การพัฒนาเมตาฮิวริสติกเพื่อแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทาง การขนส่งสำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลัง ในภาคตะวันออกเฉียงเหนือของประเทศไทย

ฐิติวรดา คำแหน

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยอุบลราชธานี
พ.ศ. 2554

ลิขสิทธิ์เป็นของมหาวิทยาลัยอุบลราชธานี

DEVELOPMENT OF META-HEURISTICS TO SOLVE THE LOCATION ROUTING PROBLEM FOR THE TAPIOCA INDUSTRY IN NORTHEAST THAILAND

THITIWORADA KHUMHAN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF ENGINEERING

MAJOR IN INDUSTRIAL ENGINEERING

FACULTY OF ENGINEERING

UBON RATCHATHANI UNIVERSITY

YEAR 2011

COPYRIGHT OF UBON RATCHATHANI UNIVERSITY

ใบรับรองวิทยานิพนธ์

มหาวิทยาลัยอุบลราชธานี

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์

เรื่อง การพัฒนาเมตาฮิวริสติกเพื่อแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งสำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลังในภาคตะวันออกเฉียงเหนือของประเทศไทย

ผู้วิจัย นางสาวฐิติวรดา คำแหน

ได้พิจารณาเห็นชอบโดย

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.สมบัติ สินธุเชาวน์)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ระพีพันธ์ ปิตาคะโส)

..... กรรมการ
(ดร.ธารชดา พันธ์นิกุล)

..... กรรมการ
(ดร.รักน้อย อัครรุ่งเรืองกล)

..... คณบดี
(ผู้ช่วยศาสตราจารย์ ดร.นท แสงเทียน)

มหาวิทยาลัยอุบลราชธานี รับรองแล้ว

.....
(ผู้ช่วยศาสตราจารย์ ดร.อุทิศ อินทร์ประสิทธิ์)
รองอธิการบดีฝ่ายวิชาการ
ปฏิบัติราชการแทนอธิการบดี มหาวิทยาลัยอุบลราชธานี
ปีการศึกษา 2554

กิตติกรรมประกาศ

งานวิจัยนี้สำเร็จลุล่วงลงได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา คือ ผู้ช่วยศาสตราจารย์ ดร.สมบัติ สินธุเชาวน์ และคณะกรรมการทุกท่านอันประกอบไปด้วย ผู้ช่วยศาสตราจารย์ ดร.ระพีพันธ์ ปีตาคะโส และดร.ธารชุดา พันธ์นิกุล รวมไปถึงคำแนะนำและข้อเสนอแนะต่าง ๆ จากท่านกรรมการภายนอก คือ ดร.รักน้อย อัครรุ่งเรืองกล

ผู้วิจัยขอขอบคุณสำนักงานวิจัยแห่งชาติที่ให้ทุนอุดหนุนการวิจัยผ่านทางมหาวิทยาลัยอุบลราชธานีแก่ผู้วิจัยในครั้งนี้ นอกจากนี้ผู้วิจัยขอขอบคุณกระทรวงอุตสาหกรรม สำนักเศรษฐกิจการเกษตร กระทรวงเกษตรสหกรณ์ โรงงานแป้งมันเอี่ยมอีสานอุตสาหกรรมจำกัด บริษัทอุบลเกษตรพลังงานจำกัด และเจ้าหน้าที่ของหน่วยสนับสนุนงานวิจัยทุกท่านที่คอยช่วยดำเนินการประสานการทำงานให้สำเร็จลุล่วงไปด้วยดี และหวังเป็นอย่างยิ่งว่ารายงายวิจัยฉบับนี้จะเป็นประโยชน์แก่ผู้ที่เกี่ยวข้องและผู้ที่สนใจทุกท่าน ในการนำข้อมูลต่างๆไปใช้ให้เกิดประโยชน์ทั้งในทางปฏิบัติและทางวิชาการต่อไป

ท้ายที่สุดนี้ขอกราบขอบพระคุณ คุณพ่อคำประเสริฐ-คุณแม่วริวัลย์ คำแหน ตลอดจนทุกคนในครอบครัวคำแหน รวมถึงพี่ๆ น้องๆ ร่วมคณะวิศวกรรมศาสตร์ สาขาวิศวกรรมอุตสาหการ มหาวิทยาลัยอุบลราชธานี ที่คอยให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษาและขอขอบพระคุณคณะบุคคลดังกล่าวเป็นอย่างสูงมา ณ โอกาสนี้ด้วย

ฐิติวรดา คำแหน

(นางสาวฐิติวรดา คำแหน)

ผู้วิจัย

บทคัดย่อ

ชื่อเรื่อง : การพัฒนาเมตาฮิวริสติกเพื่อแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งสำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลังในภาคตะวันออกเฉียงเหนือของประเทศไทย

โดย : ฐิติวรดา คำแหน

ชื่อปริญญา : วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา : วิศวกรรมอุตสาหการ

ประธานกรรมการที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร. สมบัติ สินธุเชาวน์

ศัพท์สำคัญ : ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง ระบบอาณานิคมมด

ในงานวิจัยนี้ผู้วิจัยได้ทำการพัฒนาวิธีฮิวริสติกที่เรียกว่าวิธีอาณานิคมมด (Ant Colony Optimization; ACO) ที่เลียนแบบพฤติกรรมการหาอาหารของมด สำหรับการแก้ปัญหาการเลือกตำแหน่งที่ตั้งโรงงานผลิตแป้งมันสำปะหลังที่เหมาะสมพร้อมทั้งจัดเส้นทางการขนส่งสินค้าและวัตถุดิบ ขั้นตอนของวิธีอาณานิคมมดแบ่งเป็น 2 ส่วน ส่วนที่หนึ่งเป็นการหาผลเฉลยของปัญหาการเลือกสถานที่ตั้งซึ่งก็คือตำแหน่งที่ตั้งโรงงานผลิตแป้งมันสำปะหลัง และส่วนที่สองเป็นการหาผลเฉลยของปัญหาการจัดเส้นทางการขนส่ง (Vehicle Routing Problem; VRP) โดยคำตอบของปัญหาการจัดเส้นทางการขนส่งนี้จะสัมพันธ์กับตำแหน่งที่ตั้งของโรงงานผลิตแป้งมันสำปะหลัง สำหรับการปรับปรุงคำตอบของปัญหาการจัดเส้นทางการขนส่งนั้นจะประยุกต์ใช้วิธีการค้นหาคำตอบ 3 วิธี ได้แก่ การย้ายตำแหน่ง, การสลับเปลี่ยนตำแหน่ง และการสลับสองตำแหน่ง

ผู้วิจัยได้ทำการหาตำแหน่งที่ตั้งโรงงานผลิตแป้งมันสำปะหลังที่เหมาะสมพร้อมทั้งจัดเส้นทางการขนส่งสินค้าและวัตถุดิบ 6 กรณีด้วยกันคือ (1) โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิตและไม่มีข้อจำกัดด้านระยะทางการขนส่ง คำตอบของกรณีนี้คือตำแหน่งที่ตั้งของโรงงานควรอยู่ที่ ตำบลเมืองเดช อำเภอเดชอุดม จังหวัดอุบลราชธานี และระยะทางรวมในการขนส่งเป็น 26,092.1 กิโลเมตร (2) โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิตและกำหนดระยะทางในการขนส่งของรถขนส่งไม่เกิน 800 กิโลเมตรต่อวัน โดยตำแหน่งที่ตั้งของโรงงานคือตำแหน่งเดียวกันกับกรณีที่ 1 ซึ่งมีระยะทางรวมในการขนส่งเป็น 24,319.9 กิโลเมตร (3) โรงงานแป้งมันสำปะหลังทำการผลิตร้อยละ 80 ของกำลังการผลิตโดยไม่มีข้อจำกัดด้านระยะทางขนส่ง โดยตำแหน่งที่ตั้งของโรงงานคือ ตำแหน่งเดียวกันกับกรณีที่ 1 ซึ่งมีระยะทางรวมในการขนส่งเป็น

21,265.4 กิโลเมตร (4) โรงงานแป้งมันสำปะหลังทำการผลิตร้อยละ 80 ของกำลังการผลิต และกำหนดระยะทางในการขนส่งของรถขนส่งไม่เกิน 800 กิโลเมตรต่อวัน โดยตำแหน่งที่ตั้งของโรงงานคือ ตำบลบุเปือย อำเภอน้ำยืน จังหวัดอุบลราชธานี และระยะทางรวมในการขนส่งเป็น 17,168.9 กิโลเมตร (5) โรงงานแป้งมันสำปะหลังทำการผลิตร้อยละ 50 ของกำลังการผลิต และกำหนดระยะทางในการขนส่งของรถขนส่งไม่เกิน 800 กิโลเมตรต่อวันโดยตำแหน่งที่ตั้งของโรงงานคือ ตำบลเก่าขาม อำเภอน้ำยืน จังหวัดอุบลราชธานี โดยมีระยะทางรวมในการขนส่ง 10,942 กิโลเมตร และ (6) โรงงานแป้งมันสำปะหลังทำการผลิตร้อยละ 50 ของกำลังการผลิต และกำหนดระยะทางในการขนส่งของรถขนส่งไม่เกิน 800 กิโลเมตรต่อวัน โดยตำแหน่งที่ตั้งของโรงงานคือ ตำแหน่งเดียวกันกับกรณีที่ 5 ซึ่งมีระยะทางรวมในการขนส่งเป็น 8,832.26 กิโลเมตร

ABSTRACT

TITLE : DEVELOPMENT OF META-HEURISTICS TO SOLVE THE LOCATION ROUTING PROBLEM FOR THE TAPIOCA INDUSTRY IN NORTHEAST THAILAND

BY : THITIWORADA KHUMHAN

DEGREE : MASTER OF ENGINEERING

MAJOR : INDUSTRIAL ENGINEERING

CHAIR : ASST. PROF. SOMBAT SINDHUCHAO, Ph.D.

KEYWORDS : LOCATION ROUTING PROBLEM / ANT COLONY OPTIMIZATION

In this research, we develop a heuristic called Ant Colony Optimization (ACO) that simulates a food seeking behaviors of ants to solve for a suitable location of a tapioca starch plant and also construct a vehicle route for delivering products and collecting raw material. The procedure of ACO can be divided into 2 parts. The first part is to find the solution of the location problem, the location of the tapioca starch plant, and the second part is to solve the vehicle routing problem (VRP) whose solution is related to the location of the tapioca starch plant. To improve the solution of VRP, three local search methods that are One Move, Exchange and 2-Opt Move are applied.

Six cases of the problem are studied with different conditions as follows: (1) The production capacity of the tapioca starch plant is fully utilized with no travel distance constraint for a vehicle. The solution for this case is to locate the plate in Muang Det, District Det Udom, Ubonratchathani Province. The total travel distance for a fleet of vehicles is 26,092.1 kilometers. (2) The full production capacity of the plant is utilized with the maximum travel distance of 800 kilometers per day for a vehicle. For this case, the best location of the plant is the same one as in the first case and the total travel distance for a fleet of vehicle is 24,319.2 kilometers. (3) 80 percent of the production capacity of the tapioca starch plant is utilized without the travel distance constraint for a vehicle. The best solution for this situation is the same one as in the first case and the total travel distance for a fleet of vehicles is 21,265.4 kilometers. (4) 80 percent of the

production capacity of the tapioca starch plant is utilized with the maximum travel distance of 800 kilometers per day for a vehicle. The best location of the plant for this case is to open the plant in Bu Pueai, District Nam Yuen, Ubonratchathani Province and the total travel distance for a fleet of vehicle is 17,168.9 kilometers. (5) A half of the production capacity of the plant is utilized without the travel distance constraint for a vehicle. The best location of the plant for this case is to open the plant in Kao Kham, District Nam Yuen, Ubonratchathani Province and the total travel distance for a fleet of vehicles is 10,942 kilometers. (6) A half of the production capacity of the plant is utilized with the maximum travel distance of 800 kilometers per day for a vehicle. The best solution for this situation is the same one as in the fifth case and the total travel distance for a fleet of vehicle is 8,832.26 kilometers.

สารบัญ

	หน้า
กิตติกรรมประกาศ	ก
บทคัดย่อไทย	ข
บทคัดย่ออังกฤษ	ง
สารบัญ	ช
สารบัญตาราง	ญ
สารบัญภาพ	ฎ
บทที่	
1 บทนำ	
1.1 ความสำคัญและที่มาของปัญหา	1
1.2 วัตถุประสงค์ของงานวิจัย	6
1.3 ขอบเขตของการศึกษาวิจัย	6
1.4 ขั้นตอนการดำเนินงานวิจัย	6
1.5 ประโยชน์ที่คาดว่าจะได้รับ	7
1.6 คำนิยามศัพท์เฉพาะ	7
2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	
2.1 ปัญหาการเลือกสถานที่ตั้ง	8
2.2 ปัญหาการเดินทางของพนักงานขาย	11
2.3 ปัญหาการจัดเส้นทางสำหรับยานพาหนะ	13
2.4 ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง	16
2.5 วิธีการแก้ปัญหา	19
2.6 การจำลองพฤติกรรมการหาอาหารของมด	22
2.7 งานวิจัยที่เกี่ยวข้อง	25
3 วิธีดำเนินงานวิจัย	
3.1 ขั้นตอนการดำเนินงาน	30
3.2 กรณีศึกษาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งสำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลังในภาคตะวันออกเฉียงเหนือ	31

สารบัญ (ต่อ)

หน้า

3.3 แบบจำลองทางคณิตศาสตร์ 32
3.4 ระบบอาณานิคมมด 35
3.5 โครงสร้างของวิธีระบบอาณานิคมมดสำหรับการแก้ปัญหาสำหรับการเลือกสถานที่ตั้งและจัดเส้นทางสำหรับยานพาหนะ 38
3.6 คอมพิวเตอร์และโปรแกรมที่ใช้คำนวณ 51

4 วิเคราะห์ผลการวิจัย

4.1 การกำหนดค่าพารามิเตอร์ที่เกี่ยวข้อง 52
4.2 กรณีที่ 1 โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิต 53
4.3 กรณีที่ 2 โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิตและจำกัดระยะทางสูงสุด 53
4.4 กรณีที่ 3 โรงงานแป้งมันสำปะหลังทำการผลิต 80 เปอร์เซ็นต์ของกำหลังการผลิต 54
4.5 กรณีที่ 4โรงงานแป้งมันสำปะหลังทำการผลิต 80 เปอร์เซ็นต์ของกำหลังการผลิตและจำกัดระยะทางสูงสุด 54
4.6 กรณีที่ 5 โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำหลังการผลิตและจำกัดระยะทางสูงสุด 55
4.7 กรณีที่ 6 โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำหลังการผลิตและจำกัดระยะทางสูงสุด 56
4.8 กรณีที่ 7 โรงงานแป้งมันสำปะหลังทำการผลิต 20 เปอร์เซ็นต์ของกำหลังการผลิตและจำกัดระยะทางสูงสุด 56
4.7 วิเคราะห์ผลการทดลอง 57

5 สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย 61
5.2 ข้อเสนอแนะ 62
5.3 ปัญหาและอุปสรรค 62

สารบัญ (ต่อ)

หน้า

เอกสารอ้างอิง 63

ภาคผนวก

ก ข้อมูลที่ตั้งสถานบันสำปะหลังเขตภาคตะวันออกเฉียงเหนือ 71

ข ข้อมูลที่ตั้งของโรงงานที่ใช้แป้งมันสำปะหลังดิบในการผลิตสินค้าเป็นวัตถุดิบ 105

ค ข้อมูลระยะทางระหว่างลูกค้ากับแหล่งวัตถุดิบทั้งหมด 108

ง ผลการจัดเส้นทางของแต่ละกรณี 110

จ Source Code ของโปรแกรม Lingo 117

ฉ Source Code ของภาษา C++ ที่ใช้คำนวณ 119

ประวัติผู้วิจัย 144

สารบัญตาราง

ตารางที่		หน้า
2.1	การประยุกต์ใช้การจำลองพฤติกรรมการหาอาหารของมดเพื่อแก้ปัญหาวิศวกรรม	24
2.2	สรุปภาพรวมของงานวิจัยที่ใช้วิธีทางตรงและวิธีฮิวริสติกในการแก้ปัญหา LRP	28
4.1	การกำหนดค่าพารามิเตอร์ที่เกี่ยวข้องสำหรับวิธีการอาณานิคมมด	52
4.2	ผลการทดลองกรณีที่ 1	53
4.3	ผลการทดลองกรณีที่ 2	53
4.4	ผลการทดลองกรณีที่ 3	54
4.5	ผลการทดลองกรณีที่ 4	55
4.6	ผลการทดลองกรณีที่ 5	55
4.7	ผลการทดลองกรณีที่ 6	56
4.8	ผลการทดลองกรณีที่ 7	56
4.9	สรุปผลการแก้ปัญหา LRP ของกรณีศึกษาทั้ง 7 กรณี	57
4.10	ผลลัพธ์ที่ได้จากการประมวลด้วยโปรแกรม Lingo กรณีที่ 5	59
4.11	ผลลัพธ์ที่ได้จากอัลกอริทึมที่ออกแบบ กรณีที่ 5	59
ก.1	การกำหนดค่าพารามิเตอร์ที่เกี่ยวข้องสำหรับวิธีการอาณานิคมมด	72
ข.1	ข้อมูลที่ตั้งของโรงงานที่ใช้แป้งมันสำปะหลังดิบในการผลิตสินค้าเป็นวัตถุดิบที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด	106

สารบัญภาพ

ภาพที่		หน้า
1.1	ประเทศผู้ผลิตมันสำปะหลังที่สำคัญของโลก 5 อันดับแรก	2
1.2	สัดส่วนเนื้อที่เพาะปลูกมันสำปะหลังในประเทศไทย	3
1.3	โครงสร้างอุตสาหกรรมแป้งมันสำปะหลัง	4
1.4	ประเทศผู้นำเข้าแป้งมันสำปะหลังจากไทย ปี 2552	5
1.5	ภาพรวมอุตสาหกรรมแป้งมันสำปะหลัง	5
2.1	ลักษณะปัญหาการเดินทางของพนักงานขาย	11
2.2	ลักษณะปัญหาการจัดเส้นทางสำหรับยานพาหนะ	13
2.3	รูปแบบของปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง	16
3.1	ลำดับขั้นตอนการดำเนินงานวิจัย	30
3.2	ธรรมชาติการเดินทางของมด	35
3.3	รหัสเทียมของวิธีอาณานิคมมดสำหรับปัญหา LRP	39
3.4	Flow Chart ของวิธีอาณานิคมมดสำหรับปัญหา LRP	40
3.5	การสร้างคำตอบเริ่มต้นของมดงานในปัญหาการจัดเส้นทางสำหรับยานพาหนะ	45
3.6	การปรับปรุงคุณภาพคำตอบด้วยวิธีการย้ายตำแหน่ง	47
3.7	การปรับปรุงคุณภาพคำตอบด้วยวิธีการสลับเปลี่ยนตำแหน่ง	48
3.8	การปรับปรุงคุณภาพคำตอบด้วยวิธีการสลับสองตำแหน่ง	49
4.1	ตัวอย่างกรณีที่ 5 จำนวนตัวแปรมีจำนวนมากไม่สามารถประมวลผลหาคำตอบได้	58
4.2	ระยะเวลาและผลที่ได้จากการหาคำตอบด้วยโปรแกรม Lingo กรณีที่ 7	59
ค.1	ข้อมูลระยะทางระหว่างลูกค้ากับแหล่งวัดอุดิบทั้งหมด (ตัวอย่าง)	109

บทที่ 1
บทนำ

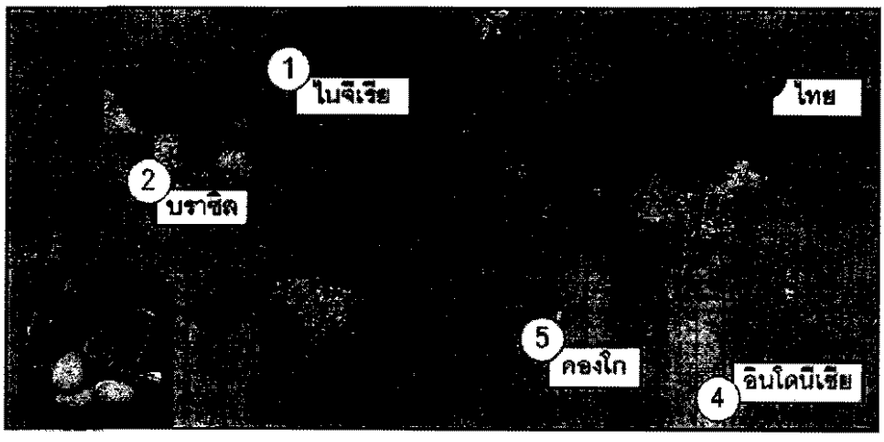
ในบทแรกนี้จะนำเสนอถึง ความสำคัญและที่มาของปัญหาที่เลือกทำวิจัย วัตถุประสงค์และขอบเขตของงานวิจัย แผนการดำเนินงานวิจัย รวมไปถึงประโยชน์ที่คาดว่าจะได้รับ เพื่อให้ทราบถึงภาพรวมและแนวความคิดเบื้องต้นของงานวิจัยนี้

1.1 ความสำคัญและที่มาของปัญหา

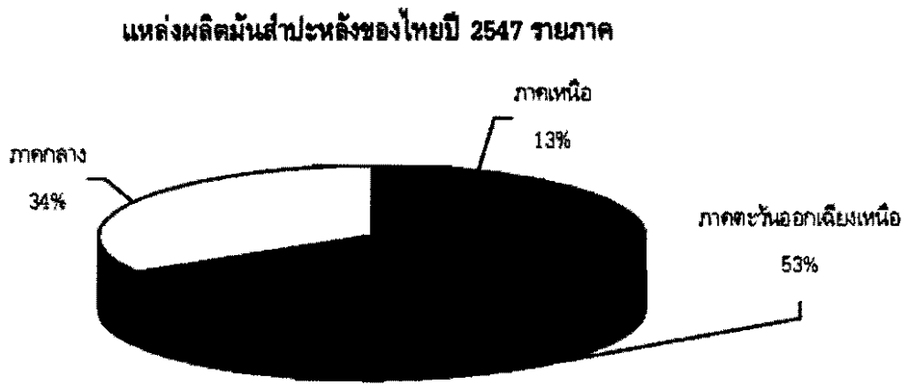
จากที่ได้เกิดภาวะวิกฤตเศรษฐกิจโลกอย่างรุนแรง เนื่องมาจากวิกฤตสถาบันการเงินในสหรัฐและยุโรป ส่งผลให้เกิดภาวะวิกฤตเศรษฐกิจแพร่กระจายไปยังประเทศต่างๆ ทั่วทุกภูมิภาคของโลก ประเทศไทยก็เป็นประเทศหนึ่งที่ได้รับผลกระทบจากวิกฤตเศรษฐกิจดังกล่าว จากผลกระทบดังกล่าวทำให้แต่ละองค์กรในทุกภาคส่วนรวมไปจนถึงภาคอุตสาหกรรมต่างดิ้นรนทำทุกวิถีทางเพื่อให้องค์กรรอดพ้นจากวิกฤตในครั้งนี้ อีกทั้งโลกของธุรกิจการค้ามีการพัฒนาไปอย่างรวดเร็ว โดยเฉพาะอย่างยิ่งการค้าในยุคของโลกาภิวัฒน์ที่การสื่อสารโทรคมนาคมสามารถสื่อสารกันได้อย่างไร้พรมแดน เป็นเหตุให้การแข่งขันทางภาคอุตสาหกรรมนั้นนับวันยิ่งจะทวีความรุนแรงมากยิ่งขึ้น ดังนั้นการลงทุนในภาคอุตสาหกรรมจึงจำเป็นต้องพิจารณาในหลายๆ ด้านอย่างรอบคอบ รวมทั้งต้องมีการวางแผนจัดการที่รัดกุม โดยเฉพาะนักลงทุนที่ต้องการลงทุนในการเปิดกิจการหรือโรงงานใหม่ เพราะต้องพิจารณาถึงความเสี่ยงและความเป็นไปได้ของโครงการ หลังจากพิจารณาถึงความเป็นไปได้ของโครงการแล้วขั้นตอนที่สำคัญลำดับถัดไปคือ การพิจารณาเลือกตำแหน่งที่ตั้งของโรงงาน เนื่องจากปัญหาการเลือกสถานที่ตั้งโรงงาน เป็นปัญหาแรกและเป็นปัญหาใหญ่ของการสร้างโรงงาน การเลือกสถานที่ตั้งโรงงานเป็นเรื่องการผูกพันระยะยาว หากมีข้อบกพร่องหรือข้อผิดพลาดจะเป็นการยากต่อการแก้ไข อีกทั้งยังมีผลกระทบต่อรายได้ที่จะได้รับ ค่าใช้จ่ายในการปฏิบัติการซึ่งรวมทั้งค่าใช้จ่ายคงที่และแปรผัน หากเลือกสถานที่ตั้งโรงงานได้เหมาะสมก็ย่อมส่งผลต่อศักยภาพในการแข่งขันขององค์กรที่เพิ่มสูงตามไปด้วย อย่างไรก็ตามการจัดตั้งโรงงานจำเป็นต้องพิจารณาปัจจัยหลายด้านไม่ว่าจะเป็นสถานที่ตั้งซึ่งต้องคำนึงถึงความคุ้มค่าทางเศรษฐศาสตร์ รวมถึงการกำหนดเส้นทางในการขนส่งวัตถุดิบจากแหล่งวัตถุดิบ ตลอดจนการขนส่งผลิตภัณฑ์ไปยังผู้บริโภค หรือที่รวมเรียกว่า "การจัดการด้านโลจิสติกส์"

การจัดการทางด้านโลจิสติกส์มีความสำคัญต่อการดำเนินธุรกิจ ซึ่งกิจกรรมด้านการขนส่งก็เป็นส่วนหนึ่งในกิจกรรมหลักของงานทางด้านโลจิสติกส์ที่ส่งผลต่อต้นทุนของสินค้าโดยตรง อีกทั้งมีผลกระทบกับความพึงพอใจของลูกค้าในด้านระยะเวลาการส่งมอบ ผู้รับผิดชอบต้องตัดสินใจว่าจะจัดสินค้าของลูกค้ารายใดไว้ที่พาหนะคันใด และตารางการเดินทางของพาหนะแต่ละคันจะเดินทางไปลูกค้าแต่ละรายโดยมีลำดับการจัดส่งอย่างไรจึงจะเกิดประสิทธิภาพสูงที่สุด

ประเทศไทยผลิตมันสำปะหลังมากเป็นอันดับ 3 ของโลก รองจากประเทศไนจีเรียและประเทศบราซิล แสดงดังภาพที่ 1.1 มันสำปะหลังเป็นสินค้าเกษตรที่มีความสำคัญชนิดหนึ่งของโลก มีปริมาณการผลิตกว่า 200 ล้านตันต่อปี โดยการผลิตมีแนวโน้มขยายตัวเพิ่มขึ้นอย่างต่อเนื่อง ซึ่งผลผลิตมันสำปะลังส่วนใหญ่กว่าร้อยละ 80 ใช้บริโภคโดยตรงหรือใช้ในอุตสาหกรรมอาหารและอาหารสัตว์ ด้านการค้ามันสำปะหลังและผลิตภัณฑ์ในตลาดโลกหลักๆ จะอยู่ในรูปของผลิตภัณฑ์แปรรูปขั้นพื้นฐาน ได้แก่ แป้งมันสำปะหลัง และมันสำปะหลังเส้นหรืออัดเม็ด โดยโครงสร้างตลาดของผลิตภัณฑ์แป้งมันสำปะหลัง มีผู้ส่งออกสำคัญ ได้แก่ ไทย และเวียดนาม สำหรับประเทศไทยมันสำปะหลังนับเป็นพืชเศรษฐกิจเชิงพาณิชย์ที่สำคัญ มีปริมาณการผลิตมากกว่า 20 ล้านตันในแต่ละปีคิดเป็นประมาณ 12% ของปริมาณผลิตรวมทั่วโลก โดยใช้ภายในประเทศคิดเป็นประมาณร้อยละ 30 ส่วนที่เหลือส่งออกไปยังตลาดโลก ซึ่งไทยเองเป็นผู้ส่งออกมันสำปะหลังและผลิตภัณฑ์อยู่ในอันดับที่หนึ่งในตลาดโลกมาอย่างต่อเนื่อง ซึ่งประเทศไทยมีพื้นที่เพาะปลูกประมาณ 8 ล้านไร่ใน 45 จังหวัด เนื้อที่เพาะปลูกมันสำปะหลังมีการกระจายอยู่ทั่วประเทศ (ยกเว้นภาคใต้) ภาคตะวันออกเฉียงเหนือมีเนื้อที่เพาะปลูกและเก็บเกี่ยวมากที่สุดรองลงมาได้แก่ภาคกลาง และภาคเหนือตามลำดับ แสดงดังภาพที่ 1.2 โดยมีเกษตรกรผู้ปลูกมันสำปะหลังมากกว่า 5 แสนครัวเรือนทั่วประเทศ



ภาพที่ 1.1 ประเทศผู้ผลิตมันสำปะหลังที่สำคัญของโลก 5 อันดับแรก (FAO, 2554)



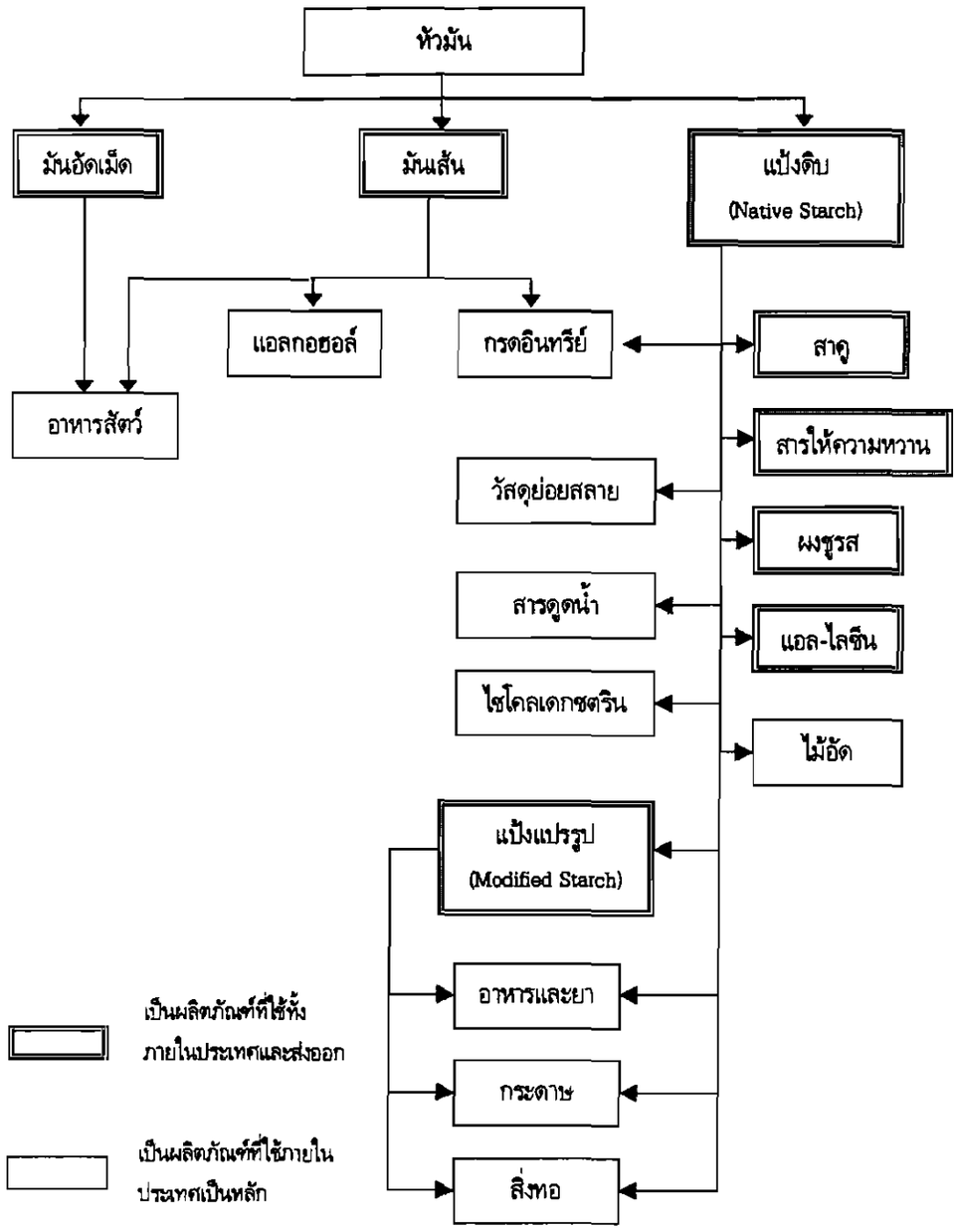
ภาพที่ 1.2 สัดส่วนเนื้อที่เพาะปลูกมันสำปะหลังในประเทศไทย (ที่มา : กลุ่มทำงานศึกษาและวิเคราะห์สินค้าเกษตรประเภทมันสำปะหลัง, 2554)

การนำมันสำปะหลังไปใช้ประโยชน์ส่วนใหญ่ผลผลิตจะถูกนำไปแปรรูปเป็นผลิตภัณฑ์ขั้นพื้นฐานก่อนนำไปบริโภคหรือส่งต่อไปใช้ในอุตสาหกรรมที่เกี่ยวข้องแบ่งเป็นผลิตภัณฑ์ 2 ชนิด หลักคือ (1) มันสำปะหลังเส้นหรืออัดเม็ด เป็นผลิตภัณฑ์แปรรูปสำหรับนำไปใช้ต่อในอุตสาหกรรม เช่น อาหารสัตว์ หรือนำไปผลิตแอลกอฮอล์สำหรับบริโภคหรือใช้เป็นพลังงานทดแทน (เอทานอล) เป็นต้น (2) แป้งมันสำปะหลัง เป็นผลิตภัณฑ์แปรรูปสำหรับนำไปบริโภคเป็นอาหารโดยตรงหรือนำไปใช้ในอุตสาหกรรมที่เกี่ยวข้อง โดยแป้งมันสำปะหลังแบ่งออกเป็น 2 ชนิดได้แก่

1.1.1. แป้งมันปะหลังดิบ (Native Starch) คือ แป้งที่เกิดจากการแปรรูปหัวมันสำปะหลังขั้นต้นโดยไม่มีการใช้เทคโนโลยีขั้นสูงร่วมในการผลิต เช่น การนำหัวมันมาบดให้เป็นแป้ง

1.1.2. แป้งมันสำปะหลังแปรรูป (Modified Starch) คือ แป้งที่ได้จากการนำแป้งดิบไปผ่านกระบวนการปรับเปลี่ยนโครงสร้างทางโมเลกุลเพื่อให้มีคุณสมบัติเฉพาะ เช่น คุณสมบัติทางด้านความเหนียว สำหรับนำไปใช้ในอุตสาหกรรมแต่ละประเภท แป้งแปรรูปเป็นผลิตภัณฑ์ที่สร้างมูลค่าเพิ่มให้แป้งมันสำปะหลังดิบเพราะราคาสูงกว่าแป้งดิบกว่าเท่าตัว

แป้งมันสำปะหลังสามารถแปลงไปเป็นอุตสาหกรรมต่อเนื่องได้หลากหลาย เช่น อาหาร, กระดาษ, สิ่งทอ, สารความหวาน, ยา, อาหารสัตว์ และพลังงาน แสดงดังภาพที่ 1.3



ภาพที่ 1.3 โครงสร้างอุตสาหกรรมแป้งมันสำปะหลัง (ที่มา : กลุ่มทำงานศึกษาและวิเคราะห์สินค้าเกษตรประเภทมันสำปะหลัง, 2554)

ประเทศไทยส่งออกแป้งมันสำปะหลังไปยังตลาดโลก (พ.ศ. 2552) ทั้ง 2 ชนิด รวมกันประมาณ 2.6 ล้านตัน โดยมีตลาดส่งออกที่สำคัญแบ่งตามชนิดของแป้งมันสำปะหลังแสดงดังภาพที่ 1.4

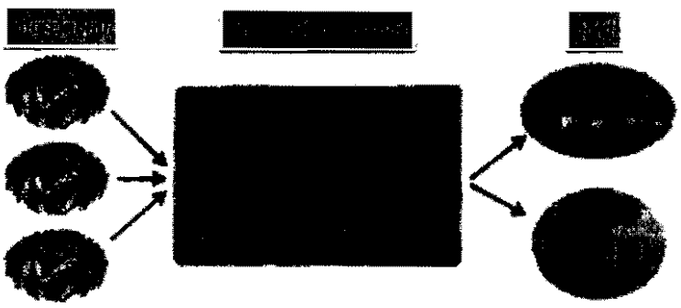
Top 10 Importers of Thai Tapioca Starch in 2009

1	CHINA	500,943	4,613
2	TAIWAN	313,365	2,756
3	INDONESIA	241,668	2,280
4	MALAYSIA	188,351	1,674
5	JAPAN	123,214	1,152
6	PHILIPPINES	70,027	657
7	SINGAPORE	50,644	468
8	BANGLADESH	37,275	317
9	KOREA,R	36,893	342
10	USA	31,715	375
	OTHERS	204,005	2,017

ภาพที่ 1.4 ประเทศผู้นำเข้าแป้งมันสำปะหลังจากไทย ปี 2552 (กลุ่มทำงานศึกษาและวิเคราะห์สินค้าเกษตรประเภทมันสำปะหลัง, 2554)

จากข้อมูลเบื้องต้นทั้งด้านพื้นที่การเพาะปลูก ปริมาณเกษตรกร ความต้องการของผู้บริโภค ประเทศไทยยังสามารถเพิ่มจำนวนโรงงานอุตสาหกรรมผลิตแป้งมันสำปะหลังได้อีก ทั้งนี้อาจจะต้องอาศัยปัจจัยต่างๆ มากมาย เช่น แหล่งวัตถุดิบ คุณภาพวัตถุดิบ การขนส่งหรือแม้กระทั่งทำเลที่ตั้งโรงงาน ซึ่งล้วนแล้วแต่เป็นปัจจัยสำคัญ

ดังนั้นงานวิจัยนี้จึงศึกษาเกี่ยวกับอุตสาหกรรมแป้งมันสำปะหลังภาพรวมแสดงดังภาพที่ 1.5 ซึ่งเกี่ยวข้องตั้งแต่เรื่องวัตถุดิบไปจนถึงการขนส่งแป้งมันที่ผลิต โดยทำการออกแบบอัลกอริทึมเพื่อแก้ปัญหาการเลือกสถานที่ตั้งของโรงงานผลิตแป้งมันสำปะหลังดิบ (Native Starch) และจัดเส้นทางการขนส่งสินค้าและวัตถุดิบ สำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลัง เพื่อให้ต้นทุนด้านเศรษฐศาสตร์มีค่าต่ำสุดหรือเหมาะสมที่สุด และเป็นไปตามข้อจำกัดต่างๆ



ภาพที่ 1.5 ภาพรวมอุตสาหกรรมแป้งมันสำปะหลัง

1.2 วัตถุประสงค์ของงานวิจัย

เพื่อออกแบบวิธีการที่สามารถใช้ในการแก้ปัญหาการเลือกตำแหน่งที่ตั้งโรงงานผลิตแป้งมันสำปะหลังที่เหมาะสมพร้อมทั้งจัดเส้นทางการขนส่งสินค้าและวัตถุดิบ ให้มีประสิทธิภาพมากที่สุด ภายใต้วัตถุประสงค์ด้านเศรษฐศาสตร์

1.3 ขอบเขตของการศึกษาวิจัย

1.3.1 งานวิจัยนี้ทำการศึกษาเพื่อแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งสำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลังเฉพาะในภาคตะวันออกเฉียงเหนือของประเทศไทยเท่านั้น

1.3.2 การเลือกสถานที่ตั้งโรงงานผลิตแป้งมันจะเลือกจากตำแหน่งที่ตั้งลานมัน 261 แห่ง เพียงหนึ่งแห่งเท่านั้น

1.3.3 การทดสอบและเปรียบเทียบอาศัยการสร้างแบบจำลองและคำนวณด้วยโปรแกรมคอมพิวเตอร์ โดยไม่มีการทดสอบกับสถานที่และยานพาหนะจริง

1.3.4 ประยุกต์ใช้วิธีฮิวริสติก (Heuristics) และวิธีเมต้าฮิวริสติก (Meta-Heuristics) ในการออกแบบวิธีการแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง

1.3.5 การจัดเส้นทางการขนส่ง รถขนส่งจะต้องเดินทางไปส่งสินค้าให้กับลูกค้าก่อนแล้วจึงเดินทางไปรับวัตถุดิบจากลานมัน

1.4 ขั้นตอนการดำเนินงานวิจัย

1.4.1 รวบรวมข้อมูลที่เกี่ยวข้องกับการวิจัย เช่น ข้อมูลความสามารถในการส่งหัวมันให้กับโรงงานแป้งมันของลานมัน ความต้องการของลูกค้า ความจุของรถบรรทุก กำลังการผลิตของโรงงานแป้งมัน ระยะทางระหว่างลานมันแต่ละแห่ง ระยะทางระหว่างลูกค้าแต่ละราย และระยะทางระหว่างลานมันกับลูกค้า เป็นต้น

1.4.2 ศึกษางานวิจัยที่เกี่ยวข้อง

1.4.3 สร้างสมการทางคณิตศาสตร์

1.4.4 ออกแบบอัลกอริทึมเพื่อใช้ในการแก้ปัญหา

1.4.5 เขียนและทดสอบโปรแกรมคอมพิวเตอร์ที่ประยุกต์ใช้ด้วยภาษา C++

1.4.6 คำนวณหาสถานที่ตั้งของโรงงานพร้อมทั้งจัดเส้นทางการขนส่งแป้งมันสำปะหลังรวมถึงเส้นทางการรับวัตถุดิบจากลานมัน

1.4.7 เปรียบเทียบระยะทางรวมของอัลกอริทึมกับโปรแกรม Lingo ซึ่งในส่วนของโปรแกรม Lingo นั้นจะแก้ปัญหาการจัดเส้นทางสำหรับยานพาหนะเท่านั้น โดยใช้ข้อมูลตำแหน่งที่ตั้งโรงงาน ตำแหน่งส่งสินค้าและตำแหน่งรับวัตถุดิบที่ได้จากการหาคำตอบของอัลกอริทึม

1.4.8 สรุปผลการวิจัยและเรียบเรียงผลงานทั้งหมดพร้อมทั้งจัดทำรูปเล่มวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ได้สถานที่ตั้งโรงงานผลิตแป้งมันสำปะหลังและเส้นทางการขนส่งที่มีความคุ้มค่าในการลงทุนสำหรับผู้สนใจที่จะลงทุนสร้างโรงงานผลิตแป้งมันสำปะหลัง

1.5.2 ได้อัลกอริทึมที่มีประสิทธิภาพสำหรับการแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งสำหรับกรณีที่ต้องไปส่งสินค้าให้กับลูกค้าก่อนแล้วจึงเดินทางไปรับวัตถุดิบ

1.5.3 สามารถนำวิธีการแก้ปัญหาที่ได้จากการวิจัยไปเผยแพร่ เพื่อเป็นแนวทางในการพัฒนาหรือแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งลักษณะอื่นๆ ต่อไปในอนาคต

1.5.4 ข้อมูลที่ได้จากการวิจัย สามารถเป็นฐานข้อมูลให้กับอุตสาหกรรมจังหวัด เพื่อให้ผู้ประกอบการใช้ในการตัดสินใจเกี่ยวกับการตั้งโรงงานผลิตแป้งมัน

1.6 คำนิยามศัพท์เฉพาะ

1.6.1 Facility Location Problem หมายถึง ปัญหาการเลือกสถานที่ตั้ง

1.6.2 Traveling Salesman Problem หมายถึง ปัญหาการเดินทางของพนักงานขาย

1.6.3 Vehicle Routing Problem หมายถึง ปัญหาการจัดเส้นทางสำหรับยานพาหนะ

1.6.4 Location Routing Problem หมายถึง ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง

1.6.5 Exact Optimization หมายถึง วิธีการหาคำตอบที่ดีที่สุด

1.6.6 Heuristic Optimization หมายถึง วิธีการหาคำตอบแบบฮิวริสติก

1.6.7 Ant System หมายถึง วิธีระบบมด

1.6.8 Elitist Ant System หมายถึง วิธีระบบมดที่ดีที่สุด

1.6.9 Max-Min Ant System หมายถึง วิธีระบบมดแบบกำหนดขอบเขตสูงสุด-ต่ำสุด

1.6.10 Rank-Base Ant System หมายถึง วิธีระบบมดแบบเรียงลำดับ

1.6.11 Ant Colony Optimization หมายถึง ระบบอาณานิคมมด

1.6.12 Meta-Heuristic หมายถึง วิธีเมต้าฮิวริสติก

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะนำเสนอถึงลักษณะของปัญหาการเลือกสถานที่ตั้ง ปัญหาการเดินทางของพนักงานขาย ปัญหาการจัดเส้นทางสำหรับยานพาหนะ ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง วิธีการหาคำตอบที่ดีที่สุดและวิธีฮิวริสติก รวมทั้งงานวิจัยที่เกี่ยวข้องกับปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง

ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง (Location Routing Problem) ตามโครงสร้างสามารถแบ่งออกเป็น 2 ปัญหา คือ ปัญหาการเลือกสถานที่ตั้ง (Facility Location Problem) และปัญหาการจัดเส้นทางการขนส่ง (Routing Problem)

2.1 ปัญหาการเลือกสถานที่ตั้ง (Facility Location Problem: LP)

ปัญหาการเลือกสถานที่ตั้งหมายถึงการวางตำแหน่งที่ตั้งของสิ่งปลูกสร้างเพื่อให้ระยะทางในการขนส่งน้อยที่สุดหรือทำให้ค่าขนส่งต่ำสุด เช่น การตั้งศูนย์กระจายสินค้าเพื่อทำให้ค่าใช้จ่ายรวมในการขนส่งสินค้าต่ำที่สุด การหาตำแหน่งที่ตั้งสถานที่เก็บวัตถุอันตรายเพื่อให้อยู่ห่างจากแหล่งชุมชนมากที่สุด หรือการตั้งศูนย์กู้ชีพฉุกเฉินเพื่อให้ระยะเวลาในการตอบสนองต่อสถานการณ์ฉุกเฉินต่ำ เป็นต้น

Owen and Daskin (1998) ได้จำแนกลักษณะของปัญหาการเลือกสถานที่ตั้งตามลักษณะของฟังก์ชันวัตถุประสงค์ออกเป็น 3 ประเภท คือ ปัญหาปกคลุม (Covering Problems) ปัญหาระยะทางเฉลี่ย (Average Distance Problems) และปัญหาศูนย์กลาง (Center Problems) ซึ่งปัญหาการเลือกสถานที่ตั้งแบบระยะทางเฉลี่ยถูกนำเสนอครั้งแรกโดย Hakimi (1964) มีชื่อเรียกว่า P-median Problem รูปแบบทางคณิตศาสตร์ของปัญหาแบบ P-median เป็นดังนี้

ดัชนี (Indices)

i ลำดับของแหล่งวัตถุดิบ โดยที่ $i = 1,2,\dots,I$

j ลำดับของจุดที่มีศักยภาพในการตั้งโรงงาน โดยที่ $j = 1,2,\dots,J$

ตัวแปรที่ทราบค่า (Parameter)

P เป็นจำนวนโรงงานที่จะเปิด

D_{ij} ระยะทางระหว่างโหนดที่ i ไปยังโหนดที่ j

ตัวแปรในการตัดสินใจ (Decision Variables)

$X_{ij} =$ 1, ถ้าแหล่งวัตถุดิบ i ส่งวัตถุดิบให้โรงงาน j

0, กรณีอื่นๆ

$Y_j =$ 1, ถ้าจุดที่มีศักยภาพในการตั้งโรงงาน j เปิดเป็นโรงงาน

0, กรณีอื่นๆ

ฟังก์ชันวัตถุประสงค์ (Objective Function)

$$Min = \sum_{i=1}^{I}\sum_{j=1}^{J} D_{ij}X_{ij} \tag{2.1}$$

สมการข้อจำกัด (Constraints)

$$\sum_{j=1}^{J} X_{ij} \leq 1 \qquad \forall_{i=1,2,..,I} \tag{2.2}$$

$$X_{ij} \leq y_j \qquad \forall_{i=(1,2,...,I),j=(1,2,..,J)} \tag{2.3}$$

$$\sum_{j=1}^{J} Y_j = P \tag{2.4}$$

$$X_{ij}\epsilon\{0,1\}, Y_j\epsilon\{0,1\} \qquad \forall_{i=(1,2,...,I),j=(1,2,..,J)} \tag{2.5}$$

ฟังก์ชันวัตถุประสงค์ (2.1) เพื่อหาค่าใช้จ่ายในการเดินทางต่ำสุด สมการเงื่อนไขที่ (2.2) เป็นการประกันว่าแหล่งวัตถุดิบจะส่งวัตถุดิบให้กับโรงงานเพียงแห่งเดียวเท่านั้น สมการที่ (2.3) เป็นการประกันว่าแหล่งวัตถุดิบจะไม่ส่งวัตถุดิบให้โรงงานที่ไม่ได้เปิด สมการที่ (2.4) จำนวนโรงงานที่เปิดเท่ากับ P แห่ง สมการเงื่อนไขสุดท้าย (2.5) เป็นการกำหนดตัวแปรตัดสินใจแบบไบนารี่

ในอีก 25 ปีต่อมา Cornuejols et al. (1990) พบว่าปัญหาของ Hakimi ยังมีจุดด้อยที่ไม่ได้พิจารณาต้นทุนคงที่ในการสร้างศูนย์กระจายสินค้าในฟังก์ชันวัตถุประสงค์ ซึ่งหากทำการเพิ่มต้นทุนคงที่ในการสร้างศูนย์กระจายสินค้าเข้าไปในฟังก์ชันวัตถุประสงค์ ชื่อปัญหาจะเปลี่ยนเป็น Uncapacitated Facility Location Problem โดยต้นทุนรวมทั้งหมดจะประกอบไปด้วยต้นทุนคงที่ในการสร้างศูนย์กระจายสินค้าและต้นทุนค่าขนส่ง รูปแบบทางคณิตศาสตร์ที่เพิ่มขึ้นมาจะเป็นดังนี้

ฟังก์ชันวัตถุประสงค์ (Objective Function)

$$Min = \sum_{i=1}^{I}\sum_{j=1}^{J} D_{ij}X_{ij} + \sum_{j=1}^{J} F_j Y_j \tag{2.6}$$

สมการข้อจำกัด (Constraints)

$$\sum_{j=1}^{J} X_{ij} \leq 1 \qquad \forall_{i=1,2,..,I} \tag{2.7}$$

$$X_{ij} \leq y_j \qquad \forall_{i=(1,2,...,I),j=(1,2,..,J)} \tag{2.8}$$

$$\sum_{j=1}^{J} Y_j = P \tag{2.9}$$

$$X_{ij}\epsilon\{0,1\}, Y_j\epsilon\{0,1\} \qquad \forall_{i=(1,2,...,I),j=(1,2,..,J)} \tag{2.10}$$

$$\sum_{i=1}^{I} X_{ij}a_i \leq C_j \qquad \forall_{j=1,2,..J} \tag{2.11}$$

ส่วนที่เพิ่มเติมจากรูปแบบทางคณิตศาสตร์ของ Hakimi ได้แก่

ตัวแปรที่ทราบค่า (Parameter)

F_j ค่าใช้จ่ายในการเปิดโรงงานที่ j

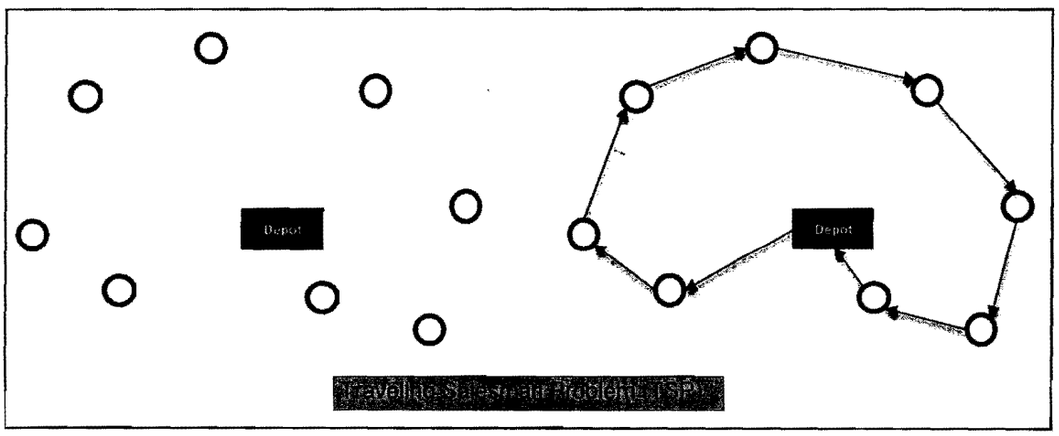
C_j ความสามารถที่โรงงาน j สามารถรับวัตถุดิบได้ต่อวัน

a_i ปริมาณวัตถุดิบที่มีในแหล่งวัตถุดิบ i

สมการวัตถุประสงค์จะบวกในส่วนของค่าใช้จ่ายในการเปิดโรงงานเข้าไปดังสมการที่ (2.6) และเพิ่มสมการเงื่อนไขที่ (2.11) คือสมการที่ประกันว่าโรงงานสามารถรับวัตถุดิบได้ไม่เกินความสามารถที่โรงงานรับวัตถุดิบได้ต่อวัน

2.2 ปัญหาการเดินทางของพนักงานขาย (Traveling Salesman Problem: TSP)

ปัญหาการเดินทางของพนักงานขาย (Traveling Salesman Problem: TSP) เป็นปัญหาที่เกี่ยวข้องกับการหาเส้นทางในการเดินทางของพนักงานขายไปยังเมืองต่างๆ โดยพนักงานขายจะต้องเดินทางไปยังเมืองต่างๆ จำนวน n เมือง ในการเดินทางของพนักงานขายจะต้องเดินทางกลับมายังจุดเริ่มต้นเป็นจุดสุดท้าย โดยมีเป้าหมายคือ การหาเส้นทางในการเดินทางไปยังเมืองต่างๆ ที่สั้นที่สุด เพื่อให้เกิดประสิทธิภาพสูงสุดในส่วนของระยะทาง, เวลาและค่าใช้จ่ายในการเดินทาง โดยขนาดของคำตอบที่เป็นไปได้ทั้งหมดจะมีค่าเท่ากับ (n-1)!/2 คำตอบ เมื่อ n คือจำนวนเมืองที่พนักงานขายจะต้องเดินทางผ่าน โดยขนาดของคำตอบที่เป็นไปได้ทั้งหมดจะเพิ่มมากขึ้นตามจำนวนเมืองที่เพิ่มขึ้น ซึ่งทำให้การหาค่าคำตอบเส้นทางที่สั้นที่สุดมีความยากเพิ่มขึ้นตามจำนวนเมืองที่เพิ่มมากขึ้นเช่นกัน ยกตัวอย่างเช่น ถ้าพนักงานขายจะต้องเดินทางไปให้ครบทั้ง 7 เมืองและกลับมายังจุดเริ่มต้นโดยมีระยะทางโดยรวมสั้นที่สุด จะมีเส้นทางที่เป็นไปได้ทั้งหมด (7-1)!/2 = 360 เส้นทาง ในขณะที่ถ้าจำนวนเมืองที่จะต้องเดินทางไปเพิ่มขึ้นเป็น 100 เมือง จะมีเส้นทางที่เป็นไปได้ทั้งหมดสูงถึง 4.67×10^{155} เส้นทาง ลักษณะของปัญหา TSP สามารถแสดงได้ดังภาพที่ 2.1



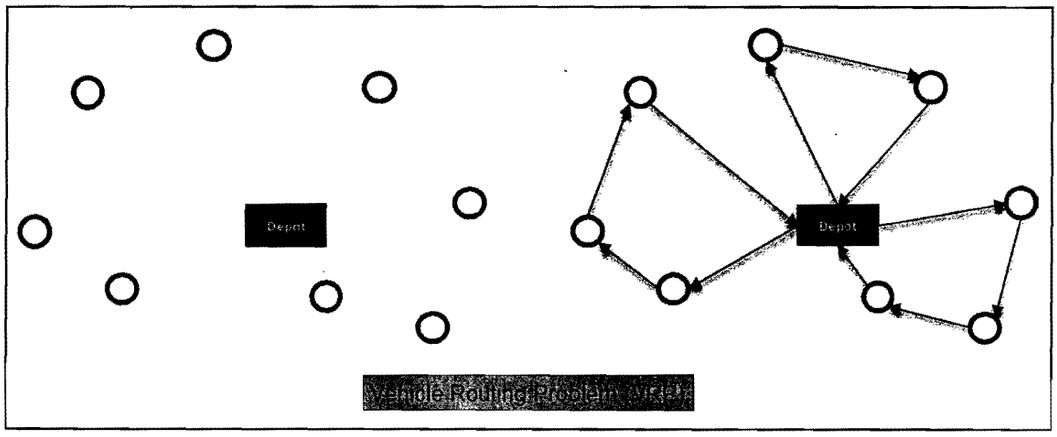
ภาพที่ 2.1 ลักษณะปัญหาการเดินทางของพนักงานขาย

ปัญหา TSP เป็นปัญหาประเภท Nondeterministic Polynomial-Time Hard (NP-hard) ซึ่งเป็นปัญหาที่หาค่าคำตอบที่ดีที่สุดได้ค่อนข้างยาก โดยจำนวนคำตอบที่เป็นไปได้จะเพิ่มขึ้นแบบเอ็กโปเนลเชียลตามจำนวนจุดของสถานที่ที่เพิ่มมากขึ้น และในการหาคำตอบยังต้องใช้เวลาในการคำนวณเพิ่มมากขึ้นตามจำนวนจุดของสถานที่ที่เพิ่มขึ้นด้วยเช่นกัน ลักษณะของปัญหา TSP แสดงด้วยแบบจำลองทางคณิตศาสตร์ได้ดังนี้

ฟังก์ชันวัตถุประสงค์ที่ (2.12) คือ ต้องการหาเส้นทางที่ให้ค่าระยะทางรวมต่ำสุด สมการข้อจำกัดที่ (2.13) แสดงว่าที่จุดปลายทางใดๆ จะมีจุดเริ่มต้นเพียงจุดเดียวเท่านั้น สมการ (2.14) แสดงถึงการเดินทางจากจุดเริ่มต้นใดๆ จะมีจุดหมายปลายทางเพียงแห่งเดียวเท่านั้น สมการ (2.15) เป็นการจำกัดการเดินทางย่อย สมการ (2.16) บังคับให้ตัวแปร X_{ij} เป็นค่าไบนารี่เท่านั้น

2.3 ปัญหาการจัดเส้นทางสำหรับยานพาหนะ (Vehicle Routing Problem: VRP)

ปัญหาการจัดเส้นทางสำหรับยานพาหนะ (Vehicle Routing Problem: VRP) เป็นปัญหาที่เกี่ยวข้องกับการขนส่งสินค้าระหว่างจุดกระจายสินค้า (Depot) ไปยังลูกค้าโดยใช้พาหนะในการขนส่ง เช่น การขนส่งน้ำดื่ม การขนส่งแก๊ส การส่งไปรษณีย์ การจัดเส้นทางการเก็บขยะ ฯลฯ โดยมีวัตถุประสงค์ในการจัดเส้นทางการขนส่งดังกล่าวให้มีประสิทธิภาพมากที่สุด การจัดเส้นทางการขนส่งที่เหมาะสมสำหรับยานพาหนะนั้นจะมุ้งเน้นไปที่การหาเส้นทางที่ดีที่สุดในการส่งสินค้าหรือวัตถุดิบให้กับลูกค้าทุกรายด้วยยานพาหนะที่มีอยู่ ภายใต้เงื่อนไขข้อจำกัดในการดำเนินงาน เช่น ความจุของรถบรรทุก เวลาในการทำงานของพนักงาน เวลาในการส่งของ เป็นต้น ปัญหา VRP ก็คือ m-TSP นั่นเอง โดยจะเป็นปัญหาในการจัดเส้นทาง m เส้นทาง ที่เริ่มจากศูนย์กระจายสินค้าไปยังจุดส่งสินค้าในที่ต่างๆ โดยจุดส่งแต่ละจุดสามารถไปได้เพียงครั้งเดียวด้วยรถคันเดียว และปริมาณความต้องการสินค้าของจุดส่งในแต่ละเส้นทางจะต้องไม่เกินความจุของรถที่ใช้ในการขนส่ง ซึ่งเส้นทางที่ได้มานั้นจะต้องมีค่าใช้จ่ายต่ำที่สุดเท่าที่จะเป็นไปได้ ลักษณะของปัญหา VRP สามารถแสดงได้ดังภาพที่ 2.2



ภาพที่ 2.2 ลักษณะปัญหาการจัดเส้นทางสำหรับยานพาหนะ

ดัชนี (Indices)

i	ลำดับของเมืองที่ i โดยที่ $i = 1,2,\dots,N$
j	ลำดับของเมืองที่ j โดยที่ $j = 1,2,\dots,N$
V	เซตของตำแหน่งที่ตั้งของเมืองทั้งหมดที่พิจารณา โดยที่ $V = \{1,2,\dots,N\}$
S	สับเซตใดๆ ที่เป็นไปได้ทั้งหมดของเซต V: $S \subset V$ โดยที่ $S \neq \emptyset, S \neq V$
$\|S\|$	จำนวนสมาชิกของเซต S

ตัวแปรที่ทราบค่า (Parameter)

C_{ij}	ระยะทางระหว่างเมืองที่ i ไปยังเมืองที่ j

ตัวแปรในการตัดสินใจ (Decision Variables)

$X_{ij} =$ 1, ถ้ามีการเดินทางจากเมือง i ไปเมือง j
0, กรณีอื่นๆ

ฟังก์ชันวัตถุประสงค์ (Objective Function)

$$Min = \sum_{i=1}^{N}\sum_{j=1}^{N} C_{ij}X_{ij} \tag{2.12}$$

สมการข้อจำกัด (Constraints)

$$\sum_{i=1}^{N} X_{ij} = 1 \qquad \forall_{j=1,2,..,N} \tag{2.13}$$

$$\sum_{j=1}^{N} X_{ij} = 1 \qquad \forall_{i=(1,2,..,N)} \tag{2.14}$$

$$\sum_{i\in S}\sum_{j\in S} X_{ij} \leq |S| - 1 \qquad \forall_{S\subset V} \tag{2.15}$$

$$X_{ij}\epsilon\{0,1\} \qquad \forall_{i=(1,2,..,N),j=(1,2,..,N)} \tag{2.16}$$

ปัญหา VRP เป็นปัญหาประเภท NP-hard โดยมีจุดเริ่มต้นจากบทความของ Dantzig and Ramser (1959) และมีนักวิจัยอีกเป็นจำนวนหนึ่งที่พัฒนาต่อๆ กันมาเรื่อยๆ รวมระยะเวลาตั้งแต่อดีตจนถึงปัจจุบันเกือบ 50 ปีมาแล้ว ยกตัวอย่างนักวิจัยที่มีบทบาทสำคัญในการพัฒนางานทางด้านปัญหา VRP เช่น Golden et al. (1977), Christofides et al. (1979), Laporte et al. (2000), Toth and Vigo (2002) และ Kytojoki et al. (2007) จากการศึกษางานวิจัยที่ผ่านมาพบว่า ลักษณะของปัญหา VRP แสดงด้วยแบบจำลองทางคณิตศาสตร์มาตรฐานได้ดังนี้

ดัชนี (Indices)

i ลำดับของลูกค้าที่ i โดยที่ $i = 1,2,\dots,N$

j ลำดับของลูกค้าที่ j โดยที่ $j = 1,2,\dots,N$

k ยานพาหนะที่ใช้ในการขนส่งสินค้าที่ k โดยที่ $k = 1,2,\dots,K$

ตัวแปรที่ทราบค่า (Parameter)

D_i ความต้องการสินค้าของลูกค้าที่ i โดยกำหนดให้ $D_1 = 0$

C_{ij} ระยะทางระหว่างลูกค้าที่ i ไปยังลูกค้าที่ j

Q_k ความสามารถในการขนส่งสินค้าของพาหนะขนส่งที่ k

ตัวแปรในการตัดสินใจ (Decision Variables)

$X_{ij} =$ 1, ถ้ามีการเดินทางจากลูกค้า i ไปลูกค้า j

0, กรณีอื่นๆ

$U_i =$ ตัวแปรสนับสนุน (Auxiliary variable) ที่มีค่ามากกว่าหรือเท่ากับ 0

ฟังก์ชันวัตถุประสงค์ (Objective Function)

$$Min = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} C_{ij}X_{ij}^{k} \qquad (2.17)$$

สมการข้อจำกัด (Constraints)

$$\sum_{i=1}^{N}\sum_{k=1}^{K} X_{ij}^{k} = 1 \qquad \forall_{j=(2,..,N)} \qquad (2.18)$$

$$\sum_{j=1}^{N}\sum_{k=1}^{K} X_{ij}^{k} = 1 \qquad \forall_{i=(2,...,N)} \qquad (2.19)$$

$$\sum_{i=1}^{N} X_{ip}^{k} - \sum_{j=1}^{N} X_{pj}^{k} = 0 \qquad \forall_{k=(1,2,..,K),p=(1,2,..,N)} \qquad (2.20)$$

$$\sum_{i=1}^{N} D_i \left(\sum_{j=1}^{N} X_{ij}^{k} \right) \leq Q_k \qquad \forall_{k=(1,2,...,K)} \qquad (2.21)$$

$$\sum_{j=2}^{N} X_{1j}^{k} \leq 1 \qquad \forall_{k=(1,2,...,K)} \qquad (2.22)$$

$$\sum_{i=2}^{N} X_{i1}^{k} \leq 1 \qquad \forall_{k=(1,2,...,K)} \qquad (2.23)$$

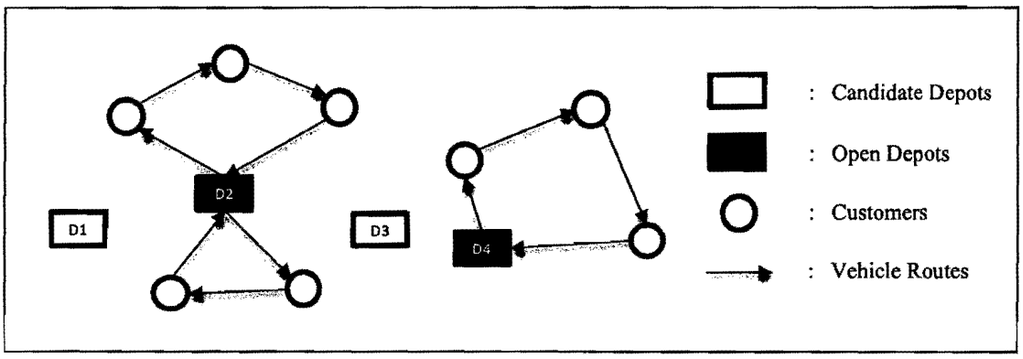
$$U_i - U_j + N\sum_{k=1}^{K} X_{ij}^{k} \leq N - 1 \qquad \forall_{i,j=(2,...,N)\ and\ i\neq j} \qquad (2.24)$$

$$X_{ij}^{k} \epsilon \{0,1\} \qquad \forall_{i,j=(1,2,...,N),k=(1,2,...,K)} \qquad (2.25)$$

สมการเป้าหมายแสดงวัตถุประสงค์หลัก (2.17) ต้องการหาเส้นทางที่ให้ค่าระยะทางรวมต่ำสุด สมการเงื่อนไขที่ (2.18) แสดงว่าที่จุดปลายทางใดๆ จะมีจุดเริ่มต้นเพียงจุดเดียวเท่านั้น สมการ (2.19) แสดงถึงการเดินทางจากจุดเริ่มต้นใดๆ จะมีจุดหมายปลายทางเพียงแห่งเดียวเท่านั้น สมการเงื่อนไขที่ (2.20) แสดงความต่อเนื่องของตำแหน่งที่เชื่อมกันอยู่ในแต่ละเส้นทาง เมื่อยานพาหนะเดินทางเข้ามายังจุดใดๆ แล้วต้องมีการเดินทางออกจากจุดนั้น สมการเงื่อนไขที่ (2.21) บังคับให้ความต้องการรวมของลูกค้าในแต่ละเส้นทางจะต้องไม่เกินความจุของยานพาหนะ สมการเงื่อนไขที่ (2.22) และ (2.23) ยืนยันว่ามียานพาหนะใช้ได้เท่าที่กำหนด สมการเงื่อนไขที่ (2.24) เป็นการจำกัดการเกิดเส้นทางย่อย สมการ และสมการเงื่อนไขสุดท้าย (2.25) บังคับให้ตัวแปร X_{ij} เป็นค่าไบนารี่เท่านั้น

2.4 ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง (Location Routing Problem: LRP)

ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งเป็นปัญหาที่มีจุดประสงค์เพื่อเลือกสถานที่ตั้งแต่ในขณะเดียวกันก็มีจุดประสงค์เพื่อแก้ปัญหาการจัดเส้นทางการขนส่งไปด้วย รูปแบบของปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งจะมีความคล้ายคลึงกับปัญหาการเลือกสถานที่ตั้ง และในขณะเดียวกันก็มีความคล้ายคลึงกับปัญหาการจัดเส้นทางการขนส่งด้วย โดยหากเราให้ลูกค้าทุกรายเชื่อมต่อกับโรงงานหรือศูนย์กระจายสินค้า ปัญหานั้นก็จะเป็นปัญหาการเลือกสถานที่ตั้งแบบมาตรฐาน แต่หากกำหนดที่ตั้งของโรงงานหรือศูนย์กระจายสินค้าตั้งแต่เริ่มต้นปัญหานั้นก็จะมีรูปแบบเป็นปัญหาการจัดเส้นทางการขนส่งดังภาพที่ 2.3



ภาพที่ 2.3 รูปแบบของปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง

จากรูปแบบปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งที่มีความซับซ้อน ซึ่งปัญหานี้ถูกจัดอยู่ในกลุ่มปัญหาประเภท Nondeterministic Polynomial-Time Hard (NP-hard) เพราะว่าเป็นการรวมปัญหา NP-Hard ทั้งสองปัญหาคือ (1) ปัญหาการเลือกสถานที่ตั้ง และ (2) ปัญหาการจัดเส้นทางการขนส่งของยานพาหนะรวมเข้าด้วยกัน Wu et al. (2002) ได้นำเสนอการปรับปรุงแบบจำลองทางคณิตศาสตร์ของ Perl and Daskin (1985) รูปแบบทางคณิตศาสตร์ของปัญหา LRP เป็นดังนี้

ดัชนี (Indices)

i	จุดศูนย์กลางการกระจายสินค้าทั้งหมดที่เป็นไปได้
j	ลูกค้าทั้งหมด
k	ยานพาหนะทั้งหมดที่ใช้ในการขนส่งสินค้า

ตัวแปรที่ทราบค่า (Parameter)

N	จำนวนของลูกค้าทั้งหมด
C_{ij}	ค่าใช้จ่ายที่ใช้ในการเดินทางจากจุด i ไปยังจุด j
G_i	ต้นทุนคงที่ของการสร้างศูนย์กระจายสินค้า i
F_k	ต้นทุนคงที่ของการใช้พาหนะ k
V_i	ความสามารถในการรองรับสินค้าของศูนย์กระจายสินค้า i
d_j	ความต้องการของลูกค้า j
Q_k	ความสามารถในการขนส่งสินค้าของพาหนะขนส่ง (หรือเส้นทาง) k

ตัวแปรในการตัดสินใจ (Decision Variables)

$X_{ij} =$	1, ถ้ามีการเดินทางจากจุด i ไปยังจุด j บนเส้นทาง k
	0, กรณีอื่นๆ
$Y_i =$	1, ถ้าศูนย์กระจายสินค้า i ถูกสร้างขึ้น
	0, กรณีอื่นๆ
$Z_{ij} =$	1, ถ้าลูกค้า j ถูกกำหนดให้ได้รับการบริการจากศูนย์กระจายสินค้า i
	0, กรณีอื่นๆ
$U_{lk} =$	ตัวแปรสนับสนุน (Auxiliary variable)

ฟังก์ชันวัตถุประสงค์ (Objective Function)

$$Min = \sum_{i\epsilon I} G_i Y_i + \sum_{i\epsilon I\cup J}\sum_{j\epsilon I\cup J}\sum_{k\epsilon K} C_{ij}X_{ijk} + \sum_{k\epsilon K} F_k \sum_{i\epsilon I}\sum_{j\epsilon J} X_{ijk} \tag{2.26}$$

สมการข้อจำกัด (Constraints)

$$\sum_{k\epsilon K}\sum_{i\epsilon I\cup J} X_{ijk} = 1 \qquad j\epsilon J \tag{2.27}$$

$$\sum_{j\epsilon J} d_j \sum_{i\epsilon I\cup J} X_{ijk} \le Q_k \qquad k\epsilon K \tag{2.28}$$

$$U_{lk} - U_{jk} + NX_{ijk} \le N = 1 \qquad i,j\epsilon J, k\epsilon K \tag{2.29}$$

$$\sum_{j\epsilon I\cup J} X_{ijk} - \sum_{j\epsilon I\cup J} X_{jik} = 0 \qquad k\epsilon K \qquad (2.30)$$

$$\sum_{i\epsilon I}\sum_{j\epsilon J} X_{ijk} \leq 1 \qquad k\epsilon K \qquad (2.31)$$

$$\sum_{j\epsilon J} d_j z_{ij} - V_i y_i \leq 0 \qquad i\epsilon I \qquad (2.32)$$

$$-z_{ij} + \sum_{u\epsilon I\cup J} \left(X_{iuk} + X_{ujk}\right) \leq 1 \qquad i\epsilon I, j\epsilon J, k\epsilon K \qquad (2.33)$$

$$X_{ijk} = 0,1 \qquad i\epsilon I, j\epsilon J, k\epsilon K \qquad (2.34)$$

$$Y_i = 0,1 \qquad i\epsilon I \qquad (2.35)$$

$$z_{ij} = 0,1 \qquad i\epsilon I, j\epsilon J \qquad (2.36)$$

$$U_{lk} \geq 0 \qquad l\epsilon J, k\epsilon K \qquad (2.37)$$

ฟังก์ชันวัตถุประสงค์ (2.26) คือผลรวมต่ำที่สุดจากผลบวกของต้นทุนคงที่ในการสร้างศูนย์กระจายสินค้า, ต้นทุนการขนส่งและต้นทุนคงที่ในการใช้ยานพาหนะขนส่ง สมการเงื่อนไขที่ (2.27) เป็นการกำหนดให้ลูกค้าแต่ละคนสามารถรับการบริการได้จากพาหนะขนส่งเพียงคนละหนึ่งคันหรือหนึ่งเส้นทางเท่านั้น สมการที่ (2.28) เป็นการกำหนดเงื่อนไขว่าปริมาณของสินค้าที่พาหนะคันนั้นๆ รับผิดชอบจะต้องไม่เกินความสามารถในการบรรทุกสินค้าสมการที่ (2.29) เป็นสมการเงื่อนไขป้องกันการเกิดซับทัวร์ สมการที่ (2.30) เป็นเงื่อนไขที่บังคับให้พาหนะขนส่งเข้าให้บริการที่จุดใดก็ต้องออกจากจุดนั้นเสมอ สมการที่ (2.31) พาหนะขนส่งแต่ละคันสามารถทำงานให้กับศูนย์กลางการจัดส่งพัสดุเพียงแห่งเดียวเท่านั้น สมการที่ (2.32) เป็นเงื่อนไขด้านความสามารถในการรองรับความต้องการของลูกค้าของศูนย์กระจายสินค้า สมการที่ (2.33) เป็นการจัดสรรลูกค้าให้เข้าใช้บริการในศูนย์กระจายสินค้าได้เพียงแห่งเดียว หากมีเส้นทางที่เชื่อมต่อระหว่างลูกค้าและศูนย์กระจายสินค้า ส่วนสมการที่ (2.34), (2.35) และ (2.36) เป็นการกำหนดตัวแปรตัดสินใจแบบไบนารี่ สมการเงื่อนไขสุดท้าย (2.37) เป็นตัวแปรสนับสนุนที่ต้องมีค่าเป็นบวกเท่านั้น ส่วนที่เพิ่มเติมจากรูปแบบทางคณิตศาสตร์ของ Perl and Daskin (1985) ได้แก่

(1) เพิ่มเติมสมการเงื่อนไขที่สามารถแก้ปัญหาการเกิดซับทัวร์ขึ้นในคำตอบของปัญหาสมการที่ (2.29)

(2) ไม่มีการนำค่าใช้จ่ายผันแปรต่อหน่วยของศูนย์กลางการกระจายสินค้ามาพิจารณาในสมการเป้าหมาย ในขณะที่มีการเพิ่มค่าใช้จ่ายในการจัดส่งของพาหนะขนส่งลงไปแทน

(3) เพื่อเพิ่มความยืดหยุ่นในการใช้งานจึงอนุญาตให้ใช้งานพาหนะขนส่งที่มีความสามารถในการบรรทุกแตกต่างกันได้

2.5 วิธีการแก้ปัญหา (Solution Approach)

ในการหาคำตอบของปัญหา LRP นั้น ในอดีตจะใช้คนเป็นผู้กำหนดสถานที่ตั้งและจัดเส้นทางโดยอาศัยประสบการณ์หรือใช้กลยุทธ์ที่ไม่ซับซ้อนในการเลือก และไม่จำเป็นต้องมีการคำนวณที่ยุ่งยากนัก เนื่องจากปัญหามีขนาดไม่ใหญ่รวมถึงมีความซับซ้อนไม่มากนัก ซึ่งผลลัพธ์ที่ได้ก็จะอยู่ในระดับที่ยอมรับได้ แต่ต่อมาธุรกิจมีการเติบโตมากขึ้น รวมถึงมีการรวมกิจการทำให้ระบบกระจายสินค้ามีความซับซ้อนมากขึ้น ส่งผลให้ตัวปัญหาในส่วนของของการจัดเส้นทางยานพาหนะมีขนาดใหญ่และซับซ้อนขึ้นด้วย ซึ่งการจัดเส้นทางที่เหมาะสมจะสามารถช่วยลดความซับซ้อนในการดำเนินงานรวมถึงสามารถลดต้นทุนลงได้เป็นจำนวนมาก รูปแบบของปัญหาที่พบในระยะหลังมักจะมีความซับซ้อน รวมถึงมีขนาดใหญ่มากเกินกว่าที่จะใช้ความสามารถของคนในการจัดเส้นทางเพื่อให้ได้คำตอบที่เหมาะสม การพัฒนานำเอาคอมพิวเตอร์มาประยุกต์ใช้เป็นส่วนหนึ่งที่ทำให้เราสามารถแก้ปัญหาที่มีความซับซ้อนได้ในเวลาที่สั้นลง สามารถนำเอาวิธีการหาคำตอบที่เป็นระบบและความซับซ้อน รวมถึงวิธีการที่มีประสิทธิภาพมาประยุกต์ใช้กับปัญหาจริงได้ การหาคำตอบสำหรับปัญหา LRP สามารถแยกออกเป็น 2 แนวทางคือ (1) การหาคำตอบแบบ Exact Optimization และ (2) การหาคำตอบแบบ Heuristic Optimization ซึ่งทั้ง 2 แนวทางนั้นจะมีจุดเด่นและจุดด้อยที่แตกต่างกันคือ การหาคำตอบแบบ Exact Optimization จะได้คำตอบในรูปของผลลัพธ์ที่ดีที่สุด ซึ่งจำเป็นต้องสร้างตัวแปรทางคณิตศาสตร์สำหรับการแก้ปัญหา โดยวิธีการดังกล่าวจะใช้ระยะเวลาในการคำนวณที่สูงมากเมื่อปัญหามีขนาดใหญ่ ส่วนวิธีการหาคำตอบแบบ Heuristic Optimization คำตอบที่ได้จะอยู่ในรูปของผลลัพธ์ที่ใกล้เคียงกับผลลัพธ์ที่ดีที่สุด โดยจะใช้เวลาในการคำนวณน้อยกว่ามากเมื่อเทียบกับการหาคำตอบแบบ Exact Optimization

2.5.1 วิธีการหาคำตอบที่ดีที่สุด (Exact Optimization)

การหาคำตอบที่เหมาะสมที่สุดเชิงการจัด (Combinatorial Optimization) คือการหาคำตอบที่ดีที่สุดจากเซตคำตอบทั้งหมดที่เป็นไปได้ของปัญหา Dorigo and Stutzle (2004) โดยที่คำตอบที่ดีที่สุดของปัญหา อาจเป็นรูปแบบของคำตอบที่ต้องการค่าที่มากที่สุด (Maximization Problem) ตัวอย่างเช่น ต้องการหากำไรสูงสุดหรืออาจเป็นรูปแบบของคำตอบที่ต้องการค่าที่น้อยที่สุด (Minimization problem) ตัวอย่างเช่น ต้องการหาต้นทุนที่ต่ำที่สุด ปัญหาประเภท Combinatorial Optimization นี้สามารถพบได้ทั่วไปในปัญหาโลจิสติกส์ วิธีที่ใช้ในการแก้ปัญหานั้นมีด้วยกันหลากหลายวิธี ซึ่งวิธีที่ได้รับความนิยมอย่างแพร่หลาย ได้แก่

การโปรแกรมเชิงเส้นตรง (Linear Programming: LP) เป็นการประยุกต์ใช้วิธีการแก้ไขปัญหาการจัดสรรทรัพยากรที่มีลักษณะความสัมพันธ์ของตัวแปรต่างๆ ที่เกี่ยวข้องเป็นแบบเชิงเส้นตรงทั้งหมด โดยมีจุดมุ่งหมายเพื่อแก้ปัญหาและตัดสินใจให้ได้ค่าของฟังก์ชั่นเป้าหมายที่ดีที่สุด ในปัจจุบันมีซอร์ฟแวร์หนึ่งที่ใช้ในการแก้ปัญหาโปรแกรมเชิงเส้นตรง มีชื่อว่า LINGO ถูกคิดค้นโดยบริษัท LINDO System ประเทศสหรัฐอเมริกา หลักการทำงานของโปรแกรมนี้สามารถเขียนสมการคณิตศาสตร์ของปัญหาได้โดยตรงหรือเขียนในรูปแบบสมการคณิตศาสตร์ทั่วไปได้ หากเป็นสมการรูปแบบทั่วไปจะเริ่มต้นด้วยการกำหนดตัวแปรและพารามิเตอร์ต่างๆ ที่เกี่ยวข้องกับปัญหา จากนั้นเขียนสมการเป้าหมายและสมการเงื่อนไขในรูปแบบของโปรแกรม LINGO

วิธีการแตกกิ่งและจำกัดขอบเขต (The Branch-and-Bound Algorithm) เป็นวิธีการที่ใช้หลักการของการตัดทอนการแจงนับ วิธีการนี้ประกอบไปด้วย 2 ขั้นตอนคือ การแตกกิ่ง (Branching) คือกระบวนการแบ่งส่วนของปัญหาที่มีขนาดใหญ่ให้เป็นปัญหาย่อย 2 ปัญหาหรือมากกว่า ส่วนการจำกัดเขต (Bounding) คือกระบวนการหาค่าขอบเขตล่าง (Lower Bound) ของคำตอบที่ดีที่สุดสำหรับปัญหาย่อยที่กำหนดให้ ในระหว่างขั้นตอนของการแตกกิ่ง จะมีการแทนที่ปัญหาเริ่มต้นด้วยเซตของปัญหาใหม่ที่มีลักษณะดังนี้ อย่างแรกปัญหาดั้งเดิมจะถูกแตกออกเป็นหลายปัญหาย่อยที่ไม่เกิดขึ้นร่วมกัน และมีการแจงกรณีทั้งหมดที่เป็นไปได้ อย่างที่สองปัญหาย่อยเหล่านี้จะเป็นกรณีหนึ่งของปัญหาเดิมที่หาคำตอบได้แล้วบางส่วน สุดท้ายปัญหาย่อยเหล่านี้จะมีขนาดเล็กกว่าปัญหาเดิม

การโปรแกรมแบบไดนามิก (Dynamic Programming) เป็นวิธีการหาค่าที่ดีที่สุด โดยแบ่งกลุ่มของจุดเชื่อมที่พิจารณาจัดเส้นทางออกเป็น 2 กลุ่ม คือ กลุ่มที่จัดเส้นทางไปแล้วและกลุ่มที่ยังไม่ได้จัดเส้นทาง จากนั้นพิจารณาเส้นทางที่ดีที่สุดทุกครั้งที่เลือกจุดเชื่อมหนึ่งจุดใดๆ เข้ากับกลุ่มของจุดเชื่อมที่จัดเส้นทางไปแล้ว

2.5.2 วิธีการหาคำตอบแบบฮิวริสติก (Heuristic Optimization)

ฮิวริสติกเป็นวิธีการค้นหาผลเฉลยของปัญหาภายใต้ระยะเวลาที่เหมาะสม และไม่ขัดแย้งกับเงื่อนไขทุกเงื่อนไขที่กำหนด โดยทั่วไปวิธีฮิวริสติกถูกสร้างขึ้นมาเพื่อใช้ในการหาคำตอบของแต่ละปัญหาเท่านั้น ดังนั้นวิธีฮิวริสติกที่สามารถหาคำตอบสำหรับปัญหาหนึ่งจึงไม่สามารถนำไปใช้หาคำตอบของอีกปัญหาหนึ่งได้ นอกจากนี้ในบางครั้งปัญหาการตัดสินใจที่กำลังพิจารณาอยู่นั้นอาจไม่สามารถเขียนให้อยู่ในรูปแบบจำลองทางคณิตศาสตร์ได้ เพราะว่าปัญหานั้นมีตัวแปรและเงื่อนไขในการตัดสินใจที่มีความซับซ้อนมาก ตัวอย่างของวิธีฮิวริสติกที่นิยมใช้ในงานวิจัยโลจิสติกส์ เช่น วิธี Saving, วิธี Matching Based, วิธี Nearest Insertion, วิธี Nearest Neighbor และวิธี

Local Search เป็นต้น ดังนั้นวิธีฮิวริสติกจึงถูกพัฒนาให้มีความยืดหยุ่น รวดเร็วและมีประสิทธิภาพมากยิ่งขึ้น เรียกว่า เมต้าฮิวริสติก (Meta-Heuristic)

Blum C. and Roli A. (2003) ได้สรุปหลักการเบื้องต้นของเมต้าฮิวริสติกไว้ว่า (1) เมต้าฮิวริสติกมีระเบียบวิธีในการค้นหาคำตอบที่ดีภายในเซตของคำตอบที่เป็นไปได้ (2) เมต้าฮิวริสติกมีจุดประสงค์เพื่อหาคำตอบที่ดีที่สุดหรือคำตอบที่ใกล้เคียงที่ดีที่สุด ภายในระยะเวลาที่เหมาะสม (3) วิธีเมต้าฮิวริสติกอาจมีทั้งแบบซับซ้อนและไม่ซับซ้อน เช่น วิธีโลคอล เซิร์ช (Local Search), วิธีระบบมด (Ant System), วิธีการทางพันธุกรรม (Genetic Algorithm), วิธีค้นหาต้องห้าม (Tabu Search) และวิธีเลียนแบบการอบอ่อน (Simulated Annealing) เป็นต้น (4) เมต้าฮิวริสติกอาจเกิดจากการรวมหลากหลายวิธีเทคนิคเพื่อค้นหาคำตอบที่ดีที่สุดภายในพื้นที่คำตอบที่เป็นไปได้ (5) เมต้าฮิวริสติกมีระเบียบขั้นตอนที่แน่นอนแต่สามารถปรับเปลี่ยนในรายละเอียดเมื่อนำไปใช้ในแต่ละปัญหา (6) เมต้าฮิวริสติกบางประเภทมีการใช้ความจำชั่วคราวมากขึ้น ในการจดจำคำตอบเดิมเพื่อให้การค้นหาคำตอบไม่ซ้ำที่เดิม

นอกจากนี้ Blum C. and Roli A. (2003) ยังได้เสนอการแบ่งเมต้าฮิวริสติกไว้ 6 ประเภทดังนี้ (1) เมต้าฮิวริสติกที่เกิดจากแรงบันดาลใจจากธรรมชาติ ได้แก่ วิธีระบบมด (Ant System), วิธีการทางพันธุกรรม (Genetic Algorithm), วิธีการเลียนแบบการอบอ่อน (Simulated Annealing) และวิธีการเกาะกลุ่มประชากรแบบ PSO (Particle Swarm Optimization) เป็นต้น (2) เมต้าฮิวริสติกที่ไม่ได้เกิดจากแรงบันดาลใจจากธรรมชาติ ได้แก่ วิธีการค้นหาต้องห้าม (Tabu Search) เป็นต้น (3) เมต้าฮิวริสติกแบบใช้ประชากรคือในหนึ่งรอบของการค้นหาคำตอบจะได้คำตอบมากกว่าหนึ่งคำตอบให้เลือก เช่น วิธีระบบมด (Ant System), วิธีการทางพันธุกรรม (Genetic Algorithm), วิธีการลอกแบบ (Memetic Algorithm) และวิธีการเกาะกลุ่มประชากรแบบ PSO (Particle Swarm Optimization) เป็นต้น (4) เมต้าฮิวริสติกแบบไม่ใช้ประชากร คือในหนึ่งรอบของการค้นหาคำตอบจะได้คำตอบออกมาเพียงหนึ่งคำตอบเท่านั้น เช่น วิธีการเลียนแบบการอบอ่อน (Simulated Annealing), วิธีการค้นหาต้องห้าม (Tabu Search) และวิธีการค้นหาในพื้นที่คำตอบที่เป็นไปได้แบบวนซ้ำ (Iterated Local Search) เป็นต้น (5) เมต้าฮิวริสติกแบบสมการเป้าหมายคงที่ คือในหนึ่งรอบของการคำนวณอาจมการเปลี่ยนสมการเป้าหมาย เพื่อให้เกิดคำตอบใหม่ๆ เกิดขึ้น เช่น วิธี Guided Local Search เป็นต้น (6) เมต้าฮิวริสติกแบบไม่มีการเปลี่ยนสมการเป้าหมาย เช่น วิธีระบบมด (Ant System), วิธีการทางพันธุกรรม (Genetic Algorithm), วิธีการเลียนแบบการอบอ่อน (Simulated Annealing) และวิธีการลอกแบบ (Memetic Algorithm) เป็นต้น

2.6 การจำลองพฤติกรรมการหาอาหารของมด

การจำลองพฤติกรรมการหาอาหารของมดเป็นขั้นตอนวิธีที่ได้แนวคิดมาจากพฤติกรรมการหาอาหารของมด ซึ่ง Beckers et al. (1992) ได้อธิบายไว้ว่ามดจะค้นหาเส้นทางสั้นที่สุดระหว่างแหล่งอาหารกับรังมด ในการเดินทางไปกลับระหว่างแหล่งอาหารกับรัง มดจะทิ้งหลักฐานที่เรียกว่าฟีโรโมน (Pheromone) ไว้บนพื้น เมื่อมีการตัดสินใจเลือกทางเดินมดก็จะเลือกเส้นทางที่มีฟีโรโมนหนาแน่นมากกว่า ขั้นตอนวิธี ACO ถูกนำเสนอครั้งแรกโดย Dorigo et al. (1996) โดยเขาทำการศึกษาการจำลองพฤติกรรมการหาอาหารของมดขึ้นมาเรียกว่า วิธีระบบมด (Ant System: AS) ซึ่งการทำงานโดยทั่วไปคือ เริ่มต้นการทำงานจะมีการกำหนดค่าระดับฟีโรโมนให้เท่ากันทุกเส้นทาง โดยมีค่าเริ่มต้นเป็นค่าต่ำๆ ค่าหนึ่ง ($ph > 0$) มดทุกตัวจะสร้างผลเฉลยโดยการเพิ่มส่วนประกอบเข้าไปทีละส่วนในรูปแบบของการเดินทางผ่านไปตามเส้นเชื่อมต่อระหว่างโหนด ส่วนประกอบถัดไปที่จะถูกเพิ่มเข้าไปเป็นส่วนหนึ่งของผลเฉลยใน AS จะใช้กฎเปลี่ยนสถานะ ซึ่งจะมีพารามิเตอร์สำคัญที่ต้องใช้ร่วมในการพิจารณาคือ ค่า Pheromone และ Heuristic Information เช่น ในกรณีการแก้ปัญหา TSP ค่าฟีโรโมนก็จะเป็นระดับฟีโรโมนระหว่างเมือง 2 เมือง ซึ่งได้มาจากการปรับปรุงค่าฟีโรโมนจากการทำงานในรอบที่ผ่านมา ส่วน Heuristic Information ก็คือระยะทางระหว่างเมือง 2 เมืองและเมื่อมดทุกตัวได้สร้างผลเฉลยเสร็จสมบูรณ์แล้วก็จะมีการปรับปรุงค่าฟีโรโมน โดยมีวัตถุประสงค์เพื่อเพิ่มค่าฟีโรโมนให้กับส่วนประกอบของผลเฉลยที่พบว่ามีคุณภาพดี จากนั้นก็จะมีการวนรอบการทำงานซ้ำจนกระทั่งเงื่อนไขที่กำหนดให้หยุด ในปัจจุบันวิธีการ ACO ได้รับความนิยมกันอย่างมากในหมู่นักวิจัยและได้คิดค้นพัฒนาวิธีการมาอย่างต่อเนื่อง ดังนั้นจึงสรุปวิธีการคิดเลียนแบบพฤติกรรมการหาอาหารของมดออกได้ 5 แบบดังนี้

2.6.1 วิธีระบบมด (Ant System: AS)

ได้ถูกนำเสนอเป็นครั้งแรกในปี ค.ศ. 1991 โดยได้ถูกคิดค้นและพัฒนาขึ้นโดย Marco Dorigo และคณะ ซึ่งที่จริงแล้วระบบมดเดิมมีอยู่ 3 แบบด้วยกันคือ Ant-density, Ant-quantity และ Ant-cycle ตามลำดับ วิธีระบบ Ant-density และวิธีระบบ Ant-quantity นั้น จะมีการอัพเดทสารฟีโรโมนทันทีขณะที่เดินทางจากโหนดหนึ่งไปยังเมืองใดๆ ขณะที่วิธีระบบ Ant-cycle นั้นจะอัพเดทสารฟีโรโมนหลังจากที่มดเดินทางครบทุกเมืองแล้ว โดยที่ปริมาณของสารฟีโรโมนที่จะอัพเดทนั้น ขึ้นอยู่กับอัตราส่วนระหว่างค่าคงที่ต่อระยะทาง (Distance) หรือคุณภาพของผลเฉลยที่ได้ (Quality solution) ท้ายที่สุดแล้ววิธีระบบ Ant-density และวิธีระบบ Ant-quantity ก็ไม่ได้รับการปรับปรุงและพัฒนาต่อไปอีก เนื่องจากมีประสิทธิภาพในการหาผลเฉลยหรือเส้นทางที่แย่กว่าเมื่อเทียบกับวิธีระบบ Ant-cycle ดังนั้นในปัจจุบัน เมื่อกล่าวถึงระบบมดก็คือวิธีระบบ Ant-cycle นั่นเอง Dorigo and Stutzle (2004)

2.6.2 วิธีระบบมดที่ดีที่สุด (Elitist Ant System: ES)

วิธีระบบมดที่ดีที่สุดเกิดจาการพัฒนา AS ให้มีประสิทธิภาพดีขึ้น เรียกการพัฒนานี้ว่า Elitist Strategy นำเสนอเป็นครั้งแรกในปี 1992 โดย Dorigo (1992) การทำงานเริ่มต้นจะเหมือนกับ AS แต่จะเพิ่มเติมในส่วนของการเก็บค่าที่ดีที่สุดของแต่ละรอบการคำนวณ (Best so far tour) เพื่อการเพิ่มร่องรอยฟีโรโมน (Update pheromone trail) โดยจะเก็บค่าที่ดีที่สุดของรอบการคำนวณที่หนึ่งแล้วอัพเดทฟีโรโมน ซึ่งเส้นทางที่เป็นค่าที่ดีที่สุด จะมีปริมาณฟีโรโมนมากกว่าเส้นทางที่มดผ่านปกติทั่วไป

2.6.3 วิธีระบบมดแบบกำหนดขอบเขตสูงสุด-ต่ำสุด (Max-Min Ant System: MMAS)

วิธีระบบมดแบบกำหนดขอบเขตสูงสุด-ต่ำสุด ถูกนำเสนอเป็นครั้งแรกในปี 1997 โดย Stutzle and Hoos (1997) ผู้ที่นำเสนอได้พัฒนาวิธี MMAS มาจากวิธี AS โดยได้พัฒนา 4 อย่างด้วยกันดังนี้ (1) กำหนดช่วงของการใช้ฟีโรโมนเริ่มต้น (Initial pheromone) ที่นำมาใช้โดยกำหนดช่วง $[\tau_{Max}, \tau_{Min}]$ เป็นสัดส่วนของความน่าจะเป็นของ p_{ij}สำหรับการเลือกเมือง j ขณะที่มดอยู่ที่เมือง i ดังนั้น $[p_{Max}, p_{Min}] \Rightarrow 0 < p_{min} = p_{ij} = p_{max} = 1$ (2) กำหนดการใช้ค่าฟีโรโมนเริ่มต้น $\tau_0 = \tau_{max}$ (3) ใช้กฎการเปลี่ยนสถานะ (State Transition Rule) (4) กฎการปรับเปลี่ยนปริมาณฟีโรโมน (Pheromone Trial Update Rule)

2.6.4 วิธีระบบมดแบบเรียงลำดับ (Rank-Base Ant System: RBS)

วิธีระบบมดแบบเรียงลำดับได้ถูกนำเสนอเป็นครั้งแรกโดย Bullnheimer et al. (1999) หลักการเบื้องต้นคือการอนุญาตให้มดสามารถปล่อยฟีโรโมนได้มากกว่า 1 ตัวในหนึ่งรอบการวนซ้ำ การเรียงลำดับมดในแต่ละรอบให้เรียงลำดับของมดที่มีระยะทางรวมสั้นที่สุด

2.6.5 วิธีระบบอาณานิคมมด (Ant Colony System: ACS)

วิธีระบบอาณานิคมมดถูกสร้างขึ้นมาเพื่อปรับปรุงประสิทธิภาพของ AS โดยการปรับปรุงครั้งนี้ต่างจากทุกครั้งที่ผ่านมากล่าวคือ การปรับปรุงครั้งนี้ไม่ได้อยู่บนพื้นฐานของ AS อีกต่อไป ซึ่งสร้างกลไลการทำงานใหม่เพื่อเพิ่มประสิทธิภาพการทำงาน ACS ถูกนำเสนอครั้งแรกโดย Dorigo and Gambardell (1996) มีความแตกต่างจาก AS สามหลักการใหญ่ๆ ดังนี้ (1) ใช้กฎการเปลี่ยนสถานะ (State Transition Rule) (2) กฎการปรับเปลี่ยนปริมาณฟีโรโมน (Pheromone Trial Update Rule) (3) เมื่อมดทุกตัวสร้างเส้นทางเดินจนครบหมดแล้ว จะทำการเพิ่มปริมาณฟีโรโมนให้แก่เส้นทางที่ดีที่สุดเท่านั้น เพื่อเป็นการเสริมเส้นทางนั้นให้เด่นชัดขึ้น การประยุกต์ใช้การจำลองพฤติกรรมการหาอาหารของมดเพื่อแก้ปัญหาวิศวกรรมสามารถสรุปได้ดังในตารางที่ 2.1

ตารางที่ 2.1 การประยุกต์ใช้การจำลองพฤติกรรมการหาอาหารของมดเพื่อแก้ปัญหาวิศวกรรม

ปัญหา	นักวิจัย	วิธีการ
Traveling Salesman Problem	Dorigo et al. (1991)	AS
	Colorni et al. (1994)	AS
	Dorigo and Gamardella (1997)	AS
	Dorigo and Gamardella (1997)	ACS
	Stutzle and Hoos (2000)	MMAS
	Randall and Lewis (2002)	ACS
	Craus and Rudeanu (2004)	ACO
	Craus and Rudeanu (2004)	ACO
	Delisle et al. (2005)	ACS
	Delisle et al. (2005)	ACS
	Lv et al. (2006)	MMAS, ACS
	Bai et al. (2009)	MMAS
	Tsutsui and Fujimoto (2010)	cAS
	Li et al. (2010)	ACO
	Fu et al. (2010)	MMAS
Vehicle Routing Problem	Bullnheimer et al. (1997)	AS
	Rizzoli et al. (2002)	ACS
	Bianchi et al. (2004)	ACS
	Wade and Salhi (2004)	AS
	Doerner et al. (2004)	AS
	Doerner et al. (2005)	D-Ant
	Doerner et al. (2006)	D-Ant
Assignment Problem	Gambardella et al. (1999)	ACO
	Stutzle and Hoos (2000)	MMAS
Quadratic Assignment Problem	Talbi et al. (2001)	ANTabu
	Tsutsui (2007)	cAS
	Tsutsui (2008)	cAS

2.7 งานวิจัยที่เกี่ยวข้อง (Literature Review)

นอกจากงานวิจัยที่เกี่ยวข้องกับ LRP แล้วการศึกษางานวิจัยที่เกี่ยวข้องกับปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งในลักษณะอื่นที่เกี่ยวกับ LRP ก็สามารถให้แนวคิดในการแก้ปัญหาได้เป็นอย่างดีเช่นกัน

Balakrishnan at el. (1987) กล่าวว่าปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งการที่จะค้นหาคำตอบจากปัญหานี้ได้อย่างมีประสิทธิภาพควรแก้ปัญหาไปพร้อมๆ กัน โดยไม่ต้องแยกปัญหาออกจากกัน เนื่องจากว่าปัญหานี้มีความสัมพันธ์ต่อกันและมีผลโดยตรงต่อคำตอบสุดท้ายจำเป็นต้องบูรณาการวิธีการหาคำตอบเข้าด้วยกัน ต่อมา Laporte (1992) นำเสนอการจัดตั้งแบบจำลองทางคณิตศาสตร์สำหรับปัญหาเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งเป็นท่านแรกการแก้ปัญหาในช่วงนั้น ได้พัฒนาวิธีการแก้ปัญหาแบบค้นหาคำตอบที่ดีที่สุด เช่น วิธีการแผนภูมิการตัดสินใจแบบต้นไม้ (Decision Thee), วิธี Branch and Bound algorithm, วิธี Gomory Cut และวิธี Graph Transformation และในบทความดังกล่าวยังแยกระดับความสำคัญของปัญหาออกเป็น 2 ปัญหา คือ ปัญหาหลัก (Master Problem) และปัญหารอง (Sub Problem) ปัญหาการเลือกสถานที่ตั้งถูกจัดอยู่ในปัญหาหลักโดยใช้วิธี Tree Search Algorithm หลังจากได้คำตอบเสร็จเรียบร้อยแล้วจะแก้ปัญหารองต่อนั่นคือ ปัญหาการจัดเส้นทางการขนส่งโดยใช้วิธีการแก้ปัญหาด้วยวิธี Branch and Bound คำตอบที่ได้จากวิธีการนี้เป็นคำตอบที่ดีที่สุด (Optimal Solution) แต่วิธีการดังกล่าวยังมีข้อจำกัดในด้านขนาดของปัญหาซึ่งขนาดของปัญหาเป็นปัญหาขนาดเล็กลูกค้าไม่เกิน 40 ราย

Nagy and Salhi (1996) นำเสนอวิธีฮิวริสติกเพื่อแก้ปัญหาข้อจำกัดของขนาดปัญหา โดยวิธีฮิวริสติกที่นำมาใช้เป็นวิธีการประมาณการแบ่งลูกค้าออกเป็นกลุ่มๆ เรียกว่า "Cluster-Based" ในการจัดสรรลูกค้าให้กับศูนย์กระจายสินค้าและขั้นตอนต่อจากนั้นจะเป็นการสร้างเส้นทางด้วยวิธี Clarke and Wright Saving Heuristic อีก 3 ปีต่อมา Tuzun and Burke (1999) ได้นำเสนอวิธีการค้นหาต้องห้าม (Tabu Search) แบบสองเฟสเพื่อใช้ในการแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งที่มีขนาดของปัญหา 200 ลูกค้าโดยเปรียบเทียบคำตอบที่ได้และเวลาที่ใช้กับวิธีฮิวริสติกแบบ Saving ซึ่งผลที่ได้ปรากฏว่าวิธีการค้นหาต้องห้าม ที่ออกแบบมาได้คำตอบที่ดีกว่าวิธี Saving แต่ใช้เวลาในการคำนวณมากกว่า

Wu et al. (2002) ได้ระบุถึงรูปแบบของตัวแบบคณิตศาสตร์ของปัญหานี้ว่าโดยทั่วไปจะประกอบไปด้วย จำนวนสถานที่ตั้ง ความต้องการสินค้าของลูกค้า จำนวนสถานที่ตั้งของจุดที่น่าจะตั้งศูนย์กระจายสินค้า ตลอดจนชนิดและขนาดของยานพาหนะ ซึ่งข้อมูลเหล่านี้จะสามารถหาหรือกำหนดได้ ส่วนการวางแผนการกระจายสินค้าและจัดเส้นทางการขนส่งนั้นจะต้องได้รับการออกแบบเพื่อตอบสนองความต้องการของลูกค้าแต่ละราย ลูกค้าแต่ละรายจะได้รับการจัดส่งสินค้า

โดยยานพาหนะหนึ่งคันอย่างแน่นอน ความต้องการโดยรวมในแต่ละเส้นทางจะต้องน้อยกว่าหรือเท่ากับความจุของยานพาหนะที่ได้รับมอบหมายให้ไปในเส้นทางนั้นๆ และแต่ละเส้นทางจะต้องเริ่มและจบที่ศูนย์กระจายสินค้าเดียวกัน ตามที่ได้ออกแบบไว้ Wu et al. ได้ทำการทดลองใช้วิธีผสมผสานระหว่างวิธีการค้นหาต้องห้าม (Tabu Search) กับวิธีการเลียนแบบการอบอ่อน (Simulated Annealing) เพื่อแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งโดยแบ่งการทำงานออกเป็นสองเฟสคือ (1) เฟสของการเลือกสถานที่ตั้ง (2) เฟสของการจัดเส้นทางการขนส่ง แล้วทำการเปรียบเทียบคำตอบที่ได้และเวลาที่ใช้ในการคำนวณกับงานของ Perl and Daskin (1985) ซึ่งได้แบ่งการแก้ปัญหาออกเป็นสามเฟส ผลการทดลองพบว่าวิธีการเลียนแบบการอบอ่อนสามารถให้คำตอบที่ดีกว่าวิธี Saving Method และยังใช้เวลาในการคำนวณน้อยกว่าอีกด้วยเนื่องจากมีการลดขั้นตอนในการคำนวณลงจากเดิม

Jitimar (2003) นำเสนอวิธีการของ Ant Colony Optimization (ACO) ในการแก้ปัญหาการเลือกสถานที่ตั้งโรงงานผลิตก๊าซชีวภาพจากมูลสุกร เพื่อใช้ในการแก้ปัญหากลิ่นจากมูลสุกรกับเขตชุมชนการแก้ปัญหานี้ ได้มีการแบ่งปัญหาออกเป็น 2 ปัญหาคือ ปัญหาสถานที่ตั้งโรงงานและปัญหาการจัดเส้นทางสำหรับยานพาหนะขนส่ง และใช้วิธี ACO เป็นเครื่องมือในการแก้ปัญหาสามารถวิเคราะห์หาตำแหน่งที่โรงงานผลิตชีวภาพและใช้ระยะทางการขนส่งที่เหมาะสม

สุพรรณ สุดสนธิ์ และสมบัติ สินธุเชาวน์ (2549) ได้ประยุกต์วิธีการหาคำตอบโดยวิธีอัลกอริทึมเชิงพันธุกรรม (Genetic algorithm: GA) สำหรับปัญหาการตัดสินใจเลือกสถานที่ตั้งศูนย์กระจายสินค้าและการคัดเลือกลูกค้าให้กับศูนย์กระจายสินค้าแบบสมดุลความต้องการสินค้า พบว่าฮิวริสติกที่พัฒนาขึ้นมาสามารถค้นหาผลลัพธ์จากปัญหาได้ในระดับที่น่าพอใจ ถ้าเป็นปัญหาขนาดของลูกค้าไม่เกิน 20 ราย จะพบผลลัพธ์เป็นคำตอบที่ดีที่สุด แต่ถ้าปัญหามีขนาดใหญ่ขึ้นค่าเปอร์เซ็นต์ความผิดพลาดจะมากขึ้นตามลำดับ ระดับพารามิเตอร์ตัวดำเนินการจะเปลี่ยนแปลงไปตามขนาดของปัญหาและมีผลโดยตรงกับเวลาที่ใช้ประมวลผลเพิ่มมากขึ้น

Nagy and Salhi (2007) ได้สรุปรูปแบบของปัญหาการเลือกสถานที่และจัดเส้นทางการขนส่งของยานพาหนะ โดยแบ่งออกเป็นกลุ่มๆ ตามโครงสร้างของปัญหา ชนิดของข้อมูล ช่วงเวลาในการวางแผน วิธีการที่ใช้ในการแก้ปัญหา ฟังก์ชันวัตถุประสงค์ จำนวนศูนย์กระจายสินค้า โครงสร้างของเส้นทาง ฯลฯ ผู้ศึกษาวิจัยปัจจุบันส่วนมากจะอยู่ในรูปแบบของการหาที่ตั้งของศูนย์กระจายสินค้าเพื่อที่จะส่งสินค้าไปยังลูกค้าจำนวนหนึ่ง โดยมีลักษณะของข้อมูลเป็นแบบดีเทอร์มินีสติกและส่วนใหญ่จะใช้วิธีการแบบฮิวริสติกในการแก้ปัญหา ในส่วนของวิธีการแก้ปัญหาแบบทางตรง (Exact Method) จะมีข้อจำกัดในเรื่องของขนาดของปัญหา โดยสามารถแก้ปัญหาด้วยวิธี Exact Method จะเหมาะกับปัญหาที่มีขนาดเล็กหรือมีจำนวนลูกค้าไม่เกิน 40 ราย

ดังนั้นวิธีฮิวริสติกจึงถูกนำมาใช้ในการแก้ปัญหาที่มีขนาดใหญ่ขึ้นและมีความซับซ้อนมากขึ้น โดยวิธีฮิวริสติกสำหรับการแก้ปัญหาการเลือกสถานที่ตั้งและการจัดเส้นทางการขนส่งของยานพาหนะสามารถแบ่งได้เป็น 4 วิธีการหลักๆ คือ (1) การแก้ปัญหาแบบลำดับขั้น (Sequential methods) เป็นวิธีการที่แก้ปัญหาตามองค์ประกอบของปัญหา ซึ่งหมายถึงการจัดแยกปัญหาขนาดใหญ่ออกเป็นปัญหาขนาดเล็กหรือโดยทั่วไปเรียกกันว่า "Cluster first and route second" ซึ่งมีผู้วิจัยทำงานด้านนี้อยู่หลายท่านด้วยกัน เช่น Webb (1968), Watson-Gandy and Dohrn (1973), Christofides and Elilon (1979), Balakrishan et al. (1987) และ Srtivastava and Benton (1990) นำเสนอการใช้วิธี Sequential Solution กับการแก้ปัญหาสถานที่ตั้งศูนย์กระจายสินค้าและการจัดเส้นทางสำหรับพาหนะขนส่ง, (2) การแก้ปัญหาแบบจัดกลุ่ม (Cluster-based method) เช่น เซตที่ 1 เป็นเซตของลูกค้ากับศูนย์กระจายสินค้าที่ถูกเลือกให้เปิดดำเนินงาน อาศัยหลักการแบ่งกลุ่มจากหลักการเพื่อนบ้านใกล้เคียง (Nearest Neighborhood Search) เซตที่ 2 จะแบ่งกลุ่มเซตจากเซตที่ 1 ออกเป็นสับเซตเล็กๆ คือเซตลูกค้าที่ถูกกำหนดหรือเลือกให้ใช้พาหนะขนส่งบนเส้นทางเดียวกัน ต่อจากนั้นก็จะทำการแก้ปัญหาแต่ละกลุ่มเซต วิธีการของ Clustering-based จะมีลักษณะคล้ายคลึงกับวิธี Sequential Methods คือไม่มีการพิจารณาผลลัพธ์ย้อนกลับในการแก้ปัญหาระหว่างสองปัญหาย่อย(คำตอบที่ได้จากปัญหาแรกจะไม่ถูกนำไปใช้กับปัญหาที่สอง) จะถูกแก้ทีละปัญหานั่นเอง, (3) การแก้ปัญหาแบบทำซ้ำ (Iterative method) เป็นวิธีการแก้ปัญหาแบบการแยกย่อยปัญหาขนาดใหญ่ออกเป็นปัญหาขนาดเล็ก ปัญหาสถานที่ตั้งศูนย์กระจายสินค้าและปัญหาการจัดเส้นทางสำหรับพาหนะขนส่งนี้จะแยกปัญหาออกเป็น 2 ปัญหาย่อย จากนั้นก็นำวิธีการของ Iterative มาแก้ปัญหาทีละปัญหา ซึ่งคำตอบที่ได้จากขั้นตอนแรก จะส่งผลถึงการแก้ปัญหาของเฟสอื่นๆ ด้วย จุดสำคัญของการแก้ปัญหานี้กับวิธีการแก้ปัญหาด้วยวิธีของ Iterative จะต้องทราบว่าจุดไหนบ้างที่จะต้องเชื่อมโยงกันระหว่างเฟสแรกกับเฟสอื่นๆ เพื่อส่งผลคำตอบที่ดีขึ้นในทุกๆรอบกระทำซ้ำ และ (4) การแก้ปัญหาแบบฮิวริสติก (Hierarchical method) คือวิธีการของฮิวริสติกที่ใช้ในการแก้ปัญหาแบบลำดับชั้น เช่น การแก้ปัญหาหลักและปัญหารอง ทั้งสองปัญหาจะมีผลสืบทอดซึ่งกันและกัน เช่น การแก้ปัญหาสถานที่ตั้งศูนย์กระจายสินค้า (Location Problem) ไปจนจบกระบวนการเสียก่อนแล้วค่อยทำการแก้ปัญหาการจัดเส้นทางสำหรับพาหนะที่เป็นปัญหารอง แล้วกลับมาเริ่มต้นใหม่อีกครั้งหนึ่ง การเริ่มต้นใหม่ในรอบกระทำซ้ำนี้จะถูกแก้ไขคำตอบใหม่ให้ดีขึ้นและทำไปจนกระทั่งครบเงื่อนไขในการหยุดที่ตั้งไว้แล้วจึงค่อยส่งผลลัพธ์ที่ได้ออกมา ซึ่งมีผู้วิจัยหลายท่านเชื่อว่าเป็นวิธีการที่สามารถพัฒนาคำตอบให้ดีขึ้นไปเรื่อยๆ ตามลำดับ ตารางที่ 2.2 สรุปรูปแบบของปัญหาและวิธีการแก้ปัญหาด้วยวิธีทางตรง (Exact Optimization) และวิธีฮิวริสติก (Heuristic Optimization) จากการสรุปของ Nagy and Salhi (2007)

ตารางที่ 2.2 สรุปภาพรวมของงานวิจัยที่ใช้วิธีทางตรงและวิธีฮิวริสติกในการแก้ปัญหา LRP

ประเภทของปัญหา	วิธีการแก้ปัญหา	ผู้วิจัย	จำนวนศูนย์กระจายสินค้า	จำนวนลูกค้า
General	Cutting planes	Laporte et al. (1983)	40	40
deterministic LRP	Branch-and-bound	Laporte et al. (1988)	40	40
	Clustering-based	Barreto et al. (in press)	15	318
	Iterative	Salhi and Fraser (1996)	199	199
	Hierarchical	Nagy and Salhi (1996)	400	400
	Hierarchical	Albareda-Sambola et al. (2005)	10	30
	Hierarchical	Melechovsky et al. (2005)	20	240
Round-trip location	Numerical optimization	Drezner (1982)	3	80
Eulerian location	Brand-and-cut	Ghiani and Laporte (1999)	50	200
Minimax TS location	Graph theoretical	Averbakh and Berman (2002)	1	ไม่ระบุ
Plant cycle location	Branch-and-cut	Labbe et al. (2004)	30	120
	Clustering-based	Billionnet et al. (2005)	6	70
Planar LRP	Iterative	Salhi and Nagy (in review)	ไม่ระบุ	199

Sodsoon and Sindhuchao (2007) เสนอเทคนิคการแก้ปัญหาด้วย Max-Min ant System กับปัญหาศูนย์กระจายสินค้าที่มีหลายแห่ง โดยสินค้าที่ไปส่งเป็นสินค้าชนิดเดียวและทุกช่วงเวลาคงที่ จากนั้นทำการหาเส้นทางการขนส่งที่สั้นที่สุดจากพาหนะทุกคัน ขั้นตอนในการแก้ปัญหาแบ่งออกเป็นสองระยะคือ ระยะที่หนึ่งใช้ Max-Min ant System เป็นวิธีการสร้างคำตอบเริ่มต้น จำนวนของมดเท่ากับจำนวนศูนย์กระจายสินค้า ซึ่งมดแต่ละตัวเข้าประจำที่ศูนย์กระจายสินค้าแต่ละแห่ง แล้วทำการหมุนเวียนให้มดแต่ละตัวทำการเลือกโหนดลูกค้าในบัญชี (Candidate List) เพื่อสร้างเส้นทางพาหนะขนส่งและกำหนดลูกค้าให้ศูนย์กระจายสินค้าไปพร้อมๆ กัน ระยะที่สองคือ ทำการปรับปรุงคุณภาพคำตอบด้วยวิธี 2-opt พบว่าคำตอบที่ได้อยู่ในเกณฑ์ที่น่าพอใจและใช้เวลาอย่างเหมาะสมในการคำนวณหาผลเฉลย

สุพรรณ สุดสนธิ์และคณะ (2550) นำเสนอวิธีอาณานิคมมดและขั้นตอนการปรับปรุงคุณภาพคำตอบด้วยการสลับเปลี่ยนตำแหน่ง (Swap), การย้าย (Move), 2-opt และ 3-opt แก้ปัญหาสถานที่ตั้งศูนย์กระจายสินค้าแบบหลายแห่งและการจัดเส้นทางการขนส่ง (Multi Depot Vehicle Routing Problem: MDVRP) โดยทำการทดสอบกับปัญหาจาก OR-Library พบว่าวิธีการที่นำเสนอ

ให้คำตอบในภาพรวมที่มีเปอร์เซ็นต์ความผิดพลาดอยู่ระหว่าง 0.00%-7.43% และใช้เวลาในการค้นหาคำตอบอย่างเหมาะสม

Luarent et al. (2009) ได้ทำการทดลองใช้วิธีการค้นหาคำตอบเฉพาะที่แบบวนซ้ำกับปัญหาจัดเส้นทางการขนส่งแบบหลายแหล่งกระจายสินค้า ซึ่งได้มีการเปรียบเทียบผลเฉลยที่ได้กับอัลกอริทึมแบบวิธีการค้นหาต้องห้ามและวิธีทางตรงอื่นๆ เช่น Column Generation และ Lagrangian Relaxation โดยผลที่ได้พบว่าวิธีการค้นหาคำตอบเฉพาะที่แบบวนซ้ำใช้เวลาในการคำนวณน้อยกว่าและสามารถให้ผลเฉลยที่แตกต่างจากคำตอบที่ดีที่สุด เช่นเดียวกันกับ Houda et al. (2010) ได้มีการเปรียบเทียบอัลกอริทึมของวิธีการค้นหาเฉพาะที่แบบวนซ้ำกับวิธีการค้นหาต้องห้ามสำหรับปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง ผลปรากฏว่าวิธีการค้นหาคำตอบเฉพาะที่สามารถให้ผลเฉลยที่ดีกว่าโดยใช้เวลาในการคำนวณอย่างเหมาะสม

นัทธพงศ์ นันทสำเริง และระพีพันธ์ ปิตาคะโส (2553) นำเสนอวิธีการเมต้าฮิวริสติกโดยวิธีค้นหาเนเบอร์ฮูดผันแปรสำหรับปัญหาการเลือกสถานที่ตั้งแบบหลายขั้นตอนและหลายเงื่อนไข โดยมีเงื่อนไข 3 ประการคือ (1) ต้องใช้ต้นทุนในการดำเนินการและต้นทุนในการขนส่งน้อยที่สุด (2) ต้องส่งผลกระทบต่อสิ่งแวดล้อมน้อยที่สุด และ (3) ต้องมีความเสี่ยงต่อชุมชนอันเกิดจากการสร้างโรงงานและการขนส่งน้อยที่สุด ผลการศึกษาพบว่าวิธีค้นหาเนเบอร์ฮูดผันแปรโครงสร้างที่ไม่ซับซ้อนและให้ผลเฉลยที่ยอมรับได้เมื่อเทียบกับการแก้ปัญหาโดยใช้ซอฟท์แวร์สำเร็จรูป LINGO

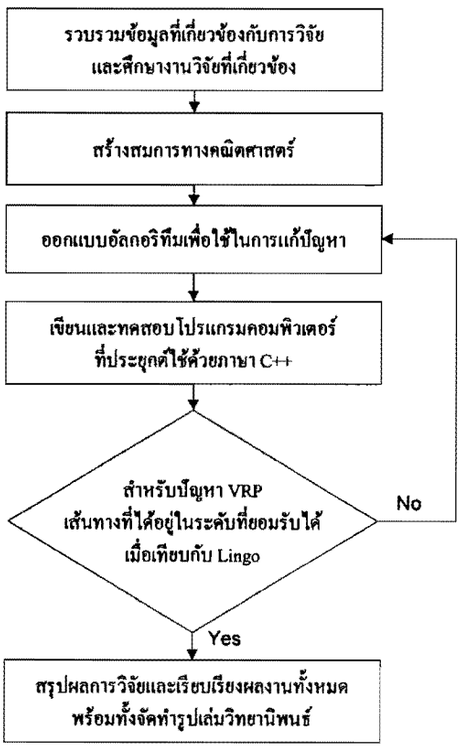
สรุปแล้วปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งมีการศึกษากันอย่างแพร่หลายและพัฒนาแยกปัญหาในลักษณะต่างๆ อีกหลายรูปแบบซึ่งตัวปัญหามีความซับซ้อนในระดับเอ็นพีฮาร์ด (NP-Hard) แนวทางที่ใช้สำหรับการแก้ปัญหามีอยู่ 2 ประเภทใหญ่ๆ คือ วิธีการหาคำตอบที่ดีที่สุด (Exact Optimization) และการหาคำตอบด้วยวิธีการแบบฮิวริสติก (Heuristic Optimization) ต่อมานักวิจัยได้พัฒนาวิธีฮิวริสติกแบบดั้งเดิมและฮิวริสติกที่ได้รับการพัฒนามีชื่อเรียกในยุคใหม่นี้ว่า เมต้าฮิวริสติก (Meta-Heuristic) ถึงแม้ว่าวิธีดังกล่าวจะไม่ได้รับประกันว่าจะให้คำตอบที่เหมาะสมที่สุด แต่ก็สามารถให้คำตอบที่น่าพอใจ โดยใช้เวลาในการคำนวณสมเหตุสมผล จึงทำให้เหมาะสมที่จะนำวิธีเมต้าฮิวริสติกที่พัฒนามาใช้เป็นแนวทางในการแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง เพื่อที่จะสามารถจัดการกับปัญหาได้ทันต่อเหตุการณ์

บทที่ 3
วิธีการดำเนินการวิจัย

ในบทนี้จะกล่าวถึงขั้นตอนในการดำเนินงานวิจัย ซึ่งจะอธิบายตั้งแต่ข้อมูลกรณีศึกษา การเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งสำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลังในภาคตะวันออกเฉียงเหนือ แบบจำลองทางคณิตศาสตร์ ระบบอาณานิคมมด โครงสร้างของวิธีระบบอาณานิคมมดสำหรับการแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางสำหรับยานพาหนะ รวมถึงข้อมูลรายละเอียดของคอมพิวเตอร์และโปรแกรมที่ใช้ในการประมวลผล

3.1 ขั้นตอนการดำเนินงาน

ขั้นตอนการดำเนินงานในการแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง โดยมีรายละเอียดดังภาพที่ 3.1



ภาพที่ 3.1 ลำดับขั้นตอนการดำเนินงานวิจัย

3.2 กรณีศึกษาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่งสำหรับอุตสาหกรรมการผลิตแป้งมันสำปะหลังในภาคตะวันออกเฉียงเหนือ

ข้อมูลกรณีศึกษาที่ใช้ในงานวิจัยประกอบด้วย

3.2.1 ข้อมูลแหล่งวัตถุดิบได้แก่ ข้อมูลของลานมันสำปะหลังในเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับกรมอุตสาหกรรมจังหวัดมีจำนวน 504 โรงงาน (ภาคผนวก ก) เพื่อให้ง่ายต่อการคำนวณทางคณะผู้วิจัยได้ใช้ตำบลเป็นจุดอ้างอิง จึงทำการรวมโรงงานที่ตั้งอยู่ในจังหวัด อำเภอ และตำบลเดียวกัน ไว้ในจุดอ้างอิงเดียวกัน โดยทำการรวมเงินทุนหมุนเวียนต่อปีของโรงงานเหล่านั้นไว้ในจุดอ้างอิงด้วยเช่นกัน เมื่อทำการรวมโรงงานตามเกณฑ์ดังกล่าวแล้วทำให้มีตำแหน่งของแหล่งวัตถุดิบจำนวน 261 แห่ง สำหรับความสามารถในการป้อนวัตถุดิบให้กับโรงงานแป้งมันสำปะหลังของแหล่งวัตถุดิบแต่ละแห่งคำนวณจากสมการที่ 3.1

$$\text{ความสามารถในการป้อนวัตถุดิบ } (\text{ตัน/วัน}) = \frac{\text{เงินทุนหมุนเวียน (บาท/ปี)}}{365 * \text{ราคาหัวมันสด (บาท/ตัน)}} \qquad (3.1)$$

โดยราคาหัวมันสดเท่ากับ 2.58 บาทต่อกิโลกรัม (อ้างอิง : รายงานสถานการณ์สินค้าเกษตรสำคัญ ธนาคารแห่งประเทศไทยสำนักงานภาคตะวันออกเฉียงเหนือ, พฤษภาคม 2554)

3.2.2 ข้อมูลลูกค้าได้แก่ ข้อมูลของโรงงานที่ใช้แป้งมันสำปะหลังดิบเป็นวัตถุดิบในการผลิตสินค้าที่จดทะเบียนโรงงานกับกรมอุตสาหกรรมจังหวัดมีจำนวน 29 โรงงาน (ภาคผนวก ข) เพื่อให้เป็นหน่วยเดียวกันกับแหล่งวัตถุดิบ จึงทำการรวบโรงงานที่ตั้งอยู่ในจังหวัด อำเภอ และตำบลเดียวกัน ไว้ในจุดอ้างอิงเดียวกัน เมื่อทำการรวบแล้วทำให้มีตำแหน่งของลูกค้าจำนวน 22 แห่ง

3.2.3 ข้อมูลระยะทางระหว่างลูกค้ากับแหล่งวัตถุดิบทั้งหมด ความต้องการของลูกค้าและความสามารถในการป้อนวัตถุดิบของลานมัน (ภาคผนวก ค)

3.2.4 ความสามารถในการบรรทุกสินค้าของรถบรรทุกไม่เกิน 15 ตันต่อคัน

3.2.5 ข้อมูลโรงงานแป้งมันสำปะหลังที่จะทำการหาตำแหน่งที่ตั้ง มีกำลังการผลิตไม่เกิน 300 ตันแป้งต่อวันหรือ ใช้วัตถุดิบ (หัวมันสด) ไม่เกิน 900 ตันต่อวัน

3.3 แบบจำลองทางคณิตศาสตร์

ลักษณะของแบบจำลองทางคณิตศาสตร์มีลักษณะของปัญหาดังต่อไปนี้

3.3.1 แบบจำลองทางคณิตศาสตร์นี้เป็นแบบจำลองของปัญหาการจัดเส้นทางการขนส่งสำหรับอุตสาหกรรมแป้งมันสำปะหลัง โดยที่ทราบตำแหน่งที่ตั้งของโรงงานแป้งมันสำปะหลัง กลุ่มลูกค้าที่จะทำการส่งสินค้า และกลุ่มลานมันที่จะไปรับวัตถุดิบ ซึ่งข้อมูลดังกล่าวได้จากการหาคำตอบด้วยอัลกอริทึมที่ออกแบบ

3.3.2 ความต้องการแป้งมันสำปะหลังของลูกค้าแต่ละรายจะถูกแบ่งออกเป็นโหนดความต้องการ (Demand Node) หลายๆโหนด โดยที่จำนวนโหนดความต้องการของลูกค้าแต่ละรายจะเท่ากับจำนวนครั้งที่รถขนส่งเดินทางไปส่งแป้งมันสำปะหลังให้กับลูกค้ารายนั้น ซึ่งทราบได้จากการแก้ปัญหาด้วยอัลกอริทึมที่ออกแบบไว้

3.3.3 ในการสร้างแบบจำลองทางคณิตศาสตร์ จะสมมติว่าโหนดความต้องการหนึ่งโหนดคือโหนดของลูกค้าหนึ่งรายซึ่งมีความต้องการแป้งมันสำปะหลังเท่ากับปริมาณที่รถขนส่งเดินทางไปส่งให้ในเที่ยวนั้น ซึ่งทราบปริมาณนี้ได้จากการแก้ปัญหาด้วยอัลกอริทึมที่ออกแบบไว้

3.3.4 ปริมาณหัวมันที่ลานมันแต่ละแห่งมีอยู่ จะถูกแบ่งออกเป็นโหนดหัวมัน (Supply Node) หลายๆโหนด โดยที่จำนวนโหนดหัวมันของลานมันแต่ละแห่งจะเท่ากับจำนวนครั้งที่รถขนส่งเดินทางไปรับหัวมันที่ลานมันนั้น ซึ่งทราบได้จากการแก้ปัญหาด้วยอัลกอริทึมที่ออกแบบไว้

3.3.5 ในการสร้างแบบจำลองทางคณิตศาสตร์ จะสมมติว่าโหนดหัวมันหนึ่งโหนดคือโหนดลานมันหนึ่งแห่งซึ่งมีปริมาณหัวมันเท่ากับปริมาณที่รถขนส่งเดินทางไปรับในเที่ยวนั้น ซึ่งทราบปริมาณนี้ได้จากการแก้ปัญหาด้วยอัลกอริทึมที่ออกแบบไว้

3.3.6 การจัดเส้นทางการขนส่ง รถขนส่งจะต้องเดินทางไปส่งสินค้าให้กับลูกค้าจนกระทั่งสินค้าที่บรรทุกไปบนรถขนส่งนั้นหมดลง รถขนส่งจึงจะสามารถเดินทางไปรับวัตถุดิบที่ลานมันได้

3.3.7 สินค้าทั้งหมดที่ต้องนำไปส่งยังลูกค้าแต่ละรายนั้นจะต้องบรรทุกขึ้นรถขนส่งจากโรงงานแป้งมันสำปะหลังเท่านั้น

3.3.8 ไม่มีข้อกำหนดเรื่องเวลาที่ต้องรับ-ส่ง (No Time Windows) หรือลำดับก่อนหลังในการรับ-ส่งสินค้า

3.3.9 มีจำนวนรถขนส่งเพียงพอต่อการขนส่งเสมอและรถขนส่งมีความสามารถในการบรรทุกสินค้าในปริมาณที่เท่ากัน

3.3.10 เมื่อโรงงานเลือกส่งสินค้าให้ลูกค้ารายใดไม่จำเป็นต้องส่งให้ครบตามความต้องการของลูกค้ารายนั้น โดยสินค้าที่ส่งให้ลูกค้ารวมกันทั้งหมดจะต้องเท่ากับปริมาณที่โรงงานผลิต

3.3.11 เมื่อโรงงานเลือกรับหัวมันจากลานมันใดไม่จำเป็นต้องรับให้ครบตามความสามารถที่ลานมันนั้นส่งให้โรงงานได้ โดยโรงงานจะรับวัตถุดิบจากลานมันรวมกันทั้งหมดเท่ากับปริมาณขนาดการผลิตของโรงงาน

3.3.1 ดัชนี (Indices)

i, j	ใช้แทนลำดับของโหนด โดยที่โรงงานจะมี $i, j = 1$
	โหนดลูกค้า $i, j \in M$ และโหนดลานมันสำปะหลัง $i, j \in N$
k	ใช้แทนลำดับของรถเที่ยวที่ $k = 1,2, \ldots, K$

3.3.2 ตัวแปรที่ทราบค่า (Parameter)

W	จำนวนโหนดทั้งหมด
M	เซตของโหนดลูกค้าที่ถูกเลือก (Demand Node)
N	เซตของโหนดลานมันสำปะหลังที่ถูกเลือก (Supply Node)
S	เซตของโหนดทั้งหมด ($S = 1 \cup M \cup N$)
R_{ij}	ระยะทางระหว่างโหนดที่ i ไปยังโหนดที่ j
H_i	ปริมาณหัวมันที่โหนดลานมัน i ที่สามารถส่งให้กับโรงงานได้
D_i	ความต้องการแป้งมันสำปะหลังของโหนดลูกค้า i
Q	ความจุของรถบรรทุก

$M \cap N = \emptyset$

3.3.3 ตัวแปรในการตัดสินใจ (Decision Variables)

$$X_{ijk} \begin{cases} 1, & \text{ถ้าโหนด } i \text{ เชื่อมกับโหนด } j \text{ บนเส้นทางของรถเที่ยวที่ } k \\ 0, & \text{อื่นๆ} \end{cases}$$

3.3.4 ฟังก์ชั่นวัตถุประสงค์ (Objective Function)

$$Min = \sum_{i\epsilon S}\sum_{j\epsilon S}\sum_{k=1}^{K} R_{ij}X_{ijk} \tag{3.2}$$

3.3.5 สมการข้อจำกัด (Constraints)

$$\sum_{i\epsilon S}\sum_{k=1}^{K} X_{ijk} = 1 \qquad \forall_{j\epsilon M\cup N} \tag{3.3}$$

$$\sum_{j\epsilon S}\sum_{k=1}^{K} X_{ijk} = 1 \qquad \forall_{i\epsilon M\cup N} \tag{3.4}$$

$$\sum_{i\epsilon S} X_{i1k} \leq 1 \qquad \forall_{k=1,2,\ldots,K} \tag{3.5}$$

$$\sum_{j\epsilon S} X_{1jk} \leq 1 \qquad \forall_{k=1,2,\ldots,K} \tag{3.6}$$

$$\sum_{i\epsilon S} X_{ijk} - \sum_{i\epsilon S} X_{jik} = 0 \qquad \forall_{j\epsilon S,k=1,2,\ldots,K} \tag{3.7}$$

$$\sum_{i\epsilon M} D_i \sum_{j\epsilon S} X_{ijk} \leq Q \qquad \forall_{k=1,2,\ldots,K} \tag{3.8}$$

$$\sum_{i\epsilon N} H_i \sum_{j\epsilon S} X_{ijk} \leq Q \qquad \forall_{k=1,2,\ldots,K} \tag{3.9}$$

$$X_{ijk} = 0 \qquad \forall_{i\epsilon N,j\epsilon M,k=1,2,\ldots,K} \tag{3.10}$$

$$\sum_{i\epsilon M}\sum_{j\epsilon N} X_{ijk} \leq 1 \qquad \forall_{k=1,2,\ldots,K} \tag{3.11}$$

$$X_{iik} = 0 \qquad \forall_{i\epsilon S,k=1,2,\ldots,K} \tag{3.12}$$

$$(W-3)X_{jik} + (W-1)X_{ijk} + U_i \leq U_j + (W-2) \qquad \forall_{i,j\epsilon M\cup N;i\neq j,k=1,2,\ldots,K} \tag{3.13}$$

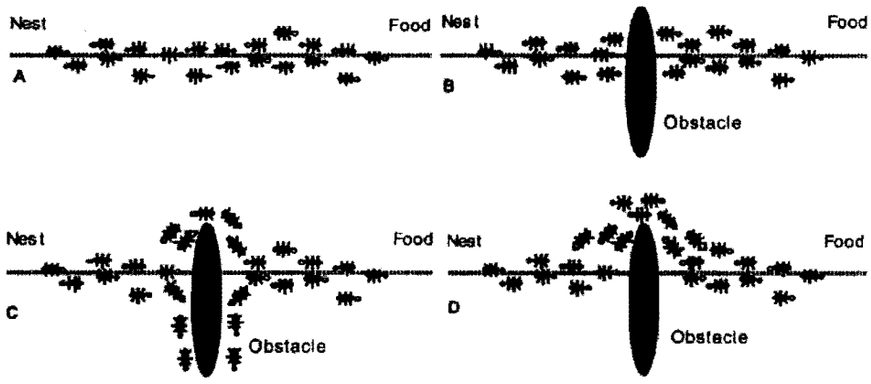
$$X_{ijk} = \{0,1\} \qquad \forall_{i\epsilon S,j\epsilon S,k=1,2,\ldots,K} \tag{3.14}$$

สมการที่ (3.2) เป็นสมการวัตถุประสงค์โดยมีเป้าหมายเพื่อหาระยะทางรวมที่รถขนส่งทุกคันวิ่งไปส่งสินค้าและรับวัตถุดิบให้มีค่าน้อยที่สุด สมการที่ (3.3) และ (3.4) เป็นสมการเงื่อนไขกำหนดให้ลูกค้าและลานมันแต่ละโหนดสามารถรับบริการจากรถขนส่งได้เพียง 1 คันเท่านั้น

สมการที่ (3.5) และ (3.6) เป็นสมการเงื่อนไขที่กำหนดว่า หากรถขนส่งมีการเดินทาง ในแต่ละเที่ยวขนส่ง รถจะต้องเดินทางออกจากโรงงานและต้องเดินทางกลับสู่โรงงาน สมการที่ (3.7) บังคับว่าหากรถขนส่งเดินทางเข้าโหนดจะต้องเดินทางออกจากโหนดนั้น สมการที่ (3.8) ปริมาณสินค้าที่ไปส่งให้ลูกค้าโดยรถแต่ละคันจะต้องไม่เกินความจุของรถขนส่ง สมการที่ (3.9) ปริมาณหัวมันที่ไปรับโดยรถขนส่งแต่ละคันจะต้องไม่เกินความจุของรถขนส่ง สมการที่ (3.10) และ (3.11) บังคับให้รถขนส่งต้องเดินทางไปส่งสินค้าให้ลูกค้าก่อนแล้วจึงเดินทางไปรับวัตถุดิบจากลานมัน สมการที่ (3.12) เป็นสมการเงื่อนไขบังคับให้ไม่สามารถเชื่อมเส้นทางที่เป็นโหนดเดียวกันได้ สมการที่ (3.13) สมการป้องกันการเกิดเส้นทางย่อย (Subtour Elimination Constraint) และสมการที่ (3.14) เป็นการกำหนดตัวแปรตัดสินใจแบบไบนารี่

3.4 ระบบอาณานิคมมด (Ant Colony Optimization: ACO)

วิธีอาณานิคมมดได้แรงบันดาลใจมาจากพฤติกรรมการออกหาอาหารของมด มดจะเดินทางจากรังไปสู่แหล่งอาหารและกลับมาที่รังอีกครั้งหลังจากได้อาหารแล้ว ดังภาพที่ 3.2 โดยเริ่มแรกของการเลือกเส้นทาง เพื่อจะเดินไปสู่แหล่งอาหารของมดนั้น จะเริ่มจากการลองผิดลองถูก แต่ระหว่างที่มดเดินทางนั้นมดจะทำการปล่อยสารเคมีที่เรียกว่า ฟีโรโมน (Pheromone) เพื่อให้มดตัวอื่นๆ เดินตามกลิ่นหรือร่องรอยของฟีโรโมนนั้นๆ ฟีโรโมนมีคุณสมบัติที่สามารถระเหยไปได้เมื่อระยะเวลาผ่านไป ดังนั้นเมื่อระยะทางที่ยาวจนเกินไปจะทำให้ฟีโรโมนระเหยหรือเจือจางระหว่างเดินทางในขณะเดียวกันหากระยะทางสั้นจะทำให้มีโอกาสเพิ่มฟีโรโมนทำให้ฟีโรโมนมีความเข้มข้นขึ้น จากการศึกษาพบว่ามดชอบเดินตามรอยฟีโรโมนที่มีกลิ่นแรงจึงส่งผลให้มีโอกาสเพิ่มมากขึ้นที่มดตัวอื่นๆ จะเดินตามเส้นทางที่มีระยะทางที่สั้นกว่าเพื่อไปยังแหล่งอาหาร



ภาพที่ 3.2 ธรรมชาติการเดินทางของมด (Dorigo, Di Caro and Gambardella., 1997)

ระบบมด (Ant System: AS) ได้ถูกนำมาเผยแพร่ครั้งแรกโดย Macro Dorico ในวิทยานิพนธ์ระดับปริญญาเอกของเขาในปี ค.ศ. 1992 และตั้งแต่ปี ค.ศ. 1995 เป็นต้นมา Dorico ร่วมมือกับ Gambardella และ Stutzle พัฒนาวิธีการ AS เป็นวิธีการอาณานิคมมดโดยปรับปรุงประสิทธิภาพให้ช่วยหาคำตอบได้ดีขึ้น วิธีอาณานิคมมดประกอบด้วยขั้นตอนดังนี้

3.4.1 การกำหนดพารามิเตอร์ที่เกี่ยวข้อง เช่น จำนวนรอบที่ต้องการวนซ้ำ, จำนวนประชากรมด, กำหนดค่าฟีโรโมน (τ) เริ่มต้นที่ 1 เท่ากันหมด ยกเว้นการเชื่อมเมืองเดียวกัน เช่น 1-1 หรือ 2-2 จะมีค่าเป็นศูนย์ เนื่องจากไม่สามารถดำเนินการได้ อัตราการระเหยของค่าฟีโรโมน (ρ) ซึ่ง $0 < \rho < 1$ ค่าพารามิเตอร์ (α) โดยที่ $\alpha > 0$ และพารามิเตอร์ (β) โดยที่ $\beta > 0$

3.4.2 มดสร้างคำตอบจากพารามิเตอร์ที่ตั้งไว้ กำหนดให้มดอยู่ที่เมือง i และกำลังที่จะเลือกว่าจะเดินไปเมืองใด และกำหนดให้เมืองที่มดเลือกไปแทนด้วย j

$$p(i,j) = \frac{[\tau(i,j)]^{\alpha}[\eta(i,j)]^{\beta}}{\sum_{j\epsilon J}[\tau(i,j)]^{\alpha}[\eta(i,j)]^{\beta}} \tag{3.15}$$

$$cumpop(i,j) = \sum_{k=1}^{j} p(i,k) \tag{3.16}$$

โดยที่

$\tau(i,j)$ คือ ค่าฟีโรโมนของเส้นทางจากเมือง i ไปเมือง j

$\eta(i,j) = \frac{1}{\delta(i,j)}$ คือ ส่วนกลับของระยะทาง

$\delta(i,j)$ คือ ระยะทางระหว่างเมือง i และเมือง j

J คือ เซตของเมืองที่ยังไม่ถูกเลือกเข้าไปอยู่ในเส้นทางใดๆ

$p(i,j)$ คือ ความน่าจะเป็นของการเลือกเมือง j เมื่อมดอยู่ที่เมือง i

$cumpop(i,j)$ คือ ความน่าจะเป็นสะสมของการเลือกเมือง j เมื่อมดอยู่ที่เมือง i

จากสมการที่ (3.15) เป็นสมการหาความน่าจะเป็นในการที่แต่ละเมืองมีโอกาสถูกเลือกเป็นเมืองที่จะเดินทางไป โดยเมืองที่มีค่าผลคูณระหว่างค่า ฟีโรโมนยกกำลังแอลฟากับค่าส่วนกลับระยะทางยกกำลังเบต้ามาก ความน่าจะเป็นที่มดจะเลือกไปเมืองนั้นจะมีค่าสูง เมืองใดที่ค่าผลคูณระหว่างฟีโรโมนกำลังแอลฟากับค่าส่วนกลับระยะทางยกกำลังเบต้าต่ำ ความน่าจะเป็นที่มดจะเลือกไปเมืองนั้นก็ต่ำด้วย แต่ทั้งนี้ทั้งนั้นผลรวมของความน่าจะเป็นทั้งหมดของเมืองที่มดยังไม่ได้ไปมีค่าเท่ากับ 1 เมื่อได้ความน่าจะเป็นแล้วจะใช้หลักการของวงกลมรูลเล็ตวีล (Roulette Wheel) มาใช้ในการสุ่มหาเมืองที่จะเดินทางต่อไป โดยหาความน่าจะเป็นสะสมดังสมการ (3.16) ความน่าจะเป็นสะสมได้จากผลรวมของความน่าจะเป็นของเมืองในลำดับแรกของลำดับของคู่แข่งขัน (Candidate

list: CL) จนกระทั่งถึงเมืองที่ j จากนั้นสุ่มตัวเลขมาหนึ่งตัวซึ่งมีค่าอยู่ระหว่าง {0,1} หากตัวเลขสุ่มตกอยู่ในช่วงใดจะเลือกเมืองนั้นเป็นลำดับถัดจากเมืองที่ i

ส่วนที่ทำให้ ACO ต่างจากวิธีอื่นๆ คือ การหาความน่าจะเป็น ในวิธีอื่นๆ สามารถหาได้ด้วยวิธีการที่ได้กล่าวมาแล้วข้างต้น แต่ใน ACO นั้นจะเพิ่มพารามิเตอร์อีก 1 ค่าคือ q_0 ซึ่ง $0 < q_0 < 1$ จากนั้นสุ่ม q อีกหนึ่งค่าที่อยู่ในช่วง $0 < q < 1$ เช่นกัน

ถ้า $q \leq q_0$ จะคำนวณความน่าจะเป็นจากสมการที่ 3.17 โดยที่ q คือตัวเลขสุ่มที่อยู่ในช่วง 0 ถึง 1

$$p(i,j) = \begin{cases} 1, & if\ j = argmax_{j \epsilon J}\{[\tau(i,j)]^{\alpha}[\eta(i,j)]^{\beta}\} \\ 0, & otherwise \end{cases} \tag{3.17}$$

จากสมการ (3.17) หมายถึง ความน่าจะเป็นที่จะเลือกเมือง j เพื่อเดินทางต่อจากเมือง i จะเท่ากับ 1 เมื่อเมือง j นั้น ให้ค่าผลคูณระหว่างค่าฟีโรโมนยกกำลังแอลฟ่ากับค่าส่วนกลับระยะทางยกกำลังเบต้ามากสุด ในบรรดาเมืองคู่แข่งขันทั้งหมด ในกรณีที่ $q \leq q_0$ และในกรณีอื่นๆ ให้ใช้วิธีการเดิมที่กล่าวมาแล้วข้างต้นคือสมการที่ (3.15) และ (3.16) ในการหาค่าน่าจะเป็น

มดแต่ละตัวจะเริ่มเดินทางออกจาก Depot จากนั้นจะเลือกลูกค้ารายถัดไปจากบัญชีผู้แข่งขันตามกฎในสมการ (3.15), (3.16) และ (3.17) จะเดินทางกลับ Depot เมื่อสินค้าหมด แล้วจึงออกเดินทางจาก Depot ใหม่จนกว่าจะเดินทางไปยังลูกค้าครบทุกรายจึงจะถือว่าได้คำตอบเริ่มต้นของมดหนึ่งตัว

3.4.3 การปรับค่าฟีโรโมน มดตัวที่สร้างเส้นทางที่ดีที่สุดเท่านั้นจึงจะสามารถปล่อยฟีโรโมนได้ เพื่อสร้างความเด่นชัดให้กับเส้นทางนั้น

$$\tau(i,j) = (1-\rho)\left(\tau(i,j)\right) \tag{3.18}$$

$$\tau(i,j) = (1-\rho)\left(\tau(i,j)\right) + g \tag{3.19}$$

โดยที่

ρ = อัตราการระเหยของค่าฟีโรโมนซึ่งมีค่า $(0 \leq \rho \leq 1)$

g = ส่วนกลับของระยะทางรวมของเส้นทางที่สั้นที่สุดในรอบนั้นๆ

ค่าฟีโรโมนจะถูกปรับตามสมการที่ (3.18) ถ้ามดที่สร้างเส้นทางที่ดีที่สุดไม่ได้เดินทางผ่านจุดเชื่อมนั้น และค่าฟีโรโมนจะถูกปรับตามสมการที่ (3.19) ถ้ามดตัวนั้นเดินทางผ่าน

จุดเชื่อม คำตอบที่ดีเพียงพอจากอัลกอริทึมนี้ เป็นคำตอบที่ได้จากการทำงานของกลไกทั้ง 3 ขั้นตอน และกระทำซ้ำตามเงื่อนไขที่กำหนดไว้

3.5 โครงสร้างของวิธีระบบอาณานิคมมดสำหรับการแก้ปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางสำหรับยานพาหนะ (Structure of Ant Colony Optimization for LRP)

วิธีอาณานิคมมดที่ใช้ในการแก้ปัญหาการจัดเส้นทางโดยทั่วไปแล้วเมื่อเปรียบเทียบกับธรรมชาติของมด จะถือว่ามีมดเพียงชนิดเดียวเท่านั้นคือ มดงานซึ่งทำหน้าที่ในการออกหาอาหารเพียงอย่างเดียวเท่านั้น โดยมดงานเริ่มเดินทางจากสถานที่ตั้งของรังที่มดอาศัยอยู่ เมื่อเปรียบเทียบกับปัญหาการจัดเส้นทางแล้ว อาจกล่าวได้ว่า รังของมดก็คือโรงงานที่รถเริ่มออกเดินทางไปส่งสินค้าซึ่งทราบตำแหน่งที่ตั้งของโรงงานแล้ว แต่วิธีอาณานิคมมดที่ใช้ในการแก้ปัญหาการเลือกสถานที่ตั้งพร้อมทั้งจัดเส้นทางการขนส่งด้วยแล้ว จะมีความพิเศษกว่าคือมีมด 2 ชนิด ซึ่งจะมีมดงานเหมือนกับปัญหาการจัดเส้นทาง แต่จะต่างกันที่มีราชินีมดเพิ่มเข้ามาเพราะว่าปัญหา LRP นั้น ยังไม่ทราบตำแหน่งที่ตั้งโรงงาน กล่าวคือยังไม่ทราบที่ตั้งของรังมดนั่นเอง ดังนั้นจึงมีราชินีมดเพิ่มเข้ามาเพื่อทำหน้าที่ในการเลือกสถานที่ตั้งของรังมด ซึ่งราชินีมดจะทำการเลือกสถานที่ตั้งรังที่อยู่ใกล้กับแหล่งอาหารมากที่สุด โดยอาศัยข้อมูลจากการเดินทางไปหาอาหารของมดงานเพื่อช่วยในการตัดสินใจเลือกสถานที่ตั้งของรัง ผู้วิจัยพัฒนาวิธีระบบอาณานิคมมดเพื่อแก้ปัญหาการเลือกสถานที่ตั้งและการจัดเส้นทางสำหรับยานพาหนะซึ่งแบ่งออกเป็น 6 ขั้นตอนดังนี้

3.5.1 การกำหนดค่าพารามิเตอร์ต่างๆ ที่เกี่ยวข้อง

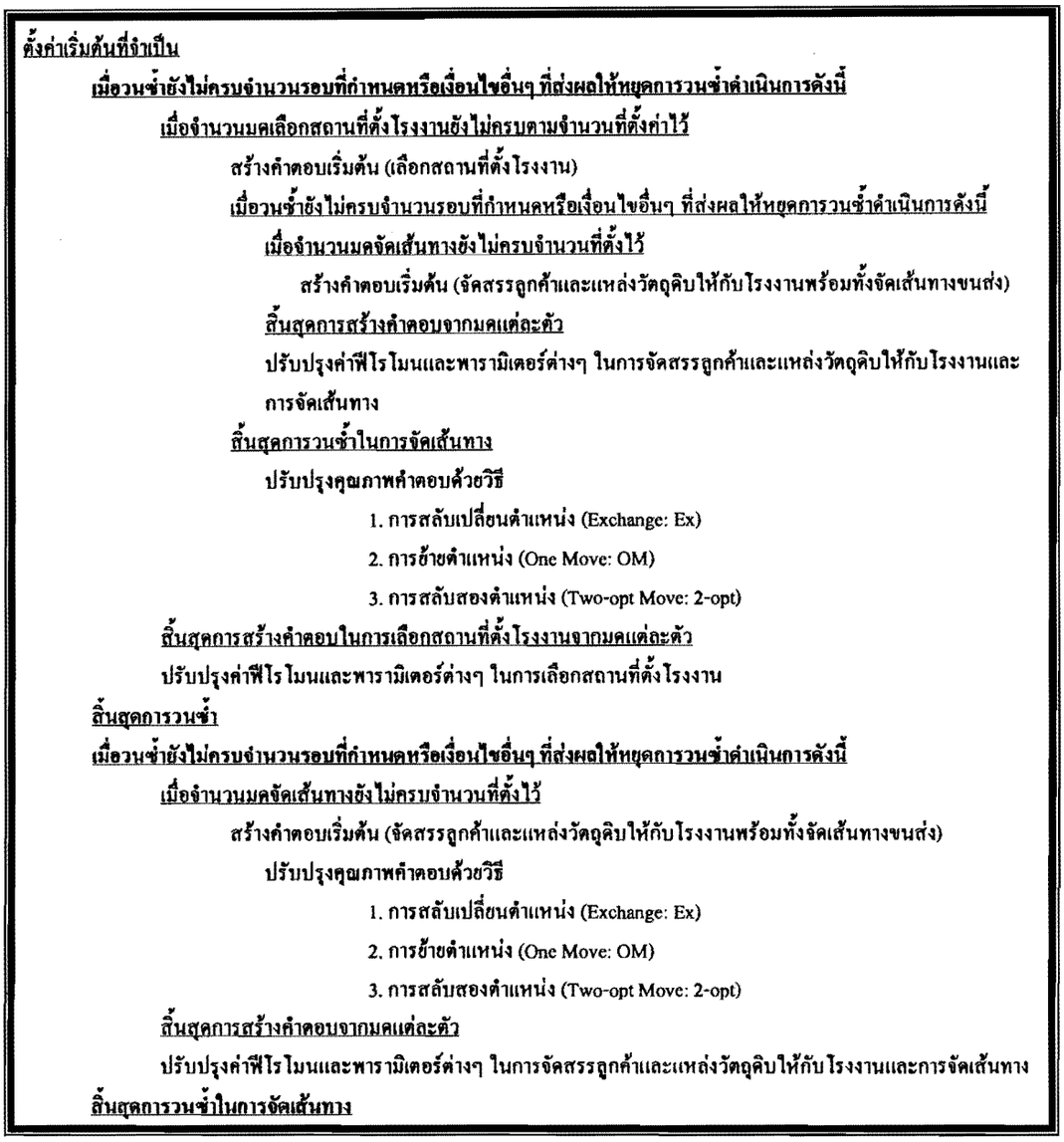
3.5.2 การสร้างผลเฉลยเริ่มต้นด้วยวิธีระบบอาณานิคมมดของปัญหาการเลือกสถานที่ตั้ง

3.5.3 การสร้างผลเฉลยเริ่มต้นด้วยวิธีระบบอาณานิคมมดของปัญหาการจัดเส้นทางสำหรับยานพาหนะ

3.5.4 การปรับปรุงค่าฟีโรโมนของปัญหาการจัดเส้นทางสำหรับยานพาหนะ

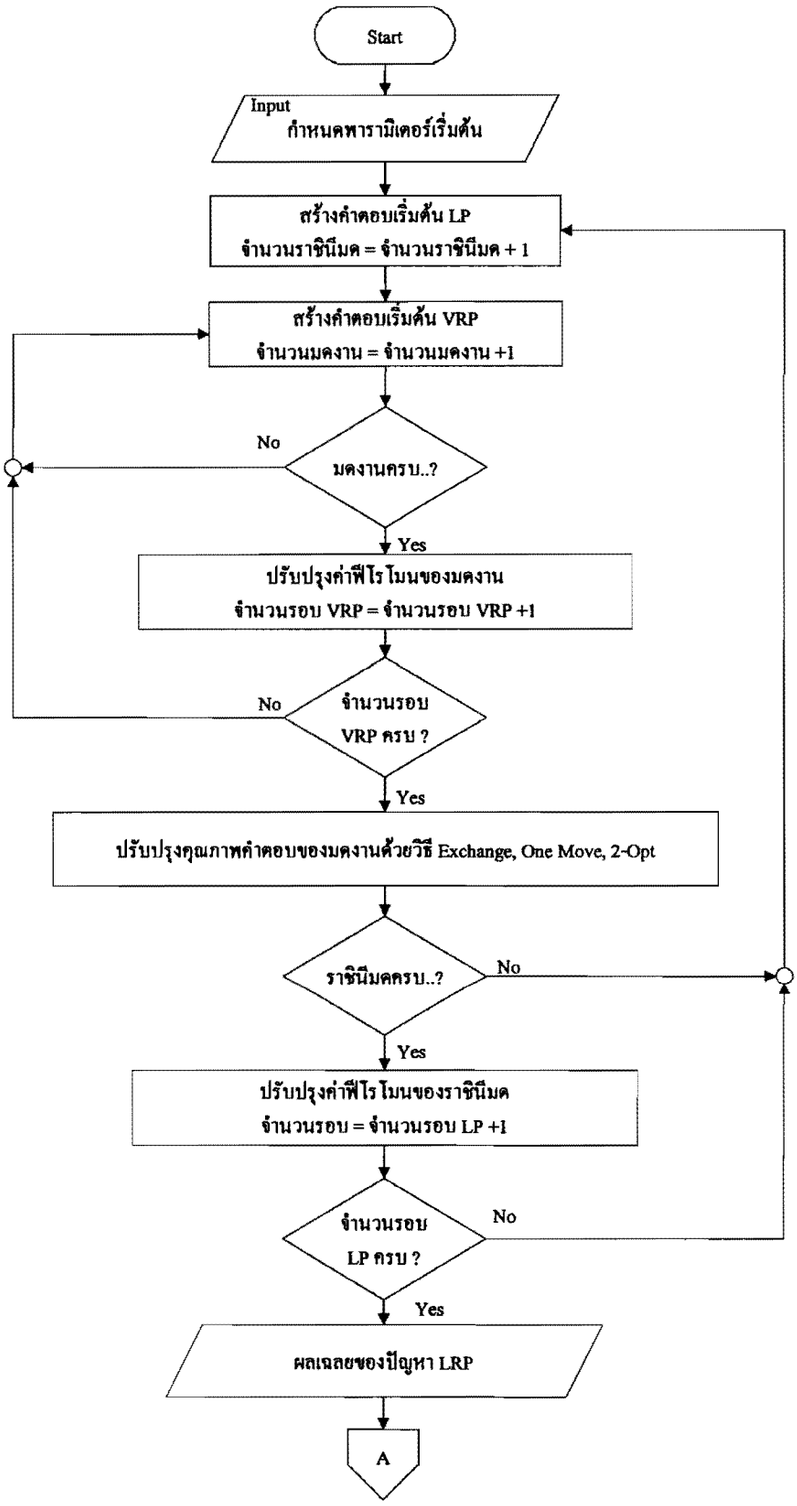
3.5.5 การปรับปรุงคุณภาพคำตอบของปัญหาการจัดเส้นทางสำหรับยานพาหนะ

3.5.6 การปรับปรุงค่าฟีโรโมนของปัญหาการเลือกสถานที่ตั้ง

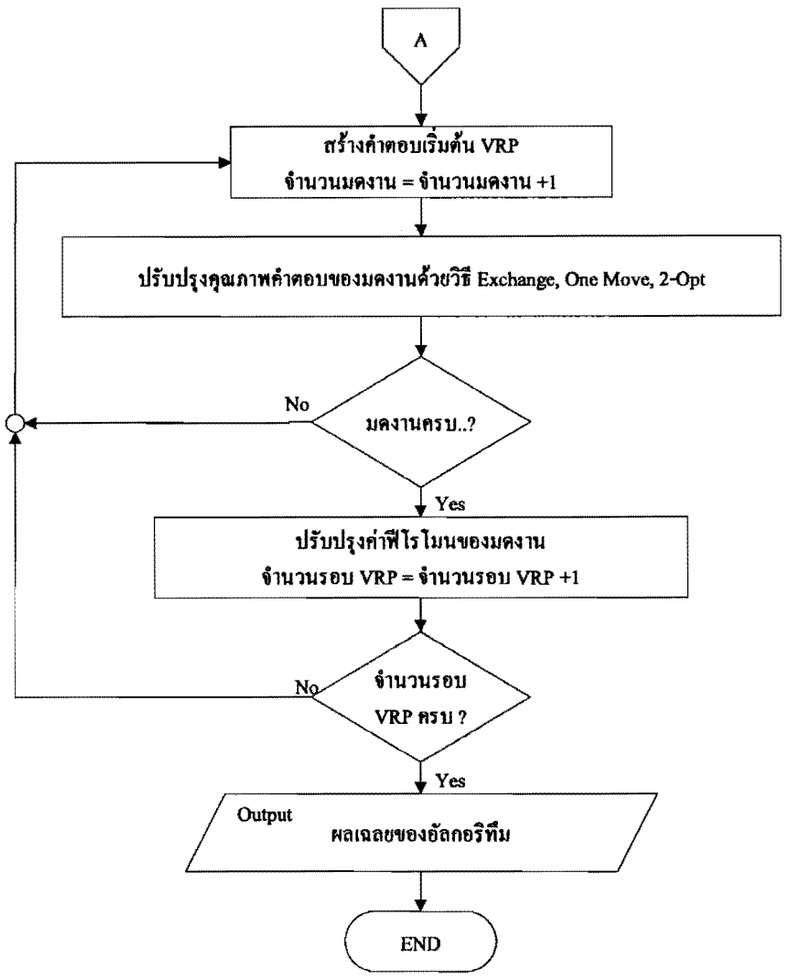


ภาพที่ 3.3 รหัสเทียมของวิธีอาณานิคมมดสำหรับปัญหา LRP

จากภาพที่ 3.3 จะเห็นว่ามี 2 ส่วนที่สำคัญ คือส่วนของการหาผลเฉลยของปัญหาการเลือกสถานที่ตั้งและส่วนของการหาผลเฉลยของปัญหาการจัดเส้นทางการขนส่ง โดยส่วนของการเลือกสถานที่ตั้งจะเป็นกรอบใหญ่ที่คลุมในส่วนของการจัดเส้นทางไว้ กล่าวคือ การจัดเส้นทางจะเป็นส่วนหนึ่งของการเลือกสถานที่ตั้ง เนื่องจากผลเฉลยของการจัดเส้นทางจะเป็นคำตอบของมดที่หาผลเฉลยของการเลือกสถานที่ตั้ง หลังจากทราบตำแหน่งที่ตั้งโรงงานจากผลเฉลยของอัลกอริทึมแล้ว จะทำการปรับปรุงคุณภาพคำตอบของปัญหาการจัดเส้นทางสำหรับยานพาหนะอีกครั้ง เพื่อปรับปรุงคำตอบให้มีประสิทธิภาพ



ภาพที่ 3.4 Flow Chart ของวิธีอาณานิคมมดสำหรับปัญหา LRP



ภาพที่ 3.4 Flow Chart ของวิธีอาณานิคมมดสำหรับปัญหา LRP (ต่อ)

จากภาพที่ 3.4 ขั้นตอนการทำงานของอัลกอริทึมเริ่มจาก การกำหนดพารามิเตอร์เริ่มต้น ขั้นตอนต่อมาคือ การสร้างคำตอบเริ่มต้นของปัญหาการเลือกสถานที่ตั้ง โดยราชินีมดจะทำการเลือกตำแหน่ง จากแหล่งอาหารหนึ่งแห่งจากแหล่งอาหารทั้งหมดเปรียบคือ จะทำการเลือกแหล่งวัตถุดิบหนึ่งแห่งจากแหล่งวัตถุดิบทั้งหมดเพื่อสร้างเป็นโรงงาน เมื่อได้ตำแหน่งที่ตั้งรังมดแล้วก็เป็นหน้าที่ของมดงานที่จะทำการจัดเส้นทางการขนส่งนั่นก็คือ ขั้นตอนการสร้างคำตอบเริ่มต้นของปัญหาการจัดเส้นทางการขนส่ง มดงานตัวที่สร้างคำตอบที่ดีที่สุดจะถูกบันทึกไว้และจะถูกปรับปรุงค่าหากมีมดงานตัวอื่นที่สร้างคำตอบที่ดีกว่า มดงานจะถูกสร้างขึ้นเท่ากับจำนวนมดงานที่กำหนดไว้ตั้งแต่เริ่มต้น เมื่อมดงานถูกสร้างครบตามจำนวนแล้วขั้นตอนต่อมาคือ การปรับปรุงค่าฟีโรโมนของมดงาน ซึ่งมดงานตัวที่สร้างเส้นทางที่ดีที่สุดเท่านั้นที่จะเป็นตัวปล่อยฟีโรโมน จากขั้นตอนการสร้าง

คำตอบเริ่มต้นของมดงาน จนกระทั่งเสร็จสิ้นการปรับปรุงค่าฟีโรโมนของมดงานคือหนึ่งรอบการทำงานของกระบวนการแก้ปัญหาการจัดเส้นทางการขนส่ง จากนั้นจะกระทำซ้ำจนครบจำนวนรอบการแก้ปัญหาการขนส่งที่กำหนดไว้ หลังจากได้เส้นทางจากขั้นตอนการแก้ปัญหาการจัดเส้นทางการขนส่งแล้วผลเฉลยดังกล่าวจะถูกปรับปรุงคุณภาพคำตอบด้วยวิธี Exchange, One-Move และ 2-Opt ผลที่ได้จากการปรับปรุงคุณภาพคำตอบคือ เส้นทางการขนส่งของรังมดที่ถูกเลือกโดยราชินีมด จากขั้นตอนการสร้างคำตอบเริ่มต้นของปัญหาการเลือกสถานที่ตั้งจนถึงตรงนี้จะได้ราชินีมดหนึ่งตัว ราชินีมดที่มีเส้นทางการขนส่งที่สั้นที่สุดจะถูกเก็บค่าไว้และจะถูกปรับปรุงค่า หากมีราชินีมดตัวที่ให้เส้นทางที่สั้นกว่า ราชินีมดจะถูกสร้างขึ้นเท่ากับจำนวนที่กำหนดไว้ตั้งแต่เริ่มต้น เมื่อราชินีมดครบตามจำนวนแล้วจึงจะทำการปรับปรุงค่าฟีโรโมนของราชินีมด โดยราชินีมดที่สร้างคำตอบที่ดี ในงานวิจัยนี้ใช้ 16^{*} อันดับแรกของราชินีมดทั้งหมดให้ทำหน้าที่ปล่อยฟีโรโมน จากขั้นตอนการสร้างคำตอบเริ่มต้นของปัญหาการเลือกสถานที่ตั้งจนถึงการปรับปรุงค่าฟีโรโมนของราชินีมดคือ หนึ่งรอบการทำงานของกระบวนการแก้ปัญหาการเลือกสถานที่ตั้ง จากนั้นจะกระทำซ้ำขั้นตอนดังกล่าวจนกว่าจะครบจำนวนรอบที่กำหนดไว้

หลังจากเสร็จสิ้นกระบวนการแก้ปัญหาการเลือกสถานที่ตั้งจะได้ตำแหน่งที่ตั้งโรงงาน จากนั้นจะทำการจัดเส้นทางการขนส่งซ้ำอีกครั้ง แต่จะเพิ่มจำนวนมดงานและจำนวนรอบการกระทำซ้ำ โดยขั้นตอนเริ่มจากสร้างคำตอบเริ่มต้นของมดงานซึ่งมดงานทุกตัวจะผ่านการปรับปรุงคุณภาพคำตอบด้วยวิธี Exchange, One-Move และ 2-Opt มดงานตัวที่ผ่านการปรับปรุงคุณภาพคำตอบแล้วให้ค่าระยะทางน้อยที่สุดจะถูกบันทึกไว้ เมื่อสร้างมดงานครบตามจำนวนที่กำหนดแล้วทำการปรับปรุงค่าฟีโรโมนของมดงาน โดยมดงานตัวที่ให้ค่าระยะทางน้อยที่สุดเท่านั้นจะเป็นตัวปล่อยฟีโรโมนจบขั้นตอนนี้คือหนึ่งรอบการแก้ปัญหาของมดงานกระทำซ้ำจนครบตามจำนวนรอบที่กำหนด จึงจะได้ผลเฉลยของอัลกอริทึม

* หมายเหตุ ที่มาของการเลือกราชินีมดที่สร้างคำตอบที่ดีที่สุด 16 อันดับแรกเพื่อปล่อยฟีโรโมนคือ เมื่อมองปัญหา TSP จะมีค่าฟีโรโมนเท่ากับจำนวนเมืองคูณด้วยจำนวนเมือง และหากมดสามารถปล่อยฟีโรโมนได้เพียงหนึ่งตัวจุดที่จะถูกเพิ่มค่าฟีโรโมนจะเท่ากับจำนวนเมือง ซึ่งจำนวนเมืองมีค่าดังสมการ (3.20) เทียบกับปัญหา LP จะมีค่าฟีโรโมนเท่ากับจำนวนเมืองดังนั้นจุดที่ถูกเพิ่มค่าฟีโรโมนของปัญหา LP มีค่าดังสมการ (3.21) ในงานวิจัยนี้มีแหล่งวัตถุดิบ 261 แหล่งจะได้ว่า $\sqrt{261} \approx 16$

$$\text{จำนวนเมือง} = \sqrt{\text{จำนวนฟีโรโมนทั้งหมด}} \tag{3.20}$$

$$\text{จุดที่ถูกเพิ่มค่าฟีโรโมนของปัญหา } LP = \sqrt{\text{จำนวนเมือง}} \tag{3.21}$$

3.5.1 การกำหนดค่าพารามิเตอร์ต่างๆ ที่เกี่ยวข้อง

พารามิเตอร์ที่ต้องตั้งค่าไว้ก่อนการเริ่มกระบวนการมีดังต่อไปนี้

3.5.1.1 จำนวนรอบที่ต้องการวนซ้ำ (Iteration) ซึ่งแบ่งเป็น 2 ส่วนคือ จำนวนรอบในการวนซ้ำเพื่อหาคำตอบของปัญหาการเลือกสถานที่ตั้งและจำนวนรอบในการวนซ้ำเพื่อหาคำตอบของปัญหาการจัดเส้นทาง

3.5.1.2 จำนวนประชากรมด (Ant Population) โดยจะมีมดอยู่ 2 ชนิดด้วยกัน ได้แก่ ราชินีมดจะเป็นมดที่ใช้ในการหาผลเฉลยของปัญหาการเลือกสถานที่ตั้ง และมดงานจะเป็นมดที่ใช้ในการหาผลเฉลยในการจัดเส้นทางการขนส่ง

3.5.1.3 ค่าฟีโรโมน (Pheromone) แบ่งเป็น 2 ส่วนได้แก่ $L\tau_i$ คือค่าฟีโรโมนของแหล่งวัตถุดิบ i ในการเลือกตำแหน่งสถานที่ตั้ง และ $R\tau_{ij}$ คือค่าฟีโรโมนในการเดินทางจากแหล่งวัตถุดิบ i ไปยังลูกค้า j โดยค่าฟีโรโมนนี้ทำหน้าที่เหมือนกับฟีโรโมนของมด และมีหน้าที่เหมือนกัน คือทำหน้าที่เพิ่มความน่าสนใจของตำแหน่งที่ตั้งที่ราชินีมดตัวต่อไปจะเลือกทำรัง และฟีโรโมนในการเดินทางจะเพิ่มความน่าสนใจของเส้นทางที่มดงานตัวต่อไปจะเลือกเดินทาง

3.5.1.4 อัตราการระเหยของค่าฟีโรโมน (Evaporation Rate) ก็แบ่งเป็น 2 ส่วนด้วยกันคือ $L\rho$ อัตราการระเหยของฟีโรโมนที่ใช้ในการเลือกตำแหน่งที่ตั้ง และ $R\rho$ อัตราการระเหยของฟีโรโมนที่ใช้ในการจัดเส้นทางสำหรับยานพาหนะ โดยที่ $0 < L\rho < 1$ และ $0 < R\rho < 1$

3.5.1.5 ค่าพารามิเตอร์ได้แก่ $L\alpha$ เป็นค่าการให้น้ำหนักของฟีโรโมนในการเลือกสถานที่ตั้ง, $R\alpha$ เป็นค่าการใช้น้ำหนักของฟีโรโมนในการจัดเส้นทาง, $L\beta$ เป็นค่าการให้น้ำหนักของส่วนกลับระยะทางรวมของแต่ละจุดที่จะเลือกเป็นสถานที่ตั้งโรงงาน และ $R\beta$ เป็นการให้น้ำหนักของส่วนกลับระยะทางในการจัดเส้นทางการขนส่งโดยที่ $L\alpha, R\alpha, L\beta, R\beta > 0$

3.5.2 การสร้างผลเฉลยเริ่มต้นด้วยวิธีระบบอาณานิคมมดของปัญหาการเลือกสถานที่ตั้ง

ราชินีมดกำลังเลือกว่าจะสร้างรังที่ตำแหน่งใด กำหนดให้ตำแหน่งที่ราชินีมดจะเลือกเพื่อสร้างรังแทนด้วย i

$$Lp(i) = \frac{[L\tau(i)]^{L\alpha}[L\eta(i)]^{L\beta}}{\sum_{i=1}^{I}[L\tau(i)]^{L\alpha}[L\eta(i)]^{L\beta}} \tag{3.22}$$

$$Lcumpop(i) = \sum_{j=1}^{i} Lp(j) \tag{3.23}$$

โดยที่

$L\eta(i) = \frac{1}{\sum_{j=1}^{M+N} D_{ij}}$ คือ ส่วนกลับของระยะทางรวมของทุกจุดมายังจุด i

D_{ij} คือ ระยะทางจากแหล่งวัตถุดิบ i ไปยังแหล่งวัตถุดิบและลูกค้า j

I คือ เซตของแหล่งวัตถุดิบ

$Lp(i)$ คือ ความน่าจะเป็นในการเลือกแหล่งวัตถุดิบ i เปิดเป็นโรงงาน

$Lcumpop(i)$ คือ ความน่าจะเป็นสะสมของการเลือกแหล่งวัตถุดิบ i เปิดเป็นโรงงาน

จากสมการที่ (3.22) เป็นสมการหาความน่าจะเป็นในการที่แหล่งวัตถุดิบแต่ละแห่งจะถูกเลือกเพื่อเปิดเป็นโรงงาน ซึ่งผลรวมของความน่าจะเป็นในการเลือกสถานที่ตั้งรวมทุกจุดแล้วเท่ากับ 1 เมื่อได้ความน่าจะเป็นแล้วจะใช้หลักการของวงกลมรูลเล็ตวีล (Roulette Wheel) มาใช้ในการสุ่มหาแหล่งวัตถุดิบที่จะใช้เปิดเป็นโรงงาน โดยหาความน่าจะเป็นสะสมดังสมการที่ (3.23)

ความน่าจะเป็นสะสมของเมือง i หาได้จากผลรวมของความน่าจะเป็นของเมืองในลำดับแรกของคู่แข่งขันจนกระทั่งถึงเมือง i จากนั้นสุ่มตัวเลขมาหนึ่งตัวอยู่ระหว่าง 0-1 หากตัวเลขสุ่มตกอยู่ในช่วงใดจะเลือกเมืองนั้นเปิดเป็นโรงงาน

3.5.3 การสร้างผลเฉลยเริ่มต้นด้วยวิธีระบบอาณานิคมมดของปัญหาการจัดเส้นทางสำหรับยานพาหนะ

หลังจากที่ได้ตำแหน่งที่ตั้งของรังมดจากข้อ 3.5.2 แล้วก็ถึงหน้าที่ของมดงานที่จะต้องออกหาอาหาร โดยสร้างคำตอบจากพารามิเตอร์ที่ตั้งไว้ กำหนดให้มดงานอยู่ที่เมือง i และกำลังจะเลือกว่าจะเดินไปเมืองใด และกำหนดให้เมืองที่มดงานเลือกไป แทนด้วย j

$$Rp(i,j) = \frac{[R\tau(i,j)]^{R\alpha}[R\eta(i,j)]^{R\beta}}{\sum_{j=1}^{J}[R\tau(i,j)]^{R\alpha}[R\eta(i,j)]^{R\beta}} \tag{3.24}$$

$$Rcumpop(i,j) = \sum_{k=1}^{j} Rp(i,j) \tag{3.25}$$

โดยที่

$R\eta(i,j) = \frac{1}{D(i,j)}$ คือ ส่วนกลับของระยะทาง

$D(i,j)$ คือ ระยะทางระหว่างเมือง i ไปเมือง j

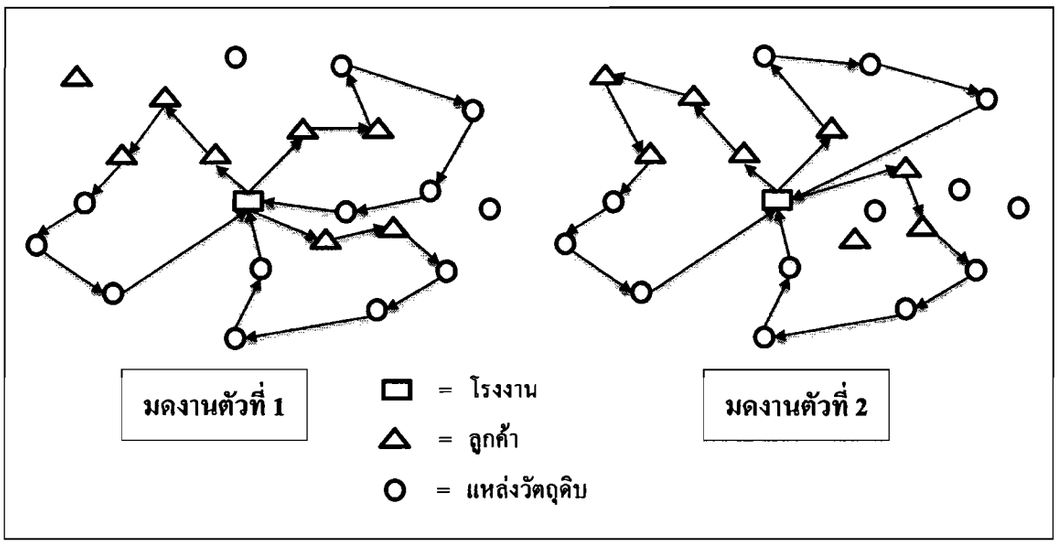
J คือ เซตของเมืองที่ยังไม่ถูกเลือกเข้าไปปอยู่ในเส้นทางใดๆ

$Rp(i,j)$ คือ ความน่าจะเป็นในการเลือกเมือง j เมื่อมดงานอยู่ที่เมือง i

$Rcumpop(i,j)$ คือ ความน่าจะเป็นสะสมของการเลือกเมือง j เมื่อมดงานอยู่ที่เมือง i

ในการที่มดงานจะเลือกไปเมืองใดนั้นจะใช้หลักการคล้ายคลึงกับการเลือกตำแหน่งที่ตั้งโรงงงาน แต่จะใช้สมการที่ (3.24) และ (3.25) ในการหาความน่าจะเป็นและความน่าจะเป็นสะสมแทน

ในงานวิจัยนี้รถขนส่งแต่ละคันจะเดินทางไปส่งสินค้าให้กับลูกค้าจนสินค้าที่บรรทุกไปในรถนั้นหมดก่อนจึงจะเดินทางไปรับวัตถุดิบจากแหล่งวัตถุดิบ เมื่อวัตถุดิบเต็มรถบรรทุกแล้วจึงจะกลับมายังโรงงาน และเริ่มเดินทางออกจากโรงงานใหม่อีกครั้ง จนกว่าจะเกินกำลังการผลิตของโรงงานแล้วจึงจะหยุดการเดินรถ การที่เดินทางไปส่งสินค้าก่อนแล้วจึงเดินทางไปรับวัตถุดิบนั้นเป็นการลดการเดินทางกลับโรงงานด้วยรถเที่ยวเปล่า จึงเป็นการช่วยลดค่าใช้จ่ายหรือลดระยะทางอีกวิธีหนึ่ง การสร้างคำตอบเริ่มต้นของมดงานแสดงดังภาพที่ 3.5



ภาพที่ 3.5 การสร้างคำตอบเริ่มต้นของมดงานในปัญหาการจัดเส้นทางสำหรับยานพาหนะ

3.5.4 การปรับค่าฟีโรโมนสำหรับปัญหาการจัดเส้นทางการขนส่ง

ในแต่ละรอบการกระทำซ้ำของปัญหาการจัดเส้นทางการขนส่ง จะมีการปรับค่าฟีโรโมนในทุกรอบของการกระทำซ้ำ โดยในแต่ละรอบมดงานตัวที่สร้างเส้นทางที่ดีที่สุดเท่านั้นจึงจะสามารถปล่อยฟีโรโมนได้ เพื่อสร้างความเด่นชัดให้กับเส้นทางนั้น

$$R\tau(i,j) = (1 - R\rho)\left(R\tau(i,j)\right) \tag{3.26}$$

$$R\tau(i,j) = (1 - R\rho)\left(R\tau(i,j)\right) + Rg \tag{3.27}$$

โดยที่

$R\tau(i,j)$ = ฟีโรโมนในการเดินทางจากเมือง i ไปยังเมือง j

$R\rho$ = อัตราการระเหยของค่าฟีโรโมนที่ใช้ในการจัดเส้นทางสำหรับยานพาหนะ

Rg = ส่วนกลับของระยะทางรวมของเส้นทางที่สั้นที่สุดในรอบนั้นๆ ของมดงาน

ในทุกรอบของการกระทำซ้ำฟีโรโมนของทุกๆ จุดเชื่อมจะมีการระเหยหรือเจือจางลงด้วยอัตรา $R\rho$ และมดงานตัวที่สร้างเส้นทางที่ดีที่สุดในแต่ละรอบจะเป็นตัวเพิ่มฟีโรโมนให้กับเส้นทางที่สั้นที่สุดนั้น กล่าวคือค่าฟีโรโมนจะถูกปรับตามสมการที่ (3.26) หากมดงานตัวที่สร้างเส้นทางที่ดีที่สุดไม่ได้เดินทางผ่านจุดเชื่อมนั้น ฟีโรโมนก็จะระเหยไปอย่างเดียว แต่ถ้ามดงานตัวที่สร้างเส้นทางที่ดีที่สุดเดินทางผ่านจุดเชื่อมนั้นค่าฟีโรโมนจะถูกปรับตามสมการ (3.27) คือมีการระเหยและก็มีการเพิ่มฟีโรโมน โดยการบวกค่าส่วนกลับระยะทางรวมของเส้นทางที่สั้นที่สุดในรอบนั้นๆ เข้าไปด้วย

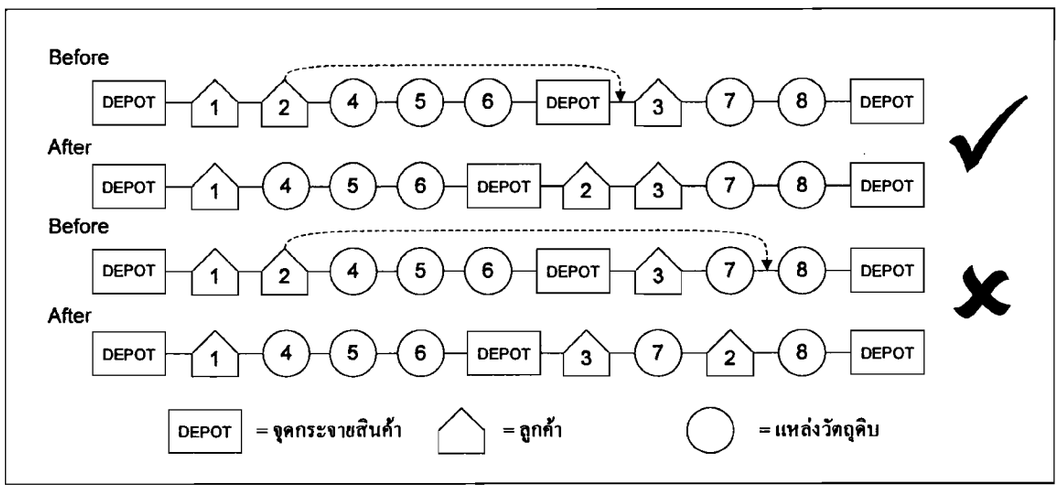
3.5.5 การปรับปรุงคุณภาพคำตอบของปัญหาการจัดเส้นทาง

ผลเฉลยที่ได้จากการแก้ปัญหาการจัดเส้นทางสำหรับยานพาหนะเท่านั้นที่จะผ่านการปรับปรุงคุณภาพคำตอบ ซึ่งในงานวิจัยนี้ผู้วิจัยเลือกใช้การปรับปรุงคำตอบเฉพาะที่ (Local Search) ทั้งสิ้น 3 วิธีด้วยกัน ได้แก่

3.5.5.1 การย้ายตำแหน่ง (One Move: OM)

การย้ายตำแหน่งแบ่งเป็น 2 ส่วน คือการย้ายลูกค้าและการย้ายแหล่งวัตถุดิบ ซึ่งการย้ายในสองส่วนนี้จะทำการย้ายในลักษณะที่เหมือนกันคือ เป็นการย้ายลูกค้า/แหล่งวัตถุดิบหนึ่งราย จากเส้นทางหนึ่งไปยังอีกเส้นทางหนึ่ง โดยไม่มีการย้ายแบบสลับในเส้นทางเดิมที่ลูกค้า/แหล่งวัตถุดิบรายนั้นอยู่ และการย้ายจะต้องไม่ขัดแย้งกับเงื่อนไข กล่าวคือ เมื่อทำการย้ายแล้วเส้นทางใหม่ที่เกิดขึ้นจะต้องเดินทางไปส่งของให้ลูกค้าก่อนแล้วจึงเดินทางไปรับวัตถุดิบตามเดิมจะต้องไม่มีการไปรับไปส่งสลับไปมาระหว่างลูกค้าและแหล่งวัตถุดิบ และเส้นทางใหม่ที่เกิดขึ้นนั้นจะต้องมีความต้องการสินค้า/วัตถุดิบที่ต้องรับขึ้นรถไม่เกินความจุของรถบรรทุกแสดงดังภาพที่ 3.6

ในงานวิจัยนี้จะเริ่มจากการย้ายลูกค้า ลูกค้าจะถูกเลือกและย้ายไปแทรกในเส้นทางอื่นที่ไม่ใช่เส้นทางเดิมของลูกค้ารายนั้นอยู่ ในทุกๆ ตำแหน่งที่เป็นไปได้ แล้วบันทึกระยะทางรวมที่เกิดขึ้นไว้เพื่อเลือกตำแหน่งที่ย้ายแล้วดีที่สุด(ระยะทางรวมน้อยสุด) จากนั้นทำการย้ายแหล่งวัตถุดิบ เช่นเดียวกันกับการย้ายลูกค้า การย้ายลูกค้าและแหล่งวัตถุดิบจะถูกเปรียบเทียบกันแล้วเลือกย้ายตำแหน่งที่ย้ายแล้วดีที่สุด จากนั้นทำการซ้ำการย้าย ทุกกรณีที่เป็นไปได้จนกว่าจะไม่สามารถลดระยะทางรวมได้จึงหยุดการทำซ้ำ

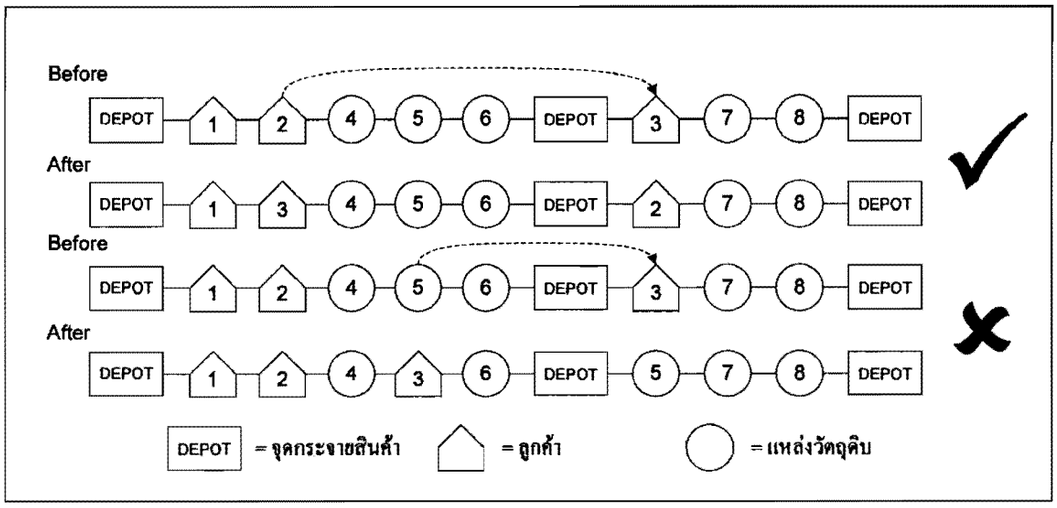


ภาพที่ 3.6 การปรับปรุงคุณภาพคำตอบด้วยวิธีการย้ายตำแหน่ง

3.5.5.2 การสลับเปลี่ยนตำแหน่ง (Exchange: Ex)

การสลับเปลี่ยนตำแหน่งแบ่งออกเป็น 2 ส่วนคือ การสลับลูกค้าและการสลับแหล่งวัตถุดิบ การสลับเปลี่ยนลูกค้าเป็นการสลับระหว่างลูกค้ารายหนึ่งจากเส้นทางหนึ่งกับลูกค้าอีกรายหนึ่งในอีกเส้นทางหนึ่ง และเช่นเดียวกันการสลับเปลี่ยนแหล่งวัตถุดิบก็เป็นการสลับระหว่างแหล่งวัตถุดิบรายหนึ่งจากเส้นทางหนึ่งกับแหล่งวัตถุดิบอีกรายหนึ่งในอีกเส้นทางหนึ่ง โดยจะไม่มีการสลับระหว่างลูกค้ากับแหล่งวัตถุดิบ การสลับจะต้องไม่ขัดแย้งกับเงื่อนไขแสดงดังภาพที่ 3.7

ในงานวิจัยนี้จะเริ่มจากทำการสลับเปลี่ยนตำแหน่งลูกค้าในทุกๆ ตำแหน่งที่สามารถสลับได้ แล้วบันทึกระยะทางรวมที่เกิดขึ้น เพื่อเลือกสลับในตำแหน่งที่สลับแล้วดีที่สุด จากนั้นทำการสลับเปลี่ยนตำแหน่งแหล่งวัตถุดิบในทุกๆ ตำแหน่งที่สามารถสลับได้ แล้วบันทึกระยะทางรวมที่เกิดขึ้น เพื่อเลือกสลับในตำแหน่งที่สลับแล้วดีที่สุด ตำแหน่งที่ดีที่สุดของการย้ายลูกค้าและการย้ายแหล่งวัตถุดิบจะถูกเปรียบเทียบกันแล้วเลือกสลับในตำแหน่งที่สลับแล้วดีที่สุด จากนั้นทำซ้ำการสลับเปลี่ยนตำแหน่ง การสลับเปลี่ยนตำแหน่งจะหยุดเมื่อทุกๆ ตำแหน่งที่สามารถสลับเปลี่ยนได้ส่งผลให้ระยะทางรวมเพิ่มขึ้นหรือเท่าเดิม



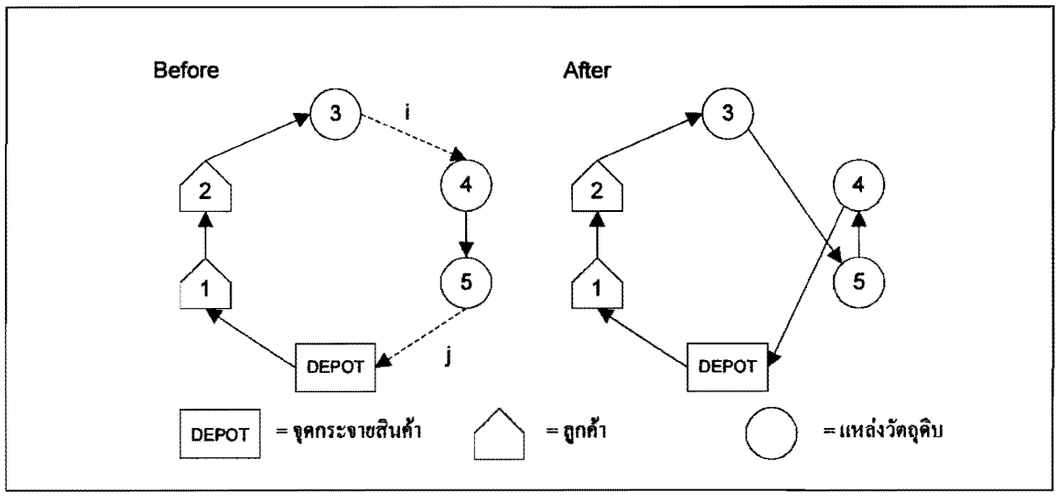
ภาพที่ 3.7 การปรับปรุงคุณภาพคำตอบด้วยวิธีการสลับเปลี่ยนตำแหน่ง

3.5.5.3 การสลับสองตำแหน่ง (Two-opt Move: 2-Opt)

วิธี 2-Opt คล้ายกับวิธี One move และ Exchange คือ วิธี 2-Opt แบ่งออกเป็น 2 ส่วน และวิธี 2-Opt จะต้องไม่ทำให้เส้นทางใหม่ที่เกิดขึ้นมีการไปรับไปส่งสลับไปมาระหว่างลูกค้าและแหล่งวัตถุดิบ แต่ต่างกันที่การย้ายแบบ 2-Opt เป็นการสลับลูกค้า/แหล่งวัตถุดิบภายในเส้นทางเดิมที่ลูกค้า/แหล่งวัตถุดิบรายนั้นอยู่เท่านั้น วิธี 2-Opt เริ่มจากการเลือกเส้นเชื่อม 2 เส้นที่ไม่อยู่ติดกัน จากนั้นทำการสลับเส้นทางการเชื่อมทั้ง 2 เส้นทางนั้น โดยลูกค้า/แหล่งวัตถุดิบที่อยู่จุดเริ่มต้นของเส้นเชื่อมแรกจะสลับไปเชื่อมกับลูกค้า/แหล่งวัตถุดิบที่อยู่จุดเริ่มต้นของเส้นเชื่อมที่สอง ซึ่งจะทำให้ลำดับของลูกค้า/แหล่งวัตถุดิบสองจุดที่ถูกเลือกนั้นถูกเรียงกลับในทิศตรงกันข้าม และขั้นตอนสุดท้ายลูกค้า/แหล่งวัตถุดิบที่อยู่จุดปลายของเส้นเชื่อมแรกจะถูกเชื่อมกับลูกค้า/แหล่งวัตถุดิบที่อยู่จุดปลายของเส้นเชื่อมที่สองแสดงดังภาพที่ 3.8

ในงานวิจัยนี้จะเริ่มจากการใช้วิธี 2-Opt กับลูกค้าก่อน โดยจะสามารถใช้วิธี 2-Opt กับลูกค้าได้ก็ต่อเมื่อเส้นทางนั้นมีลูกค้าอย่างน้อยสามราย หรือมีลูกค้าอย่างน้อยสองรายแต่มีแหล่งวัตถุดิบอย่างน้อยหนึ่งแห่งในเส้นทางนั้น การทำ 2-Opt จะทำในทุกๆ ตำแหน่งที่เป็นไปได้แล้วบันทึกระยะทางที่เกิดขึ้นไว้ เลือกสลับเส้นทางเชื่อมในตำแหน่งที่สลับแล้วดีที่สุด จากนั้นใช้วิธี 2-Opt กับแหล่งวัตถุดิบ เช่นเดียวกันกับกรณีลูกค้า คือ จะพิจารณาทุกๆ ตำแหน่งที่เป็นไปได้แล้วบันทึกระยะทางที่เกิดขึ้นไว้ แล้วเลือกสลับเส้นทางเชื่อมในตำแหน่งที่สลับแล้วดีที่สุดและจะสามารถใช้วิธี 2-Opt กับแหล่งวัตถุดิบได้ก็ต่อเมื่อเส้นทางนั้นมีแหล่งวัตถุดิบอย่างน้อยสามแห่ง หรือมีแหล่งวัตถุดิบอย่างน้องสองแห่งแต่มีลูกค้าอย่างน้อยหนึ่งรายในเส้นทางนั้น การสลับลูกค้า

ตำแหน่งที่ดีที่สุดจะถูกเปรียบเทียบกับตำแหน่งการสลับแหล่งวัตถุดิบที่ดีที่สุดแล้วเลือกสลับเส้นเชื่อมในตำแหน่งที่สลับแล้วดีที่สุด หลังจากนั้นทำซ้ำวิธี 2-Opt ทุกกรณีที่เป็นไปได้จนกว่าจะไม่สามารถลดระยะทางรวมได้



ภาพที่ 3.8 การปรับปรุงคุณภาพคำตอบด้วยวิธีการสลับสองตำแหน่ง

ตัวอย่างการใช้วิธี 2-opt กับแหล่งวัตถุดิบแสดงดังภาพที่ 3.8 เส้นทางที่ถูกเลือกเส้นแรก (*i*) คือเส้นเชื่อมระหว่างหมายเลข 3-4 และเส้นเชื่อมที่สอง (*j*) คือเส้นเชื่อมระหว่างหมายเลข 5-Depot จากนั้นทำการสลับเส้นทางการเชื่อมทั้ง 2 เส้นทางนั้น โดยแหล่งวัตถุดิบที่อยู่จุดเริ่มต้นของเส้นเชื่อมแรก (หมายเลข 3) จะสลับไปเชื่อมกับแหล่งวัตถุดิบที่อยู่จุดเริ่มต้นของเส้นเชื่อมที่สอง (หมายเลข 5) ซึ่งจะทำให้ลำดับของแหล่งวัตถุดิบสองจุดที่ถูกเลือกนั้นถูกเรียงกลับในทิศตรงกันข้าม กล่าวคือ จากเดิมเดินทางจากหมายเลข 4 ไปหมายเลข 5 สลับเป็นเดินทางจากหมายเลข 5 ไปหมายเลข 4 และขั้นตอนสุดท้ายแหล่งวัตถุดิบที่อยู่จุดปลายของเส้นเชื่อมแรก (หมายเลข 4) จะถูกเชื่อมกับโหนดที่อยู่จุดปลายของเส้นเชื่อมที่สอง (Depot) จึงได้เส้นทางจากเดิม Depot-1-2-3-4-5-Depot เป็น Depot-1-2-3-5-4- Depot

3.5.6 การปรับค่าฟีโรโมนสำหรับปัญหาการเลือกสถานที่ตั้ง

ในแต่ละรอบการกระทำซ้ำของปัญหาการเลือกสถานที่ตั้ง ค่าระยะทางรวมจากการหาผลเฉลยของปัญหาการจัดเส้นทางการขนส่งคือค่าฟังก์ชันสำหรับประเมินค่าความเหมาะสม (Fitness function) ในการค้นหาคำตอบของราชินีมดแต่ละตัว

$$L\tau(i) = (1 - L\rho)\left(L\tau(i)\right) \quad (3.28)$$

$$L\tau(i) = (1 - L\rho)\left(L\tau(i)\right) + Lg \quad (3.29)$$

โดยที่

$L\tau(i)$ = ฟีโรโมนของแหล่งวัตถุดิบ i ในการเลือกตำแหน่งที่ตั้ง

$L\rho$ = อัตราการระเหยของค่าฟีโรโมนที่ใช้ในการเลือกตำแหน่งที่ตั้ง

Lg = ส่วนกลับของระยะทางรวมของเส้นทางที่สั้นที่สุดในรอบนั้นๆ ของราชินีมด

ในแต่ละรอบการกระทำซ้ำ ราชินีมดตัวที่ให้ค่า Fitness function ต่ำสุด 16 อันดับแรกจะเป็นตัวที่ปล่อยฟีโรโมน หมายความว่าถ้าราชินีมดตัวที่ให้ค่า Fitness function ต่ำสุด 16 อันดับแรก ไม่ได้เลือกเมืองนั้นในการทำรังหรือไม่ได้เลือกแหล่งวัตถุดิบนั้นเพื่อเปิดเป็นโรงงาน จะทำการปรับค่าฟีโรโมนตามสมการ (3.28) เพื่อทำให้ฟีโรโมนระเหยไป แต่หากราชินีมดตัวที่ให้ค่า Fitness function ต่ำสุด 16 อันดับแรก เลือกเมืองนั้นในการทำรังหรือเลือกแหล่งวัตถุดิบนั้นเพื่อเปิดเป็นโรงงาน จะทำการปรับค่าฟีโรโมนตามสมการ (3.29)โดยจะมีการทำให้ฟีโรโมนระเหยพร้อมทั้งเพิ่มฟีโรโมนด้วยการบวกค่าส่วนกลับของค่า Fitness function ต่ำสุดในแต่ละรอบนั้นเข้าไป เพื่อสร้างความโดดเด่นให้กับเมืองนั้น ผลเฉลยของอัลกอริทึมนี้จะได้จากการทำซ้ำทั้ง 6 ขั้นตอนตามเงื่อนไขหรือข้อกำหนดอื่นๆ ที่กำหนดไว้

หลังจากทราบตำแหน่งที่ตั้งโรงงานจากผลเฉลยของอัลกอริทึมแล้ว จะทำการปรับปรุงคุณภาพคำตอบของปัญหาการจัดเส้นทางสำหรับยานพาหนะอีกครั้ง เพื่อปรับปรุงคำตอบให้มีประสิทธิภาพมากยิ่งขึ้น โดยขั้นตอนวิธีการนั้นทำเช่นเดียวกันกับขั้นตอนที่ 3.5.3, 3.5.4 และ 3.5.5 แต่สำหรับขั้นตอน 3.5.4 การปรับปรุงคุณภาพคำตอบนั้น จากเดิมผลเฉลยที่ได้จากการแก้ปัญหาการจัดเส้นทางสำหรับยานพาหนะเท่านั้นที่จะผ่านการปรับปรุงคุณภาพคำตอบ เปลี่ยนเป็นผลเฉลยที่ได้จากมดงานทุกตัวจะผ่านการปรับปรุงคุณภาพคำตอบแทน

3.6 คอมพิวเตอร์และโปรแกรมที่ใช้คำนวณ

ในการทดลองตั้งแต่ขั้นตอนการเริ่มหาคำตอบเริ่มต้นไปจนถึงการปรับปรุงคำตอบนั้น ผู้วิจัยใช้เครื่องมือในการทดลองคือ คอมพิวเตอร์และโปรแกรมที่ใช้ในการคำนวณ โดยมีรายละเอียดของคุณลักษณะ(Specification) ของคอมพิวเตอร์และโปรแกรมที่ใช้ดังต่อไปนี้

3.6.1 คุณลักษณะของคอมพิวเตอร์ที่ใช้

ชนิด (Type) : PC

ยี่ห้อ (Brand) : HP Compaq

รุ่น (Model) : dx2710 MT

CPU : Intel(R) Core™2 Duo CPU E8400 @ 3.0 GHz

RAM : 2.00 GB

3.6.2 โปรแกรมที่ใช้

ในการทดลองนี้ผู้วิจัยเลือกใช้โปรแกรม Dev C++ (Version 4.9.9.2) ในการคำนวณและประมวลผล

บทที่ 4
วิเคราะห์ผลการวิจัย

ในบทนี้จะกล่าวถึงการกำหนดค่าพารามิเตอร์ที่เกี่ยวข้องและการวิเคราะห์ผลการวิจัย งานวิจัยนี้จะทำการทดลองหาตำแหน่งที่ตั้งของโรงงานแป้งมันสำปะหลังจากตำแหน่งที่ตั้งของลานมันสำปะหลัง โดยจะเดินทางไปส่งสินค้า (แป้งมันสำปะหลังดิบ) ให้กับลูกค้าก่อน หลังจากนั้นจึงเดินทางไปรับวัตถุดิบ (หัวมันสำปะหลังสด) จากลานมันสำปะหลัง ซึ่งทำการทดลอง 7 กรณี ซึ่งในแต่ละกรณีทำการทดลองซ้ำทั้งหมด 5 ครั้ง โดยกรณีสุดท้าย โรงงานแป้งมันสำปะหลังทำการผลิต 20 เปอร์เซ็นต์ของกำลังการผลิต ทำการทดลองเพื่อเปรียบเทียบผลคำตอบสำหรับปัญหาการจัดเส้นทางการขนส่งสำหรับยานพาหนะของอัลกอริทึมกับผลของโปรแกรม Lingo เพื่อเปรียบเทียบประสิทธิภาพของอัลกอริทึม

4.1 การกำหนดค่าพารามิเตอร์ที่เกี่ยวข้อง

วิธีระบบอาณานิคมมด (Ant Colony Optimization) มีพารามิเตอร์ที่เกี่ยวข้องดังที่กล่าวมาในบทที่ 3.5.1 ซึ่งผู้วิจัยได้กำหนดค่าพารามิเตอร์ต่างๆ ดังตารางที่ 4.1

ตารางที่ 4.1 การกำหนดค่าพารามิเตอร์ที่เกี่ยวข้องสำหรับวิธีการอาณานิคมมด

กำหนดพารามิเตอร์	Location Problem (LP)	Vehicle Routing Problem (VRP)	Final Vehicle Routing Problem
1. จำนวนรอบที่ต้องการวนซ้ำ	100	5	50
2. จำนวนประชากรมด	50	5	50
3. ฟีโรโมน (τ)	เริ่มต้นที่ 1	เริ่มต้นที่ 1	เริ่มต้นที่ 1
4. อัตราการระเหยของฟีโรโมน (ρ)	0.02	0.02	0.02
5. อัลฟา (α)	1	1	1
6. เบต้า (β)	5	5	5

4.2 กรณีที่ 1 โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิต

โรงงานแป้งมันสำปะหลังที่จะก่อตั้งทำการผลิตเต็มกำลังการผลิตของโรงงานคือ 300 ตันแป้งต่อวัน หรือสามารถรับวัตถุดิบได้ 900 ตันต่อวันและไม่จำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคัน

ตารางที่ 4.2 ผลการทดลองกรณีที่ 1

ครั้งที่	ตำแหน่งที่เปิดโรงงาน	จำนวนรถบรรทุก (คัน)	ระยะทางรวม (กม.)
1	30	62	26519.4
2	30	62	26351.4
3	30	61	26399.6
4	30	61	26269.6
5	30	62	26092.1

จากตารางที่ 4.2 จะเห็นว่าตำแหน่งที่เปิดเป็นโรงงานแป้งมันสำปะหลังที่ทำให้ระยะทางรวมในการขนส่งสั้นที่สุด คือ ตำแหน่งลานมันสำปะหลังที่ 30 (ตำบลเมืองเดช อำเภอเดชอุดม จังหวัดอุบลราชธานี) ข้อมูลเส้นทางการขนส่งแสดงในภาคผนวก ง

4.3 กรณีที่ 2 โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิตและจำกัดระยะทางสูงสุด

โรงงานแป้งมันสำปะหลังที่จะก่อตั้งทำการผลิตเต็มกำลังการผลิตของโรงงานคือ 300 ตันแป้งต่อวัน หรือสามารถรับวัตถุดิบได้ 900 ตันต่อวันและจำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคันเท่ากับ 800 กิโลเมตรต่อวัน

ตารางที่ 4.3 ผลการทดลองกรณีที่ 2

ครั้งที่	ตำแหน่งที่เปิดโรงงาน	จำนวนรถบรรทุก (คัน)	ระยะทางรวม (กม.)
1	30	66	24468.5
2	30	66	24415.1
3	30	67	24400.3
4	30	67	24319.2
5	30	66	24405.5

จากตารางที่ 4.3 จะเห็นว่าตำแหน่งที่เปิดเป็นโรงงานแป้งมันสำปะหลังที่ทำให้ระยะทางรวมในการขนส่งสั้นที่สุด คือ ตำแหน่งลานมันสำปะหลังที่ 30 (ตำบลเมืองเดช อำเภอเดชอุดม จังหวัดอุบลราชธานี) ข้อมูลเส้นทางการขนส่งแสดงในภาคผนวก ง

4.4 กรณีที่ 3 โรงงานแป้งมันสำปะหลังทำการผลิต 80 เปอร์เซ็นต์ของกำลังการผลิต

โรงงานแป้งมันสำปะหลังที่จะก่อตั้งทำการผลิตเต็มกำลังการผลิตของโรงงานคือ 240 ตันแป้งต่อวัน หรือสามารถรับวัตถุดิบได้ 720 ตันต่อวันและไม่จำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคัน

ตารางที่ 4.4 ผลการทดลองกรณีที่ 3

ครั้งที่	ตำแหน่งที่เปิดโรงงาน	จำนวนรถบรรทุก (คัน)	ระยะทางรวม (กม.)
1	30	49	21265.4
2	30	50	21470.1
3	30	49	21685.4
4	231	49	21631
5	30	49	21314.6

จากตารางที่ 4.4 จะเห็นว่าตำแหน่งที่เปิดเป็นโรงงานแป้งมันสำปะหลังที่ทำให้ระยะทางรวมในการขนส่งสั้นที่สุด คือ ตำแหน่งลานมันสำปะหลังที่ 30 (ตำบลเมืองเดช อำเภอเดชอุดม จังหวัดอุบลราชธานี) ข้อมูลเส้นทางการขนส่งแสดงในภาคผนวก ง

4.5 กรณีที่ 4 โรงงานแป้งมันสำปะหลังทำการผลิต 80 เปอร์เซ็นต์ของกำลังการผลิตและจำกัดระยะทางสูงสุด

โรงงานแป้งมันสำปะหลังที่จะก่อตั้งทำการผลิตเต็มกำลังการผลิตของโรงงานคือ 240 ตันแป้งต่อวัน หรือสามารถรับวัตถุดิบได้ 720 ตันต่อวันและจำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคันเท่ากับ 800 กิโลเมตรต่อวัน

ตารางที่ 4.5 ผลการทดลองกรณีที่ 4

ครั้งที่	ตำแหน่งที่เปิดโรงงาน	จำนวนรถบรรทุก (คัน)	ระยะทางรวม (กม.)
1	40	44	17572.2
2	30	51	17684.6
3	40	44	17168.9
4	30	52	17524
5	30	51	17584.1

จากตารางที่ 4.5 จะเห็นว่าตำแหน่งที่เปิดเป็นโรงงานแป้งมันสำปะหลังที่ทำให้ระยะทางรวมในการขนส่งสั้นที่สุด คือ ตำแหน่งลานมันสำปะหลังที่ 40 (ตำบลบุเปือย อำเภอน้ำยืน จังหวัดอุบลราชธานี) ข้อมูลเส้นทางการขนส่งแสดงในภาคผนวก ง

4.6 กรณีที่ 5 โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำลังการผลิต

โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำลังการผลิตของโรงงานคือ $\frac{300}{2} = 150$ ตันแป้งต่อวัน หรือสามารถรับวัตถุดิบได้ 450 ตันต่อวันและไม่จำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคัน

ตารางที่ 4.6 ผลการทดลองกรณีที่ 5

ครั้งที่	ตำแหน่งที่เปิดโรงงาน	จำนวนรถบรรทุก (คัน)	ระยะทางรวม (กม.)
1	41	28	11736
2	43	30	10942
3	41	28	11896.1
4	42	30	12174.5
5	41	28	12316.2

จากตารางที่ 4.6 จะเห็นว่าตำแหน่งที่เปิดเป็นโรงงานแป้งมันสำปะหลังที่ทำให้ระยะทางรวมในการขนส่งสั้นที่สุด คือ ตำแหน่งลานมันสำปะหลังที่ 43 (ตำบลเก่าขาม อำเภอน้ำยืน จังหวัดอุบลราชธานี) ข้อมูลเส้นทางการขนส่งแสดงในภาคผนวก ง

4.7 กรณีที่ 6 โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำลังการผลิตและจำกัดระยะทางสูงสุด

โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำลังการผลิตของโรงงานคือ $\frac{300}{2} = 150$ ตันแป้งต่อวัน หรือสามารถรับวัตถุดิบได้ 450 ตันต่อวันและจำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคันเท่ากับ 800 กิโลเมตรต่อวัน เพื่อให้สามารถเดินทางไปและกลับมายังโรงงานได้ภายใน 1 วัน

ตารางที่ 4.7 ผลการทดลองกรณีที่ 6

ครั้งที่	ตำแหน่งที่เปิดโรงงาน	จำนวนรถบรรทุก (คัน)	ระยะทางรวม (กม.)
1	41	30	9372.92
2	41	30	9423.01
3	41	30	9264.4
4	41	30	9415.38
5	40	25	8832.26

จากตารางที่ 4.7 จะเห็นว่าตำแหน่งที่เปิดเป็นโรงงานแป้งมันสำปะหลังที่ทำให้ระยะทางรวมในการขนส่งสั้นที่สุด คือ ตำแหน่งลานมันสำปะหลังที่ 40 (ตำบลบุเปือย อำเภอน้ำยืน จังหวัดอุบลราชธานี) ข้อมูลเส้นทางการขนส่งแสดงในภาคผนวก ง

4.8 กรณีที่ 7 โรงงานแป้งมันสำปะหลังทำการผลิต 20 เปอร์เซ็นต์ของกำลังการผลิต

โรงงานแป้งมันสำปะหลังทำการผลิต 20 เปอร์เซ็นต์ของกำลังการผลิตของโรงงานคือ $300 \times \frac{20}{100} = 60$ ตันแป้งต่อวัน หรือสามารถรับวัตถุดิบได้ 180 ตันต่อวัน สำหรับกรณีนี้ผู้วิจัยทำการทดลองหาคำตอบเพียงหนึ่งครั้งเท่านั้น เนื่องจากทำการทดลองเพื่อเปรียบเทียบประสิทธิภาพของอัลกอริทึมเท่านั้น

ตารางที่ 4.8 ผลการทดลองกรณีที่ 7

ครั้งที่	ตำแหน่งที่เปิดโรงงาน	จำนวนรถบรรทุก (คัน)	ระยะทางรวม (กม.)
1	40	5	3911.34

จากตารางที่ 4.8 จะเห็นว่าตำแหน่งที่เปิดเป็นโรงงานแป้งมันสำปะหลังที่ทำให้ระยะทางรวมในการขนส่งสั้นที่สุด คือ ตำแหน่งลานมันสำปะหลังที่ 40 (ตำบลบุเปือย อำเภอน้ำยืน จังหวัดอุบลราชธานี) ข้อมูลเส้นทางการขนส่งแสดงในภาคผนวก ง ผลการแก้ปัญหาทั้ง 7 กรณี แสดงดังตารางที่ 4.9

ตารางที่ 4.9 สรุปผลการแก้ปัญหา LRP ของกรณีศึกษาทั้ง 7 กรณี

กรณีที่	ลักษณะ	เปิดที่	ระยะทาง (km.)	เวลา (s)
1	ผลิตเต็มกำลังการผลิต	30	26,092.5	9,026.44
2	ผลิตเต็มกำลังการผลิตและจำกัดระยะทางสูงสุด	30	24,319.2	6,235.53
3	ผลิต 80 เปอร์เซ็นต์ของกำหลังการผลิต	30	21,265.4	6,858.92
4	ผลิต 80 เปอร์เซ็นต์ของกำหลังการผลิตและจำกัดระยะทางสูงสุด	40	17,168.9	4,332.41
5	ผลิต 50 เปอร์เซ็นต์ของกำหลังการผลิต	43	10,942	4,012.59
6	ผลิต 50 เปอร์เซ็นต์ของกำหลังการผลิตและจำกัดระยะทางสูงสุด	40	8,832.26	2,745.77
7	ผลิต 20 เปอร์เซ็นต์ของกำหลังการผลิต	40	3,911.34	500.995

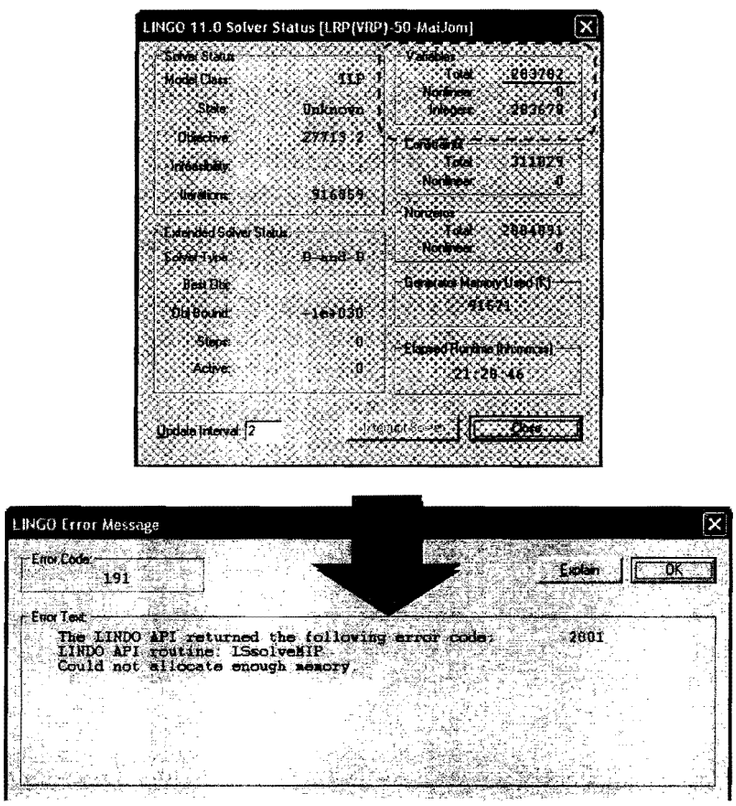
4.7 วิเคราะห์ผลการทดลอง

จากผลการทดลองจะเห็นว่ากรณีที่ 1, 3 และกรณีที่ 5 ระยะทางบางเส้นทางนั้น มีระยะทางไกลเกินกว่าที่จะเดินทางไปและกลับมายังโรงงานแป้งมันสำปะหลังได้ภายใน 1 วัน เนื่องจากไม่มีการจำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคัน ผู้วิจัยจึงทำการกำหนดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคัน ดังกรณีที่ 2, 4 และกรณีที่ 6 สมมติว่า รถบรรทุกเดินทางเฉลี่ยที่ 60 กิโลเมตรต่อชั่วโมง ดังนั้น 800 กิโลเมตร จึงใช้เวลาเดินทางทั้งสิ้น 13.3 ชั่วโมง ทำให้สามารถเดินทางไปและกลับมายังโรงงานแป้งมันสำปะหลังได้ภายใน 1 วัน เพื่อให้ง่ายต่อการวางแผนไปรับวัตถุดิบ และไปส่งสินค้า

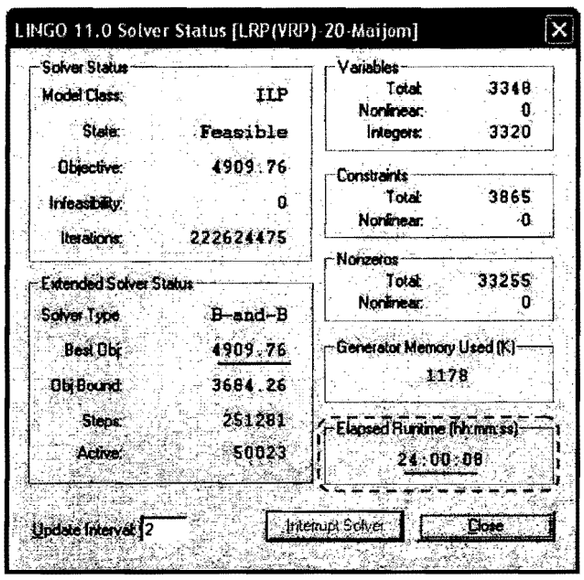
กรณีที่กำลังการผลิตที่เท่ากัน จะเห็นว่ากรณีที่จำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคันนั้น จะมีระยะทางรวมน้อยกว่ากรณีที่ไม่จำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคัน คือกรณีที่ 2 ระยะทางรวมน้อยกว่ากรณีที่ 1, กรณีที่ 4 ระยะทางรวมน้อยกว่ากรณีที่ 3 และกรณีที่ 6 ระยะทางรวมน้อยกว่ากรณีที่ 5 เนื่องจากว่าการกำหนดระยะทางสูงสุดในการ

เดินทางของรถบรรทุกแต่ละคันเป็นการลดขอบเขตของคำตอบให้แคบลง ดังนั้นกลุ่มลานมันและกลุ่มลูกค้าที่ต้องเดินทางไปรับ-ส่งสินค้าอยู่บริเวณใกล้เคียงกัน จึงทำให้ระยะทางรวมที่ได้น้อยกว่ากรณีที่ไม่ได้จำกัดระยะทางสูงสุดในการเดินทางของรถบรรทุกแต่ละคัน

ผู้วิจัยได้ทำการทดลองกรณีที่ 7 โรงงานแป้งมันสำปะหลังทำการผลิต 20 เปอร์เซ็นต์ของกำลังการผลิต เพื่อที่จะนำผลลัพธ์ที่ได้ไปเปรียบเทียบกับโปรแกรม Lingo เนื่องจากในกรณีที่ 1 - 6 มีจำนวนตัวแปรที่ไม่ทราบค่าเป็นจำนวนมาก ดังภาพที่ 4.1 ซึ่งคอมพิวเตอร์ที่ใช้ในการทดลองมีประสิทธิภาพไม่เพียงพอ โปรแกรม Lingo จึงไม่สามารถที่จะประมวลผลได้ สำหรับกรณีที่ 7 มีจำนวนตัวแปรน้อย ดังนั้นผู้วิจัยจึงนำข้อมูลตำแหน่งที่ตั้งโรงงานแป้ง กลุ่มลูกค้าและกลุ่มลานมัน ที่ได้จากผลการทดลองกรณีที่ 7 มาประมวลผลเพื่อแก้ปัญหาการจัดเส้นทางการขนส่งด้วยโปรแกรม Lingo โดยมี Source code โปรแกรม Lingo ดังภาคผนวกที่ จ และกำหนดให้โปรแกรมประมวลผลนานเป็นเวลาประมาณ 24 ชั่วโมง แล้วจึงหยุดการทำงานของโปรแกรม Lingo ดังภาพที่ 4.2 เพื่อใช้คำตอบ Best Objective ที่ได้จากโปรแกรม Lingo ณ เวลานี้ ดังแสดงในตารางที่ 4.9 มาเปรียบเทียบกับคำตอบที่ได้จากการประมวลผลด้วยอัลกอริทึมที่ออกแบบดังตารางที่ 4.10



ภาพที่ 4.1 ตัวอย่างกรณีที่ 5 จำนวนตัวแปรมีจำนวนมากไม่สามารถประมวลผลหาคำตอบได้



ภาพที่ 4.2 ระยะเวลาและผลที่ได้จากการหาคำตอบด้วยโปรแกรม Lingo กรณีที่ 7

ตารางที่ 4.10 ผลลัพธ์ที่ได้จากการประมวลด้วยโปรแกรม Lingo กรณีที่ 7

No	Routes	Sum_H (ton)	Sum_D (ton)	Distance (km.)
1	40-13-163-38-40	13.856	15	769.369
2	40-153-165-90-76-84-112-40	14.934	0	1111.44
3	40-13-113-40	15	15	731.926
4	40-13-129-114-145-155-132-78-199-98-40	14.734	15	1451.85
5	40-13-113-111-146-162-159-160-40	14.966	15	845.177
Total_sum		180	60	4909.76

ตารางที่ 4.11 ผลลัพธ์ที่ได้จากอัลกอริทึมที่ออกแบบ กรณีที่ 7

No	Routes	Sum_H (ton)	Sum_D (ton)	Distance (km.)
1	40-13-113-40	15	15	731.926
2	40-13-113-111-114-145-146-153-155-199-98-40	14.994	15	1387.91
3	40-13-129-78-76-90-84-165-132-40	14.986	15	981.297
4	40-13-112-162-163-159-160-40	14.973	15	781.144
5	40-38-40	13.537	0	29.064
Total_sum		180	180	3911.34

หมายเหตุ

Sum_H = ปริมาณหัวมันสดที่ไปรับจากลานมันสำปะหลังโดยรถบรรทุกแต่ละคัน (ตัน)

Sum_D = ปริมาณแป้งมันดิบที่ไปส่งให้ลูกค้าโดยรถบรรทุกแต่ละคัน (ตัน)

Distance = ระยะทางของรถบรรทุกแต่ละคัน (กิโลเมตร)

Total_Sum_H = ปริมาณหัวมันสดที่ไปรับทั้งหมด ซึ่งรวมกับปริมาณหัวมันสด ณ จุดที่เปิดเป็นโรงงานแป้งมันสำปะหลัง (ตัน)

จากตารางที่ 4.9 จะเห็นว่าวิธี Exact Method ประมวลผลด้วยโปรแกรม Lingo ที่แก้ปัญหาเฉพาะการจัดเส้นทางการขนส่ง (Vehicle Routing Problem) ใช้เวลาประมวลผลค่อนข้างนานเมื่อเทียบกับอัลกอริทึมที่ออกแบบไว้ ซึ่งอัลกอริทึมที่ออกแบบนั้น สามารถแก้ปัญหาได้ทั้งปัญหาการเลือกสถานที่ตั้งและจัดเส้นทางการขนส่ง (Location Routing Problem) ไปพร้อมๆ กัน และผลลัพธ์ที่ได้จากอัลกอริทึมให้ค่าระยะทางรวมที่สั้นกว่าผลลัพธ์ที่ได้จากโปรแกรม Lingo ซึ่งแสดงถึงประสิทธิภาพในการหาคำตอบของอัลกอริทึมที่ออกแบบโดยใช้เวลาที่เหมาะสม

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้เป็นการนำเสนอวิธีการแก้ปัญหาการเลือกตำแหน่งโรงงานผลิตแป้งมันสำปะหลังที่เหมาะสมในภาคตะวันออกเฉียงเหนือ พร้อมทั้งจัดเส้นทางการขนส่งสินค้าและรับวัตถุดิบให้มีประสิทธิภาพมากที่สุดภายใต้วัตถุประสงค์ด้านเศรษฐศาสตร์ อย่างไรก็ตามปัญหา LRP นับเป็นปัญหาที่มีความยุ่งยากซับซ้อนปัญหาหนึ่งและจัดเป็นปัญหาเอ็นพีฮาร์ด (NP-Hard) ฉะนั้นวิธีการที่นำมาแก้ปัญหาเหล่านี้ จึงมีความหลากหลายวิธีการทั้งวิธีการหาคำตอบที่ดีที่สุด (Exact Method) และวิธีฮิวริสติก (Heuristic Method) แต่ละวิธีจะมีข้อดีและข้อด้อยแตกต่างกันออกไป เนื่องจากกรณีศึกษามีขนาดของปัญหาที่ใหญ่ คือมีจำนวนลูกค้าและแหล่งวัตถุดิบรวมกัน 283 แห่ง ทำให้เกิดทางเลือกที่เป็นไปได้จำนวนมหาศาล จึงไม่สามารถใช้วิธีการคำนวณแบบกำหนดการเชิงเส้น (Linear Programming) หรือสามารถหาค่าที่ดีที่สุด (Optimal Solution) ได้จากวิธี Exact Method โดยทั่วไป ดังนั้นผู้วิจัยจึงได้พัฒนาวิธีเมต้าฮิวริสติกที่มีชื่อเรียกว่า "วิธีระบบอาณานิคมมด (Ant Colony Optimization)" เพื่อแก้ปัญหา LRP ของกรณีศึกษานี้

ผลจากการทดสอบสามารถสรุปได้ 3 กรณีด้วยกัน (1) หากจะทำการสร้างโรงงานแป้งมันสำปะหลังที่ทำการผลิตเต็มกำลังการผลิตของโรงงานคือ 300 ตันแป้งต่อวัน ควรเลือกเปิดโรงงานแป้งมันสำปะหลังที่ ต.เมืองเดช อ.เดชอุดม จ.อุบลราชธานี ซึ่งจะเป็นจุดที่ให้ระยะทางรวมในการขนส่งที่ต่ำคือ 24,319.2 กิโลเมตร และควรเลือกกลุ่มลูกค้าและกลุ่มลานมันพร้อมทั้งจัดเส้นทางการขนส่งตามผลเฉลยของกรณีที่จำกัดระยะทางในการขนส่ง เนื่องจากให้ค่าระยะทางที่ต่ำกว่ากรณีที่ไม่จำกัดระยะทางและง่ายต่อการวางแผนในการรับ-ส่งสินค้าและวัตถุดิบได้ภายในระยะเวลา 1 วัน อันจะส่งผลให้โรงงานแป้งมันสำปะหลังสามารถผลิตอย่างต่อเนื่องได้ (2) หากจะทำการสร้างโรงงานแป้งมันสำปะหลังที่ทำการผลิต 80 เปอร์เซ็นต์ของกำลังการผลิตของโรงงาน ผู้ประกอบการควรเลือกเปิดโรงงานแป้งมันสำปะหลังที่ ต.บุเปือย อ.น้ำยืน จ.อุบลราชธานี ซึ่งเป็นคำตอบของกรณีที่จำกัดระยะทางในการขนส่ง โดยมีระยะทางรวมในการขนส่งคือ 17,168.9 กิโลเมตร (3) หากจะทำการสร้างโรงงานแป้งมันสำปะหลังที่ทำการผลิต 50 เปอร์เซ็นต์ของกำลังการผลิตของโรงงานคือ 150 ตันแป้งต่อวัน ควรเลือกเปิดโรงงานแป้งมันสำปะหลังที่ ต.บุเปือย อ.น้ำยืน จ.อุบลราชธานี โดยมีระยะทางรวมในการขนส่งคือ 8,832.26 กิโลเมตร และควรเลือกกลุ่มลูกค้าและ

กลุ่มลานมันพร้อมทั้งจัดเส้นทางการขนส่งตามผลเฉลยของกรณีที่จำกัดระยะทางในการขนส่ง เนื่องจากให้ค่าระยะทางที่ต่ำกว่ากรณีที่ไม่จำกัดระยะทางและง่ายต่อการวางแผนในการรับ-ส่งสินค้าและวัตถุดิบได้ภายในระยะเวลา 1 วัน อันจะส่งผลให้โรงงานแป้งมันสำปะหลังสามารถผลิตอย่างต่อเนื่องได้

5.2 ข้อเสนอแนะ

งานวิจัยนี้เป็นปัญหาขนาดใหญ่จำนวนรอบในการกระทำซ้ำจึงสมควรมีค่ามาก เพื่อให้ได้คำตอบที่หลากหลาย ซึ่งพารามิเตอร์ที่ผู้วิจัยใช้อาจยังไม่ดีเท่าที่ควรเพราะเนื่องด้วยความจำกัดด้านเวลาและความสามารถของคอมพิวเตอร์ที่ใช้ในการคำนวณ

สำหรับวิธีอาณานิคมมดมีพารามิเตอร์ต่างๆ ที่เกี่ยวข้องอยู่หลายค่าด้วยกันที่มีผลต่อการค้นหาคำตอบของแต่ละปัญหา โดยค่าพารามิเตอร์ที่เหมาะสมสำหรับปัญหาที่แตกต่างกันอาจมีค่าที่แตกต่างกันด้วย จึงควรทดสอบเพื่อเลือกค่าที่เหมาะสมกับปัญหานั้นๆ งานวิจัยในอนาคตผู้วิจัยจะได้พัฒนาขั้นตอนการปรับปรุงคุณภาพคำตอบให้ทรงประสิทธิภาพมากยิ่งขึ้น และประยุกต์ใช้วิธีอาณานิคมมดกับกรณีศึกษาอื่นๆ พร้อมทั้งวิเคราะห์หาค่าพารามิเตอร์ที่เหมาะสมต่อไป

5.3 ปัญหาและอุปสรรค

5.3.1 ความยากในการเก็บข้อมูล เพราะข้อมูลที่ใช้ในการวิจัยเกี่ยวข้องกับหลายหน่วยงาน

5.3.2 ข้อมูลที่ทำการขอกับหน่วยงานต่างๆ บางข้อมูลเป็นความลับ เช่น ข้อมูลลูกค้าของโรงงานแป้งมันสำปะหลัง เป็นต้น

5.3.3 เนื่องจากข้อมูลที่เกี่ยวข้องกับงานวิจัยนี้มีจำนวนมาก จึงต้องใช้เวลานานในการเก็บรวบรวมข้อมูล

5.3.4 ขาดความชำนาญในการเขียนภาษา C++ จึงใช้เวลาค่อนข้างนานในการศึกษาและเขียนโปรแกรม

เอกสารอ้างอิง

เอกสารอ้างอิง

นัทธพงศ์ นันทสำเริง และระพีพันธ์ ปีตาคะโส. "วิธีค้นหาเนเบอร์ฮูดผันแปรสำหรับปัญหาการเลือกสถานที่ตั้งแบบหลายขั้นตอนและหลายเงื่อนไข", ใน การประชุมวิชาการข่ายงานวิศวกรรมอุตสาหการ. น.420. อุบลราชธานี : โรงพิมพ์มหาวิทยาลัยอุบลราชธานี, 2553.

สุพรรณ สุดสนธิ์ และสมบัติ สินธุเชาวน์. "การประยุกต์วิธีการหาคำตอบโดยวิธีอัลกอริทึมเชิงพันธุกรรม สำหรับปัญหาการเลือกสถานที่ศูนย์กระจายสินค้าในภาคธุรกิจการขนส่งและลอจิสติกส์", ใน การประชุมวิชาการข่ายงานวิศวกรรมอุตสาหการ. น.101. โรงแรมอินเตอร์คอนติเนนตัล : กรุงเทพฯ, 2549.

สุพรรณ สุดสนธิ์ และคณะ. "วิธีอาณานิคมและขั้นตอนการปรับปรุงคุณภาพคำตอบสำหรับปัญหาสถานที่ตั้งศูนย์กระจายสินค้าแบบหลายแห่งและการจัดเสนทางการขนส่ง", ใน การประชุมวิชาการข่ายงานวิศวกรรมอุตสาหการ. น.81. โรงแรม Royal Phuket City : ภูเก็ต, 2550.

กลุ่มทำงานศึกษาและวิเคราะห์สินค้าเกษตรประเภทมันสำปะหลัง. ผลการศึกษาสินค้าเกษตรประเภทมันสำปะหลัง. รายงานต่อสำนักงานคณะกรรมการกำกับการซื้อขายสินค้าเกษตรล่วงหน้า, 2554.

Bai and et al. "Max-min ant system on gpu with cuda", in Proceedings of the 2009 Fourth International Conference on Innovative Computing. p.801-804. Changchun China : Jilin University, 2009.

Balakrishnan A., Ward J.E. and Wong R.T. "Integrated facility location and vehicle routing models: recent work and future prospects", American Journal of Mathematical and Management Sciences. 7: 35-61, 1987.

Beckers, R., Deneubourg, J. L. and Goss S. "Trails and U-turns in the selection of the shortest path by the ant Lasius Niger", J. Theoretical Biology. 159: 397-415, 1992.

Bianchi, L., Gambardella, L.M. and Dorigo, M. "An ant colony optimization approaches to the probabilistic traveling salesman problem", Mathematical Modeling and Algorithms. 3(4): 403-425, 2004.

Blum, C. and Fouldm L. R. "Metaheuristics in combinatorial optimization: Overview and conceptual comparison", ACM Computing Surveys. 35: 268-308, 2003.

เอกสารอ้างอิง (ต่อ)

Bullnheimer, B., Hartl, R.F. and Strauss, C. "Applying the Ant System to the Vehicle routing Problem", in Second Metaheuristics International Conference. France : Sophia-Antipolis, 1997.

Bullnheimer, B., Hartl, R.F. and Strauss, C. "A new rank-based version of the Ant System: A computational study", Central European Journal for Operations Research and Economics. 7: 25-38, 1999.

Craus, M. and Rudeanu, L. "Multi-level parallel framework", International Journal of Computing. 3(3), 2004.

Craus, M. and Rudeanu, L. "Parallel framework for ant-like algorithms", in Proceedings of the 3rd International Symposium on Parallel and Distributed Computing. p.36-41, IEEE Computer Society, 2004.

Christofides, N., Mingozzi, A. and Toth, P. "The Location Routing Problem", Combinatorial Optimization. 237-279, 1979.

Colorni, A., Dorigo, M., Maniezoo, V. and Trubian, M. "Ant system for job-shop scheduling", Belgian Journal of Operations Research Statistics and Computer Science(JORBEL). 34: 39-53, 1994.

Dantzig, G.B. and Ramser, J.H. "The truck dispatching problem", Management Science. 6: 80-91, 1959.

Delisle, P., Gravel, M., Krajecki, M., Gagné, C. and Price, W. "Comparing parallelization of an ACO: message passing vs. shared memory", in Proceedings of the 2nd International Workshop on Hybrid Metaheuristics, Lecture Notes in Computer Science. p.1-11, Springer-Verlag Berlin Heidelberg, 2005.

Delisle, P., Gravel, M., Krajecki, M., Gagné, C. and Price, W. "A shared memory parallel implementation of ant colony optimization", in Proceedings of the 6th Metaheuristics International Conference. p. 257-264, Vienne: Autriche, 2005.

Doerner, K. and et al. "Parallel ant systems for the capacitated vehicle routing problem", in Proceedings of the 4th European Conference Evolutionary Computation in Combinatorial Optimization. p.72-83, Springer-Verlag Berlin Heidelberg, 2004.

เอกสารอ้างอิง (ต่อ)

Doerner, K., Hartl, R. and Lucka, M. "A parallel version of the D-ant algorithm for the vehicle routing problem" in Proceedings of the International Workshop on Parallel Numerics. p.109-118, Austria: University of Salzburg, 2005.

Doerner, K., Hartl, R., Benkner, S. and Lucka, M. "Parallel cooperative savings based ant colony optimization – multiple search and decomposition approaches", Parallel Processing Letters. 16(3): 351-370, 2006.

Dorigo M. Optimization learning and natural algorithms. Ph.D. dissertation. Italy: Politecnico Milano, 1992.

Dorigo, M., Maniezzo, V. and Colorni, A. "Positive feedback as a search strategy", Technical report. p.91-016, 1991.

Dorigo, M., Maniezzo, V. and Colorni, A. "Ant system: Optimization by a colony of cooperating agents", IEEE Trans SystManCybern. p.29-41, IEEE Transactions on Systems, 1996.

Dorigo, M. and Gambardella, L.M. "Ant Colonies for the traveling salesman problem", BioSystem. 43(2): 72-81, 1997.

Dorigo, M. and Stutzle, T. Ant Colony Optimization (Bradford Books). The MIT Press, 2004.

Houda Derbel and et al. "An Iterated Local Search for Solving A Location-Routing Problem", Electronic Notes in Discrete Mathematics. 36: 875-882, 2010.

G. Cornuejols, G. L. Nemhauser and L. A. Wolsey. "The uncapacitated facility location problem", Discrete Location Theory. 199-171, 1990.

Gambardella, L.M., Taillard, E. and Dorigo, M. "Ant colonies for the quadratic assignment problem", Operational Research Society. 50(2): 167-176, 1999.

Golden, B., T. Magnanti and H. Nguyen. "Implementing vehicle routing algorithms" Networks. 7: 113-148, 1977.

Fu, J., Lei, L. and Zhou, G. "A parallel ant colony optimization algorithm with gpu-acceleration based on all-in-roulette selection", in Proceedings of the 3rd International Workshop on Advanced Computational Intelligence. p.260-264, China: Jiangsu, 2010.

เอกสารอ้างอิง (ต่อ)

Jitimar Piyamalmas. Optimization of biogas plant location. Master thesis of engineering. Bangkok: King monggkut's university of technology thonburi, 2003.

Kytojoki, J., T. Nuotio, O. Braysy and M. Gendreau. "An efficient variable neighborhood search heuristic for very large scale vehicle routing problems", Computers and Operations Research. 34: 2743-2757, 2007.

Laporte, G. "The vehicle routing problem: An overview of exact and approximate algorithm", European of Journal Operational Research. 59: 345-358, 1992.

Laporte, G. and et al. "Classical and modern heuristics for the vehicle routing problem", Internetional Transactions in Operational Research. 7: 285-300, 2000.

Laurent and et al. "Iterated local search for the multiple depot vehicle scheduling problem", Computers & Industrial Engineering. 57: 277-286, 2009.

Li, N., Gao, D., Gong, G. and Chen, Z. "Realization of parallel ant colony algorithm based on tbb multi-core platform", in Proceedings of the International Forum on Information Technology and Applications 1. p.177-180, IEEE computer society, 2010.

Lv, Q., Xia, X. amd Qian, P., "A parallel ACO approach based on one pheromone matrix", in Proceedings of the 5th International Workshop on Ant Colony Optimization and Swarm Intelligence. 4150: p.332-339, Springer-Verlag Berlin Heidelberg, 2006.

Nagy, G. and Salhi, S. "Nested heuristic methods for the location-routing problem", Journal of Operational Research Society. 47: 1166-1174, 1996.

Nagy, G. and Salhi, S. "Location-routing: Issues, models and methods", European Journal of operational research. 177: 649-672, 2007.

Perl J. and M. Daskin. "A warehouse location problem", Transportation Research Quarterly. 5: 381-396, 1985.

Randall, M. and Lewis, A. "A parallel implementation of ant colony optimization", Journal of Parallel and Distributed Computing. 62(9): 1421–1432, 2002.

Rizzoli, A.E. and et al. "Time dependent vehicle routing problem with an ant colony system", Tech. Rep. IDSIA-17-03. Manno: Istituto Dalle Molle di Studi sull'Intelligenza Artificial (IDSIA), 2002.

เอกสารอ้างอิง (ต่อ)

S. H. Owen and M. S. Daskin. "Strategic facility location: A review", European Journal of Operational Research. 111: 423-447, 1998.

S. L. Hakimi. "Optimum location of switching centers and the absolute centers and medians of a graph", Operations Research. 12: 450-459, 1964.

Sodsoon S. and Sindhuchao S. "A Max Min Ant System for Multi-Depot Routing Problem", in The 2nd International Conference on Operations and Supply Chain Management (OSCM-2007). p.1165-1174, Novotel: Bangkok, 2007.

Srivastava, R. Benton, W.C. "The location-routing problem: considerations in physical distribution system design", Computers in Operations Research. 17: 427-435, 1990.

Stutzle, T. and Hoos, H. "Improvements on the ant system, introducing the MAX-MIN ant system", in Proc. ICANNGA97-Third Int. Conf. Artificial Neural Networks and Genetic Algorithm. Germany: Springer-Verlag, 1997.

Stutzle, T. and Hoos, H. "MAX-MIN Ant System", Future Generation Computer System. 16(8): 889-914, 2000.

Taibi, E., Roux, O., Fonlupt, C. and Robillard, D. "Parallel ant colonies for the quadratic assignment problem", Future Generation Computer System. 17(4): 441-449, 2001.

Toth, P. and Vigo, D. "An overview of vehicle routing problem", SIAM Monographs on Discrete Mathematics and Applications. p.1-26, Philadeplphia : SIAM, 2002.

Tsutsui, S. "Cunning ant system for quadratic assignment problem with local search and parallelization", in Proceedings of the 2nd International Conference on Pattern Recognition and Machine Intelligence. 4815: p.269-278, Germany: Springer-Verlag, 2007.

Tsutsui, S. "Parallel ant colony optimization for the quadratic assignment problems with symmetric multi processing", in Proceedings of the 6th International Conference on Ant Colony Optimization and Swarm Intelligence. 5217: p.363-370, Germany: Springer-Verlag, 2008.

เอกสารอ้างอิง (ต่อ)

Tsutsui, S. and Fujimoto, N. "Parallel ant colony optimization algorithm on a multi core processor", in Proceedings of the 7th International Conference on Swarm intelligence. p.488-495, Germany: Springer-Verlag, 2010.

Tuzun D. and Burke L. I. "A two-phase tabu search approach to the location routing problem", European Journal of Operational Research. 116: 87-99, 1999.

Wade, A. and Salhi, S. "An ant system algorithm for the mixed vehicle routing problem with backhauls", Meta-heuristic: Computer Decision-Making. p.669-719, USA : Kluwer Academic Publishers Norwell, 2004.

Watson Gandy, C.D.T., Dohrn, P.J. Depot location with van salesman – a practical approach. Omega, 1973.

Webb, M.H.J. "Cost functions in the location of depot for multiple-delivery journeys", Operational Research Quarterly. 19:311-320, 1968.

Wu, T. H., Low C. and Bai J. W. "Heuristic solutions to multi-depot location routing problems", Computers & Operations Research. 29: 1393-1415, 2002.

ภาคผนวก

ภาคผนวก ก
ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
23	1	จ3-9(6)-5/43อบ	สหกรณ์การเกษตรน้ำขุ่น จำกัด	ในเมือง	เมืองอุบลราชธานี	อุบลราชธานี	550000	0.584
24	2	จ3-9(6)-1/54อบ	ศรีสมหวังพืชผล	ลาดควาย	ศรีเมืองใหม่	อุบลราชธานี	1000000	1.062
25	3	จ3-9(6)-86/51อบ	ดี 555 บ้านไร่เจริญ	นาเลิน	ศรีเมืองใหม่	อุบลราชธานี	5000000	5.310
26	4	จ3-9(6)-23/51อบ	นางนิตยา ซังทิพย์	หัวดอน	เขื่องใน	อุบลราชธานี	100000	0.106
27	5	จ3-9(6)-87/51อบ	ศรีมงคลพืชผล	หนองเหล่า	เขื่องใน	อุบลราชธานี	63280000	67.198
28	6	จ3-9(6)-66/51อบ	ต.เขมราฐธัญญาพืช	เขมราฐ	เขมราฐ	อุบลราชธานี	45000000	47.786
29	7	จ3-9(6)-13/51อบ	แก้วพืชผล	นาแวง	เขมราฐ	อุบลราชธานี	100000	0.106
30	8	จ3-9(6)-22/52อบ	โชคไพศาล	เมืองเดช	เดชอุดม	อุบลราชธานี	3000000	3.186
31	9	จ3-9(6)-107/52อบ	พัฒนกิจพืชผล	แก้ง	เดชอุดม	อุบลราชธานี	2000000	2.124
32	10	จ3-9(6)-8/43อบ	ศรีโกศล	ท่าโพธิ์ศรี	เดชอุดม	อุบลราชธานี	500000	0.531
33	11	จ3-9(6)-35/52อบ	สมพงษ์พืชผล	บัวงาม	เดชอุดม	อุบลราชธานี	37500000	39.822
34	12	จ3-9(6)-91/51อบ	ชัยสงวนพืชผล	ตบหู	เดชอุดม	อุบลราชธานี	500000	0.531
35	13	จ3-9(6)-35/51อบ	อุบลแสงทองพืชผล	นาจะหลวย	นาจะหลวย	อุบลราชธานี	4000000	4.248
36	14	จ3-9(6)-65/51อบ	เบจกิจพืชผล	โนนสวรรค์	นาจะหลวย	อุบลราชธานี	21000000	22.300
37	15	จ3-9(6)-69/51อบ	นายบรรยงค์ ประโยธิน	โขง	น้ำยืน	อุบลราชธานี	100000	12.870
	16	จ3-9(6)-28/52อบ	ไชยมงคลพืชผล 2	โขง	น้ำยืน	อุบลราชธานี	12000000	
	17	จ3-9(6)-14/49อบ	ศรีสมบูรณ์พืชผล	โขง	น้ำยืน	อุบลราชธานี	20000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
38	18	จ3-9(6)-51/50อบ	ศรีโกศล	ยาง	น้ำยืน	อุบลราชธานี	93000000	98.758
39	19	จ3-9(6)-3/52อบ	ศรีสมบูรณ์พืชผล 2	โคมประดิษฐ์	น้ำยืน	อุบลราชธานี	3000000	17.415
	20	จ3-9(6)-27/51อบ	เก่งไมตรี	โคมประดิษฐ์	น้ำยืน	อุบลราชธานี	400000	
	21	จ3-9(6)-29/52อบ	ไชยมงคลพืชผล	โคมประดิษฐ์	น้ำยืน	อุบลราชธานี	12000000	
	22	จ3-9(6)-22/51อบ	เรืองศิริพืชผล	โคมประดิษฐ์	น้ำยืน	อุบลราชธานี	1000000	
40	23	จ3-9(6)-1/43อบ	รุ่งภิญโญ	บุเปือย	น้ำยืน	อุบลราชธานี	200000	106.510
	24	จ3-9(6)-3/51อบ	ณัฐวัตรพืชผล	บุเปือย	น้ำยืน	อุบลราชธานี	27000000	
	25	3-9(6)-4/35อบ	เรืองศิริพืชผล	บุเปือย	น้ำยืน	อุบลราชธานี	2100000	
	26	จ3-9(6)-7/54อบ	โชคประทุมพืชผล	บุเปือย	น้ำยืน	อุบลราชธานี	1000000	
	27	จ3-9(6)-51/49อบ	ศรีโกศล 2	บุเปือย	น้ำยืน	อุบลราชธานี	1000000	
	28	3-9(6)-4/25อบ	เรืองศิริพืชผล	บุเปือย	น้ำยืน	อุบลราชธานี	500000	
	29	จ3-9(6)-44/50อบ	ทองเจริญพืชผล	บุเปือย	น้ำยืน	อุบลราชธานี	500000	
	30	จ3-9(6)-48/50อบ	ลานมันกรุงทอง 2	บุเปือย	น้ำยืน	อุบลราชธานี	31000000	
	31	จ3-9(6)-20/51อบ	รุ่งภิญโญ 2	บุเปือย	น้ำยืน	อุบลราชธานี	24000000	
	32	จ3-9(6)-18/50อบ	ห้างหุ้นส่วนสามัญนิติบุคคล	บุเปือย	น้ำยืน	อุบลราชธานี	1000000	
	33	จ3-9(6)-27/52อบ	ตั้งโชคดี	บุเปือย	น้ำยืน	อุบลราชธานี	12000000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
41	34	จ3-9(6)-1/46อบ	ชุมเจริญ	สีวิเชียร	น้ำยืน	อุบลราชธานี	500000	31.356
	35	2-9(6)-1/31อบ	สมชัย	สีวิเชียร	น้ำยืน	อุบลราชธานี	200000	
	36	จ3-9(6)-25/50อบ	สมชัย	สีวิเชียร	น้ำยืน	อุบลราชธานี	500000	
	37	จ3-9(6)-11/51อบ	ลานมันห้วยแก้ว	สีวิเชียร	น้ำยืน	อุบลราชธานี	24828000	
	38	จ3-9(6)-1/51อบ	ยิ่งเจริญ	สีวิเชียร	น้ำยืน	อุบลราชธานี	2000000	
	39	3-9(6)-9/31อบ	กรุงทองพืชผล	สีวิเชียร	น้ำยืน	อุบลราชธานี	500000	
	40	3-9(6)-13/35อบ	บุญเจริญ	สีวิเชียร	น้ำยืน	อุบลราชธานี	1000000	
42	41	จ3-9(6)-2/49อบ	ไชยมงคลพืชผล	ยางใหญ่	น้ำยืน	อุบลราชธานี	50000	0.850
	42	จ3-9(6)-6/53อบ	ประสงค์พืชผล	ยางใหญ่	น้ำยืน	อุบลราชธานี	750000	
43	43	จ3-9(6)-50/50อบ	เรืองศิริ	เก่าขาม	น้ำยืน	อุบลราชธานี	5000000	6.053
	44	จ3-9(6)-43/50อบ	สุขสมบูรณ์การเกษตร	เก่าขาม	น้ำยืน	อุบลราชธานี	700000	
44	45	จ3-9(6)-119/52อบ	เอกเจริญรุ่งเรือง	คอนสาย	ตระการพืชผล	อุบลราชธานี	400000	67.675
	46	จ3-9(6)-44/51อบ	บริษัท เนรมิตพืชผล จำกัด	คอนสาย	ตระการพืชผล	อุบลราชธานี	63280000	
	47	จ3-9(6)-16/51อบ	โชคประทีป	คอนสาย	ตระการพืชผล	อุบลราชธานี	50000	
45	48	จ3-9(6)-81/52อบ	ทรัพย์ไพบูลย์ เทรดดิ้ง กรุ๊ป	ห้วยฝ้ายพัฒนา	ตระการพืชผล	อุบลราชธานี	30000000	31.857
46	49	จ3-9(6)-7/31อบ	อุบลปอไทยพืชผล	แสนสุข	วารินชำราบ	อุบลราชธานี	500000	0.531
47	50	จ3-9(6)-41/52อบ	เฮงพาณิชย์	ดอนจิก	พิบูลมังสาหาร	อุบลราชธานี	75000000	79.643

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
48	51	จ3-9(6)-47/50อบ	บริษัท อุบลเกษตรพลังงาน จำกัด	นาดี	นาเยีย	อุบลราชธานี	20000000	21.238
49	52	จ3-9(6)-47/52อบ	สมบูรณ์พืชผล	ตาเกา	น้ำขุ่น	อุบลราชธานี	2000000	28.353
	53	จ3-9(6)-3/43อบ	ชัยเจริญพืชผล	ตาเกา	น้ำขุ่น	อุบลราชธานี	200000	
	54	จ3-9(6)-6/52อบ	โนนงามพืชผล	ตาเกา	น้ำขุ่น	อุบลราชธานี	12000000	
	55	จ3-9(6)-5/48อบ	สุชินพืชผล	ตาเกา	น้ำขุ่น	อุบลราชธานี	100000	
	56	จ3-9(6)-1/53อบ	ลานมันบุญโฮมพืชผล	ตาเกา	น้ำขุ่น	อุบลราชธานี	400000	
	57	จ3-9(6)-30/52อบ	สหกรณ์การเกษตรน้ำขุ่น จำกัด	ตาเกา	น้ำขุ่น	อุบลราชธานี	12000000	
50	58	จ3-9(6)-26/52อบ	ประกายรุ่งเรือง	ขี้เหล็ก	น้ำขุ่น	อุบลราชธานี	6000000	6.371
51	59	จ3-9(6)-11/52อบ	หน่อยพืชผล	บัวงาม	บุณฑริก	อุบลราชธานี	12000000	12.743
52	60	3-9(6)-47/31กส	ลานมันหนองสอ	บึงวิชัย	เมืองกาฬสินธุ์	กาฬสินธุ์	100000	0.106
53	61	3-9(6)-1/33กส	เจริญพร 2	คำใหญ่	ห้วยเม็ก	กาฬสินธุ์	200000	0.998
	62	จ3-9(6)-1/41กส	ศิริโพธิ์ทอง	คำใหญ่	ห้วยเม็ก	กาฬสินธุ์	500000	
	63	3-9(6)-2/27กส	ยิ่งเจริญ	คำใหญ่	ห้วยเม็ก	กาฬสินธุ์	40000	
	64	3-9(6)-4/30กส	เจริญพรพืชผล	คำใหญ่	ห้วยเม็ก	กาฬสินธุ์	200000	
54	65	จ3-9(6)-1/44กส	ส.เนินทอง	สหัสขันธ์	สหัสขันธ์	กาฬสินธุ์	200000	0.212
55	66	จ3-9(6)-4/52กส	สามมิตรพืชผล	โนนแหลมทอง	สหัสขันธ์	กาฬสินธุ์	600000	0.637
56	67	จ3-9(6)-1/42กส	วัชรดล	โนนบุรี	สหัสขันธ์	กาฬสินธุ์	100000	0.106

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
57	68	จ3-9(6)-57/31กส	นาย ทองสุข โคตรวงศ์	ทุ่งคลอง	คำม่วง	กาฬสินธุ์	210000	0.223
58	69	จ3-9(6)-3/35กส	ทวีทรัพย์	หนองกุงศรี	หนองกุงศรี	กาฬสินธุ์	200000	0.212
59	70	จ3-9(6)-16/54กส	ลานมันไทยสมบูรณ์	สมเด็จ	สมเด็จ	กาฬสินธุ์	200000	2.549
	71	จ3-9(6)-17/51กส	ลานมันศิริชัย	สมเด็จ	สมเด็จ	กาฬสินธุ์	600000	
	72	3-9(6)-115/31กส	ต.พืชผล	สมเด็จ	สมเด็จ	กาฬสินธุ์	100000	
	73	จ3-9(6)-1/46กส	ลานมันจ่าโย	สมเด็จ	สมเด็จ	กาฬสินธุ์	200000	
	74	3-9(6)-21/31กส	ทองคำพืชผล	สมเด็จ	สมเด็จ	กาฬสินธุ์	200000	
	75	จ3-9(6)-1/43กส	บำเพ็ญพืชผล	สมเด็จ	สมเด็จ	กาฬสินธุ์	1000000	
	76	3-9(6)-10/28กส	งี่ซงพาณิชย์	สมเด็จ	สมเด็จ	กาฬสินธุ์	100000	
60	77	จ3-9(6)-8/52กส	ลานมันตั้งพัฒนา	หนองแวง	สมเด็จ	กาฬสินธุ์	800000	2.761
	78	จ3-9(6)-12/51กส	โชคเจริญสมเด็จ	หนองแวง	สมเด็จ	กาฬสินธุ์	500000	
	79	จ3-9(6)-9/52กส	ลานมันงี่ซง	หนองแวง	สมเด็จ	กาฬสินธุ์	800000	
	80	จ3-9(6)-7/48กส	รุ่งโรจน์พัฒนา	หนองแวง	สมเด็จ	กาฬสินธุ์	500000	
61	81	3-9(6)-10/29กส	วิชาญมันเส้น	หมูม่น	สมเด็จ	กาฬสินธุ์	100000	0.106
62	82	จ3-9(6)-10/51กส	ลานมันง่วนเชียง	ศรีสมเด็จ	สมเด็จ	กาฬสินธุ์	800000	1.115
	83	จ3-9(6)-9/51กส	ลิ้มไทเฮง	ศรีสมเด็จ	สมเด็จ	กาฬสินธุ์	250000	
63	84	จ3-9(6)-36/51กส	ลานมันกาฬสินธุ์อินเตอร์	ลำห้วยหลัว	สมเด็จ	กาฬสินธุ์	500000	0.531

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
64	85	3-9(6)-1/31กส	คำบงพืชผล	คำบง	ห้วยผึ้ง	กาฬสินธุ์	300000	0.319
65	86	จ3-9(6)-2/41นพ	ลานมันเฮียตี๋	อุ่มเหม้า	ธาตุพนม	นครพนม	200000	0.319
	87	จ3-9(6)-1/41นพ	รัตนพืชผล	อุ่มเหม้า	ธาตุพนม	นครพนม	100000	
66	88	3-9(6)-6/30ชย	ห้างหุ้นส่วนจำกัด เล้าซุ่ยเซ้งพืชผล	รอบเมือง	เมืองชัยภูมิ	ชัยภูมิ	300000	0.319
67	89	3-9(6)-36/31ชย	วานนท์การเกษตร 2	บ้านเล่า	เมืองชัยภูมิ	ชัยภูมิ	300000	0.765
	90	3-9(6)-58/31ชย	อ.พืชผล	บ้านเล่า	เมืองชัยภูมิ	ชัยภูมิ	420000	
68	91	จ3-9(6)-6/50ชย	ส. พาณิชย์	ท่าหินโงม	เมืองชัยภูมิ	ชัยภูมิ	400000	0.829
	92	3-9(6)-48/31ชย	ลานมันบุญส่ง	ท่าหินโงม	เมืองชัยภูมิ	ชัยภูมิ	100000	
	93	จ3-9(6)-42/31ชย	บุญเรือง 2	ท่าหินโงม	เมืองชัยภูมิ	ชัยภูมิ	281000	
69	94	3-9(6)-74/31ชย	สหวานนท์	บ้านเขว้า	บ้านเขว้า	ชัยภูมิ	400000	2.761
	95	จ3-9(6)-11/54ชย	นาย สุพรม มานะหาญ	บ้านเขว้า	บ้านเขว้า	ชัยภูมิ	2000000	
	96	3-9(6)-11/22ชย	นายฮวด ตั้งสุริยานนท์	บ้านเขว้า	บ้านเขว้า	ชัยภูมิ	200000	
70	97	จ3-9(6)-7/50ชย	ตั้งพูนผล	ซับสีทอง	เมืองชัยภูมิ	ชัยภูมิ	400000	0.425
71	98	จ3-9(6)-98/52ชย	เตียเซ่งเฮง	บ้านกอก	จัตุรัส	ชัยภูมิ	1000000	1.062
72	99	จ3-9(6)-1/43ชย	นางหนูคูณ เอมดี	หนองบัวบาน	จัตุรัส	ชัยภูมิ	200000	0.319
	100	3-9(6)-9/31ชย	ทวีพืชผล	หนองบัวบาน	จัตุรัส	ชัยภูมิ	100000	
73	101	จ3-9(6)-116/52ชย	เอส.พี.เพื่อนเกษตร	บ้านขาม	จัตุรัส	ชัยภูมิ	1500000	1.593

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
74	102	3-9(6)-10/21ชย	เกียมกี่	หนองบัวโคก	จัตุรัส	ชัยภูมิ	60000	0.064
75	103	3-9(6)-1/24ชย	นายบุญช่วย สถิตย์ธรรม	ส้มป่อย	จัตุรัส	ชัยภูมิ	100000	0.106
76	104	จ3-9(6)-28/53ชย	วังทรัพย์การเกษตร	บ้านชวน	บำเหน็จณรงค์	ชัยภูมิ	100000	0.106
77	105	3-9(6)-19/21ชย	ซินเฮียงเซ้ง	บ้านเพชร	บำเหน็จณรงค์	ชัยภูมิ	150000	1.540
	106	3-9(6)-22/21ชย	ลานมันนายเกื้อ	บ้านเพชร	บำเหน็จณรงค์	ชัยภูมิ	300000	
	107	จ3-9(6)-1/44ชย	ก.รุ่งโรจน์	บ้านเพชร	บำเหน็จณรงค์	ชัยภูมิ	1000000	
78	108	จ3-9(6)-43/51ชย	วรจิต	โคกเริงรมย์	บำเหน็จณรงค์	ชัยภูมิ	10000000	10.619
79	109	3-9(6)-59/31ชย	กิจรุ่งเรือง	หนองบัวระเหว	หนองบัวระเหว	ชัยภูมิ	400000	1.858
	110	จ3-9(6)-1/40ชย	นายอนุรักษ์ กิจเลิศนิรันดร์	หนองบัวระเหว	หนองบัวระเหว	ชัยภูมิ	1000000	
	111	3-9(6)-3/34ชย	ชูมิตร	หนองบัวระเหว	หนองบัวระเหว	ชัยภูมิ	300000	
	112	จ3-9(6)-3/36ชย	นายบุญชู สถิตย์ธรรม	หนองบัวระเหว	หนองบัวระเหว	ชัยภูมิ	50000	
80	113	จ2-9(6)-1/41ชย	ก.รุ่งโรจน์	วังตะเฆ่	หนองบัวระเหว	ชัยภูมิ	590000	0.627
81	114	3-9(6)-12/31ชย	ชนะพืชผล	วะตะแบก	เทพสถิต	ชัยภูมิ	200000	2.761
	115	3-9(6)-2/35ชย	นายสนั่น โคศิลา	วะตะแบก	เทพสถิต	ชัยภูมิ	100000	
	116	จ3-9(6)-9/50ชย	นายสุชาติ ทองชำนาญ	วะตะแบก	เทพสถิต	ชัยภูมิ	1000000	
	117	3-9(6)-30/31ชย	นายณรงค์ศักดิ์ ทองชำนาญ	วะตะแบก	เทพสถิต	ชัยภูมิ	300000	
	118	จ3-9(6)-8/50ชย	ลานมันเหรียญชัย	วะตะแบก	เทพสถิต	ชัยภูมิ	1000000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
82	119	3-9(6)-10/34ชย	ลานมันสมเกียรติ์	ห้วยยายจิ๋ว	เทพสถิต	ชัยภูมิ	300000	1.593
	120	จ3-9(6)-1/42ชย	ก.พาณิชย์	ห้วยยายจิ๋ว	เทพสถิต	ชัยภูมิ	400000	
	121	3-9(6)-17/31ชย	ตองเอ	ห้วยยายจิ๋ว	เทพสถิต	ชัยภูมิ	200000	
	122	จ3-9(6)-10/50ชย	ตั้งเฮง	ห้วยยายจิ๋ว	เทพสถิต	ชัยภูมิ	600000	
83	123	จ3-9(6)-2/46ชย	บ้านนาพืชผล	นายางกลัก	เทพสถิต	ชัยภูมิ	1000000	1.168
	124	จ3-9(6)-47/49ชย	นายพิสิษฐ์ภูธน เชิญสันติกุล	นายางกลัก	เทพสถิต	ชัยภูมิ	100000	
84	125	จ3-9(6)-4/36ชย	ถวิลพืชผล	บ้านไร่	เทพสถิต	ชัยภูมิ	20000	0.021
85	126	จ3-9(6)-27/48ชย	สมชายพืชผล	โป่งนก	เทพสถิต	ชัยภูมิ	1000000	1.529
	127	3-9(6)-2/34ชย	ศิลาทองพืชผล	โป่งนก	เทพสถิต	ชัยภูมิ	200000	
	128	จ2-9(6)-4/49ชย	นางสนาม นามจันดา	โป่งนก	เทพสถิต	ชัยภูมิ	100000	
	129	จ3-9(6)-4/46ชย	ลานมันป่าอ้อย	โป่งนก	เทพสถิต	ชัยภูมิ	90000	
	130	จ3-9(6)-1/37ชย	นางสาวอัญชลี พืดขุนทด	โป่งนก	เทพสถิต	ชัยภูมิ	50000	
86	131	3-9(6)-5/31ชย	อุทัยพืชผล	โคกสะอาด	ภูเขียว	ชัยภูมิ	400000	0.425
87	132	จ3-9(6)-46/47ชย	ห้างหุ้นส่วนจำกัด เลียงเจริญการเกษตร	ช่องสามหมอ	แก้งคร้อ	ชัยภูมิ	5000000	6.371
	133	3-9(6)-60/31ชย	เลียงเฮงพืชผล	ช่องสามหมอ	แก้งคร้อ	ชัยภูมิ	1000000	
88	134	จ3-9(6)-83/52ชย	กำนันประคองพืชพล	วังทอง	ภักดีชุมพล	ชัยภูมิ	500000	11.150
	135	จ3-9(6)-42/52ชย	ผาทองพืชผล	วังทอง	ภักดีชุมพล	ชัยภูมิ	10000000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
89	136	3-9(6)-25/31ชย	ไทยพืชผล	หนองฉิม	เนินสง่า	ชัยภูมิ	200000	0.212
90	137	3-9(6)-6/31ชย	ส.พัฒนา	ซับใหญ่*	ซับใหญ่	ชัยภูมิ	300000	1.119
	138	จ3-9(6)-1/41ชย	สิทธิชัยพืชผล	ซับใหญ่	ซับใหญ่	ชัยภูมิ	150000	
	139	3-9(6)-33/31ชย	พูลผล	ซับใหญ่	ซับใหญ่	ชัยภูมิ	200000	
	140	3-9(6)-9/34ชย	ลานมันบ้านโนนสะอาด	ซับใหญ่	ซับใหญ่	ชัยภูมิ	204000	
	141	3-9(6)-16/31ชย	วรศักดิ์พืชผล	ซับใหญ่	ซับใหญ่	ชัยภูมิ	200000	
91	142	2-9(6)-23/20ขก	เรืองอารีย์	ท่าพระ	เมืองขอนแก่น	ขอนแก่น	100000	0.425
	143	2-9(6)-25/20ขก	สหะผล	ท่าพระ	เมืองขอนแก่น	ขอนแก่น	300000	
92	144	2-9(6)-104/21ขก	เสริมสุขพืชผล	น้ำพอง	น้ำพอง	ขอนแก่น	400000	0.956
	145	3-9(6)-3/32ขก	ชัยฤทธิ์	น้ำพอง	น้ำพอง	ขอนแก่น	500000	
93	146	จ2-9(6)-1/40ขก	นางพิกุล กิตติวราพล	หนองกุง	น้ำพอง	ขอนแก่น	100000	2.283
	147	จ2-9(6)-1/42ขก	ชัยวัฒน์ คำแก่นคูณ นาย	หนองกุง	น้ำพอง	ขอนแก่น	2000000	
	148	3-9(6)-48/31ขก	นายวิชิต ตราชู	หนองกุง	น้ำพอง	ขอนแก่น	50000	
94	149	2-9(6)-53/19ขก	เจียบฮวด	ม่วงหวาน	น้ำพอง	ขอนแก่น	300000	0.319
95	150	2-9(6)-7/34ขก	นายประสิทธิ์ สมบัติมาก	บ้านขาม	น้ำพอง	ขอนแก่น	50000	0.053

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
96	151	จ2-9(6)-7/49ขก	โรงมันทวีกิจ	หนองโก	กระนวน	ขอนแก่น	100000	1.380
	152	3-9(6)-70/31ขก	บ้านไร่	หนองโก	กระนวน	ขอนแก่น	200000	
	153	จ3-9(6)-2/43ขก	วรวุฒิพืชผล	หนองโก	กระนวน	ขอนแก่น	1000000	
97	154	3-9(6)-54/31ขก	บริษัท แก่นเจริญ จำกัด	ห้วยยาง	กระนวน	ขอนแก่น	1000000	1.062
98	155	3-9(6)-20/22ขก	เซ่งหลีฮวดพืชผล	บ้านฝาง	กระนวน	ขอนแก่น	24000	0.025
99	156	2-9(6)-9/21ขก	ห้างหุ้นส่วนจำกัด บ้านไผ่ไทยแสงพืชผล	บ้านไผ่	บ้านไผ่	ขอนแก่น	500000	0.903
	157	จ2-9(6)-1/41ขก	นายสงวนศักดิ์ โชควัฒน์วิกุล	บ้านไผ่	บ้านไผ่	ขอนแก่น	200000	
	158	2-9(6)-44/19ขก	ห้างหุ้นส่วนจำกัด บ้านไผ่อุทัยพืชผล	บ้านไผ่	บ้านไผ่	ขอนแก่น	150000	
100	159	3-9(6)-7/19ขก	ชาญศิริพืชผล	ในเมือง	บ้านไผ่	ขอนแก่น	240000	0.255
101	160	จ3-9(6)-1/40ขก	เจริญผล	บ้านลาน	บ้านไผ่	ขอนแก่น	50000	0.053
102	161	2-9(6)-4/25ขก	ห้างหุ้นส่วนจำกัด บ้านไผ่รุ่งเรือง	แคนเหนือ	บ้านไผ่	ขอนแก่น	100000	3.069
	162	จ2-9(6)-2/54ขก	ลานมันสหเจริญ	แคนเหนือ	บ้านไผ่	ขอนแก่น	600000	
	163	จ3-9(6)-78/52ขก	ห้างหุ้นส่วนจำกัด อื้อจี่กี่ก่อสร้าง	แคนเหนือ	บ้านไผ่	ขอนแก่น	2000000	
	164	3-9(6)-9/19ขก	นายวิเชษฐ์ วิทยาวรากุล	แคนเหนือ	บ้านไผ่	ขอนแก่น	190000	
103	165	จ2-9(6)-1/47ขก	เกษตรสิน	หินตั้ง	บ้านไผ่	ขอนแก่น	500000	0.690
	166	จ3-9(6)-1/43ขก	นายสมยศ ภควัตศิตานนท์	หินตั้ง	บ้านไผ่	ขอนแก่น	150000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
104	168	จ2-9(6)-1/45ขก	บุญสิริ สุรางค์ศรีรัตน์ นาง	หัวหนอง	บ้านไผ่	ขอนแก่น	7000000	7.964
	167	ข2-9(6)-1/45ขก	ลานมันอุทัย	หัวหนอง	บ้านไผ่	ขอนแก่น	500000	
105	169	2-9(6)-9/20ขก	ดิ้งเซ้ง	โจดหนองแก	พล	ขอนแก่น	500000	0.531
106	170	จ2-9(6)-3/51ขก	สีไทยการค้า	กุดเค้า	มัญจาคีรี	ขอนแก่น	500000	1.009
	171	3-9(6)-6/23ขก	ศรีไทย	กุดเค้า	มัญจาคีรี	ขอนแก่น	450000	
107	172	จ2-9(6)-8/49ขก	ศิริพืชผล	เขาสวนกวาง	เขาสวนกวาง	ขอนแก่น	500000	1.062
	173	จ2-9(6)-6/49ขก	ปรีดาพืชผล	เขาสวนกวาง	เขาสวนกวาง	ขอนแก่น	500000	
108	174	จ2-9(6)-5/41ขก	โชคประเสริฐ	คำม่วง	เขาสวนกวาง	ขอนแก่น	2000000	2.124
109	175	จ2-9(6)-1/51ขก	ลานพัฒนา	กระนวน	ซำสูง	ขอนแก่น	200000	0.212
110	176	3-9(6)-8/29ขก	นายภิญโญ ไทยทองสุขสกุล	โนนศิลา	โนนศิลา	ขอนแก่น	100000	0.106
111	177	จ3-9(6)-6/37นม	ห้างหุ้นส่วนจำกัด ส่องแสงพืชผล	หัวทะเล	เมืองนครราชสีมา	นครราชสีมา	500000	4.035
	178	2-9(6)-146/21นม	โรงสีวัฒนานครราชสีมา จำกัด	หัวทะเล	เมืองนครราชสีมา	นครราชสีมา	3000000	
	179	3-9(6)-8/31นม	สงวนวงษ์อุตสาหกรรม จำกัด	หัวทะเล	เมืองนครราชสีมา	นครราชสีมา	100000	
	180	2-9(6)-176/31นม	เลี่ยงเฮง	หัวทะเล	เมืองนครราชสีมา	นครราชสีมา	200000	
112	181	จ3-9(6)-5/44นม	บริษัท ทิพยทรัพย์ จำกัด	ไชยมงคล	เมืองนครราชสีมา	นครราชสีมา	12400000	13.168

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
113	182	จ2-9(6)-9/36นม	แชโชคชัย	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	11500000	23.171
	183	3-9(6)-2/18นม	ห้างหุ้นส่วนจำกัด เค.ซี. พืชผล	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	300000	
	184	จ2-9(6)-1/40นม	นายเสริม อักษรพิมพ์	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	6000000	
	185	2-9(6)-182/31นม	รัฐพลพืชผล	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	2160000	
	186	2-9(6)-192/31นม	บริษัท แป้งตะวันออก (1987) จำกัด	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	100000	
	187	2-9(6)-25/29นม	โรงงานแป้งมันอุดมชัย,บจก.	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	1760000	
114	188	จ3-9(6)-3/36นม	นางสาวมลฤดี ชัยฤทธิ์	ในเมือง	เมืองนครราชสีมา	นครราชสีมา	200000	0.212
115	189	จ3-9(6)-3/40นม	สุขขาวพงศ์พืชผล	แชะ	ครบุรี	นครราชสีมา	100000	0.319
	190	จ3-9(6)-1/41นม	ป.เลื่อนเจริญพืชผล	แชะ	ครบุรี	นครราชสีมา	200000	
116	191	จ3-9(6)-42/47นม	ลานมันวีระศักดิ์พืชผล	เฉลียง	ครบุรี	นครราชสีมา	100000	2.708
	192	จ3-9(6)-10/42นม	กำพลพืชผล 2	เฉลียง	ครบุรี	นครราชสีมา	800000	
	193	จ3-9(6)-7/43นม	ส.เจริญเกษตร	เฉลียง	ครบุรี	นครราชสีมา	400000	
	194	3-9(6)-26/29นม	กำพลพืชผล	เฉลียง	ครบุรี	นครราชสีมา	200000	
	195	จ3-9(6)-8/43นม	เฮงเจริญพืชผล	เฉลียง	ครบุรี	นครราชสีมา	100000	
	196	จ3-9(6)-13/43นม	สุดาวรรณพืชผล	เฉลียง	ครบุรี	นครราชสีมา	750000	
	197	3-9(6)-4/35นม	ลานมันยินดี	เฉลียง	ครบุรี	นครราชสีมา	200000	
117	198	จ3-9(6)-5/43นม	ตั้งไท้เส็งครบุรี	อรพิมพ์	ครบุรี	นครราชสีมา	1000000	1.062

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
118	199	3-9(6)-31/33นม	ลานมันแม่งเส็ง	บ้านใหม่	ครบุรี	นครราชสีมา	3000000	6.371
	200	จ3-9(6)-19/43นม	ลานมันย่งศิลป์	บ้านใหม่	ครบุรี	นครราชสีมา	2000000	
	201	3-9(6)-130/31นม	สยามพืชไร่	บ้านใหม่	ครบุรี	นครราชสีมา	1000000	
119	202	จ3-9(6)-2/46นม	ศรีสมพร	ลำเพียก	ครบุรี	นครราชสีมา	100000	0.212
	203	3-9(6)-121/31นม	นำชัยพืชผล	ลำเพียก	ครบุรี	นครราชสีมา	100000	
120	204	จ3-9(6)-5/53นม	นางสาวจุฑารัตน์ บัณฑิตสาธิสรรค์	ตะแบกบาน	ครบุรี	นครราชสีมา	800000	0.850
121	205	จ3-9(6)-1/42นม	พิริยะพงศ์	เสิงสาง	เสิงสาง	นครราชสีมา	100000	0.637
	206	จ3-9(6)-15/43นม	ดิเรกพืชผล	เสิงสาง	เสิงสาง	นครราชสีมา	500000	
122	207	จ3-9(6)-74/52นม	ลานมันกิตติคุณพืชผล	สระตะเคียน	เสิงสาง	นครราชสีมา	100000	1.434
	208	จ3-9(6)-16/47นม	ลานมันศรีสมพร	สระตะเคียน	เสิงสาง	นครราชสีมา	500000	
	209	จ3-9(6)-22/50นม	วิจัยพืชผล	สระตะเคียน	เสิงสาง	นครราชสีมา	50000	
	210	จ3-9(6)-103/51นม	ลานมันแสงชัยพืชผล	สระตะเคียน	เสิงสาง	นครราชสีมา	300000	
	211	จ3-9(6)-21/47นม	ธีรพงษ์พืชผล 3	สระตะเคียน	เสิงสาง	นครราชสีมา	100000	
	212	จ3-9(6)-12/47นม	นายธนโชค จารุทวัย	สระตะเคียน	เสิงสาง	นครราชสีมา	300000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
123	213	3-9(6)-1/34นม	วิจิตรชัยพืชผล	กุดโบสถ์	เสิงสาง	นครราชสีมา	800000	2.124
	214	จ3-9(6)-6/48นม	เอกชัยพืชผล	กุดโบสถ์	เสิงสาง	นครราชสีมา	200000	
	215	จ3-9(6)-1/40นม	สงวนศักดิ์พืชผล	กุดโบสถ์	เสิงสาง	นครราชสีมา	100000	
	216	จ3-9(6)-29/50นม	สงวนศักดิ์พืชผล	กุดโบสถ์	เสิงสาง	นครราชสีมา	100000	
	217	จ3-9(6)-30/47นม	ลานนาวีพืชผล	กุดโบสถ์	เสิงสาง	นครราชสีมา	500000	
	218	จ3-9(6)-42/53นม	เอกชัยพืชผล	กุดโบสถ์	เสิงสาง	นครราชสีมา	200000	
	219	3-9(6)-105/31นม	สามชัยพืชผล	กุดโบสถ์	เสิงสาง	นครราชสีมา	100000	
124	220	จ3-9(6)-1/36นม	ลานวัฒนานันท์	สุขไพบูลย์	เสิงสาง	นครราชสีมา	200000	1.380
	221	จ3-9(6)-1/43นม	ลานมันภู่เจริญ (เสิงสาง)"	สุขไพบูลย์	เสิงสาง	นครราชสีมา	200000	
	222	จ3-9(6)-4/44นม	ลานมันกิตติคุณพืชผล	สุขไพบูลย์	เสิงสาง	นครราชสีมา	400000	
	223	จ3-9(6)-2/41นม	รวมทรัพย์	สุขไพบูลย์	เสิงสาง	นครราชสีมา	500000	
125	224	3-9(6)-170/31นม	กิตติภัณฑ์	ศรีละกอ	จักราช	นครราชสีมา	100000	0.106
126	225	3-9(6)-9/31นม	โรงงานอัดมันเม็ดแสงอยุธยา จำกัด	กระโทก	โชคชัย	นครราชสีมา	300000	0.319
127	226	จ3-9(6)-18/48นม	ทวีทรัพย์	ทุ่งอรุณ	โชคชัย	นครราชสีมา	100000	0.212
	227	จ3-9(6)-2/42นม	วาทีทองพืชผล	ทุ่งอรุณ	โชคชัย	นครราชสีมา	100000	
128	228	จ3-9(6)-10/36นม	มีโชคพืชผล	โชคชัย	โชคชัย	นครราชสีมา	500000	0.637
	229	3-9(6)-30/21นม	โรงงานมันกิจชัยสิต	โชคชัย	โชคชัย	นครราชสีมา	100000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
129	230	จ3-9(6)-5/54นม	วี.ที.เอ็น.พืชผล	ด่านเกวียน	โชคชัย	นครราชสีมา	900000	2.867
	231	จ3-9(6)-6/54นม	นายนริช เจริญวัฒนาเสถียร	ด่านเกวียน	โชคชัย	นครราชสีมา	900000	
	232	จ3-9(6)-12/53นม	นายสายัณห์ นันทศักดิ์ศิริ	ด่านเกวียน	โชคชัย	นครราชสีมา	900000	
130	233	จ3-9(6)-6/43นม	ยะหลี	ด่านขุนทด	ด่านขุนทด	นครราชสีมา	90000	1.848
	234	3-9(6)-36/21นม	ลิ้มเจริญ	ด่านขุนทด	ด่านขุนทด	นครราชสีมา	500000	
	235	3-9(6)-20/28นม	พ.พืชผล	ด่านขุนทด	ด่านขุนทด	นครราชสีมา	200000	
	236	จ3-9(6)-15/50นม	ยะหลีพืชผล	ด่านขุนทด	ด่านขุนทด	นครราชสีมา	100000	
	237	จ3-9(6)-11/43นม	บริษัท ที วาย เอ็น ทรานสปอร์ต จำกัด	ด่านขุนทด	ด่านขุนทด	นครราชสีมา	500000	
	238	3-9(6)-27/30นม	นางสมัย ตรงจิตกุศล	ด่านขุนทด	ด่านขุนทด	นครราชสีมา	100000	
	239	3-9(6)-19/28นม	แสงชัยพืชผล	ด่านขุนทด	ด่านขุนทด	นครราชสีมา	250000	
131	240	จ3-9(6)-24/49นม	ชัยยงพืชผล	ตะเคียน	ด่านขุนทด	นครราชสีมา	1000000	1.062
132	241	จ3-9(6)-4/36นม	ธ.พืชผล	บ้านเก่า	ด่านขุนทด	นครราชสีมา	200000	0.212
133	242	จ3-9(6)-2/44นม	ศิริรสพืชผล	พันชนะ	ด่านขุนทด	นครราชสีมา	300000	4.035
	243	จ3-9(6)-69/52นม	ชัยรุ่งเรือง 2	พันชนะ	ด่านขุนทด	นครราชสีมา	3000000	
	244	จ3-9(6)-7/46นม	นายสุภัทร ถิราพิภพ	พันชนะ	ด่านขุนทด	นครราชสีมา	500000	
134	245	3-9(6)-12/21นม	ยงเจริญพืชผล	หินดาด	ด่านขุนทด	นครราชสีมา	100000	0.212
	246	จ3-9(6)-39/51นม	พูนเจริญพืชผล	หินดาด	ด่านขุนทด	นครราชสีมา	100000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
135	247	3-9(6)-10/34นม	ไทยแสงทอง	ห้วยบง	ด่านขุนทด	นครราชสีมา	300000	3.159
	248	จ3-9(6)-9/43นม	นายจิรวัฒน์ ฐิติธนวัฒน์	ห้วยบง	ด่านขุนทด	นครราชสีมา	520000	
	249	จ3-9(6)-8/36นม	ลิ้มซังพืชผล	ห้วยบง	ด่านขุนทด	นครราชสีมา	100000	
	250	3-9(6)-54/31นม	ช.พืชผล	ห้วยบง	ด่านขุนทด	นครราชสีมา	500000	
	251	จ2-9(6)-4/43นม	ยิ่งเฮงพืชผล	ห้วยบง	ด่านขุนทด	นครราชสีมา	1255000	
	252	จ3-9(6)-5/42นม	นายเสกสรรค์ สิทธิวีระกุล	ห้วยบง	ด่านขุนทด	นครราชสีมา	300000	
136	253	จ3-9(6)-7/44นม	วีระพันธ์พืชผล	ด่านใน	ด่านขุนทด	นครราชสีมา	800000	0.850
137	254	2-9(6)-130/20นม	ตั้งจุ้นฮวด	ด่านช้าง	บัวใหญ่	นครราชสีมา	50000	7.486
	255	จ2-9(6)-5/43นม	ชัยวิวัฒน์	ด่านช้าง	บัวใหญ่	นครราชสีมา	6800000	
	256	2-9(6)-136/20นม	รวมทรัพย์	ด่านช้าง	บัวใหญ่	นครราชสีมา	100000	
	257	จ3-9(6)-33/50นม	ห้างหุ้นส่วนจำกัด เตียเอี่ยมฮวดพืชผล	ด่านช้าง	บัวใหญ่	นครราชสีมา	100000	
138	258	จ2-9(6)-1/36นม	น.ส.สมหมาย โจวเจริญ	นกออก	ปักธงชัย	นครราชสีมา	980000	1.572
	259	จ2-9(6)-1/49นม	นายศิวัชริศ ตั้งสุจริต	นกออก	ปักธงชัย	นครราชสีมา	100000	
	260	3-9(6)-1/35นม	เนี้ยวบริการ	นกออก	ปักธงชัย	นครราชสีมา	200000	
	261	3-9(6)-68/21นม	ก.ทวีชัยพืชผล	นกออก	ปักธงชัย	นครราชสีมา	200000	
139	262	3-9(6)-144/21นม	ดำรงชัย	งิ้ว	ปักธงชัย	นครราชสีมา	100000	1.168
	263	3-9(6)-93/22นม	ดำเนินชัย	งิ้ว	ปักธงชัย	นครราชสีมา	10000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
140	264	จ2-9(6)-1/45นม	ห้างหุ้นส่วนจำกัด เจริญภัณฑ์พืชผล	ในเมือง	พิมาย	นครราชสีมา	500000	0.743
	265	3-9(6)-18/30นม	ห้างหุ้นส่วนจำกัด สหชัยพิมาย	ในเมือง	พิมาย	นครราชสีมา	200000	
141	266	จ3-9(6)-1/49นม	เจริญผล	ทับสวาย	ห้วยแถลง	นครราชสีมา	400000	0.425
142	267	จ3-9(6)-1/46นม	ไร่พิกุลทอง	ชุมพวง	ชุมพวง	นครราชสีมา	500000	0.531
143	268	จ3-9(6)-4/41นม	นายธีระชัย เกรียงไกรเวคิน	มะเกลือเก่า	สูงเนิน	นครราชสีมา	300000	0.319
144	269	จ2-9(6)-3/43นม	ไทยณรงค์	นากลาง	สูงเนิน	นครราชสีมา	540000	0.573
145	270	จ3-9(6)-3/47นม	ลานมันน้ำฝนการเกษตร	ขามทะเลสอ	ขามทะเลสอ	นครราชสีมา	50000	0.053
146	271	3-9(6)-4/33นม	สมศักดิ์พืชผล	โป่งแดง	ขามทะเลสอ	นครราชสีมา	200000	1.274
	272	จ3-9(6)-79/52นม	เชียงโชคพืชผล	โป่งแดง	ขามทะเลสอ	นครราชสีมา	1000000	
147	273	3-9(6)-148/31นม	พรประเสริฐ	หนองสรวง	ขามทะเลสอ	นครราชสีมา	100000	0.106

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
148	274	3-9(6)-26/31นม	นายสันติ แป๊ะหลี	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	300000	3.292
	275	3-9(6)-179/31นม	รัชนีพืชผล	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	100000	
	276	จ3-9(6)-8/48นม	บริษัท เอี่ยมชัญชร จำกัด	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	1000000	
	277	3-9(6)-3/29นม	เจริญชัย	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	200000	
	278	จ3-9(6)-3/44นม	ลานมันเอี่ยมเฮง	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	200000	
	279	จ3-9(6)-38/50นม	บิ๊กเฮง	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	500000	
	280	จ3-9(6)-3/41นม	พรประเสริฐ 2	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	200000	
	281	3-9(6)-51/22นม	อู่ประเสริฐผล	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	100000	
	282	จ3-9(6)-4/37นม	ลานมันอุทัย	บึงอ้อ	ขามทะเลสอ	นครราชสีมา	500000	
149	283	3-9(6)-33/31นม	บริษัท ที.เอช.แสงตะวัน จำกัด	สีดิ้ว	สีดิ้ว	นครราชสีมา	500000	17.628
	284	2-9(6)-22/31นม	อ.พืชผล	สีดิ้ว	สีดิ้ว	นครราชสีมา	1800000	
	285	3-9(6)-1/27นม	ห้างหุ้นส่วนจำกัด วิชัยพืชผลสีดิ้ว	สีดิ้ว	สีดิ้ว	นครราชสีมา	13150000	
	286	3-9(6)-69/22นม	ลานมันทวีผล	สีดิ้ว	สีดิ้ว	นครราชสีมา	350000	
	287	2-9(6)-185/31นม	วิชัย	สีดิ้ว	สีดิ้ว	นครราชสีมา	800000	
150	288	3-9(6)-12/28นม	ธีระพืชผล	บ้านหัน	สีดิ้ว	นครราชสีมา	2200000	2.336

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
151	289	จ3-9(6)-37/47นม	เอส.พี.เกษตรผล	กฤษณา	สีคิ้ว	นครราชสีมา	500000	3.321
	290	จ3-9(6)-1/38นม	นายสันติ ศุภโชคกิจมงคล	กฤษณา	สีคิ้ว	นครราชสีมา	100000	
	291	2-9(6)-9/22นม	แสงจันทร์	กฤษณา	สีคิ้ว	นครราชสีมา	300000	
	292	2-9(6)-5/16นม	บุญเจือพืชผล	กฤษณา	สีคิ้ว	นครราชสีมา	2227000	
152	293	จ3-9(6)-3/43นม	เสน่ห์พืชผล	ลาดบัวขาว	สีคิ้ว	นครราชสีมา	400000	0.425
153	294	3-9(6)-109/22นม	พรชัยสีคิ้ว	กุดน้อย	สีคิ้ว	นครราชสีมา	150000	0.478
	295	3-9(6)-12/31นม	พรเจริญ	กุดน้อย	สีคิ้ว	นครราชสีมา	300000	
154	296	จ3-9(6)-14/43นม	บริษัท กิจเจริญสีคิ้วพืชผล จำกัด	มิตรภาพ	สีคิ้ว	นครราชสีมา	500000	0.637
	297	3-9(6)-22/28นม	วิกรมพืชผล	มิตรภาพ	สีคิ้ว	นครราชสีมา	100000	
155	298	3-9(6)-13/29นม	สิทธิพงษ์พืชผล	หนองบัวน้อย	สีคิ้ว	นครราชสีมา	100000	0.743
	299	3-9(6)-63/31นม	นายสันติ ศุภโชคกิจมงคล	หนองบัวน้อย	สีคิ้ว	นครราชสีมา	100000	
	300	3-9(6)-12/29นม	เฮงฮวดหลี	หนองบัวน้อย	สีคิ้ว	นครราชสีมา	500000	
156	301	3-9(6)-20/33นม	ลานมัน นี้ฮง 2	หนองสาหร่าย	ปากช่อง	นครราชสีมา	500000	3.982
	302	จ3-9(6)-2/40นม	ตรงพาณิชย์	หนองสาหร่าย	ปากช่อง	นครราชสีมา	3000000	
	303	3-9(6)-82/31นม	แสนสุข-สุขแสน	หนองสาหร่าย	ปากช่อง	นครราชสีมา	50000	
	304	3-9(6)-143/31นม	ทรายทอง	หนองสาหร่าย	ปากช่อง	นครราชสีมา	200000	
157	305	3-9(6)-92/31นม	พงศ์ศักดิ์พานิช	ขนงพระ	ปากช่อง	นครราชสีมา	300000	0.319

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
158	306	จ3-9(6)-45/50นม	ธงชัยพืชผล	หนองน้ำแดง	ปากช่อง	นครราชสีมา	1000000	1.062
159	307	3-9(6)-102/19นม	นายประจักษ์ อภิเมธีธำรง	หนองบุนนาก	หนองบุญมาก	นครราชสีมา	100000	0.212
	308	3-9(6)-99/21นม	นายทวีชัย คมวีระวงศ์	หนองบุนนาก	หนองบุญมาก	นครราชสีมา	100000	
160	309	3-9(6)-50/21นม	นำเจริญพืชผล	ไทยเจริญ	หนองบุญมาก	นครราชสีมา	100000	0.212
	310	3-9(6)-77/22นม	พรพูลทรัพย์	ไทยเจริญ	หนองบุญมาก	นครราชสีมา	100000	
161	311	2-9(6)-20/25นม	มิตรชัยพืชผล	หนองหัวแรต	หนองบุญมาก	นครราชสีมา	6175000	8.681
	312	3-9(6)-12/35นม	ลานมันสมพงษ์	หนองหัวแรต	หนองบุญมาก	นครราชสีมา	2000000	
162	313	จ3-9(6)-1/45นม	บริษัท ทาพีโอก้า ฟาร์ม จำกัด	แหลมทอง	หนองบุญมาก	นครราชสีมา	1000000	1.062
163	314	จ3-9(6)-5/46นม	ลานมันทวีวัฒน์	หนองไม้ไผ่	หนองบุญมาก	นครราชสีมา	300000	0.319
164	315	จ2-9(6)-6/43นม	ลี้เซ่งฮวด	แก้งสนามนาง	แก้งสนามนาง	นครราชสีมา	1230000	1.306
165	316	3-9(6)-175/31นม	นายเชิดศักดิ์ ชัยรัตนพิพัฒน์	สำนักตะคร้อ	เทพารักษ์	นครราชสีมา	40000	0.042
166	317	จ3-9(6)-7/35นม	ไทยเจริญ	หนองหอย	พระทองคำ	นครราชสีมา	300000	0.212
167	318	3-9(6)-25/31นม	อึ้งลักเส็ง 2	พังเทียม	พระทองคำ	นครราชสีมา	200000	0.212
168	319	จ3-9(6)-20/47นม	นายสุวิทย์ กาญจนกุล	ขุย	ลำทะเมนชัย	นครราชสีมา	500000	0.531
169	320	จ3-9(6)-6/46นม	โชคชัชวาลการเกษตร	ช่องแมว	ลำทะเมนชัย	นครราชสีมา	50000	0.053
170	321	3-9(6)-64/19นม	ทวีวัฒน์	ท่าช้าง	เฉลิมพระเกียรติ	นครราชสีมา	100000	0.106
171	322	จ2-9(6)-1/31บร	ชิวพืชผล	ดูเมือง	ดูเมือง	บุรีรัมย์	120000	0.127

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
172	323	จ2-9(6)-4/25บร	ไทยวิจารณ์พืชผล	สะเดา	นางรอง	บุรีรัมย์	500000	0.531
173	324	3-9(6)-121/31บร	ทวีผล	หนองกี่	หนองกี่	บุรีรัมย์	50000	0.053
174	325	จ3-9(6)-3/37บร	วิจิตรรัตน์	ดอนอะราง	หนองกี่	บุรีรัมย์	1000000	1.104
	326	3-9(6)-52/31บร	อมรรัตน์พืชผล	ดอนอะราง	หนองกี่	บุรีรัมย์	40000	
175	327	จ3-9(6)-17/47บร	สกุลทองพืชผล	ทุ่งกระตาดพัฒนา	หนองกี่	บุรีรัมย์	450000	1.561
	328	จ3-9(6)-25/51บร	พรทิพย์	ทุ่งกระตาดพัฒนา	หนองกี่	บุรีรัมย์	1000000	
	329	3-9(6)-18/31บร	เอื้อมพร	ทุ่งกระตาดพัฒนา	หนองกี่	บุรีรัมย์	20000	
176	330	จ3-9(6)-27/50บร	เอื้อมพร	ทุ่งกระเต็น	หนองกี่	บุรีรัมย์	2400000	2.549
177	331	3-9(6)-5/32บร	วีระศักดิ์พืชไร่	สำโรงใหม่	ละหานทราย	บุรีรัมย์	50000	0.053
178	332	จ3-9(6)-4/53บร	ลิ้มเฮงการเกษตร	หนองแวง	ละหานทราย	บุรีรัมย์	100000	0.106
179	333	จ3-9(6)-48/47บร	เจริญรัตน์พืชผล	หนองไม้งาม	บ้านกรวด	บุรีรัมย์	200000	0.212
180	334	จ3-9(6)-1/37บร	ภุชงค์พูนผล	สตึก	สตึก	บุรีรัมย์	200000	0.212
181	335	จ2-9(6)-3/20บร	สยามพูนผล	นิคม	สตึก	บุรีรัมย์	100000	0.106
182	336	จ3-9(6)-15/51บร	บริษัท ข้าวหอมปุ๋ยธรรมชาติ จำกัด	โคกมะม่วง	ปะคำ	บุรีรัมย์	200000	0.850
	337	3-9(6)-1/34บร	มังกรทอง	โคกมะม่วง	ปะคำ	บุรีรัมย์	100000	
	338	จ3-9(6)-1/39บร	เอกเจริญพืชผล	โคกมะม่วง	ปะคำ	บุรีรัมย์	500000	
183	339	จ2-9(6)-4/51บร	สวัสดีพืชผล	ทุ่งจังหัน	โนนสุวรรณ	บุรีรัมย์	1000000	1.062

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
184	340	3-9(6)-13/31บร	ทรายทองพืชผล	โนนดินแดง	โนนดินแดง	บุรีรัมย์	50000	1.646
	341	จ3-9(6)-28/47บร	นายชั้น ถมฉิมพลี	โนนดินแดง	โนนดินแดง	บุรีรัมย์	300000	
	342	จ3-9(6)-1/36บร	เตียง้วนเอียะ	โนนดินแดง	โนนดินแดง	บุรีรัมย์	300000	
	343	จ2-9(6)-1/41บร	เตียง้วนเอียะ	โนนดินแดง	โนนดินแดง	บุรีรัมย์	900000	
185	344	จ2-9(6)-1/37ลย	นายปกรณ์ จียาศักดิ์	นาอ้อ	เมืองเลย	เลย	420000	0.765
	345	จ3-9(6)-2/43ลย	ชัชวาลย์พืชผล	นาอ้อ	เมืองเลย	เลย	300000	
186	346	จ3-9(6)-40/49ลย	โชคพิชัย	กกดู่	เมืองเลย	เลย	400000	1.487
	347	จ3-9(6)-26/50ลย	นายสายฝน กองทอง	กกดู่	เมืองเลย	เลย	1000000	
187	348	จ3-9(6)-57/47ลย	ลานมันรุ่งเรืองกิจ	นาดินคำ	เมืองเลย	เลย	100000	0.106
188	349	จ3-9(6)-2/48ลย	ทวีพืชผล	นาแขม	เมืองเลย	เลย	300000	0.425
	350	จ3-9(6)-1/37ลย	โล้รุ่งเรือง	นาแขม	เมืองเลย	เลย	100000	
189	351	จ3-9(6)-1/41ลย	บริษัท จ.เจริญมาร์เก็ตติ้ง จำกัด	ชาตุ	เชียงคาน	เลย	500000	1.593
	352	จ3-9(6)-124/52ลย	ลานมัน มันทอง	ชาตุ	เชียงคาน	เลย	1000000	
190	353	จ3-9(6)-1/39ลย	ศรีชาวนาพูนผล	นาซ่าว	เชียงคาน	เลย	500000	0.743
	354	3-9(6)-2/35ลย	นายมนตรี เครือทองศรี	นาซ่าว	เชียงคาน	เลย	200000	
191	355	จ3-9(6)-31/53ลย	นางสาวรุ่งฤทัย มหาดไทย	หนองผือ	ท่าลี่	เลย	2000000	2.124
192	356	จ3-9(6)-10/53ลย	นางสาวสลิลรัตน์ พงส์เผ่าทอง	หนองหญ้าปล้อง	วังสะพุง	เลย	1000000	1.062

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
193	357	จ2-9(6)-3/50ลย	นายคำคิด แก้วกันหา	หนองงิ้ว	วังสะพุง	เลย	100000	0.319
	358	จ2-9(6)-2/50ลย	นางเล็ก ไชยสิทธิ์	หนองงิ้ว	วังสะพุง	เลย	100000	
	359	จ3-9(6)-19/49ลย	มนสวรรค์พืชผล	หนองงิ้ว	วังสะพุง	เลย	100000	
194	360	จ3-9(6)-1/44ลย	ห้างหุ้นส่วนจำกัด ภูหลวงเกษตรรุ่งเรือง	เขาหลวง	วังสะพุง	เลย	1000000	1.062
195	361	3-9(6)-2/31ลย	สมพงษ์พานิช	ศรีสงคราม	วังสะพุง	เลย	200000	0.743
	362	จ3-9(6)-24/48ลย	ไทยแมท	ศรีสงคราม	วังสะพุง	เลย	500000	
196	363	จ3-9(6)-7/43ลย	นางสุภาพร นนท์คำจันทร์	ห้วยส้ม	ภูกระดึง	เลย	50000	0.478
	364	จ3-9(6)-82/51ลย	ริยะจันทร์การเกษตร	ห้วยส้ม	ภูกระดึง	เลย	400000	
197	365	จ2-9(6)-1/50ลย	อุลิตพืชผล	ท่าช้างคล้อง	ผาขาว	เลย	100000	0.106
198	366	จ3-9(6)-3/48ลย	มนตรีพาณิชย์	เอราวัณ	เอราวัณ	เลย	100000	0.106
199	367	3-9(6)-1/31ลย	เอราวัณพืชผล	ผาอินทร์แปลง	เอราวัณ	เลย	3000	0.003
200	368	3-9(6)-3/35ลย	สหชัย	หนองหิน	หนองหิน	เลย	200000	0.212
201	369	จ3-9(6)-46/52ยส	ส.โชคชัย	ตาดทอง	เมืองยโสธร	ยโสธร	200000	0.212
202	370	3-9(6)-2/31ยส	สหผลพืชไร่	เชื่องคำ	เมืองยโสธร	ยโสธร	500000	0.531
203	371	จ3-9(6)-60/47ยส	ลานมันชัยยนต์	กุดชุม	กุดชุม	ยโสธร	50000	0.584
204	372	จ3-9(6)-40/52ยส	นายจุลทรัพย์ หลวงมณี	บุ่งค้า	เลิงนกทา	ยโสธร	200000	0.212
205	373	จ3-9(6)-1/42รอ	พรทวี	รอบเมือง	หนองพอก	ร้อยเอ็ด	200000	0.212

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
206	374	3-9(6)-2/35รอ	ทรัพย์ทวี	ผานำ้ย้อย	หนองพอก	ร้อยเอ็ด	200000	0.212
207	375	จ2-9(6)-2/31มค	โรงสีข้าวบรบือ	บรบือ	บรบือ	มหาสารคาม	1000000	1.465
	376	จ2-9(6)-23/22มค	แต้ฮะสุน	บรบือ	บรบือ	มหาสารคาม	60000	
	377	จ2-9(6)-1/46มค	แต้หม่งเส็ง	บรบือ	บรบือ	มหาสารคาม	100000	
	378	จ2-9(6)-2/46มค	ตั้งชิวฮะ	บรบือ	บรบือ	มหาสารคาม	220000	
208	379	จ3-9(6)-75/52มห	สุวิทย์พืชผล	โพนทราย	เมืองมุกดาหาร	มุกดาหาร	2100000	2.230
209	380	จ3-9(6)-8/51มห	บริษัท พรีเมียร์ คลอลิตี้ สตาร์ช จำกัด	คำป่าหลาย	เมืองมุกดาหาร	มุกดาหาร	1000000	1.380
	381	จ3-9(6)-3/37มห	ประเสริฐพืชผล	คำป่าหลาย	เมืองมุกดาหาร	มุกดาหาร	300000	
210	382	จ2-9(6)-1/41มห	ลานมันเทียนชัย	ดงหลวง	ดงหลวง	มุกดาหาร	410000	0.435
211	383	จ3-9(6)-1/44สน	ศรีรุ่งเรือง	วาริชภูมิ	วาริชภูมิ	สกลนคร	100000	0.106
212	384	จ3-9(6)-1/42สน	ห้างหุ้นส่วนจำกัด ส่องดาวพืชผล	ส่องดาว	ส่องดาว	สกลนคร	700000	0.743
213	385	3-9(6)-40/21สน	รุ่งโรจน์พืชผล	สร้างค้อ	ภูพาน	สกลนคร	200000	0.212
214	386	จ3-9(6)-2/43นค	บัวเลียนพืชผล	ศรีสำราญ*	พรเจริญ*	บึงกาฬ	500000	0.531
215	387	จ3-9(6)-113/52นค	แก่นคำพืชผล	เซกา*	เซกา*	บึงกาฬ	700000	0.743
216	388	จ2-9(6)-4/43นค	ไทยอีสานการค้า	ท่ากกแดง*	เซกา*	บึงกาฬ	100000	0.106
217	389	จ3-9(6)-14/53นค	ลานมันทรัพย์เจริญ	นำ้จั้น*	เซกา*	บึงกาฬ	500000	1.062
	390	จ3-9(6)-1/43นค	ลานมันทรัพย์เจริญ	นำ้จั้น*	เซกา*	บึงกาฬ	500000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
218	391	จ2-9(6)-6/43นค	เซกาใต้พืชผล	หนองทุ่ม*	เซกา*	บึงกาฬ	300000	0.319
219	392	จ3-9(6)-4/43นค	แอสโซ่พืชผล	โพธิ์หมากแข้ง*	บึงโขงหลง*	บึงกาฬ	100000	0.106
220	393	จ2-9(6)-1/52นค	สมภารลานมัน	ท่าดอกคำ*	บึงโขงหลง*	บึงกาฬ	300000	0.531
	394	จ2-9(6)-10/49นค	รุ่งเรืองพืชไร่	ท่าดอกคำ*	บึงโขงหลง*	บึงกาฬ	200000	
221	395	จ3-9(6)-3/43นค	บุญมากพืชผล	ชุมภูพร*	ศรีวิไล*	บึงกาฬ	500000	0.531
222	396	จ2-9(6)-1/43นค	พัฒนาพืชผล	นาแสง*	ศรีวิไล*	บึงกาฬ	1015000	1.078
223	397	จ3-9(6)-1/45นค	น้อยมาวิน	เฝ้าไร่	เฝ้าไร่	หนองคาย	150000	0.159
224	398	จ3-9(6)-6/43นค	ลานมันชัยมงคล	วังหลวง	เฝ้าไร่	หนองคาย	1000000	1.062
225	399	จ2-9(6)-1/48นภ	บริษัท ข้าวศรีไทยใหม่ จำกัด	หนองบัว	เมืองหนองบัวลำภู	หนองบัวลำภู	200000	0.425
	400	จ2-9(6)-2/48นภ	บริษัท ศรีไทยใหม่การเกษตร จำกัด	หนองบัว	เมืองหนองบัวลำภู	หนองบัวลำภู	200000	
226	401	จ3-9(6)-37/49นภ	ห้างหุ้นส่วนจำกัด อุดรไพบูลย์	โนนทัน	เมืองหนองบัวลำภู	หนองบัวลำภู	800000	11.469
	402	3-9(6)-1/26นภ	ห้างหุ้นส่วนจำกัด อุดรไพบูลย์	โนนทัน	เมืองหนองบัวลำภู	หนองบัวลำภู	10000000	
227	403	2-9(6)-1/33นภ	พูลทรัพย์	ค่านช้าง	นากลาง	หนองบัวลำภู	540000	0.573

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
228	404	3-9(6)-1/27นภ	รุ่งแสง	กุดดินจี่	นากลาง	หนองบัวลำภู	740000	1.317
	405	3-9(6)-4/31นภ	จันทาพืชผล	กุดดินจี่	นากลาง	หนองบัวลำภู	100000	
	406	3-9(6)-2/31นภ	บุญให้พืชผล	กุดดินจี่	นากลาง	หนองบัวลำภู	200000	
	407	จ3-9(6)-41/51นภ	บุญให้พืชผล	กุดดินจี่	นากลาง	หนองบัวลำภู	200000	
	408	2-9(6)-12/27อด	รุ่งแสง	กุดดินจี่*	*นากลาง	หนองบัวลำภู	200000	
229	409	3-9(6)-1/31นภ	หนองบัวไทยพิพัฒน์	ศรีบุญเรือง	ศรีบุญเรือง	หนองบัวลำภู	100000	0.106
230	410	3-9(6)-14/31นภ	เทพเจริญผล	เทพคีรี	นาวัง	หนองบัวลำภู	100000	0.106
231	411	จ3-9(6)-34/52ศก	รุ่งเรืองพืชผล	เมือง	กันทรลักษ์	ศรีสะเกษ	500000	0.531
232	412	จ3-9(6)-2/38ศก	นางสุมิน เจริญศรีสมบัติ	ละลาย	กันทรลักษ์	ศรีสะเกษ	1000000	1.062
233	413	จ2-9(6)-2/52ศก	บริษัท แสงเจริญรุ่งเรือง จำกัด	จานใหญ่	กันทรลักษ์	ศรีสะเกษ	300000	1.699
	414	จ3-9(6)-90/51ศก	นายมงคล อมรโรจน์วรวุฒิ	จานใหญ่	กันทรลักษ์	ศรีสะเกษ	1000000	
	415	จ3-9(6)-33/52ศก	บริษัท แสงเจริญรุ่งเรือง จำกัด	จานใหญ่	กันทรลักษ์	ศรีสะเกษ	300000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
234	416	3-9(6)-36/31ศก	ชัยทวีพืชผล	หนองหญ้าลาด	กันทรลักษ์	ศรีสะเกษ	100000	1.561
	417	จ3-9(6)-87/52ศก	แสงพาพร	หนองหญ้าลาด	กันทรลักษ์	ศรีสะเกษ	150000	
	418	จ2-9(6)-2/43ศก	ไทยเจริญ	หนองหญ้าลาด	กันทรลักษ์	ศรีสะเกษ	210000	
	419	3-9(6)-3/22ศก	จั่วฮวดพืชผล	หนองหญ้าลาด	กันทรลักษ์	ศรีสะเกษ	200000	
	420	จ3-9(6)-92/52ศก	ก.พืชผล	หนองหญ้าลาด	กันทรลักษ์	ศรีสะเกษ	500000	
	421	จ2-9(6)-1/43ศก	ชัยอุดม	หนองหญ้าลาด	กันทรลักษ์	ศรีสะเกษ	310000	
235	422	จ3-9(6)-3/42ศก	ลานมันศิริสมบัติ	ปรือใหญ่	ขุขันธ์	ศรีสะเกษ	500000	0.531
236	423	3-9(6)-7/22ศก	ดาวรุ่งพืชผล	ห้วยใต้	ขุขันธ์	ศรีสะเกษ	60000	0.064
237	424	จ2-9(6)-2/36ศก	มิตรขุนหาญ	สำโรงพลัน	ไพรบึง	ศรีสะเกษ	1000000	1.062
238	425	3-9(6)-60/31ศก	เก่งไมตรี	สิ	ขุนหาญ	ศรีสะเกษ	100000	4.428
	426	จ2-9(6)-1/41ศก	นายอนันต์ อดิเรกสมบัติ	สิ	ขุนหาญ	ศรีสะเกษ	1370000	
	427	3-9(6)-59/31ศก	ปิยะพืชผล	สิ	ขุนหาญ	ศรีสะเกษ	2000000	
	428	จ3-9(6)-3/43ศก	ลานมันอภิศักดิ์	สิ	ขุนหาญ	ศรีสะเกษ	200000	
	429	จ3-9(6)-11/53ศก	เกียหมิ่ง	สิ	ขุนหาญ	ศรีสะเกษ	500000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
239	430	จ3-9(6)-1/42ศก	นางสาเยือน มิ่งเมือง	บักดอง	ขุนหาญ	ศรีสะเกษ	50000	0.796
	431	จ3-9(6)-2/41ศก	ลานมันสวัสดี	บักดอง	ขุนหาญ	ศรีสะเกษ	50000	
	432	จ2-9(6)-1/38ศก	อึ้งเจริญ	บักดอง	ขุนหาญ	ศรีสะเกษ	200000	
	433	3-9(6)-38/31ศก	ศรีเพชรพืชผล	บักดอง	ขุนหาญ	ศรีสะเกษ	200000	
	434	จ3-9(6)-2/42ศก	นายเฉลา คำศรี	บักดอง	ขุนหาญ	ศรีสะเกษ	50000	
	435	จ2-9(6)-1/39ศก	มิตรไทยพืชผล	บักดอง	ขุนหาญ	ศรีสะเกษ	200000	
240	436	3-9(6)-1/35ศก	กิตติ	กันทรอม	ขุนหาญ	ศรีสะเกษ	200000	0.212
241	437	จ3-9(6)-1/43ศก	ชนนิยมพืชผล	ดงรัก	ภูสิงห์	ศรีสะเกษ	100000	0.106
242	438	จ3-9(6)-25/53ศก	นายหน่อย ฉินพิพัฒนา	หนองหว้า	เบญจลักษ์	ศรีสะเกษ	500000	2.655
	439	จ3-9(6)-32/53ศก	ภควรรทน์พืชผล	หนองหว้า	เบญจลักษ์	ศรีสะเกษ	2000000	
243	440	จ3-9(6)-33/53ศก	รนนท์พืชผล	ท่าคล้อ	เบญจลักษ์	ศรีสะเกษ	5000000	5.310
244	441	จ3-9(6)-64/52สร	เฉลิมพลพืชผล	นาบัว	เมืองสุรินทร์	สุรินทร์	10000000	10.619
245	442	จ3-9(6)-89/51สร	สหกรณ์ชาวไร่มันสำปะหลังจังหวัดสุรินทร์	ตะเคียน	กาบเชิง	สุรินทร์	500000	0.531
246	443	จ3-9(6)-34/49สร	นางสุณี ธูปมงคล	ตาตุม	สังขะ	สุรินทร์	150000	0.159
247	444	จ3-9(6)-88/51สร	อุดมผลบัวเชด	บัวเชด	บัวเชด	สุรินทร์	100000	0.106

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
248	445	จ3-9(6)-101/52อจ	ลานมัน ผ.เพิ่มพูน	บุ่ง	เมืองอำนาจเจริญ	อำนาจเจริญ	400000	3.610
	446	จ3-9(6)-102/52อจ	เพิ่มพูน ชัญทิพย์	บุ่ง	เมืองอำนาจเจริญ	อำนาจเจริญ	2000000	
	447	จ3-9(6)-28/48อจ	ลานมันสุขสวัสดิ์	บุ่ง	เมืองอำนาจเจริญ	อำนาจเจริญ	1000000	
249	448	จ3-9(6)-10/52อจ	บริษัท ศรีบุญเรืองการเกษตร จำกัด	ไก่คำ	เมืองอำนาจเจริญ	อำนาจเจริญ	10000000	10.619
250	449	จ3-9(6)-59/52อจ	ลานมันห้างหุ้นส่วน สาม เค เครือเกษตร	นายม	เมืองอำนาจเจริญ	อำนาจเจริญ	200000	0.531
	450	จ3-9(6)-38/53อจ	โรงสีโชคสว่าง	นายม	เมืองอำนาจเจริญ	อำนาจเจริญ	300000	
251	451	จ3-9(6)-76/52อจ	ทรัพน์ไพศาล	คำเขื่อนแก้ว	ชานุมาน	อำนาจเจริญ	100000	0.212
	452	จ3-9(6)-58/52อจ	ลานมันสำราญสุข	คำเขื่อนแก้ว	ชานุมาน	อำนาจเจริญ	100000	
252	453	จ3-9(6)-98/51อจ	ส. พานิช	คำโพน	ปทุมราชวงศา	อำนาจเจริญ	500000	1.062
	454	จ3-9(6)-15/53อจ	โรงสีโชคประเสริฐ อำนาจเจริญ	คำโพน	ปทุมราชวงศา	อำนาจเจริญ	500000	
253	455	จ3-9(6)-53/47อจ	อู่เฮงฮวด	นาหว้า	ปทุมราชวงศา	อำนาจเจริญ	200000	0.212
254	456	จ3-9(6)-9/53อจ	ห้างหุ้นส่วนจำกัด เกษตรธัญญเจริญลานมัน	ไร่สีสุก	เสนางคนิคม	อำนาจเจริญ	1000000	1.062
255	457	2-9(6)-5/18อด	ห้างหุ้นส่วนจำกัด อุดร ไทยแลนด์พืชผล	นิคมสงเคราะห์	เมืองอุดรธานี	อุดรธานี	500000	0.531
256	458	จ3-9(6)-5/49อด	บริษัท อุดรพัฒนกิจการเกษตร จำกัด	หนองนาคำ	เมืองอุดรธานี	อุดรธานี	15000000	15.929
257	459	2-9(6)-8/22อด	สันติชัย	กุดจับ	กุดจับ	อุดรธานี	50000	0.053
258	460	จ3-9(6)-18/53อด	กุดจับพืชผล	เมืองเพีย	กุดจับ	อุดรธานี	300000	0.425
	461	3-9(6)-136/31อด	กุดจับพืชผล	เมืองเพีย	กุดจับ	อุดรธานี	100000	

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
259	462	2-9(6)-6/28อด	นายสนั่น บัลลังก์	หมากหญ้า	หนองวัวซอ	อุดรธานี	210000	0.648
	463	2-9(6)-6/21อด	จรูญพืชผล	หมากหญ้า	หนองวัวซอ	อุดรธานี	400000	
260	464	จ2-9(6)-1/41อด	โชตเจริญพืชผล	น้ำพ่น	หนองวัวซอ	อุดรธานี	820000	0.871
261	465	2-9(6)-46/20อด	นายจรูญ แซ่เตียว	หนองบัวบาน	หนองวัวซอ	อุดรธานี	305000	0.324
262	466	จ2-9(6)-4/47อด	อุดรพืชผล	หนองวัวซอ	หนองวัวซอ	อุดรธานี	200000	0.212
263	467	3-9(6)-4/13อด	ไทยกสิกรรม	พันดอน	กุมภวาปี	อุดรธานี	400000	0.425
264	468	2-9(6)-5/31อด	เอื้อปัญญาพืชผล	เวียงคำ	กุมภวาปี	อุดรธานี	300000	0.319
265	469	จ2-9(6)-1/44อด	ศรีเจริญพืชผล	โนนสะอาด	โนนสะอาด	อุดรธานี	1000000	2.177
	470	2-9(6)-16/20อด	ส่งเสริมพืชผล	โนนสะอาด	โนนสะอาด	อุดรธานี	1000000	
	471	2-9(6)-49/31อด	สุรศักดิ์พืชผล	โนนสะอาด	โนนสะอาด	อุดรธานี	50000	
266	472	จ2-9(6)-12/49อด	เทียนชัย	ศรีธาตุ	ศรีธาตุ	อุดรธานี	300000	0.319
267	473	จ3-9(6)-1/45อด	ลานมันทวีศักดิ์พืชผล	หัวนาคำ	ศรีธาตุ	อุดรธานี	500000	0.531
268	474	3-9(6)-1/32อด	โชคประสิทธิ์	นายูง	ศรีธาตุ	อุดรธานี	80000	0.085
269	475	3-9(6)-12/28อด	โชคทวี	หนองหญ้าไซ	วังสามหมอ	อุดรธานี	300000	0.319
270	476	จ2-9(6)-11/49อด	เทียนชัย	ผาสุก	วังสามหมอ	อุดรธานี	300000	0.319

ตารางที่ ก.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
271	477	จ2-9(6)-9/37อด	นายประดิษฐ์ ศรีวารมย์	วังสามหมอ	วังสามหมอ	อุดรธานี	30000	1.104
	478	จ3-9(6)-2/52อด	ลานมัน บริวัฒน์ พืชผล	วังสามหมอ	วังสามหมอ	อุดรธานี	1000000	
	479	จ2-9(6)-7/37อด	ส.ทวีผล	วังสามหมอ	วังสามหมอ	อุดรธานี	10000	
272	480	จ2-9(6)-6/37อด	เพิ่มพูลผล	บ้านจันทน์	บ้านดุง	อุดรธานี	500000	0.531
273	481	2-9(6)-140/31อด	รุ่งชัยพืชผล	บ้านผือ	บ้านผือ	อุดรธานี	150000	0.159
274	482	จ2-9(6)-2/36อด	พ.พืชผล	ข้าวสาร	บ้านผือ	อุดรธานี	1000000	1.062
275	483	จ3-9(6)-100/52อด	กลางใหญ่รุ่งเรืองการเกษตร	กลางใหญ่	บ้านผือ	อุดรธานี	1000000	1.168
	484	จ3-9(6)-2/43อด	ประภาสพืชผล	กลางใหญ่	บ้านผือ	อุดรธานี	100000	
276	485	3-9(6)-17/32อด	แสงสวรรค์	คำด้วง	บ้านผือ	อุดรธานี	300000	0.329
	486	จ2-9(6)-7/36อด	นายสุริยนต์ ใจหาญ	คำด้วง	บ้านผือ	อุดรธานี	10000	
277	487	จ3-9(6)-10/54อด	ลานมันพูนสมบัติ	นางัว	น้ำโสม	อุดรธานี	1000000	3.430
	488	2-9(6)-5/35อด	ห้างหุ้นส่วนจำกัด อิตาลี่น้ำโสมพืชผล	นางัว	น้ำโสม	อุดรธานี	600000	
	489	3-9(6)-73/31อด	ส.เจริญ	นางัว	น้ำโสม	อุดรธานี	30000	
	490	3-9(6)-17/31อด	เดชเจริญผล	นางัว	น้ำโสม	อุดรธานี	100000	
	491	จ3-9(6)-13/47อด	ประเสริฐศรี	นางัว	น้ำโสม	อุดรธานี	500000	
	492	จ3-9(6)-15/54อด	นางัว อุดมผล	นางัว	น้ำโสม	อุดรธานี	1000000	
278	493	จ3-9(6)-37/53อด	รุ้งผล	น้ำโสม	น้ำโสม	อุดรธานี	5000000	5.310

ตารางที่ ค.1 ข้อมูลที่ตั้งลานมันสำปะหลังเขตภาคตะวันออกเฉียงเหนือของประเทศไทยที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด (ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	เงินทุนหมุนเวียน (บาท/ปี)	วัตถุดิบรวม (ตัน/วัน)
279	494	จ3-9(6)-71/52อด	ต.เจริญ	บ้านหยวก	น้ำโสม	อุดรธานี	1000000	1.083
	495	จ3-9(6)-6/43อด	แสงอรุณ	บ้านหยวก	น้ำโสม	อุดรธานี	20000	
280	496	3-9(6)-5/34อด	ห้างหุ้นส่วนจำกัด สุธีราพืชไร่	ศรีสำราญ	น้ำโสม	อุดรธานี	150000	0.531
	497	จ3-9(6)-1/43อด	ลานมันอุดมโชค	ศรีสำราญ	น้ำโสม	อุดรธานี	200000	
	498	จ3-9(6)-4/43อด	พูนสมบัติ	ศรีสำราญ	น้ำโสม	อุดรธานี	50000	
	499	จ3-9(6)-11/48อด	นางัวอุดมผล	ศรีสำราญ	น้ำโสม	อุดรธานี	100000	
281	500	จ3-9(6)-1/35อด	นายบุญธรรม แสนจักร์	หนองแวง	น้ำโสม	อุดรธานี	200000	0.265
	501	3-9(6)-30/31อด	ขวัญเมือง	หนองแวง	น้ำโสม	อุดรธานี	50000	
282	502	จ3-9(6)-3/43อด	ปัทมาวรรณ	นายูง	นายูง	อุดรธานี	500000	0.563
	503	3-9(6)-97/31อด	นายเที่ยง คำภูแก้ว	นายูง	นายูง	อุดรธานี	30000	
283	504	จ3-9(6)-65/52อด	ลานมันศิริพรพืชผล	บ้านก้อง	นายูง	อุดรธานี	500000	0.531

หมายเหตุ

ก.1.1 ช่องที่ 1 แสดงการรวบแหล่งวัตถุดิบ (ลานมัน) ที่อยู่ตำบลเดียวกันไว้ด้วยกัน สำหรับการคำนวณจะใช้หมายเลขลานมัน (หมายเลข 23-283) อ้างอิงจากคอลัมน์นี้

ก.1.2 ช่องที่ 2 แสดงลำดับของแหล่งวัตถุดิบ ทั้งหมดก่อนการรวบแหล่งวัตถุดิบที่อยู่ตำบลเดียวกันไว้ด้วยกัน

ก.1.3 ช่องที่ 3 – 7 แสดงข้อมูลของแหล่งวัตถุดิบ

ก.1.4 ช่องที่ 8 แสดงเงินทุนหมุนเวียนของแหล่งวัตถุดิบแต่ละแห่ง (หน่วยเป็นบาทต่อปี)

ก.1.5 ช่องที่ 9 แสดงความสามารถในการป้อนวัตถุดิบให้กับโรงงานแป้งมันสำปะหลังของแหล่งวัตถุดิบแต่ละแห่ง

ก.1.6 ราคาหัวมันสด 2.58 บาทต่อกิโลกรัม (อ้างอิง : รายงานสถานการณ์สินค้าเกษตรสำคัญ ธนาคารแห่งประเทศไทยสำนักงานภาคตะวันออกเฉียงเหนือ, พฤษภาคม 2554)

ภาคผนวก ข

ข้อมูลที่ตั้งของโรงงานที่ใช้แป้งมันสำปะหลังดิบในการผลิตสินค้าเป็นวัตถุดิบ ที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด

ตารางที่ ข.1 ข้อมูลที่ตั้งของโรงงานที่ใช้แป้งมันสำปะหลังดิบในการผลิตสินค้าเป็นวัตถุดิบที่จดทะเบียนโรงงานกับอุตสาหกรรมจังหวัด

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	แป้งแห้ง (กก./วัน)	แป้งดิบ (กก./วัน)	ความต้องการรวม (ตัน/วัน)
1	1	3-9(2)-5/53กจ	บริษัท ไทย ควอลิตี้ สตาร์ช จำกัด	หนองประดู่	เลาขวัญ	กาญจนบุรี	150000	218750	218.750
2	2	3-9(2)-1/37กส	บริษัท เนชั่นแนล สตาร์ช แอนด์ เคมิเคิล	นาจารย์	เมืองกาฬสินธุ์	กาฬสินธุ์	200000	291666.667	291.667
3	3	3-9(2)-1/31กส	บริษัท เอเชียโมดิไฟด์สตาร์ช จำกัด	โพนทอง	เมืองกาฬสินธุ์	กาฬสินธุ์	200000	291666.667	583.333
	4	3-9(2)-1/42กส	บริษัท เอเชียโมดิไฟด์สตาร์ช จำกัด	โพนทอง	เมืองกาฬสินธุ์	กาฬสินธุ์	200000	291666.667	
4	5	3-9(2)-1/19จบ	บริษัท สหสินวัฒนา แป้งแปรรูป จำกัด	ทับไทร	โป่งน้ำร้อน	จันทบุรี	150000	218750	218.750
5	6	3-9(2)-1/45ฉช	บริษัท สีมาอินเตอร์โปรดักส์ จำกัด	เกาะขนุน	พนมสารคาม	ฉะเชิงเทรา	200000	291666.667	291.667
6	7	ศ3-9(2)-1/39ฉช	บริษัทโซนิส สตาร์ช เทคโนโลยี จำกัด	เขาหินซ้อน	พนมสารคาม	ฉะเชิงเทรา	200000	291666.667	291.667
7	8	3-9(2)-2/37ชย	บริษัท สยาม ควอลิตี้ สตาร์ช จำกัด	โคกเริงรมย์	บำเหน็จณรงค์	ชัยภูมิ	150000	218750	218.750
8	9	3-9(2)-4/49อน	บริษัทโชคชัย โมดิฟายค์ สตาร์ช จำกัด	ทัพหลวง	บ้านไร่	นครปฐม	250000	364583.333	364.583
9	10	3-9(2)-2/51นม	บริษัท เอี่ยมเฮง โมดิฟาย สตาร์ช จำกัด	กุดโบสถ์	เสิงสาง	นครราชสีมา	200000	291666.667	291.667
10	11	3-9(2)-5/50นม	บริษัท ไทยอาซาฮี จำกัด	ทุ่งอรุณ	โชคชัย	นครราชสีมา	200000	291666.667	291.667
11	12	3-9(2)-1/19นม	บริษัท ชัยภูมิพืชผล จำกัด	มิตรภาพ	สีคิ้ว	นครราชสีมา	200000	291666.667	291.667
12	13	3-9(2)-1/34นม	บริษัท คอร์น โปรดักส์ จำกัด	สีคิ้ว	สีคิ้ว	นครราชสีมา	300000	437500	437.500
13	14	3-9(2)-4/20นม	บริษัท สงวนวงษ์อุตสาหกรรม จำกัด	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	800000	1166666.67	1750.000
	15	จ3-9(2)-1/44นม	บริษัท ออส-ไทย สตาร์ชเซส จำกัด	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	200000	291666.667	
	16	จ3-9(2)-3/54นม	บริษัท สงวนวงษ์สตาร์ชโค้ตติ้ง จำกัด	หนองบัวศาลา	เมืองนครราชสีมา	นครราชสีมา	200000	291666.667	
14	17	3-9(2)-2/42นม	บริษัท พี.วี.ดี.อินเตอร์เนชั่นแนล จำกัด	หนองหัวแรต	หนองบุญมาก	นครราชสีมา	200000	291666.667	291.667

(ต่อ)

รวมลูกค้า	ลำดับลูกค้า	เลขทะเบียนโรงงาน	ชื่อโรงงาน/ชื่อเจ้าของ	ตำบล	อำเภอ	จังหวัด	แป้งแห้ง (กก./วัน)	แป้งดิบ (กก./วัน)	ความต้องการรวม (ตัน/วัน)
15	18	3-9(2)-1/36นม	บริษัท เยนเนรัล สตาร์ช จำกัด	อรพิมพ์	ครบุรี	นครราชสีมา	400000	583333.333	583.333
16	19	3-9(2)-2/23รย	บริษัท ตระกูลคำ จำกัด	ชากพง	แกลง	ระยอง	150000	218750	218.750
17	20	3-9(2)-3/13รย	บริษัท เนชั่นแนล สตาร์ช แอนด์ เคมิเคิล	ทุ่งควายกิน	แกลง	ระยอง	400000	583333.333	583.333
18	21	ส3-9(2)-1/30รย	บริษัท สำปะหลังพัฒนา จำกัด	บ้านฉาง	บ้านฉาง	ระยอง	150000	218750	218.750
19	22	3-9(2)-10/49รบ	บริษัท อุตสาหกรรมแป้งมันบ้านโป่ง	เบิกไพร	บ้านโป่ง	ราชบุรี	150000	218750	437.500
	23	3-9(2)-6/47รบ	บริษัท นีโอเทคฟู้ด จำกัด	เบิกไพร	บ้านโป่ง	ราชบุรี	150000	218750	
20	24	3-9(2)-8/49ลบ	บริษัท บลูเวฟ สตาร์ช จำกัด	ศิลาทิพย์	ชัยบาดาล	ลพบุรี	200000	291666.667	291.667
21	25	3-9(2)-2/29สป	บริษัท เอเชียกลูโคส จำกัด	สำโรงกลาง	พระประแดง	สมุทรปราการ	150000	218750	218.750
22	26	3-9(2)-1/37สบ	บริษัท เจ้าคุณเกษตรพืชผล จำกัด	สองคอน	แก่งคอย	สระบุรี	200000	291666.667	291.667

หมายเหตุ

ก.1.1 ช่องที่ 1 แสดงการรวบลูกค้าที่อยู่ตำบลเดียวกันไว้ด้วยกัน สำหรับการคำนวณจะใช้หมายเลขลูกค้า (หมายเลข 1-22) อ้างอิงจากคอลัมน์นี้

ก.1.2 ช่องที่ 2 แสดงลำดับของลูกค้าทั้งหมดก่อนการรวบลูกค้าที่อยู่ตำบลเดียวกันไว้ด้วยกัน

ก.1.3 ช่องที่ 3 – 7 แสดงข้อมูลของลูกค้า

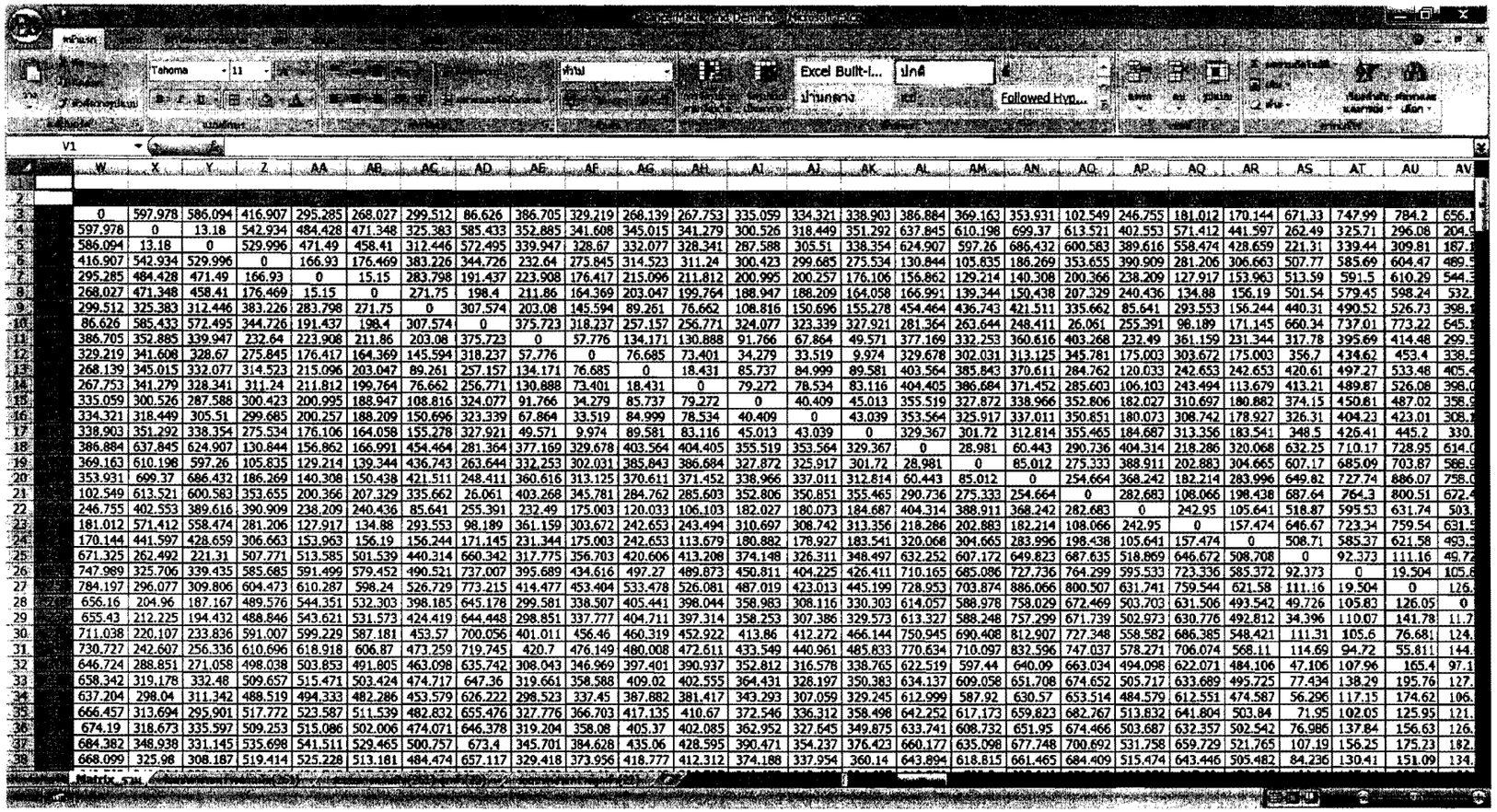
ก.1.4 ช่องที่ 8 แสดงความต้องการวัตถุดิบแป้งแห้งของลูกค้า (หน่วยเป็นกิโลกรัมต่อวัน)

ก.1.5 ช่องที่ 9 แสดงความต้องการวัตถุดิบแป้งดิบของลูกค้า โดยวัตถุดิบแป้งดิบ (Wet Cake) 87.5 กิโลกรัม สามารถผลิตเป็นวัตถุดิบแป้งแห้ง 60 กิโลกรัม

ก.1.6 ช่องที่ 10 แสดงความต้องการวัตถุดิบรวม (แป้งดิบ) สำหรับการคำนวณจะใช้ความต้องการวัตถุดิบแป้งดิบ (Wet Cake) เนื่องจากทำการหาที่เปิดโรงงานผลิตแป้งดิบ

ภาคผนวก ค

ข้อมูลระยะทางระหว่างลูกค้ากับแหล่งวัตถุดิบทั้งหมด (ตัวอย่าง) ในแผ่นซีดี ชื่อไฟล์ Distance Matrix and Demand.xlsx



	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV
1																										
2																										
3	0	597.978	586.094	416.907	295.285	268.027	299.512	86.626	386.705	329.219	268.139	267.753	335.059	334.321	338.903	386.884	369.163	353.931	102.549	246.755	181.012	170.144	671.33	747.99	784.2	656.1
4	597.978	0	13.18	542.934	484.428	471.348	325.383	585.433	352.885	341.608	345.015	341.279	300.526	318.449	351.292	637.845	610.198	699.37	613.521	402.553	571.412	441.597	262.49	325.71	296.08	204.9
5	586.094	13.18	0	529.996	471.49	458.41	312.446	572.495	339.947	328.67	332.077	328.341	287.588	305.51	338.354	624.907	597.26	686.432	600.583	389.616	558.474	428.659	221.31	339.44	309.81	187.1
6	416.907	542.934	529.996	0	166.93	176.469	383.226	344.726	232.64	275.845	314.523	311.24	300.423	299.685	275.534	130.844	105.835	186.269	353.655	390.909	281.206	306.663	507.77	585.69	604.47	489.5
7	295.285	484.428	471.49	166.93	0	15.15	283.798	191.437	223.908	176.417	215.096	211.812	200.995	200.257	176.106	156.862	129.214	140.308	200.366	238.209	127.917	153.963	513.59	591.5	610.29	544.3
8	268.027	471.348	458.41	176.469	15.15	0	271.75	198.4	211.86	164.369	203.047	199.764	188.947	188.209	164.058	166.991	139.344	150.438	207.329	240.436	134.88	156.19	501.54	579.45	598.24	532.1
9	299.512	325.383	312.446	383.226	283.798	271.75	0	307.574	203.08	145.594	89.261	76.662	108.816	150.696	155.278	454.464	436.743	421.511	335.662	85.641	293.553	156.244	440.31	490.52	526.73	398.1
10	86.626	585.433	572.495	344.726	191.437	198.4	307.574	0	375.723	318.237	257.157	256.771	324.077	323.339	327.921	281.364	263.644	248.411	26.061	255.391	98.189	171.145	660.34	737.01	773.22	645.1
11	386.705	352.885	339.947	232.64	223.908	211.86	203.08	375.723	0	57.776	134.171	130.888	91.766	67.864	49.571	377.169	332.253	360.616	403.268	232.49	361.159	231.344	317.78	395.69	414.48	299.5
12	329.219	341.608	328.67	275.845	176.417	164.369	145.594	318.237	57.776	0	76.685	73.401	34.279	33.519	9.974	329.678	302.031	313.125	345.781	175.003	303.672	175.003	356.7	434.62	453.4	338.5
13	268.139	345.015	332.077	314.523	215.096	203.047	89.261	257.157	134.171	76.685	0	18.431	85.737	84.999	89.581	403.564	385.843	370.611	284.762	120.033	242.653	242.653	420.61	497.27	533.48	405.4
14	267.753	341.279	328.341	311.24	211.812	199.764	76.662	256.771	130.888	73.401	18.431	0	79.272	78.534	83.116	404.405	386.684	371.452	285.603	106.103	243.494	113.679	413.21	489.87	526.08	398.0
15	335.059	300.526	287.588	300.423	200.995	188.947	108.816	324.077	91.766	34.279	85.737	79.272	0	40.409	45.013	355.519	327.872	338.966	352.806	182.027	310.697	180.882	374.15	450.81	487.02	358.9
16	334.321	318.449	305.51	299.685	200.257	188.209	150.696	323.339	67.864	33.519	84.999	78.534	40.409	0	43.039	353.564	325.917	337.011	350.851	180.073	308.742	178.927	326.31	404.23	423.01	308.7
17	338.903	351.292	338.354	275.534	176.106	164.058	155.278	327.921	49.571	9.974	89.581	83.116	45.013	43.039	0	329.367	301.72	312.814	355.465	184.687	313.356	183.541	348.5	426.41	445.2	330.1
18	386.884	637.845	624.907	130.844	156.862	166.991	454.464	281.364	377.169	329.678	403.564	404.405	355.519	353.564	329.367	0	28.981	60.443	290.736	404.314	218.286	320.068	632.25	710.17	728.95	614.0
19	369.163	610.198	597.26	105.835	129.214	139.344	436.743	263.644	332.253	302.031	385.843	386.684	327.872	325.917	301.72	28.981	0	85.012	275.333	388.911	202.883	304.665	607.17	685.09	703.87	588.9
20	353.931	699.37	686.432	186.269	140.308	150.438	421.511	248.411	360.616	313.125	370.611	371.452	338.966	337.011	312.814	60.443	85.012	0	254.664	368.242	182.214	283.996	649.82	727.74	886.07	758.0
21	102.549	613.521	600.583	353.655	200.366	207.329	335.662	26.061	403.268	345.781	284.762	285.603	352.806	350.851	355.465	290.736	275.333	254.664	0	282.683	108.066	198.438	687.64	764.3	800.51	672.4
22	246.755	402.553	389.616	390.909	238.209	240.436	85.641	255.391	232.49	175.003	120.033	106.103	182.027	180.073	184.687	404.314	388.911	368.242	282.683	0	242.95	105.641	518.87	595.53	631.74	503.1
23	181.012	571.412	558.474	281.206	127.917	134.88	293.553	98.189	361.159	303.672	242.653	243.494	310.697	308.742	313.356	218.286	202.883	182.214	108.066	242.95	0	157.474	646.67	723.34	759.54	631.9
24	170.144	441.597	428.659	306.663	153.963	156.19	156.244	171.145	231.344	175.003	242.653	113.679	180.882	178.927	183.541	320.068	304.665	283.996	198.438	105.641	157.474	0	508.71	585.37	621.58	493.9
25	671.325	262.492	221.31	507.771	513.585	501.539	440.314	660.342	317.775	356.703	420.606	413.208	374.148	326.311	348.497	632.252	607.172	649.823	687.635	518.869	646.672	508.708	0	92.373	111.16	49.72
26	747.989	325.706	339.435	585.685	591.499	579.452	490.521	737.007	395.689	434.616	497.27	489.873	450.811	404.225	426.411	710.165	685.086	727.736	764.299	595.533	723.336	585.372	92.373	0	19.504	105.8
27	784.197	296.077	309.806	604.473	610.287	598.24	526.729	773.215	414.477	453.404	533.478	526.081	487.019	423.013	445.199	728.953	703.874	886.066	800.507	631.741	759.544	621.58	111.16	19.504	0	126.1
28	656.16	204.96	187.167	489.576	544.351	532.303	398.185	645.178	299.581	338.507	405.441	398.044	358.983	308.116	330.303	614.057	588.978	758.029	672.469	503.703	631.506	493.542	49.726	105.83	126.05	0
29	655.43	212.225	194.432	488.846	543.621	531.573	424.419	644.448	298.851	337.777	404.711	397.314	358.253	307.386	329.573	613.327	588.248	757.299	671.739	502.973	630.776	492.812	34.396	110.07	141.78	11.7
30	711.038	220.107	233.836	591.007	599.229	587.181	453.57	700.056	401.011	456.46	460.319	452.922	413.86	412.272	466.144	750.945	690.408	812.907	727.348	558.582	686.385	548.421	111.31	105.6	76.681	124.1
31	730.727	242.607	256.336	610.696	618.918	606.87	473.259	719.745	420.7	476.149	480.008	472.611	433.549	440.961	485.833	770.634	710.097	832.596	747.037	578.271	706.074	568.11	114.69	94.72	55.811	144.8
32	646.724	288.851	271.058	498.038	503.853	491.805	463.098	635.742	308.043	346.969	397.401	390.937	352.812	316.578	338.765	622.519	597.44	640.09	663.034	494.098	622.071	484.106	47.106	107.96	165.4	97.1
33	658.342	319.178	332.48	509.657	515.471	503.424	474.717	647.36	319.661	358.588	409.02	402.555	364.431	328.197	350.383	634.137	609.058	651.708	674.652	505.717	633.689	495.725	77.434	138.29	195.76	127.1
34	637.204	298.04	311.342	488.519	494.333	482.286	453.579	626.222	298.523	337.45	387.882	381.417	343.293	307.059	329.245	612.999	587.92	630.57	653.514	484.579	612.551	474.587	56.296	117.15	174.62	106.1
35	666.457	313.694	295.901	517.772	523.587	511.539	482.832	655.476	327.776	366.703	417.135	410.67	372.546	336.312	358.498	642.252	617.173	659.823	682.767	513.832	641.804	503.84	71.95	102.05	125.95	121.1
36	674.19	318.673	335.597	509.253	515.086	502.006	474.071	646.378	319.204	358.08	405.37	402.085	362.952	327.645	349.875	633.741	608.732	651.95	674.466	503.687	632.357	502.542	76.986	137.84	156.63	126.6
37	684.382	348.938	331.145	535.698	541.511	529.465	500.757	673.4	345.701	384.628	435.06	428.595	390.471	354.237	376.423	660.177	635.098	677.748	700.692	531.758	659.729	521.765	107.19	156.25	175.23	182.1
38	668.099	325.98	308.187	519.414	525.228	513.181	484.474	657.117	329.418	373.956	418.777	412.312	374.188	337.954	360.14	643.894	618.815	661.465	684.409	515.474	643.446	505.482	84.236	130.41	151.09	134.1

ภาพที่ ค.1 ข้อมูลระยะทางระหว่างลูกค้ากับแหล่งวัตถุดิบทั้งหมด (ตัวอย่าง)

กรณีที่ 1 โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิต

Solution
Number of Vehicle = 62
Best Distance = 26092.5
Runtime = 9026.44 Second
Open Factory = 30
Best Routes:

Routes	Sum_H	Sum_D	Distance
30-3-60-59-62-61-55-271-269-266-97-96-53-30	13.306	15	791.308
30-3-91-100-103-101-102-104-99-105-125-163-159-30	14.764	15	990.701
30-3-52-56-54-57-270-211-212-267-264-265-107-108-92-67-66-79-75-72-71-74-166-170-168-181-30	14.06	15	1284.87
30-3-180-176-174-183-123-121-120-119-122-182-184-177-178-179-241-235-240-30	13.22	15	1151.5
30-3-205-203-248-254-204-251-252-253-29-25-24-30	14.912	15	814.851
30-3-171-169-142-140-162-161-128-126-127-115-124-172-30	14.305	15	1010.61
30-3-244-237-242-30	14.83	15	703.793
30-3-86-229-197-196-200-195-187-185-189-190-188-186-191-193-194-192-198-199-230-227-274-260-262-268-206-30	14.977	15	1533.15
30-3-106-70-68-69-164-137-46-30	15	15	971.651
30-3-202-201-250-249-30	14.972	15	588.396
30-3-87-89-130-131-155-153-150-136-167-141-30	14.723	15	1116.98
30-3-213-64-63-263-255-261-259-257-258-272-223-224-214-221-222-217-220-219-215-216-218-210-65-209-208-30	14.101	15	1508.5
30-3-27-30	14.92	15	548.954
30-3-80-85-83-84-81-82-90-77-73-157-158-152-154-143-160-30	14.907	15	1331.91
30-3-58-273-275-276-282-281-283-278-280-277-279-228-98-30	14.605	15	1190.28
30-43-42-35-39-41-30	15	15	207.32
30-3-129-138-117-118-116-173-30	9.664	0	981.911
30-233-238-239-232-231-234-30	15	15	224.81
30-23-44-26-27-30	14.914	15	294.708
30-3-76-133-165-135-134-132-147-148-146-145-175-245-246-247-30	14.76	15	1155.22
30-33-30	15	0	54.05
30-33-30	14.988	15	54.05
30-48-47-30	15	0	145.201
30-36-30	15	0	80.53
30-51-30	14.974	0	105.248
30-36-50-30	15	0	136.539
30-38-30	13.436	0	90.238
30-38-30	15	0	90.238
30-47-30	15	0	136.198
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-47-30	15	0	136.198
30-38-30	14.863	0	90.238
30-38-30	15	0	90.238
30-40-30	15	0	118.434
30-32-38-243-30	15	0	143.34
30-49-30	14.811	0	105.824
30-48-30	12.743	0	63.318
30-49-30	15	0	105.824
30-40-30	15	0	118.434
30-40-30	15	0	118.434
30-47-30	15	0	136.198
30-40-30	14.962	0	118.434
30-39-30	15	0	148.2
30-47-30	15	0	136.198
30-31-33-30	15	0	103.284
30-40-30	15	0	118.434
30-40-30	15	0	118.434
30-40-30	15	0	118.434
30-3-207-110-111-114-144-156-151-139-236-30	15	0	1146.58

Routes	Sum_H	Sum_D	Distance
30-47-30	14.203	0	136.198
30-41-30	15	0	142.42
30-3-94-226-225-93-95-109-30	15	0	956.863
30-41-30	15	0	142.42
30-3-27-30	15	0	548.954
30-27-30	15	0	166.928
30-34-40-37-30	15	0	134
30-45-30	15	0	26092.1
30-44-30	14.951	0	245.352
30-44-30	15	0	245.352
30-45-30	14.943	0	249.016
30-27-30	13.306	15	166.928
Total_Sum	900	300	24706.3

กรณีที่ 2 โรงงานแป้งมันสำปะหลังทำการผลิตเต็มกำลังการผลิตและจำกัดระยะทาง

Solution
Number of Vehicle = 67
Best Distance = 24319.2
Runtime = 6235.53 Second
Open Factory = 30
Best Routes:

Routes	Sum_H	Sum_D	Distance
30-14-161-128-126-127-115-124-173-30	11.601	15	701.606
30-14-163-159-175-174-183-123-172-30	6.913	15	693.593
30-14-112-144-114-170-141-236-30	14.548	15	773.217
30-14-27-30	15	15	707.428
30-14-146-148-145-113-30	12.79	15	791.637
30-14-237-243-30	6.372	15	635.542
30-14-143-138-139-238-233-242-30	11.841	15	790.873
30-14-117-118-116-119-120-121-122-182-30	14.124	15	793.043
30-14-162-125-140-142-168-169-171-181-180-30	3.471	15	778.149
30-14-154-152-30	1.062	15	799.013
30-14-147-111-129-30	7.008	15	799.084
30-14-176-184-177-178-179-234-30	6.127	15	735.854
30-14-149-30	15	15	778.791
30-14-149-153-30	3.106	15	796.865
30-14-160-231-49-30	14.096	15	699.546
30-14-26-27-30	7.304	15	719.885
30-14-244-245-246-247-241-235-240-239-232-30	14.122	15	786.469
30-14-27-30	15	15	707.428
30-31-43-50-30	14.548	0	121.795
30-14-10-113-30	15	15	729.03
30-38-30	15	0	90.238
30-33-30	15	0	54.05
30-33-30	15	0	54.05
30-33-45-253-252-251-204-254-30	14.439	0	433.995
30-36-30	15	0	80.53
30-48-30	15	0	63.318
30-23-44-29-28-208-209-65-210-213-206-30	14.434	0	763.592
30-51-30	12.743	0	105.248
30-40-30	15	0	118.434
30-49-30	15	0	105.824
30-46-202-52-56-54-55-57-61-62-60-63-64-205-203-48-30	14.212	0	777.277
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-47-30	15	0	136.198
30-249-30	10.619	0	238.178
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-38-42-40-34-30	11.649	0	118.264
30-47-30	15	0	136.198
30-41-30	15	0	142.42
30-24-25-47-30	11.015	0	281.178

Routes	Sum_H	Sum_D	Distance
30-41-30	15	0	142.42
30-39-30	15	0	148.2
30-40-30	15	0	118.434
30-47-30	15	0	136.198
30-14-27-30	15	15	707.428
30-47-30	15	0	136.198
30-27-30	15	0	166.928
30-40-30	15	0	118.434
30-47-30	15	0	136.198
30-40-30	15	0	118.434
30-40-30	15	0	118.434
30-40-30	15	0	118.434
30-36-35-39-32-30	14.494	0	168.778
30-40-30	15	0	118.434
30-37-41-30	14.226	0	156.72
30-44-30	15	0	245.352
30-28-30	15	0	331.154
30-44-30	15	0	245.352
30-45-30	15	0	249.016
30-28-30	15	0	331.154
30-45-30	15	0	249.016
30-91-104-101-103-207-201-250-248-30	14.95	0	747.587
30-44-30	15	0	245.352
30-44-30	15	0	245.352
30-28-30	15	0	331.154
Total_Sum	900	300	24319.2

กรณีที่ 3 โรงงานแป้งมันสำปะหลังทำการผลิต 80 เปอร์เซ็นต์ของกำลังการผลิต

Solution
Number of Vehicle = 49
Best Distance = 21265.4
Runtime = 6858.92 Second
Open Factory = 30
Best Routes:

Routes	Sum_H	Sum_D	Distance
30-14-161-162-279-280-283-281-276-273-257-262-263-57-205-206-203-201-30	14.774	15	1417.57
30-14-128-126-127-115-117-118-116-119-120-121-124-173-30	14.76	15	792.281
30-14-175-174-183-123-122-182-184-177-178-179-172-176-170-168-142-169-171-181-30	14.686	15	1028.61
30-14-112-160-30	13.38	15	702.789
30-14-129-78-90-76-166-30	14.923	15	878.51
30-14-113-30	15	15	700.569
30-14-154-152-157-135-79-75-72-71-74-130-131-155-150-153-163-159-30	14.947	15	1099.48
30-14-113-111-141-236-237-30	13.757	15	735.481
30-14-145-148-136-132-133-165-81-77-73-89-66-30	14.909	15	1018.64
30-14-149-30	15	15	778.791
30-14-156-158-149-151-134-82-84-85-80-30	14.975	15	1163.89
30-14-138-139-143-144-146-147-167-164-67-68-70-86-107-265-58-53-202-26-30	14.06	15	1357.14
30-231-234-233-242-243-30	11.756	0	143.8
30-14-245-246-247-241-235-240-238-239-232-43-30	13.984	15	782.053
30-14-140-100-101-102-180-244-30	14.951	15	944.846
30-14-125-103-104-106-197-193-186-188-190-189-187-198-199-230-229-30	14.869	15	1499.09
30-33-30	15	0	54.05
30-48-30	15	0	63.318
30-33-30	15	0	54.05
30-34-42-39-35-48-30	14.282	0	220.929
30-49-30	15	0	105.824
30-36-30	15	0	80.53
30-40-30	15	0	118.434

Routes	Sum_H	Sum_D	Distance
30-33-24-29-253-252-251-204-254-23-46-30	14.865	0	509.085
30-39-30	15	0	148.2
30-38-30	15	0	90.238
30-50-36-30	13.671	0	136.539
30-40-30	15	0	118.434
30-51-30	12.743	0	105.248
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-38-30	15	0	90.238
30-32-38-64-63-59-62-61-55-54-56-52-30	14.97	0	738.356
30-14-114-137-110-99-91-95-98-266-267-268-269-270-212-223-216-211-213-210-65-209-250-30	14.774	15	1578.78
30-47-30	15	0	136.198
30-47-30	12.431	0	136.198
30-40-30	15	0	118.434
30-47-30	15	0	136.198
30-40-30	15	0	118.434
30-47-30	15	0	136.198
30-40-30	15	0	118.434
30-41-30	15	0	142.42
30-47-30	15	0	136.198
30-40-30	15	0	118.434
30-40-30	15	0	118.434
30-49-30	13.353	0	105.824
30-31-37-30	14.994	0	127.219
Total_Sum	720	240	21265.4

กรณีที่ 4 โรงงานแป้งมันสำปะหลังทำการผลิต 80 เปอร์เซ็นต์ของกำลังการผลิตและจำกัดระยะทาง

Solution
Number of Vehicle = 45
Best Distance = 17168.9
Runtime = 4332.41 Second
Open Factory = 40
Best Routes:

Routes	Sum_H	Sum_D	Distance
40-9-123-122-172-41-40	14.494	15	780.431
40-9-121-120-119-116-118-40	12.689	15	780.593
40-9-183-174-175-125-141-244-236-40	12.796	15	774.618
40-9-184-177-178-179-245-246-247-241-235-240-40	10.034	15	799.512
40-9-117-138-139-161-159-40	8.054	15	795.301
40-9-237-233-33-40	15	15	694.916
40-9-124-163-162-170-168-169-171-239-40	8.998	15	794.83
40-9-128-129-111-238-40	14.493	15	775
40-9-113-40	15	15	694.916
40-9-115-127-126-112-160-40	13.593	0	112.738
40-9-182-173-140-232-43-40	15	15	784.911
40-9-113-176-234-231-40	15	15	784.911
40-9-142-181-180-37-40	12.339	15	739.012
40-50-243-242-40	11.551	0	198.605
40-49-40	15	15	694.916
40-42-51-40	15	15	694.916
40-38-40	15	0	38.5
40-41-40	15	0	29.064
40-38-40	15	0	46.534
40-38-40	15	0	29.064
40-38-40	15	0	29.064
40-38-40	15	0	29.064
40-38-40	15	0	29.064
40-38-31-32-26-201-202-203-250-46-40	15	0	29.064
40-49-40	15	15	785.406
40-39-40	15	0	38.5
40-41-40	13.353	0	46.534

40-36-40	14.472	0	741.448
40-39-35-36-40	12.123	0	476.36
40-48-248-23-30-40	15	0	122.624
40-47-40	15	0	122.624
40-33-40	12.838	0	765.393
40-33-40	15	0	226.78
40-48-40	15	0	226.78
40-34-47-24-25-29-253-252-251-204-206-205-254-40	6.663	0	74.128
40-44-209-65-210-208-40	15	0	226.78
40-47-40	15	0	226.78
40-45-40	15	0	226.78
40-47-40	15	0	40.778
40-47-40	15	0	167.456
40-9-27-40	15	15	785.406
40-47-40	15	0	327.158
40-27-40	15	0	327.158
40-9-27-40	15	0	54.44
40-9-27-40	15	0	54.44
Total_Sum	720	240	17168.9

กรณีที่ 5 โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำลังการผลิต

Solution
Number of Vehicle = 30
Best Distance = 10942
Runtime = 4012.59 Second
Open Factory = 43
Best Routes:

Routes	Sum_H	Sum_D	Distance
43-14-161-162-163-159-115-124-174-175-160-173-43	14.903	15	796.306
43-14-128-126-127-117-118-116-120-121-123-236-43	14.984	15	771.342
43-14-129-78-90-76-74-89-43	14.987	15	941.309
43-14-138-139-143-154-152-153-150-155-131-132-133-130-147-169-43	14.994	15	1067.04
43-14-113-43	15	15	732.39
43-14-113-111-114-144-146-166-101-100-199-98-268-43	14.898	15	1520.26
43-14-112-145-165-84-83-75-72-170-43	14.983	15	1069.55
43-14-141-176-183-122-182-172-184-177-178-179-246-247-241-235-238-239-43	14.994	15	976.704
43-14-149-43	15	15	810.604
43-14-125-140-69-66-67-164-137-142-168-171-181-180-43	14.993	15	1051.95
43-38-43	15	0	16.926
43-38-43	15	0	16.926
43-38-43	15	0	16.926
43-38-43	15	0	16.926
43-38-43	15	0	16.926
43-38-43	15	0	16.926
43-42-35-38-43	13.856	0	82.424
43-41-40-34-31-43	13.289	0	95.028
43-40-43	15	0	28.312
43-40-43	15	0	28.312
43-40-43	15	0	28.312
43-40-43	15	0	28.312
43-40-43	15	0	28.312
43-40-43	15	0	28.312
43-40-43	15	0	28.312
43-32-30-23-26-233-237-240-231-50-43	14.282	0	443.396
43-49-43	15	0	37.948
43-37-234-43	14.431	0	154.379
43-41-43	15	0	53.738
43-49-43	13.353	0	37.948
Total_Sum	450	150	10942

กรณีที่ 6 โรงงานแป้งมันสำปะหลังทำการผลิต 50 เปอร์เซ็นต์ของกำลังการผลิตและจำกัดระยะทาง

Solution
Number of Vehicle = 25
Best Distance = 8832.26
Runtime = 2745.77 Second
Open Factory = 40
Best Routes:

Routes	Sum_H	Sum_D	Distance
40-14-161-163-159-236-240-40	9.488	15	695.824
40-14-160-175-174-124-120-121-123-183-172-247-241-235-40	10.204	15	797.713
40-14-126-127-115-116-177-234-50-40	11.543	15	799.991
40-14-162-125-140-141-239-49-40	14.601	15	793.027
40-14-112-145-231-40	13.752	15	796.976
40-14-113-40	15	15	735.029
40-14-129-176-246-233-242-30-34-40	13.646	15	799.715
40-14-139-138-117-118-173-238-40	14.654	15	780.776
40-14-170-168-169-171-181-244-237-232-40	13.666	15	786.408
40-14-128-144-114-111-113-40	13.628	15	799.551
40-38-40	15	0	29.064
40-43-31-40	8.177	0	58.334
40-38-40	15	0	29.064
40-38-40	15	0	29.064
40-38-40	15	0	29.064
40-38-40	15	0	29.064
40-36-40	15	0	40.778
40-38-40	15	0	29.064
40-38-42-35-40	13.856	0	74.603
40-37-40	12.87	0	35.32
40-36-46-23-253-248-250-26-32-40	13.405	0	485.862
40-39-40	15	0	54.44
40-41-40	15	0	38.5
40-49-40	15	0	46.534
40-41-40	15	0	38.5
Total_Sum	450	150	8832.26

กรณีที่ 7 โรงงานแป้งมันสำปะหลังทำการผลิต 20 เปอร์เซ็นต์ของกำลังการผลิต

Solution
Number of Vehicle = 5
Best Distance = 3911.34
Runtime = 500.995 Second
Open Factory = 40
Best Routes:

Routes	Sum_H	Sum_D	Distance
40-13-113-40	15	15	731.926
40-13-113-111-114-145-146-153-155-199-98-40	14.994	15	1387.91
40-13-129-78-76-90-84-165-132-40	14.986	15	981.297
40-13-112-162-163-159-160-40	14.973	15	781.144
40-38-40	13.537	0	29.064
Total_Sum	180	60	3911.34

หมายเหตุ

Sum_H = ปริมาณหัวมันสดที่ไปรับจากลานมันสำปะหลังโดยรถบรรทุกแต่ละคัน (ตัน)

Sum_D = ปริมาณแป้งมันดิบที่ไปส่งให้ลูกค้าโดยรถบรรทุกแต่ละคัน (ตัน)

Distance = ระยะทางของรถบรรทุกแต่ละคัน (กิโลเมตร)

Total_Sum_H = ปริมาณหัวมันสดที่ไปรับทั้งหมด ซึ่งรวมกับปริมาณหัวมันสด ณ จุดที่เปิดเป็นโรงงานแป้งมันสำปะหลัง (ตัน)

ภาคผนวก จ

Source code โปรแกรม Lingo

```
MODEL:
SETS:
        CITY / 1..28/:D,U;
        ROUTE/1..5/;
        CXC( CITY, CITY):R;
        CXK( CITY,ROUTE);
        LINK( CITY, CITY,ROUTE):X;
ENDSETS
DATA:
        W = 28; !จำนวนโหนดทั้งหมด;
        M = 5; !ลำดับสุดท้ายของโหนดลูกค้า;
        Q = 15; !ความจุของรถบรรทุก;
        D = @FILE('Demand20.txt'); !ความต้องการโหนด i;
        R = @FILE('Distance20.txt'); !ระยะทางระหว่างโหนด i ไปโหนด j;
ENDDATA

        MIN = @SUM(LINK(I,J,K):R(I,J)*X(I,J,K));

        @FOR(CITY(J)|(J #GT# 1) :@SUM(CXK(I,K):X(I,J,K))=1);
        @FOR(CITY(I)|(I #GT# 1) :@SUM(CXK(J,K):X(I,J,K))=1);
        @FOR(ROUTE(K):@SUM(CITY(I):X(I,1,K))<=1);
        @FOR(ROUTE(K):@SUM(CITY(J):X(1,J,K))<=1);
        @FOR(CXK(J,K):@SUM(CITY(I):X(I,J,K))-
@SUM(CITY(I):X(J,I,K))=0);
        @FOR(ROUTE(K):@SUM(CITY(I)|(I #GT# 1) #AND# (I #LE#
M):D(I)*@SUM(CITY(J):X(I,J,K)))<= Q);

        @FOR(ROUTE(K):@SUM(CITY(I)|(I #GT#
M):D(I)*@SUM(CITY(J):X(I,J,K)))<= Q);
        @FOR(LINK(I,J,K)|(I #GT# M) #AND# (J #GT# 1) #AND# (J #LE#
M):X(I,J,K)= 0);

        @FOR(ROUTE(K):@SUM(CXC(I,J)|(J #GT# M) #AND# (I #GT# 1) #AND#
(I #LE# M):X(I,J,K))<=1);

        @FOR(LINK(I,I,K):X(I,I,K)= 0);
        @FOR(LINK:@BIN(X));
        @FOR(LINK(I,J,K)|(I #GT# 1) #AND# (J #GT# 1):
        ((W-3)*X(J,I,K))+((W-1)*X(I,J,K))+U(I) <= U(J)+(W-2) );

END
```

ภาคผนวก ฉ

Source Code ของภาษา C++ ที่ใช้คำนวณ

```
#include<iostream>
#include<fstream>
#include<string>
#include<stdlib.h>
#include<time.h>
#include<math.h>
#include<conio.h>
#include<climits>
#include<limits>

using namespace std;

double Random(int&No);
int CalPop(int Start,int Finish,double* Attractiveness,double SumAttractiveness,
          int& No,bool Test);
void AntLRP(int AntPopulationL,int AntPopulationR,int MaxLoopL,int MaxLoopR,
          int Alpha_L,int Beta_L,int Alpha_R,int Beta_R,double EvaporationRate_L,
          double EvaporationRate_R,int AntFinal,int MaxLoopFinal);
int OpenFactory(double* Pheromone_L,int Alpha_L,int Beta_L,double* SumDistance_L,
          int& No);
void SelectM(bool&Possible,int Factory,int**& Route,double**& DSent,
          double*& DofRoute,int*& MofRoute,int& TotalRoute,double*&SumDisRoute,
          double** Pheromone_R,int Alpha_R,int Beta_R,int& No,bool Test);
void SelectN(bool&Possible,int Factory,int**& Route,double**& HSent,
          double*& HofRoute,int*& MofRoute,int*& NofRoute,
          int& TotalRoute,double*&SumDisRoute,double** Pheromone_R,int Alpha_R,
          int Beta_R,int& No,bool Test);
void Update(int FactoryModel,int& Factory,int** RouteModel,int**& Route,
          double** DSentModel,double**& DSent,double** HSentModel,
          double**& HSent,double* DofRouteModel,double*& DofRoute,
          double* HofRouteModel,double*& HofRoute,
          int* MofRouteModel,int*& MofRoute,int* NofRouteModel,int*& NofRoute,
          int TotalRouteModel,int& TotalRoute,
          double*SumDisRouteModel,double*&SumDisRoute);
double CallDistance(bool Show,int TotalRoute,int** Route,int* MofRoute,int* NofRoute);
void OneMove(int Factory,int**& Route,double**& DSent,double**& HSent,
          double*& DofRoute,double*& HofRoute,int*& MofRoute,int*& NofRoute,
          int& TotalRoute,double*&SumDisRoute,double&TotalDistance);
void Show(int Factory,int** Route,double** DSent,double** HSent,
          double* DofRoute,double* HofRoute,int* MofRoute,int* NofRoute,
          int TotalRoute,double*SumDisRoute,double TotalDistance);
void Exchage(int**& Route,double**& DSent,double**& HSent,double*& DofRoute,
          double*& HofRoute,int*& MofRoute,int*& NofRoute,int& TotalRoute,
          double*&SumDisRoute,double&TotalDistance,bool Test);
void TwoOptMove(int**& Route,double**& DSent,double**& HSent,int*& MofRoute,
          int*& NofRoute,int& TotalRoute,double*&SumDisRoute,double&TotalDistance);
void Reset(int**& Route,double**& DSent,double**& HSent,double*& DofRoute,
          double*& HofRoute,int*& MofRoute,int*& NofRoute,int& TotalRoute,
          double*&SumDisRoute,double&TotalDistance);

ofstream myfile;

//****************************    Set Parameter    *******************************

int V = 720;
int Capa = 240;
double MaxDistance = 100000;
int M = 22,N = 261;
int Total_Nod = M+N;
int Qk = 15;
double**Distance =(double**)calloc(Total_Nod+1,sizeof(double*));
double*H = new double [Total_Nod+1];
double*D = new double [Total_Nod+1];
int NoRandom = 1;

//*******************************    main    ************************************

int main(){
int TTT;
clock_t starttime, endtime;
double Time;
int i=0,j=0;
```

```
for(i=1;i<=Total_Nod;i++){
          Distance[i] = (double*)calloc(Total_Nod+1,sizeof(double));
}

//----------------------Open File-------------------------------

fstream file;
file.open("DATA.txt",ios::in);

if(file.fail()){
          cout<<"Error opening DATA.txt\n";
          getch();
          return 1;

}//check Error

myfile.open ("AntLRP.txt",ios::out);

//--------------------  Add Demand and Distance ------------

for(i=1;i<=M;i++){
          file>>D[i];
}

for(i=M+1;i<=Total_Nod;i++){
          file>>H[i];
}

for(i=1;i<=(Total_Nod);i++){
          for(j=1;j<=(Total_Nod);j++){
                    file >> Distance[i][j];
          }
}

for(i=1;i<=(Total_Nod);i++){
          for(j=1;j<=(Total_Nod);j++){
                    if(Distance[i][j] != Distance[j][i]){
                    cout<<"Error Distance["<<i<<"]["<<j<<"] !=
                    Distance["<<j<<"]["<<i<<"]\n";
                    cout<<"Distance["<<i<<"]["<<j<<"]= "<<Distance[i][j]<<endl;
                    cout<<"Distance["<<j<<"]["<<i<<"]= "<<Distance[j][i]<<endl;
                    }
          }
}

file.close();

//=================================== Ant LRP ================================

starttime = clock();
AntLRP(50,5,100,5,1,5,1,5,0.02,0.02,50,50);
endtime = clock();
Time = ((double)(endtime-starttime))/CLOCKS_PER_SEC;
cout<<"\n\nTotal Time = "<<Time<<'\n';
myfile<<"\n\nTotal Time = "<<Time<<'\n';

//============================================================================
//getch();
/*End main*/}

//*****************************    Ant LRP    ********************************
void AntLRP(int AntPopulationL,int AntPopulationR,int MaxLoopL,int MaxLoopR,
          int Alpha_L,int Beta_L,int Alpha_R,int Beta_R,double EvaporationRate_L,
          double EvaporationRate_R,int AntFinal,int MaxLoopFinal){

int i=0,j=0,jj=0;
int Factory,FactoryQueen,Factory1,Factory2,Factory3;
int TotalRoute,TotalRouteQueen,TotalRoute1,TotalRoute2,TotalRoute3;
double* Pheromone_L= new double [Total_Nod+1];
double* SumDistance_L = new double [Total_Nod+1];
double** Pheromone_R =  (double**)calloc(Total_Nod+1,sizeof(double*));
double Z,ZQueen,Z1,Z2,Z3;
int** S = (int**)calloc(Total_Nod+1,sizeof(int*));
```

```
int** SQueen = (int**)calloc(Total_Nod+1,sizeof(int*));
int** S1 = (int**)calloc(Total_Nod+1,sizeof(int*));
int** S2 = (int**)calloc(Total_Nod+1,sizeof(int*));
int** S3 = (int**)calloc(Total_Nod+1,sizeof(int*));
double** DSent = (double**)calloc(Total_Nod+1,sizeof(double*));
double** DSentQueen = (double**)calloc(Total_Nod+1,sizeof(double*));
double** DSent1 = (double**)calloc(Total_Nod+1,sizeof(double*));
double** DSent2 = (double**)calloc(Total_Nod+1,sizeof(double*));
double** DSent3 = (double**)calloc(Total_Nod+1,sizeof(double*));
double* DofRoute = new double [Total_Nod+1];
double* DofRouteQueen = new double [Total_Nod+1];
double* DofRoute1 = new double [Total_Nod+1];
double* DofRoute2 = new double [Total_Nod+1];
double* DofRoute3 = new double [Total_Nod+1];
int* MofRoute = new int [Total_Nod+1];
int* MofRouteQueen = new int [Total_Nod+1];
int* MofRoute1 = new int [Total_Nod+1];
int* MofRoute2 = new int [Total_Nod+1];
int* MofRoute3 = new int [Total_Nod+1];
double** HSent = (double**)calloc(Total_Nod+1,sizeof(double*));
double** HSentQueen = (double**)calloc(Total_Nod+1,sizeof(double*));
double** HSent1 = (double**)calloc(Total_Nod+1,sizeof(double*));
double** HSent2 = (double**)calloc(Total_Nod+1,sizeof(double*));
double** HSent3 = (double**)calloc(Total_Nod+1,sizeof(double*));
double* HofRoute = new double [Total_Nod+1];
double* HofRouteQueen = new double [Total_Nod+1];
double* HofRoute1 = new double [Total_Nod+1];
double* HofRoute2 = new double [Total_Nod+1];
double* HofRoute3 = new double [Total_Nod+1];
int* NofRoute = new int [Total_Nod+1];
int* NofRouteQueen = new int [Total_Nod+1];
int* NofRoute1 = new int [Total_Nod+1];
int* NofRoute2 = new int [Total_Nod+1];
int* NofRoute3 = new int [Total_Nod+1];
double*SumDisRoute = new double [Total_Nod+1];
double*SumDisRouteQueen = new double [Total_Nod+1];
double*SumDisRoute1 = new double [Total_Nod+1];
double*SumDisRoute2 = new double [Total_Nod+1];
double*SumDisRoute3 = new double [Total_Nod+1];
bool PossibleM,PossibleN;
int NoLoopNImpossible;
double* SumDisQueen = new double [N+1];
double* DummySumDisQueen = new double [N+1];
int *UpdatePheL = new int [17];
double CheckDisL;
int * NoUpdate  = new int [N+1];
int TTT = 0;
double* Attractiveness_L =  new double [Total_Nod+1];
double SumAttractiveness_L = 0;

for(i=0;i<=Total_Nod;i++){
          Pheromone_R[i]=(double*)calloc(Total_Nod+1,sizeof(double));
          S[i] = (int*)calloc(Total_Nod+1,sizeof(int));
          SQueen[i] = (int*)calloc(Total_Nod+1,sizeof(int));
          S1[i] = (int*)calloc(Total_Nod+1,sizeof(int));
          S2[i] = (int*)calloc(Total_Nod+1,sizeof(int));
          S3[i] = (int*)calloc(Total_Nod+1,sizeof(int));
          DSent[i]= (double*)calloc(5,sizeof(double));
          DSentQueen[i]= (double*)calloc(5,sizeof(double));
          DSent1[i]= (double*)calloc(5,sizeof(double));
          DSent2[i]= (double*)calloc(5,sizeof(double));
          DSent3[i]= (double*)calloc(5,sizeof(double));
          HSent[i]= (double*)calloc(Total_Nod+1,sizeof(double));
          HSentQueen[i]= (double*)calloc(Total_Nod+1,sizeof(double));
          HSent1[i]= (double*)calloc(Total_Nod+1,sizeof(double));
          HSent2[i]= (double*)calloc(Total_Nod+1,sizeof(double));
          HSent3[i]= (double*)calloc(Total_Nod+1,sizeof(double));
}

int Ant_F,loopF;
int loopL,loopR,Ant_L,Ant_R;
int No = 1;
```

```
for(i=1;i<=Total_Nod;i++){
          Pheromone_L[i] = 1;
}

for(i=M+1;i<=Total_Nod;i++){
          SumDistance_L[i] = 0;
          for(j=1;j<=Total_Nod;j++){
                    SumDistance_L[i] = SumDistance_L[i]+Distance[i][j];
          }
          SumDistance_L[i] = 100.0/SumDistance_L[i];
          SumDistance_L[i] = pow(SumDistance_L[i],Beta_L);
}

Z =  500000;

for(loopL=1;loopL<=MaxLoopL;loopL++){
ZQueen =  600000;
          For(Ant_L=1;Ant_L<=AntPopulationL;Ant_L++){
                    cout<<"loopL: "<<loopL<<'\t';
                    cout<<"Ant_L: "<<Ant_L<<'\n';

New:      Factory1 = OpenFactory(Pheromone_L,Alpha_L,Beta_L,SumDistance_L,No);

          for(i=1;i<=Total_Nod;i++){
          for(j=1;j<=Total_Nod;j++){
                    if(i!=j){
                              Pheromone_R[i][j] = 1.0;
                    }else{
                              Pheromone_R[i][j] = 0.0;
                    }
          }}

Z3 = 700000;

for(loopR=1;loopR<=MaxLoopR;loopR++){

Z2 = 800000;

for(Ant_R=1;Ant_R<=AntPopulationR;Ant_R++){

Reset(S1,DSent1,HSent1,DofRoute1,HofRoute1,MofRoute1,NofRoute1,TotalRoute1,
          SumDisRoute1,Z1);
SelectM(PossibleM,Factory1,S1,DSent1,DofRoute1,MofRoute1,TotalRoute1,SumDisRoute1,
          Pheromone_R,Alpha_R,Beta_R,No,false);

if(PossibleM == false){
          goto New;
}

NoLoopNImpossible = 0;

do{

SelectN(PossibleN,Factory1,S1,HSent1,HofRoute1,MofRoute1,NofRoute1,TotalRoute1,
          SumDisRoute1,Pheromone_R,Alpha_R,Beta_R,No,false);

if(PossibleN == 1){
          break;
}

NoLoopNImpossible += 1;

}while(NoLoopNImpossible <= 10);
if(PossibleN == false){
          Z1 = 9000000;
          goto New;
}
Z1 = CallDistance(false,TotalRoute1,S1,MofRoute1,NofRoute1);
if(Z1 < Z2){
          Z2 = Z1;
Update(Factory1,Factory2,S1,S2,DSent1,DSent2,HSent1,HSent2,DofRoute1,DofRoute2,
          HofRoute1,HofRoute2,MofRoute1,MofRoute2,NofRoute1,NofRoute2,TotalRoute1,
          TotalRoute2,SumDisRoute1,SumDisRoute2);
```

```
}
}//Ant_R//

if(Z2<Z3){
          Z3 = Z2;
Update(Factory2,Factory3,S2,S3,DSent2,DSent3,HSent2,HSent3,DofRoute2,DofRoute3,
          HofRoute2,HofRoute3,MofRoute2,MofRoute3,NofRoute2,NofRoute3,TotalRoute2,
          TotalRoute3,SumDisRoute2,SumDisRoute3);
}
for(i=1;i<=Total_Nod;i++){
          for(j=1;j<=Total_Nod;j++){
                    Pheromone_R[i][j]= Pheromone_R[i][j]*(1.0-EvaporationRate_R);
          }
}
for(i=1;i<=TotalRoute2;i++){
          for(j=1;j<=MofRoute2[i]+NofRoute2[i]+1;j++){
                    Pheromone_R[S2[i][j]][S2[i][j+1]] += 100.0/Z2;
          }
}
}//End loopR//

Exchage(S3,DSent3,HSent3,DofRoute3,HofRoute3,MofRoute3,NofRoute3,TotalRoute3,
          SumDisRoute3,Z3,true);
OneMove(Factory3,S3,DSent3,HSent3,DofRoute3,HofRoute3,MofRoute3,NofRoute3,TotalRoute3,
          SumDisRoute3,Z3);
TwoOptMove(S3,DSent3,HSent3,MofRoute3,NofRoute3,TotalRoute3,SumDisRoute3,Z3);

if(Z3<ZQueen){
          ZQueen = Z3;

Update(Factory3,FactoryQueen,S3,SQueen,DSent3,DSentQueen,HSent3,HSentQueen,DofRoute3,
          DofRouteQueen,HofRoute3,HofRouteQueen,MofRoute3,MofRouteQueen,NofRoute3,
          NofRouteQueen,TotalRoute3,TotalRouteQueen,SumDisRoute3,SumDisRouteQueen);
}

SumDisQueen[Ant_L] = Z3;
DummySumDisQueen[Ant_L]= Z3;
NoUpdate[Ant_L] = Factory3;

}//End Ant_L//

if(ZQueen < Z){
          Z = ZQueen;

Update(FactoryQueen,Factory,SQueen,S,DSentQueen,DSent,HSentQueen,HSent,DofRouteQueen,
          DofRoute,HofRouteQueen,HofRoute,MofRouteQueen,MofRoute,NofRouteQueen,
          NofRoute,TotalRouteQueen,TotalRoute,SumDisRouteQueen,SumDisRoute);
}

for(j=1;j<=16;j++){
          CheckDisL = 500000;
          for(i=1;i<=AntPopulationL;i++){
                    if(DummySumDisQueen[i]< CheckDisL){
                              CheckDisL = DummySumDisQueen[i];
                              UpdatePheL[j] = i;
                    }
          }
DummySumDisQueen[UpdatePheL[j]]= 500000;
}

for(i=1;i<=Total_Nod;i++){
          Pheromone_L[i]=Pheromone_L[i]*(1-EvaporationRate_L);
}

for(j=1;j<=16;j++){
          Pheromone_L[NoUpdate[UpdatePheL[j]]] +=  100.0/SumDisQueen[UpdatePheL[j]];
}

}//End loopL//

cout<<"Total Distance = "<<Z<<endl;
Show(Factory,S,DSent,HSent,DofRoute,HofRoute,MofRoute,NofRoute,TotalRoute,SumDisRoute,
          Z);
```

```
for(i=1;i<=Total_Nod;i++){
          for(j=1;j<=Total_Nod;j++){
                    if(i!=j){
                              Pheromone_R[i][j] = 1.0;
                    }else{
                              Pheromone_R[i][j] = 0.0;
                    }
          }
}
Z3 = 500000;

for(loopF=1;loopF<= MaxLoopFinal;loopF++){
          cout<<"loopF: "<<loopF<<endl;
          Z2 = 600000;
          for(Ant_F=1;Ant_F<=AntFinal;Ant_F++){
New2: Reset(S1,DSent1,HSent1,DofRoute1,HofRoute1,MofRoute1,NofRoute1,TotalRoute1,
          SumDisRoute1,Z1);
SelectM(PossibleM,Factory,S1,DSent1,DofRoute1,MofRoute1,TotalRoute1,SumDisRoute1,
          Pheromone_R,Alpha_R,Beta_R,No,false);

if(PossibleM == false){
          Z1 = 700000;
          cout<<"Error: Factory = "<<Factory<<"  **Impossible**!!!!!!!\n";
                    cout<<"Infeasible Customer\n";
goto New2;
}
NoLoopNImpossible = 0;

do{

SelectN(PossibleN,Factory,S1,HSent1,HofRoute1,MofRoute1,NofRoute1,TotalRoute1,
          SumDisRoute1,Pheromone_R,Alpha_R,Beta_R,No,false);

if(PossibleN == 1){
          break;
}

NoLoopNImpossible += 1;
}while(NoLoopNImpossible <= 10);
if(PossibleN == false){
          Z1 = 7000000;
          cout<<"Error: Factory = "<<Factory<<"  **Impossible**!!!!!!!\n";
          cout<<"Infeasible Supply\n";
          goto New2;
}

Z1 = CallDistance(false,TotalRoute1,S1,MofRoute1,NofRoute1);
Exchage(S1,DSent1,HSent1,DofRoute1,HofRoute1,MofRoute1,NofRoute1,TotalRoute1,
          SumDisRoute1,Z1,true);
OneMove(Factory,S1,DSent1,HSent1,DofRoute1,HofRoute1,MofRoute1,NofRoute1,TotalRoute1,
          SumDisRoute1,Z1);
TwoOptMove(S1,DSent1,HSent1,MofRoute1,NofRoute1,TotalRoute1,SumDisRoute1,Z1);

if(Z1 < Z2){
          Z2 = Z1;
Update(Factory,Factory2,S1,S2,DSent1,DSent2,HSent1,HSent2,DofRoute1,DofRoute2,
          HofRoute1,HofRoute2,MofRoute1,MofRoute2,NofRoute1,NofRoute2,TotalRoute1,
          TotalRoute2,SumDisRoute1,SumDisRoute2);
}
}//Ant_F//

if(Z2<Z3){
          Z3 = Z2;
Update(Factory,Factory3,S2,S3,DSent2,DSent3,HSent2,HSent3,DofRoute2,DofRoute3,
          HofRoute2,HofRoute3,MofRoute2,MofRoute3,NofRoute2,NofRoute3,TotalRoute2,
          TotalRoute3,SumDisRoute2,SumDisRoute3);
}
for(i=1;i<=Total_Nod;i++){
          for(j=1;j<=Total_Nod;j++){
                    Pheromone_R[i][j]= Pheromone_R[i][j]*(1.0-EvaporationRate_R);
          }
}
```

```
for(i=1;i<=TotalRoute2;i++){
          for(j=1;j<=MofRoute2[i]+NofRoute2[i]+1;j++){
                    Pheromone_R[S2[i][j]][S2[i][j+1]] += 100.0/Z2;
          }
}
}//End loopF//

if(Z3<Z){
          Z = Z3;
Update(Factory,Factory,S3,S,DSent3,DSent,HSent3,HSent,DofRoute3,DofRoute,HofRoute3,
          HofRoute,MofRoute3,MofRoute,NofRoute3,NofRoute,TotalRoute3,TotalRoute,
          SumDisRoute3,SumDisRoute);
}

myfile<<"\n######################### And ############################\n";
Show(Factory,S,DSent,HSent,DofRoute,HofRoute,MofRoute,NofRoute,TotalRoute,SumDisRoute,
          Z);

/*End AntLRP*/}

//*********************************  2-Opt Move  ******************************
void TwoOptMove(int**& Route,double**& DSent,double**& HSent,int*& MofRoute,
          int*& NofRoute,int& TotalRoute,double*&SumDisRoute,double&TotalDistance){

int i=0,j=0,m=0,n=0;
double MinDisImprove;
bool CheckImprove;
double NTD = TotalDistance;
int PositionOld,PositionNew;
int Dummy;
double DummyDH;
m=0;
int loop;
int* DummyRoute = new int [Total_Nod+1];
double* DummyD = new double [Total_Nod+1];
double* DummyH = new double [Total_Nod+1];
double CheckTotal;

for(loop=1;loop<=TotalRoute;loop++){
do{
          CheckImprove = false;
          MinDisImprove = TotalDistance;

if(MofRoute[loop] >= 2){

for(j=1;j<= MofRoute[loop]+1-2;j++){
          for(n=j+2;n<= MofRoute[loop]+1;n++){
                    NTD =  TotalDistance+Distance[Route[loop][j]][Route[loop][n]]+
                    Distance[Route[loop][j+1]][Route[loop][n+1]]-
                    Distance[Route[loop][j]][Route[loop][j+1]]-
                    Distance[Route[loop][n]][Route[loop][n+1]];
          if(NTD<MinDisImprove){
                    MinDisImprove = NTD;
                    PositionOld = j;
                    PositionNew = n;
                    CheckImprove = true;
          }
          }
}
}//if(MofRoute[loop] >= 2)

if(NofRoute[loop] >= 2){
for(j=MofRoute[loop]+1;j<= MofRoute[loop]+NofRoute[loop]+1-2;j++){
          for(n=j+2;n<= MofRoute[loop]+NofRoute[loop]+1;n++){
                    NTD =  TotalDistance+Distance[Route[loop][j]][Route[loop][n]]+
                    Distance[Route[loop][j+1]][Route[loop][n+1]]-
                    Distance[Route[loop][j]][Route[loop][j+1]]-
                    Distance[Route[loop][n]][Route[loop][n+1]];
          if(NTD<MinDisImprove){
                    MinDisImprove = NTD;
                    PositionOld = j;
                    PositionNew = n;
                    CheckImprove = true;
```

```
            }
            }
}
}

if(CheckImprove){
            for(i=1;i<=MofRoute[loop]+NofRoute[loop]+2;i++){
                        DummyRoute[i] = Route[loop][i];
                        DummyH[i] = HSent[loop][i];
            if(i<=4){
                        DummyD[i] = DSent[loop][i];
            }
}

SumDisRoute[loop] += MinDisImprove-TotalDistance;
TotalDistance = MinDisImprove;
Dummy = Route[loop][PositionNew];

if(Dummy<=M){
            DummyDH = DSent[loop][PositionNew];
            for(j=PositionOld+2;j<= PositionNew;j++){
                        Route[loop][j] = DummyRoute[PositionNew-1-m];
            if(j<=4){
                        DSent[loop][j] = DummyD[PositionNew-1-m];
            }
            m += 1;
            }
                        Route[loop][PositionOld+1] = Dummy;
                        DSent[loop][PositionOld+1] = DummyDH;
            }else{
                        DummyDH = HSent[loop][PositionNew];
            for(j=PositionOld+2;j<= PositionNew;j++){
                        Route[loop][j] = DummyRoute[PositionNew-1-m];
                        HSent[loop][j] = DummyH[PositionNew-1-m];
                        m += 1;
            }
                        Route[loop][PositionOld+1] = Dummy;
                        HSent[loop][PositionOld+1] = DummyDH;
            }

m = 0;
}
}while(CheckImprove);
}
delete []DummyRoute;
delete []DummyD;
delete []DummyH;

}

//********************************  Exchage Move  ********************************
void Exchage(int**& Route,double**& DSent,double**& HSent,double*& DofRoute,
            double*& HofRoute,int*& MofRoute,int*& NofRoute,int& TotalRoute,
            double*&SumDisRoute,double&TotalDistance,bool Test){

int i=0,j=0,m=0,n=0;
double CheckDis1,CheckDis2;
double CheckCapa1,CheckCapa2;
double NewDisRoute1,NewDisRoute2;
double MinDisImprove;
bool CheckImprove;
double NTD = TotalDistance;
int RouteOld,RouteNew;
int PositionOld,PositionNew;
int Dummy;
double DummyDH;
double CheckTotalDistance;
int TTT;

do{
            CheckImprove = false;
            MinDisImprove = TotalDistance;
```

```
for(i=1;i<=TotalRoute-1;i++){
for(j=2;j<=MofRoute[i]+2;j++){
          for(m=i+1;m<=TotalRoute;m++){
          for(n=2;n<=MofRoute[m]+2;n++){
          if(Route[i][j]<= M){
                    if(Route[m][n]<=M){
                              CheckCapa1 = DofRoute[i]-DSent[i][j]+DSent[m][n];
                              CheckCapa2 = DofRoute[m]-DSent[m][n]+DSent[i][j];
                    }else if((j==MofRoute[i]+1)&&(MofRoute[i]>=1)&&
                    (Route[m][n]!= Route[i][1])){
                              CheckCapa1 = HofRoute[i]+HSent[m][n];
                              CheckCapa2 = DofRoute[m]+DSent[i][j];
                    }else{
                              CheckCapa1 = 5000; CheckCapa2 = 5000;}
          }else if((n == MofRoute[m]+1)&&(MofRoute[m]>=1)&&
          (Route[i][j] != Route[i][1])){
                    CheckCapa1 = DofRoute[i]+DSent[m][n];
                    CheckCapa2 = HofRoute[m]+HSent[i][j];
          }else{
                    CheckCapa1 = 5000; CheckCapa2 = 5000;
          }
CheckDis1 = SumDisRoute[i]+Distance[Route[i][j-1]][Route[m][n]]+
          Distance[Route[m][n]][Route[i][j+1]]-Distance[Route[i][j-1]][Route[i][j]]-
          Distance[Route[i][j]][Route[i][j+1]];
CheckDis2 = SumDisRoute[m]+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n+1]]-Distance[Route[m][n-1]][Route[m][n]]-
          Distance[Route[m][n]][Route[m][n+1]];

if((CheckDis1<=MaxDistance)&&(CheckDis2<=MaxDistance)&&(CheckCapa1<=Qk)&&
          (CheckCapa2<=Qk)){

NTD = TotalDistance+Distance[Route[i][j-1]][Route[m][n]]+
          Distance[Route[m][n]][Route[i][j+1]]+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n+1]]-Distance[Route[i][j-1]][Route[i][j]]-
          Distance[Route[i][j]][Route[i][j+1]]-Distance[Route[m][n-1]][Route[m][n]]-
          Distance[Route[m][n]][Route[m][n+1]];

if(NTD<MinDisImprove){
          MinDisImprove = NTD;
          RouteOld = i;
          RouteNew = m;
          PositionOld = j;
          PositionNew = n;
          NewDisRoute1 = CheckDis1;
          NewDisRoute2 = CheckDis2;
          CheckImprove = true;
}
}
}
}
}
}//End for(i=1;i<=TotalRoute;i++) and if(MofRoute[i]>=1)//

for(i=1;i<=TotalRoute-1;i++){
          if(NofRoute[i]>=1){
          for(j=MofRoute[i]+2;j<=MofRoute[i]+NofRoute[i]+1;j++){
          for(m=i+1;m<=TotalRoute;m++){
                    if(NofRoute[m]>=1){
                    for(n=MofRoute[m]+2;n<=MofRoute[m]+NofRoute[m]+1;n++){
                              CheckCapa1 = HofRoute[i]-HSent[i][j]+HSent[m][n];
                              CheckCapa2 = HofRoute[m]-HSent[m][n]+HSent[i][j];

CheckDis1 = SumDisRoute[i]+Distance[Route[i][j-1]][Route[m][n]]+
          Distance[Route[m][n]][Route[i][j+1]]-Distance[Route[i][j-1]][Route[i][j]]-
          Distance[Route[i][j]][Route[i][j+1]];

CheckDis2 = SumDisRoute[m]+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n+1]]-Distance[Route[m][n-1]][Route[m][n]]-
          Distance[Route[m][n]][Route[m][n+1]];

if((CheckDis1<=MaxDistance)&&(CheckDis2<=MaxDistance)&&(CheckCapa1<=Qk)&&
          (CheckCapa2<=Qk)){
```

```
NTD = TotalDistance+Distance[Route[i][j-1]][Route[m][n]]+
          Distance[Route[m][n]][Route[i][j+1]]+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n+1]]-Distance[Route[i][j-1]][Route[i][j]]-
          Distance[Route[i][j]][Route[i][j+1]]-Distance[Route[m][n-1]][Route[m][n]]-
          Distance[Route[m][n]][Route[m][n+1]];

if(NTD<MinDisImprove){
          MinDisImprove = NTD;
          RouteOld = i;
          RouteNew = m;
          PositionOld = j;
          PositionNew = n;
          NewDisRoute1 = CheckDis1;
          NewDisRoute2 = CheckDis2;
          CheckImprove = true;
}
}
}
}
}
}
}}//End for(i=1;i<=TotalRoute;i++) and if(NofRoute[i]>=1)//

if(CheckImprove){
          TotalDistance = MinDisImprove;
          SumDisRoute[RouteOld] = NewDisRoute1;
          SumDisRoute[RouteNew] = NewDisRoute2;
          Dummy = Route[RouteOld][PositionOld];
if((Dummy<=M)||(Route[RouteNew][PositionNew]<=M)){
          if((Dummy<=M)){
          if(Route[RouteNew][PositionNew]<=M){
                    DummyDH = DSent[RouteOld][PositionOld];
                    DofRoute[RouteOld] += DSent[RouteNew][PositionNew]-
                              DSent[RouteOld][PositionOld];
                    DofRoute[RouteNew] += DSent[RouteOld][PositionOld]-
                              DSent[RouteNew][PositionNew];
                    DSent[RouteOld][PositionOld] = DSent[RouteNew][PositionNew];
                    DSent[RouteNew][PositionNew] = DummyDH;
          }else{
                    HofRoute[RouteOld] += HSent[RouteNew][PositionNew];
                    HofRoute[RouteNew] -= HSent[RouteNew][PositionNew];
                    DofRoute[RouteNew] += DSent[RouteOld][PositionOld];
                    DofRoute[RouteOld] -= DSent[RouteOld][PositionOld];
                    HSent[RouteOld][PositionOld] = HSent[RouteNew][PositionNew];
                    HSent[RouteNew][PositionNew] = 0;
                    DSent[RouteNew][PositionNew] = DSent[RouteOld][PositionOld];
                    DSent[RouteOld][PositionOld] = 0;
                    NofRoute[RouteOld] += 1;
                    NofRoute[RouteNew] -= 1;
                    MofRoute[RouteNew] += 1;
                    MofRoute[RouteOld] -= 1;

          }
          }else{
                    DofRoute[RouteOld] += DSent[RouteNew][PositionNew];
                    DofRoute[RouteNew] -= DSent[RouteNew][PositionNew];
                    HofRoute[RouteNew] += HSent[RouteOld][PositionOld];
                    HofRoute[RouteOld] -= HSent[RouteOld][PositionOld];
                    DSent[RouteOld][PositionOld] = DSent[RouteNew][PositionNew];
                    DSent[RouteNew][PositionNew] = 0;
                    HSent[RouteNew][PositionNew] = HSent[RouteOld][PositionOld];
                    HSent[RouteOld][PositionOld] = 0;
                    NofRoute[RouteOld] -= 1;
                    NofRoute[RouteNew] += 1;
                    MofRoute[RouteNew] -= 1;
                    MofRoute[RouteOld] += 1;
          }
          }else{
                    HofRoute[RouteOld] += HSent[RouteNew][PositionNew]-
                              HSent[RouteOld][PositionOld];
                    HofRoute[RouteNew] += HSent[RouteOld][PositionOld]-
                              HSent[RouteNew][PositionNew];
                    DummyDH = HSent[RouteOld][PositionOld];
```

```
                    HSent[RouteOld][PositionOld] = HSent[RouteNew][PositionNew];
                    HSent[RouteNew][PositionNew] = DummyDH;
          }
          Route[RouteOld][PositionOld] = Route[RouteNew][PositionNew];
          Route[RouteNew][PositionNew] = Dummy;
}
}while(CheckImprove);

/*End Exchage*/}

//********************************  One Move  *********************************
void OneMove(int Factory,int**& Route,double**& DSent,double**& HSent,
          double*& DofRoute,double*& HofRoute,int*& MofRoute,int*& NofRoute,
          int& TotalRoute,double*&SumDisRoute,double&TotalDistance){
int i=0,j=0,m=0,n=0;
double NewDisRoute1,NewDisRoute2;
double CheckDis1,CheckDis2;
double CheckCapa;
double MinDisImprove;
bool CheckImprove;
double NTD = TotalDistance;
int RouteOld,RouteNew;
int PositionOld,PositionNew;
int Dummy;
double DummyDH;
int TTT=0;
TotalRoute += 1;
Route[TotalRoute][1]=Factory;
Route[TotalRoute][2]=Factory;
SumDisRoute[TotalRoute]=0;
MofRoute[TotalRoute]=0;
NofRoute[TotalRoute]=0;
DofRoute[TotalRoute]=0;
HofRoute[TotalRoute]=0;

do{
          CheckImprove = false;
          MinDisImprove = TotalDistance;
for(i=1;i<=TotalRoute;i++){
if(MofRoute[i]>=1){
for(j=2;j<=MofRoute[i]+1;j++){
for(m=1;m<=TotalRoute;m++){
if((m==i)&&(MofRoute[m]>1)){
for(n=2;n<=MofRoute[m]+1;n++){
if(j!= n){
if(j<n){

NTD = TotalDistance+Distance[Route[i][j-1]][Route[i][j+1]]+
          Distance[Route[m][n]][Route[i][j]]+Distance[Route[i][j]][Route[m][n+1]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]]-
          Distance[Route[m][n]][Route[m][n+1]];
}else{

NTD = TotalDistance+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n]]+Distance[Route[i][j-1]][Route[i][j+1]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]]-
          Distance[Route[m][n-1]][Route[m][n]];
}

if(NTD<MinDisImprove){
          MinDisImprove = NTD;
          RouteOld = i;
          RouteNew = m;
          PositionOld =j;
          PositionNew =n;
          CheckImprove = true;
}
}
}
}else if(m != i){
```

```
for(n=2;n<=MofRoute[m]+2;n++){

CheckDis1 = SumDisRoute[i]+Distance[Route[i][j-1]][Route[i][j+1]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]];

CheckDis2 = SumDisRoute[m]+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n]]-Distance[Route[m][n-1]][Route[m][n]];

CheckCapa = DofRoute[m]+DSent[i][j];

if((CheckDis1<=MaxDistance)&&(CheckDis2<=MaxDistance)&&(CheckCapa<=Qk)){

NTD =  TotalDistance+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n]]+Distance[Route[i][j-1]][Route[i][j+1]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]]-
          Distance[Route[m][n-1]][Route[m][n]];
}

if(NTD<MinDisImprove){
          MinDisImprove = NTD;
          RouteOld = i;
          RouteNew = m;
          PositionOld =j;
          PositionNew =n;
          NewDisRoute1 =  CheckDis1;
          NewDisRoute2 =  CheckDis2;
          CheckImprove = true;
}
}
}
}
}
}}//End for(i=1;i<=TotalRoute;i++) and if(MofRoute[i]>=1)//

for(i=1;i<=TotalRoute;i++){
if(NofRoute[i]>=1){
for(j=MofRoute[i]+2;j<=MofRoute[i]+NofRoute[i]+1;j++){
for(m=1;m<=TotalRoute;m++){
if((m==i)&&(NofRoute[i]>1)){
for(n=MofRoute[m]+2;n<=MofRoute[m]+NofRoute[m]+1;n++){
if(j!= n){
if(j<n){

NTD = TotalDistance+Distance[Route[i][j-1]][Route[i][j+1]]+
          Distance[Route[m][n]][Route[i][j]]+Distance[Route[i][j]][Route[m][n+1]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]]-
          Distance[Route[m][n]][Route[m][n+1]];

}else{

NTD = TotalDistance+Distance[Route[i][j-1]][Route[i][j+1]]+
          Distance[Route[m][n-1]][Route[i][j]]+Distance[Route[i][j]][Route[m][n]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]]-
          Distance[Route[m][n-1]][Route[m][n]];
}

if(NTD<MinDisImprove){
          MinDisImprove = NTD;
          RouteOld = i;
          RouteNew = m;
          PositionOld =j; PositionNew =n;
          CheckImprove = true;
}
}}
}else if(m != i){

for(n=MofRoute[m]+2;n<=MofRoute[m]+NofRoute[m]+2;n++){

CheckDis1 = SumDisRoute[i]+Distance[Route[i][j-1]][Route[i][j+1]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]];
CheckDis2 = SumDisRoute[m]+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n]]-Distance[Route[m][n-1]][Route[m][n]];
```

```
CheckCapa = HofRoute[m]+HSent[i][j];

if((CheckDis1<=MaxDistance)&&(CheckDis2<=MaxDistance)&&(CheckCapa<=Qk)){

NTD =  TotalDistance+Distance[Route[m][n-1]][Route[i][j]]+
          Distance[Route[i][j]][Route[m][n]]+Distance[Route[i][j-1]][Route[i][j+1]]-
          Distance[Route[i][j-1]][Route[i][j]]-Distance[Route[i][j]][Route[i][j+1]]-
          Distance[Route[m][n-1]][Route[m][n]];
}

if(NTD<MinDisImprove){
          MinDisImprove = NTD;
          RouteOld = i;
          RouteNew = m;
          PositionOld =j;
          PositionNew =n;
          NewDisRoute1 =  CheckDis1;
          NewDisRoute2 =  CheckDis2;
          CheckImprove = true;
}
}
}}
}
}}//End for(i=1;i<=TotalRoute;i++) and if(NofRoute[i]>=1)//

if(CheckImprove == true){
if(RouteOld == RouteNew){
if(PositionOld<PositionNew){
          Dummy = Route[RouteOld][PositionOld];
if(Dummy <= M){
          DummyDH = DSent[RouteOld][PositionOld];
for(i=PositionOld;i<=PositionNew-1;i++){
          Route[RouteOld][i] = Route[RouteOld][i+1];
if(i<=3){
          DSent[RouteOld][i]= DSent[RouteOld][i+1];
}
}
          DSent[RouteNew][PositionNew] = DummyDH;
          Route[RouteNew][PositionNew] = Dummy;
}else{
          DummyDH = HSent[RouteOld][PositionOld];
for(i=PositionOld;i<=PositionNew-1;i++){
          Route[RouteOld][i] = Route[RouteOld][i+1];
          HSent[RouteOld][i]= HSent[RouteOld][i+1];
}
          HSent[RouteNew][PositionNew] = DummyDH;
          Route[RouteNew][PositionNew] = Dummy;
}
}else{
          Dummy = Route[RouteOld][PositionOld];
if(Dummy <= M){
          DummyDH = DSent[RouteOld][PositionOld];
for(i=PositionOld;i>= PositionNew+1;i--){
          Route[RouteOld][i] = Route[RouteOld][i-1];
if(i<=4){
          DSent[RouteOld][i] = DSent[RouteOld][i-1];
}
}
          DSent[RouteNew][PositionNew] = DummyDH;
          Route[RouteNew][PositionNew] = Dummy;
}else{
          DummyDH = HSent[RouteOld][PositionOld];

for(i=PositionOld;i>= PositionNew+1;i--){
          Route[RouteOld][i] = Route[RouteOld][i-1];
          HSent[RouteOld][i] = HSent[RouteOld][i-1];
}
          HSent[RouteNew][PositionNew] = DummyDH;
          Route[RouteNew][PositionNew] = Dummy;
}
}
          SumDisRoute[RouteOld] +=  MinDisImprove-TotalDistance;
          TotalDistance =  MinDisImprove;
```

```
}else{
            Dummy = Route[RouteOld][PositionOld];
if(Dummy <= M){
            DofRoute[RouteOld] -= DSent[RouteOld][PositionOld];
            DofRoute[RouteNew] += DSent[RouteOld][PositionOld];
            DummyDH = DSent[RouteOld][PositionOld];
            MofRoute[RouteOld] -=1;
            MofRoute[RouteNew] +=1;

for(i=PositionOld;i<=MofRoute[RouteOld]+NofRoute[RouteOld]+2;i++){
            Route[RouteOld][i] = Route[RouteOld][i+1];
            HSent[RouteOld][i] = HSent[RouteOld][i+1];

if(i<=3){
            DSent[RouteOld][i] = DSent[RouteOld][i+1];
}
}
HSent[RouteOld][MofRoute[RouteOld]+NofRoute[RouteOld]+2]=0;

for(i=MofRoute[RouteNew]+NofRoute[RouteNew]+2;i>=PositionNew;i--){
            Route[RouteNew][i] = Route[RouteNew][i-1];
            HSent[RouteNew][i] = HSent[RouteNew][i-1];
if(i<=4){
            DSent[RouteNew][i] = DSent[RouteNew][i-1];
}
}
            Route[RouteNew][PositionNew] = Dummy;
            DSent[RouteNew][PositionNew] = DummyDH;
            HSent[RouteNew][MofRoute[RouteNew]+1] = 0;
}else{
            HofRoute[RouteOld] -= HSent[RouteOld][PositionOld];
            HofRoute[RouteNew] += HSent[RouteOld][PositionOld];
            DummyDH = HSent[RouteOld][PositionOld];
            NofRoute[RouteOld] -=1;
            NofRoute[RouteNew] +=1;

for(i=PositionOld;i<=MofRoute[RouteOld]+NofRoute[RouteOld]+2;i++){
            Route[RouteOld][i] = Route[RouteOld][i+1];
            HSent[RouteOld][i] = HSent[RouteOld][i+1];
}

for(i=MofRoute[RouteNew]+NofRoute[RouteNew]+2;i>=PositionNew;i--){
            Route[RouteNew][i] = Route[RouteNew][i-1];
            HSent[RouteNew][i] = HSent[RouteNew][i-1];
}
            Route[RouteNew][PositionNew] = Dummy;
            HSent[RouteNew][PositionNew] = DummyDH;
}
            SumDisRoute[RouteOld] = NewDisRoute1;
            SumDisRoute[RouteNew] = NewDisRoute2;
            TotalDistance =  MinDisImprove;
}
}
}while(CheckImprove);
int Old=1,New=TotalRoute;
do{
while(MofRoute[New]+NofRoute[New]==0){
            New -=1;
            TotalRoute -= 1;
}

if(MofRoute[Old]+NofRoute[Old]==0){
            TotalRoute -= 1;
            MofRoute[Old] = MofRoute[New];
            NofRoute[Old] = NofRoute[New];
            DofRoute[Old] = DofRoute[New];
            HofRoute[Old] = HofRoute[New];
            SumDisRoute[Old] = SumDisRoute[New];

for(i=1;i<=MofRoute[New]+NofRoute[New]+2;i++){
              Route[Old][i] = Route[New][i];
              HSent[Old][i] = HSent[New][i];
}
```

```
for(i=1;i<=4;i++){
          DSent[Old][i] = DSent[New][i];
}
          New -= 1;
}
          Old += 1;
}while(New > Old);
}

//****************************  Calculate Didtance  **************************
double CallDistance(bool Show,int TotalRoute,int** Route,int* MofRoute,int* NofRoute){

int i=0,j=0;
double TotalDistance = 0;
double*SumDisRoute = new double [TotalRoute+1];

for(i=1;i<=TotalRoute;i++){
          SumDisRoute[i] = 0;
          for(j=1;j<=MofRoute[i]+NofRoute[i]+1;j++){
                    SumDisRoute[i] += Distance[Route[i][j]][Route[i][j+1]];
                    TotalDistance += Distance[Route[i][j]][Route[i][j+1]];
          }
}

if(Show){
          for(i=1;i<=TotalRoute;i++){
                     cout<<"SumDisRoute["<<i<<"] =\t"<<SumDisRoute[i]<<endl;
                   myfile<<"SumDisRoute["<<i<<"] =\t"<<SumDisRoute[i]<<endl;
          }
}

delete []SumDisRoute;
return TotalDistance;

}


//****************************  Update  **************************************
void Update(int FactoryModel,int& Factory,int** RouteModel,int**& Route,
          double** DSentModel,double**& DSent,double** HSentModel,
          double**& HSent,double* DofRouteModel,double*& DofRoute,
          double* HofRouteModel,double*& HofRoute,int* MofRouteModel,
          int*& MofRoute,int* NofRouteModel,int*& NofRoute,int TotalRouteModel,
          int& TotalRoute,double*SumDisRouteModel,double*&SumDisRoute){

int i=0,j=0;
Factory = FactoryModel;
TotalRoute = TotalRouteModel;

for(i=1;i<=TotalRouteModel;i++){
          MofRoute[i] = MofRouteModel[i];
          NofRoute[i] = NofRouteModel[i];
          SumDisRoute[i] = SumDisRouteModel[i];
          DofRoute[i] = DofRouteModel[i];
          HofRoute[i] = HofRouteModel[i];
for(j=1;j<= MofRouteModel[i]+NofRouteModel[i]+2;j++){
          Route[i][j] = RouteModel[i][j];
          HSent[i][j] = HSentModel[i][j];
}
for(j=1;j<=4;j++){
          DSent[i][j] = DSentModel[i][j];
}
}
}

//****************************  Selection N  *********************************
void SelectN(bool&Possible,int Factory,int**& Route,double**& HSent,
          double*& HofRoute,int*& MofRoute,int*& NofRoute,int& TotalRoute,
          double*&SumDisRoute,double** Pheromone_R,int Alpha_R,int Beta_R,int& No,
          bool Test){

int i=0,j=0;
int NumberN;
```

```
int Number = 0;
int* NRemaining = new int [N];
int* NPossible =  new int [N+1];
int NoNPossible = 0;
double* Attractiveness =  new double [N+1];
double SumAttractiveness = 0;
double *HRemaining = new double [M+N+1];
double SumSupply = H[Factory];
int NoRouteN = 1;
int NoPointN = 3;
int ReferencePoint;
double CheckDistance;
double CheckSupply;
double HTruck;
int NEqualM[9] = {0,78,113,117,123,127,149,154,161};
int MEqualN[9] = {0,7,13,15,9,10,12,11,14};
int** RouteBegin = (int**)calloc(Total_Nod+1,sizeof(int*));
int* MofRouteBegin = new int [Total_Nod+1];
int  TotalRouteBegin;
double* SumDisRouteBegin = new double [Total_Nod+1];

for(i=1;i<=Total_Nod;i++){
          RouteBegin[i] =  (int*)calloc(Total_Nod+1,sizeof(int));
}
TotalRouteBegin = TotalRoute;

for(i=1;i<=TotalRoute;i++){
          MofRouteBegin[i] = MofRoute[i];
          SumDisRouteBegin[i] = SumDisRoute[i];
for(j=1;j<= MofRouteBegin[i]+1;j++){
          RouteBegin[i][j] = Route[i][j];
}
}

int TTT = 0;
ReferencePoint = Route[1][2];
Possible = true;

for(i=1;i<=150;i++){
          HofRoute[i] = 0;
          NofRoute[i] = 0;
for(j=1;j<=150;j++){
          HSent[i][j] = 0;
}
}

for(i=M+1;i< Factory ;i++){
          Number += 1;
          NRemaining[Number] = i;
          HRemaining[i]= H[i];
}

for(i = Factory+1;i<= M+N;i++){
          Number += 1;
          NRemaining[Number] = i;
          HRemaining[i]= H[i];
}
do{
          NumberN = 0;
if(ReferencePoint <= M){
for(i=1;i<=8;i++){
if(ReferencePoint == MEqualN[i]){
if(HRemaining[NEqualM[i]]>0){
          NumberN = NEqualM[i];
          break;
}
}
}
}

if(NumberN == 0){
          NoNPossible = 0;
for(i=1;i<=Number;i++){
```

```
if(HRemaining[NRemaining[i]]< Qk){
          HTruck =  HRemaining[NRemaining[i]];
}else{
          HTruck = Qk;
}

CheckDistance = SumDisRoute[NoRouteN]+Distance[ReferencePoint][NRemaining[i]]+
          Distance[NRemaining[i]][Factory];
CheckSupply = HofRoute[NoRouteN]+HTruck;

if((CheckDistance <= MaxDistance)&&(CheckSupply <= Qk)){
          NoNPossible += 1;
          NPossible[NoNPossible] = NRemaining[i];
}
}

SumAttractiveness = 0;
if(NoNPossible > 0){

for(i=1;i<=NoNPossible;i++){

Attractiveness[i] = pow(Pheromone_R[ReferencePoint][NPossible[i]],Alpha_R)*
          pow(1.0/Distance[ReferencePoint][NPossible[i]],Beta_R);
SumAttractiveness += Attractiveness[i];
}

NumberN =  CalPop(1,NoNPossible,Attractiveness,SumAttractiveness,No,false);
NumberN =  NPossible[NumberN];

}else{

if(NoRouteN < TotalRoute){
          SumDisRoute[NoRouteN] += Distance[ReferencePoint][Factory];
          Route[NoRouteN][NoPointN] =  Factory;
          NoRouteN +=1;
          goto Update;
}else{

if(NoPointN > 2){
          SumDisRoute[NoRouteN] += Distance[ReferencePoint][Factory];
          Route[NoRouteN][NoPointN] =  Factory;
          NoRouteN +=1;
          goto Update;
}else{
          Possible = false;
          goto Fail;
}
}
}
}//if(NumberN == 0)//

if(HRemaining[NumberN]< Qk){
          HTruck =  HRemaining[NumberN];
}else{
          HTruck = Qk;
}

if( V-SumSupply < HTruck){
          HTruck = V-SumSupply;
}
          HRemaining[NumberN] -= HTruck;

if(HRemaining[NumberN]<= 0){
for(i=1;i<=Number;i++){
          if(NRemaining[i]== NumberN){
          for(j=i;j<=Number-1;j++){
                    NRemaining[j]= NRemaining[j+1];
          }
                    Number -= 1;
                    break;
          }
}
}
```

```
SumDisRoute[NoRouteN] += Distance[ReferencePoint][NumberN];
Route[NoRouteN][NoPointN] =  NumberN;
NofRoute[NoRouteN] += 1;
HofRoute[NoRouteN] += HTruck;
HSent[NoRouteN][NoPointN] = HTruck;
SumSupply += HTruck;

Update:
if(NoRouteN <= TotalRoute){
          NoPointN = 1+MofRoute[NoRouteN]+NofRoute[NoRouteN]+1;
          ReferencePoint =  Route[NoRouteN][NoPointN-1];
}else{
          TotalRoute += 1;
          NoPointN = 2;
          Route[NoRouteN][1] = Factory;
          SumDisRoute[NoRouteN] = 0;
          ReferencePoint =  Factory;
          MofRoute[NoRouteN] = 0;
          HofRoute[NoRouteN] = 0;
          NofRoute[NoRouteN] = 0;
}

Fail:
if(Possible == false){
          TotalRoute = TotalRouteBegin;
for(i=1;i<=TotalRoute;i++){
          HofRoute[i] = 0;
          SumDisRoute[i] = SumDisRouteBegin[i];
for(j=1;j<= MofRoute[i]+1;j++){
          HSent[i][j] = 0;
          Route[i][j] = RouteBegin[i][j];
}
          MofRoute[i] = MofRouteBegin[i];
          NofRoute[i] = 0;
}
break;
}
}while(SumSupply < V);
if(NofRoute[TotalRoute]<=0){
          TotalRoute -= 1;
}

if(Route[TotalRoute][MofRoute[TotalRoute]+NofRoute[TotalRoute]+2] <= 0){
          Route[TotalRoute][MofRoute[TotalRoute]+NofRoute[TotalRoute]+2]=Factory;
          SumDisRoute[TotalRoute] += Distance[Route[TotalRoute][MofRoute[TotalRoute]+
                    NofRoute[TotalRoute]+1]][Factory];
}
delete []NRemaining;
delete []NPossible;
delete []Attractiveness;
delete []HRemaining;
for(i=1;i<=Total_Nod;i++){
delete[] RouteBegin[i];
}
delete[] RouteBegin;
delete []MofRouteBegin;
delete []SumDisRouteBegin;

/*End Selection N*/}

//*****************************  Selection M  *********************************
void SelectM(bool&Possible,int Factory,int**& Route,double**& DSent,double*& DofRoute,
          int*& MofRoute,int& TotalRoute,double*&SumDisRoute,double** Pheromone_R,
          int Alpha_R,int Beta_R,int& No,bool Test){

int i=0,j=0;
int*NEqualM = new int [9];
NEqualM[1] = 78;
NEqualM[2] = 113;
NEqualM[3] = 117;
NEqualM[4] = 123;
NEqualM[5] = 127;
NEqualM[6] = 149;
```

```
NEqualM[7] = 154;
NEqualM[8] = 161;
int*MEqualN = new int [9];
MEqualN[1]=7;
MEqualN[2]=13;
MEqualN[3]=15;
MEqualN[4]=9;
MEqualN[5]=10;
MEqualN[6]=12;
MEqualN[7]=11;
MEqualN[8]=14;
int NumberM;
int NoTruck;
double DRemaining;
double SumDemand;
double* Attractiveness =  new double [M+1];
double SumAttractiveness;
int* MPossible =  new int [M+1];
int NoMPossible;

AAA:
          TotalRoute = 0;
          NumberM = 0;
          NoMPossible = 0;
          SumAttractiveness = 0;

for(i=1;i<=(Capa/Qk)+2;i++){
          Route[i][1] =  Factory;
}

//***********  Check M Possible ************

NoMPossible = 0;
for(i=1;i<=M;i++){
if(Distance[Factory][i]*2<= MaxDistance){
          NoMPossible += 1;
          MPossible[NoMPossible] = i;
}
}

if(Test == true){
          myfile<<"NoMPossible: "<<NoMPossible<<endl;
}

if(NoMPossible == 0){
          Possible = false;
}else{
          Possible = true;
          NumberM = 0;
for(i=1;i<=8;i++){
if(Factory == NEqualM[i]){
          NumberM = MEqualN[i];
          DRemaining = D[MEqualN[i]];
          break;
}
}
if(Test == true){
          myfile<<"NumberM: "<<NumberM<<endl;
}

if(NumberM == 0){
          SumAttractiveness = 0;
for(i=1;i<=NoMPossible;i++){
Attractiveness[i] = pow(Pheromone_R[Factory][MPossible[i]],Alpha_R)*
          pow(100.0/Distance[Factory][MPossible[i]],Beta_R);
SumAttractiveness += Attractiveness[i];
}

NumberM = CalPop(1,NoMPossible,Attractiveness,SumAttractiveness,No,false);
NumberM = MPossible[NumberM];
DRemaining = D[NumberM];
```

```
if(Test == true){
          myfile<<"NumberM=0,NumberM:"<<NumberM<<endl;
}
}

if(Test == true){
          myfile<<"DRemaining:"<<DRemaining<<endl;
}

AddM:
if(DRemaining >= Capa){
          TotalRoute = Capa/Qk;
          SumDemand = TotalRoute*Qk;
for(i=1;i<= TotalRoute;i++){
          SumDisRoute[i] = Distance[Factory][NumberM];
          Route[i][2] =  NumberM;
          MofRoute[i] = 1;
          DofRoute[i] = Qk;
          DSent[i][2] = Qk;
}

if(SumDemand < Capa){
          TotalRoute += 1;
          SumDisRoute[TotalRoute] = Distance[Factory][NumberM];
          Route[TotalRoute][2] =  NumberM;
          MofRoute[TotalRoute] = 1;
          DofRoute[TotalRoute] = Capa-SumDemand;
          DSent[TotalRoute][2] = Capa-SumDemand;
          SumDemand += Capa-SumDemand;
}

if(Test == true){
          myfile<<"DRemaining>=Capa,SumDemand:"<<SumDemand<<endl;
}
}else{
          TotalRoute = DRemaining/Qk;
          SumDemand = TotalRoute*Qk;

for(i=1;i<= TotalRoute;i++){
          SumDisRoute[i] = Distance[Factory][NumberM];
          Route[i][2] =  NumberM;
          MofRoute[i] = 1;
          DofRoute[i] = Qk;
          DSent[i][2] = Qk;
}

if(SumDemand < DRemaining){
          TotalRoute += 1;
          SumDisRoute[TotalRoute] = Distance[Factory][NumberM];
          Route[TotalRoute][2] =  NumberM;
          MofRoute[TotalRoute] = 1;
          DofRoute[TotalRoute] = DRemaining-SumDemand;
          DSent[TotalRoute][2] = DRemaining-SumDemand;
          SumDemand += DRemaining-SumDemand;
}

if(Test == true){
          myfile<<"else DRemaining>=Capa,SumDemand:"<<SumDemand<<endl;
}
SumAttractiveness = 0;

for(i=1;i<=NoMPossible;i++){
if(MPossible[i] != NumberM){
          Attractiveness[i] = pow(Pheromone_R[Factory][MPossible[i]],Alpha_R)*
                    pow(100.0/Distance[Factory][MPossible[i]],Beta_R);
          SumAttractiveness += Attractiveness[i];
}else{ Attractiveness[i] = 0;
}
}

NumberM = CalPop(1,NoMPossible,Attractiveness,SumAttractiveness,No,false);
NumberM = MPossible[NumberM];
```

```
if(Test == true){
          myfile<<"NumberM2: "<<NumberM<<endl;
}

if(Test == true){
          myfile<<"Qk-DofRoute[TotalRoute]: "<<Qk-DofRoute[TotalRoute]<<
          "\tCapa-SumDemand: "<<Capa-SumDemand<<endl;
}

if(Qk-DofRoute[TotalRoute]<= Capa-SumDemand){
if(Test == true){
          myfile<<"Qk-DofRoute[TotalRoute]<= Capa-SumDemand"<<endl;
}

if(Qk-DofRoute[TotalRoute]>0){
if(Distance[Factory][Route[TotalRoute][2]]+Distance[Route[TotalRoute][2]][NumberM]+
          Distance[NumberM][Factory]<= MaxDistance){
                    Route[TotalRoute][3] =  NumberM;
                    MofRoute[TotalRoute] = 2;
                    SumDemand += Qk-DofRoute[TotalRoute];
                    DSent[TotalRoute][3] = Qk-DofRoute[TotalRoute];
                    DofRoute[TotalRoute] += Qk-DofRoute[TotalRoute];
                    SumDisRoute[TotalRoute] = Distance[Factory][Route[TotalRoute][2]]+
                              Distance[Route[TotalRoute][2]][NumberM];
if(Test == true){
          myfile<<"Dis<=MaxDistance,SumDemand: "<<SumDemand<<endl;
}

}}

if(Capa-SumDemand > 0){
          NoTruck = (Capa-SumDemand)/Qk;
if(Test == true){
          myfile<<"Capa-SumDemand!=0,NoTruck: "<<NoTruck<<endl;
}

for(i=1;i<=NoTruck;i++){
          TotalRoute += 1;
          SumDisRoute[TotalRoute] = Distance[Factory][NumberM];
          Route[TotalRoute][2] =  NumberM;
          MofRoute[TotalRoute] = 1;
          DofRoute[TotalRoute] = Qk;
          DSent[TotalRoute][2] = Qk;
          SumDemand += Qk;
}

if(SumDemand < Capa){
          TotalRoute += 1;
          SumDisRoute[TotalRoute] = Distance[Factory][NumberM];
          Route[TotalRoute][2] =  NumberM;
          MofRoute[TotalRoute] = 1;
          DofRoute[TotalRoute] = Capa-SumDemand;
          DSent[TotalRoute][2] = Capa-SumDemand;
          SumDemand += Capa-SumDemand;

if(Test == true){
          myfile<<"SumDemand < Capa,TotalRoute: "<<TotalRoute<<endl;
}
}
}
}else{
if(Test == true){
          myfile<<"else Qk-DofRoute[TotalRoute]<= Capa-SumDemand"<<endl;
}

if(Distance[Factory][Route[TotalRoute][2]]+Distance[Route[TotalRoute][2]][NumberM]+
Distance[NumberM][Factory]<= MaxDistance){
          Route[TotalRoute][3] =  NumberM;
          MofRoute[TotalRoute] = 2;
          DofRoute[TotalRoute] += Capa-SumDemand;
          DSent[TotalRoute][3] = Capa-SumDemand;
          SumDemand += Capa-SumDemand;
SumDisRoute[TotalRoute] = Distance[Factory][Route[TotalRoute][2]]+
```

```
            Distance[Route[TotalRoute][2]][NumberM];

if(Test == true){
            myfile<<"Dis<=MaxDistance,SumDemand: "<<SumDemand<<endl;
}
}else{
            TotalRoute += 1;
            SumDisRoute[TotalRoute] = Distance[Factory][NumberM];
            Route[TotalRoute][2] =  NumberM;
            MofRoute[TotalRoute] = 1;
            DofRoute[TotalRoute] += Capa-SumDemand;
            DSent[TotalRoute][2] = Capa-SumDemand;
            SumDemand += Capa-SumDemand;
if(Test == true){
            myfile<<"else Dis<=MaxDistance,SumDemand: "<<SumDemand<<endl;
}
}
}
}
}//End else_if(NoMPossible == 0)

if(Test == true){
            myfile<<"************* EnD ******************************\n"<<endl;
}
if(SumDemand < Capa){ goto AAA;}
            delete []NEqualM;
            delete []MEqualN;
            delete []Attractiveness;
            delete []MPossible;

/*End SelectM*/}

//*******************************    Open Factory    **********************************
int OpenFactory(double* Pheromone_L,int Alpha_L,int Beta_L,double* SumDistance_L,
            int&No){

double* Attractiveness_L =  new double [Total_Nod+1];
double SumAttractiveness_L = 0;
int Factory;
int i=0;
SumAttractiveness_L = 0;

for(i=M+1;i<=Total_Nod;i++){
            Attractiveness_L[i] = pow(Pheromone_L[i],Alpha_L)*SumDistance_L[i];
            SumAttractiveness_L += Attractiveness_L[i];
}

Factory = CalPop(M+1,Total_Nod,Attractiveness_L,SumAttractiveness_L,No,false);
delete []Attractiveness_L;
return Factory;

/*End OpenFactory*/}

//************************  Calculate Pop & Selection ***************************
int CalPop(int Start,int Finish,double* Attractiveness,double SumAttractiveness,
            int& No,bool Test){

int TTT = 0;
double* Cumpop = new double [284];
double* Pop = new double [284];
double NoRand;
int i=0;
int Number;

for(i=Start;i<=Finish;i++){
            Pop[i] = Attractiveness[i]/SumAttractiveness;
            Cumpop[Start-1] = 0;
}
for(i=Start;i<=Finish;i++){
            Cumpop[i] = Cumpop[i-1]+Pop[i];
}
do{
            TTT += 1;
```

```
            NoRand = Random(No);
if(NoRand == 1 ){
            Number = Finish;
}else{

for(i=Start;i<=Finish;i++){
if(Cumpop[i-1] < Cumpop[i]){
if((Cumpop[i-1]< NoRand)&&(NoRand<=Cumpop[i])){
            Number = i;
            break;
}}
}
}
if(TTT > 2){
            cout<<"Eror CalPop()TTT: "<<TTT<<endl;
            myfile<<"Eror CalPop()TTT: "<<TTT<<endl;
}
if((Number<Start)||(Number>Finish)){
            cout<<"Start: "<<Start<<"\tFinish: "<<Finish<<"\tNumber: "<<Number<<endl;
            cout<<"NoRand: "<<NoRand<<endl;
for(i=Start;i<=Finish;i++){
            cout<<Cumpop[i]<<'\t';
}
cout<<endl;
cin>>TTT;
}
}while((Number<Start)||(Number>Finish));
delete &NoRand;
delete []Cumpop;
delete []Pop;
return Number;

/*End CalPop*/}

//*******************************    Random    **********************************
double Random(int&No){

int i=0;
srand(time(NULL)+NoRandom);
int range = 100-1+1;
double R;

for(i=1;i<=No;i++){
            R = (rand()/100%range+1)/100.0;
}

No += 1;

if(No>=200){
            No=1;
}
NoRandom +=1;
if(NoRandom >=500){
            NoRandom = 1;
}
return R;
}

//*******************************    Reset    **********************************
void Reset(int**& Route,double**& DSent,double**& HSent,double*& DofRoute,
            double*& HofRoute,int*& MofRoute,int*& NofRoute,int& TotalRoute,
            double*&SumDisRoute,double&TotalDistance){

int i=0,j=0;
TotalDistance = 0;
for(i=1;i<=Total_Nod;i++){
for(j=1;j<= Total_Nod;j++){
            Route[i][j] = 0;
            HSent[i][j] = 0;
if(j<=4){
            DSent[i][j] = 0;
}
}
```

```
            SumDisRoute[i] = 0;
            MofRoute[i] =0;
            NofRoute[i] =0;
            DofRoute[i] =0;
            HofRoute[i] =0;
}
            TotalRoute = 0;
}

//*******************************    Show   ************************************
void Show(int Factory,int** Route,double** DSent,double** HSent,
            double* DofRoute,double* HofRoute,int* MofRoute,int* NofRoute,
            int TotalRoute,double*SumDisRoute,double TotalDistance){

int i=0,j=0;

myfile<<"Factory : "<<Factory<<endl;
myfile<<"TotalRoute : "<<TotalRoute<<endl;
myfile<<"TotalDistance : "<<TotalDistance<<endl;

for(i=1;i<=TotalRoute;i++){
            myfile<<"MofRoute["<<i<<"]:\t"<<MofRoute[i]<<'\t'<<"NofRoute["<<i<<"]:\t"
            <<NofRoute[i]<<endl;
}

for(i=1;i<=TotalRoute;i++){
            myfile<<"SumDisRoute["<<i<<"]:\t"<<SumDisRoute[i]<<endl;
}

for(i=1;i<=TotalRoute;i++){
            myfile<<"DofRoute["<<i<<"]:\t"<<DofRoute[i]<<endl;
}

for(i=1;i<=TotalRoute;i++){
            myfile<<"HofRoute["<<i<<"]:\t"<<HofRoute[i]<<endl;
}

myfile<<"H[Factory]:\t"<<H[Factory]<<endl;

for(i=1;i<=TotalRoute;i++){
            myfile<<"Rout["<<i<<"]:\t";
for(j=1;j<=MofRoute[i]+NofRoute[i]+2;j++){
            myfile<<Route[i][j]<<"\t";
}myfile<<endl;
}

for(i=1;i<=TotalRoute;i++){
            myfile<<"DSent["<<i<<"]: ";
for(j=1;j<=4;j++){
            myfile<<DSent[i][j]<<'\t';
}myfile<<endl;
}

for(i=1;i<=TotalRoute;i++){
            myfile<<"HSent["<<i<<"]: ";

for(j=1;j<=MofRoute[i]+NofRoute[i]+2;j++){
            myfile<<HSent[i][j]<<"\t";
}myfile<<endl;
}
}
```

ประวัติผู้วิจัย

ชื่อ	นางสาวฐิติวรดา คำแหน
วันเดือนปีเกิด	20 กันยายน 2530
ประวัติการศึกษา	พ.ศ.2549 - 2553
	วิศวกรรมศาสตรบัณฑิต (วิศวกรรมอุตสาหการ)
	มหาวิทยาลัยอุบลราชธานี
	พ.ศ.2553 - ปัจจุบัน
	วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมอุตสาหการ)
	มหาวิทยาลัยอุบลราชธานี
ทุนการศึกษา	ทุนอุดหนุนการวิจัย ปีงบประมาณ 2554
	มหาวิทยาลัยอุบลราชธานี (ผู้ช่วยวิจัย)
งานวิจัยที่สนใจ	การจัดการการดำเนินงานและการผลิต
	(Operation and Production Management)
	การจัดการโลจิสติกส์ (Logistical Management)
	การพัฒนาซอร์ฟแวร์ (Software Development)
	การวิจัยการดำเนินงาน (Operation Research)
	การออกแบบและการทดลอง (Design of Experiment)
	ฮิวริสติกส์และเมต้าฮิวริสติกส์
	(Heuristic and Meta-Heuristic)
	e-mail: thitiworada_99@hotmail.com