



การประยุกต์ใช้ขั้นตอนวิธีการป็นเขาและขั้นตอนวิธีการจำลองการอบเหนียวสำหรับปัญหาพนักงาน  
ขาย ในคลังสินค้าบนโปรแกรมไมโครซอฟท์เอกซ์เซล



โดย  
นางสาวจิราพร สกุลวรารักษ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาการจัดการงานวิศวกรรม แผน ก แบบ ก 2 ปริญญาวิทยาศาสตรมหาบัณฑิต

ภาควิชาวิศวกรรมอุตสาหการและการจัดการ

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

ปีการศึกษา 2562

ลิขสิทธิ์ของบัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

การประยุกต์ใช้ขั้นตอนวิธีการป็นเขาและขั้นตอนวิธีการจำลองการอบเหนียวสำหรับปัญหา  
พนักงานขาย ในคลังสินค้าบนโปรแกรมไมโครซอฟท์เอกซ์เซล



โดย  
นางสาวจิราพร สกุลวรารักษ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาการจัดการงานวิศวกรรม แผน ก แบบ ก 2 ปริญญามหาบัณฑิต  
ภาควิชาวิศวกรรมอุตสาหกรรมและการจัดการ  
บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร  
ปีการศึกษา 2562  
ลิขสิทธิ์ของบัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร

TRAVELING SALESMAN PROBLEM IN WAREHOUSE BY USING HILL CLIMBING  
AND SIMULATED ANNEALING ALGORITHMS ON MICROSOFT EXCEL



A Thesis Submitted in Partial Fulfillment of the Requirements  
for Master of Engineering (ENGINEERING MANAGEMENT)  
Department of INDUSTRIAL ENGINEERING AND MANAGEMENT  
Graduate School, Silpakorn University  
Academic Year 2019  
Copyright of Graduate School, Silpakorn University

หัวข้อ	การประยุกต์ใช้ขั้นตอนวิธีการป็นเขาและขั้นตอนวิธีการจำลองการ อบเหนียวสำหรับปัญหาพนักงานขาย ในคลังสินค้าบนโปรแกรม ไมโครซอฟท์เอกซ์เซล
โดย	จิราพร สกุลวรารักษ์
สาขาวิชา	การจัดการงานวิศวกรรม แผนก ก แบบ ก 2 ปริญญามหาบัณฑิต
อาจารย์ที่ปรึกษาหลัก	ผู้ช่วยศาสตราจารย์ ดร. คณศ พันธุ์สวาสดี

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร ได้รับพิจารณาอนุมัติให้เป็นส่วนหนึ่งของการศึกษา  
ตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต

	คณบดีบัณฑิตวิทยาลัย
(รองศาสตราจารย์ ดร.จุไรรัตน์ นันทานิช)	
พิจารณาเห็นชอบโดย	
	ประธานกรรมการ
(รองศาสตราจารย์ ดร.ประจวบ กล่อมจิตร)	
	อาจารย์ที่ปรึกษาหลัก
(ผู้ช่วยศาสตราจารย์ ดร.คณศ พันธุ์สวาสดี)	
	ผู้ทรงคุณวุฒิภายนอก
(ผู้ช่วยศาสตราจารย์ ดร.กนกพร ศรีปฐมสวัสดิ์ )	

60405201 : การจัดการงานวิศวกรรม แผน ก แบบ ก 2 ปริญญามหาบัณฑิต

คำสำคัญ : ปัญหาการเดินทางของพนักงานขาย, ขั้นตอนวิธีการป็นเขา, ขั้นตอนวิธีการจำลองการอบ  
เหนียว

นางสาว จิราพร สุกุลวราภักษ์: การประยุกต์ใช้ขั้นตอนวิธีการป็นเขาและขั้นตอนวิธีการ  
จำลองการอบเหนียวสำหรับปัญหาพนักงานขาย ในคลังสินค้าบนโปรแกรมไมโครซอฟท์เอกซ์เซล  
อาจารย์ที่ปรึกษาวิทยานิพนธ์ : ผู้ช่วยศาสตราจารย์ ดร. คณิศ พันธุ์สวาสดี

การหีบสินค้าเป็นหนึ่งในการดำเนินการในคลังสินค้าที่ใช้ต้นทุนสูงถึงร้อยละ 55-65 ของ  
ต้นทุนทั้งหมดในคลังสินค้า [1] ดังนั้นการเพิ่มประสิทธิภาพของกระบวนการนี้จะช่วยลดเวลาและลด  
ต้นทุนโดยรวมของคลังสินค้าทั้งหมด ปัญหานี้เป็นที่รู้จักกันว่าปัญหาพนักงานขายเดินทางซึ่งเป็น  
ปัญหา NP-Hard การวิจัยครั้งนี้ใช้ขั้นตอนวิธีการป็นเขาเพื่อแก้ปัญหา เนื่องจากความเรียบง่ายรวดเร็ว  
และราคาถูกและใช้ขั้นตอนวิธีการจำลองการอบเหนียว เนื่องจากมันเป็นวิธีการที่ง่ายและมีโอกาสที่  
ผลลัพธ์จะเป็นคำตอบที่ดีที่สุด โปรแกรมนี้ทำงานบนไมโครซอฟท์ เอกซ์เซล ในการวิจัยนี้กำหนด  
ขอบเขตโดยจำนวนตำแหน่งของชั้นวางสินค้าถูกกำหนดไว้ที่ 5,10,15, 20 และ 30 จำนวนรอบการ  
ทำงานของขั้นตอนวิธีการป็นเขาเท่ากับจำนวนรอบการทำงานของขั้นตอนวิธีการจำลองการอบ  
เหนียว สำหรับขั้นตอนวิธีการจำลองการอบเหนียวค่าเริ่มต้นของอุณหภูมิ คือ 10, 50,100, 250 และ  
500 และกำหนดอัตราการลดลงของอุณหภูมิ (อัลฟา) ที่ 0.5 และ 0.99 ผลลัพธ์แสดงขึ้นเมื่อเวลา  
ทำงานเพิ่มผลลัพธ์ของขั้นตอนวิธีการจำลองการอบเหนียวมีวิธีแก้ปัญหาที่ดีกว่าขั้นตอนวิธีการป็นเขา  
ผลลัพธ์ยังแสดงว่าค่าเริ่มต้นของอุณหภูมิ อัตราการลดลงของอุณหภูมิ (อัลฟา) มีผลต่อผลลัพธ์ซึ่งหาก  
เพิ่มขึ้นจะส่งผลให้ผลลัพธ์ที่ได้ดีขึ้น แต่เวลาทำงานก็นานขึ้นเช่นกัน

60405201 : Major (ENGINEERING MANAGEMENT)

Keyword : Traveling Salesman Problem, Hill Climbing Algorithm, Simulated Annealing Algorithm

MISS JIRAPORN SAKULWARARAK : TRAVELING SALESMAN PROBLEM IN WAREHOUSE BY USING HILL CLIMBING AND SIMULATED ANNEALING ALGORITHMS ON MICROSOFT EXCEL THESIS ADVISOR : ASSISTANT PROFESSOR KANATE PANSAWAT, Ph.D.

Order Picking is one operation in the warehouse, that has been using 55-65% of the warehouse's total cost [1]. Therefore, increasing the efficiency of this process will help to reduce time-consuming and reduce the total cost of the entire warehouse. This problem is also known as the Traveling Salesman Problem which is an NP-Hard problem. This research uses the Hill Climbing Algorithm to solve the problem due to its simplicity, fast and cheap and uses the Simulated Annealing Algorithm because it's quite a simple method and also there is a chance that the result will be Global. This program works on Microsoft Excel. In this research the scope is set by , the number of positions of shelves is set by 5, 10, 15, 20 and 30. The number of running-time of the Hill Climbing Algorithm is as same as a number of running-times of the Simulated Annealing Algorithm. For Simulated Annealing Algorithm is set of Initial T is 10, 50, 100, 250 and 500 and set the rate of cooling (alpha) at 0.5 and 0.99. The results show when the running-time increase the result of Simulated Annealing Algorithm have a better solution than Hill Climbing Algorithm. The results also show that the initial T, the rate of cooling (alpha) affects the result, which, if increased, results in better results. But the working time is longer as well

## กิตติกรรมประกาศ

ผู้วิจัยขอขอบคุณภาควิชาวิศวกรรมอุตสาหการและการจัดการ คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยศิลปากร ที่ได้สนับสนุนและอำนวยความสะดวกในการทำงานวิจัยนี้ ขอขอบคุณอาจารย์ที่ปรึกษาวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ คณิศ พันธุ์สวัสดิ์ ที่ได้ให้คำปรึกษา คำแนะนำ และแนวทางในการทำวิทยานิพนธ์ฉบับนี้ ขอขอบคุณพี่ห้องธุรการ ที่คอยช่วยเหลือในการจัดทำเอกสาร รวมถึงขอบคุณเพื่อนของข้าพเจ้า นาย ตฤณ ปฐมนิธิภิญโญ ที่คอยช่วยเหลือ ให้คำปรึกษา และให้กำลังใจตลอดจนสำเร็จจุล่ง

และสุดท้ายนี้ผู้วิจัยขอขอบคุณบิดา มารดา และครอบครัวที่ส่งเสริม และสนับสนุนให้ผู้วิจัยได้มีโอกาสใน การศึกษาหาความรู้เพื่อนำไปพัฒนาตนเอง

จิราพร สกุลวารักษ์



## สารบัญ

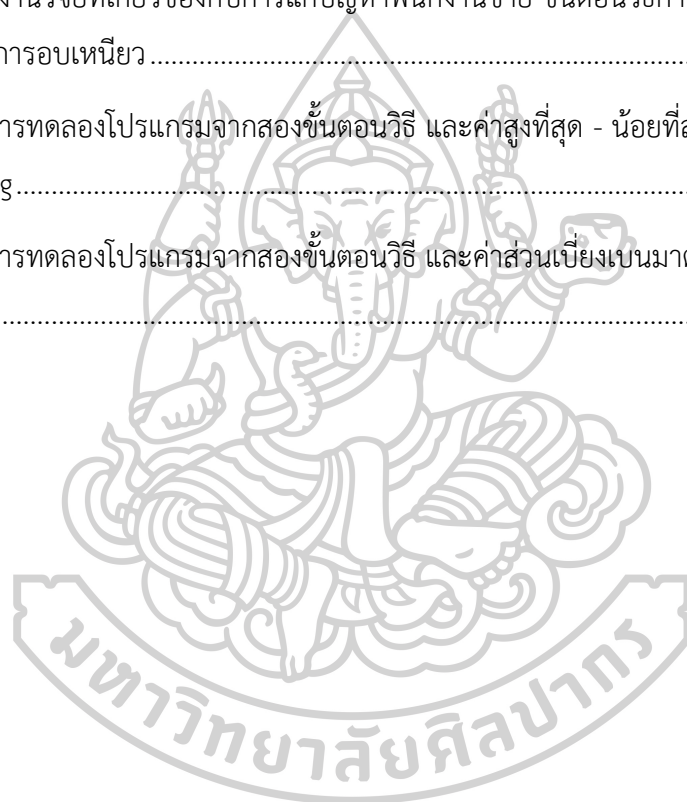
	หน้า
บทคัดย่อภาษาไทย .....	ง
บทคัดย่อภาษาอังกฤษ .....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ฌ
สารบัญรูปภาพ.....	ญ
บทที่ 1 .....	1
บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของงานวิจัย .....	2
1.3 ขอบเขตงานวิจัย .....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.5 สมมติฐาน .....	3
บทที่ 2 .....	4
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....	4
2.1 ทฤษฎี.....	4
2.1.1 ปัญหาการเดินทางของพนักงานขาย .....	4
2.1.2 ขั้นตอนวิธีการปีนเขา (Hill Climbing Algorithm: HCA) .....	5
2.2 งานวิจัยที่เกี่ยวข้อง.....	10
2.2.1 ตัวอย่างการแก้ปัญหาพนักงานขายด้วยขั้นตอนวิธีต่าง ๆ.....	10
2.2.2 การประยุกต์ใช้ขั้นตอนวิธีการปีนเขา .....	11



2.2.3 การประยุกต์ใช้ขั้นตอนวิธีการจำลองการอบเหนียว .....	12
บทที่ 3 .....	14
วิธีการทดลอง .....	14
3.1 การดึงข้อมูลจากตารางแผนภูมิจากไป (From to Chart) .....	15
3.2 การเลือกตำแหน่งจากชั้นวางสินค้า .....	16
3.3 สร้างขั้นตอนวิธี .....	17
3.3.1 ขั้นตอนวิธีการป็นเขา .....	19
3.3.2 ขั้นตอนวิธีการจำลองการอบเหนียว .....	21
3.4 ทดสอบ และตรวจสอบความถูกต้อง .....	23
3.5 เปรียบเทียบผลที่ได้ .....	23
บทที่ 4 .....	24
ผลและสรุปผลการวิจัย .....	24
บทที่ 5 .....	35
สรุปผลการวิจัยและข้อเสนอแนะ .....	35
5.1 สรุปผลการวิจัย .....	35
5.2 ข้อเสนอแนะ .....	35
รายการอ้างอิง .....	35
ประวัติผู้เขียน .....	39

## สารบัญตาราง

	หน้า
ตาราง 1 ตารางแสดงรอบการทำงานของโปรแกรม จากการค่า $\alpha$ และค่า T เริ่มต้นต่าง ๆ ของ ขั้นตอนวิธีการจำลองการอบเหนียว.....	3
ตาราง 2 ผลลัพธ์ระยะทางที่ได้จากขั้นตอนต่าง ๆ ในจำนวนเมืองที่ต่างกัน.....	10
ตาราง 3 สรุปรงานวิจัยที่เกี่ยวข้องกับการแก้ปัญหาพนักงานขาย ขั้นตอนวิธีการป็นเขา และขั้นตอน วิธีการจำลองการอบเหนียว.....	12
ตาราง 4 ผลการทดลองโปรแกรมจากสองขั้นตอนวิธี และค่าสูงสุด - น้อยที่สุดจาก Linear Programming.....	25
ตาราง 5 ผลการทดลองโปรแกรมจากสองขั้นตอนวิธี และค่าส่วนเบี่ยงเบนมาตรฐานจากการคำนวณ .....	31



## สารบัญรูปภาพ

	หน้า
ภาพที่ 1 ขั้นตอนการดำเนินงาน .....	14
ภาพที่ 2 ตัวอย่างคลังสินค้าที่สร้างขึ้น .....	15
ภาพที่ 3 ตัวอย่างแผนภูมิจากไป .....	16
ภาพที่ 4 ตัวอย่างเมื่อกำหนดช่องทางเดินและตำแหน่งชั้นวางสินค้า .....	17
ภาพที่ 5 ตัวอย่างตาราง From to Chart เฉพาะค่าที่ต้องการ .....	17
ภาพที่ 6 ตัวอย่างตาราง Form to Chart ใหม่จากการสลับตำแหน่งช่องทางเดิน .....	18
ภาพที่ 7 การรวมระยะทางทั้งหมด .....	18
ภาพที่ 8 Flow Chart การทำงานของขั้นตอนวิธีการป็นเขา .....	19
ภาพที่ 9 Flow Chart การทำงานของขั้นตอนวิธีการจำลองการอบเหนียว .....	21
ภาพที่ 10 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และผลจาก Linear Programming สำหรับช่อง ทางเดิน 5 ช่อง .....	27
ภาพที่ 11 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงสุด น้อยที่สุดจาก Linear .....	27
ภาพที่ 12 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงสุด น้อยที่สุดจาก Linear Programming สำหรับช่องทางเดิน 15 ช่อง .....	28
ภาพที่ 13 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงสุด น้อยที่สุดจาก Linear Programming สำหรับช่องทางเดิน 20 ช่อง .....	28
ภาพที่ 14 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงสุด น้อยที่สุดจาก Linear Programming สำหรับช่องทางเดิน 30 ช่อง .....	29
ภาพที่ 15 กราฟผลของขั้นตอนวิธีการป็นเขาในแต่ละช่องทางเดินเมื่อเทียบกับค่าระยะทางรวมน้อย ที่สุด .....	30
ภาพที่ 16 กราฟผลของขั้นตอนวิธีการจำลองการอบเหนียวในแต่ละช่องทางเดินเมื่อเทียบกับค่า ระยะทางรวมน้อยที่สุด .....	30

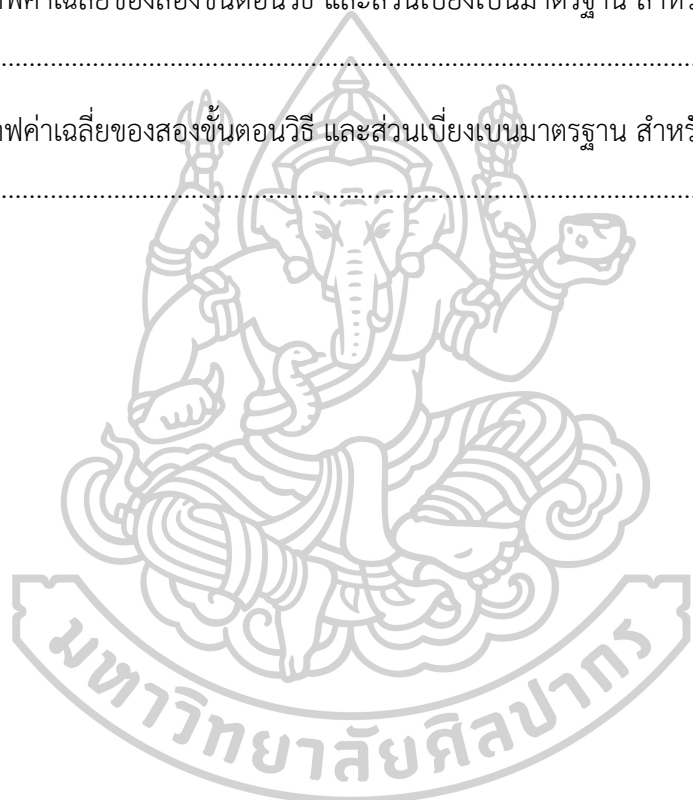
ภาพที่ 17 กราฟค่าเฉลี่ยของสองชั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 5 ช่อง  
 ..... 32

ภาพที่ 18 กราฟค่าเฉลี่ยของสองชั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 10 ช่อง  
 ..... 33

ภาพที่ 19 กราฟค่าเฉลี่ยของสองชั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 15 ช่อง  
 ..... 33

ภาพที่ 20 กราฟค่าเฉลี่ยของสองชั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 20 ช่อง  
 ..... 34

ภาพที่ 21 กราฟค่าเฉลี่ยของสองชั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 30 ช่อง  
 ..... 34



## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญ

การเดินหยิบสินค้าในคลังสินค้าตามใบคำสั่งซื้อ (Order Picking) จัดเป็นกิจกรรมที่มีต้นทุนสูงที่สุดในคลังสินค้า (Warehouse) โดยสูงถึงร้อยละ 55-65 ของต้นทุนทั้งหมดในคลังสินค้า และ ในต้นทุนนี้ ค่าใช้จ่ายในการเดินหยิบสินค้าตามชั้นวาง (Traveling) ถูกนำมาเทียบได้สูงถึงร้อยละ 50 ซึ่งนอกจากจะใช้ต้นทุนสูงแล้วยังใช้ระยะเวลาในการดำเนินงานสูงอีกด้วย [1] ดังนั้นการเพิ่มประสิทธิภาพการเดินหยิบสินค้านั้นจึงมีความจำเป็นสำหรับการลดต้นทุน ลดระยะเวลา และเพิ่มผลกำไร

การเดินหยิบสินค้าตามใบคำสั่งซื้อให้มีระยะทางสั้นที่สุด เป็นหนึ่งในการเพิ่มประสิทธิภาพของการเดิน ซึ่งการหยิบสินค้าตั้งแต่ต้นทาง (ประตูทางเข้า) เดินหยิบจนครบทุกชั้น แล้วสิ้นสุดที่ปลายทาง (ประตูทางออก) โดยต้นทางและปลายทางเป็นจุดเดียวกัน ปัญหานี้เป็นปัญหาเดียวกับกับปัญหาการเดินทางของพนักงานขาย (Traveling Salesman Problem) [2, 3]

ปัญหาการเดินทางของพนักงานขายเป็นปัญหาการตัดสินใจในการเดินทางไปยังเมืองต่างๆ ของพนักงานขายจนครบทุกเมืองแล้วจึงเดินทางกลับมายังเมืองเริ่มต้น สำหรับการแก้ปัญหาพนักงานขายนั้นมีหลากหลายวิธี ยกตัวอย่างเช่น ขั้นตอนวิธีอาณานิคมมด (Ant Colony Algorithm: ACO) ขั้นตอนวิธีเชิงพันธุกรรม (Genetic Algorithm: GA) ขั้นตอนวิธีการปีนเขา (Hill Climbing Algorithm: HCA) ขั้นตอนวิธีจำลองการอบเหนียว (Simulated Annealing Algorithm: SA) เป็นต้น [4] พบว่าขั้นตอนวิธีส่วนใหญ่ เมื่อปัญหามีขนาดใหญ่ขึ้นระยะเวลาในการหาคำตอบก็ยิ่งเพิ่มขึ้นแบบเอกโพเนนเชียล ทั้งยังมีโอกาสที่จะได้คำตอบแบบ Local Optimal สูงกว่าคำตอบแบบ Global Optimal

จากหลากหลายขั้นตอนวิธีผู้วิจัยได้เลือกใช้ 2 ขั้นตอนวิธีมาแก้ไขปัญหานี้ ได้แก่ ขั้นตอนวิธีการปีนเขา เนื่องจากขั้นตอนวิธีการปีนเขาเป็นการสุ่มจึงทำให้ระยะเวลาในการคำนวณไม่สูงจนเกินไปแม้ขนาดของปัญหาจะใหญ่ขึ้นก็ตาม อีกทั้งการทำงานก็ไม่ซับซ้อน จึงค่อนข้างเป็นที่นิยมเพราะสะดวกและรวดเร็ว แต่ข้อเสียคือมีโอกาสติดอยู่ในคำตอบแบบ Local Optimal สูง ส่วนอีกขั้นตอนวิธีที่ใช้ คือ ขั้นตอนวิธีจำลองการอบเหนียวซึ่งถูกประยุกต์ใช้มาจากการอบอ่อนโลหะ โดยหลักการการทำงานคล้ายกับขั้นตอนวิธีการปีนเขาเพียงแต่มีการยอมรับคำตอบที่แย่ในบางกรณีด้วยเหตุนี้จึงช่วยลดโอกาสในการได้คำตอบแบบ Local Optimal และเพิ่มโอกาสในการได้คำตอบแบบ Global Optimal และเนื่องด้วยสองขั้นตอนวิธีนี้มีการทำงานที่คล้ายคลึงกัน ผู้วิจัยจึงเลือกนำมา

วิเคราะห์ และ เปรียบเทียบผลลัพธ์กัน โดยผู้วิจัยได้เลือกทำบนโปรแกรมไมโครซอฟท์เอกซ์เซล (Microsoft Excel) เนื่องจากเป็นโปรแกรมที่ใช้ทำงานง่าย และนิยมใช้กันอย่างแพร่หลาย

## 1.2 วัตถุประสงค์ของงานวิจัย

เพื่อนำปัญหาการเดินของพนักงานขาย มาประยุกต์ใช้ในคลังสินค้า สำหรับการเดินหยิบสินค้าในคลังสินค้าตามใบคำสั่งซื้อสินค้าเพื่อเพิ่มประสิทธิภาพ (Capability) ด้วยขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียว

เพื่อเปรียบเทียบประสิทธิภาพของขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียว

## 1.3 ขอบเขตงานวิจัย

- 1) กำหนดจำนวนช่องตำแหน่งทางเดินเป็น 5 ,10 ,15 ,20 ,30 ช่อง
- 2) กำหนดค่า T เริ่มต้น (Initial Temperature) ในขั้นตอนวิธีการจำลองการอบเหนียว เป็น [17] 10, 50, 100, 250 และ 500
- 3) กำหนดค่า  $\alpha$  เป็น 0.5 และ 0.99
- 4) กำหนดให้การหาค่าตอบแต่ละจำนวน หรือแต่ละค่า T ทำทั้งหมด 5 รอบ
- 5) กำหนดให้รอบการทำงาน of โปรแกรมของขั้นตอนวิธีการป็นเขาเป็น 10, 13, 14, 15, 16, 688, 848, 917, 1008, 1077 รอบ

เนื่องจากผู้วิจัยต้องการที่จะเปรียบเทียบขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียว ซึ่งจากค่า  $\alpha$  และค่า T เริ่มต้น ที่ต่างกันส่งผลให้จำนวนรอบการทำงาน of โปรแกรมของขั้นตอนวิธีการจำลองการอบเหนียวต่างกัน ดังนั้นผู้วิจัยจึงอ้างอิงจำนวนรอบการทำงาน of โปรแกรมของขั้นตอนวิธีการจำลองการอบเหนียวที่ได้ มาใช้กับจำนวนรอบการทำงาน of โปรแกรมของขั้นตอนวิธีการป็นเขาโดยตารางแสดงรอบการทำงาน จากการค่า  $\alpha$  และค่า T เริ่มต้น ของขั้นตอนวิธีการจำลองการอบเหนียวเป็นดังตารางที่ 1

**ตาราง 1** ตารางแสดงรอบการทำงานของโปรแกรม จากการค่า  $\alpha$  และค่า T เริ่มต้นต่าง ๆ ของขั้นตอนวิธีการจำลองการอบเหนียว

$\alpha$	0.5					0.99				
ค่า T เริ่มต้น	10	50	100	250	500	10	50	100	250	500
จำนวนรอบการทำงานที่ได้	10	13	14	15	14	688	848	917	1008	1077

#### 1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1) เพิ่มประสิทธิภาพในการเดินหีบสินค้าตามใบคำสั่งซื้อในคลังสินค้า
- 2) เพิ่มความสะดวกต่อการคำนวณ และใช้งานด้วยการใช้โปรแกรม Microsoft Excel
- 3) เพื่อวิเคราะห์เปรียบเทียบประสิทธิภาพระหว่างขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียว
- 4) เพื่อเปรียบเทียบตัวแปรที่เกี่ยวข้องได้แก่ Initial T,  $\alpha$  และจำนวนช่องตำแหน่งทางเดินว่ามีผลต่อคำตอบหรือไม่

#### 1.5 สมมติฐาน

- 1) ระยะทางการเดินหีบสินค้าสั้นลง ประหยัดเวลา และค่าใช้จ่าย
- 2) T,  $\alpha$  จำนวนรอบการทำงานขอโปรแกรม และ จำนวนช่องตำแหน่งทางเดิน มีผลต่อการคำตอบที่ได้
- 3) ขั้นตอนวิธีการจำลองการอบเหนียวมีประสิทธิภาพดีกว่าขั้นตอนวิธีการป็นเขา

## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

#### 2.1 ทฤษฎี

##### 2.1.1 ปัญหาการเดินทางของพนักงานขาย

ปัญหาการเดินทางของพนักงานขาย เป็นรูปแบบหนึ่งของวิธีในการดำเนินการแก้ปัญหาการขนส่งที่มีวัตถุประสงค์ คือ ต้องการหาเส้นทางที่ทำให้การเดินทางไปยังลูกค้าทุกคน หรือเมืองทุกเมืองมีระยะทางรวมสั้นที่สุด โดยพบลูกค้าหรือผ่านเมืองแต่ละเมืองเพียงแค่ครั้งเดียว แล้วจึงวนกลับมาจุดเริ่มต้นเหมือนการเดินทางวนรอบ หากลากเส้นเชื่อมระหว่างลูกค้าแต่ละคน หรือเมืองแต่ละเมืองจะได้กราฟบริบูรณ์แบบไม่มีทิศทาง (Complete undirected graph) โดยระยะทางและค่าใช้จ่ายในการเดินทางแต่ละเส้น (Weight) จะมีค่าไม่ติดลบ [5]

การแก้ปัญหาการเดินทางของพนักงานขาย จัดเป็นปัญหาแบบยาก (Non-deterministic Polynomial-time Hard :NP Hard) [6] โดยปัญหาดังกล่าวเป็นที่สนใจและถูกหยิบยกขึ้นมาศึกษาเพื่อหาทางออกเป็นครั้งแรกในปี 1930 ถือเป็นหนึ่งในปัญหาที่มีความละเอียดอ่อนและสำคัญอย่างมากในเรื่องของการเพิ่มประสิทธิภาพ [7] ซึ่งหลักการของปัญหานี้สามารถนำมาประยุกต์ใช้ได้หลากหลายรูปแบบ เช่น การวางแผนการเคลื่อนที่ที่มีประสิทธิภาพที่สุดของแขนหุ่นยนต์ (Robotic arm) ที่เจาะเข้าไปในจุด  $n$  บนพื้นผิวของชิป VLSI , การวางแผนการเรียงลำดับทำงานของโมดูลซอฟต์แวร์  $n$  โมดูลเพื่อลดเวลาการสลับการทำงานของซีพียู (Context Switch) เป็นต้น [8]

รูปแบบทางคณิตศาสตร์ของปัญหาการเดินทางของพนักงานขาย (Mathematical Model) [9]

กำหนดให้

$i$	=	ตำแหน่งจุดเริ่มต้นของแต่ละเส้นทางภายในเส้นทางรวม
$j$	=	ตำแหน่งปลายทางของแต่ละเส้นทางภายในเส้นทางรวม
$N$	=	เซตของจำนวนเมือง หรือจำนวนลูกค้าทั้งหมดที่พิจารณา
$S$	=	สับเซต (Subset) ใด ๆ ที่เป็นไปได้ทั้งหมดของ $N$ ; $S \subset N$ โดยที่ $S \neq \emptyset$ และ $S \neq N$
$D_{ij}$	=	ระยะทาง หรือ ต้นทุนในการเดินทางจากตำแหน่ง $i$ ไป $j$
$x_{ij}$	=	แสดงถึงเส้นทางระหว่างตำแหน่ง $i$ และ $j$



### ตัวแปรตัดสินใจ

$$x_{ij} \begin{cases} 1 & \text{เมื่อมีการเดินทางจาก } i \text{ ไป } j \\ 0 & \text{เมื่อไม่มีการเดินทางจาก } i \text{ ไป } j \end{cases}$$

### แบบจำลองทางคณิตศาสตร์

$$\text{Minimization } \sum_{j \in N} x_{ij} D_{ij} \quad (2.1)$$

### สมการข้อขาย

$$\sum_{j \in N} x_{ij} = 1 \quad \forall i \in N \quad (2.2)$$

$$\sum_{i \in N} x_{ij} = 1 \quad \forall j \in N \quad (2.3)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1 \quad (2.4)$$

$$x_{ij} \in \{0,1\} \quad (2.5)$$

สมการที่ (2.1) แสดงระยะทาง หรือ ต้นทุนที่ต่ำที่สุด

สมการที่ (2.2) กำหนดให้เดินทางจากจุดเริ่มต้นใด ๆ แล้วจะมีจุดปลายทางเพียงจุดเดียวเท่านั้น

สมการที่ (2.3) กำหนดให้จุดปลายทางใด ๆ จะมีจุดเริ่มต้นเพียงจุดเดียวเท่านั้น

สมการที่ (2.4) เป็นการกำจัดเส้นทางย่อย (Sub-tour Elimination)

สมการที่ (2.5) กำหนดให้ตัวแปรการตัดสินใจมีค่าเป็น 0 หรือ 1 เท่านั้น

### 2.1.2 ขั้นตอนวิธีการปีนเขา (Hill Climbing Algorithm: HCA)

ขั้นตอนวิธีการปีนเขา ถูกนำเสนอครั้งแรกโดยเพริล ในปี ค.ศ. 1984 [10] จัดเป็นการค้นหาเฉพาะที่ (Local Search) แบบต่อเนื่อง ใช้สำหรับ การหาค่าที่เหมาะสมที่สุด (Optimization) โดยวิธี Optimization techniques จากชื่อ ขั้นตอนวิธีการปีนเขา ดังนั้นวิธีการจึงเลือกได้แค่ ปีนขึ้นเขา (Uphill) หรือปีนลงเขา (Downhill) เพียงอย่างเดียว [11]

ขั้นตอนวิธีการปีนเขาเนื่องจากปีนขึ้น หรือลงได้เพียงอย่างเดียว จึงมีโอกาสติด Local Optima ได้สูง จึงมีการพัฒนาเพื่อแก้ไขปัญหานี้ เช่น Tabu Search, Greedy Algorithm, Simulated Annealing เป็นต้น

ขั้นตอนวิธีนี้จะใช้ได้ต้องมีประสิทธิภาพ เมื่อมีเขาเพียงแค่ลูกเดียว เนื่องจากไม่ว่าจะอยู่ตำแหน่งใดผลจะขึ้น หรือลงเสมอ ทำให้เกิด Local Optima และยังมีค่าใช้จ่ายที่ถูกถึงแม้จะมีขนาดใหญ่ [8]

#### ขั้นตอนการทำงาน ดังนี้ [12]

1. การสร้างสถานะเริ่มต้น โดยการสุ่ม เพื่อสร้างเป็นคำตอบแรก สำหรับใช้ในการเปรียบเทียบกับสถานะถัดไป
2. การสร้างสถานะใหม่ เป็นการสร้างคำตอบใหม่ด้วยการปรับเปลี่ยนตัวแปรของปัญหาจากสถานะเดิมเล็กน้อยเพื่อนำไปเปรียบเทียบกับสถานะเริ่มต้น
3. การตรวจสอบสถานะ เป็นขั้นตอนการเปรียบเทียบสถานะของคำตอบระหว่างสถานะใหม่กับสถานะเดิม ซึ่งถ้าหากสถานะไหนดีกว่าก็就会被กำหนดเป็นสถานะปัจจุบันเพื่อใช้ในการเปรียบเทียบครั้งต่อไป จนกว่าไม่พบสถานะที่ดีกว่าเดิม
4. ตรวจสอบสถานะเป้าหมายว่าตรงตามที่กำหนดหรือไม่
5. ทำขั้นตอนที่ 2-4 จนครบตรงตามเงื่อนไข

#### อัลกอริทึมขั้นตอนวิธีการป็นเขา [11]

- Random initial state E (1)
- Repeat (2)
- i. Apply Random perturbations to the Next state (2.1)
  - ii. Evaluate  $dE = E' - E$  (2.2)
    - if  $dE \leq 0$  (2.2.1)
 

Current State = New State
    - Else  $dE > 0$  (2.2.2)
 

Current State = Current State
- Until conditions is met (2.3)

### การทำงานของรหัสเทียม

- (1) สุ่มหาคำตอบแรก (Initial solution) กำหนดให้เป็น E
- (2) ทำการวนซ้ำ
  - (2.1) สุ่มหาคำตอบถัดไป (Next solution) โดยอิงมาจากคำตอบแรก
  - (2.2) คำนวณผลต่างของคำตอบใหม่และเก่า เพื่อเข้าเงื่อนไขการเลือกคำตอบ กำหนดให้ผลต่าง แทนด้วย  $dE$  ; ( $dE = E_2 - E_1$ )
    - (2.2.1) ถ้าคำตอบใหม่ดีกว่าคำตอบเก่า ให้เก็บคำตอบใหม่ไว้ แล้วไปยัง (2.1)
    - (2.2.2) ถ้าคำตอบใหม่แยกว่า ให้เก็บคำตอบเดิม แล้วไปยัง (2.1)
  - (2.3) หยุดการวนซ้ำเมื่อครบตามเงื่อนไข (ตามจำนวนรอบที่ต้องการ)

#### 2.1.3 วิธีการจำลองการอบเหนียว

วิธีการจำลองการอบเหนียว ถูกนำเสนอโดย Scott Kirkpatrick, C. Daniel Gelatt and Mario P. Vecchi ในปีค.ศ.1983 ซึ่งดัดแปลงมาจากขั้นตอนวิธี Metropolis-Hastings วิธีการคือให้ความร้อนแก่โลหะ หรือแก้วที่อุณหภูมิเหนือจุดหลอมเหลวแล้วทำการแช่ไว้ในอุณหภูมิที่ต้องการ เพื่อให้โลหะหรือแก้วคงสภาพตามที่เราต้องการ หลังจากนั้นลดอุณหภูมิอย่างช้า ๆ เพื่อให้เกิดผลึกโดยสมบูรณ์ [13, 14]

ขั้นตอนวิธีการอบเหนียวถูกนำมาประยุกต์ใช้สำหรับการแก้ปัญหาการหาค่าเหมาะสมที่สุดเชิงการจัด (Combinatorial Optimization Problem) ซึ่งเป็นปัญหาแบบ NP-Hard เช่น การหาค่าต่ำสุด หรือสูงสุดของปัญหา โดยขั้นตอนวิธีส่วนใหญ่้นั้นเมื่อปัญหามีขนาดใหญ่ขึ้น การหาคำตอบจะค้นหาแบบทุกค่า มีผลทำให้ระยะเวลาการคำนวณใช้เวลานานแสดงได้ดังรูปแบบของกราฟเอกโพเนนเชียล (Exponential graph) หรือ หาคำตอบแบบสุ่มค่า (Random Solution) ส่งผลให้คำตอบที่ได้มีค่าเป็น Local Optimal แทนที่จะเป็น Global Optimal [15] ขั้นตอนวิธีนี้ถูกพัฒนามาจากขั้นตอนวิธีการป็น ต่างกันที่ขั้นตอนวิธีการจำลองการอบเหนียวยอมรับค่าที่แยกว่าเพื่อเลี่ยง Local Optimal Solution [16]

สำหรับการเลือกค่าอุณหภูมิเริ่มต้น (Initial Temperature) และอัตราการลด (Rate of Cooling) ของ Temperature ของขั้นตอนวิธีนั้น มีผลต่อทั้งด้านเวลาในการคำนวณและคำตอบสุดท้าย โดยค่าทั้งสองจะแปรผกผันกัน ยกตัวอย่างหาก Initial Temperature สูง Rate of Cooling ควรต่ำ [17] เปรียบเทียบการใช้ [18]

Initial Temperature สูง	→	Optimization time สูง
Initial Temperature ต่ำ	→	ติด Local Optimal สูง
Rate of Cooling สูง	→	ติด Local Optimal สูง
Rate of Cooling ต่ำ	→	Optimization time สูง

รูปแบบทางคณิตศาสตร์ของขั้นตอนวิธีการจำลองการอบเหนียว [13, 16]

กำหนดให้	$E$	=	สถานะปัจจุบัน (Current State)
	$E'$	=	สถานะถัดมา (New State)
	$dE$	=	ค่าความแตกต่างระหว่าง $E'$ และ $E$ ( $E'-E$ )
	$T$	=	อุณหภูมิ (Temperature)
	$P$	=	ความน่าจะเป็นในสถานะ $E$ ณ อุณหภูมิ $T$
	$k_B$	=	ค่าคงตัวบ็อลทซ์มันน์ (Boltzmann's constant)

สมการขอขยาย

$$P \text{ [ยอมรับ } E' \text{ เป็น Current State]} = \begin{cases} \text{Exponential } (-dE / Tk_B) & \text{ถ้า } dE > 0 \\ 1 & \text{ถ้า } dE \leq 0 \end{cases}$$

สำหรับกรณี  $dE > 0$  ที่ยอมรับ  $E'$  เป็น Current State กำหนดเป็น  $P(dE)$

สุ่มค่า (0,1) กำหนดเป็น Random [0,1)

ถ้า  $P(dE) > \text{Random } [0,1)$  ; ยอมรับ  $E'$  เป็น Current State

$P(dE) \leq \text{Random } [0,1)$  ; ไม่ยอมรับ  $E'$  และคง Current State เดิม

### รหัสเทียม (Pseudo Code) [16]

Random initial state E (1)

Initialize T

Repeat (2)

i. Apply Random perturbations to the Next state (2.1)

ii. Evaluate  $dE = E' - E$  (2.2)

if  $dE \leq 0$

Current State = New State

Else  $dE > 0$

$P = e^{-\frac{dE}{T}}$  (2.3)

If  $P > \text{random}[0,1)$

Current State = New State

Else

Current State = Current State

$T = T \times \alpha$  (2.4)

Until conditions is met

### การทำงานของรหัสเทียม

(1) สุ่มหาคำตอบแรก (Initial solution) กำหนดให้เป็น E และ กำหนดค่า T เริ่มต้น

(2) ทำการวนซ้ำ จนครบตามเงื่อนไข คือ ค่า T น้อยตามที่ต้องการ

(2.1) สุ่มหาคำตอบถัดไป (Next solution) โดยอิงมาจากคำตอบแรก

(2.2) คำนวณผลต่างของคำตอบใหม่และเก่า เพื่อเข้าเงื่อนไขการเลือก

คำตอบ กำหนดให้ผลต่าง แทนด้วย  $dE$  ; ( $dE = E_2 - E_1$ )

ถ้าคำตอบใหม่ดีกว่าคำตอบเก่า ให้เก็บคำตอบใหม่ไว้ แล้วไปยัง (2.1)

ถ้าคำตอบใหม่แยกว่า ให้เข้าตามเงื่อนไข ดังนี้

(2.3) คำนวณ  $P = e^{-\frac{dE}{T}}$  ถ้า  $P > \text{random}[0,1)$  เก็บคำตอบใหม่ไว้แล้วไปยัง (2.1) แต่ถ้า ไม่ใช่ให้เก็บคำตอบเดิม แล้วไปยัง (2.1)

(2.4) ลด T ด้วยสมการ  $T = T \times \alpha$  จนตรงเงื่อนไขข้อ (2)

## 2.2 งานวิจัยที่เกี่ยวข้อง

### 2.2.1 ตัวอย่างการแก้ปัญหาพนักงานขายด้วยขั้นตอนวิธีต่าง ๆ

**Wang Hui (2011) [4]** ได้ศึกษาเปรียบเทียบขั้นตอนวิธีในการแก้ปัญหาพนักงานขายได้แก่ ขั้นตอนวิธีเชิงพันธุกรรม ขั้นตอนวิธีโครงข่ายประสาทเทียม (Hopfield Neural Network) และ ขั้นตอนวิธีอาณานิคมมด ทำการเปรียบเทียบเวลาในการหาคำตอบโดยการเปรียบเทียบ  $O(n)$  ความซับซ้อนของข้อมูลโดยใช้โปรแกรมแมตแล็บ (Matlab) และผลลัพธ์ที่ได้โดยโปรแกรม TSPLB eil51 จากนั้นนำผลที่ได้จากทั้ง 3 การเปรียบเทียบคำนวณวิเคราะห์เมตริก (Paired Comparison Matrix) พบว่าขั้นตอนวิธีอาณานิคมมดได้ผลลัพธ์ที่ดีที่สุด ต่อมาคือ ขั้นตอนวิธีเชิงพันธุกรรม ขั้นตอนวิธีโครงข่ายประสาทเทียม รองมาตามลำดับ

**Haider A Abdulkarim และ Ibrahim F Alshammari (2015) [19]** ได้ศึกษาจากการเดินทางไปยังเมืองต่าง ๆ ในประเทศสหรัฐอเมริกา โดยเปรียบเทียบการเดินทางของเมือง ได้แก่ 20 เมือง 100เมือง และ1000เมือง และได้ศึกษาเปรียบเทียบระหว่างขั้นตอนวิธีประเภท ขั้นตอนวิธีเชิงพันธุกรรม และ ขั้นตอนวิธีการค้นหาเพื่อนบ้าน (Nearest Neighbor Algorithm) โยใช้โปรแกรมแมตแล็บ (Matlab) พบว่าขั้นตอนวิธีประเภทละโมบได้ผลลัพธ์ที่ดีที่สุด

**Ansari, Abdul Quaiyum และ Sapna Katiyar (2015) [20]** ได้ศึกษาเปรียบเทียบระหว่างขั้นตอนวิธีเชิงพันธุกรรม ขั้นตอนวิธีอาณานิคมมด และยังผสานขั้นตอนวิธีอาณานิคมมด เข้ากับขั้นตอนวิธีแบบนกคuckoo (Cuckoo search) (ACO-CS) รวมถึงผสาน ขั้นตอนวิธีอาณานิคมมด เข้ากับ ขั้นตอนวิธีเชิงพันธุกรรม (ACO-GA) และทำการทดลองด้วยจำนวนเมือง 15, 20, 25, 30, 42 และ 50 ผลการทดลองระยะทางที่ได้จากขั้นตอนวิธีต่าง ๆ ในจำนวนเมืองที่ต่างกัน แสดงดังตารางที่ 2 ผลลัพธ์พบว่า ACO-GA ได้ผลลัพธ์ที่ดีที่สุดเมื่อเทียบกับขั้นตอนวิธีอื่น ๆ

**ตาราง 2** ผลลัพธ์ระยะทางที่ได้จากขั้นตอนต่าง ๆ ในจำนวนเมืองที่ต่างกัน

Number of Cities	Distance by GA	Distance by ACO	Distance by ACO-GA	Distance by ACO-CS
15	158	167	146	158
20	231	321	199	229
25	1889	2113	1430	19902
30	433	543	402	425
42	21144	2543	1807	2032
50	542	669	449	538

Ondřej, Miča (2015) [21] ได้ศึกษาเปรียบเทียบการแก้ปัญหาการเดินทางของพนักงานขายด้วยขั้นตอนวิธีการค้นหาแบบทาบู ขั้นตอนวิธีการจำลองการอบเหนียว ขั้นตอนวิธีเชิงพันธุกรรม และ ขั้นตอนวิธีอาณานิคมมด (Ant Colony Algorithm: ACO) มีการทดลองจำนวนโหนดที่แตกต่างกันทั้งหมด 8 จำนวน ได้แก่ 51, 76, 96, 130, 159, 198, 225 และ 262 พบว่า ขั้นตอนวิธีเชิงพันธุกรรม จะมีความแปรปรวนเพิ่มขึ้นตามจำนวนของโหนด ขั้นตอนวิธีการจำลองการอบเหนียวและขั้นตอนวิธีการค้นหาแบบทาบู มีผลลัพธ์ที่ค่อนข้างใกล้เคียงกัน ขั้นตอนวิธีที่ดีที่สุดคือขั้นตอนวิธีอาณานิคมมด แต่มีความซับซ้อนของขั้นตอนเช่นกัน

ศรี กิจการ, พลอย พรรณ, แสงสว่าง และ อร อุไร. (2014) [22] ได้ยกกรณีศึกษาของเส้นทางการเดินรถขนส่งเครื่องสำอางเพื่อออกแบบและเพิ่มประสิทธิภาพ โดยมีจุดกระจายสินค้าเพียงแห่งเดียวและส่งไปยังร้านค้าย่อย 20 แห่งในเขตกรุงเทพฯ และปริมณฑล ผู้วิจัยได้ใช้ขั้นตอนวิธีการหาคำตอบที่ใกล้เคียงที่สุด (Nearest neighbor heuristic) เปรียบเทียบกับ ขั้นตอนวิธีการจำลองการอบเหนียว พบว่าขั้นตอนวิธีการจำลองการอบเหนียวสามารถจัดเส้นทางการเดินรถได้อย่างเหมาะสมและมีประสิทธิภาพมากกว่า เนื่องจากมีระยะทางรวมในการขนส่งต่ำกว่าเดิมบริษัทจึงเลือกใช้วิธีนี้ในการแก้ไขเพื่อลดเวลา รวมถึงค่าใช้จ่าย

### 2.2.2 การประยุกต์ใช้ขั้นตอนวิธีการป็นเขา

Jiang, Tuping, Gang Ren, และ Xing Zhao (2013) [23] ได้ทำแบบจำลองเน็ตเวิร์ค (Network Model) สำหรับการอพยพเมื่อเกิดภัยพิบัติธรรมชาติ ให้มีความรวดเร็ว และลดการจราจรติดขัดตามเส้นถนน (Bottleneck) โดยแบบจำลองได้ออกแบบให้มียุทธศาสตร์สองระดับ (Bi-Level) ระดับบน (Hi-Level) คือการหาค่าของ ขั้วเน็ตเวิร์ค (เน็ตเวิร์คจริง) สำหรับเส้นทางการสวนทาง โดยใช้ขั้นตอนวิธีการค้นหาแบบทาบู ในส่วนระดับล่าง (Low-Level) ใช้ขั้นตอนวิธีการป็นเขา ในการหาแผนการของเส้นทางที่ดีที่สุดในการลดทางแยกที่อาจทำให้เกิดรถติด ผลที่ได้แสดงให้เห็นว่า การใช้ขั้นตอนวิธีการแบบสองระดับ นอกจากจะได้ผลที่ดีแล้วยังได้ประสิทธิภาพที่สูงกว่าการใช้ขั้นตอนวิธีเดียว

Özcan, Samet C., และ Hilal Kaya (2018) [24] ได้ออกแบบเมืองอัจฉริยะ (Smart city) โดยประยุกต์ใช้ขั้นตอนวิธีการป็นเขา ผสานกับการใช้ OpenStreetMap (OSM) เพื่อเพิ่มประสิทธิภาพในการใช้งานจริง ให้ผู้ใช้งานสามารถออกแบบการเที่ยวตามสถานที่ต่าง ๆ ได้ด้วยตนเอง บนแอปพลิเคชันจะคำนวณเส้นทางการเดินทางไปยังสถานที่ที่ผู้ใช้เลือกให้มีระยะทางสั้นที่สุดและเวลาในการคำนวณเหมาะสมที่สุด ผู้วิจัยเริ่มทดลองกับเมืองโรมและอังการาเนื่องจากเป็นเมืองท่องเที่ยว โดยได้เลือกสถานที่เด่น ๆ เช่น พิพิธภัณฑ์ สวนสาธารณะ โบสถ์ เป็นต้น โดยผลลัพธ์ที่ได้ออกมาเป็นที่น่าพอใจ แต่หากมีสถานที่เพิ่มขึ้นคำตอบอาจจะผันผวนได้

### 2.2.3 การประยุกต์ใช้ขั้นตอนวิธีการจำลองการอบเหนียว

Grabusts, Peter, Jurijs Musatovs, and Vladimir Golenkov (2019) [25] ได้นำขั้นตอนวิธีการจำลองการอบเหนียวเพื่อหาเส้นทางที่สั้นที่สุดในการเดินทางระหว่าง 8 โรงงาน เป็นโรงงานเกี่ยวกับผลิตภัณฑ์ประเภทนมในประเทศเบลารุสโดยใช้ตำแหน่งจากแผนที่จริง (GPS) ตามละติจูด ลองจิจูด

Behnck, Lucas P และคณะ (2015) [26] ได้ประยุกต์ใช้ขั้นตอนวิธีการจำลองการอบเหนียวในการวางแผนเส้นทางของอากาศยานไร้คนขับ (Unmanned Aerial Vehicles: UAV) โดยใช้ ArduCopter และ Rastberry Pi ในการจำลอง UAV 2 รุ่น ในการทดลอง ผู้วิจัยได้ประยุกต์ใช้ขั้นตอนวิธีเพื่อให้รุ่นจำลอง UAV บินในระยงทางที่สั้นที่สุดและบินไปตามจุดสนใจของแต่ละรุ่น (Point Of Interest: POI) แบ่งปัจจัยในการทดสอบได้แก่  $\lambda = 0.15$  and  $\lambda = 0.30$  โดยผลที่ได้  $\lambda = 0.30$  UAV จำลองสามารถบินครบทุกจุด POI ส่วน  $\lambda = 0.15$  บินได้ถูกต้องประมาณ 70-75 เปอร์เซ็นต์

**ตาราง 3** สรุปงานวิจัยที่เกี่ยวข้องกับการแก้ปัญหาพนักงานขาย ขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียว

ผู้วิจัย	ชื่องานวิจัย	อัลกอริทึมที่ใช้	วัตถุประสงค์งานวิจัย
Wang Hui	An algorithm for Dyna order-picking in warehouse operations	SA, GA, ACO	เปรียบเทียบขั้นตอนวิธีต่าง ๆ ในการแก้ปัญหาพนักงานขาย โดยเปรียบเทียบเวลาในการหาคำตอบ, ความซับซ้อนของข้อมูล และความซับซ้อนของข้อมูล
Haider A และคณะ	Comparison of Algorithms for solving TSP	GA, NN, Greedy	ศึกษาจากการเดินทางไปยังเมืองต่าง ๆ ในประเทศสหรัฐอเมริกาโดยใช้ GA, NN, Greedy
Ansari, Abdul และคณะ	Comparison and analysis of solving TSP using GA, ACO and hybrid of ACO with GA, CS	GA, ACO, ACO-CS, ACO-GA	เปรียบเทียบขั้นตอนวิธีต่าง ๆ ในการแก้ปัญหาพนักงานขาย โดยผลสาน ACO กับ GA,CS,GA

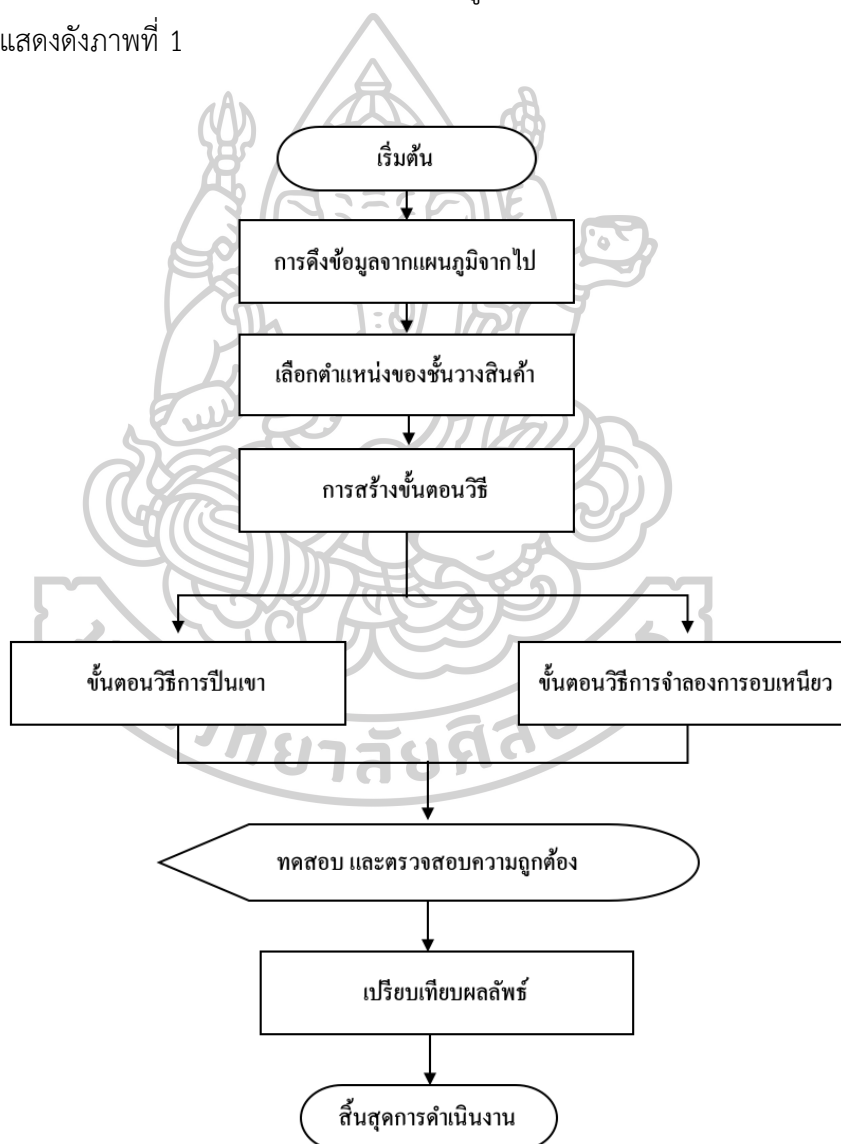


ตาราง 3 (ต่อ) สรุปงานวิจัยที่เกี่ยวข้องกับการแก้ปัญหาพนักงานขาย ขั้นตอนวิธีการป็นเขา และ ขั้นตอนวิธีการจำลองการอบเหนียว

ผู้วิจัย	ชื่องานวิจัย	อัลกอริทึมที่ใช้	วัตถุประสงค์งานวิจัย
Ondřej, Miča	Comparison of metaheuristic methods by solving traveling salesman problem	SA, GA, ACO, Tabu	เปรียบเทียบขั้นตอนวิธีต่าง ๆ ในการแก้ปัญหาพนักงานขาย โดยเปรียบเทียบ SA, GA, ACO, Tabu
ศรี กิจการ และคณะ	การ ออกแบบ เพื่อ เพิ่ม ประสิทธิภาพ เส้นทาง เติมนรถขนส่ง เครื่องสำอาง: กรณี ศึกษา.	SA, NN	คำนวณหาเส้นทางรถขนส่ง เครื่องสำอางที่สั้นที่สุดเปรียบเทียบระหว่าง SA และ NN
Jiang และคณะ	Evacuation route optimization based on Tabu search algorithm and hill-climbing algorithm	HCA, Tabu	แบบจำลองเน็ตเวิร์ค (Network Model) สำหรับการอพยพเมื่อเกิดภัยพิบัติธรรมชาติ โดยใช้ HCA เข้ากับ Tabu
Özcan และคณะ	An Analysis of Traveling Salesman Problem Utilizing Hill Climbing Algorithm for a Smart City Touristic Search on OpenStreetMap (OSM)	HCA	ออกแบบ Application สำหรับ Smart city โดยการประยุกต์ ขั้นตอนวิธีการป็นเขาผสานกับการใช้ OpenStreetMap (OSM)
Grabusts และคณะ	The application of simulated annealing method for optimal route detection between objects	SA	หาเส้นทางที่สั้นที่สุดในการเดินทางระหว่าง 8 โรงงาน โดยใช้ SA ในการคำนวณ
Behnck และคณะ	A modified simulated annealing algorithm for SUAVs path planning	SA	ประยุกต์ใช้ขั้นตอนวิธีการจำลองการอบเหนียวในการวางแผนเส้นทางของอากาศยานไร้คนขับ (Unmanned Aerial Vehicles) โดยใช้ ArduCopter และ Rastberry

### บทที่ 3 วิธีการทดลอง

งานวิจัยฉบับนี้เป็นการใช้ Microsoft Excel เพื่อเพิ่มประสิทธิภาพในการเดินหยิบสินค้าในคลังสินค้าตามคำสั่งซื้อ (Order Picking) โดยประยุกต์ใช้ขั้นตอนวิธีการป็นเขาและขั้นตอนวิธีการจำลองการอบเหนียว ด้วยภาษา Visual basic for application โดยขั้นตอนการดำเนินงานประกอบด้วย การดึงข้อมูลจากแผนภูมิจากไป (From to Chart) การเลือกตำแหน่งชั้นวางสินค้า การสร้างขั้นตอนวิธี การทดสอบและตรวจสอบความถูกต้อง และการวิเคราะห์เปรียบเทียบผลรายละเอียดแสดงดังภาพที่ 1



ภาพที่ 1 ขั้นตอนการดำเนินงาน

ภาพที่ 2 แสดงตัวอย่างของคลังสินค้าที่สร้างขึ้นบนโปรแกรม Microsoft Excel และภาพที่ 4 แสดงตัวอย่างเมื่อโปรแกรมทำการแปลงคลังสินค้าจากภาพที่ 2 ให้เป็นตำแหน่งช่องทางเดิน และตำแหน่งชั้นวางสินค้า โดยช่องสีเหลือง คือกำแพงหรือขอบเขตของคลังสินค้า ช่องสีน้ำตาลแทนด้วยตำแหน่งช่องทางเดิน เริ่มจากเลข 1 คือประตูทางเข้าออก ช่องสีฟ้า คือตำแหน่งของชั้นวางสินค้า

จากรูปจะเห็นว่า มีช่องทางเดินทั้งหมด 156 ช่อง และชั้นวางสินค้าทั้งหมด 160 ชั้น

W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
W																W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W																W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W	S		S	S		S	S		S	S		S	S		S	W
W																W
W	W	D	W	W	W	W	W	W	W	W	W	W	W	W	W	W

ภาพที่ 2 ตัวอย่างคลังสินค้าที่สร้างขึ้น

### 3.1 การดึงข้อมูลจากรางแผนภูมิจากไป (From to Chart)

เป็นการดึงข้อมูลของตารางแผนภูมิจากไปของคลังสินค้านั้น ๆ ที่สร้างขึ้น เพื่อนำมาคำนวณหา ระยะทางรวมในการเดินในชั้นตอนถัดไป โดยตารางแผนภูมิจากไปจะมีขนาดตามจำนวนช่องทางเดิน ของคลังสินค้าที่ได้สร้างขึ้น เพื่อบอกระยะทางของทุกจุดไปยังทุกจุด

จากภาพที่ 3 คือตัวอย่างของแผนภูมิจากไป (Form To Chart) ของคลังสินค้าที่ถูกสร้างขึ้นใน ภาพที่ 4 โดยผู้วิจัยได้ย่อขนาดตารางเพื่อให้เห็นว่าตารางเต็มมีขนาด 156 x 156

	A	B	C	D	E	F	G	H	ET	EU	EV	EW	EX	EY	EZ	FA	FB
1		1	2	3	4	5	6	7	.....	149	150	151	152	153	154	155	156
2	1	0	22	21	22	23	24	25	.....	7	8	9	10	11	12	13	14
3	2	22	0	1	2	3	4	5	.....	27	28	29	30	31	32	33	34
4	3	21	1	0	1	2	3	4	.....	26	27	28	29	30	31	32	33
5	4	22	2	1	0	1	2	3	.....	25	26	27	28	29	30	31	32
6	5	23	3	2	1	0	1	2	.....	24	25	26	27	28	29	30	31
7	6	24	4	3	2	1	0	1	.....	23	24	25	26	27	28	29	30
8	7	25	5	4	3	2	1	0	.....	22	23	24	25	26	27	28	29
9	8	26	6	5	4	3	2	1	.....	21	22	23	24	25	26	27	28
153	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....	.....
154	152	10	30	29	28	27	26	25	.....	3	2	1	0	1	2	3	4
155	153	11	31	30	29	28	27	26	.....	4	3	2	1	0	1	2	3
156	154	12	32	31	30	29	28	27	.....	5	4	3	2	1	0	1	2
157	155	13	33	32	31	30	29	28	.....	6	5	4	3	2	1	0	1
158	156	14	34	33	32	31	30	29	.....	7	6	5	4	3	2	1	0

ภาพที่ 3 ตัวอย่างแผนภูมิจากไป

### 3.2 การเลือกตำแหน่งจากชั้นวางสินค้า

การเลือกตำแหน่งชั้นวางสินค้านั้นจะเลือกจากคลังสินค้าที่ร่างบน Microsoft Excel ดังตัวอย่างจากภาพที่ 4 เมื่อเลือกตำแหน่งชั้นวางแล้ว โปรแกรมจะคำนวณดูว่าชั้นวางนั้น ๆ ติดกับช่องทางเดินอะไรในทางซ้ายและขวา หากติดกับทั้งสองด้าน จะเลือกช่องทางเดินด้านซ้าย จากนั้นโปรแกรมจะเก็บค่าช่องทางเดินดังกล่าว แล้วแสดงเป็นค่าที่หัวตารางใหม่ เพื่อนำมาดึงค่าจากตาราง From to Chart ใหญ่ (ภาพที่.4) ไปเป็น From to Chart เฉพาะค่าที่ต้องการสำหรับการคำนวณหาระยะทางรวม

ตัวอย่าง เมื่อเลือกหยิบสินค้า S11, S47, S76, S112, S122 จากภาพที่ 4 โปรแกรมทำการแปลงจากตำแหน่งชั้นวางเป็นช่องทางเดินได้ดังนี้ 22, 55, 69, 102, 122 ตามลำดับ ตำแหน่งเหล่านี้จะกลายเป็นหัวตารางของตาราง From to Chart ใหม่ จากนั้นโปรแกรมจะดึงข้อมูลตารางของ From to Chart ใหญ่มาใส่ในตารางใหม่ เฉพาะตำแหน่งที่ต้องการ ได้ดังภาพที่ 5 โดยในตารางนั้นจะเริ่มจากตำแหน่งที่ 1 เสมอ เนื่องจากเป็นประตูทางเข้าและออก เลขหัวตารางยังสามารถบอกถึงลำดับการเรียงของตำแหน่งช่องทางเดินเช่นกัน เช่น จากภาพที่ 4 เรียงลำดับโดย 1 ไปยัง 22 ไปยัง 59 ไปยัง 102 ไปยัง 122

W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
W	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	W
W	S1	17	S2	S3	18	S4	S5	19	S6	S7	20	S8	S9	21	S10	W
W	S11	22	S12	S13	23	S14	S15	24	S16	S17	25	S18	S19	26	S20	W
W	S21	27	S22	S23	28	S24	S25	29	S26	S27	30	S28	S29	31	S30	W
W	S31	32	S32	S33	33	S34	S35	34	S36	S37	35	S38	S39	36	S40	W
W	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	W
W	S41	52	S42	S43	53	S44	S45	54	S46	S47	55	S48	S49	56	S50	W
W	S51	57	S52	S53	58	S54	S55	59	S56	S57	60	S58	S59	61	S60	W
W	S61	62	S62	S63	63	S64	S65	64	S66	S67	65	S68	S69	66	S70	W
W	S71	67	S72	S73	68	S74	S75	69	S76	S77	70	S78	S79	71	S80	W
W	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	W
W	S81	87	S82	S83	88	S84	S85	89	S86	S87	90	S88	S89	91	S90	W
W	S91	92	S92	S93	93	S94	S95	94	S96	S97	95	S98	S99	96	S100	W
W	S101	97	S102	S103	98	S104	S105	99	S106	S107	100	S108	S109	101	S110	W
W	S111	102	S112	S113	103	S114	S115	104	S116	S117	105	S118	S119	106	S120	W
W	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	W
W	S121	122	S122	S123	123	S124	S125	124	S126	S127	125	S128	S129	126	S130	W
W	S131	127	S132	S133	128	S134	S135	129	S136	S137	130	S138	S139	131	S140	W
W	S141	132	S142	S143	133	S144	S145	134	S146	S147	135	S148	S149	136	S150	W
W	S151	137	S152	S153	138	S154	S155	139	S156	S157	140	S158	S159	141	S160	W
W	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	W
W	W	1	W	W	W	W	W	W	W	W	W	W	W	W	W	W

ภาพที่ 4 ตัวอย่างเมื่อกำหนดช่องทางเดินและตำแหน่งชั้นวางสินค้า

### 3.3 สร้างขั้นตอนวิธี

ในการสร้างขั้นตอนวิธีนั้น ผู้วิจัยได้แบ่งออกเป็น สองขั้นตอนวิธีได้แก่ ขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียว ซึ่งทั้งสองขั้นตอนวิธีนั้นมีส่วนที่เหมือนกันคือ การสุ่มหาคำตอบใหม่ และการรวมระยะทาง โดยผู้วิจัยใช้การสลับแบบ 2 opt จากคำตอบเดิมเพื่อสร้างคำตอบใหม่ โดยจะเปลี่ยนตำแหน่งช่องทางเดินที่หัวตารางแล้วจึงดึงข้อมูลจากตาราง From to Chart ใหม่ อีกครั้งเพื่อหาระยะทางรวมใหม่

ตัวอย่าง สลับช่องทางเดินตำแหน่งที่ 22 กับ 69 จาก ภาพที่ 5 จะได้ตาราง From to Chart ใหม่ ดังภาพที่ 6

	A	B	C	D	E	F	G
1		1	22	55	69	102	122
2	1	0	19	24	18	7	5
3	22	19	0	13	13	12	14
4	55	24	13	0	8	17	19
5	69	18	13	8	0	11	13
6	102	7	12	17	11	0	2
7	122	5	14	19	13	2	0

ภาพที่ 5 ตัวอย่างตาราง From to Chart เฉพาะค่าที่ต้องการ

	A	B	C	D	E	F	G
1		1	69	55	22	102	122
2	1	0	18	24	19	7	5
3	69	18	0	8	13	11	13
4	55	24	8	0	13	17	19
5	22	19	13	13	0	12	14
6	102	7	11	17	12	0	2
7	122	5	13	19	14	2	0

ภาพที่ 6 ตัวอย่างตาราง Form to Chart ใหม่จากการสลับตำแหน่งช่องทางเดิน

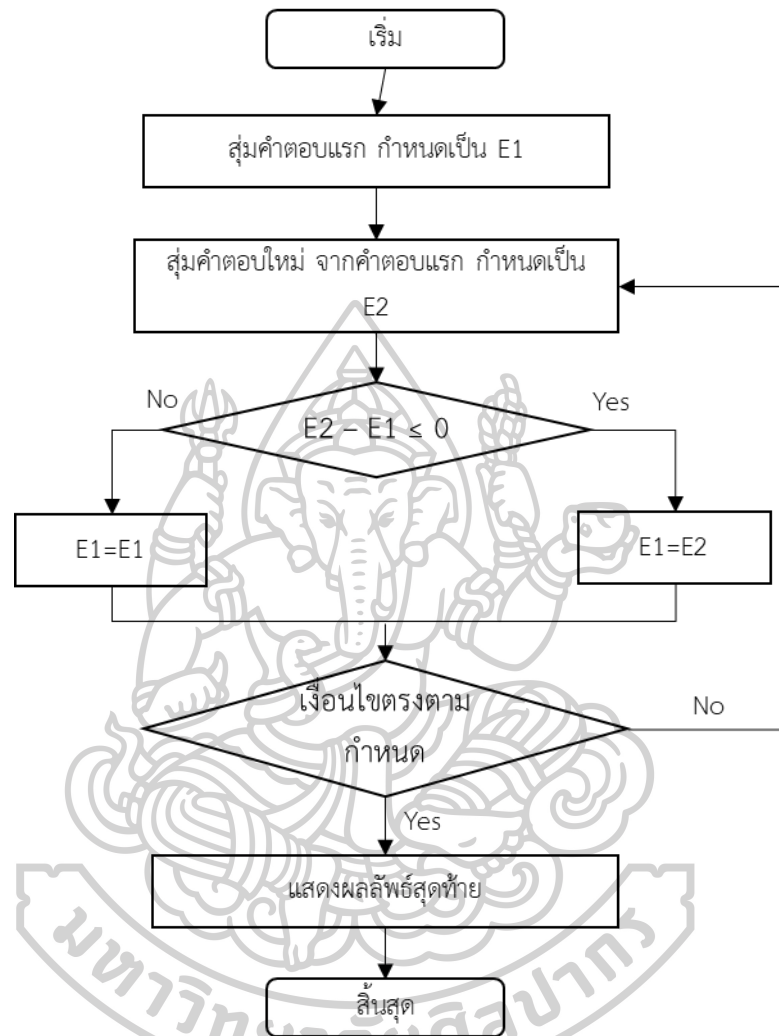
ในส่วนการรวมระยะทาง จากตาราง From to Chart ดังภาพที่ 7 รวมระยะทางตามแนวสีแดงรวมกับช่องสี่เหลี่ยมสีฟ้า

	A	B	C	D	E	F	G
1		1	69	55	22	102	122
2	1	0	18	24	19	7	5
3	69	18	0	8	13	11	13
4	55	24	8	0	13	17	19
5	22	19	13	13	0	12	14
6	102	7	11	17	12	0	2
7	122	5	13	19	14	2	0

ภาพที่ 7 การรวมระยะทางทั้งหมด

### 3.3.1 ขั้นตอนวิธีการป็นเขา

การทำงานของขั้นตอนวิธีการป็นเขา แสดงดังภาพที่ 8



ภาพที่ 8 Flow Chart การทำงานของขั้นตอนวิธีการป็นเขา

ในการดำเนินงานของขั้นตอนวิธีการป็นเขานัน รายละเอียดเป็น ดังนี้

(1) กำหนดตัวแปร

Path	แทน	ลำดับของตำแหน่งช่องทางเดินเริ่มต้น (Initial State) ที่ได้จากการสุ่มเริ่มต้น
E1	แทน	ระยะทางรวมจาก Path (Initial Solution)

(2) กำหนดรอบการทำงานของโปรแกรมตามขอบเขตที่กำหนด  
วนซ้ำ (Loop)

(3) สุ่มหาคำตอบใหม่ โดยผู้วิจัยใช้การสลับแบบ 2 opt ตามที่ได้กล่าวไว้ จากนั้น  
กำหนดตัวแปร

NewPath	แทน	ลำดับของตำแหน่งช่องทางเดินใหม่ (New State) ที่ได้จากการสุ่มเริ่มต้น
E2	แทน	ระยะทางรวมจาก NewPath (New Solution)

(4) เปรียบเทียบคำตอบเก่าและใหม่ โดย กำหนด  $dE$  แทน ผลต่างของสองจำนวน  
( $dE = E2 - E1$ )

ถ้า  $dE \leq 0$  จะกำหนดให้  $E1 = E2$  และ  $Path = NewPath$   
(เก็บค่า E2 ไว้เป็น E1 แล้วจึงสุ่มหาคำตอบใหม่จาก NewPath)

ถ้า  $dE > 0$  จะกำหนดให้  $E1 = E1$  และ  $Path = Path$   
(ปฏิเสธค่า E2 แล้วจึงสุ่มหาคำตอบใหม่จาก Path อีกครั้ง)

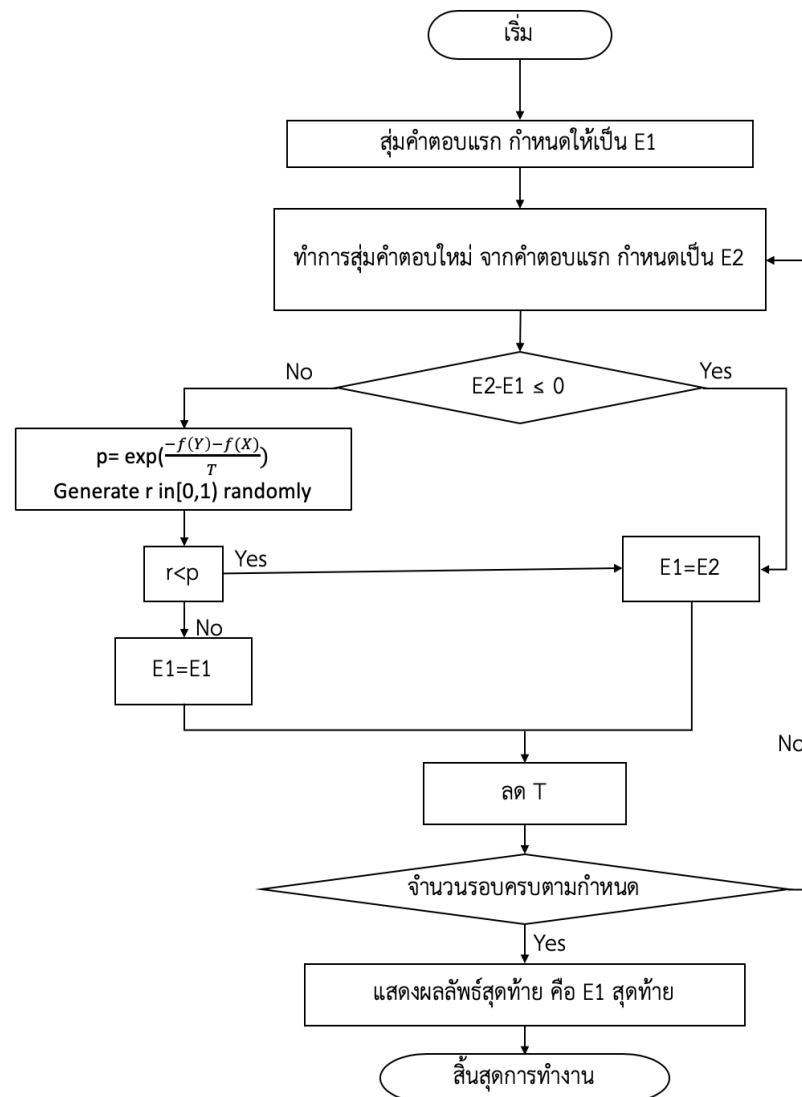
(5) ทำซ้ำข้อ (3) – (5) จนครบเงื่อนไขข้อ (2)

(6) เมื่อครบเงื่อนไข แสดงผลลัพธ์เป็น E1 และ Path ล่าสุด



### 3.3.2 ขั้นตอนวิธีการจำลองการอบเหนียว

การทำงานของขั้นตอนวิธีการจำลองการอบเหนียว เป็นดังภาพที่ 9



ภาพที่ 9 Flow Chart การทำงานของขั้นตอนวิธีการจำลองการอบเหนียว

ในการดำเนินงานของขั้นตอนวิธีการจำลองการอบเหนียวนั้น รายละเอียดเป็นดังนี้

(1) กำหนดตัวแปร

Path แทน ลำดับของตำแหน่งช่องทางเดินเริ่มต้น ที่ได้จากการสุ่มเริ่มต้น

E1 แทน ระยะทางรวมจาก Path (Initial Solution)

(2) กำหนดค่า T เริ่มต้นตามขอบเขตที่ได้กำหนดไว้

(3) กำหนด  $T < 0.01$  (หมายถึงการวนซ้ำไปเรื่อย เมื่อ T มีค่าน้อยกว่าหรือเท่ากับ 0.01 ให้หยุด)

วนซ้ำ (Loop)

(4) สุ่มหาคำตอบใหม่ โดยผู้วิจัยใช้การสลับแบบ 2 opt ตามที่ได้กล่าวไว้ จากนั้นกำหนดตัวแปร

NewPath แทน ลำดับของตำแหน่งช่องทางเดินใหม่ ที่ได้จากการสุ่มเริ่มต้น

E2 แทน ระยะทางรวมจาก NewPath (New Solution)

(5) เปรียบเทียบคำตอบเก่าและใหม่ โดย กำหนด dE แทน ผลต่างของสองจำนวน ( $dE = E2 - E1$ )

ถ้า  $dE \leq 0$  จะกำหนดให้  $E1 = E2$  และ  $Path = NewPath$

(เก็บค่า E2 ไว้เป็น E1 แล้วจึงสุ่มหาคำตอบใหม่จาก NewPath

ถ้า  $dE > 0$  ทำตามเงื่อนไข กำหนด

$Prob = \text{Exp}(-1 \times dE / T)$

$rand = \text{random}[0,1)$

ถ้า  $rand \leq Prob$  จะกำหนดให้  $E1 = E2$  และ  $Path = NewPath$

(เก็บค่า E2 ไว้เป็น E1 แล้วจึงสุ่มหาคำตอบใหม่จาก

NewPath

ถ้า  $rand > Prob$  จะกำหนดให้  $E1 = E1$  และ  $Path = Path$

(ปฏิเสธค่า E2 แล้วจึงสุ่มหาคำตอบใหม่จาก Path อีกครั้ง)

(6)  $T = T \times \alpha$  โดยที่  $\alpha$  มีค่าตามขอบเขตที่ได้กำหนดไว้

(7) ทำซ้ำข้อ (4) – (6) จนครบเงื่อนไขข้อ (3)

(8) เมื่อครบเงื่อนไข แสดงผลลัพธ์เป็น E1 และ Path ล่าสุด

### 3.4 ทดสอบ และตรวจสอบความถูกต้อง

ตรวจสอบความถูกต้องของคำตอบจากการคำนวณผลลัพธ์สุดท้ายที่ได้ (ค่า E1 ที่ได้จากขั้นตอนวิธี ที่ 3.3) ด้วยสมการดังนี้

$$\text{Sum} = \sum_{i,j}^n (d_{ij})$$

โดย  $i$  คือ ตำแหน่งเริ่มต้น  
 $j$  คือ ตำแหน่งปลายทาง  
 $n$  คือ จำนวนโหนดทั้งหมด  
 $d_{ij}$  คือ ระยะทางจาก  $i$  ไป  $j$

### 3.5 เปรียบเทียบผลที่ได้

เปรียบเทียบผลลัพธ์ที่ได้จากโปรแกรม เทียบกับผลลัพธ์ที่ได้จาก Linear Programming และเปรียบเทียบผลที่ได้ของขั้นตอนวิธีการป็นเขา กับขั้นตอนวิธีการจำลองการอบเหนียว

โดยคำนวณหาค่าเฉลี่ยของผลลัพธ์ของทั้งสองขั้นตอนวิธีจากโปรแกรม และค่าระยะทางรวม น้อยที่สุด (Min) และสูงที่สุด (Max) จาก Linear Programming เพื่อนำมาจัดทำกราฟ รวมถึงคำนวณหาค่าส่วนเบี่ยงเบนมาตรฐานเพื่อหาความแปรปรวนของผลลัพธ์ที่ได้ในแต่ละครั้ง

#### บทที่ 4 ผลและสรุปผลการวิจัย

ผู้วิจัยได้ทำการทดลองโปรแกรมเพื่อดูประสิทธิภาพในการแก้ปัญหาการเดินหีบสินค้าในคลังสินค้าจากการประยุกต์ใช้ของทั้งสองขั้นตอนวิธีบนโปรแกรม Microsoft Excel โดยเปรียบเทียบผลลัพธ์ที่ได้จากโปรแกรม กับผลลัพธ์ที่ได้จาก Linear Programming และเปรียบเทียบผลลัพธ์จากขั้นตอนวิธีการป็นเขา กับขั้นตอนวิธีการจำลองการอบเหนียวในการทดลองของแต่ละจำนวนช่องทางเดิน (5, 10, 15, 20, 30) ที่ได้กำหนดไว้ในขอบเขต

การทดลอง ผู้วิจัยได้สุ่มตำแหน่งช่องทางเดินของแต่ละจำนวนข้างต้น เพื่อนำมาคำนวณในโปรแกรม ดังนี้

ช่องทางเดิน 5 ช่อง

ได้แก่ ตำแหน่งช่องทางเดินที่ 20, 55, 69, 102, 122

ช่องทางเดิน 10 ช่อง

ได้แก่ ตำแหน่งช่องทางเดินที่ 18, 19, 22, 24, 36, 67, 122, 132, 133, 135

ช่องทางเดิน 15 ช่อง

ได้แก่ ตำแหน่งช่องทางเดินที่ 17, 22, 26, 33, 34, 54, 60, 65, 67, 102, 105, 125, 133, 137, 141

ช่องทางเดิน 20 ช่อง

ได้แก่ ตำแหน่งช่องทางเดินที่ 19, 20, 22, 26, 29, 32, 54, 56, 62, 64, 69, 87, 89, 91, 95, 98, 105, 128, 136, 139

ช่องทางเดิน 30 ช่อง

ได้แก่ ตำแหน่งช่องทางเดินที่ 3, 9, 11, 17, 18, 23, 35, 41, 47, 54, 57, 63, 71, 77, 80, 85, 94, 107, 109, 111, 114, 119, 125, 127, 131, 138, 139, 144, 147, 152

ผลการทดลองที่ได้มีรายละเอียด ดังต่อไปนี้

จากตารางที่ 4 แสดงค่าเฉลี่ย ( $\bar{X}$ ) จากการทดลองโปรแกรมซ้ำทั้งหมด 5 รอบ และ ผลที่ได้จาก Linear Programming ประกอบไปด้วยค่าระยะทางรวมสูงที่สุด (Max) และค่าระยะทางรวมน้อยที่สุด (Min)



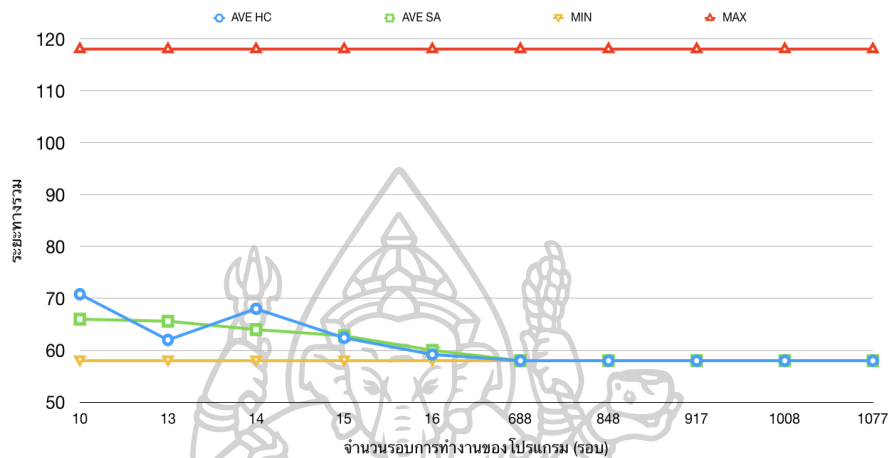
ตารางที่ 4 นำมาแสดงดังภาพที่ 10 – 14 เป็นรูปภาพเปรียบเทียบระหว่างผลที่ได้ของทั้งสองขั้นตอนวิธี และผลที่ได้ของ Linear Programming โดยผู้วิจัยได้กำหนดให้

แกน X	แทน	จำนวนรอบการทำงานของโปรแกรม
แกน Y	แทน	ระยะทางรวม
เส้นสีแดง (สามเหลี่ยมหัวตั้ง)	แทน	ค่าระยะทางรวมสูงสุดที่สุด จาก Linear Programming เส้นสีเหลือง (สามเหลี่ยมหัวกลับ)
เหลือง (สามเหลี่ยมหัวกลับ)	แทน	ค่าระยะทางรวมน้อย แทน ค่าเฉลี่ยจากผลของขั้นตอนวิธีการ
ฟ้า (วงกลม)	แทน	ค่าเฉลี่ยจากผลของขั้นตอนวิธีการจำลองการอบเหนียว
เส้นสีเขียว (สี่เหลี่ยม)	แทน	ที่สุด จาก Linear Programming เส้นสีฟ้า (วงกลม)
ปีกกา	แทน	ระยะห่าง (Gap) ระหว่างสองขั้นตอนวิธี กับ ค่าระยะทางรวมน้อยที่สุด

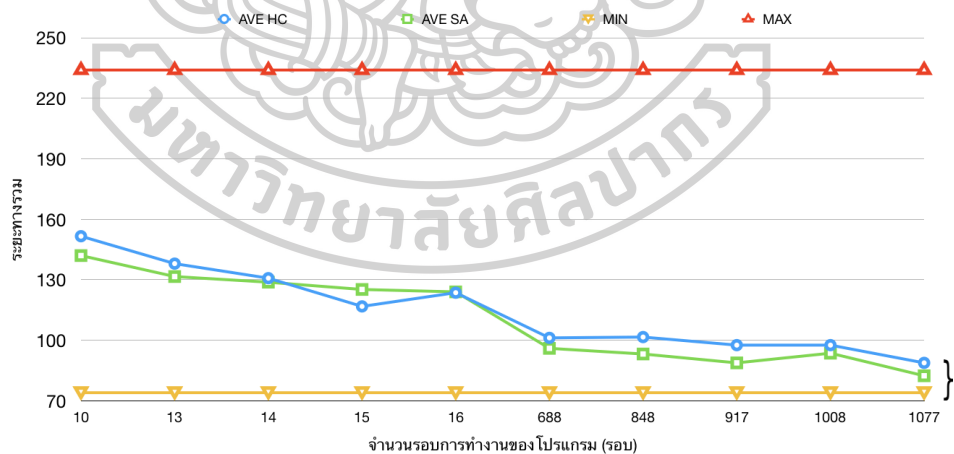
จากภาพจะเห็นได้ว่าผลลัพธ์จากการทดลองโปรแกรมในจำนวนช่องทางเดินต่าง ๆ ของทั้งสองขั้นตอนวิธีดีกว่าค่าระยะทางรวมสูงสุดที่สุดจาก Linear Programming หากพิจารณาด้วยตาจะพบว่าผลที่ได้จากขั้นตอนวิธีการปีนเขาและขั้นตอนวิธีการจำลองการอบเหนียว อยู่ใกล้ผลระยะทางรวมน้อยที่สุด มากกว่า ระยะทางรวมสูงสุด กล่าวคือ เส้นกราฟสีฟ้า และสีเขียว อยู่ใกล้เส้นกราฟสีเหลืองมากกว่าเส้นกราฟสีแดง และ แนวโน้มเส้นกราฟของทั้งสองขั้นตอนวิธี คู่เข้าหา (Coverage) เส้นระยะทางรวมน้อยที่สุด โดยหากดูระยะห่าง (ปีกกา) ระหว่างเส้นกราฟของสองขั้นตอนวิธี (สีเขียวและสีฟ้า) กับเส้นกราฟของระยะทางรวมน้อยที่สุด (สีเหลือง) จะพบว่ายิ่งจำนวนช่องทางเดินมาก ระยะห่าง (ปีกกา) จะยิ่งมากขึ้นตามไปด้วย

ในส่วนการเปรียบเทียบระหว่างสองขั้นตอนวิธีผู้วิจัยได้พิจารณาภาพกราฟที่ 10 - 14 เป็นสองช่วง ช่วงแรกคือ จำนวนรอบการทำงานของโปรแกรมที่ 10, 13, 14, 15, 16 ช่วงที่สอง คือ จำนวนรอบการทำงานของโปรแกรมที่ 688, 848, 917, 1008, 1077 เนื่องจากจำนวนรอบการทำงานของโปรแกรมที่ค่อนข้างแตกต่างกันอย่างชัดเจน รวมถึงผลลัพธ์ที่ได้ ผู้วิจัยจึงเลือกวิเคราะห์แยกกัน ช่วงแรกพบว่าผลที่ได้ของทั้งสองขั้นตอนวิธีใกล้เคียงกัน แต่เนื่องจากจำนวนรอบการทำงานของโปรแกรมที่น้อยจนเกินไปและใกล้เคียงกันมาก ผลที่ได้จึงไม่สามารถระบุได้ชัดเจนว่าขั้นตอนวิธีใดดีกว่ากัน อย่างเช่นภาพที่ 10 พบว่า ในจำนวนรอบการทำงานที่ 10 ผลของขั้นตอนวิธีการจำลองการอบเหนียว ดีกว่าขั้นตอนวิธีการปีนเขา เมื่อจำนวนรอบการทำงานเป็น 13 ผลของขั้นตอนวิธีการจำลองการปีนเขากลับดีกว่า และเมื่อเปลี่ยนจำนวนรอบการทำงานเป็น 14 ขั้นตอน

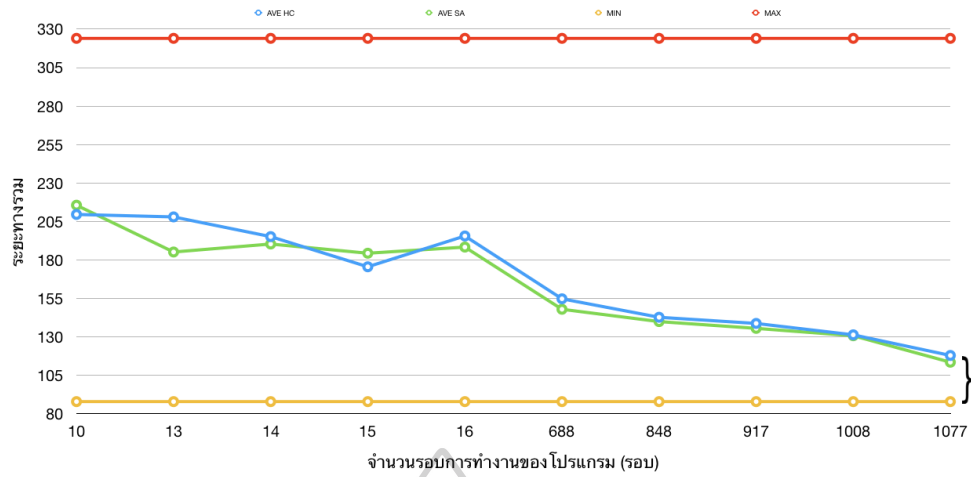
วิธีการจำลองการรอบเหนียวกลับมาดีกว่าเหมือนเดิม เป็นต้น ในช่วงที่สองพบว่าผลที่ได้ของทั้งสองขั้นตอนวิธีใกล้เคียงกันเช่นเดียวกับช่วงแรก แต่เมื่อจำนวนรอบการทำงานมากทำให้สามารถเห็นข้อแตกต่างได้ว่าขั้นตอนวิธีการจำลองการรอบเหนียว ได้ผลลัพธ์ที่ดีกว่าขั้นตอนวิธีการป็นเขา ภาพที่ 10 - 14 แสดงเรียงตามจำนวนช่องทางเดินหีบสินค้า 5, 10, 15, 20, 30 ตามลำดับ



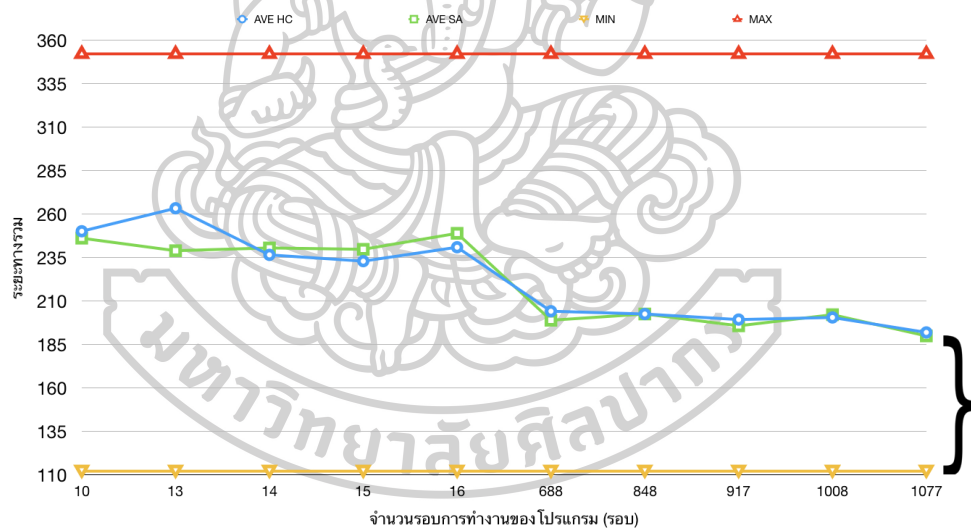
ภาพที่ 10 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และผลจาก Linear Programming สำหรับช่องทางเดิน 5 ช่อง



ภาพที่ 11 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงที่สุด น้อยที่สุดจาก Linear Programming สำหรับช่องทางเดิน 10 ช่อง

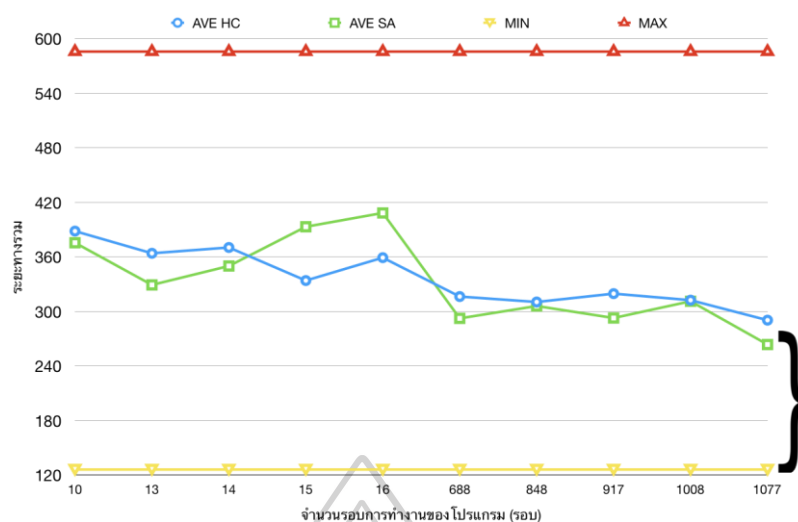


ภาพที่ 12 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงที่สุด น้อยที่สุดจาก Linear Programming สำหรับช่องทางเดิน 15 ช่อง



ภาพที่ 13 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงที่สุด น้อยที่สุดจาก Linear Programming สำหรับช่องทางเดิน 20 ช่อง





ภาพที่ 14 กราฟผลลัพธ์ของทั้งสองขั้นตอนวิธี และค่าสูงที่สุด น้อยที่สุดจาก Linear Programming สำหรับช่องทางเดิน 30 ช่อง

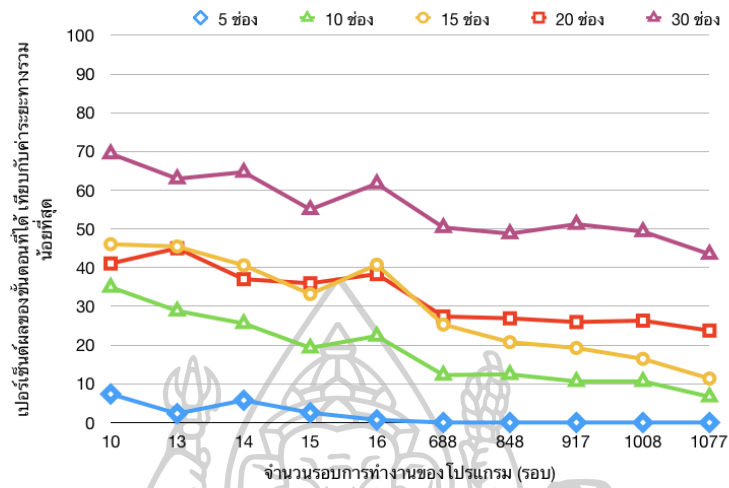
ผู้วิจัยได้คำนวณผลลัพธ์จากขั้นตอนที่ได้ เทียบกับค่าระยะทางรวมน้อยที่สุด เพื่อเปรียบเทียบจำนวนช่องทางเดินที่ต่างกัน ว่าส่งผลต่อผลลัพธ์หรือไม่ แสดงดังภาพที่ 15 และ 16

กำหนดให้ เส้นสีฟ้า คือ จำนวนช่องทางเดิน 5 ช่อง  
 เส้นสีเขียว คือ จำนวนช่องทางเดิน 10 ช่อง  
 เส้นสีเหลือง คือ จำนวนช่องทางเดิน 15 ช่อง  
 เส้นสีแดง คือ จำนวนช่องทางเดิน 20 ช่อง  
 เส้นสีม่วง คือ จำนวนช่องทางเดิน 30 ช่อง

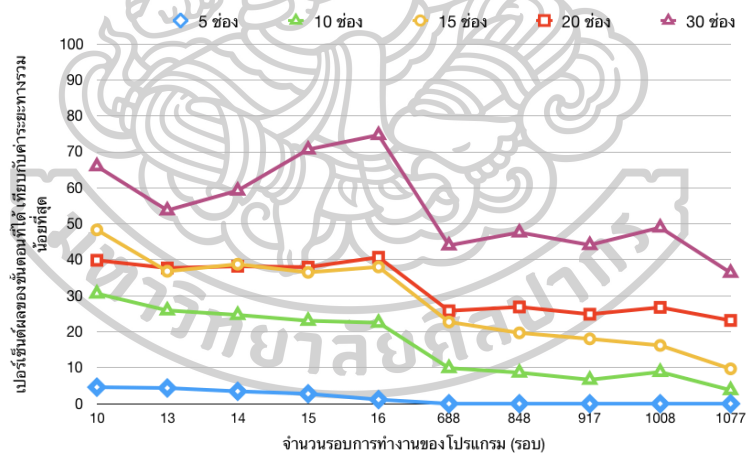
จากภาพกราฟที่ 15 และ 16 เป็นผลที่ได้จากภาพที่ 10 – 14 พบว่ายิ่งจำนวนช่องทางเดินมากขึ้น ส่งผลให้ผลลัพธ์ที่ได้มีค่าระยะทางห่างจากค่าระยะทางรวมน้อยที่สุดมากขึ้นเช่นกัน ตามที่กล่าวไปในย่อหน้าข้างต้น โดยที่ยังเส้นกราฟเข้าใกล้ 0 แสดงว่าผลที่ได้มีค่าใกล้กับค่าระยะทางรวมน้อยที่สุดมากเท่านั้น จะเห็นได้ว่าจากทั้งสองขั้นตอนวิธีเส้นสีฟ้า (ช่องทางเดิน 5 ช่อง) เข้าใกล้ 0 มากกว่าเส้นสีม่วง (ช่องทางเดิน 30 ช่อง)

จากผลลัพธ์ที่กล่าวมาสามารถวิเคราะห์ได้ว่าการประยุกต์ใช้ขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียวในโปรแกรม Microsoft Excel นั้นสามารถเพิ่มประสิทธิภาพการเดินทางหีบสินค้าในคลังสินค้าได้ โดยเมื่อทดลองโปรแกรมของทั้งสองขั้นตอนวิธีระยะทางรวมที่ได้มีค่าน้อยกว่าระยะทางรวมสูงที่สุดจาก Linear Programming ทุกจำนวน

ภาพที่ 15 – 16 แสดงตามขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียวตามลำดับ



ภาพที่ 15 กราฟผลของขั้นตอนวิธีการป็นเขาในแต่ละช่องทางเดินเมื่อเทียบกับค่าระยะทางรวมน้อยที่สุด



ภาพที่ 16 กราฟผลของขั้นตอนวิธีการจำลองการอบเหนียวในแต่ละช่องทางเดินเมื่อเทียบกับค่าระยะทางรวมน้อยที่สุด

ผู้วิจัยได้นำผลที่ได้จากการทดลองโปรแกรมมาคำนวณหาส่วนเบี่ยงเบนมาตรฐาน (Standard Deviation; SD) เพื่อนำมาวิเคราะห์การกระจายของผลลัพธ์ที่ได้ แสดงดังตารางที่ 5

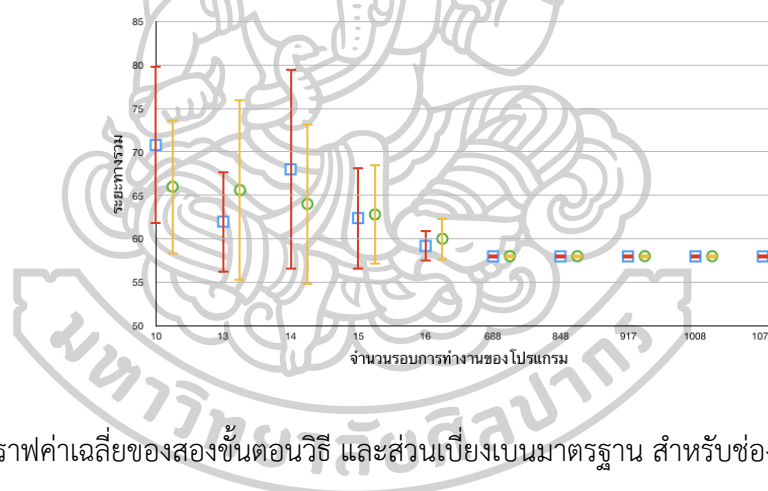
ตาราง 5 ผลการทดลองโปรแกรมจากสองขั้นตอนวิธี และค่าส่วนเบี่ยงเบนมาตรฐานจากการคำนวณ

จำนวน ช่อง ทางเดิน	ขั้นตอน วิธี		จำนวนรอบการทำงานของโปรแกรม									
			10	13	14	15	16	688	848	917	1008	1077
5	HCA	$\bar{X}$	70.80	62	68	62.40	59.20	58	58	58	58	58
		SD	9.12	5.83	11.58	5.90	1.79	0	0	0	0	0
	SA	$\bar{X}$	66	65.60	64	62.80	60	58	58	58	58	58
		SD	7.75	10.43	9.27	5.76	2.45	0	0	0	0	0
10	HCA	$\bar{X}$	151.60	138	130.80	116.80	123.60	101.20	101.60	97.60	97.60	88.80
		SD	15.13	11.75	3.35	22.21	7.13	8.44	7.27	8.30	12.44	7.29
	SA	$\bar{X}$	142	131.60	128.80	125.20	124	96	93.20	88.80	93.60	82.40
		SD	8.12	13.81	19.22	8.32	7.07	10.30	8.20	5.40	9.32	5.55
15	HCA	$\bar{X}$	209.6	208	195.2	175.6	195.6	154.8	142.8	138.8	131.4	118
		SD	16.76	28.81	19.88	29.10	34.51	4.60	28.00	13.08	24.90	22.14
	SA	$\bar{X}$	215.6	185.2	190.4	184.4	188.4	148	140	135.6	130.8	113.6
		SD	18.99	26.89	5.18	16.94	23.55	9.38	25.73	10.04	20.72	22.20
20	HCA	$\bar{X}$	250	263.20	236.40	232.80	240.80	204.00	202.40	199.20	200.40	191.80
		SD	26.76	15.27	7.54	4.60	14.53	4.47	15.90	2.28	6.99	12.97
	SA	$\bar{X}$	246	238.80	240.40	239.60	248.80	198.80	202.40	195.60	202	189.80
		SD	13.34	13.83	12.60	9.94	24.27	4.60	6.39	9.21	6.16	12.22
30	HCA	$\bar{X}$	388.40	364	370.40	334.00	359.20	316.40	310.40	319.60	312.40	290.40
		SD	30.80	15.30	17.97	21.86	40.06	12.76	20.37	41.34	12.44	40.73
	SA	$\bar{X}$	375.60	329.20	350.00	393.20	408.40	292.40	306	292.80	311.20	263.60
		SD	24.67	22.52	25.18	17.81	27.18	41.96	8.94	44.96	9.12	51.58

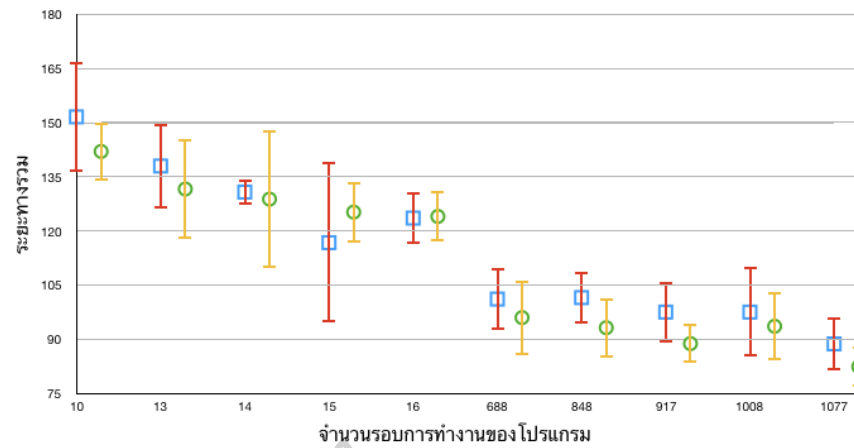
นำผลจากตารางที่ 5 มาสร้างกราฟเพื่อดูการกระจายของผลลัพธ์ที่ได้ ดังภาพที่ 17 - 21 โดย

แกน X	แทน	ระยะทางรวม
แกน Y	แทน	จำนวนรอบการทำงานของโปรแกรม
สีเหลี่ยมสีฟ้า	แทน	ค่าเฉลี่ยของขั้นตอนวิธีการป็นเขา และเส้นสีแดงคือ ค่า SD
วงกลมสีเขียว	แทน	ค่าเฉลี่ยของขั้นตอนวิธีการจำลองการอบเหนียว และเส้นสีเหลืองคือ ค่า SD

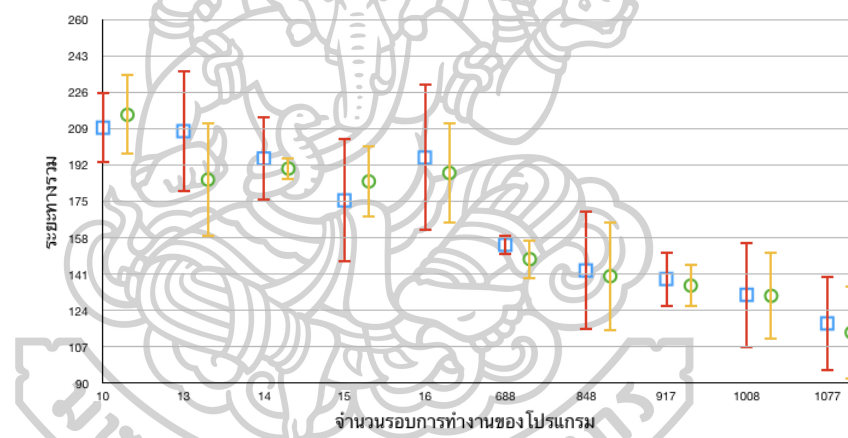
พบว่าผลลัพธ์มีการกระจายตัวมาก ซึ่งก็คือในการทดลองโปรแกรมด้วยขอบเขตต่าง ๆ อย่งละ 5 รอบ ผลที่ได้แต่ละรอบ ค่าตอบค่อนข้างกว้างเนื่องจากการทำงานของสองขั้นตอนวิธี คือการสุ่มค่าคำตอบแรกขึ้นมา ดังนั้นโอกาสในการสุ่มแต่ละค่าจึงหลากหลาย ผลลัพธ์ที่ได้จึงหลากหลายเช่นกัน ภาพที่ 17 - 21 แสดงเรียงตามจำนวนช่องทางเดินหีบสินค้า 5, 10, 15, 20, 30 ตามลำดับ



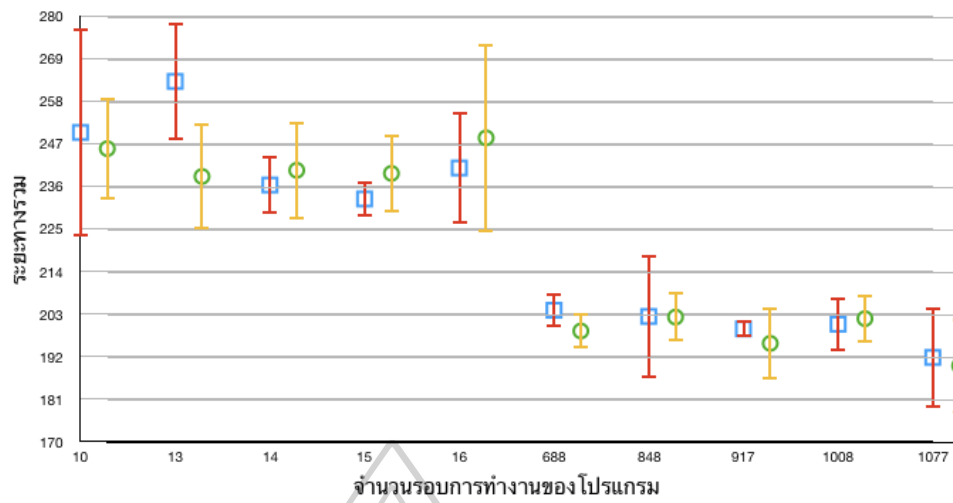
ภาพที่ 17 กราฟค่าเฉลี่ยของสองขั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 5 ช่อง



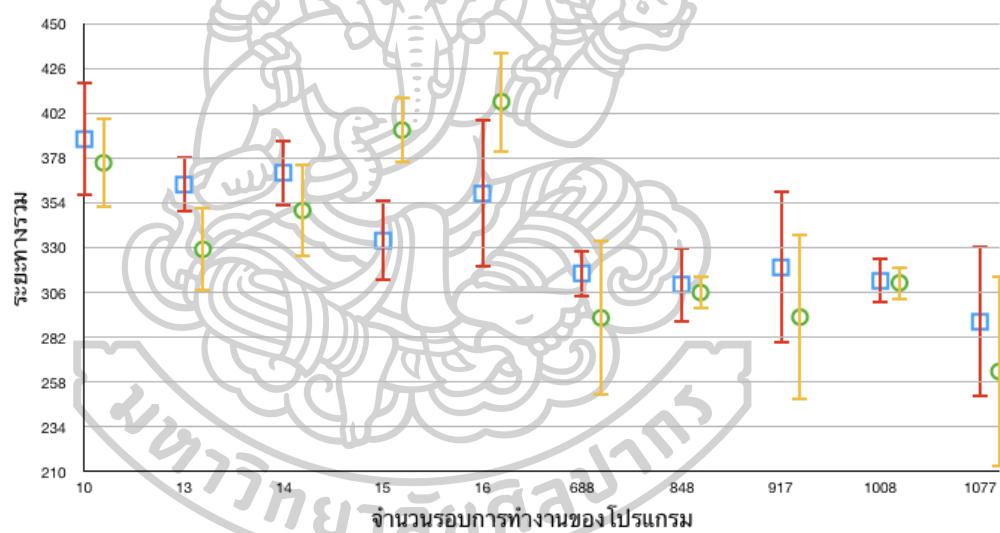
ภาพที่ 18 กราฟค่าเฉลี่ยของสองขั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 10 ช่อง



ภาพที่ 19 กราฟค่าเฉลี่ยของสองขั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 15 ช่อง



ภาพที่ 20 กราฟค่าเฉลี่ยของสองขั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 20 ช่อง



ภาพที่ 21 กราฟค่าเฉลี่ยของสองขั้นตอนวิธี และส่วนเบี่ยงเบนมาตรฐาน สำหรับช่องทางเดิน 30 ช่อง

## บทที่ 5

### สรุปผลการวิจัยและข้อเสนอแนะ

#### 5.1 สรุปผลการวิจัย

งานวิจัยฉบับนี้เป็นการแก้ปัญหาการเดินทางของพนักงานขาย ในคลังสินค้า โดยใช้ขั้นตอนวิธีการป็นเขา และขั้นตอนวิธีการจำลองการอบเหนียว บนโปรแกรม Microsoft Excel ด้วยตัวแปรที่แตกต่างกันไปทั้ง จำนวนรอบในการทำงานของโปรแกรม , ค่า  $T$  เริ่มต้น , อัตราการลดของ  $T$  ( $\alpha$ ) และ จำนวนช่องทางเดิน โดยในแต่ละค่าตัวแปรทำการทดลองทั้งหมด 5 รอบแล้วจึงนำมาหาผลเฉลี่ย

สรุปได้ว่าทั้งสองขั้นตอนวิธีสามารถเพิ่มประสิทธิภาพการเดินทางหีบสินค้าในคลังสินค้าได้ โดยที่ระยะทางรวมของทั้งสองขั้นตอนวิธี น้อยกว่าระยะทางรวมสูงที่สุด ที่คำนวณจาก Linear Programming โดยจำนวนช่องทางเดินมีผลต่อผลลัพธ์ โดยเมื่อจำนวนช่องทางเดินมีจำนวนมากขึ้น ส่งผลให้ผลลัพธ์ที่ได้ด้อยห่างจากระยะทางรวมน้อยที่สุดเช่นกัน และเมื่อเปรียบเทียบระหว่างสองขั้นตอนวิธี เมื่อจำนวนรอบการทำงานของโปรแกรมน้อย หรือ อัตราการลดของ  $T$  ( $\alpha$ ) เป็น 0.5 ผลของทั้งสองขั้นตอนวิธีมีความแปรปรวน เนื่องจากจำนวนรอบที่น้อยจนเกินไปจึงส่งผลต่อผลลัพธ์ และเมื่อจำนวนรอบการทำงานของโปรแกรมมากขึ้น หรือ อัตราการลดของ  $T$  ( $\alpha$ ) เป็น 0.99 พบว่าขั้นตอนวิธีของการจำลองการอบเหนียวได้ผลลัพธ์ที่ดีกว่าขั้นตอนวิธีการป็นเขาแต่ไม่มากนัก และผลลัพธ์ที่ได้ดีกว่าจำนวนรอบการทำงานของโปรแกรมน้อย (อัตราการลดของ  $T$  ( $\alpha$ ) เป็น 0.5) แสดงให้เห็นว่าจำนวนรอบมีผลโดยตรงต่อผลลัพธ์ และเวลาในการทำงานของโปรแกรมเช่นกัน

จากย่อหน้าข้างต้นแสดงให้เห็นว่าผลลัพธ์ที่ได้เป็นไปในทิศทางเดียวกันกับผลงานวิจัยอื่น ๆ ที่ผู้วิจัยได้ทำการศึกษา มา คือ ขั้นตอนวิธีการจำลองการอบเหนียวได้ผลลัพธ์ที่ดีกว่าขั้นตอนการป็นเขา และ แสดงให้เห็นถึงค่าตัวแปรต่าง ๆ ที่ส่งผลต่อผลลัพธ์ ดังเช่น ค่าอัตราการลดของ  $T$  ( $\alpha$ ) เมื่อมีค่าน้อย จะส่งผลให้จำนวนรอบการทำงานน้อย เป็นผลให้ผลลัพธ์ที่ได้ไม่ต่ำเท่าที่ควร จึงเป็นเหตุผลที่งานวิจัยอื่น ๆ นั้นใช้ค่าอัตราการลดของ  $T$  ( $\alpha$ ) สูงหรือใกล้เคียง 1 ทั้งสิ้น

#### 5.2 ข้อเสนอแนะ

5.2.1. หากต้องการความรวดเร็วในการหาผลลัพธ์ ควรใช้อัตราการลดของ  $T$  เท่ากับ 0.5 แต่อย่างไรก็ตามผลลัพธ์ที่ได้จะไม่ใช่ว่าค่าตอบที่ดีที่สุด ติดใน Local Optimal

5.2.2 หากต้องการผลลัพธ์ที่ดีที่สุด มีโอกาสเป็น Global Optimal สูงขึ้น ควรใช้อัตราการลดของ  $T$  เข้าใกล้ ค่า 1 เช่น 0.98 , 0.99 เป็นต้น แต่ระยะเวลาในการทำงานของโปรแกรมนานขึ้นเช่นกัน

## รายการอ้างอิง

1. Lu, W., et al., *An algorithm for dynamic order-picking in warehouse operations*. European Journal of Operational Research, 2016. 248(1): p. 107-122.
2. Key, R. and A. Dasgupta. *Warehouse Pick Path Optimization Algorithm Analysis*. in *Proceedings of the International Conference on Foundations of Computer Science (FCS)*. 2015. The Steering Committee of The World Congress in Computer Science, Computer ...
3. Theys, C., et al., *Using a TSP heuristic for routing order pickers in warehouses*. European Journal of Operational Research, 2010. 200(3): p. 755-763.
4. Hui, W., *Comparison of several intelligent algorithms for solving TSP problem in industrial engineering*. Systems Engineering Procedia, 2012. 4: p. 226-235.
5. Cormen, T.H., *Algorithms unlocked*. 2013: Mit Press.
6. Goodrich, M.T. and R. Tamassia, *Algorithm design and applications*. 2014: Wiley Publishing.
7. Khalil, M., et al. *Algorithm to solve travel salesman problem efficiently*. in *2016 13th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*. 2016. IEEE.
8. Skiena, S.S., *The algorithm design manual: Text*. Vol. 1. 1998: Springer Science & Business Media.
9. Cárdenas-Montes, M., *Creating hard-to-solve instances of travelling salesman problem*. Applied Soft Computing, 2018. 71: p. 268-276.
10. Johnson, A.W. and S.H. Jacobson, *A class of convergent generalized hill climbing algorithms*. Applied Mathematics and Computation, 2002. 125(2-3): p. 359-373.
11. Al-Betar, M.A.,  *$\beta$ -Hill climbing: an exploratory local search*. Neural Computing and Applications, 2017. 28(1): p. 153-168.
12. Wongsanga, S., et al., *An Application of Hill Climbing Algorithm for Optimal Ratio of Ground Bottom Ash in Concrete*. KMUTT Research and Development Journal, 2012. 35(2): p. 173-186.



13. Kirkpatrick, S., C.D. Gelatt, and M.P. Vecchi, *Optimization by simulated annealing*. science, 1983. 220(4598): p. 671-680.
14. Du, K.-L. and M. Swamy, *Search and optimization by metaheuristics*. Techniques and Algorithms Inspired by Nature; Birkhauser: Basel, Switzerland, 2016.
15. Rutenbar, R.A., *Simulated annealing algorithms: An overview*. IEEE Circuits and Devices magazine, 1989. 5(1): p. 19-26.
16. Henderson, D., S.H. Jacobson, and A.W. Johnson, *The theory and practice of simulated annealing*, in *Handbook of metaheuristics*. 2003, Springer. p. 287-319.
17. Ben-Ameur, W., *Computing the initial temperature of simulated annealing*. Computational Optimization and Applications, 2004. 29(3): p. 369-385.
18. Shojaee, K. *Investigation on the choice of the initial temperature in the simulated annealing: A mushy state SA for TSP*. in 2009 17th Mediterranean Conference on Control and Automation. 2009. IEEE.
19. Abdulkarim, H.A. and I.F. Alshammari, *Comparison of Algorithms for Solving Traveling Salesman Problem*. German Academic Exchange Service International Journal of Engineering and Advanced Technology (IJEAT), 2015. 4(6): p. 75-79.
20. Ansari, A.Q. and S. Katiyar. *Comparison and analysis of solving travelling salesman problem using GA, ACO and hybrid of ACO with GA and CS*. in 2015 IEEE workshop on computational intelligence: theories, applications and future directions (WCI). 2015. IEEE.
21. Ondřej, M. *Comparison of metaheuristic methods by solving travelling salesman problem*. in *Proceedings of the 9th International Scientific Conference INPROFORUM: Common challenges-Different solutions-Mutual dialogue*. 2015. Jihočeská univerzita v Českých Budějovicích.
22. วงศ์, ท.จ. and ๒. วลี, กรณีย์ ศึกษา การ เพิ่ม ประสิทธิภาพ การ ใช้ รถ ขนส่ง ของ บริษัท ABC [ซีดี-รอม]/เก วลี หรือ จันท์ วงศ์. The University of the Thai Chamber of Commerce.
23. Jiang, T., G. Ren, and X. Zhao, *Evacuation route optimization based on Tabu search algorithm and hill-climbing algorithm*. Procedia-Social and Behavioral Sciences, 2013. 96: p. 865-872.
24. Özcan, S.C. and H. Kaya. *An Analysis of Travelling Salesman Problem Utilizing*

*Hill Climbing Algorithm for a Smart City Touristic Search on OpenStreetMap (OSM).* in 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT). 2018. IEEE.

25. Grabusts, P., J. Musatovs, and V. Golenkov, *The application of simulated annealing method for optimal route detection between objects.* Procedia Computer Science, 2019. 149: p. 95-101.
26. Behnck, L.P., et al., *A modified simulated annealing algorithm for SUAVs path planning.* IFAC-PapersOnLine, 2015. 48(10): p. 63-68.



## ประวัติผู้เขียน

ชื่อ-สกุล	จิราพร สกุลวรารักษ์
วัน เดือน ปี เกิด	02 เมษายน 2538
สถานที่เกิด	สงขลา
วุฒิการศึกษา	ม.1- ม.6 โรงเรียนสาธิตมหาวิทยาลัยราชภัฏนครปฐม ปริญญาตรี วิศวกรรมคอมพิวเตอร์ สถาบันนานาชาติสิรินธร มหาวิทยาลัยธรรมศาสตร์
ที่อยู่ปัจจุบัน	19/2 ถนนนาสร้าง-นาขุม ตำบลนครปฐม อำเภอเมือง จังหวัดนครปฐม 73000
รางวัลที่ได้รับ	การประชุมวิชาการและนำเสนอผลงานทางวิศวกรรม นวัตกรรม และการจัดการอุตสาหกรรมอย่างยั่งยืน ครั้งที่ 8 ประจำปี 2562

