

อัลกอริทึมสำหรับปัญหาการจัดการเส้นทางการเดินรถของพนักงานขายของแบบไม่สมมาตร

จุฑาทิพย์ พุฒพันธ์

TNI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

บัณฑิตวิทยาลัย สาขาเทคโนโลยีวิศวกรรม

สถาบันเทคโนโลยีไทย-ญี่ปุ่น

ปีการศึกษา 2557

ALGORITHMS FOR ASYMMETRIC TRAVELLING SALESMAN PROBLEMS

Jutathip Putponpai



A Thesis Submitted in Partial Fulfillment of Requirements
for The Degree of Master of Engineering Program in Engineering Technology

Graduate School

Thai-Nichi Institute of Technology

Academic Year 2014

หัวข้อวิทยานิพนธ์

อัลกอริทึมสำหรับปัญหาการจัดการเส้นทางการเดินทางของ
พนักงานขายของแบบไม่สมมาตร

โดย

จุฑาทิพย์ พุฒพันธ์

สาขาวิชา

เทคโนโลยีวิศวกรรม

อาจารย์ที่ปรึกษาวิทยานิพนธ์

ผู้ช่วยศาสตราจารย์ ดร. พิศุทธิ์ พงศ์ชัยฤกษ์

บัณฑิตวิทยาลัย สถาบันเทคโนโลยีไทย-ญี่ปุ่น อนุมัติให้บัณฑิตวิทยาลัยนี้เป็น
ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

.....คณบดีบัณฑิตวิทยาลัย

(รองศาสตราจารย์ ดร. พิชิต สุขเจริญพงษ์)

วันที่.....เดือน.....พ.ศ.....

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(ดร. บุญทริกา เกษมสันติธรรม)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. วิภาวดี วงษ์สุวรรณ)

.....กรรมการ

(ดร. กันติชา กิตติพิรัช)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์

(ผู้ช่วยศาสตราจารย์ ดร. พิศุทธิ์ พงศ์ชัยฤกษ์)

จุฑาทิพย์ พุฒพันธ์ : อัลกอริทึมสำหรับปัญหาการจัดการเส้นทางการเดินทางของพนักงานขาย
ของแบบไม่สมมาตร. อาจารย์ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร. พิศุทธิ์ พงศ์ชัยฤกษ์,
71 หน้า.

งานวิจัยนี้ศึกษาการจัดการเส้นทางการเดินทาง จากการศึกษาพบว่าระยะทางจากเมืองใด
เมืองหนึ่งไปยังอีกเมืองหนึ่ง อาจมีระยะทางไป-กลับไม่เท่ากัน อีกทั้งจำนวนเมืองมากมีผลต่อการ
จัดหาเส้นทางเดินทาง ผู้วิจัยจึงนำปัจจัยต่างๆมาประกอบการศึกษาการจัดการเส้นทางการเดินทาง ให้เดินทาง
จากเมืองหนึ่งไปยังทุกเมืองให้ได้ทั้งหมดด้วยระยะทางที่สั้นที่สุด การศึกษาวิจัยครั้งนี้ได้ศึกษาวิธีการ
หาคำตอบด้วยวิธีการ Deterministic Algorithm, Stochastic Algorithm และ Local Search
algorithm แบบ SWAP และ INSERT มาประกอบการจัดการเส้นทางเพื่อใช้ในการปรับปรุงผลลัพธ์
ของระยะทางรวม ผลจากการคำนวณแสดงให้เห็นถึงการลดลงของระยะทางรวมของเส้นทางการเดิน
ในแต่ละขั้นตอน พร้อมกับพิจารณาเปรียบเทียบระยะทางรวมของวิธีการในงานวิจัยในแต่ละขั้นตอน

บัณฑิตวิทยาลัย

สาขาวิชา เทคโนโลยีวิศวกรรม

ปีการศึกษา 2557

ลายมือชื่อนักศึกษา

ลายมือชื่ออาจารย์ที่ปรึกษา

JUTATHIP PUTPONPAI : ALGORITHMS FOR ASYMMETRIC TRAVELLING SALESMAN PROBLEMS. ADVISOR : ASSISTANCE PROFESSOR DR. PISUT PONGCHAIRERKS, 71 PP.

This paper proposes a problem-based heuristic performing well for solving asymmetric travelling salesman problem (ATSP), one of a well-known NP-hard problem. The basic concept of the proposed heuristic is to construct a round-trip route by moving from the current city to one of its adjacent cities systematically using a well selection procedure in its move's direction. The algorithms proposed in this paper include deterministic, stochastic, and local search algorithms. Those algorithms show the result of the total distance and comparison for each algorithm.

Graduate School

Field of Engineering of Technology

Academic Year 2014

Student's Signature.....

Advisor's Signature.....

กิตติกรรมประกาศ

วิทยานิพนธ์นี้กระทำให้สำเร็จลงไปด้วยดี อันเนื่องมาจากความกรุณาเอาใจใส่เป็นอย่างดีจาก ผู้ช่วยศาสตราจารย์ ดร. พิศุทธิ์ พงศ์ชัยฤกษ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ อีกทั้งยังสละเวลาอันมีค่าของท่าน มาให้คำแนะนำและให้คำปรึกษาแนวทางและการพัฒนาในการทำวิจัย ตลอดจนสอนขั้นตอนการดำเนินวิจัยตลอดการทำงานวิจัยนี้

ขอขอบพระคุณ ดร. บุญพริกา เกษมสันติธรรม ที่กรุณาเสียสละเวลาให้คำปรึกษา ข้อเสนอแนะ และความรู้ในเรื่องต่างๆ

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร. วิภาวดี วงษ์สุวรรณ (สถาบันเทคโนโลยีไทย-ญี่ปุ่น) ที่กรุณาเสียสละเวลาเป็นคณะกรรมการสอบวิทยานิพนธ์ในครั้งนี้

ขอขอบพระคุณ ดร. กันติชา กิตติพิรชล (สถาบันเทคโนโลยีไทย-ญี่ปุ่น) ที่กรุณาเสียสละเวลาเป็นคณะกรรมการสอบวิทยานิพนธ์ในครั้งนี้

ข้าพเจ้าขอขอบพระคุณบิดา มารดา และครอบครัวของข้าพเจ้าที่ให้กำลังใจ และสนับสนุนเป็นแรงผลักดันให้ข้าพเจ้าได้ต่อสู้ อดทน และให้ข้าพเจ้าสามารถลุกขึ้นสู้กับปัญหาทุกอย่างไม่ท้อแท้ และข้าพเจ้าขอขอบพระคุณคุณปู่ของข้าพเจ้าที่เปรียบเสมือนบุคคลเตือนใจให้ข้าพเจ้ารู้ว่าท่านกำลังเฝ้ามองดูความสำเร็จของข้าพเจ้า และขอขอบคุณทุกท่านที่มีได้เอื้อนนามที่มีส่วนในการช่วยเหลือให้งานวิทยานิพนธ์นี้สำเร็จด้วยดี

จุฑาทิพย์ พุฒพันธ์

TNI

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ	ฉ
สารบัญ	ช
สารบัญตาราง	ฌ
สารบัญรูป	ญ
บทที่	
1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการศึกษา	2
1.3 ขอบเขตการวิจัย	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
1.5 สมมติฐานในงานวิจัย	3
1.6 แผนการดำเนินงาน	3
2 หลักการพื้นฐาน เอกสารและงานวิจัยที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง	4
3 ระเบียบวิธีวิจัย ผลการวิจัยและการวิเคราะห์ผลการทดลอง Adjacent Area Algorithm (AAA)	12
3.1 ศึกษากระบวนการปัจจุบัน	12
3.2 กำหนดขอบเขตการศึกษา และเก็บข้อมูล	13
3.3 ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง	14
3.4 สร้างโปรแกรมแบบจำลอง และทดสอบโปรแกรม	14
3.5 วิธี Adjacent Area Algorithm (AAA)	14
3.6 การทำงานของโปรแกรม	32

สารบัญ (ต่อ)

บทที่	หน้า
4 ระเบียบวิธีวิจัย ผลการวิจัยและการวิเคราะห์ผลการทดลอง Deterministic, Stochastic, and Local Search Algorithms	38
4.1 Deterministic Algorithm	38
4.2 Stochastic Algorithm	50
4.3 Local Search	54
4.4 วิเคราะห์ผล	62
5 สรุปผลการวิจัย	67
5.1 สรุปผลการวิจัย	67
5.2 ข้อเสนอแนะ	67
บรรณานุกรม	68
ประวัติย่อผู้วิจัย	71

TNI

THAI - NICHI INSTITUTE OF TECHNOLOGY

สารบัญตาราง

ตาราง	หน้า
1.1 แผนการดำเนินงาน	3
3.1 รูปแบบการเดินทาง AAA.....	18
3.2 รูปแบบการเดินทาง AAA และ NN	27
3.3 ระยะทางและเวลาการคำนวณภาคเหนือด้วยวิธีการ AAA และ NN.....	27
3.4 ระยะทางและเวลาการคำนวณภาคกลางตอนบนด้วยวิธีการ AAA และ NN.....	28
3.5 ระยะทางและเวลาการคำนวณภาคกลางตอนล่างด้วยวิธีการ AAA และ NN.....	28
3.6 ระยะทางและเวลาการคำนวณภาคตะวันออกเฉียงเหนือด้วยวิธีการ AAA และ NN.....	29
3.7 ระยะทางและเวลาการคำนวณภาคตะวันออกเฉียงเหนือ (ภาคอีสาน) ด้วยวิธีการ AAA และ NN.....	29
3.8 ระยะทางและเวลาการคำนวณภาคใต้และตะวันตกด้วยวิธีการ AAA และ NN.....	30
3.9 เปรียบเทียบค่าที่ดีที่สุดระหว่าง AAA และ ค่าคำตอบของวิธีการ NN เริ่มจากเหนือสุดของประเทศเทียบกับ LB.....	31
3.10 เปรียบเทียบค่าร้อยละความต่างระหว่างค่าที่ดีที่สุดของวิธี AAA และค่าคำตอบของวิธีการ NN เริ่มจากเหนือสุดของประเทศ.....	31
4.1 เปรียบเทียบระยะทางรวมของ DEA Algorithms กับ NN.....	49
4.2 เปอร์เซ็นต์การพัฒนาของ DEA Algorithms จากผลลัพธ์ของ NN.....	49
4.3 เปรียบเทียบเวลาเฉลี่ยของการรัน DEA Algorithms (หน่วย : วินาที).....	49
4.4 ระยะทางรวมที่ดีที่สุดของ STA Algorithm.....	52
4.5 CPU Time (หน่วย : วินาที) ของ STA Algorithm.....	53
4.6 เปรียบเทียบระยะทางรวมของ LSS และ LSI Algorithms.....	60
4.7 เปรียบเทียบ CPU Time (หน่วย : วินาที) LSS และ LSI Algorithms.....	60
4.8 เปรียบเทียบระยะทางที่ดีที่สุดของแต่ละวิธีการ (กิโลเมตร).....	61
4.9 เปรียบเทียบค่าร้อยละเฉลี่ยและความแตกต่างของระยะทางรวมระหว่าง NN และ LSI-1 (กิโลเมตร).....	61

สารบัญรูป

รูป	หน้า
2.1 ตัวอย่างของเส้นทางการเดินทางปัญหา TSP	5
2.2 ตัวอย่างของวิธีการเดินทางแบบ Nearest Neighbor Algorithm	7
2.3 หลักการการทำงานของ Swap.....	10
2.4 หลักการการทำงานของ Insert.....	11
3.1 แผนภาพแสดงกระบวนการทำวิจัย	12
3.2 แผนที่ประเทศไทยแบบ Original Image 0 ดีกรี	15
3.3 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 90 ดีกรี	15
3.4 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 180 ดีกรี.....	16
3.5 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 270 ดีกรี.....	16
3.6 แผนที่ประเทศไทยแบบ Mirror Image 0 ดีกรี.....	16
3.7 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 90 ดีกรี.....	17
3.8 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 180 ดีกรี.....	17
3.9 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 270 ดีกรี.....	17
3.10 ภาควัดวันออกของประเทศไทย Original Image 0 Degree	19
3.11 ตัวอย่างการเดินทางตามเข็มนาฬิกาผ่านเมืองที่เป็นขอบของภาควัดวันออก Original Image 0 Degree	20
3.12 ตัวอย่างการเดินทางด้วยวิธีการ AAA ภาควัดวันออก Original Image 0 Degree	21
3.13 Flowchart AAA.....	22
3.14 ภาควัดอีสาน 0 Degree Normal Image	23
3.15 ภาควัดเหนือ 0 Degree Normal Image	24
3.16 ภาควัดวันออก 0 Degree Normal Image.....	24
3.17 ภาควัดกลางตอนล่าง 0 Degree Normal Image.....	25
3.18 ภาควัดกลางตอนบน 0 Degree Normal Image.....	25
3.19 ภาควัดใต้กับภาควัดตะวันตก 0 Degree Normal Image.....	26

สารบัญรูป (ต่อ)

รูป	หน้า
3.20 หน้าจอหลักของโปรแกรม.....	32
3.21 อธิบายส่วนประกอบของหน้าจอหลัก.....	33
3.22 รูปแบบของการแสดงผลของ Eclipse.....	35
3.23 รูปแบบของการแสดงผล .txt File.....	35
3.24 ตัวอย่างรูปแบบของการเก็บข้อมูลระยะทาง.....	36
3.25 ตัวอย่างรูปแบบของการเก็บข้อมูลของทิศ 0 Degree Normal Image.....	36
3.26 ตัวอย่างรูปแบบของการเก็บข้อมูลของเมืองที่เป็นขอบของประเทศ.....	37
4.1 แผนที่ประเทศไทยแบบ Original Image 0 ดีกรี.....	38
4.2 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 90 ดีกรี.....	39
4.3 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 180 ดีกรี.....	39
4.4 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 270 ดีกรี.....	39
4.5 แผนที่ประเทศไทยแบบ Mirror Image 0 ดีกรี.....	40
4.6 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 90 ดีกรี.....	40
4.7 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 180 ดีกรี.....	40
4.8 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 270 ดีกรี.....	41
4.9 Flowchart DEA.....	43
4.10 ภาคเหนือ 0 Degree Normal Image.....	44
4.11 ภาคอีสาน 0 Degree Normal Image.....	44
4.12 ภาคตะวันออก 0 Degree Normal Image.....	45
4.13 ภาคกลาง 0 Degree Normal Image.....	46
4.14 ภาคใต้กับภาคตะวันตก 0 Degree Normal Image.....	47
4.15 ประเทศไทย 0 Degree Normal Image.....	48
4.16 Flowchart STA.....	51
4.17 หลักการการปรับปรุงเส้นทางแบบ SWAP.....	58

สารบัญรูป (ต่อ)

รูป	หน้า
4.18 หลักการการปรับปรุงเส้นทางแบบ INSERT	59
4.19 เส้นทางดีที่สุดของการเดินทางภาคเหนือแบบวิธี LSI-1	62
4.20 เส้นทางดีที่สุดของการเดินทางภาคตะวันออกเฉียงเหนือแบบวิธี LSI-1	63
4.21 เส้นทางดีที่สุดของการเดินทางภาคตะวันออกแบบวิธี LSI-1	63
4.22 เส้นทางดีที่สุดของการเดินทางภาคกลางแบบวิธี LSI-1	64
4.23 เส้นทางดีที่สุดของการเดินทางภาคใต้และตะวันตกแบบวิธี LSI-1	65
4.24 เส้นทางดีที่สุดของการเดินทางประเทศไทยแบบวิธี LSI-1	66

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัญหา Traveling Salesman Problem (TSP) เป็นรูปแบบปัญหาของพนักงานขายในการจัดการเส้นทางการเดินทางจากจุดเริ่มต้นไปยังทุกจุดของลูกค้า โดยมีเงื่อนไขให้สามารถผ่านแต่ละจุดได้เพียงครั้งเดียว, จุดเริ่มต้นและจุดสิ้นสุดคือจุดเดียวกัน นั่นหมายความว่า เมื่อเดินทางครบทุกจุดของลูกค้าแล้วจะต้องเดินทางกลับมาสู่จุดเริ่มต้นอีกครั้ง ลักษณะของคำตอบที่ได้อาจมีระยะทางหรือระยะเวลาในการเดินทางโดยรวมต่ำที่สุด โดยรูปแบบปัญหา TSP ยังสามารถนำมาประยุกต์ใช้กับสถานการณ์จริงได้ดีที่สุดปัญหาหนึ่ง อย่างไรก็ตาม วิธีการแก้ไขปัญหา TSP ที่ใช้กันอยู่ในปัจจุบัน เช่น Nearest Neighbor Heuristics (NN Heuristics) เป็นเทคนิคในการหาจุดที่ใกล้จุดเริ่มต้นที่สุดและยังไม่เคยไปมาก่อน เป็นจุดต่อไปของการเดินทาง แต่เส้นทางการเดินทางที่ได้จาก NN Heuristics อาจไม่รับประกันว่าคำตอบที่ดีที่สุด เนื่องจาก NN Heuristics สนใจจุดต่อไปที่สั้นที่สุด การเดินทางกลับมาสู่จุดเริ่มต้นอีกครั้งอาจจะต้องใช้ระยะทางที่มาก ไม่ได้สนใจทิศทางการเดินทางและไม่ได้นำข้อมูลของตำแหน่งในแผนที่มาทำให้เกิดประโยชน์ อาทิเช่น จังหวัดใดติดกับบริเวณชายแดนหรือติดกับทะเล จังหวัดใดที่อยู่ติดกัน เป็นต้น ข้อมูลเหล่านี้เป็นประโยชน์อย่างยิ่งในการหาเส้นทางโดยสารหรือเส้นทางขนส่งที่ประหยัดระยะทาง

นอกจากนี้ จากการสำรวจพบว่า ความเป็นจริงนั้น เส้นทางขาไปกับเส้นทางขากลับระหว่างเมือง 2 เมืองมักจะไม่เท่ากัน เช่น จากวัดพระศรีรัตนศาสดาราม จังหวัดกรุงเทพมหานครไปยังวัดเทพโมฬี จังหวัดกำแพงเพชร 358 กิโลเมตร ในทางกลับกัน จากวัดเทพโมฬี จังหวัดกำแพงเพชรไปยังวัดพระศรีรัตนศาสดาราม จังหวัดกรุงเทพมหานคร 356 กิโลเมตร สืบเนื่องจากการเพิ่มประสิทธิภาพในการเลือกใช้เส้นทางถนนในระดับการกำหนดเส้นทาง เช่น ถนนทางเดียว, มอเตอร์เวย์, ตามระบบหมายเลขทางหลวง, การกลับรถ ดังนั้น ปัญหาระยะทางไป-กลับระหว่าง 2 เมือง ไม่เท่ากันเป็นรูปแบบปัญหา TSP แบบไม่สมมาตร ที่เรียกว่า Asymmetric Traveling Salesman Problem (ATSP). ATSP เป็นรูปแบบปัญหาทั่วไปของ TSP แบบ Combinatorial Optimization ที่อยู่ในกลุ่มของปัญหาที่มีความซับซ้อนในระดับ Non-Deterministic Polynomial Time Hard หรือที่เรียกว่า NP-Hard ซึ่งไม่สามารถหาคำตอบที่เหมาะสมที่สุดภายในระยะเวลาการคำนวณที่เหมาะสมได้ โดยเฉพาะเมื่อปัญหามีขนาดใหญ่ ในหลายๆ งานวิจัยจะให้ความสนใจ ภายในระยะเวลาการคำนวณที่สั้นและจำกัด ซึ่งแต่ละวิธีการอาจมีความเหมาะสมหรือใช้งานได้ดีในบางปัญหา และอาจมีประสิทธิภาพลดลงหากนำไปใช้กับปัญหาอื่นหรือมีเงื่อนไขแตกต่างกัน

โดยในงานวิจัยฉบับนี้จะมุ่งเน้นในการสร้างเส้นทางการเดินทางด้วยวิธี Problem Based Heuristics ซึ่งจะเน้นหาวิธีการเดินทางที่ง่าย และได้คำตอบที่ดีที่สุดที่รับได้ภายใต้เวลาที่เหมาะสมโดยเอาข้อมูลของตำแหน่งในแผนที่เข้ามาเพื่อควบคุมทิศทางการหาเส้นทางการเดินทาง ซึ่งจะลดการเคลื่อนที่แบบไม่แน่นอน สลับไปมาและเพื่อให้ผลลัพธ์ของงานวิจัยมีความน่าเชื่อถือ งานวิจัยนี้จึงใช้กรณีศึกษาจริง ข้อมูลจริง โดยเลือกใช้ประเทศไทยซึ่งมีลักษณะข้อมูลของระยะทางในรูปแบบ ATSP งานวิจัยนี้มุ่งเน้นการแก้โจทย์ปัญหาแบบ Large-Scale Solution ซึ่งเป็นการหาวิธีการแบบ Problem Based Heuristics โดยเริ่มจากเมืองสูงสุดของประเทศตามแผนที่ภูมิศาสตร์และใช้วัดสำคัญเป็นตัวแทนของแต่ละเมือง ซึ่งการเลือกใช้วัดสำคัญเป็นเพียงแค่สถานที่ตัวแทนของแต่ละเมืองเพื่อกำหนดตำแหน่งในการสร้างเส้นทางการเดินทางจากเมืองหนึ่งไปสู่อีกจังหวัดหนึ่ง เนื่องจากวัดสำคัญของแต่ละเมืองเป็นสถานที่ที่คนทั่วไปรู้จักโดยเฉพาะคนในท้องถิ่น และยังมีเส้นทางถนนที่สามารถนำมาใช้เป็นข้อมูลระยะทางจริงได้

1.2 วัตถุประสงค์ของการศึกษา

เพื่อศึกษาการจัดลำดับเส้นทางการเดินทางจากเมืองหนึ่งไปยังอีกเมืองหนึ่ง ให้มีระยะทางรวมที่สั้นที่สุด ภายใต้การใช้ประโยชน์จากเมืองที่เป็นขอบของประเทศ และเมืองที่มีอาณาเขตติดต่อกัน โดยใช้ทิศเป็นตัวระบุ

1.3 ขอบเขตการวิจัย

1.3.1 จำนวนเมือง 77 เมือง

1.3.2 ทิศ 8 (เหนือ, ตะวันออกเฉียงเหนือ, ตะวันออก, ตะวันออกเฉียงใต้, ใต้, ตะวันตกเฉียงใต้, ตะวันตก, และ ตะวันตกเฉียงเหนือ)

1.3.3 มีการมองภาพแผนที่ 2 รูปแบบ

1.3.3.1 มองรูปภาพปกติ (Normal View)

1.3.3.2 มองรูปภาพผ่านกระจก (Mirror View)

1.3.4 การหมุนภาพแผนที่

1.3.4.1 0 องศา

1.3.4.2 90 องศา

1.3.4.3 180 องศา

1.3.4.4 270 องศา

1.3.5 มีเมืองเริ่มต้น 4 เมือง คือ เมืองที่มีขอบเขตสูงสุดจากการหมุนองศาทวนเข็มนาฬิกา

1.5 สมมติฐานในงานวิจัย

- #### 1.4.2 ระยะทางรวมลดลง

1.5 สมมติฐานในงานวิจัย

- #### 1.5.4 การคิดเส้นทางเลือกตามความสำคัญของทิศที่กำหนด

1.6 แผนการดำเนินงาน

ประกอบด้วยขั้นตอนการศึกษาปัญหา วิเคราะห์ หาแนวทางและศึกษางานวิจัย จากนั้น
สร้างแบบจำลอง ทดลองโปรแกรมและสรุปผล และจัดทำบทความวิชาการและรูปเล่มงานวิจัย

ตารางที่ 1.1 แผนการดำเนินงาน

[illegible]

บทที่ 2

หลักการพื้นฐาน เอกสารและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการพื้นฐาน เอกสารงานวิจัย และทฤษฎีที่เกี่ยวข้องกับการแก้ปัญหาด้านการขนส่งในงานวิจัยนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ปัญหาการจัดการการเดินทาง

2.1.1.1 ปัญหาการจัดการการเดินทาง

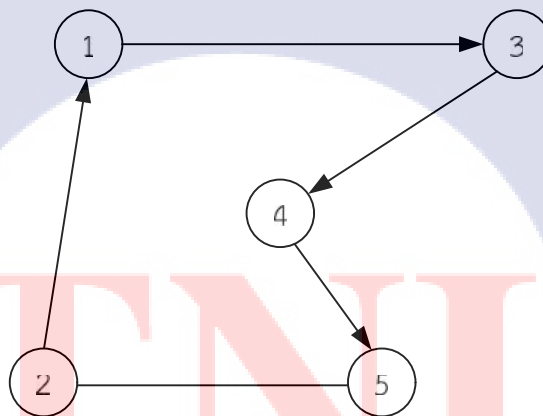
เป็นปัญหาที่เราต้องการคำตอบของการตัดสินใจที่ดีที่สุด ซึ่งอาจหมายถึงการมีระยะทางการเดินทางรวมสั้นที่สุด การมีค่าใช้จ่ายในการเดินทางน้อยที่สุด การมีเวลาการเดินทางสั้นที่สุด การจำแนกปัญหาการตัดสินใจในเส้นทางสำหรับการเดินทาง คือ ปัญหาการตัดสินใจที่ต้องการคำตอบเพื่อใช้ในการทำงาน ซึ่งอาจจะเป็นปัญหาการตัดสินใจที่ต้องการคำตอบที่ถูกต้องและแม่นยำอย่างมาก

2.1.1.2 ปัญหาการจัดเส้นทางสำหรับยานพาหนะ

ปัญหา Traveling Salesman Problem (TSP) เป็นรูปแบบปัญหาทางคณิตศาสตร์ที่ถูกพัฒนามาตั้งแต่ปี 1800 โดย Irish Mathematician Sir William Rowan Hamilton และ British Mathematician Thomas Penyngton Kirkman. ในปี 1930 ได้มีการศึกษา The General form of the TSP เป็นครั้งแรกโดย Karl Menger in Vienna and Harvard. โดยปัญหา TSP จัดอยู่ในรูปแบบของ Non-Deterministic Polynomial Time Hard หรือที่เรียกว่า NP-Hard [1] ซึ่งไม่สามารถหาคำตอบที่เหมาะสมที่สุดในระยะเวลาการคำนวณที่เหมาะสมได้. โดยเฉพาะเมื่อปัญหามีขนาดใหญ่ อาทิเช่น ถ้าจะหาเส้นทางการเดินทางที่เป็นไปได้ทั้งหมดของ 10 เมือง ซึ่งต้องอาศัยการคิดคำนวณคำตอบที่เป็นไปได้ 3,628,800 ครั้ง. โดยรูปแบบปัญหาของ TSP มีอยู่ 2 แบบ คือ Symmetric TSP คือมีระยะทางทั้งขาไปและกลับเท่ากับ ซึ่งสามารถเขียนให้อยู่ในรูปแบบของสมการทางคณิตศาสตร์ $d(i,j) = d(j,i)$ และ Asymmetric TSP หรือที่เรียกว่า ATSP โดยมีรูปแบบสมการทางคณิตศาสตร์ $d(i,j) \neq d(j,i)$. ในช่วง 50 ปีที่ผ่านมา ได้มีนักวิจัยคิดค้นและนำเสนอวิธีการในการจัดการปัญหา TSP และได้มีการกำหนดเป้าหมายด้วยวิธีการหาคำตอบต่างๆ อาทิเช่น Exact Method, Problem Based Heuristics, และ Meta-Heuristic ในการจัดเส้นทางการเดินทางตามจำนวนเมืองที่ได้กำหนดไว้ อาทิเช่น ในปี 1954 49-Cities Problem โดย Dantzig, Fulkerson, and Johnson, ในปี 1971 64-Cities Problem โดย Held and Karp, ในปี 1975 67-Cities โดย Camerini,

Fratta, and Maffioli, ในปี 1980 318 Cities โดย Crowder and Padberg, ในปี 2004 24,978-Cities Problem โดย Applegate, Bixby, Chvatal, Cook, and Helsgaun.

โดยปัญหา TSP เป็นปัญหาการเดินทางของพนักงานขาย เป็นหนึ่งในปัญหาที่เป็นปัญหาที่ได้รับความสนใจจากนักวิจัยอย่างต่อเนื่องเป็นระยะเวลายาวนานในการพัฒนาวิธีการในการหาคำตอบให้ดีและเร็ว โดยปัญหา TSP นี้ เป็นปัญหาที่ทำการตัดสินใจหาเส้นทางการเดินทางเมื่อมีเมืองหรือสถานที่ที่ต้องเดินทางไปจำนวน N เมืองหรือ N สถานที่ การเดินทางจะเดินทางจากเมืองใดเมืองหนึ่งในจำนวน N เมือง โดยเส้นทางการเดินทางนั้นๆ จะต้องเดินทางผ่านเมืองทุกเมืองใน N และกลับมาที่เมืองที่ทำการเริ่มต้นในการเดินเหมือนการเดินทางวนรอบ โดยมีเงื่อนไขว่าห้ามกลับไปยังเมืองเดิมที่เคยไป ต้องมีระยะทางรวมของการเดินทางน้อยที่สุด หรือ อาจต้องมีเวลา รวมของการเดินทางน้อยที่สุด ซึ่งปัญหา TSP หากจำนวนเมืองที่น้อย การค้นพบเส้นทางที่สั้นที่สุด อาจใช้ระยะเวลาหรือการหาคำตอบที่ใช้เวลาไม่นาน แต่ถ้าหากปัญหาเป็นระดับใหญ่ มีจำนวนเมืองที่ต้องเดินทางเป็นจำนวนมาก โอกาสที่จะได้เส้นทางการเดินทางที่ดีที่สุดอาจจะใช้ระยะเวลานาน เช่น พนักงานเดินขายจำเป็นต้องเดินส่งสินค้าเป็นจำนวน 5 เมือง โดยเริ่มต้นจากเมืองที่ 1 จะต้องเดินไปยังครบทุกเมืองโดยที่ไม่ซ้ำเมืองเดิมแล้วกลับมายังเมื่อเริ่มต้น โดยใช้ระยะทางที่สั้นที่สุด หรือ ใช้ระยะเวลาน้อยที่สุด ซึ่งพนักงานเดินขายของอาจเลือกเดิน 1-3-4-5-2-1



รูปที่ 2.1 ตัวอย่างของเส้นทางการเดินทางปัญหา TSP

เทคนิคในการหาคำตอบของปัญหาการจัดเส้นทางเดินรถ TSP สามารถแบ่งได้เป็น 3 รูปแบบ ดังนี้

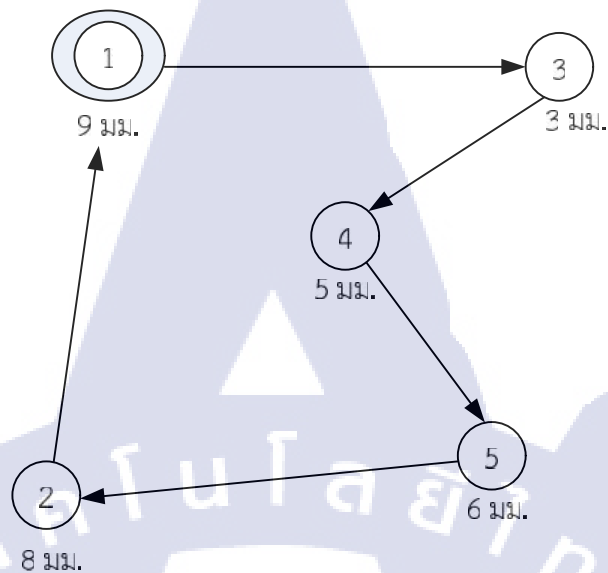
1. Exact method คือรูปแบบวิธีการหาคำตอบของปัญหาที่ได้คำตอบที่ดีที่สุดแน่นอน เช่นวิธี Brute Force [2] ซึ่งเป็นการหาทุกคำตอบที่เป็นไปได้ของปัญหาและเลือกผลลัพธ์ที่ดีที่สุด แต่วิธีการแบบนี้ไม่เหมาะกับปัญหา TSP ที่มีขนาดใหญ่ เนื่องจากจะต้องใช้เวลาในการคำนวณที่นานเพื่อให้ได้คำตอบทั้งหมด หรือวิธีการที่ไม่ขัดแย้งกับทุกเงื่อนไขที่กำหนด แต่ใช้เวลานานเมื่อมีเงื่อนไขมาก หรือมีตัวแปรที่เพิ่มขึ้น ซึ่งมีวิธีที่นิยมใช้คือแบบ Branch and Bound [3] หลักการของการแก้ปัญหา คือ เริ่มต้นด้วยการหาเส้นทางขึ้นมาเส้นทางหนึ่ง แล้วพยายามหาเส้นทางอื่นขึ้นมาเปรียบเทียบกับ ถ้าเส้นทางใหม่ที่ได้ดีกว่า ให้นำเส้นทางใหม่ที่ได้เป็นเส้นทางหลักในการเปรียบเทียบกับเส้นทางอื่นต่อไป แต่ถ้าเส้นทางใหม่ที่สร้างขึ้นเมื่อวัดแล้ว แม้นเพียงบางส่วนที่ยาวกว่าเส้นทางหลัก ก็ให้ยกเลิกเส้นทางนั้นทำเช่นนี้เรื่อยๆ จนกระทั่งได้เส้นทางที่ดีที่สุด

2. Problem Based Heuristics คือรูปแบบการหาคำตอบของปัญหา TSP ที่นิยมใช้ใน กลุ่ม NP-Hard โดยคำตอบที่ได้อาจจะไม่ใช่คำตอบที่ดีที่สุด แต่น่าจะใกล้เคียงกับคำตอบที่ดีที่สุดและเหมาะสมในระดับหนึ่งในเวลาที่รับได้ซึ่ง Heuristics Algorithm สามารถแบ่งวิธีการวิเคราะห์เพื่อหาคำตอบได้ 3 วิธี ดังนี้

2.1 Tour Construction Procedure คือวิธีการสร้างวิธีการเดินทางขึ้นอยู่กับวิธีการเดินและเงื่อนไขในการเดินทาง โดยวิธีการที่รู้จักกันคืออย่าง

2.1.1 Nearest Neighbor (NN) and Repetitive Nearest Neighbor Algorithms [4] โดยหลักการของวิธี Nearest Neighbor (NN) คือเลือกไปทางเดินที่สั้นที่สุดก่อน ทำไปแบบนี้เรื่อยๆ จนครบทุกเมืองจากนั้นเดินกลับมาถึงเมืองเริ่มต้น ซึ่งวิธีนี้เป็นวิธีที่นิยมใช้มากสำหรับการแก้ปัญหาของพนักงานเดินขายของ Traveling Salesman Problem

TNI



รูปที่ 2.2 ตัวอย่างของวิธีการเดินทางแบบ Nearest Neighbor Algorithm

จากรูปที่ 2.2 จะได้ผลลัพธ์เป็นเส้นทาง 1-3-4-5-2-1 รวมเป็นระยะทาง 31 กิโลเมตร โดยวิธีการที่ประยุกต์ใช้วิธีการของ Nearest Neighbor Algorithm เรียกว่า Repetitive Nearest Neighbor Algorithm เป็นวิธีการทำซ้ำ ซึ่งสามารถเลือกจุดเริ่มต้นโดยการแปลงผลลัพธ์ที่ได้จาก Nearest Neighbor Algorithm โดยระยะทางรวมที่ได้จะมีค่าเท่าเดิม เช่น 3-4-5-2-1-3, 4-5-2-1-3-4, 5-2-1-3-4-5, 2-1-3-4-5-2 ซึ่งเส้นทางที่กล่าวมาข้างต้นนี้ก็จะมียุทธศาสตร์รวม 31 กิโลเมตรเช่นเดียวกับ 1-3-4-5-2-1

2.1.2 Greedy Heuristic for TSP [5] หรือที่เรียกว่า ขั้นตอนวิธีประเภทละโมภ โดยวิธีนี้มีหลักการอยู่ว่า ในขณะใดๆ ก็ตาม เราจะหาคำตอบที่ดีที่สุดของปัญหา โดยการเลือกเอา คำตอบที่ดีที่สุด ณ สถานะนั้นออกมาก่อน

2.1.3 Insertion Heuristic [6]

2.2 Tour Improvement Procedure คือการนำเส้นทางเดินรถที่มีอยู่แล้วมาปรับปรุงให้ได้เส้นทางเดินรถใหม่ที่มีคำตอบที่ดีกว่าแบบเดิมโดยทำการสลับเปลี่ยนเส้นทางให้ดีขึ้น อาทิเช่น 2-opt and 3-opt [7] Speeding Up 2-opt and 3-opt, Lin-Kernighan [8]

2.3 Composite Procedure เป็นวิธีการที่รวมวิธี Tour Construction Procedure ในการสร้างเส้นทางเดินรถ และใช้วิธี Tour Improvement Procedure ในการปรับปรุงเส้นทาง

3. Meta-Heuristic เป็นวิธีการที่พัฒนาเพิ่มเติมจากวิธี Problem Based Heuristics แต่มีความซับซ้อนมากกว่า จึงทำให้ได้คำตอบที่ดีกว่า มีระเบียบวิธีในการค้นหาคำตอบที่ดีภายในพื้นที่ของคำตอบที่เป็นไปได้ อาทิเช่นวิธี

3.1 Genetic Algorithm [9] หรือที่เรียกว่าขั้นตอนวิธีเชิงพันธุกรรม เป็นเทคนิคที่มีความสามารถในการแก้ไขปัญหารูปแบบ TSP ที่เพิ่มประสิทธิภาพและการเรียนรู้ของวิวัฒนาการธรรมชาติ โดยเป็นหลักการของโครโมโซม

3.2 Tabu Search [10] หรือที่เรียกว่าการค้นหาคำตอบแบบทาบู เป็นวิธีการค้นหาคำตอบแบบเมตา-ฮิวริสติก (Meta-Heuristics) ซึ่งเป็นวิธีที่นิยมมากในการนำมาแก้ปัญหาที่ไม่เป็นโพลิเมียล (NP-Problem) เช่นปัญหาการกำหนดเส้นทางการเดินทาง ซึ่งค้นหาคำตอบจากบริเวณข้างเคียงคำตอบที่ดีที่สุด ในขณะที่ โดยจะมีข้อห้ามไม่ให้ค้นหาซ้ำกับเซตของคำตอบเดิมที่มีอยู่

3.3 Variable Neighborhood Search (VNS) ถูกสร้างขึ้นโดย Pierre Hansen และ Nenad Mladenovic ในปี 1999 ซึ่งหลักการตั้งในพัฒนาการทำ Local Search ให้มีหลักการที่แน่นอน และเพิ่มประสิทธิภาพการทำงานของ Algorithm หลัก สำหรับปัญหาที่มีขนาดใหญ่ ซึ่ง VNS ซึ่งสร้างมาเพื่อเป็นตัว Local Search ซึ่งขั้นตอนการทำงานจะง่าย ไม่ซับซ้อน ซึ่งมีขั้นตอนการทำงานดังนี้ คือ

3.3.1 กำหนดค่าของ จำนวนรอบการทำงาน ใน 1 Iteration ซึ่งจะเท่ากับจำนวนรอบการทำงานใน 1 Iteration ซึ่งผ่านการเรียงผลเฉลยแล้ว

3.3.2 การ Shaking จะเป็นการสุ่ม เพื่อจะกำหนดจุดที่จะทำการเปลี่ยนแปลง

3.3.3 การ Neighborhood Search จะสลับตำแหน่งรอบตัวของตัวที่ Shaking

3.3.4 นำค่าที่ได้ เปรียบเทียบคำตอบกับผลเฉลยตัวเดิมก่อนการ Shaking ถ้าดีกว่าจะยอมให้มีการสลับตำแหน่ง

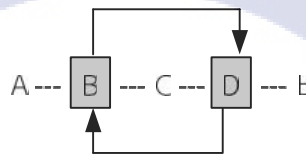
โดยในวิธีการ VNS นั้นมีนักวิจัยได้ทำการศึกษาและแก้ปัญหาวิธีการเดินทางด้วยวิธีการนี้เช่น การแก้ปัญหาดังกล่าวการใช้ VNS ผสมผสานกับ Nearest Neighbor ในการได้คำตอบ Initial Solution จากนั้นใช้วิธีการ VNS ในการปรับปรุงผลลัพธ์ของ Initial Solution [11]

3.4 Ant Colony Optimization [12] ซึ่งเลียนแบบพฤติกรรมหาอาหารของมด โดยค้นหารหาเส้นทางที่สั้นที่สุดระหว่างรังกับแหล่งอาหารโดยใช้ฟีโรโมน (Pheromone) ที่มดวางไว้ระหว่างทางเพื่อใช้ในการสื่อสารทางอ้อมกับมดตัวอื่นในฝูง

จากการทบทวนงานวรรณกรรม นักวิจัยส่วนใหญ่จะนำวิธีการที่มีอยู่แล้วมาประยุกต์ใช้ในการแก้ปัญหาดังกล่าว แต่อย่างไรก็ตาม วิธีการต่างๆ อาจจะมีประสิทธิภาพที่ดีในปัญหาที่แตกต่างกันไป แต่ในทางกลับกัน นักวิจัยกลับมองข้ามที่จะนำสิ่งที่ใกล้ตัวที่มีอยู่แล้ว อย่างเช่นข้อมูลของแผนที่ อาทิเช่น จังหวัดใดมีอาณาเขตเป็นขอบของประเทศ ซึ่งหมายถึงจังหวัดที่มีอาณาเขตติดกับ

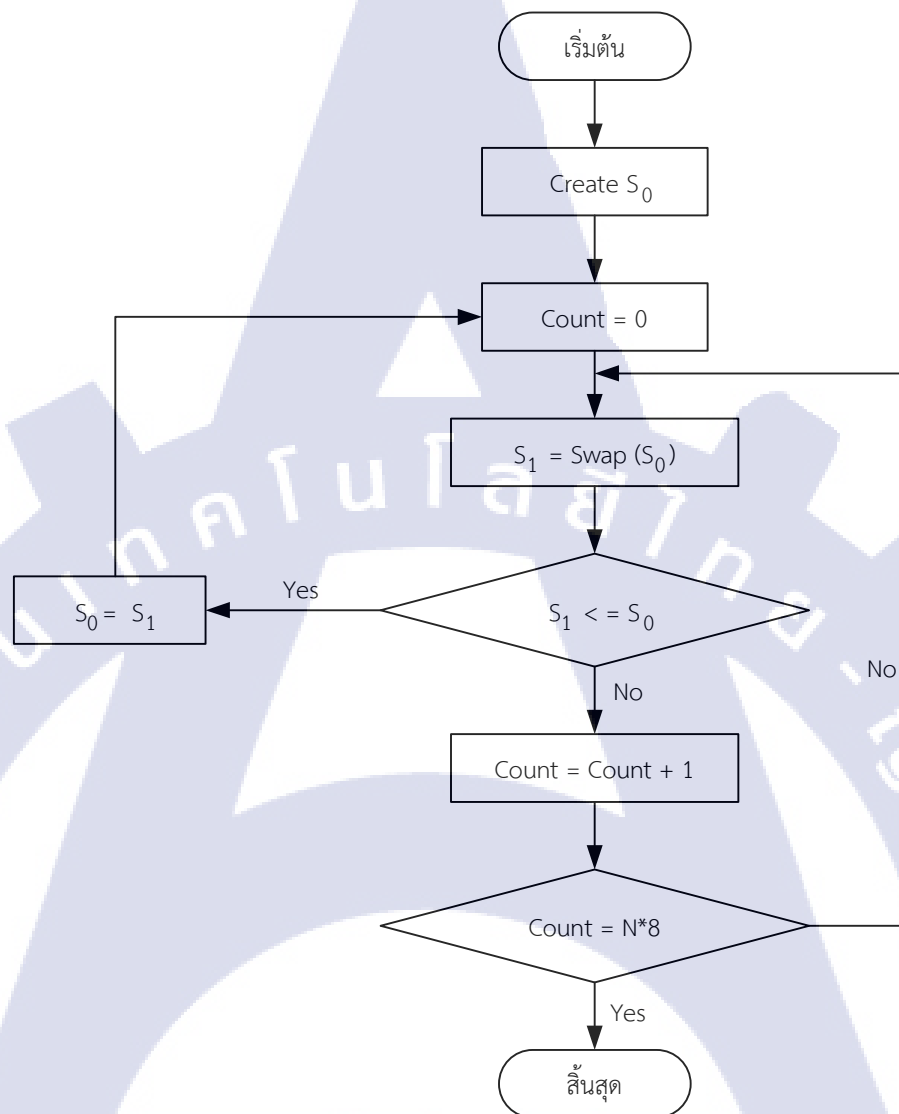
ประเทศเพื่อนบ้าน หรือทะเล และจังหวัดใดที่ติดกัน มาใช้ให้เกิดประโยชน์ ถึงแม้การแก้ปัญหาจะมีขนาดใหญ่มาก เพราะฉะนั้นงานวิจัยฉบับนี้จึงมุ่งเน้นในการหาเส้นทางการเดินทางด้วยวิธี Problem Based Heuristics ซึ่งจะเน้นหาวิธีการเดินทางที่ง่าย โดยมีแรงบันดาลใจจากการพบเห็นในชีวิตประจำวันในการกลับบ้าน ที่จะเริ่มต้นเส้นทางจากมุมห้องมุมหนึ่งแล้วไล่จนสุดอีกมุมหนึ่งแล้วไล่กลับขึ้นไป ซึ่งในงานวิจัยจะมีวิธีการทั้งหมด 3 Algorithms ได้แก่ Deterministic Algorithm เป็นวิธีการที่เริ่มต้นจากเมืองที่สูงสุดของประเทศตามแผนที่รูปภาพเดินทางตามเข็มนาฬิกาผ่านเมืองที่เป็นขอบของประเทศจนถึงเมืองที่ต่ำสุดของประเทศตามแผนที่รูปภาพ จากนั้นเดินไปยังเมืองที่ไม่เคยไปตามลำดับความสำคัญของทิศ ได้แก่-ตะวันออกเฉียงใต้-ตะวันออก-ตะวันออกเฉียงเหนือ-ตะวันตกเฉียงใต้-ตะวันตก-ตะวันตกเฉียงเหนือ-เหนือ [13] ซึ่งในงานวิจัยจะมีการปรับปรุงและทดลองลำดับความสำคัญของทิศทั้งหมด 6 รูปแบบ ในส่วนของวิธีการที่ 2 คือ Stochastic Algorithm เป็นวิธีการที่คล้ายคลึงกับวิธี Deterministic Algorithm แต่จะต่างตรงส่วนของการเดินไปยังลำดับความสำคัญของทิศจะใช้ความน่าจะเป็น โดยกำหนดค่าเปอร์เซ็นต์ในโอกาสที่จะเลือกใช้รูปแบบความสำคัญของทิศ และวิธีที่ 3 คือ Local Search Algorithms เป็นวิธีการปรับปรุงคำตอบเฉพาะที่ ในวิธีการของ Local Search ได้เลือกใช้แบบ Swap (การสลับที่) และ Insert (การแทรกที่)

รูปแบบ Swap เป็นวิธีการหนึ่งในการปรับปรุงคำตอบเฉพาะที่ซึ่งเป็นการสับเปลี่ยนตำแหน่ง 2 ตำแหน่ง โดยมีหลักการการทำงานดังรูปที่ 3

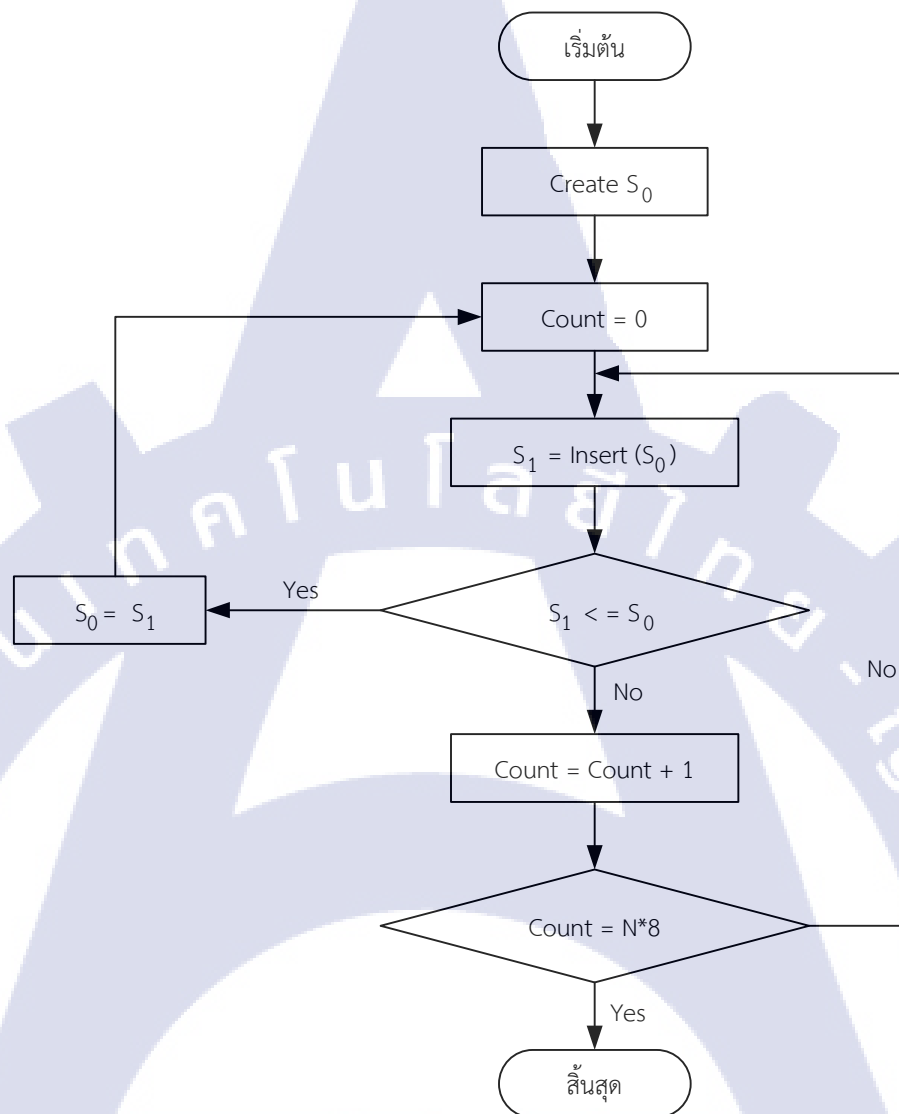


รูปแบบการ Insert ดำเนินการโดยเลือกจุดที่จะนำไปแทรก (ขั้นตอนเลือก) และตัดสินใจนำไปแทรกที่ตำแหน่งข้างหน้าตัวที่ต้องการจะแทรก โดยมีหลักการการทำงานดังรูปที่ 4





รูปที่ 2.3 หลักการทำงานของ Swap



รูปที่ 2.4 หลักการทำงานของ Insert

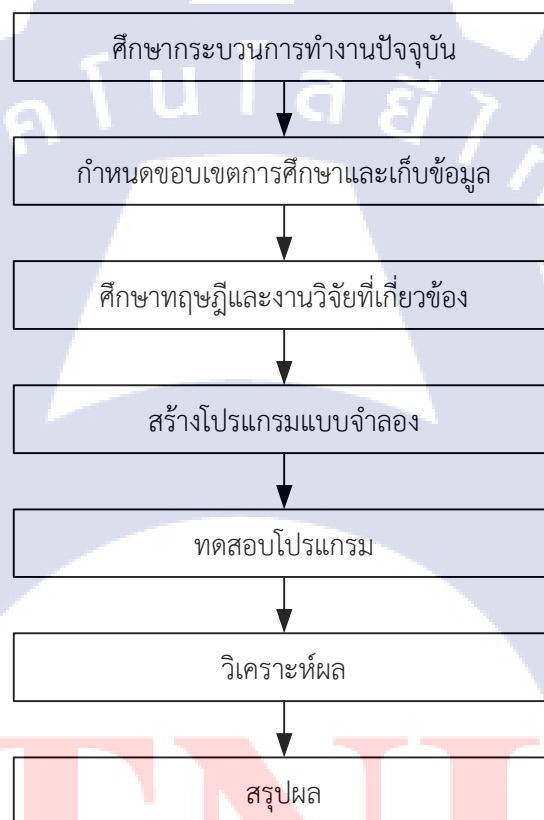
บทที่ 3

ระเบียบวิธีวิจัย ผลการวิจัยและการวิเคราะห์ผลการทดลอง

Adjacent Area Algorithm (AAA)

การดำเนินงานวิจัยจะเริ่มจากศึกษากระบวนการทำงานปัจจุบัน โดยมีขั้นตอนต่างๆ ดัง

รูปที่ 3.1



รูปที่ 3.1 แผนภาพแสดงกระบวนการทำวิจัย

3.1 ศึกษากระบวนการปัจจุบัน

3.1.1 การคำนวณเส้นทางรถที่ใช้ในปัจจุบัน

อ้างอิงจาก Google Map แผนที่เส้นทางเดินรถในปัจจุบัน โดยใช้จุดสำคัญของแต่ละเมืองเป็นจุดอ้างอิงในการหาระยะทาง หากระยะทางจากเมือง A ไป เมือง B มีระยะทางมากกว่า 1

เส้นทางให้เลือกระยะทางที่สั้นที่สุดในการเก็บข้อมูล โดยข้อมูลระยะทางที่ได้จะจัดเก็บอยู่ในรูปแบบของ Excel File

3.1.2 ศึกษาและเก็บรวบรวมปัญหา

จากการศึกษาวิธีการหาคำตอบเส้นทางของปัญหา ATSP โดยมากจะใช้วิธีการ NN ซึ่งมีหลักการเพียงแค่จากเมืองเริ่มต้นไปยังเมืองถัดไปที่ใกล้ที่สุด ทำแบบนี้ไปเรื่อยๆ จนครบทุกเมืองโดยที่ไม่มีไปซ้ำเมืองเดิมแล้วกลับมายังเมืองเริ่มต้นพบว่าการเดินทางจากเมืองสุดท้ายกลับไปยังเมืองเริ่มต้นนั้นมีระยะทางที่ห่างกันมาก ด้วยสาเหตุที่วิธีการเดินทางแบบ NN ไม่ได้คำนึงถึงการใช้ประโยชน์จากข้อมูลของแผนที่ทั้งในด้านอาณาเขตติดต่อกัน ทิศของเมือง

3.2 กำหนดขอบเขตการศึกษา และเก็บข้อมูล

3.2.1 จำนวนเมือง 77 เมือง โดยแบ่งข้อมูลการทดลองทั้งหมด 7 รูปแบบดังนี้

3.2.1.1 ประเทศไทย 77 เมือง

3.2.1.2 ภาคเหนือของประเทศไทย 9 เมือง

3.2.1.3 ภาคกลาง 22 เมือง

3.2.1.4 ภาคอีสาน 20 เมือง

3.2.1.5 ภาคใต้และภาคตะวันตก 19 เมือง

3.2.1.6 ภาคตะวันออก 7 เมือง

3.2.2 ทิศ 8 (เหนือ, ตะวันออกเฉียงเหนือ, ตะวันออก, ตะวันออกเฉียงใต้, ใต้, ตะวันตกเฉียงใต้, ตะวันตก, และ ตะวันตกเฉียงเหนือ)

3.2.3 มีการมองภาพแผนที่ 2 รูปแบบ

3.2.3.1 มองรูปภาพปกติ (Normal Image)

3.2.3.2 มองรูปภาพผ่านกระจก (Mirror Image)

3.2.4 การหมุนภาพแผนที่

3.2.4.1 0 องศา

3.2.4.2 90 องศา

3.2.4.3 180 องศา

3.2.4.4 270 องศา

3.2.5 มีเมืองเริ่มต้น 4 เมือง คือ เมืองที่มีขอบเขตสูงสุดตามแผนที่รูปภาพจากการหมุนองศาทวนเข็มนาฬิกา

3.3 ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง

จากลักษณะปัญหาที่ได้กล่าว เมื่อนำมาศึกษาร่วมกับทฤษฎี พบว่าเป็นรูปแบบปัญหา ATSP โดยใช้วิธีการ NN นั้นพบว่าการเดินทางจากเมืองสุดท้ายกลับไปยังเมืองเริ่มต้นนั้นมีระยะทางที่ห่างกันมาก ด้วยสาเหตุที่วิธีการเดินทางแบบ NN ไม่ได้คำนึงถึงการใช้ประโยชน์จากข้อมูลของแผนที่ทั้งในด้านอาณาเขตติดต่อกัน ทิศของเมือง งานวิจัยจึงได้นำข้อเสียนี้นี้มาปรับปรุงวิธีการใหม่ที่คำนึงถึง เมืองที่มีอาณาเขตติดต่อกันในแต่ละทิศ โดยมีการกำหนดทิศทั้ง 8 ได้แก่ เหนือ ตะวันออกเฉียงเหนือ ตะวันออก ตะวันออกเฉียงใต้ ตะวันตก ตะวันตกเฉียงใต้ ตะวันตก ตะวันตกเฉียงเหนือ และมีการระบุเมืองที่เป็นขอบของประเทศ โดยดูจากแผนที่ว่าเมืองใดที่มีอาณาเขตติดต่อกับทะเล หรือประเทศเพื่อนบ้าน ถือว่าเป็น เมืองที่เป็นขอบของประเทศ

3.4 สร้างโปรแกรมแบบจำลอง และทดสอบโปรแกรม

ในงานวิจัยนี้ ดำเนินการสร้างการคำนวณผ่านโปรแกรม Eclipse โดยนำเข้าข้อมูลระยะทาง ทิศการติดต่อในแต่ละเมือง เมืองที่เป็นขอบของประเทศ ซึ่งเป็น Excel File โดยโปรแกรมจะอ่านข้อมูลจาก Excel File จากนั้นผลลัพธ์ที่แสดง คือ ระยะทางรวม เส้นทางการเดินทาง และเวลาการประมวลผลผลลัพธ์ โดยข้อมูลระยะทางในแต่ละกลุ่มการทดลองจัดเก็บอยู่ในแผ่นซีดีงานวิจัย

3.4.1 เครื่องมือที่ใช้

- ฮาร์ดแวร์ : PC Computer (Windows 7, 2.67 GHz Intel ® Core™2 Quad CPU Processor, 4.00 GB of Ram)
- ซอฟต์แวร์ : Eclipse Application, Microsoft Excel

3.4.2 การวิเคราะห์

เปรียบเทียบค่าเฉลี่ยจากการคำนวณทั้ง 3 วิธีการทดลอง Deterministic, Stochastic, and Local Search Algorithms.

3.5 วิธี Adjacent Area Algorithm (AAA)

วิธีการ Adjacent Area Algorithm (AAA) ในงานวิจัยเป็นวิธีการที่ได้รับการตีพิมพ์ในงานประชุม ICIPE โดย Putponpai and Pongchairerks [13] โดยหลักการของวิธีการ AAA คือ AAA ต้องเลือกทิศทาง (0, 90, 180, 270) และ การมองภาพแบบ มองรูปภาพปกติ (Normal Image) หรือ มองรูปภาพผ่านกระจก (Mirror Image) และลำดับความสำคัญของทิศจะคงที่ตามลำดับคือ ไต้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ, และเหนือ โดยวิธีการของ AAA รูปที่ 3.2 แสดงแผนที่ประเทศไทยแบบ Original Image หมุนทวน

เข็มนาฬิกา 0 องศา, รูปที่ 3.3 แสดงแผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 90 องศา, รูปที่ 3.4 แสดงแผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 180 องศา, รูปที่ 3.5 แสดงแผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 270 องศา, รูปที่ 3.6 แสดงแผนที่ประเทศไทยแบบ Mirror Image 0 องศา, รูปที่ 3.7 แสดงแผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 90 องศา, รูปที่ 3.8 แสดงแผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 180 องศา, รูปที่ 3.9 แสดงแผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 270 องศา

รูปที่ 3.2 แผนที่ประเทศไทยแบบ Original Image 0 องศา



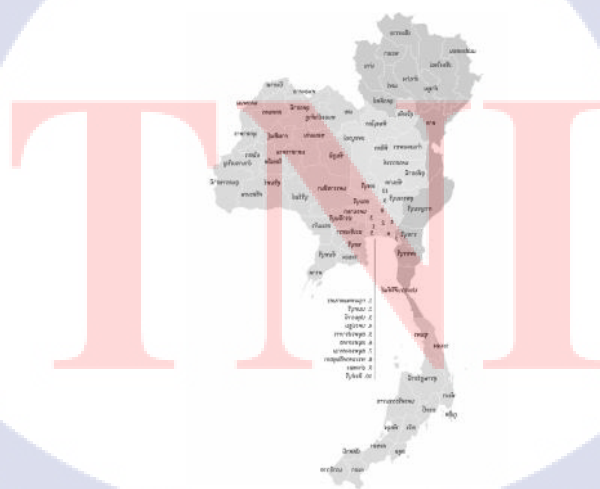
รูปที่ 3.3 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 90 องศา



รูปที่ 3.4 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 180 องศา



รูปที่ 3.5 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 270 องศา



รูปที่ 3.6 แผนที่ประเทศไทยแบบ Mirror Image 0 องศา



รูปที่ 3.7 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 90 องศา



รูปที่ 3.8 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 180 องศา



รูปที่ 3.9 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 270 องศา

ขั้นตอนการเดินทางแบบ AAA

Step1: เลือกรูปภาพ (Original หรือ Mirror) และดิกรี (0, 90, 180 หรือ 270) แผนที่สำหรับการใช้เดินทาง

ตารางที่ 3.1 รูปแบบการเดินทาง AAA

รูปแบบ	องศา	มุมมองการมองภาพ
1	0	แบบปกติ (Original Image)
2	90	แบบปกติ (Original Image)
3	180	แบบปกติ (Original Image)
4	270	แบบปกติ (Original Image)
5	0	แบบมองผ่านกระจก (Mirror Image)
6	90	แบบมองผ่านกระจก (Mirror Image)
7	180	แบบมองผ่านกระจก (Mirror Image)
8	270	แบบมองผ่านกระจก (Mirror Image)

Step2: ในแต่ละเมือง กำหนดเมืองที่มีอาณาเขตติดต่อทั้งหมด 8 ทิศ ดังนี้ ได้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ โดยมีขอบเขตของการกำหนดทิศว่าในการระบุทิศของแต่ละเมือง คือจะดูจากพื้นที่อาณาเขตติดต่อของเมืองนั้น มีอาณาเขตติดต่อมากที่สุดที่ทิศใด และในแต่ละทิศสามารถมีได้แค่เมืองเดียว และในแต่ละเมืองสามารถมีทิศได้แค่ทิศเดียวเช่นกัน ถ้ามีมากกว่า 1 เมืองในแต่ละทิศ ให้เลือกเมืองที่มีอาณาเขต ในทิศนั้นมากที่สุด

Step3: เริ่มเดินทางจากเมืองที่มีอาณาเขตสูงสุดตามแผนที่รูปภาพ

Step4: เดินทางตามเข็มนาฬิกาผ่านเมืองที่เป็นขอบของประเทศ (เมืองที่มีอาณาเขตติดต่อประเทศเพื่อนบ้าน หรือ ทะเล) จนถึงเมืองที่มีอาณาเขตต่ำสุดตามแผนที่รูปภาพ

Step5: เดินทางจากเมือง ณ ปัจจุบัน ไปยังเมืองถัดไปที่มีอาณาเขตติดต่อกับเมืองปัจจุบันที่ยังไม่เคยไปมาก่อนตาม Step2 ถ้ามีเมืองที่ยังไม่เคยไปมากกว่า 1 เมือง ให้เดินทางลำดับความสำคัญของทิศดังนี้ ได้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ หากในกรณีที่เมืองปัจจุบันไม่มีเมืองติดต่อยังไม่เคยไป ให้ไปยังเมืองที่ใกล้ที่สุด

Step6: ทำ Step5 ไปเรื่อยๆ จนกระทั่งไปครบทุกเมือง แล้วจึงค่อยเดินทางกลับมายังเมืองเริ่มต้น

ตัวอย่างวิธีการเดินทาง AAA - ได้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ



รูปที่ 3.10 ภาคตะวันออกของประเทศไทย Original Image 0 Degree

Step1: เลือกแบบใช้แผนที่ภาพแบบ Original Image 0 ดีกรี

Step2: กำหนดเมืองที่มีอาณาเขตติดต่อกับแต่ละเมืองทั้ง 8 ทิศ

Step3: เริ่มต้น คือ เมืองปราจีนบุรี คือเมืองที่มีอาณาเขตสูงสุดตามแผนที่รูปภาพ

Step4: เดินทางตามเส้นทางผ่านเมืองที่ขอบของประเทศจนถึงต่ำสุด คือ เมืองตราด

Step5: เดินทางจากเมือง ณ ปัจจุบัน ไปยังเมืองถัดไปที่มีอาณาเขตติดต่อกับเมืองปัจจุบันที่ยังไม่เคยไปมาก่อนตาม Step2 ถ้ามีเมืองที่ยังไม่เคยไปมากกว่า 1 เมือง ให้เดินทางลำดับความสำคัญของทิศ



รูปที่ 3.11 ตัวอย่างการเดินตามเข็มนาฬิกาผ่านเมืองที่เป็นขอบของภาคตะวันออก Original Image
0 Degree

Step5: เดินทางจากเมือง ณ ปัจจุบัน ไปยังเมืองถัดไปที่มีอาณาเขตติดต่อกับเมืองปัจจุบันที่ยังไม่เคยไปมาก่อนตาม Step2 ถ้ามีเมืองที่ยังไม่เคยไปมากกว่า 1 เมือง ให้เดินตามลำดับความสำคัญของทิศดังนี้ ได้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ หากในกรณีที่เมืองปัจจุบันไม่มีเมืองติดต่อยังไม่เคยไป ให้ไปยังเมืองที่ใกล้ที่สุด

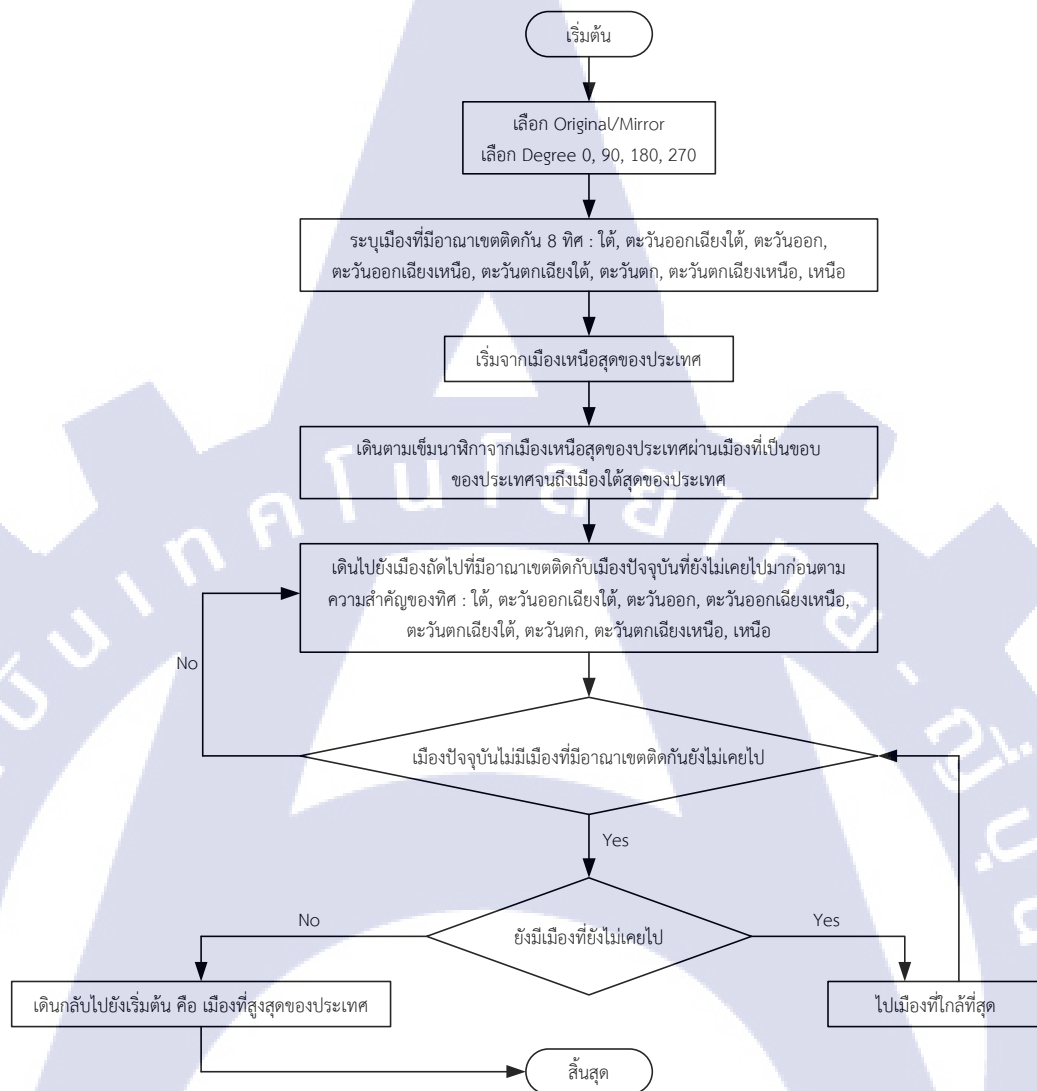
Step6: ทำ Step5 ไปเรื่อยๆ จนกระทั่งไปครบทุกเมือง แล้วจึงค่อยเดินกลับมายังเมืองเริ่มต้น



รูปที่ 3.12 ตัวอย่างการเดินทางด้วยวิธีการ AAA ภาคตะวันออก Original Image 0 Degree

เส้นทางที่ได้ คือ ปราจีนบุรี --- สระแก้ว --- จันทบุรี --- ตราด --- ระยอง --- ชลบุรี ---
ฉะเชิงเทรา --- ปราจีนบุรี

TNI

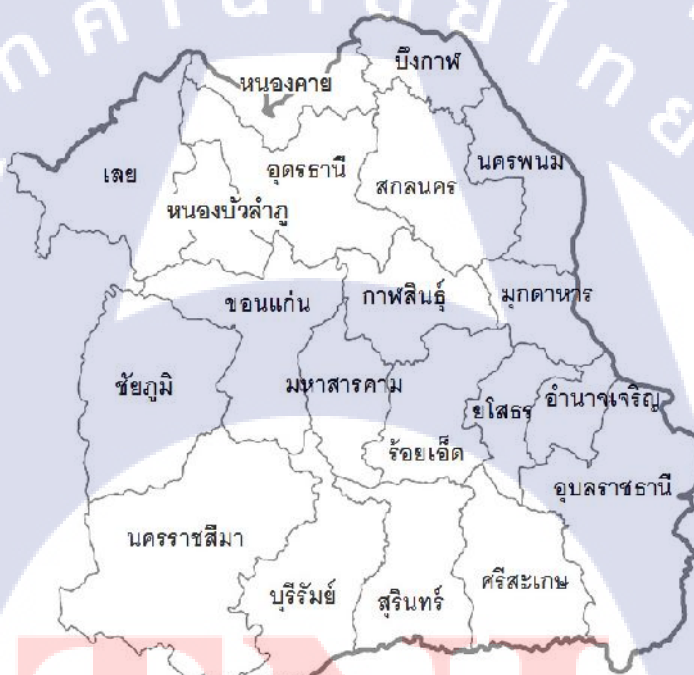


รูปที่ 3.13 Flowchart AAA

3.5.1 กลุ่มข้อมูลการทดลอง

งานวิจัยได้นำแผนที่ประเทศไทยมาใช้ในการทดลองการหาผลลัพธ์คำตอบ โดยมีการแบ่งกลุ่มการทดลองทั้งหมด 6 แบบ ให้มีความหลากหลายในรูปแบบแผนที่ของแต่ละกลุ่ม และเลือกใช้วัดสำคัญเป็นจุดอ้างอิงในแต่ละเมือง จากนั้นใช้ Google Map ในการหาระยะทางระหว่าง 2 วัดทั้งไปและกลับ โดยจะเก็บข้อมูลจาก 1 เมืองไปยังทุกเมืองที่เหลือ ถ้าระยะทางมีมากกว่า 1 ผลลัพธ์งานวิจัยจะเลือกเส้นทางที่มีระยะทางน้อยที่สุดในการเก็บข้อมูล โดยการเก็บข้อมูลจะเก็บอยู่ในรูปแบบ Excel File ซึ่งข้อมูลระยะทางในแต่ละกลุ่มการทดลองจัดเก็บอยู่ในแผ่นชีตงานวิจัย

3.5.1.1 ภาคตะวันออกเฉียงเหนือ (ภาคอีสาน) 20 เมือง



รูปที่ 3.14 ภาคอีสาน 0 Degree Normal Image

3.5.1.2 ภาคเหนือ 9 เมือง



รูปที่ 3.15 ภาคเหนือ 0 Degree Normal Image

3.5.1.3 ภาคตะวันออก 7 เมือง



รูปที่ 3.16 ภาคตะวันออก 0 Degree Normal Image

3.5.1.4 ภาคกลางตอนล่าง 15 เมือง



รูปที่ 3.17 ภาคกลางตอนล่าง 0 Degree Normal Image

3.5.1.5 ภาคกลางตอนบน 7 เมือง



รูปที่ 3.18 ภาคกลางตอนบน 0 Degree Normal Image

3.5.1.6 ภาคใต้กับภาคตะวันตก 19 เมือง



รูปที่ 3.19 ภาคใต้กับภาคตะวันตก 0 Degree Normal Image

3.5.2 ผลการวิจัยและการวิเคราะห์ผล

ในงานวิจัยได้มีการทดลองการรันวิธี AAA และวิธี NN สำหรับในแต่ละภาค งานวิจัยได้ดำเนินการทดลองวิธีการ AAA ทั้งหมด 8 รอบ ตามพื้นฐานทฤษฎี AAA และ NN ทั้งหมด 4 รอบ ดังนี้

ตารางที่ 3.2 รูปแบบการเดินทาง AAA และ NN

AAA		NN
Original Image (Degree)	Mirror Image (Degree)	(Degree)
0	0	0
90	90	90
180	180	180
270	270	270

นอกจากนี้ยังมีการทดสอบประสิทธิภาพการรันของวิธีการทั้ง AAA และ NN ด้วยฮาร์ดแวร์ PC Computer (Windows 7, 2.67 GHz Intel ® Core™2 Quad CPU Processor, 4.00 GB of Ram) ตามตารางที่ 3.3 ถึง ตารางที่ 3.8 แสดงระยะทางรวมไปกลับทั้งหมด 6 กลุ่มการทดลองจากวิธีการ AAA โดยใช้ดีกรี (0, 90, 180, หรือ 270) และการมองภาพ (Original หรือ Mirror) และวิธีการ NN และในแต่ละระยะทางรวม มีค่าเวลาการคำนวณ CPU แสดงอยู่ในวงเล็บ มีหน่วยการคำนวณเป็นวินาที

ตารางที่ 3.3 ระยะทางและเวลาการคำนวณภาคเหนือด้วยวิธีการ AAA และ NN

Degree	Original Image	Mirror Image	NN
0	1487.1 (0.003)	1535.7 (0.002)	1443.5 (0.006)
90	1582.1 (0.003)	1535.7 (0.002)	1658.1 (0.007)
180	1487.1 (0.002)	1535.7 (0.003)	1489.8 (0.007)
270	1534.4 (0.002)	1489.8 (0.002)	1772.5 (0.006)

สำหรับภาคเหนือ มี 2 เส้นทางที่พบในวิธีการ AAA มีระยะทางรวมดีสุด 1487.1 กิโลเมตร เท่ากัน โดยมีเส้นทางแบบที่ 1 คือ “เชียงใหม่-พะเยา-น่าน-อุตรดิตถ์-แพร่-ลำปาง-ลำพูน-เชียงใหม่-แม่ฮ่องสอน-เชียงราย” และเส้นทางแบบที่ 2 คือ “อุตรดิตถ์-แพร่-ลำปาง-ลำพูน-เชียงใหม่-แม่ฮ่องสอน-เชียงราย-พะเยา-น่าน-อุตรดิตถ์” ในขณะที่วิธีการ NN มีระยะทางรวมดีสุด 1443.5 กิโลเมตร โดยมีเส้นทางคือ “เชียงใหม่-พะเยา-น่าน-แพร่-อุตรดิตถ์-ลำปาง-ลำพูน-เชียงใหม่-แม่ฮ่องสอน-เชียงราย”

ตารางที่ 3.4 ระยะทางและเวลาการคำนวณภาคกลางตอนบนด้วยวิธีการ AAA และ NN

Degree	Original Image	Mirror Image	NN
0	819.2 (0.002)	792.8 (0.003)	865.8 (0.006)
90	748.5 (0.002)	964.2 (0.002)	759.3 (0.006)
180	794.4 (0.002)	818.3 (0.003)	841.2 (0.007)
270	953.9 (0.002)	853.3 (0.003)	839.5 (0.006)

สำหรับภาคกลางตอนบน มี 1 เส้นทางที่พบในวิธีการ AAA มีระยะทางรวมดีสุด 748.5 กิโลเมตร โดยมีเส้นทาง คือ “เพชรบูรณ์-นครสวรรค์-อุทัยธานี-กำแพงเพชร-สุโขทัย-พิษณุโลก-พิจิตร-เพชรบูรณ์” ในขณะที่วิธีการ NN มีระยะทางรวมดีสุด 759.3 กิโลเมตร โดยมีเส้นทางคือ “เพชรบูรณ์-พิจิตร-พิษณุโลก-สุโขทัย-กำแพงเพชร-นครสวรรค์-อุทัยธานี-เพชรบูรณ์”

ตารางที่ 3.5 ระยะทางและเวลาการคำนวณภาคกลางตอนล่างด้วยวิธีการ AAA และ NN

Degree	Original Image	Mirror Image	NN
0	912.8 (0.004)	895.5 (0.004)	1044.1 (0.009)
90	1070.0 (0.003)	1047.8 (0.004)	1042.1 (0.009)
180	962.4 (0.004)	996.4 (0.004)	966.4 (0.008)
270	1042.4 (0.004)	1040.9 (0.003)	1057.5 (0.008)

สำหรับภาคกลางตอนล่าง มี 1 เส้นทางที่พบในวิธีการ AAA มีระยะทางรวมดีสุด 895.5 กิโลเมตร โดยมีเส้นทาง คือ “ลพบุรี-สิงห์บุรี-ชัยนาท-สุพรรณบุรี-นครพนม-สมุทรสาคร-สมุทรสงคราม-กรุงเทพฯ-สมุทรปราการ-นนทบุรี-ปทุมธานี-นครนายก-สระบุรี-อยุธยา-อ่างทอง-ลพบุรี” ในขณะที่วิธีการ NN มีระยะทางรวมดีสุด 966.4 กิโลเมตร โดยมีเส้นทางคือ “สมุทรสงคราม-สมุทรสาคร-นครปฐม-นนทบุรี-

กรุงเทพ-สมุทรปราการ-ปทุมธานี-อยุธยา-อ่างทอง-ลพบุรี-สิงห์บุรี-ชัยนาท-สุพรรณบุรี-สระบุรี นครนายก-สมุทรสงคราม”

ตารางที่ 3.6 ระยะทางและเวลาการคำนวณภาคตะวันออกเฉียงเหนือ (ภาคอีสาน) ด้วยวิธีการ AAA และ NN

Degree	Original Image	Mirror Image	NN
0	759.9 (0.003)	747.9 (0.002)	797.0 (0.007)
90	759.9 (0.002)	747.9 (0.002)	797.0 (0.006)
180	743.6 (0.002)	763.6 (0.002)	743.6 (0.006)
270	759.9 (0.003)	747.9 (0.002)	747.9 (0.006)

สำหรับภาคตะวันออกเฉียงเหนือ มี 1 เส้นทางที่พบในวิธีการ AAA และ NN มีระยะทางรวมดีสุด 743.6 กิโลเมตรเท่ากัน โดยมีเส้นทาง คือ “ตราด-จันทบุรี-ระยอง-ชลบุรี-ฉะเชิงเทรา-ปราจีนบุรี-สระแก้ว-ตราด”

ตารางที่ 3.7 ระยะทางและเวลาการคำนวณภาคตะวันออกเฉียงเหนือ (ภาคอีสาน) ด้วยวิธีการ AAA และ NN

Degree	Original Image	Mirror Image	NN
0	2600.5 (0.004)	2993.3 (0.004)	2592.0 (0.009)
90	3061.5 (0.004)	2701.9 (0.004)	2949.2 (0.009)
180	2813.4 (0.005)	2720.5 (0.004)	2956.0 (0.009)
270	2900.7 (0.005)	2446.9 (0.004)	2294.3 (0.008)

สำหรับภาคตะวันออกเฉียงเหนือ (ภาคอีสาน) มี 1 เส้นทางที่พบในวิธีการ AAA มีระยะทางรวมดีสุด 2446.9 กิโลเมตร โดยมีเส้นทาง คือ “เลย-ขอนแก่น-ชัยภูมิ-นครราชสีมา-บุรีรัมย์-สุรินทร์-ศรีสะเกษ-อุบลราชธานี-อำนาจเจริญ-มุกดาหาร-ยโสธร-ร้อยเอ็ด-กาฬสินธุ์-มหาสารคาม-สกลนคร-นครพนม-บึงกาฬ-หนองคาย-อุดรธานี-หนองบัวลำภู-เลย” ในขณะที่วิธีการ NN มีระยะทางรวมดีสุด 2294.3 กิโลเมตร โดยมีเส้นทางคือ “เลย-หนองบัวลำภู-อุดรธานี-หนองคาย-บึงกาฬ-นครพนม-สกลนคร-มุกดาหาร-อำนาจเจริญ-อุบลราชธานี-ศรีสะเกษ-ยโสธร-สุรินทร์-บุรีรัมย์-ร้อยเอ็ด-กาฬสินธุ์-มหาสารคาม-ขอนแก่น-ชัยภูมิ-นครราชสีมา-เลย”

ตารางที่ 3.8 ระยะทางและเวลาการคำนวณภาคใต้และตะวันตกด้วยวิธีการ AAA และ NN

Degree	Original Image	Mirror Image	NN
0	3735.6 (0.004)	3761.9 (0.004)	4224.3 (0.009)
90	3728.8 (0.004)	3768.5 (0.004)	4132.3 (0.008)
180	3728.8 (0.004)	4373.3 (0.004)	4132.3 (0.008)
270	3735.6 (0.004)	3761.9 (0.004)	4224.3 (0.009)

สำหรับภาคใต้และตะวันตกมี 2 เส้นทางที่พบในวิธีการ AAA มีระยะทางรวมที่สุด 3728.8 กิโลเมตร โดยมีเส้นทาง คือ “ยะลา-สงขลา-สตูล-ตรัง-กระบี่-ภูเก็ต-พังงา-ระนอง-ชุมพร-ประจวบคีรีขันธ์-เพชรบุรี-ราชบุรี-กาญจนบุรี-ตาก-สุราษฎร์ธานี-นครศรีธรรมราช-พัทลุง-ปัตตานี-นราธิวาส-ยะลา” และ “นราธิวาส-ยะลา-สงขลา-สตูล-ตรัง-กระบี่-ภูเก็ต-พังงา-ระนอง-ชุมพร-ประจวบคีรีขันธ์-เพชรบุรี-ราชบุรี-กาญจนบุรี-ตาก-สุราษฎร์ธานี-นครศรีธรรมราช-พัทลุง-ปัตตานี-นราธิวาส” ในขณะที่วิธีการ NN มี 2 เส้นทางที่มีระยะทางรวมที่สุด 4132.3 กิโลเมตร โดยมีเส้นทางคือ “ยะลา-ปัตตานี-สงขลา-สตูล-พัทลุง-ตรัง-นครศรีธรรมราช-สุราษฎร์ธานี-ภูเก็ต-กระบี่-ระนอง-ชุมพร-ประจวบคีรีขันธ์-เพชรบุรี-ราชบุรี-กาญจนบุรี-ตาก-นราธิวาส-ยะลา” และ “นราธิวาส-ยะลา-ปัตตานี-สงขลา-สตูล-พัทลุง-ตรัง-นครศรีธรรมราช-สุราษฎร์ธานี-พังงา-ภูเก็ต-กระบี่-ระนอง-ชุมพร-ประจวบคีรีขันธ์-เพชรบุรี-ราชบุรี-กาญจนบุรี-ตาก-นราธิวาส”

จากการทดลองทั้ง 6 ภาคระหว่างวิธีการ AAA และ NN ปรากฏว่า AAA มีประสิทธิภาพดีกว่าใน 3 ภาค คือ ภาคกลางตอนบน, ภาคกลางตอนล่าง, และภาคใต้และภาคตะวันตก และ NN มีประสิทธิภาพดีกว่าใน 2 ภาค คือ ภาคเหนือ และ ภาคตะวันออกเฉียงเหนือ (ภาคอีสาน) และวิธี AAA และ NN มีประสิทธิภาพดีเท่ากันใน 1 ภาค คือ ภาคตะวันออก นอกจากนี้ในงานวิจัยยังได้วิเคราะห์และทำการเปรียบเทียบระยะทางรวมที่ดีที่สุดของ AAA และระยะทางรวมของวิธีการ NN เริ่มจากเหนือสุดของประเทศ เนื่องจากว่าวิธีการ NN เป็นวิธีการที่เดินไปยังเมืองถัดไปที่ใกล้ที่สุดโดยเริ่มต้นจากเมืองเหนือสุดของประเทศเทียบกับ Lower Bound (LB) โดยในงานวิจัยได้เลือกใช้การหาค่า Lower Bound จากในแต่ละ 1 เมือง ถึงทุกเมืองในเมือง N และค่า Lower Bound คือผลรวมที่สั้นที่สุดของเมือง 1 ไปยังทุกเมือง N, เมือง 1 ไปยังทุกเมือง N ฯลฯ ในส่วนของตารางที่ 3.9 จะแสดงถึงระยะทางรวมที่ดีที่สุดของ AAA, ระยะทางรวมของวิธีการ NN เริ่มจากเหนือสุดของประเทศ และ LB และในส่วนของตารางที่ 3.10 แสดงถึงค่าร้อยละความต่างระหว่างค่าที่ดีที่สุดของวิธี AAA และค่าคำตอบของวิธีการ NN เริ่มจากเหนือสุดของประเทศในแต่ละกลุ่มการทดลอง

ตารางที่ 3.9 เปรียบเทียบค่าที่ดีที่สุดระหว่าง AAA และ คำคำตอบของวิธีการ NN เริ่มจากเหนือสุดของประเทศเทียบกับ LB

Instance	LB	AAA	NN
ภาคเหนือ	958.1	1487.1	1443.5
ภาคกลางตอนบน	505.5	748.5	865.8
ภาคกลางตอนล่าง	581.3	895.5	1044.1
ภาคตะวันออกเฉียงเหนือ	1639.3	2446.9	2592
ภาคตะวันออก	593.6	743.6	797
ภาคใต้และภาคตะวันตก	1971.7	3728.8	4224.3

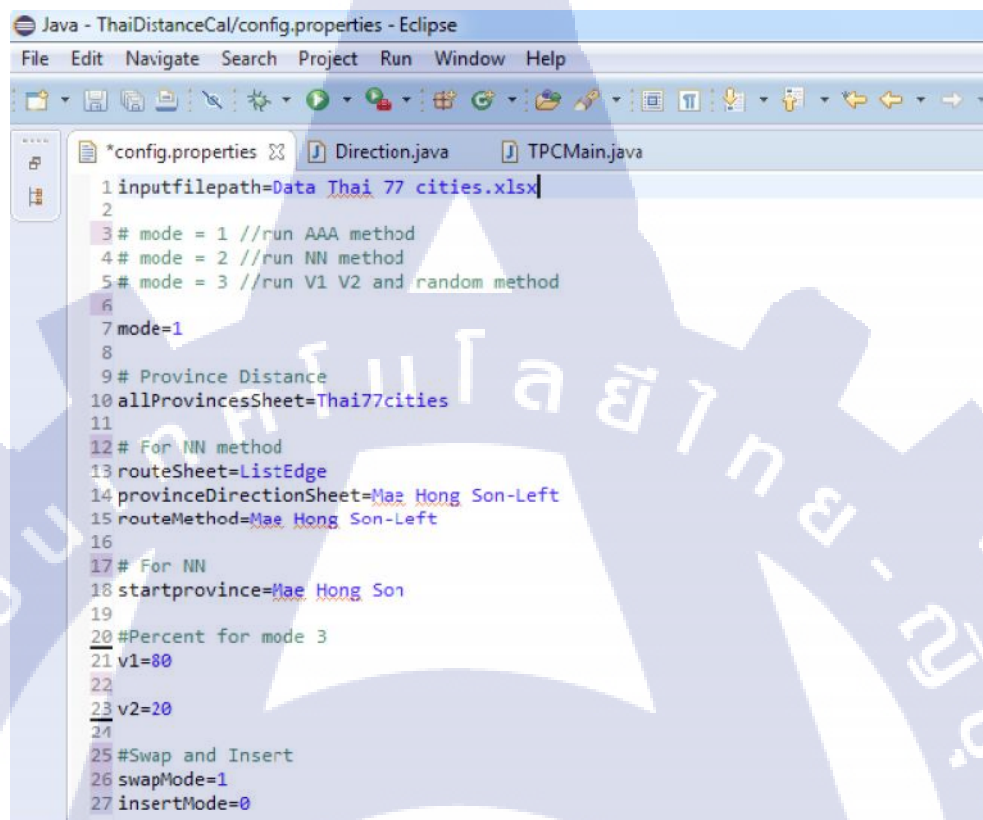
ตารางที่ 3.10 เปรียบเทียบค่าร้อยละความต่างระหว่างค่าที่ดีที่สุดของวิธี AAA และคำตอบของวิธีการ NN เริ่มจากเหนือสุดของประเทศ

Instance	NN	AAA	ค่าร้อยละความต่าง
ภาคเหนือ	1443.5	1487.1	-3.0
ภาคกลางตอนบน	865.8	748.5	13.6
ภาคกลางตอนล่าง	1044.1	895.5	16.0
ภาคตะวันออกเฉียงเหนือ	2592	2446.9	5.6
ภาคตะวันออก	797	743.6	6.7
ภาคใต้และภาคตะวันตก	4224.3	3728.8	11.7

ตามตารางที่ 3.9 แสดงการเปรียบเทียบค่าที่ดีที่สุดระหว่าง AAA, NN และ LB จะเห็นว่าวิธีการ AAA มีระยะทางรวมชนะวิธีการของ NN เริ่มจากเหนือสุดของประเทศ 5 กลุ่มการทดลอง และตารางที่ 3.10 แสดงถึงค่าร้อยละความต่างจากการพัฒนาด้วยวิธีการ AAA เมื่อเปรียบเทียบกับคำตอบของวิธีการ NN เริ่มจากเหนือสุดของประเทศ

ในส่วนของการประสิทธิภาพการรันที่ได้แสดงในตารางที่ 3.3 ถึงตารางที่ 3.8 แสดงให้เห็นว่าประสิทธิภาพการคำนวณของทั้งวิธีการแบบ AAA และวิธีการแบบ NN นั้นรวดเร็ว โดยวิธีการแบบ AAA มีเวลา CPU ในการรันสูงสุดคือ 0.005 วินาที ในขณะที่วิธีการแบบ NN มีเวลา CPU ในการรันสูงสุดคือ 0.009 วินาที ต่อประสิทธิภาพของคอมพิวเตอร์ที่ใช้ในการทดลองในงานวิจัยครั้งนี้

3.6 การทำงานของโปรแกรม



```

Java - ThaiDistanceCal/config.properties - Eclipse
File Edit Navigate Search Project Run Window Help

*config.properties | Direction.java | TPCMain.java
1 inputfilepath=Data Thai 77 cities.xlsx
2
3 # mode = 1 //run AAA method
4 # mode = 2 //run NN method
5 # mode = 3 //run V1 V2 and random method
6
7 mode=1
8
9 # Province Distance
10 allProvincesSheet=Thai77cities
11
12 # For NN method
13 routeSheet=ListEdge
14 provinceDirectionSheet=Mae Hong Son-Left
15 routeMethod=Mae Hong Son-Left
16
17 # For NN
18 startprovince=Mae Hong Son
19
20 #Percent for mode 3
21 v1=80
22
23 v2=20
24
25 #Swap and Insert
26 swapMode=1
27 insertMode=0
  
```

รูปที่ 3.20 หน้าจอหลักของโปรแกรม

```

Java - ThaiDistanceCal/config.properties - Eclipse
File Edit Navigate Search Project Run Window Help

*config.properties | Direction.java | TPCMain.java

1 inputfilepath=Data Thai 77 cities.xlsx | ← 1
2
3 # mode = 1 //run AAA method
4 # mode = 2 //run NN method
5 # mode = 3 //run V1 V2 and random method
6
7 mode=1 | ← 2
8
9 # Province Distance
10 allProvincesSheet=Thai77cities | ← 3
11
12 # For NN method
13 routeSheet=ListEdge | ← 4
14 provinceDirectionSheet=Mae Hong Son-Left | ← 5
15 routeMethod=Mae Hong Son-Left
16
17 # For NN
18 startprovince=Mae Hong Son | ← 6
19
20 #Percent for mode 3
21 v1=80 | ← 7
22
23 v2=20 | ← 8
24
25 #Swap and Insert
26 swapMode=1 | ← 8
27 insertMode=0

```

รูปที่ 3.21 อธิบายส่วนประกอบของหน้าจอหลัก

จุดที่ 1 สำหรับเลือกไฟล์ข้อมูลที่ประกอบด้วย ข้อมูลของทิศ ข้อมูลระยะทาง และข้อมูลเมืองที่เป็นขอบของประเทศ (.xls, .xlsx)

จุดที่ 2 สำหรับเลือก Mode ของรูปแบบวิธีการหาเส้นทางการเดินทาง

- Mode = 1 คือ วิธีการหาเส้นทางการเดินทางแบบจัดลำดับความสำคัญของทิศ
- Mode = 2 คือ วิธีการหาเส้นทางการเดินทางแบบ NN
- Mode = 3 คือ วิธีการแบบ Randomization

จุดที่ 3 สำหรับระบุชื่อ Tab ของไฟล์ Excel ที่มีข้อมูลระยะทางจากเมืองหนึ่งไปยังทุกเมืองของกลุ่มการทดลองที่ใช้

จุดที่ 4 สำหรับระบุชื่อ Tab ของไฟล์ Excel ที่มีข้อมูลการ List เส้นทางการเดินทางจากเมืองที่มีอาณาเขตสูงสุดของประเทศตามแผนที่รูปภาพ เดินตามเข็มนาฬิกาผ่านเมืองที่เป็นขอบของประเทศไปยังเมืองที่มีอาณาเขตต่ำสุดของประเทศตามแผนที่รูปภาพ

จุดที่ 5 สำหรับการอ้างอิงข้อมูลของทิศเมื่อเมืองจุดเริ่มต้นมีการเปลี่ยนแปลงไปตามมองศาที่เลือกใช้ และมุมมองการมองภาพแผนที่

จุดที่ 6 สำหรับการเลือก List ของการเดินทางจากเมืองที่มีอาณาเขตสูงสุดของประเทศตามแผนที่รูปภาพ เดินตามเข็มนาฬิกาผ่านเมืองที่เป็นขอบของประเทศไปยังเมืองที่มีอาณาเขตต่ำสุดของประเทศตามแผนที่รูปภาพ เมื่อเมืองจุดเริ่มต้นมีการเปลี่ยนแปลงไปตามมองศาที่เลือกใช้ และมุมมองการมองภาพแผนที่

จุดที่ 7 ใช้สำหรับการเลือกใช้แบบวิธี Randomization

จุดที่ 8 ใช้สำหรับการเลือกใช้วิธี Local Search แบบ SWAP และ INSERT โดยการระบุ 1 และ 0 หากเลือก 1 ที่ Swap นั้นหมายความว่าใช้วิธีการของ Swap

3.6.1 การใช้งานโปรแกรม

3.6.1.1 เลือกไฟล์ข้อมูล .xls ใส่ลงไปใน Inputfilepath

3.6.1.2 เลือก Mode ที่ต้องการทดสอบ

3.6.1.3 ระบุชื่อ Tab ของไฟล์ Excel ที่มีข้อมูลระยะทางจากเมืองหนึ่งไปยังทุกเมืองของกลุ่มการทดลองที่ใช้ใน All Provinces Sheet

3.6.1.4 ระบุชื่อ Tab ของไฟล์ Excel ที่มีข้อมูลการ List เส้นทางการเดินทางจากเมืองที่มีอาณาเขตสูงสุดของประเทศตามแผนที่รูปภาพ เดินตามเข็มนาฬิกาผ่านเมืองที่เป็นขอบของประเทศไปยังเมืองที่มีอาณาเขตต่ำสุดของประเทศตามแผนที่รูปภาพใน Route Sheet

3.6.1.5 ระบุ Tab Name ของทิศที่ต้องการใช้ Province Direction Sheet

3.6.1.6 ระบุชื่อเส้นทางของการเดินขอบ Route Method

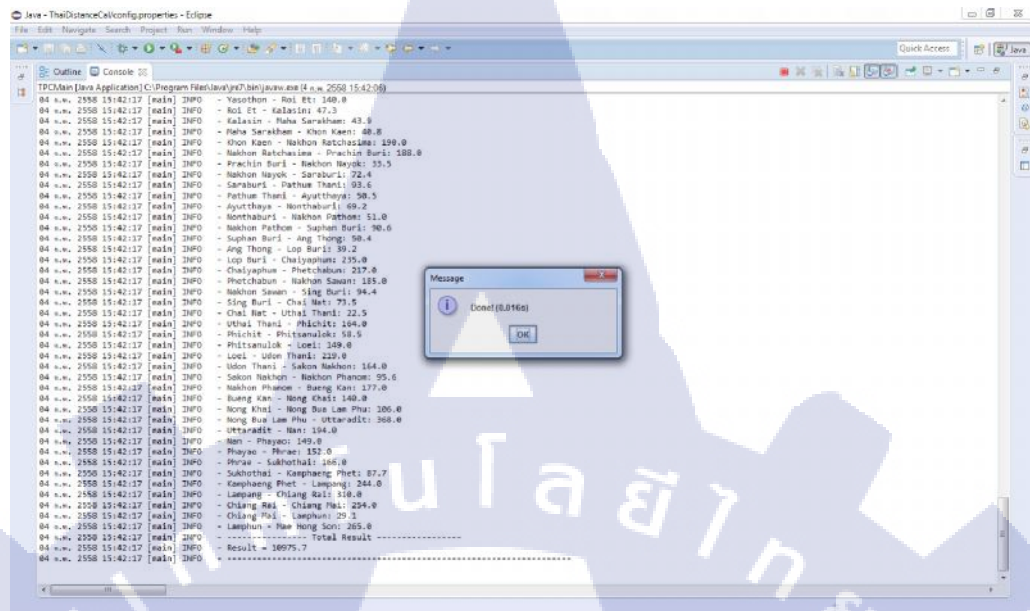
3.6.1.7 เลือกเมืองเริ่มต้น เฉพาะวิธีการ NN ใน Start Province

3.6.1.8 ระบุเปอร์เซ็นต์ของ V1 และ V2 เฉพาะวิธีการ Randomization หากเลือกใช้วิธีการนี้ต้องเลือก Mode = 3

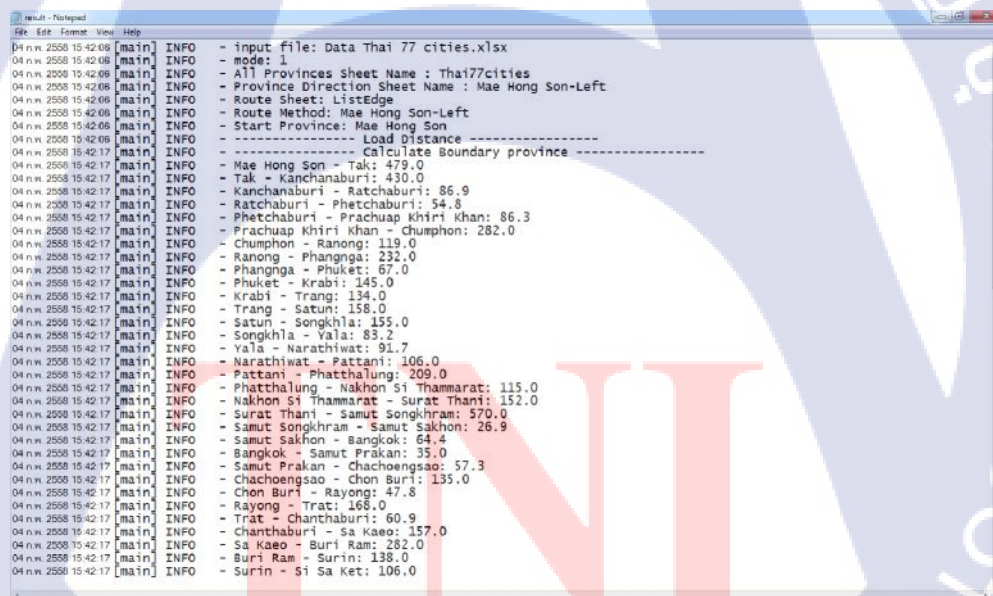
3.6.1.9 ระบุ 0 หรือ 1 สำหรับการเลือกใช้ SWAP หรือ INSERT เฉพาะวิธีการแบบ Local Search หากเลือกใช้วิธีการนี้ต้องเลือก Mode = 1

3.6.1.10 กดรันที่ 

3.6.1.11 ผลลัพธ์ออกมาในรูปแบบ .txt พร้อมเวลาที่ใช้ในการประมวลผลหาคำตอบด้วย Message Window



รูปที่ 3.22 รูปแบบของการแสดงผลของ Eclipse



รูปที่ 3.23 รูปแบบของการแสดงผล .txt File

	A	B	C	D	E	F	G	H	I
1			1	2	3	4	5	6	7
2			วัดแก้ว หรือ ปราจีนบุรี	วัดนคร ธรรม สระแก้ว	วัดโพธิ์ ราม อินท ระบุรี	วัดช่อง แสง นบพิต	วัดป่า อู่ ระยอง	วัดโกล นาค นบพิต	วัดท่า เส ตรา
3	Prachin Buri	Prachin Buri	0	116	77.9	191	182	242	292
4	Sa Kaeo	Sa Kaeo	117	0	148	246	247	157	201
5	Chachoengsao	Chachoengsao	76.8	147	0	135	134	191	250
6	Chon Buri	Chon Buri	190	247	100	0	47.0	150	217
7	Rayong	Rayong	183	238	131	47.9	0	108	168
8	Chanthaburi	Chanthaburi	241	157	191	156	103	0	61.2
9	Trat	Trat	293	201	251	215	163	60.9	0
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									
25									

รูปที่ 3.24 ตัวอย่างรูปแบบของการเก็บข้อมูลระยะทาง

	A	B	C	D	E	F	G	H	I	J
1	NO.	จังหวัด	จังหวัด	จังหวัด	จังหวัด	จังหวัด	จังหวัด	จังหวัด	จังหวัด	จังหวัด
2	1	Prachin Buri	Chachoengsao	Sa Kaeo						
3	2	Sa Kaeo	Chanthaburi				Chachoengsao	Prachin Buri		
4	3	Chachoengsao	Chanthaburi	Sa Kaeo	Prachin Buri	Chon Buri				
5	4	Chon Buri	Rayong	Chanthaburi					Chachoengsao	
6	5	Rayong		Chanthaburi					Chon Buri	
7	6	Chanthaburi	Trat				Rayong	Chachoengsao	Sa Kaeo	
8	7	Trat							Chanthaburi	
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										

รูปที่ 3.25 ตัวอย่างรูปแบบของการเก็บข้อมูลของทิศ 0 Degree Normal Image

	A	B	C	D	F	F	G
1		จุดสูงสุด					จุดต่ำสุด
2	Route1-วนซ้าย	Prachin Buri	Chachoengsao	Chon Buri	Rayong	Chanthaburi	Trat
3	Route2-วนขวา	Prachin Buri	Sa Kaeo	Charthaburi	Trat		
4	Route3-วนซ้าย	Trat	Chanthaburi	Sa Kaeo	Prachin Buri		
5	Route4-วนขวา	Trat	Chanthaburi	Rayong	Chon Buri	Chachoengsao	Prachin Buri
6	Route5-วนซ้าย	Sa Kaeo	Prachin Buri	Chachoengsao	Chon Buri		
7	Route6-วนขวา	Sa Kaeo	Chanthaburi	Trat	Rayong	Chon Buri	
8	Route7-วนซ้าย	Chon Buri	Rayong	Charthaburi	Trat	Sa Kaeo	
9	Route8-วนขวา	Chon Buri	Chachoengsao	Prachin Buri	Sa Kaeo		
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							
25							
26							
27							

รูปที่ 3.26 ตัวอย่างรูปแบบของการเก็บข้อมูลของเมืองที่เป็นขอบของประเทศ

บทที่ 4

ระเบียบวิธีวิจัย ผลการวิจัยและการวิเคราะห์ผลการทดลอง

Deterministic, Stochastic, and Local Search Algorithms

4.1 Deterministic Algorithm

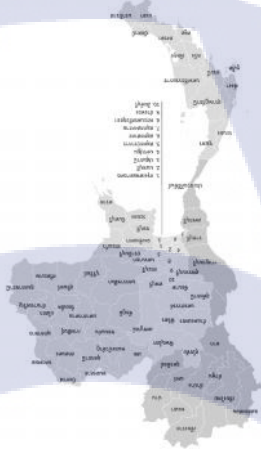
วิธีการ Deterministic Algorithm ในงานวิจัยเป็นวิธีการพัฒนาของวิธีการ AAA (Adjacent are Algorithm) [13] เพื่อที่จะปรับปรุงให้ระยะทางรวมลดลงมากยิ่งขึ้น โดยความแตกต่างระหว่าง AAA และ DEA คือ AAA ต้องเลือกทิศทาง (0, 90, 180, 270) และการมองภาพแบบ มองรูปภาพปกติ (Normal Image) หรือ มองรูปภาพผ่านกระจก (Mirror Image) ในขณะที่วิธีการแบบ DEA ทุกการผสมผสานของทิศทางมองภาพและมุมมองภาพถูกเลือกครั้งหนึ่งและจะเลือกค่าระยะทางรวมที่ดีที่สุดมา อีกทั้งวิธีการ AAA จะมีลำดับความสำคัญของทิศที่ตั้งนี้ ได้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ, และเหนือ ในขณะที่วิธีการของ DEA สามารถเลือกลำดับตามความสำคัญของทิศได้ โดยวิธีการของ DEA จะมีการหมุนองศาและมุมมองการมองภาพแผนที่ตามรูปที่ 4.1- รูปที่ 4.8



รูปที่ 4.1 แผนที่ประเทศไทยแบบ Original Image 0 องศา



รูปที่ 4.2 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 90 องศา



รูปที่ 4.3 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 180 องศา

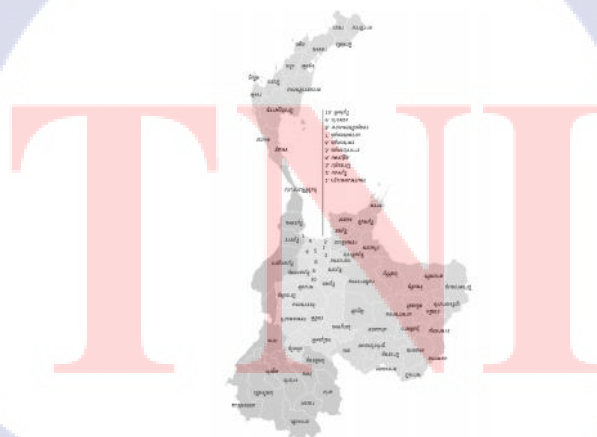


รูปที่ 4.4 แผนที่ประเทศไทยแบบ Original Image หมุนทวนเข็มนาฬิกา 270 องศา

รูปที่ 4.5 แผนที่ประเทศไทยแบบ Mirror Image 0 ดีกรี



รูปที่ 4.6 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 90 ดีกรี



รูปที่ 4.7 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 180 ดีกรี



รูปที่ 4.8 แผนที่ประเทศไทยแบบ Mirror Image โดยหมุนภาพ Original Image ทวนเข็มนาฬิกา 270 องศา

4.1.1 ขั้นตอนการเดินทางแบบ DEA

Step1: $t=1$

Step2: เลือกตัวชี้วัดของการหมุนภาพแผนที่และมุมมองการมองภาพของแผนที่

- ถ้า $t=1$, ใช้แผนที่ภาพแบบ Original Image 0 องศา
- ถ้า $t=2$, ใช้แผนที่ภาพแบบ Original Image 90 องศา
- ถ้า $t=3$, ใช้แผนที่ภาพแบบ Original Image 180 องศา
- ถ้า $t=4$, ใช้แผนที่ภาพแบบ Original Image 270 องศา
- ถ้า $t=5$, ใช้แผนที่ภาพแบบ Mirror Image โดยการหมุนภาพ Original Image ทวนเข็มนาฬิกา 0 องศา จากนั้นเปลี่ยนมุมมองการมองเป็นผ่านกระจก
- ถ้า $t=6$, ใช้แผนที่ภาพแบบ Mirror Image โดยการหมุนภาพ Original Image ทวนเข็มนาฬิกา 90 องศา จากนั้นเปลี่ยนมุมมองการมองเป็นผ่านกระจก
- ถ้า $t=7$, ใช้แผนที่ภาพแบบ Mirror Image โดยการหมุนภาพ Original Image ทวนเข็มนาฬิกา 180 องศา จากนั้นเปลี่ยนมุมมองการมองเป็นผ่านกระจก
- ถ้า $t=8$, ใช้แผนที่ภาพแบบ Mirror Image โดยการหมุนภาพ Original Image ทวนเข็มนาฬิกา 270 องศา จากนั้นเปลี่ยนมุมมองการมองเป็นผ่านกระจก

Step3: งานวิจัยได้มีการศึกษาค้นคว้าข้อมูลโดยมีการกำหนดทิศทั้งหมด 8 ทิศ โดยมีขอบเขตของการกำหนดทิศว่าในการระบุทิศของแต่ละเมือง ก็จะดูจากพื้นที่อาณาเขตติดต่อของเมืองนั้น มีอาณาเขตติดต่อกับทิศใด และในแต่ละทิศสามารถมีได้แค่เมืองเดียว และในแต่ละเมืองสามารถมีทิศได้แค่ทิศเดียวเช่นกัน ถ้ามีมากกว่า 1 เมืองในแต่ละทิศ ให้เลือกเมืองที่มีอาณาเขตในทิศนั้นมากที่สุด

Step4: เริ่มต้นจากเมืองที่มีอาณาเขตสูงสุดของประเทศบนพื้นฐานของดัชนีการหมุนและมุมมองภาพที่เลือก

Step5: เดินตามขอบของประเทศตามเข็มนาฬิกาไปจนถึงเมืองที่มีอาณาเขตต่ำสุดของประเทศตามแผนที่รูปภาพ

Step6: เดินทางจากเมือง ณ ปัจจุบัน ไปยังเมืองถัดไปที่มีอาณาเขตติดต่อกับเมืองปัจจุบันที่ยังไม่เคยไปมาก่อนตาม ถ้ามีเมืองที่ยังไม่เคยไปมากกว่า 1 เมือง ให้เดินตามลำดับความสำคัญของทิศ หากเมืองปัจจุบันไม่มีอาณาเขตติดต่อกับเมืองใด หรือเมืองปัจจุบันไม่มีเมืองอาณาเขตติดต่อยังไม่เคยไปหมดแล้ว ก็ให้ไปยังเมืองที่ใกล้ที่สุด

Step7: ทำตาม Step6 ไปเรื่อยๆ จนกระทั่งครบทุกเมืองที่ยังไม่เคยไป แล้วเดินกลับไปยังเมืองเริ่มต้น หลังจากที่ได้เส้นทางการเดินทางสิ้นสุด ให้ Π_t เป็นลำดับของเมืองในเส้นทางการเดินทางครั้งนั้นในแต่ละรอบ t และ D_t คือระยะทางรวมของ Π_t

Step8: ถ้า $t = 1$, ให้ $\Pi = \Pi_1$ และ $D = D_1$ ถ้า $t > 1$, ให้ $\Pi = \Pi_t$ และ $D = D_t$ ก็ต่อเมื่อ $D_t < D$; และจะไม่มีการเปลี่ยนแปลง และ D ถ้า $D_t \geq D$

Step9: ถ้า $t = 8$, หยุดการทำงาน, Π และ D คือเส้นทางไป-กลับที่ดีที่สุดและระยะทางรวมที่ดีที่สุด ไม่เช่นนั้นเพิ่ม t ไปทีละ 1 และทำซ้ำ Step 2

ในงานวิจัยได้มีการจัดลำดับความสำคัญของทิศเพื่อใช้ในการวิเคราะห์วิธีการแบบ DEA ทั้งหมด 6 รูปแบบดังนี้

1. DEA-1: ได้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ โดย ทิศใต้คือทิศที่มีความสำคัญอันดับ 1 ทิศตะวันออกเฉียงใต้คือทิศที่มีความสำคัญอันดับที่ 2 และทิศที่เหลือความสำคัญก็ลดหลั่นกันมา โดยลำดับความสำคัญของทิศของ DEA-1 คือลำดับความสำคัญของทิศแบบ AAA [13]

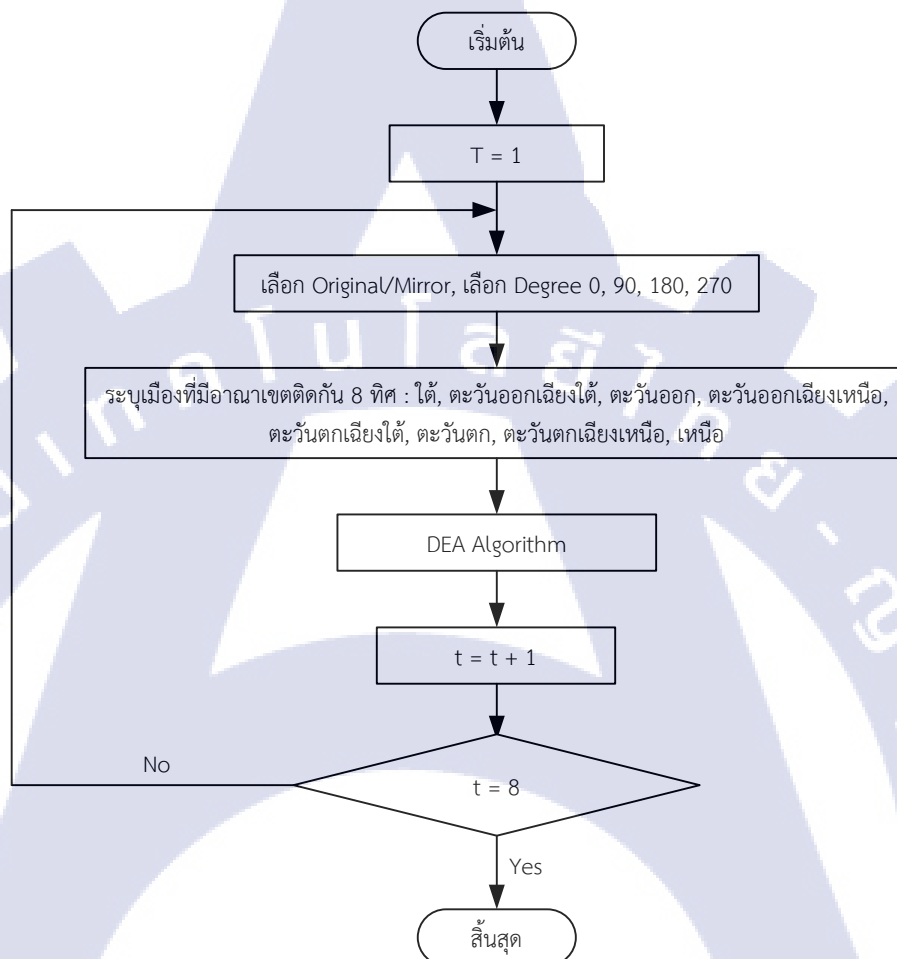
2. DEA-2: ได้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงเหนือ และเหนือ

3. DEA-3: ได้-ตะวันออกเฉียงใต้, ตะวันตกเฉียงใต้, ตะวันออก, ตะวันตก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงเหนือ และเหนือ

4. DEA-4: ได้, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ และเหนือ

5. DEA-5: ได้, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันตกเฉียงเหนือ, ตะวันออกเฉียงเหนือ และเหนือ

6. DEA-6 : ใต้, ตะวันตกเฉียงใต้, ตะวันออกเฉียงใต้, ตะวันตก, ตะวันออก, ตะวันตกเฉียงเหนือ, ตะวันออกเฉียงเหนือ และเหนือ

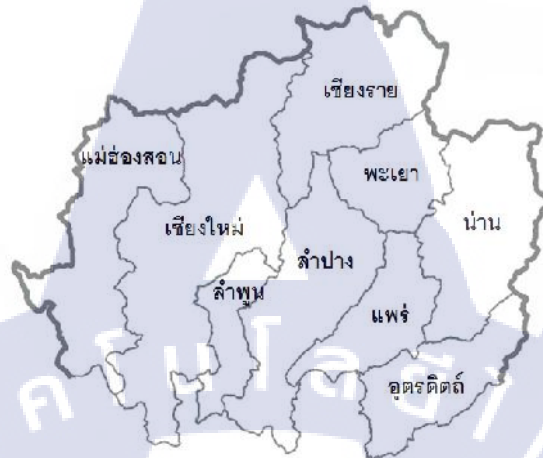


รูปที่ 4.9 Flowchart DEA

4.1.2 ผลการวิจัยและการวิเคราะห์ผล DEA

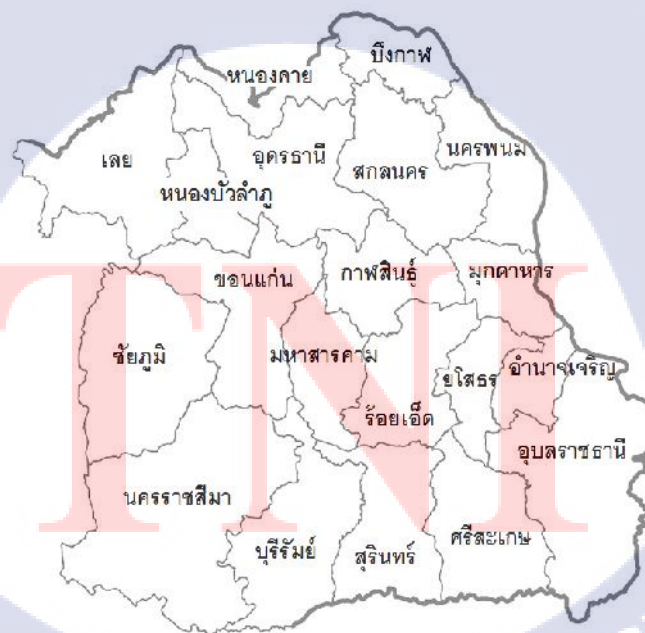
โดยงานวิจัยได้ใช้ ฮาร์ดแวร์ PC Computer (Windows 7, 2.67 GHz Intel ® Core™2 Quad CPU Processor, 4.00 GB of Ram) ในการทดสอบประสิทธิภาพการรัน ซึ่งระยะเวลาการคำนวณเฉลี่ยมีหน่วยเป็นวินาทีที่จะแสดงผลในวงเล็บตามตารางที่ 4.3 และในครั้งนี้นักวิจัยได้เลือกกลุ่มการทดลองทั้งหมด 6 กลุ่มดังรูปที่ 4.9 - 4.14 ในงานวิจัยได้มีการรันและเก็บผลวิธี 6 DEA Algorithm และ NN รัน 6 กลุ่มการทดลองเช่นกัน ซึ่งวิธีการ NN แสดงวิธีการ Nearest Neighbor Algorithm เริ่มต้นจากเมืองที่มีอาณาเขตสูงสุดของประเทศตามแผนที่รูปภาพ

4.1.2.1 ภาคเหนือ 9 เมือง



รูปที่ 4.10 ภาคเหนือ 0 Degree Normal Image

4.1.2.2 ภาคตะวันออกเฉียงเหนือ (ภาคอีสาน) 20 เมือง



รูปที่ 4.11 ภาคอีสาน 0 Degree Normal Image

4.1.2.3 ภาคตะวันออก 7 เมือง



รูปที่ 4.12 ภาคตะวันออก 0 Degree Normal Image

4.1.2.4 ภาคกลาง 22 เมือง



รูปที่ 4.13 ภาคกลาง 0 Degree Normal Image

4.1.2.5 ภาคใต้และภาคตะวันตก 19 เมือง



รูปที่ 4.14 ภาคใต้กับภาคตะวันตก 0 Degree Normal Image

4.1.2.6 ประเทศไทย 77 เมือง



รูปที่ 4.15 ประเทศไทย 0 Degree Normal Image

ตารางที่ 4.1 เปรียบเทียบระยะทางรวมของ DEA Algorithms กับ NN

Instance	NN	DEA-1	DEA-2	DEA-3	DEA-4	DEA-5	DEA-6
ภาคเหนือ	1443.5	1487.1	1487.1	1487.1	1487.1	1487.1	1487.1
ภาคตะวันออกเฉียงเหนือ	2592.0	2446.9	2600.5	2600.5	2600.5	2600.5	2600.5
ภาคตะวันออก	797.0	743.6	743.6	743.6	743.6	743.6	743.6
ภาคกลาง	1754.1	1731.3	1731.3	1731.3	1731.3	1731.3	1731.3
ภาคใต้และตะวันตก	4224.3	3728.8	3728.8	3728.8	3728.8	3728.8	3728.8
ประเทศไทย	11805.3	10650.6	10459.7	10459.7	10763.1	10629.4	11000.6

ตารางที่ 4.2 เปอร์เซ็นต์การพัฒนาของ DEA Algorithms จากผลลัพธ์ของ NN

Instance	DEA-1	DEA-2	DEA-3	DEA-4	DEA-5	DEA-6
ภาคเหนือ	-3.0	-3.0	-3.0	-3.0	-3.0	-3.0
ภาคตะวันออกเฉียงเหนือ	5.6	-0.3	-0.3	-0.3	-0.3	-0.3
ภาคตะวันออก	6.7	6.7	6.7	6.7	6.7	6.7
ภาคกลาง	1.3	1.3	1.3	1.3	1.3	1.3
ภาคใต้และตะวันตก	11.7	11.7	11.7	11.7	11.7	11.7
ประเทศไทย	9.8	11.4	11.4	8.8	10.0	6.8
ค่าเฉลี่ย	5.4	4.6	4.6	4.2	4.4	3.9

ตารางที่ 4.3 เปรียบเทียบเวลาเฉลี่ยของการรัน DEA Algorithms (หน่วย : วินาที)

Instance	DEA-1	DEA-2	DEA-3	DEA-4	DEA-5	DEA-6
ภาคเหนือ	0.002	0.003	0.003	0.003	0.003	0.003
ภาคตะวันออกเฉียงเหนือ	0.004	0.004	0.004	0.004	0.004	0.004
ภาคตะวันออก	0.002	0.002	0.003	0.003	0.003	0.003
ภาคกลาง	0.004	0.004	0.004	0.004	0.003	0.003
ภาคใต้และตะวันตก	0.004	0.004	0.004	0.004	0.004	0.004
ประเทศไทย	0.0086	0.0089	0.009	0.009	0.009	0.009

จากตารางที่ 4.1 เปรียบเทียบระยะทางรวมของ DEA Algorithms กับ NN นำมาซึ่งการคิดหาค่าเปอร์เซ็นต์การพัฒนาของ DEA algorithms จากผลลัพธ์ของ NN ในตารางที่ 4.2 พบว่า DEA-1 มีค่าเฉลี่ยในประสิทธิภาพของการปฏิบัติด้วยวิธีการ DEA-1 นั้นดีที่สุด รองลงมาเป็นอันดับ 2 คือ DEA-2 และ DEA-3 ในขณะที่ประสิทธิภาพในการคำนวณของ DEA Algorithm นั้นค่าเฉลี่ยไม่มีค่าต่างกันอย่างสิ้นเชิงตามตารางที่ 4.3

4.2 Stochastic Algorithm

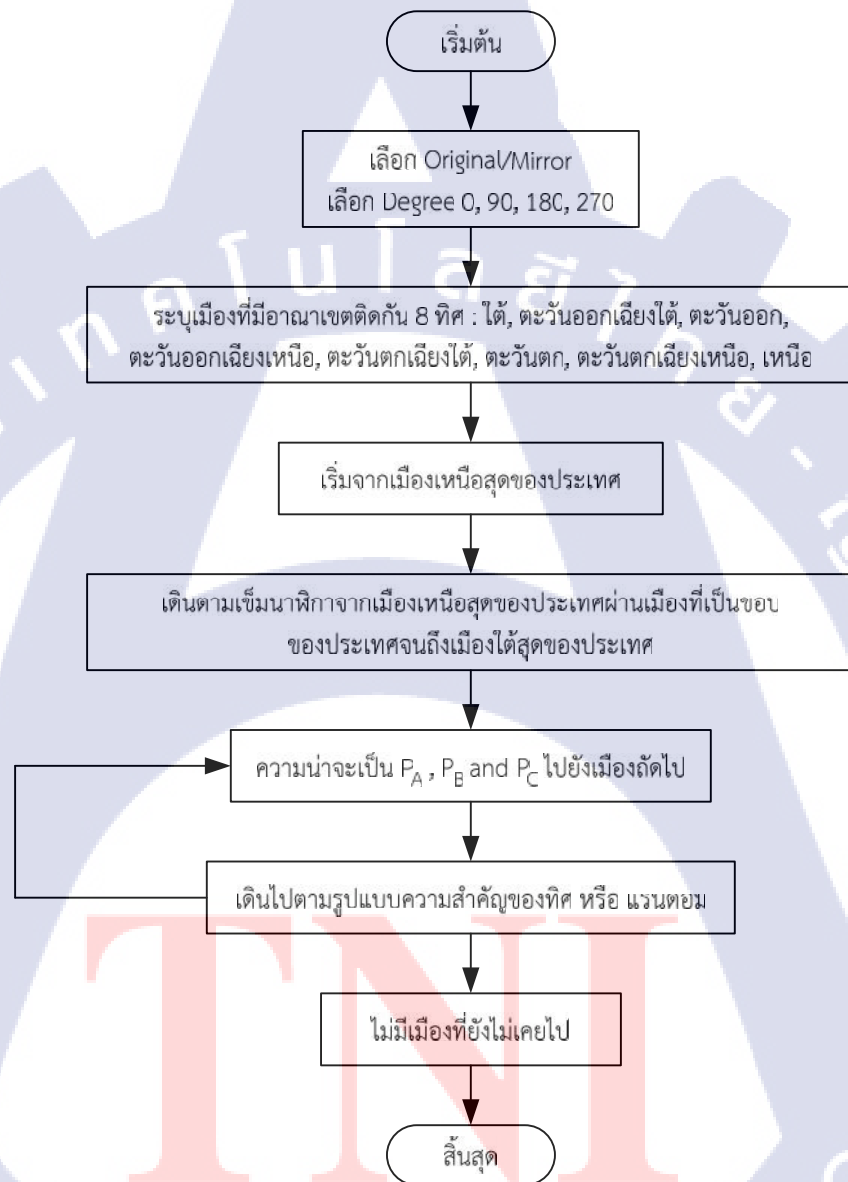
คำตอบที่ได้เสนอในงานวิจัยจากวิธีการ DEA นั้นเหมาะสมกับการใช้แก้ปัญหาสำหรับกลุ่มภาคการทดลอง แต่อาจจะไม่เหมาะสมสำหรับรูปแบบปัญหาอื่น ทำให้งานวิจัยในครั้งนี้จึงได้มีการพัฒนาและคิดวิธีการที่ดีและมีประสิทธิภาพที่สามารถใช้และแก้ไขปัญหสำหรับกลุ่มรูปแบบตัวอย่างการทดลองอื่นได้ โดยในงานวิจัยได้คิดวิธีการ Stochastic Algorithm (STA) ที่มีการดัดแปลงมาจากวิธีการจาก DEA โดยวิธีการของ STA จะมีความแตกต่างจากวิธีการ DEA ที่ Step 6

4.2.1 ขั้นตอนการเดินทางแบบ STA

Step6:เดินทางจากเมืองปัจจุบันไปยังเมืองที่มีอาณาเขตติดกันที่ยังไม่เคยไปมาก่อนที่ได้ระบุไว้ใน Step 3 ถ้ามีมากกว่า 1 เมืองที่ยังไม่เคยไป ให้เลือกไปยังเมืองถัดไปตามรูปแบบความสำคัญของทิศ (Set A, B, หรือ C) ที่มีรูปแบบความสำคัญของทิศมากที่สุดโดยเลือกจากความน่าจะเป็น P_A , P_B and P_C ซึ่งความน่าจะเป็นของ P_A , P_B and P_C คือความน่าจะเป็นที่จะเลือกไปยังรูปแบบความสำคัญของทิศรูปแบบใดรูปแบบหนึ่ง ในกรณีที่เมืองปัจจุบันไม่มีเมืองที่มีอาณาเขตติดกันที่เคยไปมาแล้ว ให้เลือกไปยังเมืองถัดไปที่ใกล้ที่สุด

จากการวิเคราะห์ผลลัพธ์ที่ได้ในวิธีการ DEA พบว่า DEA-1 มีประสิทธิภาพที่ดีสำหรับภาคตะวันออกเฉียงเหนือ ในขณะที่ DEA-2 และ DEA-3 มีประสิทธิภาพที่ดีสำหรับประเทศไทย และ NN มีประสิทธิภาพที่ดีสำหรับภาคเหนือ ดังนั้นในงานวิจัยได้มีการพัฒนาวิธีการ STA ขึ้นมาโดยใช้รูปแบบความสำคัญของทิศ DEA-1 คือ Set A, รูปแบบความสำคัญของทิศ DEA-2 คือ Set-B และการสุ่มคือ Set C นิยามของวิธีการสุ่มของ STA ในงานวิจัยคือทั้ง 8 ทิศมีลำดับความสำคัญเท่ากัน ดังนั้นจะสุ่มเพื่อเลือกไปยังเมืองถัดไปที่มีอาณาเขตติดต่อกับเมืองปัจจุบัน ซึ่งสาเหตุที่ไม่นำ DEA-3 มาประยุกต์ใช้ใน Set B สืบเนื่องมาจากผลลัพธ์ที่ได้จาก DEA-3 นั้นไม่มีความแตกต่างจาก DEA-2 ตามตารางที่ 4.2 โดยในการทดลองวิธีการ STA นั้นได้มีการปรับหาค่าความน่าจะเป็นที่เหมาะสมสำหรับ P_A , P_B and P_C โดยในวิธีการ STA ได้มีการเปรียบเทียบทุกความน่าจะเป็นของ $P_A \in \{20\%, 40\%, 60\%\}$, $P_B \in \{0\%, 20\%, 40\%, 60\%\}$, and $P_C = 100 - P_A - P_B$; เช่น STA-60-0-40 หมายความว่าใช้ P_A

= 60%, $P_B = 0\%$ and $P_C = 40\%$ โดยในงานวิจัยครั้งนี้แต่ละกลุ่มการทดลองมีการรันทั้งหมด 20 ครั้ง โดยค่าระยะทางรวมที่ดีที่สุดในการรันทั้งหมด 20 ครั้งของแต่ละกลุ่มการทดลองจะนำมาเปรียบเทียบกับค่าระยะทางรวมที่ดีที่สุดในการรันทั้งหมด 20 ครั้งของกลุ่มอื่น



รูปที่ 4.16 Flowchart STA

4.2.2 ผลการวิจัยและการวิเคราะห์ผล STA

ตารางที่ 4.4 ระยะทางรวมที่ดีที่สุดของ STA Algorithm

Probabilities	STA- ประเทศไทย	STA- ภาคกลาง	STA- ภาคตะวันออกเฉียงเหนือ	STA- ภาคเหนือ	STA- ภาคตะวันออก	STA- ภาคใต้และ ภาคตะวันตก
$P_A = 20, P_B = 20, P_C = 60$	10229 (13.3)	1731.3 (1.3)	2396.7 (7.5)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 20, P_B = 40, P_C = 40$	10269.3 (13.0)	1677.9 (4.3)	2405 (7.2)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 20, P_B = 60, P_C = 20$	10036.5 (15.0)	1693.6 (3.4)	2518.3 (2.8)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 40, P_B = 0, P_C = 60$	10524.2 (10.8)	1693.6 (3.4)	2387.8 (7.9)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 40, P_B = 20, P_C = 20$	10262.4 (13.1)	1693.6 (3.4)	2387.8 (7.9)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 40, P_B = 40, P_C = 20$	10357.6 (12.3)	1677.9 (4.3)	2405 (7.2)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 40, P_B = 60, P_C = 0$	10287.9 (12.9)	1731.3 (1.3)	2446.8 (5.6)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 60, P_B = 0, P_C = 40$	10224.4 (13.4)	1677.9 (4.3)	2345.9 (9.5)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 60, P_B = 20, P_C = 20$	10033.4 (15.0)	1703.2 (2.9)	2387.8 (7.9)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 60, P_B = 40, P_C = 0$	10650.7 (9.8)	1731.3 (1.3)	2446.8 (5.6)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 20, P_B = 0, P_C = 80$	10405.6 (11.9)	1788.4(-2.0)	2405(7.2)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)
$P_A = 60, P_B = 10, P_C = 30$	10492(11.1)	1718.5(2.0)	2446.8(5.6)	1487.1 (-3.0)	743.6 (6.7)	3728.8 (11.7)

ตารางที่ 4.5 CPU Time (หน่วย : วินาที) ของ STA Algorithm

Probabilities	STA- ประเทศไทย	STA- ภาคกลาง	STA- ภาคตะวันออกเฉียงเหนือ	STA- ภาคเหนือ	STA- ภาคตะวันออก	STA- ภาคใต้และ ภาคตะวันตก
$P_A = 20, P_B = 20, P_C = 60$	0.14	0.12	0.13	0.03	0.03	0.05
$P_A = 20, P_B = 40, P_C = 40$	0.31	0.12	0.13	0.03	0.03	0.05
$P_A = 20, P_B = 60, P_C = 20$	0.13	0.13	0.13	0.03	0.04	0.04
$P_A = 40, P_B = 0, P_C = 60$	0.14	0.12	0.13	0.03	0.03	0.04
$P_A = 40, P_B = 20, P_C = 20$	0.12	0.12	0.13	0.03	0.03	0.05
$P_A = 40, P_B = 40, P_C = 20$	0.16	0.12	0.13	0.03	0.03	0.04
$P_A = 40, P_B = 60, P_C = 0$	0.18	0.12	0.13	0.03	0.03	0.05
$P_A = 60, P_B = 0, P_C = 40$	0.14	0.12	0.13	0.03	0.04	0.05
$P_A = 60, P_B = 20, P_C = 20$	0.13	0.13	0.13	0.03	0.04	0.04
$P_A = 60, P_B = 40, P_C = 0$	0.17	0.12	0.13	0.03	0.04	0.04
$P_A = 20, P_B = 00, P_C = 80$	0.14	0.12	0.13	0.03	0.03	0.05
$P_A = 60, P_B = 10, P_C = 30$	0.14	0.12	0.13	0.03	0.04	0.05

จากตารางที่ 4.4 ได้มีการวิเคราะห์ผลลัพธ์ที่ได้ด้วยการคิดคำนวณค่าเปอร์เซ็นต์การพัฒนา คำตอบซึ่งแสดงอยู่ในวงเล็บของแต่ละคำตอบ และพบว่าการพัฒนาของคำตอบด้วยวิธีการ STA จาก คำตอบโดย NN พบว่า probabilities $P_A = 60$, $P_B = 0$, $P_C = 40$ มีสมรรถภาพที่ดีเป็นอันดับแรก และ $P_A = 60$, $P_B = 20$, $P_C = 20$ มีสมรรถภาพที่ดีเป็นอันดับ 2 ลงมาบนพื้นฐานของการปรับระดับ ค่า probabilities นอกจากนี้ประสิทธิภาพของการรัน CPU Time ของ STA Algorithm ตามตาราง ที่ 4.5 นั้นพบว่าการประมวลผลค่า CPU Time มีค่าไม่แตกต่างกันมาก

4.3 Local Search

การประยุกต์ใช้วิธีการของ Local Search ในงานวิจัยนั้นเป็นการพัฒนาวิธีการ DEA ในการปรับปรุงเส้นทางและระยะทางรวมให้ดีขึ้น โดยเป็นการสลับปรับเปลี่ยนตำแหน่งของเส้นทางที่ พนักงานขายของไปแต่ละเมืองจนครบแล้วกลับมาที่เดิม โดยในงานวิจัยได้เลือกใช้วิธีการ Swapping และการ Insert Local Search โดยมีหลักการดังนี้

4.3.1 วิธีการ LSS

โดย Logic ที่ใช้ในการ SWAP คือการ Random เมืองทดสอบมาหนึ่ง เมือง จากนั้น ตรวจสอบว่ามีเมืองใดบ้างติดกับเมืองทดสอบ จากนั้นทำการสลับเมืองทดสอบไปติดกับเมืองที่ติดกัน จากนั้นเปรียบเทียบระยะทางรวมใหม่กับระยะทางรวมปัจจุบัน หากระยะทางรวมใหม่มีค่าน้อยกว่า ระยะทางรวมปัจจุบัน ก็จะใช้ระยะทางรวมใหม่เป็นตัวเริ่มต้นเปรียบเทียบต่อไปตาม Flow Chart รูปที่ 4.17 โดยในงานวิจัยได้มีการทดลองและเก็บผลจำนวน 20 รอบการรัน

ตัวอย่างการ LSS

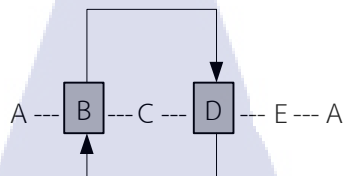
เส้นทางคำตอบที่ได้ปัจจุบันจากวิธีการแบบที่ 1

A --- B --- C --- D --- E --- A (15 km.)

ทำการ Random มา 1 เมือง ยกเว้นเมืองเริ่มต้น (A) และเมืองสุดท้าย (A) ผ่านโปรแกรม ในตัวอย่างสมมติ Random ได้เมือง B

A --- B --- C --- D --- E --- A

ทำการตรวจสอบว่าเมือง B ติดกับเมืองอะไรบ้าง ปรากฏว่า B ติดกับเมือง C (เส้นทางใน ปัจจุบัน) และ E ก็จะทำให้การ Swap เมือง B ไปติดกับ E จะได้เส้นทางตามข้างล่างดังนี้



A --- D --- C --- B --- E --- A (13 km.)

จากนั้นทำการเปรียบเทียบระยะทางรวมของเส้นทางปัจจุบันและเส้นทางใหม่ที่ได้

เส้นทางปัจจุบัน A --- B --- C --- D --- E --- A (15 km.)

เส้นทางใหม่ A --- D --- C --- B --- E --- A (13 km.)

หากเส้นทางใหม่มีระยะทางรวมดีกว่า ก็จะนำเส้นทางใหม่นี้มาเป็นเส้นทางเริ่มต้น แล้วทำการ Swap ต่อไป หากเส้นทางปัจจุบันมีระยะทางรวมดีกว่าก็จะยังใช้เส้นทางปัจจุบันเป็นตัวเริ่มต้น และทำการ Swap และเปรียบเทียบไปเรื่อยๆ จนกว่าจะครบจำนวนรอบตามที่กำหนด

4.3.2 ขั้นตอนของวิธีการ LSS

Step1: ในแต่ละเมือง กำหนดเมืองที่ติดต่อกัน 8 ทิศ ได้แก่, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ โดยมีขอบเขตของการกำหนดทิศว่าในการระบุทิศของแต่ละเมือง คือจะดูจากพื้นที่อาณาเขตติดต่อกับเมืองนั้น มีอาณาเขตติดต่อกันมากที่สุดทิศใด และในแต่ละทิศสามารถมีได้แค่เมืองเดียว และในแต่ละเมืองสามารถมีทิศได้แค่ทิศเดียวเช่นกัน ถ้ามีมากกว่า 1 เมืองในแต่ละทิศ ให้เลือกเมืองที่มีอาณาเขตในทิศนั้นมากที่สุด

Step2: ใช้วิธีการ DEA Algorithm ในการผลิต Initial Solution S_0 และ Solution Permutation P_0

Step3: ให้ $t = 0$

Step4: สุ่มเลือกเมืองใดเมืองหนึ่ง เรียกว่า เมือง u จากนั้น Random เมืองที่มีอาณาเขตติดกับเมือง u เรียกว่าเมือง v ; เส้นทาง P_0 จากนั้นทำการเปลี่ยนแปลงเส้นทาง P_1 โดยสลับเมือง u ไปแทนที่เมืองที่ติดกับ v โดยจะได้เส้นทางใหม่จาก P_1 คือ S_1

Step5: ทำการเปรียบเทียบระหว่าง S_0 และ S_1 ถ้า S_1 มีค่าระยะทางรวมน้อยกว่า หรือ เท่ากับ S_0 ให้ทำการสลับค่า S_1 ไปแทนที่ S_0 ในทางเดียวกันให้เปลี่ยน P_0 เท่ากับ P_1 จากนั้นทำซ้ำ Step3 และเพิ่มค่า t ที่ละ 1

Step6: ถ้า $t = 8N$ ให้หยุด ถ้าไม่ก็ทำซ้ำ Step4

4.3.3 วิธีการ LSI

โดย Logic ที่ใช้ในการ INSERT คือการ Random เมืองทดสอบมาหนึ่ง เมือง จากนั้นตรวจสอบว่ามีเมืองใดบ้างติดกับเมืองทดสอบ จากนั้นทำการแทรกเมืองทดสอบไปอยู่ข้างหน้าเมืองที่ติดกัน จากนั้นเปรียบเทียบระยะทางรวมใหม่กับระยะทางรวมปัจจุบัน หากระยะทางรวมใหม่มีค่าน้อยกว่าระยะทางรวมปัจจุบัน ก็จะใช้ระยะทางรวมใหม่เป็นตัวเริ่มต้นเปรียบเทียบต่อไป โดยในงานวิจัยได้มีการทดลองและเก็บผลจำนวน 20 รอบการรัน

ตัวอย่างการ LSI

เส้นทางคำตอบที่ได้ปัจจุบันจากวิธีการแบบที่ 1

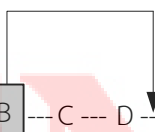
A --- B --- C --- D --- E --- A (15 km.)

ทำการ Random มา 1 เมือง ยกเว้นเมืองเริ่มต้น (A) และ เมืองสุดท้าย (A) ผ่านโปรแกรมในตัวอย่างสมมติ Random ได้เมือง B

A --- B --- C --- D --- E --- A

ทำการตรวจสอบว่าเมือง B ติดกับเมืองอะไรบ้าง ปรากฏว่า B ติดกับเมือง C (เส้นทางในปัจจุบัน) และ E ก็จะมีการ Insert เมือง B ไปอยู่ข้างหน้าติดกับ E จะได้เส้นทางตามข้างล่างดังนี้

A --- B --- C --- D --- E --- A



A --- C --- D --- B --- E --- A (13 km.)

จากนั้นทำการเปรียบเทียบระยะทางรวมของเส้นทางปัจจุบันและเส้นทางใหม่ที่ได้

เส้นทางปัจจุบัน A --- B --- C --- D --- E --- A (15 km.)

เส้นทางใหม่ A --- C --- D --- B --- E --- A (13 km.)

หากเส้นทางใหม่มีระยะทางรวมดีกว่า ก็จะนำเส้นทางใหม่นี้มาเป็นเส้นทางเริ่มต้น แล้วทำการ Insert ต่อไป หากเส้นทางปัจจุบันมีระยะทางรวมดีกว่าก็จะยังใช้เส้นทางปัจจุบันเป็นตัวเริ่มต้น และทำการ Insert และเปรียบเทียบไปเรื่อยๆ จนกว่าจะครบจำนวนรอบตามที่กำหนด

4.3.4 ขั้นตอนของวิธีการ LSI

Step1: ในแต่ละเมือง กำหนดเมืองที่ติดต่อกันทั้ง 8 ทิศ ได้แก่, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ โดยมีขอบเขตของการกำหนดทิศว่าในการระบุทิศของแต่ละเมือง คือจะดูจากพื้นที่อาณาเขตติดต่อกับเมืองนั้น มีอาณาเขตติดต่อกันมากที่สุดที่ทิศใด และในแต่ละทิศสามารถมีได้แค่เมืองเดียว และในแต่ละเมืองสามารถมีทิศได้แค่ทิศเดียวเช่นกัน ถ้ามีมากกว่า 1 เมือง ในแต่ละทิศ ให้เลือกเมืองที่มีอาณาเขตในทิศนั้นมากที่สุด

Step2: ใช้วิธีการ DEA Algorithm ในการผลิต Initial Solution S_0 และ Solution Permutation P_0

Step3: ให้ $t = 0$

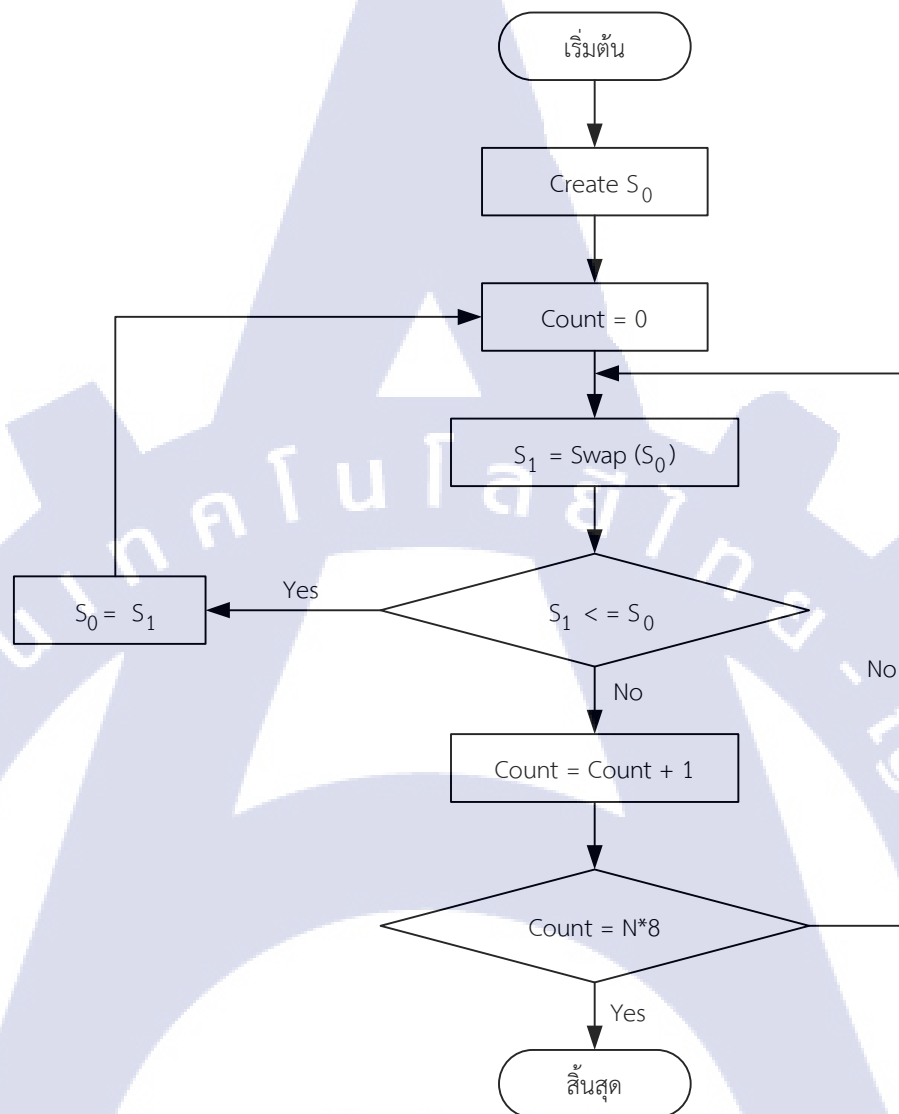
Step4: สุ่มเลือกเมืองใดเมืองหนึ่ง เรียกว่า เมือง u จากนั้นทำการสุ่มเมืองที่อาณาเขตติดกับเมือง u เรียกว่า เมือง v จากนั้นทำการเปลี่ยนแปลงตำแหน่งเมืองเพื่อให้ได้เส้นทาง P_1 โดยการลบเมือง u จากเส้นทางการเปลี่ยนแปลงและแทรกเมือง u ไปข้างหน้าเมือง v โดยจะได้เส้นทางใหม่จาก P_1 คือ S_1

Step5: ทำการเปรียบเทียบระหว่าง S_0 และ S_1 ถ้า S_1 มีค่าระยะทางรวมน้อยกว่า หรือ เท่ากับ S_0 ให้ทำการสลับค่า S_1 ไปแทนที่ S_0 ในทางเดียวกันให้เปลี่ยน P_0 เท่ากับ P_1 จากนั้นทำซ้ำ Step3 และเพิ่มค่า t ที่ละ 1

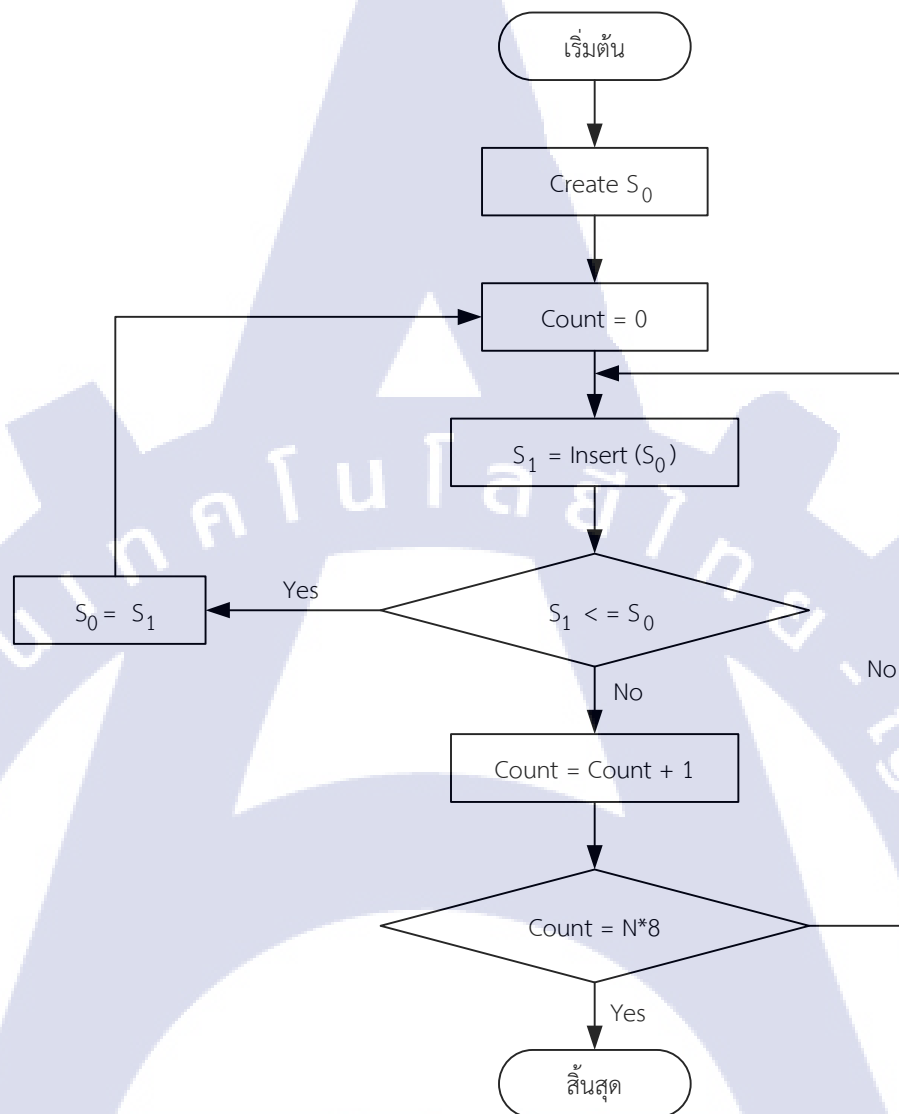
Step6: ถ้า $t = 8N$ ให้หยุด ถ้าไม่ก็ทำซ้ำ Step4

TNI

THAI-NICHI INSTITUTE OF TECHNOLOGY



รูปที่ 4.17 หลักการการปรับปรุงเส้นทางแบบ SWAP



รูปที่ 4.18 หลักการการปรับปรุงเส้นทางแบบ INSERT

4.3.4 ผลการวิจัยและการวิเคราะห์ผล LSS และ LSI

จากการวิจัยในบทที่ 3 ที่ได้กล่าวมานั้นพบว่า DEA-1 และ DEA-2 เป็นวิธีการที่มีสมรรถภาพที่ดีอันดับ 1 และ 2 ตามลำดับ เพราะฉะนั้นในงานวิจัยของวิธีการ LSS และ LSI จึงได้นำค่าที่ดีที่สุดของแต่ละกลุ่มภาคการทดลองมาเป็น S_0 ค่าเริ่มต้นของการทำ Local Search โดยวิธีการ LSS algorithm ใช้ DEA-1 เป็นคำตอบเริ่มต้นใน Step 2 ชื่อว่า LSS-1 และวิธีการ LSS Algorithm ใช้ DEA-2 เป็นคำตอบเริ่มต้นชื่อว่า LSS-2 ส่วน LSI-1 คือ LSI Algorithm ที่ใช้ DEA-1 ใน Step 2 และ LSI-2 ใช้ DEA-2 โดยในแต่ละวิธีการเปรียบเทียบค่าเส้นทางที่ดีที่สุดจากการรัน 20 ครั้งโดย LSS-1, LSS-2, LSI-1, และ LSI-2 ซึ่งในตารางที่ 4.6 ในวงเล็บคือค่าเปอร์เซ็นต์การพัฒนามลค่าตอบที่ได้จากคำตอบด้วยวิธีการ LSS-1, LSS-2, LSI-1, และ LSI-2 กับคำตอบจาก NN และในงานวิจัยมีการเปรียบเทียบ CPU Time ของแต่ละวิธีการ LSS-1, LSS-2, LSI-1, และ LSI-2 ตามตารางที่ 4.7

ตารางที่ 4.6 เปรียบเทียบระยะทางรวมของ LSS และ LSI Algorithms

Instance	LSS-1	LSS-2	LSI-1	LSI-2
ภาคเหนือ	1487.1 (-3.0)	1487.1 (-3.0)	1443.5 (0.0)	1443.5 (0.0)
ภาคตะวันออกเฉียงเหนือ	2283.8 (11.9)	2491.0 (3.9)	2283.8 (11.9)	2311.1 (10.8)
ภาคตะวันออก	743.6 (6.7)	743.6 (6.7)	743.6 (6.7)	743.6 (6.7)
ภาคกลาง	1646.8 (6.1)	1646.8 (6.1)	1550.5 (11.6)	1550.5 (11.6)
ภาคใต้และตะวันตก	3728.5 (11.7)	3728.8 (11.7)	3714.1 (12.1)	3714.1 (12.1)
ประเทศไทย	9823.5 (16.8)	10244.9 (13.2)	8746.3 (25.9)	8980.2 (23.9)

ตารางที่ 4.7 เปรียบเทียบ CPU Time (หน่วย : วินาที) LSS และ LSI Algorithms

Instance	LSS-1 - Time	LSS-2 - Time	LSI-1 - Time	LSI-2 - Time
ภาคเหนือ	0.08	0.08	0.09	0.08
ภาคตะวันออกเฉียงเหนือ	0.44	0.46	0.37	0.49
ภาคตะวันออก	0.07	0.07	0.07	0.07
ภาคกลาง	0.45	0.48	0.45	0.46
ภาคใต้และตะวันตก	0.28	0.28	0.26	0.26
ประเทศไทย	7.44	6.32	7.93	9.00

จากตารางที่ 4.6 จะเห็นว่าเปอร์เซ็นต์การพัฒนาของวิธีการ LSI-1 มีสมรรถภาพดีที่สุดในการเปรียบเทียบกับวิธีการอื่นในทั้งหมดของวิธีการ Local Search ในส่วนของการประมวลผลการรัน CPU Time จากตารางที่ 4.7 นั้นการใช้เวลาในการประมวลผลนั้นไม่มีค่าความแตกต่างกันมากของในแต่ละวิธี เช่น LSI-1 สำหรับภาคเหนือใช้ 0.09 วินาที, ภาคตะวันออกเฉียงเหนือใช้ 0.37 วินาที, ภาคตะวันออก 0.07 วินาที, ภาคกลางใช้ 0.45 วินาที, ภาคใต้และตะวันตกใช้ 0.26 วินาที และประเทศไทยใช้ 9.00 วินาที

ตารางที่ 4.8 เปรียบเทียบระยะทางที่ดีที่สุดของแต่ละวิธีการ (กิโลเมตร)

Instance	NN	AAA	DEA-1	STA	LSI-1
ภาคเหนือ	1443.5	1487.1	1487.1	1487.1	1443.5
ภาคตะวันออกเฉียงเหนือ	2592.0	2446.9	2446.9	2345.9	2283.8
ภาคตะวันออก	797.0	743.6	743.6	743.6	743.6
ภาคกลาง	1754.1	1731.3	1731.3	1677.9	1550.5
ภาคใต้และตะวันตก	4224.3	3728.8	3728.8	3728.8	3714.1
ประเทศไทย	11805.3	10650.6	10650.6	10224.1	8746.3

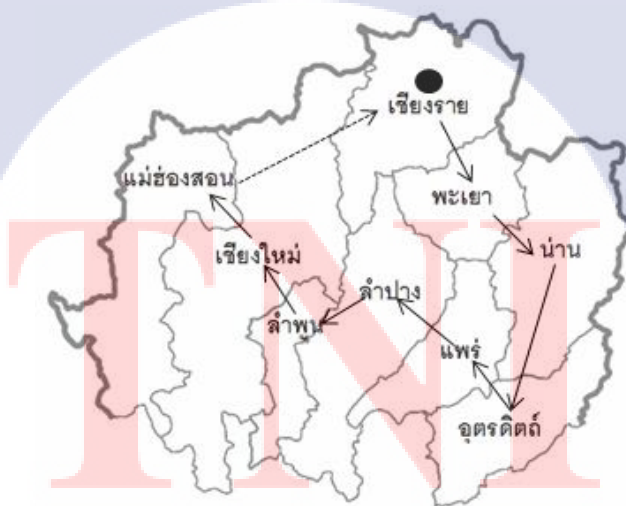
ตารางที่ 4.9 เปรียบเทียบค่าร้อยละเฉลี่ยและค่าความแตกต่างของระยะทางรวมระหว่าง NN และ LSI-1 (กิโลเมตร)

Instance	NN	LSI-1	ระยะทางรวม ลดลง	ค่าร้อยละ เฉลี่ย
ภาคเหนือ	1443.5 (0.0065)	1443.5 (0.09)	0	0.0%
ภาคตะวันออกเฉียงเหนือ	2592.0 (0.0088)	2283.8 (0.37)	308.2	11.9%
ภาคตะวันออก	797.0 (0.0063)	743.6 (0.07)	53.4	6.7%
ภาคกลาง	1754.1 (0.0077)	1550.5 (0.45)	203.6	11.6%
ภาคใต้และตะวันตก	4224.3 (0.0085)	3714.1 (0.26)	510.2	12.1%
ประเทศไทย	11805.3 (0.0133)	8746.3 (7.93)	3059	25.9%

4.4 วิเคราะห์ผล

จากตารางที่ 4.8 จะเห็นได้ว่าเมื่อมีการปรับปรุงด้วยวิธีการในงานวิจัย วิธีการ LSI-1 นั้นมีประสิทธิภาพที่ดีที่สุดสำหรับการปรับปรุงระยะทางรวมของแต่ละกลุ่มการทดลอง โดยวิธีการ LSI-1 คือวิธีการของ Local Search แบบ Insert โดยมีลำดับความสำคัญของทิศเป็นแบบ DEA-1 (ใต้, ตะวันออกเฉียงใต้, ตะวันออก, ตะวันออกเฉียงเหนือ, ตะวันตกเฉียงใต้, ตะวันตก, ตะวันตกเฉียงเหนือ และเหนือ) ซึ่งมีค่าร้อยละเฉลี่ยและค่าความแตกต่างของระยะทางรวมระหว่าง NN และ LSI-1 ตามตารางที่ 4.9 ในขณะที่การคำนวณเวลาของ CPU Time ของวิธีการ LSI-1 รัน 20 ครั้ง เทียบกับวิธีการ NN พบว่าวิธีการ NN มีการคำนวณเวลา CPU Time ที่เร็วกว่า LSI-1 ตามตารางที่ 4.9 และมีรูปแบบเส้นทางการเดินทางที่ดีที่สุดตามรูปที่ 4.19-4.24

จากผลการทดลองและวิเคราะห์กลุ่มการทดลองพบว่า รูปแบบหรือลักษณะของแผนที่ภาพที่มีลักษณะมีเมืองที่มีอาณาเขตติดกันตามแนวขอบของประเทศทอดยาวลงมาจากเมืองเหนือสุดของประเทศจนถึงเมืองต่ำสุดของประเทศ จะมีความเหมาะสมสำหรับการเลือกใช้วิธีการ AAA, DEA, STA, และ Local Search ในการพัฒนาปรับปรุงระยะทางให้มีค่าระยะทางรวมลดลงตามลำดับและมีประสิทธิภาพที่ดีหากเทียบกับวิธีการ NN แต่ถ้าหากประเทศที่มีกลุ่มเมืองที่มีเมืองอาณาเขตติดต่อกันเป็นพื้นที่มากทำให้ระยะทางจากเมืองหนึ่งไปยังอีกเมืองมีระยะทางที่มาก อาจมีผลต่อวิธีการในงานวิจัยที่มีการกำหนดความสำคัญของทิศเป็นตัวแปรสำคัญในการเดินทางไปยังเมืองถัดไป



รูปที่ 4.19 เส้นทางดีที่สุดของการเดินทางภาคเหนือแบบวิธี LSI-1



รูปที่ 4.20 เส้นทางที่ดีที่สุดของการเดินทางภาคตะวันออกเฉียงเหนือแบบวิธี LSI-1

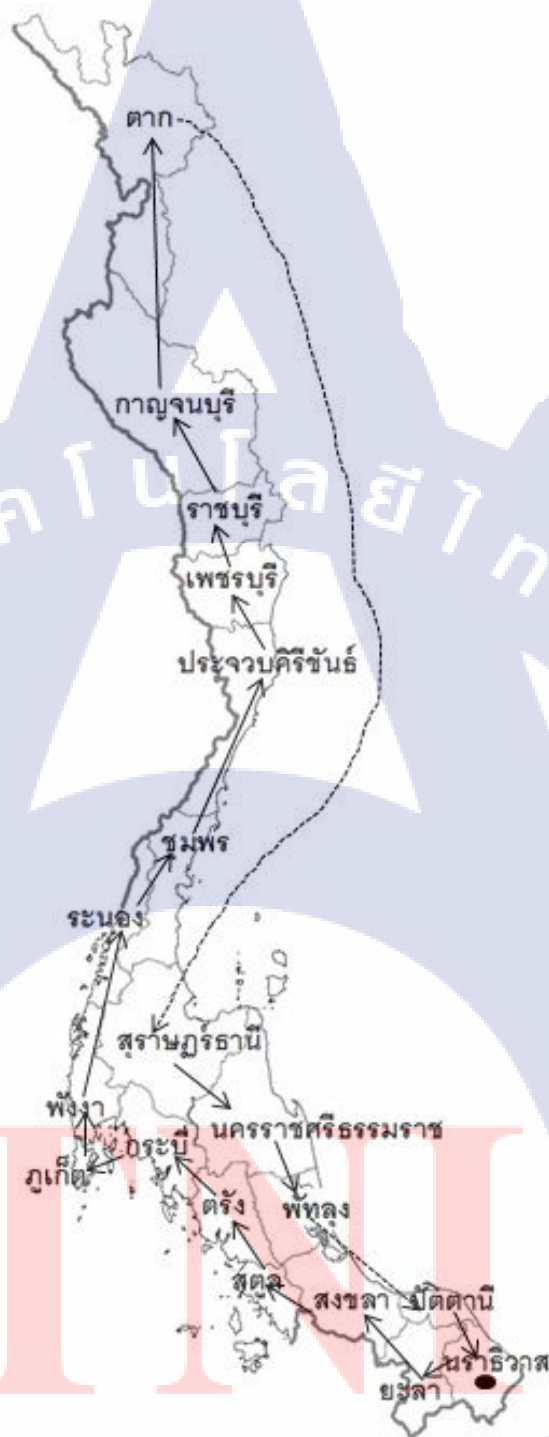


รูปที่ 4.21 เส้นทางที่ดีที่สุดของการเดินทางภาคตะวันออกเฉียงเหนือแบบวิธี LSI-1

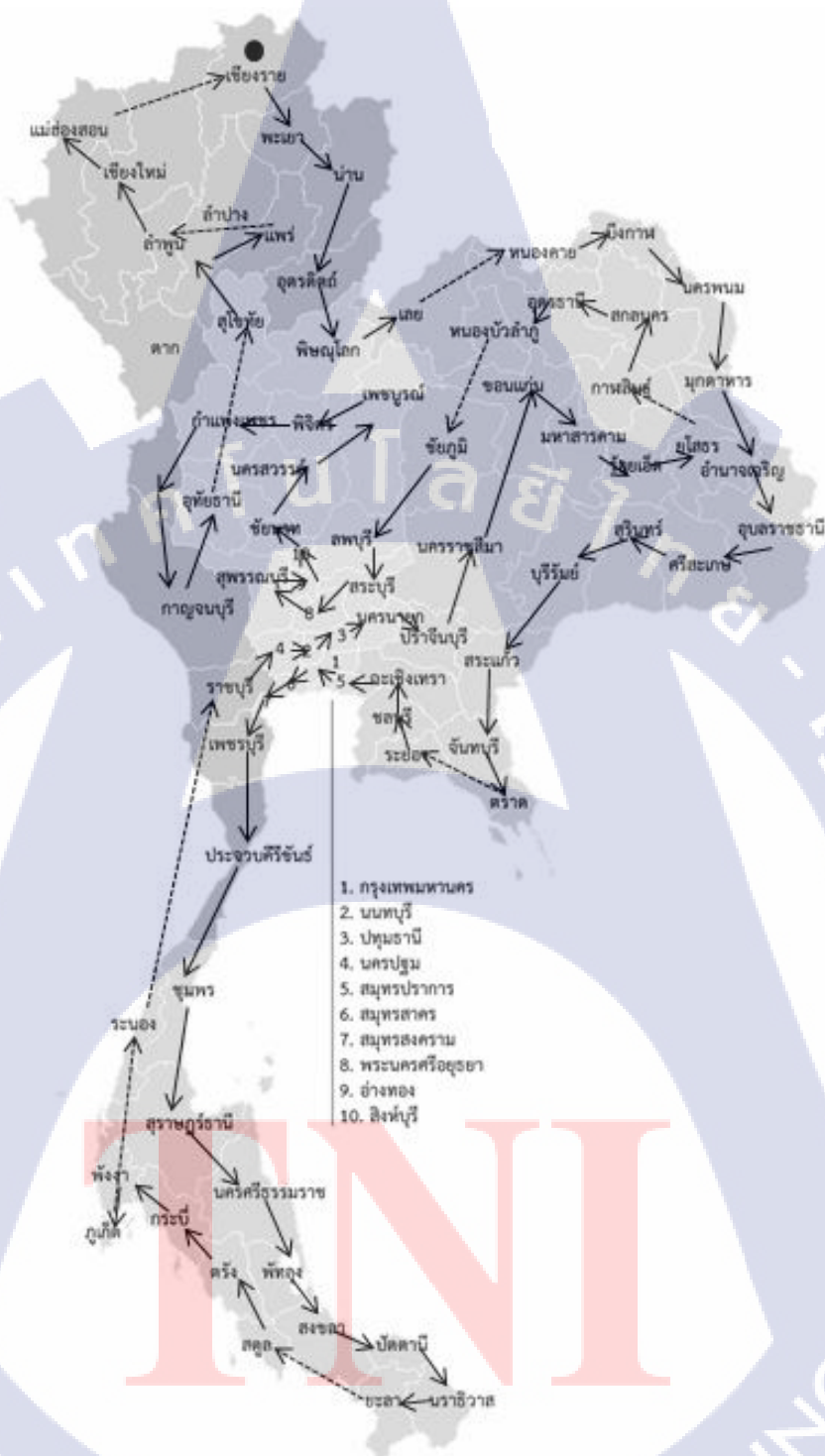


รูปที่ 4.22 เส้นทางที่ดีที่สุดของการเดินทางภาคกลางแบบวิธี LSI-1

TNI



รูปที่ 4.23 เส้นทางที่ดีที่สุดของการเดินทางภาคใต้และตะวันตกแบบวิธี LSI-1



รูปที่ 4.24 เส้นทางที่ดีที่สุดของการเดินทางประเทศไทยแบบวิธี LSI-1

บทที่ 5

สรุปผลการวิจัย

5.1 สรุปผลการวิจัย

งานวิจัยได้มีการเปรียบเทียบความแตกต่างระหว่างค่าเฉลี่ยระยะทางรวมที่ดีที่สุดในการรัน 20 ครั้งของวิธีการ LSI-1 ซึ่งเป็นวิธีการของ Local Search แบบ Insert ด้วยลำดับความสำคัญของทิศแบบ DEA1 (LSI-1) เป็นวิธีในงานวิจัยที่ได้ระยะทางรวมน้อยที่สุด และค่าเฉลี่ยระยะทางรวมที่ดีที่สุดในการรัน 20 ครั้งของวิธีการ STA-60-0-40 ซึ่งเป็นค่าที่ดีที่สุดของ STA และเมื่อทำการทดสอบ Paired T-test โดยตั้งสมมติฐานว่า ผลของค่าร้อยละความต่างจากค่าเฉลี่ยระยะทางรวมที่ดีที่สุดของวิธีการ LSI-1 ที่มีการปรับปรุงมาจากวิธีการ NN มีค่าร้อยละความต่างที่มีประสิทธิภาพเท่ากับผลของค่าร้อยละความต่างจากค่าเฉลี่ยระยะทางรวมที่ดีที่สุดของวิธีการ STA-60-0-40 ที่มีการปรับปรุงมาจากวิธีการ NN ที่ระดับความเชื่อมั่น 0.05 พบว่าค่า P-Value เท่ากับ 0.034 แสดงว่าผลของค่าเฉลี่ยระยะทางรวมที่ดีที่สุดของวิธีการ LSI-1 มีประสิทธิภาพที่ดีกว่าผลของค่าเฉลี่ยระยะทางรวมที่ดีที่สุดของวิธีการ STA-60-0-40 สำหรับประสิทธิภาพในการคำนวณเวลาของ CPU Time ระหว่างการปรับปรุงเส้นทางด้วยวิธี LSI-1 ซึ่งเป็นวิธีที่ดีที่สุดในการปรับปรุงระยะทางรวมลดลง โดยวิธีการของ LSI-1 มีค่าเฉลี่ยของเวลาการคำนวณเป็น 1.528 วินาที ในขณะที่วิธีการของ STA-60-0-40 ซึ่งเป็นวิธีการที่มีระยะทางรวมดีที่สุดของ STA มีค่าเฉลี่ยของเวลาการคำนวณเป็น 0.085 วินาที ส่วนวิธีการของ AAA และ วิธีการ DEA-1 ซึ่งเป็นวิธีการที่มีระยะทางรวมดีที่สุดของ DEA มีค่าเฉลี่ยของเวลาการคำนวณเป็น 0.004 วินาทีและวิธีการ Nearest Neighbor (NN) พบว่า มีค่าเฉลี่ยของเวลาการคำนวณเป็น 0.009 วินาที

5.2 ข้อเสนอแนะ

จากงานวิจัยการปรับปรุงระยะทางรวมด้วยวิธีการ Local Search ในรูปแบบ Insert และ Swap ซึ่งอาจมีค่าระยะทางรวมลดลงที่ต่ำกว่านี้หากมีวิธีการในการแทรกเมืองหรือการสลับเมืองที่แตกต่างกันไป และเพิ่มการวิเคราะห์เปรียบเทียบการปรับปรุงระยะทางด้วยวิธีการ Insert และ Swap



บรรณานุกรม

TNI

บรรณานุกรม

- [1] M. R. Garey and D. S. Johnson, *Computers and Intractability : a Guide to the Theory of NP-Completeness*. New York : Macmillan Higher Education, Chapter 2, 1979.
- [2] H. Calgor. “The Brute Force Algorithm”. [Online]. Available: [http://www.ctl.ua.edu/math103/hamilton/HCalgor.htm#THE BRUTE FORCE ALGORITHM](http://www.ctl.ua.edu/math103/hamilton/HCalgor.htm#THE%20BRUTE%20FORCE%20ALGORITHM), August 1, 2013.
- [3] N. Ascheuer, M. Fischetti, and M. Grotschel, “Solving the Asymmetric Travelling Salesman Problem with Time Windows by Branch-and-Cut.” *Mathematical Programming*, vol. 90, no.3, pp. 475-506, June 2001.
- [4] B. A. Alsaalibi, M. B. Jelodar and I. Venkat, “A Comparative Study Between the Nearestneighbor and Genetic Algorithms: a Revisit to the Traveling Salesman Problem.” *International Journal of Computer Science and Electronics Engineering*, vol. 1, no.1, pp. 34-38, May 2013.
- [5] M.R Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Francisco: W.H. Freeman, 1979.
- [6] D. J. Rosenkrantz, R. R. Stearns and P. M. Lewis, “An Analysis of Several Heuristics for the Traveling Salesman Problem.” *SIAM Journal on Computing*, vol. 6, no.3, pp. 563-581, May 1977.
- [7] Lin S., “Computer Solutions of the Traveling Salesman Problem.” *Bell System Technical Journal*, vol. 44, no.1-3, pp. 2245-2269, January 1965.
- [8] Lin S., and Kernighan B.W., “An Effective Heuristic Algorithm for the Traveling-Salesman Problem.” *Oper*, vol. 113 no.21, pp. 498-516, January 1973.
- [9] J. Majumdar and A. K. Bhunia, “Genetic Algorithm for Asymmetric Traveling Salesman Problem with Imprecise Travel Times.” *Journal of Computational and Applied Mathematics*, vol. 235, no.9, pp. 3063-3078, August 2011.
- [10] S. Basu, “Tabu Search Implementation on Traveling Salesman Problem and its Variations: A Literature Survey,” *American Journal of Operations Research*, India, pp. 163-173, June 2012.

- [11] I. Piriyaniti and P. Pongchairerks, "Variable Neighbourhood Search Algorithms for Asymmetric Travelling Salesman Problem," *International Journal of Operational Research*, vol. 18, no.2, pp. 157-170, April 2013.
- [12] Z. C. S. S. Hlaing and M. A. Khine, "Solving Traveling Salesman Problem by Using Improved Ant Colony Optimization Algorithm," *International Journal of Information and Education Technology*, vol. 1, no.5, pp. 404-409, December 2011.
- [13] P. Putponpai and P. Pongchairerks, "A Problem-Based Heuristic for Asymmetric Travelling Salesman Problems," *Journal of Industrial and Intelligent Information*, vol. 3, No.3, September 2015.



THAI-NICHI INSTITUTE OF TECHNOLOGY

TNI