38 8 50 5 84 6 29 1 23 0 82
8 57 5 16 1 52 7 74 2 38 3 54 6 62 9 37 4 54 0 52
2 30 4 79 3 68 1 61 8 11 6 89 7 89 0 81 9 81 5 57
0 91 8 8 3 33 7 55 5 20 2 20 4 32 6 84 1 66 9 24
9 40 0 7 4 19 8 7 6 83 2 64 5 56 3 54 7 8 1 39
3 91 2 64 5 40 0 63 7 98 4 74 8 61 1 6 6 42 9 15
1 80 7 39 8 24 3 75 4 75 5 6 6 44 0 26 2 87 9 22
1 15 7 43 2 20 0 12 8 26 6 61 3 79 9 22 5 8 4 80
2 62 3 96 4 22 9 5 0 63 6 33 7 10 8 18 1 36 5 40
1 96 0 89 5 64 3 95 9 23 7 18 8 15 2 64 6 38 4 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.19 ปัญหา la18

Lawrence 10x10 instance

10 10

6 54 0 87 4 48 3 60 7 39 8 35 1 72 5 95 2 66 9 5

3 20 9 46 6 34 5 55 0 97 8 19 4 59 2 21 7 37 1 46

4 45 1 24 8 28 0 28 7 83 6 78 5 23 3 25 9 5 2 73

9 12 1 37 4 38 3 71 8 33 2 12 6 55 0 53 7 87 5 29

3 83 2 49 6 23 9 27 7 65 0 48 4 90 5 7 1 40 8 17

1 66 4 25 0 62 2 84 9 13 6 64 7 46 8 59 5 19 3 85

1 73 3 80 0 41 2 53 9 47 7 57 8 74 4 14 6 67 5 88

5 64 3 84 6 46 1 78 0 84 7 26 8 28 9 52 2 41 4 63

1 11 0 64 7 67 4 85 3 10 5 73 9 38 8 95 6 97 2 17

ตารางที่ ก.20 ปัญหา la19

Lawrence 10x10 instance

10 10

2 44 3 5 5 58 4 97 0 9 7 84 8 77 9 96 1 58 6 89

4 15 7 31 1 87 8 57 0 77 3 85 2 81 5 39 9 73 6 21

9 82 6 22 4 10 3 70 1 49 0 40 8 34 2 48 7 80 5 71

1 91 2 17 7 62 5 75 8 47 4 11 3 7 6 72 9 35 0 55

6 71 1 90 3 75 0 64 2 94 8 15 4 12 7 67 9 20 5 50

7 70 5 93 8 77 2 29 4 58 6 93 3 68 1 57 9 7 0 52

6 87 1 63 4 26 5 6 2 82 3 27 7 56 8 48 9 36 0 95

0 36 5 15 8 41 9 78 3 76 6 84 4 30 7 76 2 36 1 8

5 88 2 81 3 13 6 82 4 54 7 13 8 29 9 40 1 78 0 75

9 88 4 54 6 64 7 32 0 52 2 6 8 54 5 82 3 6 1 26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.21 ปัญหา la20

Lawrence 10x10 instance

10 10

6 9 1 81 4 55 2 40 8 32 3 37 0 6 5 19 9 81 7 40

7 21 2 70 9 65 4 64 1 46 5 65 8 25 0 77 3 55 6 15

2 85 5 37 0 40 3 24 1 44 6 83 4 89 8 31 7 84 9 29

4 80 6 77 7 56 0 8 2 30 5 59 3 38 1 80 9 41 8 97

0 91 6 40 4 88 1 17 2 71 3 50 9 59 8 80 5 56 7 7

2 8 6 9 3 58 5 77 1 29 8 96 0 45 9 10 4 54 7 36

4 70 3 92 1 98 5 87 6 99 7 27 8 86 9 96 0 28 2 73

1 95 7 92 3 85 4 52 6 81 9 32 8 39 0 59 2 41 5 56

3 60 8 45 0 88 2 12 1 7 5 22 4 93 9 49 7 69 6 27

0 21 2 61 3 68 5 26 6 82 9 71 8 44 4 99 7 33 1 84

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
โปรแกรมคอมพิวเตอร์

การกำหนดค่าตัวแปรสำหรับการรับค่าเมตริกปัญหาและการสร้างเมตริกซ์และ list เริ่มต้นต่างๆ

การกำหนดค่าตัวแปรสำหรับการรับค่าต่างๆ การรับค่าจะประกอบด้วย การรับค่าเมตริกซ์ปัญหา รับค่าจำนวนมด จำนวนรอบ ชื่อของปัญหา ค่าอัตราการระเหย และอื่นๆ โดยมีการกำหนดค่าตัวแปรดังตารางที่ ข.1 และ ข.1

ตารางที่ ข.1 การกำหนดค่าตัวแปรสำหรับการรับค่าต่างๆ

ตัวแปร	ประเภทตัวแปร	สำหรับเก็บค่า
doublematrix_input	double[,]	เมตริกซ์ปัญหา
numAnt	int	จำนวนมด
Iteration	int	จำนวนรอบ
problem_name	string	ชื่อของชุดเมตริกซ์ปัญหา
q0	double	ค่าคงที่ q0
evapolate	double	ค่าอัตราการระเหย

ตารางที่ ข.2 การกำหนดค่าตัวแปรสำหรับการหาเส้นทาง

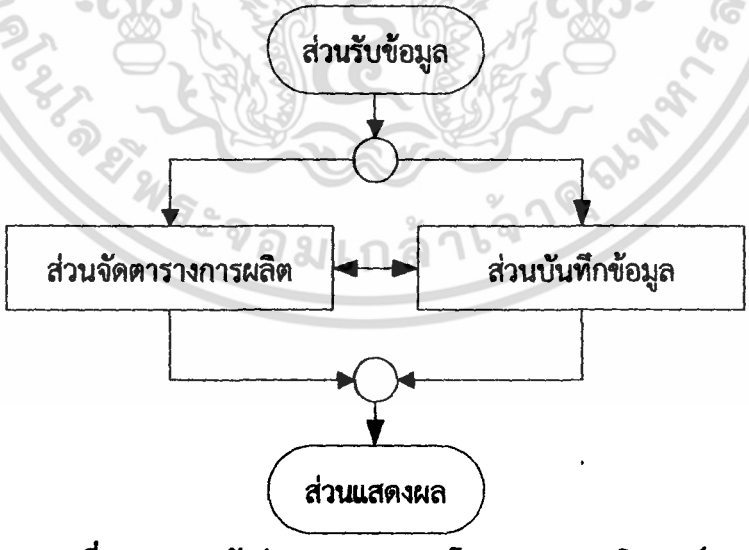
ตัวแปร	ประเภทตัวแปร	สำหรับเก็บค่า
num_job	int	จำนวนงาน
num_Machine	int	จำนวนเครื่องจักร
doublematrix_input_jagged	double[ ][ ]	เมตริกซ์ปัญหาแบบแยกแถว
num_Operat	int	จำนวนการดำเนินการ (จำนวนเครื่องจักร × จำนวนงาน)
pheromoneMatrix	double[,]	เมตริกซ์ความหนาแน่นของฟีโรโมน
Alpha	double	ค่าน้ำหนักของฟีโรโมน (α)
Beta	double	ค่าน้ำหนักของฮิวริสติค (β)
makespane	double	ค่าเวลาทั้งหมดในการดำเนินการ
criticalPath_List	List<int>	List ของ part วิกฤต
Visit	List<int>	List ของหมายเลขการดำเนินการหลังการจัดตารางการผลิต
Visit_Forward	List<int>	List ของหมายเลขการดำเนินการหลังการจัดตารางการผลิต จากคอลัมน์แรกไปท้ายคอลัมน์สุดท้ายของเมตริกซ์ปัญหา (Forward Ant)
Visit_Back	List<int>	List ของหมายเลขการดำเนินการหลังการจัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		ตารางการผลิต จากคอลัมน์สุดท้ายไปท้าย คอลัมน์แรกของเมตริกซ์ปัญหา (Backward Ant)
Process_time	double	เวลาในการดำเนินงาน
End_time	double	เวลาเสร็จ
st_time	double	เวลาเริ่ม
st_timejob	double	เวลาเริ่มของแต่ละงาน
st_timemachine	double	เวลาเริ่มของแต่ละเครื่องจักร
Eta_Matrix	double [,]	เมตริกซ์ค่าข้อมูลทาง ฮิวริสติคคิดจากเวลาการดำเนินการ
EtaWorkload	double [,]	เมตริกซ์ค่าข้อมูลทาง ฮิวริสติคคิดจากภาระงานบนแต่ละเครื่องจักร
Etatotal	double [,]	เมตริกซ์ค่าข้อมูลทาง ฮิวริสติคสุทธิ
Unvisit	int[ ][ ]	List ของหมายเลขการดำเนินการที่ไม่ได้จัดตารางการผลิต
alowlist	List<int>	List ของหมายเลขการดำเนินการที่มีโอกาสเป็นการดำเนินการถัดไป

การประยุกต์ใช้ขั้นตอนวิธีอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

โปรแกรมคอมพิวเตอร์ที่ได้จัดทำขึ้นนั้นมีส่วนประกอบทั้งหมด 4 ส่วนคือ ส่วนรับข้อมูล ส่วนจัดตารางการผลิต ส่วนบันทึกข้อมูลและส่วนแสดงผล โดยทั้ง 4 ส่วนมีลำดับการทำงานดังรูปที่ ข.1



รูปที่ ข.1 แผนผังส่วนประกอบของโปรแกรมคอมพิวเตอร์

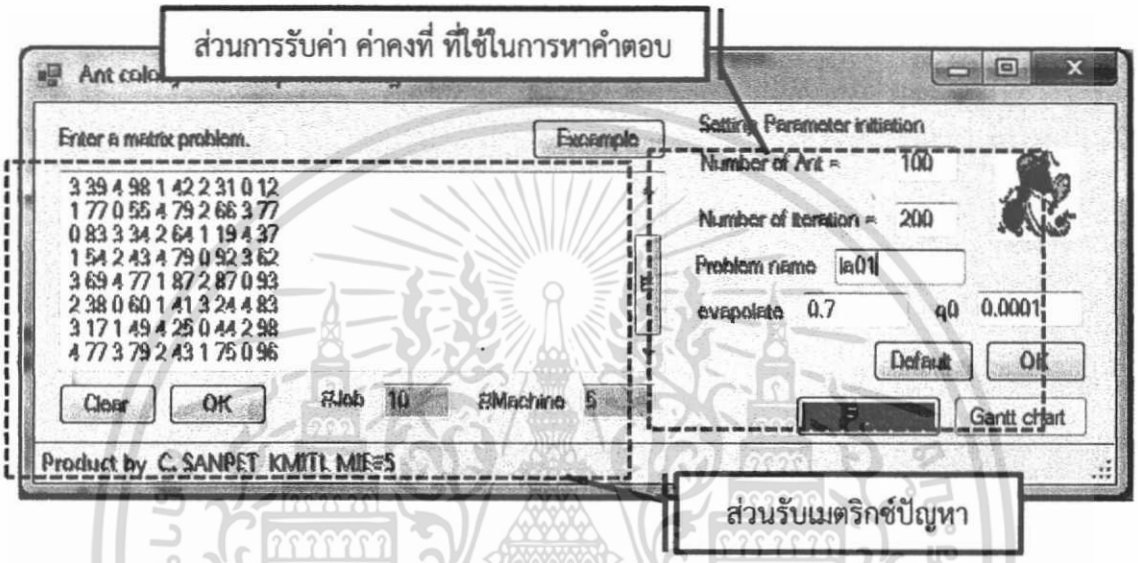
รูปที่ ข.1 แสดงถึงลำดับการทำงาน โดยเริ่มจากผู้ใช้ทำการใส่ค่าเมตริกซ์ปัญหาและกำหนดค่าคงที่ต่างๆ ซึ่งขั้นตอนนี้จะอยู่ในส่วนการรับข้อมูล หลังจากนั้นผู้ใช้สั่งให้โปรแกรมคอมพิวเตอร์เริ่มทำงาน โดยโปรแกรมคอมพิวเตอร์จะทำงานทั้งส่วนของการจัดตารางการผลิตไปพร้อมกับส่วนการบันทึกข้อมูล โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์จะมีการบันทึกข้อมูลต่างๆระหว่างการจัดตารางการผลิตไปจนการจัดตารางการผลิตเสร็จสิ้น หลังจากนั้นโปรแกรมคอมพิวเตอร์จะแสดงผลคำตอบต่างๆออกมาทางหน้าต่าง โดยผู้ใช้สามารถเรียกดูผลคำตอบโดยละเอียดได้จากไฟล์ที่ได้บันทึกไว้

การรับข้อมูล

ในส่วนของการรับข้อมูลของโปรแกรมคอมพิวเตอร์ที่จัดทำขึ้น เป็นขั้นตอนในการรับค่าต่างๆ รวมไปถึงการกำหนดค่าคงที่ที่ใช้ในการคำนวณหาคำตอบ โดยมีหน้าต่างในส่วนของการรับข้อมูลดังรูปที่ ข.2



รูปที่ ข.2 หน้าต่างส่วนของการรับข้อมูล

จากรูปที่ ข.2 แสดงหน้าต่างส่วนของการรับข้อมูลสามารถแบ่งออกได้เป็นสองส่วนคือรับเมตริกซ์ปัญหาและส่วนการรับค่าคงที่ที่ใช้หาคำตอบ โดยสามารถอธิบายได้ดังนี้

- **ส่วนรับค่าเมตริกซ์ปัญหา**

ในส่วนนี้เป็นส่วนที่ผู้ใช้จะต้องใส่ข้อมูลค่าเมตริกซ์ของชุดปัญหา โดยโปรแกรมคอมพิวเตอร์ที่จัดทำขึ้นนั้นได้รองรับการใส่ข้อมูลในรูปแบบของ OR – Library และรองรับการคัดลอกและวางของข้อมูล ซึ่งรูปแบบของเมตริกซ์ปัญหาของ OR – Library ผู้ใช้สามารถเรียกดูได้จากส่วนการเรียกดูตัวอย่างการรับค่าเมตริกซ์ปัญหา เมื่อผู้ใช้ใส่ข้อมูลเสร็จแล้ว ผู้ใช้คลิกที่ปุ่ม OK เพื่อให้โปรแกรมคอมพิวเตอร์รับค่าเมตริกซ์ปัญหา ซึ่งปุ่ม OK จะไม่สามารถคลิกได้ถ้ายังไม่มีการใส่ข้อมูลใดๆลงไปในช่องเมตริกซ์ปัญหา โดยถ้ารูปแบบหรือข้อมูลมีความผิดพลาดโปรแกรมคอมพิวเตอร์จะแสดงหน้าต่างแจ้งความผิดพลาด หลังจากนั้นโปรแกรมคอมพิวเตอร์จะให้ใส่ค่าใหม่อีกครั้ง เมื่อผ่านขั้นตอนการการรับค่าเมตริกซ์ปัญหาไปแล้ว โปรแกรมคอมพิวเตอร์จะแสดงจำนวนงานในช่อง #Job และแสดงจำนวนเครื่องจักรในช่องของ #Machine หลังจากขั้นตอนนี้ผู้ใช้จะเข้าสู่ขั้นตอนการกำหนดค่าคงที่ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ส่วนการรับค่าค่าคงที่ที่ใช้ในการหาคำตอบ**

ในขั้นตอนนี้ผู้ใช้ต้องกำหนดค่าคงที่ที่มีความสำคัญกับการหาคำตอบโดยขั้นตอนวิธีอาณานิคมมด ซึ่งได้กล่าวไว้ในบทที่ 2 และ 3 โดยค่าคงที่ในตารางที่ ข.3

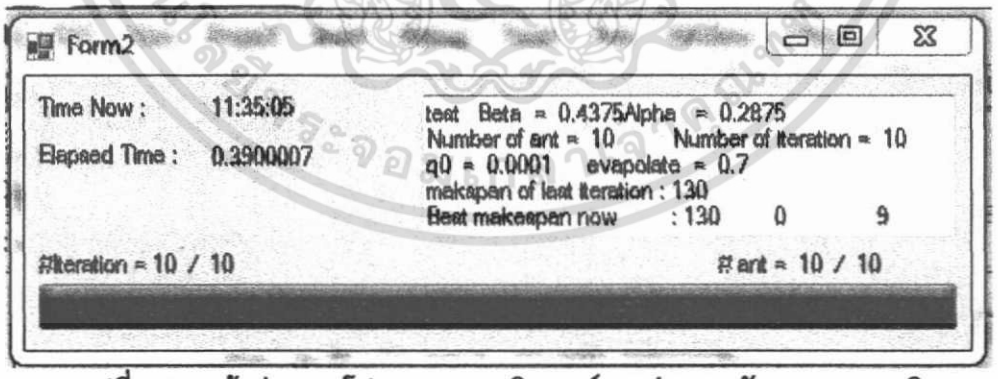
ตารางที่ ข.3 คำอธิบายความหมายของป้ายชื่อ (Label) ของส่วนการรับค่าค่าคงที่ที่ใช้ในการหาคำตอบ จากรูปที่ ข.2

ป้ายชื่อ (Label)	ค่าคงที่ที่ผู้ใช้ต้องกำหนด
Number of Ant =	จำนวนมด
Number of iteration =	จำนวนรอบการวนซ้ำ
Problem name	ชื่อของปัญหา (ใช้ในการบันทึกผล)
Evaporate	อัตราการระเหย
q0	พารามิเตอร์การสำรวจ

โดยค่าบางค่าจะมีการกำหนดค่าที่เหมาะสมไว้แล้ว ถ้าผู้ใช้ต้องการกลับมาใช้ค่าเดิมที่โปรแกรมคอมพิวเตอร์กำหนดไว้ ทำได้โดยผู้ใช้คลิกปุ่ม Default ด้านล่าง แต่ถ้าผู้ใช้กำหนดค่าต่างๆเรียบร้อยแล้ว ผู้ใช้จึงคลิกปุ่ม OK เพื่อให้โปรแกรมคอมพิวเตอร์เก็บค่าต่างๆไว้ หลังจากนั้นคลิกปุ่ม RUN เพื่อเริ่มการแก้ปัญหา ซึ่งปุ่ม OK และ ปุ่ม RUN จะไม่สามารถคลิกได้ถ้าผู้ใช้ยังข้อมูลใดๆไม่ครบถ้วน

การจัดตารางการผลิต

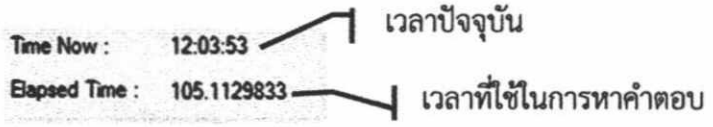
ขั้นตอนนี้เป็นขั้นตอนที่โปรแกรมคอมพิวเตอร์ทำการจัดตารางการผลิตโดยการประยุกต์ใช้ขั้นตอนวิธีอาณานิคมมด ซึ่งเป็นขั้นตอนที่ทำควบคู่ไปกับการบันทึกข้อมูล โดยในขั้นตอนนี้มีหน้าต่างแสดงการทำงานดังรูปที่ ข.3



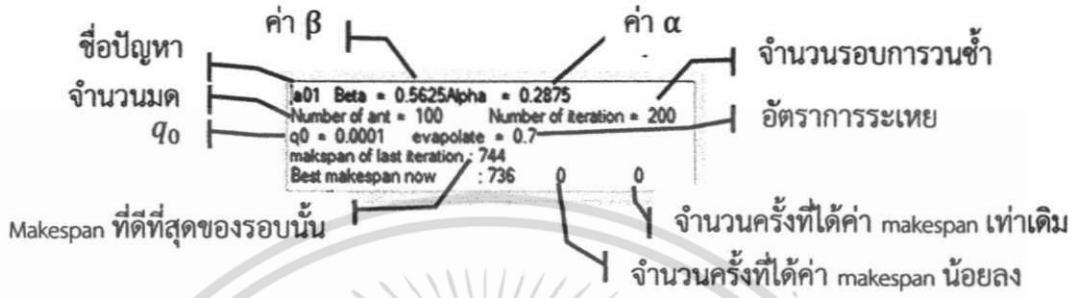
รูปที่ ข.3 หน้าต่างของโปรแกรมคอมพิวเตอร์ระหว่างการจัดตารางการผลิต

ข้อมูลต่างๆที่แสดงอยู่บนหน้าต่างในรูปที่ ข.3 จะมีการปรับเปลี่ยน (Update) ไปตามผลคำตอบที่โปรแกรมคอมพิวเตอร์หาได้ในขณะนั้น โดยสามารถอธิบายความหมายของค่าต่างๆที่มีการแสดงผลได้ดังรูปที่ ข.4

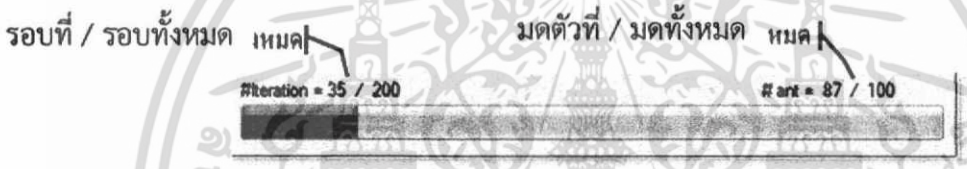
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) ส่วนแสดงเวลาระหว่างการหาคำตอบ



(ข) ส่วนแสดงผลลัพธ์ของการหาคำตอบ



(ค) ส่วนแสดงความคืบหน้าของการหาคำตอบ

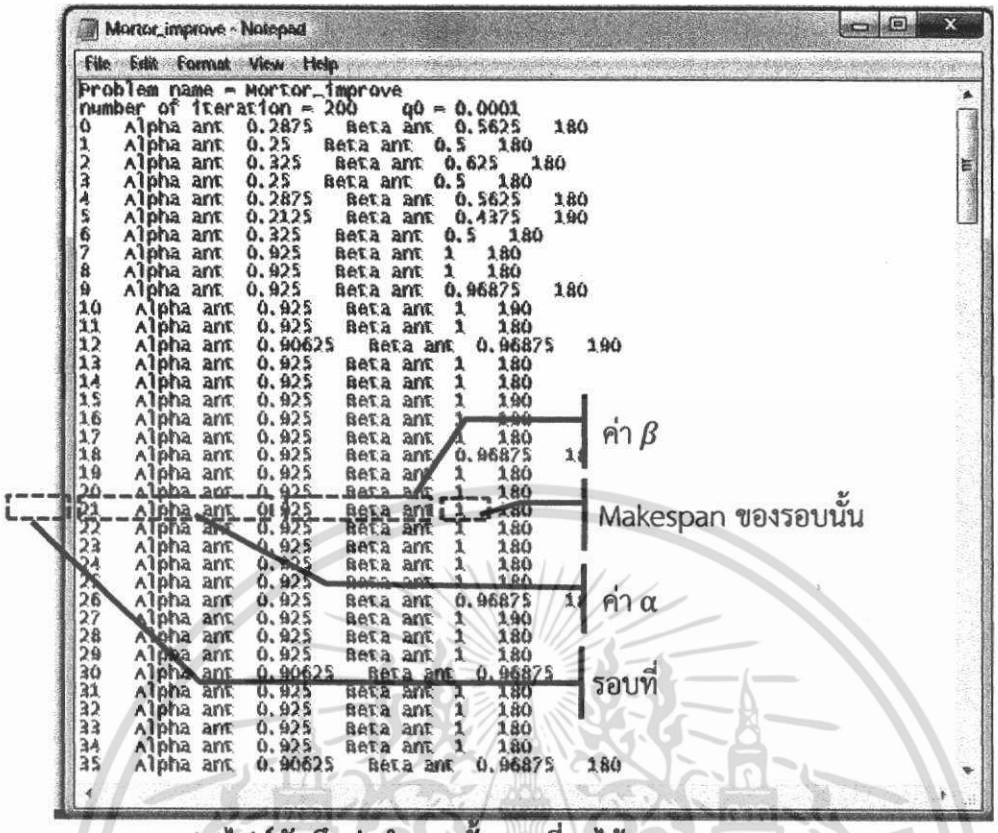
รูปที่ ข.4 คำอธิบายความหมายของค่าต่างๆที่มีการแสดงผล

การแสดงผลดังกล่าวจะทำให้ผู้ใช้สามารถรับรู้ความเคลื่อนไหวของโปรแกรมคอมพิวเตอร์ เพื่อใช้ในการตัดสินใจเกี่ยวกับการตั้งค่าคงที่ต่างๆของการหาคำตอบในครั้งต่อๆไป เพราะผู้ใช้จะทราบถึงอัตราของการลู่เข้าหาคำตอบว่าเร็วหรือช้า หรือไม่มีการลู่เข้าหาคำตอบ

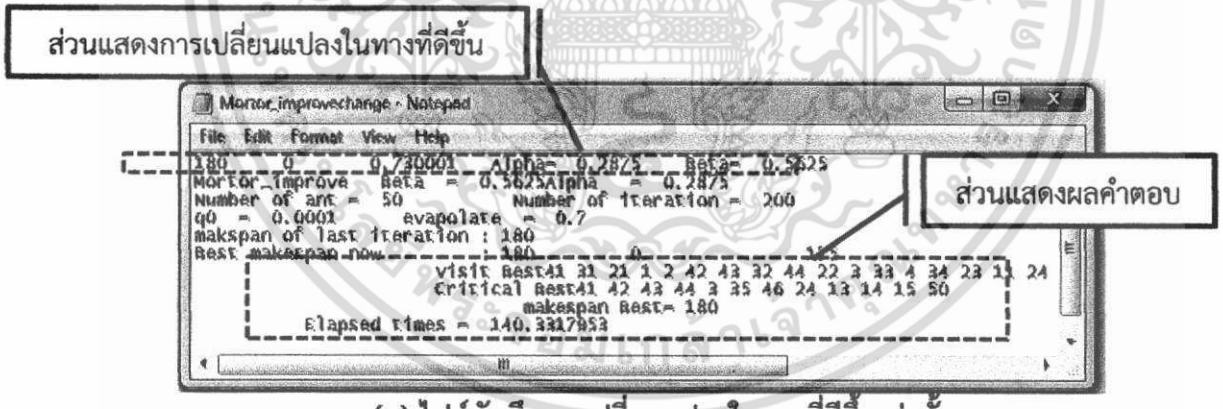
การบันทึกข้อมูลและการแสดงผล

ผู้วิจัยได้ออกแบบให้โปรแกรมคอมพิวเตอร์บันทึกข้อมูลและแสดงผลลัพธ์อยู่ในรูปแบบของแฟ้มข้อความไฟล์ Text Document (.txt) โดยผู้ใช้สามารถเรียกดูไฟล์ที่บันทึกได้หลังจากโปรแกรมคอมพิวเตอร์ได้ทำการหาคำตอบเสร็จแล้ว ซึ่งไฟล์ดังกล่าวมี 3 ไฟล์ ไฟล์แรกจะบันทึกค่าคำตอบทั้งหมดที่หาได้ของทุกๆรอบการวนซ้ำ โดยไฟล์ที่บันทึกใช้ชื่อเดียวกับชื่อปัญหาที่ผู้ใช้กำหนด ไฟล์ที่สองจะบันทึกการเปลี่ยนแปลงในทางที่ดีขึ้นเท่านั้น โดยใช้ชื่อในการบันทึกเดียวกับชื่อปัญหาที่ผู้ใช้กำหนดเพิ่มคำว่า "change" เข้าไป และสุดท้ายไฟล์ที่สามจะบันทึกคำตอบที่ดีที่สุดที่โปรแกรมคอมพิวเตอร์หาได้และค่าเวลาเริ่มเต้นและเวลาสิ้นสุดของแต่ละการดำเนิน ซึ่งสามารถนำไปเขียนเป็นแผนภูมิแกนต์และนำไปใช้ในงานจริงได้ โดยใช้ชื่อในการบันทึกเดียวกับชื่อปัญหาที่ผู้ใช้กำหนดเพิ่มคำว่า "Gant" เข้าไป โดยทั้งสามไฟล์ทำการบันทึกที่ D/: ซึ่งในแต่ละไฟล์มีลักษณะดังรูปที่ ข.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

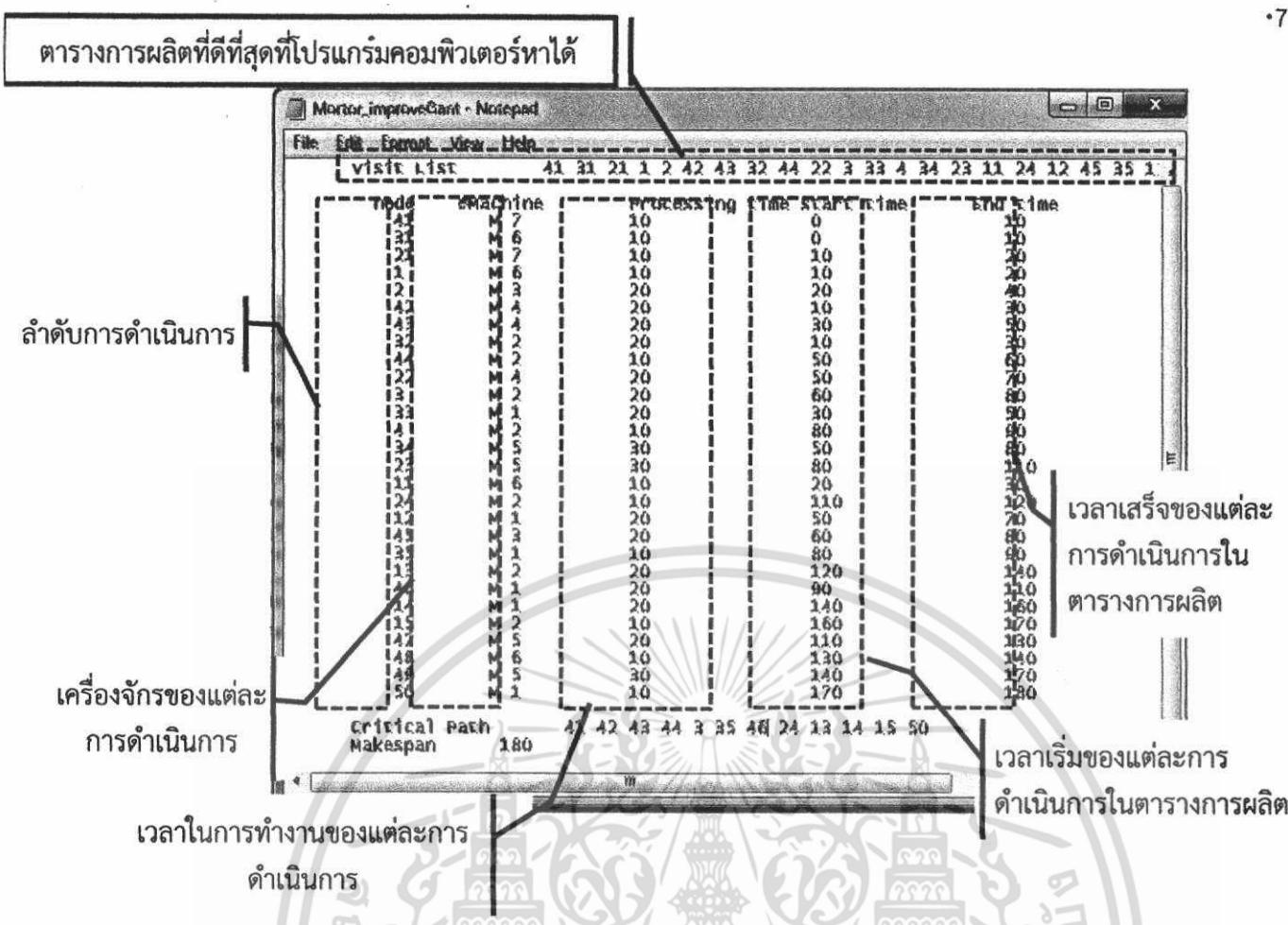


(ก)ไฟล์บันทึกค่าคำตอบทั้งหมดที่หาได้ของทุกๆรอบการวน



(ข) ไฟล์บันทึกการเปลี่ยนแปลงในทางที่ดีขึ้นเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ค) ไฟล์บันทึกคำตอบที่ดีที่สุดที่โปรแกรมคอมพิวเตอร์หาได้และค่าเวลาเริ่มเต้นและเวลาสิ้นสุดของแต่ละการดำเนินการ

รูปที่ ข.5 อธิบายไฟล์การบันทึกข้อมูลของการจัดตารางการผลิต

จากรูปที่ ข.5 แสดงไฟล์การบันทึกข้อมูลของการจัดตารางการผลิตโดยสามารถอธิบายแต่ละไฟล์ดังนี้ จากรูป ข.5 ก. คือไฟล์บันทึกค่าคำตอบทั้งหมดที่หาได้ของทุกๆรอบการวน ในไฟล์นี้ในส่วนบนแสดงค่าคงที่ต่างๆปัญหา ถัดลงมาจะแสดงค่า α และ β ที่ดีที่สุดที่หาได้ในแต่ละรอบ และค่า Makespan ของค่า α และ β นั้น

ไฟล์ต่อมาคือไฟล์บันทึกการเปลี่ยนแปลงในทางที่ดีขึ้นเท่านั้น ดังรูปที่ ข.5 ข. ในส่วนนี้จะแสดงผล 2 ส่วน คือ ส่วนที่ 1 ส่วนแสดงการเปลี่ยนแปลงในทางที่ดีขึ้น ซึ่งจะแสดงค่า Makespan รอบที่ และค่า α และ β เมื่อคำตอบจากโปรแกรมคอมพิวเตอร์ดีขึ้น และส่วนที่ 2 ส่วนแสดงผลคำตอบ ในส่วนนี้จะแสดงค่า ลำดับการทำงาน เส้นทางวิกฤต Makespan และค่าเวลาที่ในการหาคำตอบทั้งหมด มีหน่วยเป็นวินาที

ไฟล์สุดท้ายคือไฟล์บันทึกคำตอบที่ดีที่สุดที่โปรแกรมคอมพิวเตอร์หาได้และค่าเวลาเริ่มเต้นและเวลาสิ้นสุดของแต่ละการดำเนินการ ดังรูปที่ ข.5 ค ในไฟล์นี้จะแสดงลำดับการผลิตลำดับการดำเนินการ เวลาเริ่มและเวลาเสร็จของแต่ละการดำเนินการ ไฟล์นี้จะเป็นไฟล์ที่สามารถนำไปใช้ในการผลิตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างโปรแกรมคอมพิวเตอร์บางส่วนของการจัดตารางการผลิตด้วยโปรแกรมคอมพิวเตอร์ Visual C#

- **โปรแกรมคอมพิวเตอร์การทดลองหาค่าคำตอบที่ดีที่สุดโดยมีการปรับเปลี่ยนค่า α และ β อย่างอัตโนมัติ**

```
public void antwalking(int num_machine, int num_job,int num_Operat,
            double[] EtaArray, double Alpha, double Beta, double[,] T_sequence,double[] operation_process, int
            num_ant, int Iteration, string prob_name,double q0, double evaporation, double[,] doublematrix_input,int
            Max_operation)
{
      string str0 = "";
      string str_Gant;
      List<int> visitOld = new List<int>();
      List<int> Visit_update = new List<int>();
      List<int> Visit_update_Iter = new List<int>();
      List<int> criticalPath_listUpdate = new List<int>();
      List<int> criticalPath_listUpdate_Iter = new List<int>();
      int[][] jagged_operationtable = new int[num_job][];
      List<int> List_Operation = new List<int>();
      double AlphaUpdate = 0;
      double BetaUpdate = 0;
      double AlphaUpdate_ANT = 0;
      double BetaUpdate_ANT = 0;
      double AlphaUpdate_Iter = 0;
      double BetaUpdate_Iter = 0;
      double makspan_update = 0;
      double makspan_update_Iter = 0;
      double makespan_Old = 0;
      double makespan_New = 0;
      double Alpha_temp = 0;
      double Beta_temp = 0;
      int count_change = 0;
      int count_same = 0;
      double[,] pheromoneMatrix = new double[(num_job * ax_operation) + 2, (num_job * Max_operation) + 1];
      List<double> AlphaList1 = new List<double>();
      List<double> BetaList1 = new List<double>();
       AlphaList1.Add(0.25);
       AlphaList1.Add(0.2125);
       AlphaList1.Add(0.2125);
       AlphaList1.Add(0.2875);
```

```
AlphaList1.Add(0.2875);

BetaList1.Add(0.5);
BetaList1.Add(0.4375);
BetaList1.Add(0.5625);
BetaList1.Add(0.4375);
BetaList1.Add(0.5625);

List<double> AlphaList2 = new List<double>();
List<double> BetaList2 = new List<double>();
AlphaList2.Add(0);
AlphaList2.Add(-0.0375);
AlphaList2.Add(-0.0375);
AlphaList2.Add(0.0375);
AlphaList2.Add(0.0375);

BetaList2.Add(0);
BetaList2.Add(-0.0625);
BetaList2.Add(0.0625);
BetaList2.Add(-0.0625);
BetaList2.Add(0.0625);

TextWriter writeText_file = new StreamWriter(@"D:\" + prob_name + ".txt");                //print to file .text
TextWriter writeText_file2 = new StreamWriter(@"D:\" + prob_name +"change"+ ".txt");      //print to file .text
TextWriter writeText_file3 = new StreamWriter(@"D:\" + prob_name + "Gant" + ".txt");      //print to file .text
DateTime time = new DateTime(DateTime.Now.Ticks);                                          //set start time now
progressBar2.Maximum = Iteration;                                          //set parameter on progress bar
progressBar2.Minimum = 0;
progressBar2.Value = 0;

writeText_file.WriteLine("Problem name = " + prob_name);
writeText_file.WriteLine("number of iteration = " + Iteration + "    q0 = " + q0);

jagged_operationtable =CREATE_OperationTableJagged(num_job, Max_operation, doublematrix_input);
List_Operation = CREATE_List_operation(num_job, jagged_operationtable);

pheromoneMatrix = Create_pheromoneMatrix(num_job, num_machine,Max_operation);      //create pheromone matrix

labelTimeNow.Text = DateTime.Now.ToLongTimeString();                               //show time now
```

```
labelTimeNow.Update();
labelElapsed.Text =
DateTime.Now.Subtract(time).TotalSeconds.ToString();                    //show elapsed time
labelElapsed.Update();
label5.Update();                                                        //show label in for loop
label6.Update();
double Alpha_Old=0.250;
double Beta_Old=0.500;

 for (int iter = 0; iter < Iteration; iter++) //loop Iteration
     {
         for (int ant = 0; ant < num_ant; ant++)  //loop #ant
         {
             double makespane_indexBest = 0;
             double makespane = 0;
             List<int> criticalPath_List = new List<int>();
             List<int> Visit = new List<int>();
             List<int> critical_indexBest = new List<int>();
             List<int> Visit_indexBest = new List<int>();

             for (int index = 0; index < 5; index++)
             {
                 if (iter == 0)
                 {
                     Alpha = AlphaList1[index];
                     Beta = BetaList1[index];
                 }
                 else if(Alpha_temp==Alpha_Old&&Beta_temp==Beta_Old)      //if alpha new = alpha old
                 {
                     Alpha = Alpha_temp+(AlphaList2[index]/2);
                     Beta = Beta_temp + (BetaList2[index]/2);
                 }
                 else if(Alpha_temp!=Alpha_Old&&Beta_temp!=Beta_Old)      //if alpha new != alpha old
                 {
                     if (makespan_New < makespan_Old)
                     {
                         double decrease = Math.Abs(makespan_New - makespan_Old) / makspan_update;
                         if (decrease >= 0.10)
                         {
```

```
        int sgnAlpha = Math.Sign(Alpha_temp - Alpha_Old);
        int sgnBeta = Math.Sign(Beta_temp - Beta_Old);
        double Alpha_center = Alpha_temp + (sgnAlpha * (0.0375 * 3));
        double Beta_center = Beta_temp + (sgnBeta * (0.0625 * 3));
        if (index == 0)
        {
            Alpha = Alpha_center;
            Beta = Beta_center;
        }
        else
        {
            Alpha = Alpha_center + (AlphaList2[index]);
            Beta = Beta_center + (BetaList2[index]);
        }
        Alpha_temp = Alpha_center;
        Beta_temp = Beta_center;
    }
    else if (decrease >= 0.05)
    {
        int sgnAlpha = Math.Sign(Alpha_temp - Alpha_Old);
        int sgnBeta = Math.Sign(Beta_temp - Beta_Old);
        double Alpha_center = Alpha_temp + (sgnAlpha * (0.0375 * 2));
        double Beta_center = Beta_temp + (sgnBeta * (0.0625 * 2));
        if (index == 0)
        {
            Alpha = Alpha_center;
            Beta = Beta_center;
        }
        else
        {
            Alpha = Alpha_center + (AlphaList2[index]);
            Beta = Beta_center + (BetaList2[index]);
        }
        Alpha_temp = Alpha_center;
        Beta_temp = Beta_center;
    }
    else
    {
        Alpha = Alpha_temp + AlphaList2[index];
```

```
                Beta = Beta_temp + BetaList2[index];
            }
        }
        else
        {
            Alpha = Alpha_Old+ AlphaList2[index];
            Beta = Beta_Old + BetaList2[index]; ;
        }
    }
    if (Alpha <= 0)
    {
        Alpha = 0.025;
    }
    if (Beta <= 0)
    {
        Beta = 0;
    }
    if (Alpha >= 0.925)
    {
        Alpha = 0.925;
    }
    if(Beta>=1)
    {
        Beta = 1;
    }
    ////////////////////
    // ANT ROUNTING //
    ////////////////////
    Object[] From_Ant_Rount = ANT_ROUNT(Alpha, Beta, num_machine, num_job, EtaArray,
             jagged_operationtable, num_Operat,pheromoneMatrix, q0, T_sequence,
             operation_process, doublematrix_input,List_Operation,prob_name,Max_operation);

    makespane = (double)From_Ant_Rount[0];
    criticalPath_List = (List<int>)From_Ant_Rount[1];
    Visit = (List<int>)From_Ant_Rount[2];

    ////////////////////////////////////////////////////////////////////////
    if (index == 0)
    {
```

```
            makespane_indexBest = makespane;
            List<int> cri_indexOld = new List<int>(criticalPath_List);
            List<int> V_indexOld = new List<int>(Visit);
            critical_indexBest = cri_indexOld;
            Visit_indexBest = V_indexOld;
            AlphaUpdate = Alpha;
            BetaUpdate = Beta;
        }
        else
        {
            if (makespane_indexBest > makespane)
            {
                makespane_indexBest = makespane;
                List<int> cri_indexOld = new List<int>(criticalPath_List);
                List<int> V_indexOld = new List<int>(Visit);
                critical_indexBest = cri_indexOld;
                Visit_indexBest = V_indexOld;
                AlphaUpdate = Alpha;
                BetaUpdate = Beta;
            }
        }
    }
    ////////////////////////////////////////////////////////////////////////////
    if (ant == 0)                                          //correct makespan best of ant
    {
        List<int> paralist = new List<int>(critical_indexBest);
        List<int> paralist2 = new List<int>(Visit_indexBest);
        makspan_update = makespane_indexBest;
        criticalPath_listUpdate = paralist;                // correct new critical path
        Visit_update = paralist2;
        AlphaUpdate_ANT = AlphaUpdate;
        BetaUpdate_ANT = BetaUpdate;
    }
    else
    {
        if (makespane_indexBest < makspan_update)
        {
            List<int> paralist = new List<int>(critical_indexBest);
            List<int> paralist2 = new List<int>(Visit_indexBest);
```

```
                        makspan_update = makespane_indexBest;
                        criticalPath_listUpdate = paralist;                    // correct new critical path
                        Visit_update = paralist2;
                        AlphaUpdate_ANT = AlphaUpdate;
                        BetaUpdate_ANT = BetaUpdate;
                    }
                }
                criticalPath_List.Clear();
                Visit.Clear();

                label3.Text = "# ant =  " + (ant + 1) + "  /  " + num_ant;
                label3.Update();
                labelTimeNow.Text = DateTime.Now.ToLongTimeString();        //show time now
                labelTimeNow.Update();
                labelElapsed.Text = DateTime.Now.Subtract(time).TotalSeconds.ToString();  //show elapsed time
                labelElapsed.Update();
            }

            if (iter == 0)                                  // if first Iteration
            {
                makespan_Old = makspan_update + 1;
                makespan_New = makspan_update;
                Alpha_Old = 0.250;
                Beta_Old = 0.500;
            }
            else
            {
                makespan_Old = makespan_New;
                makespan_New = makspan_update;
                Alpha_Old = Alpha_temp;
                Beta_Old = Beta_temp;
            }
            Alpha_temp = AlphaUpdate_ANT;
            Beta_temp = BetaUpdate_ANT;
            writeText_file.WriteLine(iter + "   Alpha ant  " + AlphaUpdate_ANT + "   Beta ant  " +
BetaUpdate_ANT + "   " + makspan_update);

            ////////////////////////////
            //Local update phreromone//
```

```
                ////////////////////////////
                pheromoneMatrix = Update2_pheromoneMatrix(num_job, num_machine, pheromoneMatrix,
criticalPath_listUpdate, evaporation, num_ant, makspan_update,Max_operation);
                if (iter == 0)
                {
                    List<int> paralist3 = new List<int>(criticalPath_listUpdate);
                    List<int> paralist4 = new List<int>(Visit_update);
                    makspan_update_Iter = makspan_update;
                    criticalPath_listUpdate_Iter = paralist3;
                    Visit_update_Iter = paralist4;
                    AlphaUpdate_Iter = AlphaUpdate_ANT;
                    BetaUpdate_Iter = BetaUpdate_ANT;
                    string sec = DateTime.Now.Subtract(time).TotalSeconds.ToString();
                    writeText_file2.WriteLine(makspan_update_Iter + "\t" + iter + "\t" + sec+"   Alpha=  "+
AlphaUpdate_Iter+"    Beta=  "+ BetaUpdate_Iter);
                }
                else
                {
                    if (makspan_update < makspan_update_Iter)
                    {
                        List<int> paralist3 = new List<int>(criticalPath_listUpdate);
                        List<int> paralist4 = new List<int>(Visit_update);
                        makspan_update_Iter = makspan_update;
                        criticalPath_listUpdate_Iter = paralist3;
                        Visit_update_Iter = paralist4;
                        count_change = count_change + 1;

                        AlphaUpdate_Iter = AlphaUpdate_ANT;
                        BetaUpdate_Iter = BetaUpdate_ANT;
                        string sec = DateTime.Now.Subtract(time).TotalSeconds.ToString();
                        writeText_file2.WriteLine(makspan_update_Iter + "\t" + iter  + "\t" + sec + "   Alpha=  " +
AlphaUpdate_Iter + "    Beta=  " + BetaUpdate_Iter);
                    }
                    else if (makspan_update == makspan_update_Iter)
                    {
                        count_same = count_same + 1;
                    }
                }
                ////////////////////////////
```

```
                //Grobal update pheromone//
                ////////////////////////////
                pheromoneMatrix = Update3_pheromoneMatrix(num_job, num_machine, pheromoneMatrix,
criticalPath_listUpdate_Iter, evaporation, num_ant,Max_operation);
                criticalPath_listUpdate.Clear();
                Visit_update.Clear();

                ////////////////////////////
                //Show data on textbox   //
                ////////////////////////////
                str0 = "";
                str0 = prob_name+"   Beta  =  " + BetaUpdate_Iter + "Alpha   =  " + AlphaUpdate_Iter +
Environment.NewLine;
                str0 += "Number of ant =  " + num_ant + "          Number of iteration =  " + Iteration +
Environment.NewLine;
                str0 += "q0  =  " + q0 + "      evapolate  =  " + evaporation + Environment.NewLine;
                str0 += "makspan of last iteration : " + makspan_update + Environment.NewLine;
                str0 += "Best makespan now         : " + makspan_update_Iter + "          " + count_change + "
" + count_same;

                textBox1.Text = str0;
                textBox1.Update();

                label4.Text = "#Iteration = " + (iter + 1) + "  /  " + Iteration;   //show # this Iteration
                label4.Update();
                progressBar2.Value = iter + 1;                                     //show progress bar #Iteration

            }
            string secend = DateTime.Now.Subtract(time).TotalSeconds.ToString();
            writeText_file.WriteLine(str0);
            writeText_file2.WriteLine(str0);
            writeText_file2.WriteLine("                              " + F1.convertlistIntTostring(Visit_update_Iter, "Visit
Best"));
            writeText_file2.WriteLine("                                " +
F1.convertlistIntTostring(criticalPath_listUpdate_Iter, "Critical Best"));
            writeText_file2.WriteLine("                                        makespan Best= " + makspan_update_Iter);
            writeText_file2.WriteLine("          Elapsed times =  " + secend);

            writeText_file3.WriteLine("      Visit List    " + "\t" + convertlistIntTostring(Visit_update_Iter));
```

```
            str_Gant = Gant_string(Visit_update_Iter, num_machine, num_job,
T_sequence,operation_process,Max_operation);

            writeText_file3.WriteLine();
            writeText_file3.WriteLine(str_Gant);
            writeText_file3.WriteLine();
            writeText_file3.WriteLine("      Critical Path       " +
convertlistIntTostring(criticalPath_listUpdate_Iter));
            writeText_file3.WriteLine("      Makespan      "+makspan_update_Iter);
            writeText_file.Close();
            writeText_file2.Close();
            writeText_file3.Close();

        }
```

- **การปรับปรุงค่าฟีโรโมนแบบแคบ (Local pheromone update)**

```
        private double[,] Update2_pheromoneMatrix(int num_job, int num_machine, double[,] pheroMatirx_Old,
            List<int> critical_list1,double evaporation, int num_ant,double makspan2,int Max_operate)
        {
            double[,] pheroMatrix_New = new double[(num_job * Max_operate) + 2, (num_job * Max_operate)+1];

            Array.Copy(pheroMatirx_Old, pheroMatrix_New, pheroMatirx_Old.Length);
            double pheromone_min = 0.0001;
            List<int> invert_critcalList1 = new List<int>();
            List<int> templist1 = new List<int>(critical_list1);    //List tempolary of critical path List (copy
List)
            templist1.Add(num_job * Max_operate + 1);               //Add last node

            for (int x = 1; x <= num_job * Max_operate+1; x++)
            {
                invert_critcalList1.Add(x);                         //Create invert critical path list
            }
            int From = 0;
            foreach (var item in templist1)
            {
                List<int> TempOper_To_list = new List<int>();
                for (int o = 1; o <= num_job * Max_operate+1; o++)
```

```
                {
                    TempOper_To_list.Add(o);                        //List tempolary of all operation + final node
                }
                pheroMatrix_New[From, item - 1] = (pheroMatirx_Old[From, item - 1]) + (10.00 / makspan2);  //update
node in criticalpath list

                TempOper_To_list.Remove(item);
                List<int> Temp_remove = new List<int>(TempOper_To_list); //copy List
                foreach (var item_no in Temp_remove)
                {
                    pheroMatrix_New[From, item_no - 1] = pheroMatirx_Old[From, item_no - 1] - (10.00 / makspan2);

                    if (pheroMatrix_New[From, item_no - 1] < pheromone_min) //don't update node is not criticalpath
list
                    {
                        pheroMatrix_New[From, item_no - 1] = pheromone_min;
                    }
                }
                From = item;
                invert_critcalList1.Remove(item);   //add node is not criticalpath list to invert criticalpath list
            }
            int last_in_list = templist1[templist1.Count - 1];
            invert_critcalList1.Add(last_in_list);
            List<int> All_node = new List<int>();
            for (int o1 = 1; o1 <= num_job *Max_operate+1; o1++)
            {
                All_node.Add(o1);
            }
            foreach (var From2 in invert_critcalList1)
            {
                foreach (var item_All in All_node)
                {
                    pheroMatrix_New[From2, item_All - 1] = pheroMatirx_Old[From2, item_All - 1] - (10.00 /
makspan2);

                    if (pheroMatrix_New[From2, item_All - 1] < pheromone_min)
                    {
                        pheroMatrix_New[From2, item_All - 1] = pheromone_min;
                    }
```

```
            }
        }
        return pheroMatrix_New;
    }
```

- **การปรับค่าแบบฟีโรโมนแบบวงกว้าง (Global pheromone update)**

```
    private double[,] Update3_pheromoneMatrix(int num_job, int num_machine, double[,] pheroMatirx_Old, List<int>
critical_list,
        double evaporation, int num_ant,int Max_operation)
    {
        double[,] pheroMatrix_New = new double[(num_job * Max_operation) + 2, (num_job * Max_operation) + 1];

        Array.Copy(pheroMatirx_Old, pheroMatrix_New, pheroMatirx_Old.Length);
        double pheromone_min = 0.0001;
        List<int> invert_critcalList1 = new List<int>();
        List<int> templist2 = new List<int>(critical_list);
        templist2.Add(num_job * Max_operation + 1);                    //Add last node

        for (int x = 1; x <= num_job * Max_operation + 1; x++)
        {
            invert_critcalList1.Add(x);
        }
        int From = 0;
        foreach (var item in templist2)
        {
            List<int> TempOper_To_list = new List<int>();
            for (int o = 1; o <= num_job * Max_operation + 1; o++)
            {
                TempOper_To_list.Add(o);
            }
            pheroMatrix_New[From, item - 1] = pheroMatirx_Old[From, item - 1]+0.3;
            TempOper_To_list.Remove(item);
            List<int> Temp_remove = new List<int>(TempOper_To_list);
            foreach (var item_no in Temp_remove)
            {
                pheroMatrix_New[From, item_no - 1] = ((1 - evaporation) * pheroMatirx_Old[From, item_no - 1]);
```

```
                if (pheroMatrix_New[From, item_no - 1] < pheromone_min)
                {
                    pheroMatrix_New[From, item_no - 1] = pheromone_min;
                }
            }
            From = item;
            invert_critcalList1.Remove(item);
        }
        int last_in_list = templist2[templist2.Count - 1];
        invert_critcalList1.Add(last_in_list);
        List<int> All_node = new List<int>();
        for (int o1 = 1; o1 <= num_job * Max_operation + 1; o1++)
        {
            All_node.Add(o1);
        }
        foreach (var From2 in invert_critcalList1)
        {
            foreach (var item_All in All_node)
            {
                pheroMatrix_New[From2, item_All - 1] = ((1 - evaporation) * pheroMatirx_Old[From2, item_All -
1]);

                if (pheroMatrix_New[From2, item_All - 1] < pheromone_min)
                {
                    pheroMatrix_New[From2, item_All - 1] = pheromone_min;
                }
            }
        }
        return pheroMatrix_New;
    }
```

- **การปรับปรุงคุณภาพคำตอบ (Local improvement)**

```
    private List<int> Local_Improve(List<int> critical_PathList, int num_machine,
        int num_job, double[,] T_sequence, List<int> VisitOld,
        int[][] jagged_operation_table,List<int> List_oper,int Max_operate)
     {
```

```
int block = 1;
int count = 1;
Hashtable member_block = new Hashtable(); ;
Hashtable number_inBlock = new Hashtable();
bool changeBlock = false;
List<int> member = new List<int>();
List<int> Critical_Local = new List<int>();
List<int> Visit_Local_Improve = new List<int>(VisitOld);
int machineOld = 0;
foreach (int node in critical_PathList)
 {
    int job;
    int machine;

    job = findJobnumber(node, Max_operate);
    machine = Convert.ToInt32(findMachinenumber(T_sequence, job, node, Max_operate));
    if (count == 1)
    {
        member.Add(node);
        machineOld = machine;
        count = count+1;
    }
    else
    {
        if (machine == machineOld)
        {
            member.Add(node);
            machineOld = machine;
            count = count + 1;
        }
        else if (machine != machineOld)
        {
            List<int> memberCopy = new List<int>(member);
            number_inBlock.Add(block, memberCopy.Count);          //correct number in each block
            member_block.Add(block, memberCopy);                  //correct list member in to hastable
            member.Clear();
            changeBlock = true;
            block = block + 1;
        }
```

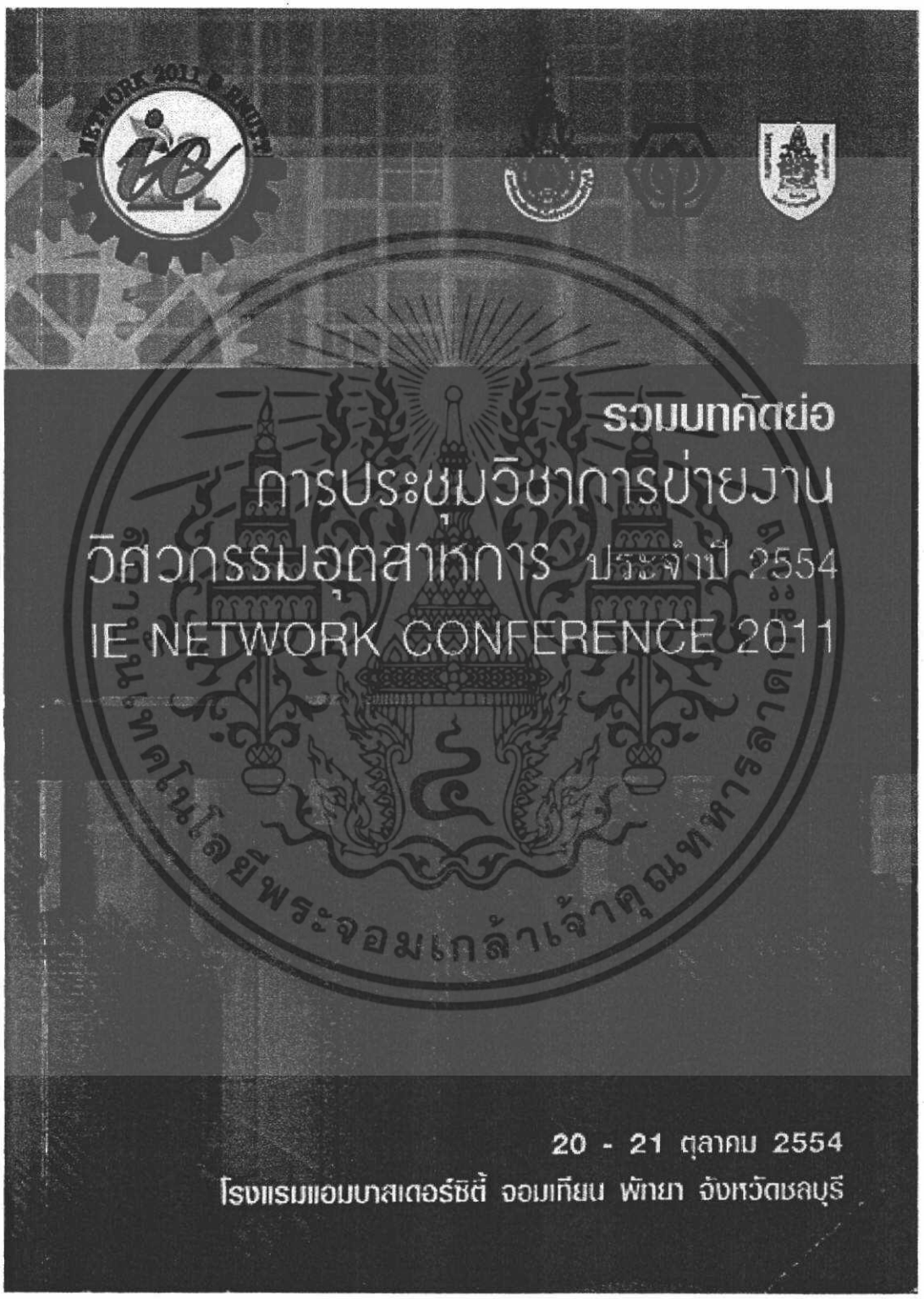
```
    }
    if (changeBlock == true)
    {
        member.Add(node);
        machineOld = machine;
        count = count + 1;
        changeBlock = false;
    }
    if (count-1 == critical_PathList.Count)
    {
        List<int> memberCopy = new List<int>(member);
        number_inBlock.Add(block, memberCopy.Count);                    //correct number in each block
        member_block.Add(block, memberCopy);                            //correct list member in to hastable
    }
}
foreach(DictionaryEntry BLOCK_NUM in member_block)
{
    if (Convert.ToInt32(BLOCK_NUM.Key) == 1)                                        // the fist block
    {
        if (Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)]) == 2)   //first : 2
        {
            ArrayList Temp_Array = new ArrayList();
            Temp_Array.Add(member_block[BLOCK_NUM.Key]);
            List<int> List_to_swap = (List<int>)Temp_Array[0];
            int F_temp = List_to_swap[0];
            int L_temp = List_to_swap[1];

            int index_F = VisitOld.IndexOf(F_temp);
            int index_L = VisitOld.IndexOf(L_temp);

            Visit_Local_Improve[index_F] = L_temp;
            Visit_Local_Improve[index_L] = F_temp;

        }
        else if (Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)]) > 2)      //first >2
        {
            ArrayList Temp_Array = new ArrayList();
            Temp_Array.Add(member_block[BLOCK_NUM.Key]);
            List<int> List_to_swap = (List<int>)Temp_Array[0];
```

```
                int F_temp = List_to_swap[Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)])-
1];             //Last
                int L_temp = List_to_swap[Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)])-
2];             //Last-1

                int index_F = VisitOld.IndexOf(F_temp);
                int index_L = VisitOld.IndexOf(L_temp);
                Visit_Local_Improve[index_F] = L_temp;
                Visit_Local_Improve[index_L] = F_temp;
            }
        }
        else if (Convert.ToInt32(BLOCK_NUM.Key) == block)                    // the last block
        {

            if (Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)]) >= 2)
            {
                ArrayList Temp_Array = new ArrayList();
                Temp_Array.Add(member_block[BLOCK_NUM.Key]);
                List<int> List_to_swap = (List<int>)Temp_Array[0];
                int F_temp = List_to_swap[0];
                int L_temp = List_to_swap[1];

                int index_F = VisitOld.IndexOf(F_temp);
                int index_L = VisitOld.IndexOf(L_temp);

                Visit_Local_Improve[index_F] = L_temp;
                Visit_Local_Improve[index_L] = F_temp;
            }
        }
        else
        {
            if (Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)]) == 2)
            {
                ArrayList Temp_Array = new ArrayList();
                Temp_Array.Add(member_block[BLOCK_NUM.Key]);
                List<int> List_to_swap = (List<int>)Temp_Array[0];
                int F_temp = List_to_swap[0];
                int L_temp = List_to_swap[1];
```

```
                int index_F = VisitOld.IndexOf(F_temp);
                int index_L = VisitOld.IndexOf(L_temp);

                Visit_Local_Improve[index_F] = L_temp;
                Visit_Local_Improve[index_L] = F_temp;
            }
            else if (Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)]) >= 4)
            {
                ArrayList Temp_Array = new ArrayList();
                Temp_Array.Add(member_block[BLOCK_NUM.Key]);
                List<int> List_to_swap = (List<int>)Temp_Array[0];
                int F_temp = List_to_swap[0];
                int S_temp = List_to_swap[1];

                int index_F = VisitOld.IndexOf(F_temp);
                int index_S = VisitOld.IndexOf(S_temp);

                Visit_Local_Improve[index_F] = S_temp;
                Visit_Local_Improve[index_S] = F_temp;

                int BL_temp = List_to_swap[Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)]) -
1];             //Last
                int L_temp = List_to_swap[Convert.ToInt32(number_inBlock[Convert.ToInt32(BLOCK_NUM.Key)]) -
2];             //Last-1

                int index_BL = VisitOld.IndexOf(BL_temp);
                int index_L = VisitOld.IndexOf(L_temp);

                Visit_Local_Improve[index_BL] = L_temp;
                Visit_Local_Improve[index_L] = BL_temp;
            }
        }
    }
    List<int> TempList = new List<int>(Visit_Local_Improve);
    loop: for (int _job = 1; _job <= num_job; _job++)                    //checked sequece in job
    {
        List<int> rowjob = seperate_JaddetIntMatrix_to_List(_job - 1,jagged_operation_table);
        int operOld = 0;
        int index_inVisitOld = -1;
```

```
        foreach (int oper in rowjob)
        {
            bool check_contain = Visit_Local_Improve.Contains(oper);
            if (check_contain == false)
            {
                MessageBox.Show("Visit Local ERROr");
            }
            else
            {
                int index_inVisit = Visit_Local_Improve.IndexOf(oper);
                if (index_inVisit < index_inVisitOld)
                {
                    Visit_Local_Improve[index_inVisitOld] = oper;        //swap operation
                    Visit_Local_Improve[index_inVisit] = operOld;
                    goto loop;
                }
                operOld = oper;
                index_inVisitOld = Visit_Local_Improve.IndexOf(operOld);
            }
        }
    }
    int oper_check = 0;
    for (int y = 0; y < List_oper.Count; y++)
    {
        oper_check = Visit_Local_Improve[y];
        bool u = List_oper.Contains(oper_check);
        if (u == false)
        {
            MessageBox.Show("Visit improve Local ERROr");
        }
    }
    return Visit_Local_Improve;
}
```

ภาคผนวก ค
บทความวิชาการที่ได้รับการตีพิมพ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประชุมวิชาการทางงานวิศวกรรมอุตสาหการ ประจำปี 2554
20-21 ตุลาคม 2554

การประยุกต์ใช้วิธีขั้นตอนอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

A Modified Ant Colony Algorithm for Job Shop Scheduling Problem

สรรเพ็ชญ์ ไชยศิลปสังข์[1]* อุดม จันทร์จรัสสุข[2]

[1,2]สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แขวงลาดกระบัง เขตลาดกระบังจังหวัดกรุงเทพมหานคร รหัสไปรษณีย์ 10520
E-mail: saanpet1251@hotmail.com*

บทคัดย่อ

งานวิจัยนี้นำเสนอวิธีขั้นตอนอาณานิคมมด(Ant Colony Optimization, ACO) สำหรับการค้นหาคำตอบที่ดีที่สุดของปัญหาการจัดตารางการผลิตแบบตามสั่ง แต่การหาคำตอบด้วยวิธีขั้นตอนอาณานิคมมดมีปัญหาในการเลือกใช้ค่าน้ำหนักของฟีโรโมน(α) และค่าน้ำหนักของฮิวริสติค(β) ในการเลือกการดำเนินการถัดไป ซึ่งค่า α และ β ในแต่ละปัญหามีค่าไม่เท่ากัน ดังนั้นในงานวิจัยนี้จะพัฒนาสมรรถนะในการหาคำตอบจากวิธีอาณานิคมมด โดยการหาค่า α และ β ที่เหมาะสมกับปัญหาใดๆ โดยการทดลองหาผลลำตอบที่ดีที่สุดและหาค่า α และ β ที่เหมาะสมกับแต่ละปัญหา การทดสอบประสิทธิภาพของโปรแกรมได้จัดทำการเขียนชุดคำสั่งด้วยโปรแกรม Visual C# ทำการทดสอบในชุดปัญหามาตรฐาน OR-Library พบว่าได้คำตอบที่มีประสิทธิภาพสูง คือผลคำตอบมีเปอร์เซ็นต์ความผิดพลาดที่ยอมรับได้(MRE%<10%) เมื่อเทียบกับคำตอบที่ดีที่สุด (optimal solution) และได้ผลลำตอบที่ดีที่สุดจำนวน 4 ชุดปัญหาคิดเป็น 26 % โดยการทดลองหาคำตอบนั้นทำให้ทราบค่า α และ β ที่มีความเหมาะสมของแต่ละชุดปัญหา และสามารถวิเคราะห์ความสัมพันธ์ของค่า α และ β ที่จะนำไปสู่คำตอบที่ดีที่สุด จากแนวโน้มของผลคำตอบที่เกิดจากการเปลี่ยนแปลงค่า α และ β

คำหลัก การจัดตารางการผลิตแบบตามสั่ง,วิธีขั้นตอนอาณานิคมมด

1. บทนำ

ปัจจุบันในทุกอุตสาหกรรมมีการแข่งขันสูงและการจัดตารางการผลิตเป็นส่วนสำคัญ การจัดตารางการผลิตที่มีประสิทธิภาพจะทำให้ลดค่าใช้จ่าย ลดเวลาในการผลิตและเกิดความได้เปรียบในการแข่งขันทางธุรกิจ อีกทั้งการจัดตารางการผลิตที่ดีนั้นเป็นประเด็นหลักในการประเมินประสิทธิภาพขององค์กร บ่อยครั้งปัญหาของระบบการผลิตจะเกี่ยวข้องกับรูปแบบและประสิทธิภาพของการจัดตารางการผลิต ตัวอย่างเช่น ถ้าสินค้าผลิตเสร็จก่อนเวลาส่งมอบทำให้เสียค่าใช้จ่ายเพิ่มในการเก็บสินค้านั้นและยังส่งผลถึงค่าใช้จ่ายในการจ้างคนงานแบบล่วงเวลาของกระบวนผลิตถัดไปที่นำส่วนประกอบนั้นไปใช้ โดยเฉพาะกับส่วนประกอบที่มีอายุจำกัด ในทางกลับกันถ้าการผลิตเสร็จล่าช้า กระบวนการผลิตถัดไปต้องหยุดหรือต้องเสียค่าใช้จ่ายเพิ่มในการสั่งซื้อส่วนประกอบจากที่อื่นมาทดแทน นั่นก็ส่งผลถึงสมดุลของสายการผลิต(line balance)อีกด้วย

ในระบบการผลิตในอุตสาหกรรมทั่วไปมี 2 ลักษณะคือ การผลิตแบบตามการไหล(flow shop)และการผลิตแบบตามสั่ง(job shop) การผลิตทั้งสองชนิดนี้แตกต่างกันตามลักษณะและทิศทางการผลิต ซึ่งวิธีการผลิตแบบตามสั่งมีแนวโน้มขยายตัวมากขึ้น เนื่องจากการผลิตแบบตามสั่งจะก่อให้เกิดความหลากหลายของผลิตภัณฑ์มากกว่าการผลิตแบบตามการไหล แต่มีความยุ่งยากและซับซ้อนมากกว่า เนื่องจากมีทิศทางการไหลของงานไม่แน่นอน ประกอบด้วยกระบวนการหลายๆกระบวนการ

จากที่กล่าวมาข้างต้นจะเห็นได้ว่าปัญหาในการจัดตารางการผลิตมีความสำคัญเป็นอย่างมาก สามารถแก้ปัญหาโดยการประยุกต์ใช้หลักของการวิจัยการดำเนินงาน(Operation Research) ซึ่งปัญหานั้นมีความซับซ้อนในการหาคำตอบที่ดีที่สุด(Optimal solution) ถ้าแก้ปัญหาด้วยวิธีดีเทอร์มินิสติกส์(Determiniatic) จะใช้เวลานานในการหาคำตอบ ถึงแม้คำตอบที่ได้จะเป็นคำตอบที่ดีที่สุดจริงๆ แต่ในสภาวะในปัจจุบัน งานมีจำนวนมากและต้องการคำตอบภายในเวลาที่ยอมรับได้เพื่อเพิ่มความคล่องตัวในการผลิต จึงจำเป็นต้องใช้วิธีการหาคำตอบที่ใช้เวลาน้อยกว่า นั่นก็คือวิธีฮิวริสติค (Heuristic) แม้วิธีฮิวริสติคอาจจะไม่ได้คำตอบที่ดีที่สุดแต่คำตอบที่ได้ถือว่ายอมรับได้หรือใกล้เคียงคำตอบที่ดีที่สุด

วิธีฮิวริสติคมีอยู่หลายวิธี โดยวิธีที่เลือกมาใช้ในการวิจัยนี้คือวิธีขั้นตอนอาณานิคมมด (Ant Colony Algorithm)อยู่ในกลุ่มวิธีการทางปัญญาประดิษฐ์(Artificial Intelligent) ซึ่งจะใช้หลักความน่าจะเป็นมาช่วยในการหาคำตอบ แต่วิธีนี้มีข้อด้อยในเรื่องของสมรรถนะเนื่องจากเวลาที่ใช้ในการหาคำตอบค่อนข้างมาก และถ้าเปลี่ยนรูปแบบของปัญหา ก็ต้องมีการเปลี่ยนแปลงค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค(β) ซึ่งค่าคงที่ทั้งสองนี้ต้องใช้เวลามากในการหาค่าที่เหมาะสม โดยในรอบ 10 ปีที่ผ่านมามีการวิจัยและการพัฒนาวิธีการหาคำตอบแบบอาณานิคมมดเพียง 10% เมื่อเทียบกับวิธีอื่น เช่นวิธีการค้นหาแบบทาบู(Tabu Search) และวิธีเชิงพันธุกรรม(Genetic Algorithm)[1] ซึ่งน้อยมาก จึงทำให้วิธีการนี้ไม่ได้รับการพัฒนาเท่าที่ควร

ดังนั้นในงานวิจัยนี้จะพัฒนาสมรรถนะในการหาคำตอบจากวิธีอาณานิคมมด โดยการหาค่าน้ำหนักของฟีโรโมน(α) และค่าน้ำหนักของฮิวริสติค(β)ที่เหมาะสมกับปัญหาใดๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

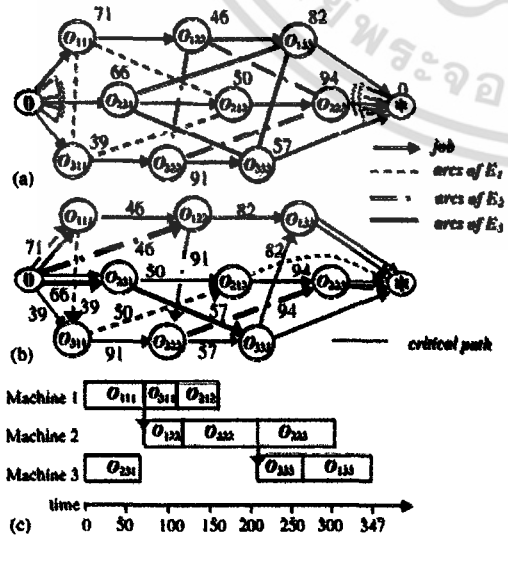
2. ทฤษฎีที่เกี่ยวข้อง

2.1 การผลิตแบบตามสั่ง (Job shop)

การผลิตแบบตามสั่ง(Job shop) ให้ความสำคัญกับตัวผลิตภัณฑ์เป็นสำคัญ เนื่องจากการผลิตแบบตามสั่ง จะมีขั้นตอนหรือลำดับการดำเนินงานที่แตกต่างกันตามชนิดของผลิตภัณฑ์ โดยที่การผลิตจะต้องคอยสนองความต้องการของตลาดและลูกค้าเป็นหลัก โดยทั่วไปการผลิตแบบตามสั่งนี้ จะมีลำดับการผลิตที่ซับซ้อนกว่าการผลิตแบบตามการไหลและยังต้องพิจารณาขั้นตอนในระหว่างการผลิต เช่นการลำเลียง และขนถ่ายสินค้าระหว่างการผลิต (intermediate product) จากอุปกรณ์หนึ่งไปยังอีกอุปกรณ์หนึ่ง ลักษณะของการผลิตแบบตามสั่งนั้น เป็นการผลิตแบบ make to order ที่มีปริมาณการผลิตไม่มาก เมื่อเทียบกับการผลิตแบบ make to stock เพราะว่ามีการผลิตที่ไม่สม่ำเสมอ ซึ่งประสิทธิภาพในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง(job shop scheduling problem, JSSP)มีผลต่อประสิทธิภาพของการผลิตและคุณภาพของผลิตภัณฑ์ รูปแบบของปัญหาการผลิตแบบตามสั่ง จะมีงานที่แตกต่างกัน n งาน จัดสรรลงใน m เครื่องจักรที่แตกต่างกัน โดยมีข้อกำหนดคือ (1)งานแต่ละงานจะทำบนเครื่องจักรทีละ 1 เครื่อง (2)ไม่มีงานไหนมีความสำคัญกว่างานอื่น (3)ทุกการดำเนินการไม่สามารถขาดตอน(interrupt) (4)แต่ละเครื่องจักรสามารถทำงานเพียง 1 งานในเวลานั้น (5)ไม่มีการกำหนดเวลาเริ่มต้นและเวลาส่งงาน[2]

2.2 การแสดงการหาเส้นทางโดย Disjunctive graph

ในทุกๆปัญหาของการจัดตารางการผลิตแบบตามสั่ง(JSP) สามารถเขียนให้อยู่ในรูปของ disjunctive graph โดยมีรูปแบบ $DG=(N,A,E_{jk})$ เมื่อ N เป็นเซตของการดำเนินการจาก dummy start ไปจนถึงการดำเนินการสุดท้าย A คือเซตของเส้นที่ต่อระหว่างการดำเนินการ E_{jk} คือเซตของเส้นที่ไม่เชื่อมต่อระหว่างการดำเนินการของงานใดๆ ที่กระทำบนเครื่องจักรเดียวกัน(disjunctive arc)[4]



รูปที่ 1 (a)Disjunctive graph,(b)การแสดงลำดับการผลิตลงบน disjunctive graph,(c)Gant diagram ของลำดับการผลิต[4]

ในกราฟแสดงค่าเวลาการดำเนินการ(processing time)และลำดับการดำเนินการบนแต่ละเครื่องจักร โดย disjunctive graph สามารถแสดงลำดับของการดำเนินการ(operation sequence)ของแต่ละเครื่องจักร และเมื่อเขียน disjunctive graph ใหม่ในรูปของแผนภูมิแกนต์ (Gant diagram) ก็สามารถทราบถึงเส้นทางวิกฤต (Critical path) และค่าเวลาในการทำงานทั้งหมด(makespan)

2.3 การหาคำตอบด้วยวิธีอาณานิคมมด (Ant Colony Optimization, ACO)

M.Dorigo และคณะ[9] ได้เสนอแนวคิดวิธีการหาคำตอบโดยใช้วิธีอาณานิคมมด ซึ่งการค้นหาอาศัยการเลียนแบบพฤติกรรมการหาอาหารของมดจริง เทียบได้กับการหาเส้นทางจากรังมดไปยังแหล่งอาหาร โดยมดจะอาศัยสารเคมีที่เรียกว่า ฟีโรโมน(pheromone)ที่ปล่อยไว้บนเส้นทาง ใช้ในการจดจำเส้นทางและใช้ในการสื่อสารกับมดตัวอื่น Dorigo[9] ได้สรุปคุณลักษณะของวิธีการหาคำตอบด้วยวิธีอาณานิคมมดดังนี้ (1)เป็นอัลกอริทึมที่เอนกประสงค์(Versatile)คือสามารถใช้ได้กับปัญหาเดียวกันได้หลากหลายลักษณะ (2)มีความคงทน(Robust)คือสามารถใช้งานได้ในปัญหาที่มีการเปลี่ยนแปลงเล็กน้อย (3)อาศัยพื้นฐานประชากร(Population Based)คือใช้การป้อนกลับที่เป็นประโยชน์(Positive Feedback)ช่วยในการหาคำตอบ ซึ่งจะทำให้ระบบมีความยืดหยุ่นที่จะดำเนินการค้นหาแบบขนาน (parallel implementation)

ในบทความของ M.Dorigo[10] ได้แสดงวิธี การหาเส้นทางของมด มดจะใช้การดมกลิ่นของฟีโรโมนที่มีการปล่อยไว้บนเส้นทาง มดตัวต่อมาจะพิจารณาจากความแข็งแรงของกลิ่นฟีโรโมนช่วยในการตัดสินใจเลือกเส้นทางเดินต่อไป โดยมีปัจจัยที่มีผลต่อความเข้มข้นคือ การระเหย(evaporate) ซึ่งการระเหยนั้นขึ้นอยู่กับเวลา ถ้าความเร็วในการเดินของมดลดลงที่ และระยะทางของเส้นทางแต่ละเส้นทางแตกต่างกัน ระยะทางที่ยาวกว่าต้องใช้เวลาในการเดินทางนานกว่า ซึ่งจะทำให้ความเข้มข้นของฟีโรโมนน้อยกว่าระยะทางสั้น ทำให้มดตัวอื่นจะมีความน่าจะเป็นในการเลือกเส้นทางที่มีความเข้มข้นของฟีโรโมนมากนั่นก็คือเส้นทางที่สั้นกว่านั่นเอง

อัลกอริทึมระบบมด(Ant System) ของการแก้ปัญหาคนเดินทางขายของ(Travelling Salesman Problem ,TSP)[11] สามารถนำมาประยุกต์ใช้ในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง (job-shop scheduling) โดยจะมีกฎการเปลี่ยนสถานะ (state transition rule) ดังนี้

$$Q_n = \begin{cases} max_p \left\{ \tau_p(O_p, O_{nt'})^{\alpha} \eta(O_p, O_{nt'})^{\beta} \right\} & if q \le q_0 \\ P & if q > q_0 \end{cases} \quad (1)$$

โดย O_n = การดำเนินการถัดไปที่มดจะเลือก
τ_p = ความเข้มของฟีโรโมนตำแหน่งปัจจุบัน
η = ข้อมูลทางฮิวริสติค
O_p = การดำเนินการในขณะนั้น
$O_{nt'}$ = การดำเนินการถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

q = จำนวนจริงที่ได้จากการสุ่มตัวเลขที่มีการแจกแจงแบบยูนิฟอร์มในช่วง[0,1]

q_0 = พารามิเตอร์การสำรวจ($0 \leq q_0 \leq 1$)

α = ค่าน้ำหนักของฟีโรโมน

β = ค่าน้ำหนักของฮิวริสติค

P = การดำเนินการที่สามารถกระทำได้ (feasible move) โดยเลือกจากกฎสัดส่วนของการสุ่ม (random proportional rule) ดังนี้

$$P_n = \frac{\tau_p(O_p, O_{nt})^{\alpha}\eta(O_p, O_{nt})^{\beta}}{\sum_{O_{nt} \in CL}\left\{\tau_p(O_p, O_{nt})^{\alpha}\eta(O_p, O_{nt})^{\beta}\right\}} \quad (2)$$

โดย P_n = ความน่าจะเป็นของแต่ละการดำเนินการ

CL = รายชื่อการดำเนินการที่สามารถทำต่อไป (Candidate list)

จากสมการที่ (1)ค่า q_0ที่กำหนดขึ้นมาจะนำมาพิจารณากับค่า q ที่สุ่มได้ เมื่อค่า qน้อยกว่าหรือเท่ากับ q_0(ตั้งค่า q_0เข้าใกล้ 1 จะทำให้ระบบให้ความสำคัญกับการหาเส้นทางที่ดีที่สุด,exploitation activity)มดจะเลือกจากการดำเนินการที่มีค่าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคมากที่สุดจากรายชื่อการดำเนินการที่สามารถทำต่อไป แต่ถ้าค่า q มากกว่า q_0(ตั้งค่า q_0เข้าใกล้ 0 จะทำให้ระบบให้ความสำคัญกับการสำรวจ,exploration activity)มดจะเลือกจากสมการที่(2)โดยจะเลือกจากความน่าจะเป็นของค่าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคจากรายชื่อการดำเนินการที่สามารถทำต่อไปถ้าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคต่ำ ความน่าจะเป็นที่มดจะเลือกการดำเนินการนั้นจะมีค่าต่ำ ในทางกลับกันถ้าผลคูณของความเข้มข้นของฟีโรโมนและค่าทางฮิวริสติคสูง ความน่าจะเป็นที่มดจะเลือกการดำเนินการนั้นจะมีค่าสูง โดยผลรวมของความน่าจะเป็นทั้งหมดของการดำเนินการที่ผ่านไปได้ทำมีค่าเท่ากับ 1 และด้วยเหตุนี้ทำให้การหาคำตอบด้วยวิธีอาณานิคมมดสามารถเลือกการดำเนินการถัดไปที่หลากหลาย ซึ่งอาจจะนำไปสู่เส้นทางอื่นที่ดีกว่าเส้นทางเดิมที่มีอยู่

อีกหนึ่งพารามิเตอร์ที่สำคัญในการเลือกของสมการที่ (1), (2) คือ η ซึ่งเป็นข้อมูลทางฮิวริสติคมีการวิจัยมากมายที่แสดงถึงการได้มาของ η เช่น M.Dorigo[9] ได้ทำการวิจัยเกี่ยวกับ TSP กำหนดค่า η เท่ากับ $\frac{1}{d}$ เมื่อ d เป็นระยะทางระหว่างเมือง ซึ่งถ้าเปรียบเทียบกับปัญหาที่เป็นการจัดตารางการผลิต η จะเท่ากับหนึ่งส่วนเวลาดำเนินการใดๆ J.Heinonen[12] ได้กำหนด η ไว้ดังนี้

$$\eta = \frac{C}{d_{O_p, O_{nt}}} \quad (3)$$

โดย $d_{O_p, O_{nt}}$ = ระยะทางระหว่าง node O_pไปO_{nt}

C = ค่าคงที่

ในการวิจัยนี้จะใช้ข้อมูลทางฮิวริสติคของ Y.Zhong[13] โดยกำหนดข้อมูลทางฮิวริสติคให้ซับซ้อนขึ้น โดยคำนึงถึงความสมดุลของภาระงาน(balance of workload) บนเครื่องจักรแต่ละเครื่อง ให้ค่า η เกิดจากการถ่วงน้ำหนักสองส่วน ในส่วนแรกเป็นข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่มีเวลาในการดำเนินการ(processing time)น้อย ส่วนที่สองเป็นข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่ทำอยู่บนเครื่องจักรที่มีภาระงานในเวลานั้นน้อยกว่าเครื่องจักรอื่น เพื่อเป็นการรักษาสมดุลของภาระงานในทุกเครื่องจักร

$$\eta = \eta_H \times \eta_{workload} \quad (4)$$

$$\eta_{workload} = \left(\frac{t_{max} - t_{next'}}{t_{max} - t_{min}}\right) x_1 + x_2 \quad (5)$$

โดย η_H= ข้อมูลทาง ฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่มีเวลาในการดำเนินการ(processing time)น้อย มีค่าเท่ากับ $\frac{C}{d_{O_p, O_{nt}}}$

$\eta_{workload}$ = ข้อมูลทางฮิวริสติคของการมอบหมายการดำเนินการไปยังการดำเนินการที่ทำอยู่บนเครื่องจักรที่มีภาระงานในเวลานั้นน้อยกว่าเครื่องจักรอื่น

x_1, x_2 = เป็นค่าคงที่สัดส่วนของน้ำหนักของค่าภาระงาน ($x_1 + x_2 = 1$)

ในการหาเส้นทางของมดแต่ละตัวในแต่ละรอบนั้น จะมีฟังก์ชันจุดประสงค์(objective function) เพื่อบอกว่าเส้นทางที่ดีที่สุดเป็นของมดตัวใดดังนี้

$$opt = min(T) = min\{max[T_1, T_2, T_3, \ldots, T_m]\} \quad (6)$$

โดย T_{i_m} =เวลาที่ใช้ในการดำเนินการ

เมื่อได้ค่าฟังก์ชันจุดประสงค์มาแล้วก็นำมาหา Fitness function เพื่อให้การเปรียบเทียบคำตอบในแต่ละรอบของการหาเส้นทาง ซึ่ง fitness function จะปรับค่าคำตอบจากฟังก์ชันจุดประสงค์มาให้อยู่ในช่วง [0,100] โดยมีสมการดังนี้

$$f_i = 100 \times \frac{opt}{T_{Ant}} \quad (7)$$

โดย opt = ค่าคำตอบที่ดีที่สุดในขณะนั้น

T_{Ant} = ค่าคำตอบจากมดใดๆ

อธิบายได้ว่า ถ้าเส้นทางใดมีค่าเวลาในการดำเนินการทั้งหมด (makespan)น้อยที่สุดหรือเป็นค่าที่ดีที่สุดของรอบนั้น จะมีค่า fitness function เท่ากับ 100 โดยเส้นทางอื่นๆที่หาจากมดตัวอื่นๆ จะมีค่าน้อยกว่าร้อย การปรับค่าผลคำตอบจากในการดำเนินการทั้งหมดมาเป็น fitness function จะทำให้การเปรียบเทียบมีประสิทธิภาพมากขึ้น เพราะค่าของตัวเลขมีขอบเขตเดียวกัน

2.4 การปรับค่าฟีโรโมน (Pheromone updating)

ในการวิจัยนี้ใช้การปรับค่าฟีโรโมน(Pheromone updating) ใน 2 กรณีคือการปรับปรุงค่าแบบแคบ(local pheromone updating)และอีกกรณีหนึ่งคือการปรับค่าแบบวงกว้าง(global pheromone

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

updating)โดยจะปรับค่าฟีโรโมนใหม่ เมื่อครบ 1 รอบ(มดทุกตัวหาเส้นทางหมดแล้ว)

การปรับปรุงค่าแบบแคบ

$$\tau_p(O_p, O_{nt'}) = \begin{cases} \tau_{old}(O_p, O_{nt'}) + \frac{y}{C_{max}} & \text{ถ้า } O_p, O_{nt'} \in \text{tour } S_{tl} \\ \tau_{old}(O_p, O_{nt'}) - \frac{y}{C_{max}} & \text{อื่นๆ} \end{cases} \quad (8)$$

การปรับค่าแบบวงกว้าง

$$\tau_p(O_p, O_{nt'}) = \begin{cases} \tau_{old}(O_p, O_{nt'}) + z & \text{ถ้า } O_p, O_{nt'} \in \text{tour } S_{tg} \\ \tau_{old}(O_p, O_{nt'}) \times (1-\rho) & \text{อื่นๆ} \end{cases} \quad (9)$$

โดย S_{tl} = ลำดับของการดำเนินการที่ให้ค่าเวลาในการดำเนินการทั้งหมด(makespan)น้อยที่สุดของรอบนั้นๆ (เส้นทางวิกฤตใช้ในการระบุเส้นทางที่จะปรับปรุงค่าฟีโรโมน)

S_{tg} = ลำดับของการดำเนินการที่ให้ค่าเวลาในการดำเนินการทั้งหมดน้อยที่สุดที่หาได้ในขณะนั้น

τ_{old} = ค่าความเข้มข้นของฟีโรโมนก่อนการปรับปรุงค่า

ρ = ค่าคงที่การระเหย(evaporating)($\rho \in [0,1)$)

y, z = ค่าคงที่

C_{max}= ค่าเวลาในการดำเนินการทั้งหมด(makespan)

กำหนดค่าความเข้มขั้นของฟีโรโมนที่น้อยที่สุดและมากที่สุด (τ_{min}, τ_{max}) ซึ่งถ้าค่าระดับของฟีโรโมนต่ำกว่าค่าต่ำสุดหรือสูงกว่าค่าสูงสุดที่ตั้งไว้ จะปรับค่าให้เท่ากับค่าต่ำที่สุดและสูงที่สุดที่ตั้งไว้ตามลำดับ[14]

2.5การปรับปรุงคุณภาพคำตอบ (Local improvement)

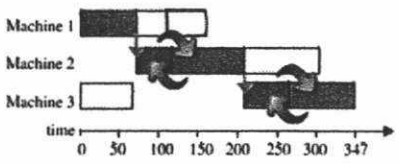
หลังจากได้คำตอบจากการวนครบ 1 รอบแล้ว จะทำการปรับปรุงคุณภาพของตารางการผลิต โดยในการวิจัยนี้จะนำวิธีการปรับปรุงคุณภาพคำตอบของ E.Nowicki[15] มาประยุกต์ใช้งาน หลักการปรับปรุงคือ ถ้าหลังจากการปรับปรุงแล้วได้คำตอบที่ดีขึ้น ก็จะใช้คำตอบนั้น แต่ถ้าคำตอบที่ปรับปรุงไม่ดีขึ้น ก็จะใช้คำตอบเดิมก่อนปรับปรุง โดยวิธีการปรับปรุงจะเป็นการสลับกัน block ข้างเคียง (Neighborhood solution)ดังนี้

- วิเคราะห์ เส้นทางวิกฤต โดยแบ่งเป็น block ตามเครื่องจักร
- ทำการปรับปรุงเส้นทางวิกฤตดังนี้

ใน block ที่ 1: สลับสองการดำเนินการสุดท้าย

ในblock อื่นๆ : สลับสองการดำเนินการแรกและสุดท้าย เฉพาะ block ที่มีมากกว่า 2 การดำเนินการ

ใน block สุดท้าย : สลับสองการดำเนินการแรกแต่ที่สำคัญ การสลับกันของการดำเนินการ ต้องสอดคล้องกับลำดับการผลิตด้วยดังรูปที่ 2



รูปที่ 2 แสดงวิธีการปรับปรุงคุณภาพคำตอบของ E.Nowicki[15]

3.ความสำคัญของค่าน้ำหนักของฟีโรโมน (α) และค่าน้ำหนักของฮิวริสติค (β)

กฎการเปลี่ยนสถานะ(state tranaition rule) พารามิเตอร์สองตัวที่มีความสำคัญมากคือ ค่าน้ำหนักของฟีโรโมน(α) และค่าน้ำหนักของฮิวริสติค(β) จะบอกความสัมพันธ์เป็นสัดส่วนระหว่างความเข้มข้นของฟีโรโมนกับค่าข้อมูลทางฮิวริสติค(η) M.Dorigo[9]ได้แสดงค่าความสัมพันธ์ของ α และ β ดังตารางที่ 1

ตารางที่ 1 แสดงค่าความสัมพันธ์ของค่า α และ β ที่จะให้คำตอบที่ดีที่สุด[9]

α	β
0.5	5.0
1.0	1.0
1.0	2.0
1.0	5.0

ค่า α และ β ตามตารางที่1 เป็นค่าที่เหมาะสมในการใช้งาน ถ้าวิเคราะห์เรื่องความสัมพันธ์ผลก็คือ ถ้าαมีค่ามากมดจะเลือกเส้นทางจากความเข้มข้นของฟีโรโมนเป็นหลัก จะทำให้ประสิทธิภาพในการค้นหาแบบวงกว้าง(global search)ลดลง แต่ถ้า β มีค่ามากมดจะเลือกเส้นทางที่ระยะทางสั้น(Local area) แต่ก็จะทำให้ประสิทธิภาพในการค้นหาแบบวงกว้างลดลงเช่นกัน จะทำให้ระบบค้นหาเป็นการค้นหาคำตอบในวงแคบ(Local search)

ถ้าทั้ง α และ β มีค่ามาก forward feedback จะส่งผลถึงคำตอบ ซึ่งถ้าปัญหามีขนาดใหญ่ ผลคำตอบอาจจะเป็นคำตอบที่ดีที่สุดในวงแคบ(local optimum)

ถ้าทั้ง α และ β มีค่าน้อย มดจะไม่สนใจความสำคัญของข้อมูลใดๆ การหาเส้นทางก็จะเป็นการสุ่ม[14]

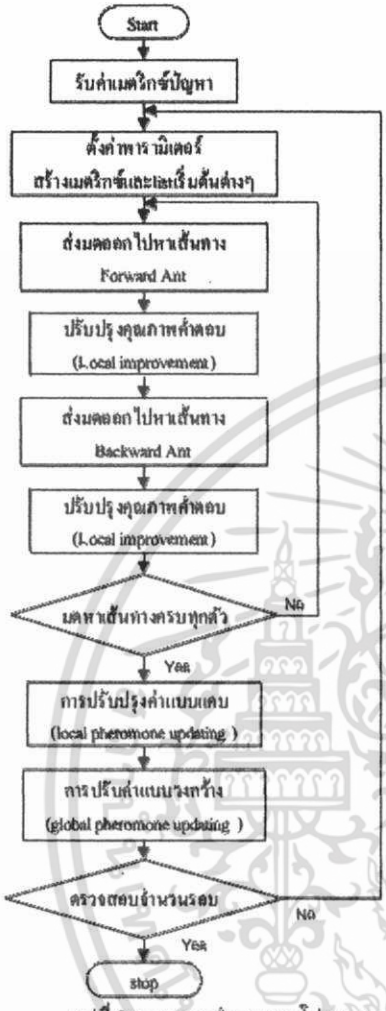
ถ้า $\alpha = 0$ การค้นหาเส้นทางจะไม่นำความเข้มข้นของฟีโรโมนมาพิจารณา ทำให้จะเลือกการดำเนินการที่ใกล้หรือเวลาดำเนินการน้อย ในทางกลับกัน ถ้า $\beta = 0$ การค้นหาจะพิจารณาเฉพาะความเข้มข้นของฟีโรโมนเท่านั้น ซึ่งโดยทั่วไปแล้วจะนำไปสู่สถานการณ์เมื่อยล้า(stagnation situation)เพราะมดทุกตัวจะเดินตามเส้นทางเดิมๆ[12]

จะเห็นได้ว่าค่า α และ β นั้นมีความสำคัญมากในการหาการดำเนินการถัดไปในกระบวนการเปลี่ยนสถานะ ในการวิจัยที่ผ่านมาการหาค่า α และ β จะใช้วิธีการสุ่มและทดลองหาคำตอบออกมา ซึ่งต้องทดลองสุ่มไปเรื่อยๆจนกว่าจะได้คำตอบเป็นที่น่าพอใจ ซึ่งการหาค่า α และ β ด้วยวิธีการนี้ต้องใช้เวลานาน ยิ่งไปกว่านั้นค่า α และ β ที่ได้อาจจะไม่ใช่ค่าที่เหมาะสมที่สุด

เพราะฉะนั้นในการวิจัยนี้จะทำการหาค่า α และ β ที่เหมาะสมกับแต่ละรูปแบบของปัญหา เพื่อเพิ่มประสิทธิภาพในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.การเขียนโปรแกรมจัดตารางการผลิต



รูปที่ 3 แสดงการทำงานของโปรแกรม

5.การหาและเปรียบเทียบคำตอบ

โปรแกรมการจัดตารางการผลิตแบบตามสั่งด้วยวิธีอาณานิคมมดจะทำการทดลองหาคำตอบในชุดปัญหามาตรฐาน Replicate Lawrence หรือ ชุดปัญหา RL[18] โดยเมตริกซ์ของปัญหานำมาจาก OR-Library ซึ่งเป็นชุดปัญหาที่ได้รับการยอมรับจากการวิจัยต่างๆในเรื่องของความใกล้เคียงกับสภาวะงานอุตสาหกรรมจริง โดยจะทำการทดลองในปัญหา La01 จนถึง La15 คำตอบที่ได้จากการทดลองคือค่าเวลาในการดำเนินการทั้งหมด(makespan) ซึ่งถ้าคำตอบมีค่าน้อยแสดงว่าตารางการผลิตนั้นมีประสิทธิภาพสูง โดยการทดลองหาค่าน้ำหนักของฟีโรโมน(α)และค่าน้ำหนักของฮิวริสติค(β) ที่เหมาะสมจะใช้การปรับเปลี่ยนค่า α และ β และทดลองหาคำตอบที่ดีที่สุด หลังจากนั้นนำผลคำตอบที่ได้เปรียบเทียบกับคำตอบที่ดีที่สุด (optimal solution)และหาค่าเปอร์เซ็นต์ความผิดพลาด(MRE%, mean relative error)และเปรียบเทียบกับวิธีฮิวริสติคอื่นได้แก่ RD-ACS: เป็นวิธีอาณานิคมมด ของการจัดตารางการผลิตแบบตามสั่งด้วยเส้นทางที่ยืดหยุ่นและสามารถแยกเวลาคิดตั้ง[4], TSSB: เป็นวิธีการค้นหาแบบทาบู(tabu search)โดยอาศัยการ shifting bottleneck [17], HGA: เป็นวิธีการทางพันธุกรรมในการจัดตารางการผลิตแบบตามสั่ง[18], วิธีค้นหาคำตอบด้วยวิธีBeam search [19]

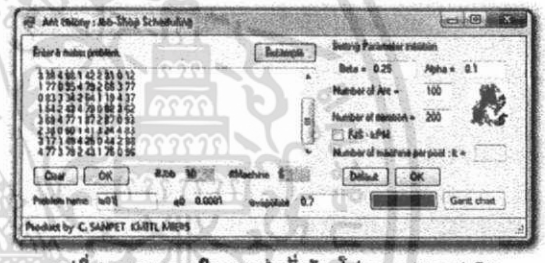
6.ผลการดำเนินการ

การทดสอบประสิทธิภาพของโปรแกรม ได้จัดทำการเขียนชุดคำสั่งด้วยโปรแกรม Visual C# ดังรูปที่ 4 และทดสอบอัลกอริทึมกับคอมพิวเตอร์รุ่น Pentium Core Quad ,CPU 2.5 GHz., Ram 2 GB. โดยได้ใช้ค่าคงที่ต่างๆดังนี้

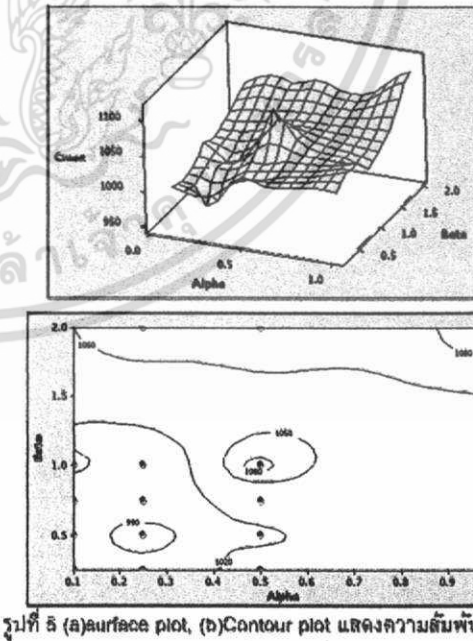
จำนวนมด 200 ตัว/รอบ, วนซ้ำ 200 รอบ, $q_0 = 0.0001$, $\rho = 0.7$, $C = 50$, $x_1 = 0.8$, $x_2 = 0.2$, $y = 10$, $\varepsilon = 0.3$ โดยการทดลองหาค่าน้ำหนักของฟีโรโมน(α)และค่าน้ำหนักของฮิวริสติค(β)ที่เหมาะสมของแต่ละปัญหา ได้กำหนดขอบเขตของค่าทั้งสองไว้ดังนี้

$$\alpha = \{0.1, 0.25, 0.5, 1\} \quad \beta = \{0.25, 0.5, 0.75, 1, 2\}$$

หลังจากทดลองปรับเปลี่ยนค่า α และ β ได้ผลการทดลองรูปที่ 5 และตารางที่ 2



รูปที่ 4 แสดงการเขียนชุดคำสั่งด้วยโปรแกรม Visual C#



รูปที่ 5 (a)surface plot, (b)Contour plot แสดงความสัมพันธ์ระหว่างค่า α และ β เปรียบเทียบกับค่าเวลาในการดำเนินการทั้งหมด (makespan)ในชุดปัญหา La09

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2 แสดงค่าเวลาที่ใช้ในการดำเนินการ(makespan)ของการวิจัยนี้เปรียบเทียบกับวิธีอื่นและค่า α และ β ที่เหมาะสมแต่ละปัญหา

ปัญหา	จำนวน งาน x เครื่องจักร	คำตอบที่ดีที่สุด (optimal)	HGA (2005)[18]	TSSR (2000)[17]	RD-ACS (2007)[4]	Beam search (1999)[20]	การวิจัยนี้ เฉลี่ย	max	min	MRE	β	α
la01	10x5	666	666	666	666	666	698	721	681	0.02	0.75	0.25
la02	10x5	655	655	655	665	704	726	734	716	0.09	0.75	0.10
la03	10x5	597	603	597	604	650	696	737	657	0.10	0.50	0.25
la04	10x5	590	598	590	611	620	649	653	643	0.09	0.50	0.10
la05	10x5	593	593	593	593	593	593	593	593	0.00	0.50	0.10
la06	15x5	926	926	926	926	926	949	962	933	0.01	0.50	0.10
la07	15x5	890	890	890	890	890	959	982	941	0.08	0.75	0.10
la08	15x5	863	863	863	863	863	941	951	927	0.07	0.50	0.10
la09	15x5	951	951	951	951	951	963	974	951	0.00	0.50	0.25
la10	15x5	958	958	958	958	958	963	973	958	0.00	0.25	0.10
la11	20x5	1222	1222	1222	1222	1222	1294	1299	1292	0.06	0.50	0.25
la12	20x5	1039	1039	1039	1039	1039	1099	1091	1098	0.06	0.25	0.10
la13	20x5	1150	1150	1150	1150	1150	1217	1228	1185	0.03	0.50	0.10
la14	20x5	1292	1292	1292	1292	1292	1292	1292	1292	0.00	0.25	0.10
la15	20x5	1207	1207	1207	1212	1207	1375	1408	1338	0.11	0.50	0.25

จากรูปที่ 5 สามารถวิเคราะห์แนวโน้มของผลคำตอบที่เกิดจากการเปลี่ยนค่า α และ β คือ ถ้าค่า α และ β มากขึ้นจะส่งผลให้ค่าเวลาในการดำเนินการทั้งหมด(makespan) มีแนวโน้มมากขึ้น และสามารถวิเคราะห์ได้ว่า ค่า α ไม่มีผลต่อค่าเวลาในการดำเนินการทั้งหมด และถ้านำผลการทดลองจากตารางที่ 2 มาวิเคราะห์ร่วมกันจะสรุปได้ว่า ค่า β ต้องมีค่ามากกว่า α จึงจะทำให้เกิดคำตอบที่ดีที่สุด และผลคำตอบของการวิจัยนี้เปรียบเทียบกับผลคำตอบที่ดีที่สุดและการวิจัยอื่น ให้ผลที่ยอมรับได้ โดยให้ค่าเปอร์เซ็นต์ความผิดพลาด (MRE%)ไม่เกิน 10% ในปัญหา la01-la14 ส่วนในปัญหาที่มีค่าเปอร์เซ็นต์ความผิดพลาดมากกว่า10% เนื่องจากจำนวนงานและจำนวนรอบมีค่าน้อยสำหรับการแก้ปัญหานั้น และจากการทดลองหาคำตอบที่ดีที่สุดการวิจัยนี้ได้ผลคำตอบที่ดีที่สุดในชุดปัญหา la05, la09, la10, la14

7.สรุป

งานวิจัยนี้นำเสนอวิธีขั้นตอนวิธีนิคมมด สำหรับการค้นหาคำตอบที่ดีที่สุดของปัญหาการจัดตารางการผลิตแบบตามสั่ง ในชุดปัญหามาตรฐาน OR-Library พบว่าได้คำตอบที่มีประสิทธิภาพสูง คือ ผลคำตอบมีเปอร์เซ็นต์ความผิดพลาดที่ยอมรับได้(MRE%<10%) เมื่อเทียบกับคำตอบที่ดีที่สุด(optimal solution) และได้ผลคำตอบที่ดีที่สุดจำนวน 4 ชุดปัญหาคิดเป็น 26 % โดยการทดลองหาคำตอบนั้นทำให้ทราบค่าน้ำหนักของฟีโรโมน(α)และค่าน้ำหนักของฮิวริสติค (β)ที่มีความเหมาะสมของแต่ละชุดปัญหา และสามารถวิเคราะห์ความสัมพันธ์ของค่า α และ β ที่จะนำไปสู่คำตอบที่ดีที่สุด จากแนวโน้มของผลคำตอบที่เกิดจากการเปลี่ยนแปลงค่า α และ β

ในอนาคตผู้วิจัยจะพัฒนาโปรแกรมการจัดตารางการผลิตแบบตามสั่งโดยใช้วิธีขั้นตอนวิธีนิคมมด ในด้านการหาค่า α และ β ที่เหมาะสมของแต่ละปัญหาให้มีประสิทธิภาพมากขึ้น โดยจะพัฒนาให้การหาค่า α และ β นั้นเป็นอย่างอัตโนมัติ โดยทำการเขียนชุดคำสั่งโปรแกรมที่มีการวิเคราะห์แนวโน้มของผลคำตอบที่ได้จากการปรับเปลี่ยนค่า α และ β ซึ่งจะทำให้การหาคำตอบที่ดีที่สุดของปัญหาการจัดตารางการผลิตแบบตามสั่งด้วยวิธีขั้นตอนวิธีนิคมมดมีความยืดหยุ่น(flexible)มากขึ้น สามารถใช้หาคำตอบในปัญหาการจัดตารางการผลิตแบบตามสั่งที่หลากหลายมากขึ้น

เอกสารอ้างอิง

[1] Fan K. and Zhang R. 2010. An Analysis Of Research In Job Shop Scheduling Problem (2000-2009).IEEE international conference on advanced management science(ICAMS), Aug. 1-23,2010: 282-288.

[2] เอกชาติ หัตถา. 2543. การประยุกต์เจเนติกส์อัลกอริธึมส์สำหรับการจัดลำดับการผลิตชนิดตามงานในกระบวนการเคมีแบบกะ. วิทยานิพนธ์ วิศวกรรมศาสตร์มหาบัณฑิต ภาควิชาวิศวกรรมเคมี มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี.

[3] Chang T. and Feng L. 2003. A New Hybrid Heuristic Technique for Solving Job-shop Scheduling problem. IEEE International Workshop on Intelligent Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Acquisition and Advanced Computing System: Technology and Applications, Sept. 8-10, 2003: 53-58.

[4] Andrea R.and Gino D. 2007. Flexible job-shop scheduling with routingflexibility and separable setup times using ant colony optimisation method. Robotics and Computer-Integrated Manufacturing,23:503–516

[5] ธนิรุจน์ ตันเทอดทิดม์. 2546. การศึกษาถึงวิธีการแก้ปัญหาการจัดเรียงลำดับงานที่เป็นแบบ Multiple Machine-Multiple Job โดยใช้วิธี Genetic Algorithm. วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธรรมศาสตร์.

[6] Sabuncuoglu I. and Bayiz M. 1999. Theory and Methodology Job shop scheduling with beam search. European Journal of Operational Research ,118: 390-412.

[7] Ching L. and Fang L. 1999. A tabu search algorithm for the open shop scheduling problem.Computers & Operations Research,26: 109-126.

[8] Mamalis A G. andMalagardis I. 1998. Determination of due dates in job shop scheduling by simulated annealing.Computer Integrated Manufacturing Systems,9: 65-72.

[9] Dorigo M., Maniezzo V. and Colorni A. 1996. Ant System: Optimization by a Colony of Cooperating Agents. IEEE transactions on systems, man, and cybernetics, Feb, 26-1, 1996: 29-41.

[10] DorigoM. and Gambarella L.M. 1997. Ant colony for the traveling salesman problem. Biosystems, 43:73-81).

[11] Stützle T. and Dorigo M. 1999. ACO algorithms for the traveling salesman problem, in: Evolutionary Algorithms in Engineering and Computer Science,Wiley, New York, 1999: 163–183.

[12] Heinonen J. andPettersson F. 2007. Hybrid ant colony optimization and visibility studies applied to a job-shop scheduling problem. Applied Mathematics and Computation,187: 989–998.

[13] Zhong Y. 2008. Ant Colony Optimization with Local Search Applied to the Flexible Job Shop Scheduling Problems.IEEE. 2008: 1015-1020).

[14] Keqi Wang Li Li. 2009.Multi-objective Flexible Job Shop Schedule Based on Improved Ant Colony Algorithm.International Conference on Information and Automation, June.22 -25. 2009: 1183-1187)

[15] Nowicki E. and Smutnicki C. 1996.A fast table search algorithm for the job-shop problem.Management Science. 42: 97–813).

[16] Lawrence S. 1984. Resource constrained project scheduling: an experimental investigation of heuristic scheduling techniques (Supplement),Graduate School of Industrial Administration,Carnegie-Mellon University, Pittsburgh, Pennsylvania.

[17] Ferdinando P. and Emanuela M.2000. A tabu search method guided by shifting bottleneck for the job shop scheduling problem. European Journal of Operational Research,120: 297-310.

[18] GoncalvesJ.F.,MendesJ. J. D. M. and M. G. C. Resende. 2005. A hybrid genetic algorithm for the job shop scheduling problem, 167: 77–95.

[19] Sabuncuoglu I. and Bayiz,M., 1999. Job shop scheduling with beam search ..Eur. J. Oper. Res., 118: 390–412.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายสรรเพ็ชญ์ ไชยศิลปสังข์
วัน เดือน ปีเกิด	27 กุมภาพันธ์ 2531 ที่ปัตตานี
ที่อยู่	3/38 ซ.2 ถ.หนองจิก ต.สะบารัง อ.เมือง จ.ปัตตานี 94000
E-mail	sanpet1251@hotmail.com
ประวัติการศึกษา	
พ.ศ. 2546-2548	โรงเรียนเบญจมราชูทิศจังหวัดปัตตานี
พ.ศ. 2549 -2552	วิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมอิเล็กทรอนิกส์และโทรคมนาคม (เกียรตินิยมอันดับ2) มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
พ.ศ. 2553-2554	วิศวกรรมศาสตร์มหาบัณฑิต สาขาวิชาวิศวกรรมอุตสาหการ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ผลงานทางวิชาการ	“การประยุกต์ใช้วิธีขั้นตอนอาณานิคมมดในการแก้ปัญหาการจัดตารางการผลิตแบบตามสั่ง” การประชุมวิชาการข่ายงานวิศวกรรมอุตสาหการ ประจำปี 2554

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้