การหาค่าดีที่สุด ในการกระจายสินค้าด้วยรถยนต์ไฟฟ้า

Freight Distribution Optimization Using Electric Vehicles

ชลวิทย์ พูนศักดิ์อุดมสิน[1], ชวิศ ปรีชาติวงศ์[2] และ รศ.ดร.มาโนช[3]

[1,2,3]ภาควิชาวิศวกรรมโยธา คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย จ. กรุงเทพฯ 10330

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อศึกษา ออกแบบ และพัฒนาโปรแกรมในการจัดเส้นทางกระจายสินค้าภายใต้ข้อกำหนดเช่น ข้อกำหนดระยะทาง ปริมาณสินค้าที่ขนส่งได้ในหนึ่งรอบการเดินรถ กรอบเวลา และ จำนวนจุดหมายปลายทาง โดยการเขียนโปรแกรมด้วยภาษาไพทอน ซึ่งเป็นภาษาที่มีชุดคำสั่งที่สามารถทำความเข้าใจได้ง่าย กระชับ สามารถทำงานบนระบบปฏิบัติการได้หลากหลาย และมี library มากมายที่มีประสิทธิภาพสูง และสามารถใช้งานได้ง่าย โดยโปรแกรมนี้มีการใช้ Algorithm ในการจัดเส้นทางกระจายสินค้าที่สามารถทำงานได้อย่างถูกต้อง รวดเร็วและประหยัดทรัพยากร นอกจากนี้ยังมีการดึงเอาข้อมูลจากฐานข้อมูลแผนที่มาใช้ร่วมด้วย จากคุณสมบัติของโปรแกรมที่กล่าวไว้ข้างต้นทำให้การจัดเส้นทางการกระจายสินค้าด้วยโปรแกรมนี้ สามารถลดจำนวนรถที่ใช้ขนส่งลง ลดระยะเวลาและระยะทางในการเดินทาง รวมถึงลดต้นทุนในการขนส่งกระจายสินค้า และ ทำให้การขนส่งนั้นมีประสิทธิภาพมากขึ้นได้ โดยภายในรายงานฉบับนี้จะมีการเปรียบเทียบผลลัพธ์เพื่อให้สามารถมั่นใจในประสิทธิภาพของโปรแกรมได้

คำสำคัญ: การจัดเส้นทางกระจายสินค้า; ภาษาไพทอน; Algorithm

Abstract

This paper presents the research aims to analyze, design, and develop a software program for routing and distributing items. Python was used to develop this program because it has a short command list that is simple to understand, high-performance libraries, and multi-platform portability. The application utilizes an algorithm to optimize the distribution of products while considering numerous constraints like distance, cargo capacity, time windows, and the number of destinations. Additionally, it makes use of map information from databases to increase effectiveness. By utilizing this program, the transportation system becomes more efficient since it requires fewer vehicles, reduces trip time, distances, and lowers the cost of distribution. To demonstrate confidence in the program's effectiveness, this report compares the program's outcomes.

Key words: freight Distribution; Python; algorithm

1. คำนำ

การขนส่งมีความสำคัญอย่างมากในการพัฒนาประเทศ ด้านเศรษฐกิจและสังคม รวมทั้งเทคโนโลยีสมัยใหม่ นอกจากนี้การขนส่งยังมีบทบาทต่อการดำรงชีวิตความเป็นอยู่ของประชากร ทั้งยังมีส่วนช่วยในการติดต่อไปมาหาสู่กันได้สะดวกและรวดเร็ว ยิ่งมีการพัฒนาด้านถนนหนทางที่ดีขึ้น ยานพาหนะมีประสิทธิภาพมากขึ้นแล้ว การขนส่งก็จะมีบทบาทในประเทศมากขึ้น ปัจจุบันการขนส่งจำเป็นต้องใช้เครื่องมือที่มีประสิทธิภาพ โดยอาศัยหลักการทางวิทยาศาสตร์ และ วิศวกรรมศาสตร์ มาช่วยปรับปรุงแก้ไขให้ดีขึ้น โดยการขนส่งเป็นสิ่งที่ให้อรรถประโยชน์ในหลายด้าน ยกตัวอย่างเช่น ทางด้านเวลา หมายถึง ความพอใจที่ผู้บริโภคได้รับเนื่องจากได้สินค้าหรือบริการในเวลาที่ตนต้องการ และ ด้านสถานที่ หมายถึง ความพอใจที่ผู้บริโภคได้รับเนื่องจากได้สินค้าหรือบริการในสถานที่ที่ตนสะดวก เป็นต้น

ปัจจุบันมีตัวเลือกของยานพาหนะจำนวนมากมาย แต่ที่กำลังเป็นที่นิยมในสังคมไทยและต่างประเทศนั้น คือ รถยนต์ไฟฟ้า ด้วยเหตุผลหลักทางด้านต้นทุนค่าเชื้อเพลิงต่ำกว่ารถยนต์ที่ขับเคลื่อนด้วยน้ำมัน รวมถึงเป็นยานพาหนะที่เป็นมิตรต่อสิ่งแวดล้อม ดังนั้นอุตสาหกรรมการขนส่งจึงเริ่มใช้รถยนต์ไฟฟ้าอย่างแพร่หลาย เพื่อให้การเดินทางปลอดภัย และสะดวกสบายมากขึ้น ปราศจากการปล่อยมลพิษทางอากาศสู่ชุมชน

การขนส่งสินค้าที่มีประสิทธิภาพนั้นยังจำเป็นต้องมีการวางแผนที่ดี โดยวัตถุประสงค์ของการแก้ปัญหาการออกแบบเส้นทาง คือ เพื่อลดจำนวนรถที่ใช้ขนส่ง เพื่อลดระยะทางในการเดินทาง ลดระยะเวลาในการเดินทาง เมื่อระยะทางและระยะเวลาลดลงค่าใช้จ่ายต้นทุนที่เกิดขึ้นในการขนส่งก็จะลดลงตามไปด้วย รวมถึง เพื่อเพิ่มความพึงพอใจให้กับผู้รับบริการด้วย ซึ่งวิธีการวิเคราะห์เส้นทาง สามารถทำได้โดยใช้การเขียนโปรแกรมร่วมด้วย เพื่อช่วยให้การวิเคราะห์คำนวณนั้นเป็นไปได้อย่างถูกต้อง และ รวดเร็ว อีกทั้งการเขียนโปรแกรมยังสามารถจัดการกับข้อมูลที่มีขนาดใหญ่มากขึ้นได้ดี ซึ่งการเขียนโปรแกรมยังมีโอกาสในการพัฒนาต่อยอดในอนาคตอีกด้วย

2. เอกสารและทฤษฎีที่เกี่ยวข้อง

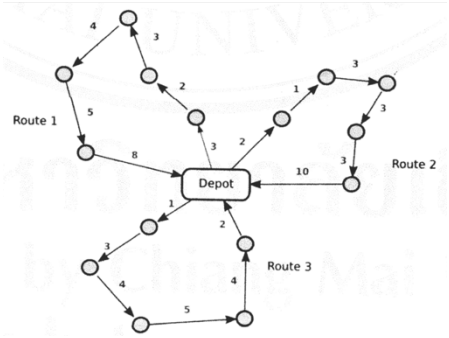
2.1 กราฟ (Graphs) [4]

กราฟประกอบด้วย 2 ส่วน คือ โหนดต่างๆ หรือจุดยอด (Vertex) และแขนเชื่อมระหว่างจุดยอด ซึ่งเรียกว่าเอดจ์ (Edges) หรือ อาร์ซ (Arcs) ดังนั้นดราฟสามารถเขียนได้ด้วยสัญลักษณ์ $G(V_i, E_j)$ โดย V_i คือจุดยอด และ E_j คือแขนเชื่อม

2.2 ปัญหาการจัดเส้นทางสำหรับยานพาหนะ (Vehicle Routing Problem; VRP) [7]

ปัญหาการจัดเส้นทางนั้นเป็นส่วนหนึ่งในการจัดการ ด้านโลจิสติกส์ (logistics) คือปัญหาการตัดสินใจเลือกทางเลือกที่ดีที่สุดในการเคลื่อนย้ายวัตถุ โดยวางแผนจัดลำดับ รวมถึงการกระจายสินค้าโดยการจัดการและแก้ปัญหาอย่างมีประสิทธิภาพจะช่วยลดต้นทุนการดำเนินงานที่เกิดขึ้นได้ โดยเป้าหมายสำคัญคือ ออกแบบกลุ่มของยานพาหนะทุกคันให้มีการเดินทางโดยใช้ต้นทุนต่ำที่สุด ซึ่งมีจุดเริ่มต้นและสิ้นสุด รวมถึงเส้นทางที่จะขนส่งโดยพิจารณาถึงเงื่อนไขหรือข้อจำกัดต่างๆ เช่น กรอบเวลาที่กำหนด ระยะทางในการเดินทาง และจำนวนยานพาหนะที่ใช้ในการเดินทาง ซึ่งปัญหาการจัดเส้นทางเดินรถมีหลักการดังนี้

1. ในแต่ละเส้นทางต้องมีจุดเริ่มต้นและสิ้นสุดที่จุดกระจายสินค้า
2. ลูกค้าแต่ละรายต้องได้รับบริการจากรถขนส่งสินค้าคันเดียวเท่านั้น
3. ปริมาณสินค้าของลูกค้าที่ได้รับบริการจากรถขนส่งสินค้าต้องไม่เกินปริมาตรความจุของรถขนส่งสินค้า



รูปที่ 1 ปัญหาการจัดเส้นทางเดินรถ

2.3 ปัญหาวิถีสั้นที่สุด (Shortest Path Problem) [1][2][5][6]

เป็นปัญหาที่ต้องการหาวิถีสั้นที่สุดระหว่างจุดยอด 2 จุดภายในกราฟ ตามแต่ละประเภทของกราฟ คือวิถีที่ผลรวมของน้ำหนักในวิถีนั้นน้อยที่สุดในบรรดาวิถีทั้งหมด สำหรับกราฟไม่ถ่วงน้ำหนัก นิยามให้วิถีสั้นที่สุดคือวิถีที่มีเส้นเชื่อมน้อยที่สุด การแก้ปัญหาในแต่ละรูปแบบมีวิธีการที่แตกต่างกันออกไป ซึ่งทำให้มีประสิทธิภาพมากกว่าการแก้ปัญหาวิถีสั้นสุดแบบคู่เดียวหลายๆครั้ง ขั้นตอนวิธีในการแก้ปัญหาวิถีสั้นสุด จะใช้แนวคิดของการคลายเส้นเชื่อม (relaxation) นั่นคือขณะเริ่มต้น คำตอบวิถีสั้นสุดจะยังไม่ถูกต้อง เส้นเชื่อม e จะเรียกว่า ตึง (tense) ถ้าสามารถทำการคลายเส้นเชื่อมที่ตึงเพื่อให้ได้คำตอบของวิถีสั้นสุดที่ดียิ่งขึ้นเรื่อย ๆ สุดท้ายถ้าไม่พบเส้นเชื่อมที่ตึงก็แปลว่าวิถีที่ได้ เป็นวิถีสั้นที่สุดแล้ว

2.4 วิธีที่ใช้ในการแก้ปัญหาการจัดเส้นทางเดินรถ [3]

2.4.1 วิธีการหาคำตอบที่เหมาะสมที่สุด (Optimal Solution)

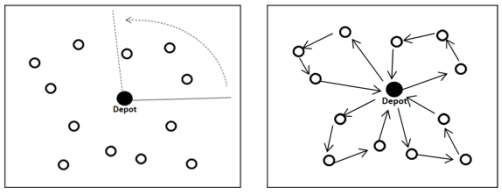
วิธีการแตกกิ่งและกำหนดขอบเขต (Branch and Bound) คือวิธีในการแก้ปัญหาการตัดสินใจแบบไม่ต่อเนื่องโดยอาศัยการแบ่งหรือแตกกิ่งทางเลือกของปัญหาออกเป็นแผนภูมิต้นไม้ ขั้นตอนวิธีการเริ่มจากปัญหาที่มีขนาดใหญ่ แล้วแบ่งปัญหานั้นเป็นปัญหาย่อย ๆ โดยคำตอบที่ได้จากปัญหาย่อยจะต้องไม่ขัดแย้งกับเงื่อนไขเดิม จากนั้นกระทำซ้ำกับทุกปัญหาย่อย ๆ จนกระทั่งพบปัญหาย่อยที่ให้ผลเฉลยที่ดีที่สุด แต่การแก้ปัญหาด้วยวิธีนี้ต้องการคอมพิวเตอร์ในการประมวลผลที่มีประสิทธิภาพสูง และ ใช้เวลาในการประมวลผลนาน ทำให้ขนาดของปัญหาที่สามารถหาคำตอบได้จากวิธีนี้มีจำกัด

2.4.2 วิธีการหาคำตอบที่ใกล้เคียงเหมาะสมที่สุด (Near Optimal Solution) [8][9]

ฮิวริสติกส์ (Heuristics) เป็นหนึ่งในวิธีการหาผลเฉลยที่ดีเพียงพอในเวลาที่จำกัด โดยใช้สามัญสำนึกของมนุษย์เข้าช่วยในการแก้ไขปัญหาผนวกเข้ากับแบบจำลองทางคณิตศาสตร์อย่างง่ายเพื่อให้ได้คำตอบที่รวดเร็วที่สุด ขั้นตอนวิธีการแก้ปัญหาแบบฮิวริสติกส์ มีอยู่มากมาย ยกตัวอย่างเช่น

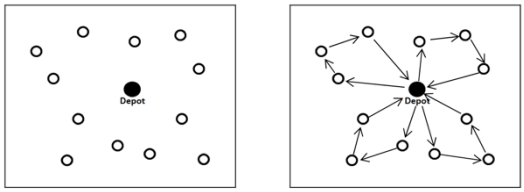
1. Sweep Heuristic หรือการจัดเส้นทางแบบกวาดเป็นการแบ่งกลุ่มลูกค้าโดยการกวาดรวมลูกค้าตามเข็มนาฬิกา พร้อมกับพิจารณา ความต้องการไปด้วย จนเต็มความจุของตัวรถ จึงกลับมายัง Depot และเริ่มต้นอีกครั้งจนครบทุกจุดรับส่ง โดยมีหลักการคำนวณหาเส้นทางดังนี้

1. เลือกโหนดเริ่มต้นโดยไม่รวม Depot โดยพิจารณาโหนดที่ใกล้ Depot มากที่สุด
2. กวาดรวมลูกค้าทิศทางทวนเข็มนาฬิกา พร้อมพิจารณาความต้องการไปด้วย เพื่อไม่ให้เกินความจุของยานพาหนะ
3. ทำตามขั้นตอนจนรวมจุดส่งสินค้าครบทุกจุด
4. ใช้วิธี TSP จัดเส้นทางของยานพาหนะแต่ละคัน



รูปที่ 2 การจัดเส้นทางด้วยวิธี Sweep Heuristic

2. Nearest Neighbor เป็นวิธีค้นหาจุดส่งสินค้าที่อยู่ใกล้กับจุดส่งจุดสุดท้ายโดยปริมาณจะต้องไม่เกินความจุของยานพาหนะที่ใช้ในการขนส่งสินค้า และ กลับสู่ Depot เมื่อปริมาณความจุของยานพาหนะเต็มความจุ จากนั้นสร้างเส้นทางใหม่จนครบทุกจุดรับส่งสินค้าโดยการสร้างเส้นทางเริ่มต้นจะสร้างจากจุดที่อยู่ใกล้คลังสินค้ามากที่สุด จากนั้นค้นหาจุดที่อยู่ใกล้จุดสุดท้ายของเส้นทางมากที่สุด และ เพิ่มจุดที่อยู่กับจุดสุดท้ายมากที่สุดลงในเส้นทาง ทำซ้ำจนครบทุดจุด แสดงตัวอย่างดังรูปที่ 4



รูปที่ 3 การจัดเส้นทางด้วยวิธี Nearest Neighbor

3. ระเบียบวิธีการวิจัย

3.1 กำหนดปัจจัยและเงื่อนไขในการเลือกเส้นทางการกระจายสินค้า

- ความสามารถในการบรรทุกของรถสามล้อไฟฟ้า ต้องออกแบบให้สามารถปรับค่าปริมาตรของกล่องขนส่งสินค้าได้
- ระยะการใช้งานของรถต่อการชาร์จแบตเตอรี่ 1 รอบ สามารถวิ่งได้ประมาณ 80-100 กิโลเมตร หรือ ประมาณ 3 ชั่วโมง และต้องคำนวณ ให้กลับมายังจุดชาร์จแบตได้
- ระยะเวลาในการขนส่งต่อรอบ รายการขนส่งที่เข้ามาต้องส่งให้ทันในช่วงเวลาใด และแต่ละรอบขนส่งให้เวลานานเท่าไร เวลาที่ได้รับรายการขนส่งสินค้าสามารถทำให้สำเร็จได้ตามเวลาในการทำงาน ของทางบริษัทหรือไม่
- ขอบเขตพื้นที่และระยะทางของการขนส่ง ระยะทางของการขนส่งแต่ละจุดสามารถลดมาเชื่อมกันเพื่อสามารถช่วยลดจำนวนรอบในการออกไปขนส่งได้สถานที่ที่ต้องการจัดส่งขึ้นตรงกับจุดกระจายสินค้าที่ใดสามารถวางแผนได้อย่างไรบ้าง

3.2 การทำ Flowchart

- แผนภาพแสดงลำดับการเคลื่อนไหวหรือการกระทำของคนหรือสิ่งที่เกี่ยวข้องในระบบต่างๆ ข้อดีของ Flow Chart คือการแสดงภาพรวมและช่วยจัดลำดับขั้นตอนการทำงานในระบบได้อย่างเป็นระเบียบและถูกต้อง เพื่อเพิ่มประสิทธิภาพในการทำงาน
- การเลือกเส้นทางในการกระจายสินค้าจำเป็นต้องวิเคราะห์เงื่อนไข และควบคุมปัจจัยที่เกี่ยวข้อง การใช้ Flowchart สามารถช่วยจัดระเบียบกระบวนการทางความคิดเพื่อทำให้งานดำเนินไปอย่างมีประสิทธิภาพที่สุด

3.3 การเขียนโปรแกรมโดยภาษาไพทอน

- เป็นภาษาคอมพิวเตอร์ที่มีความสามารถสูง ถูกสร้างโดย Guido van Rossum และถูกเผยแพร่ครั้งแรกในปีคริสตศักราช 1997 มีต้นแบบการพัฒนามาจากภาษา C ทำให้โครงสร้างภาษาของไพทอนมีความคล้ายคลึงกับภาษา C อย่างมาก ไพทอนเป็นภาษาแบบ interpreter โดยจุดประสงค์ของการพัฒนาคือการทำให้โค้ดอ่านได้เข้าใจง่ายขึ้น มีโครงสร้างของภาษาที่สั้นและกระชับ เพื่อให้นักพัฒนาโปรแกรมสามารถเข้าใจแนวคิดการเขียนโค้ดโดยใช้บรรทัดที่น้อยลงจากภาษา C++ และ Java นอกจากนี้ไพทอนยังถูกออกแบบ โครงสร้างของภาษาให้สนับสนุนการเขียนโค้ดให้เข้าใจง่ายทั้งในการทำโปรแกรมขนาดเล็กไปจนถึงโปรแกรมขนาดใหญ่
- Python เป็น Open Source ที่มีผู้เข้ามาพัฒนาจำนวนมาก ทำให้สามารถศึกษาข้อมูลเพิ่มเติมได้ค่อนข้างสะดวก นอกจากนี้มี Library ให้เลือกใช้ได้อย่างหลากหลาย จึงสามารถช่วยให้วางแผนการเลือกเส้นทางได้อย่างมีประสิทธิภาพมากขึ้น

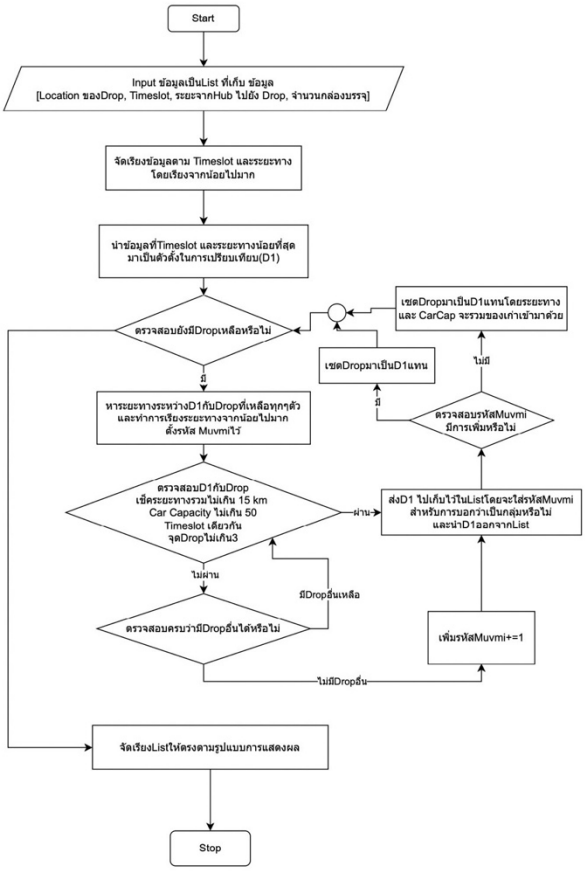
3.4 การดึงข้อมูลแผนที่ [10]

- การดึงข้อมูลแผนที่จาก OpenStreetMap ซึ่งเป็นผู้ให้บริการฐานข้อมูลในสารสนเทศของแผนที่ โดยมีจุดประสงค์ คือ แบ่งปันเพื่อส่งเสริมและพัฒนาให้สามารถเข้าถึงข้อมูลเชิงพื้นที่ได้ฟรี ซึ่ง OpenStreetMap มีบริการเปิด API สำหรับผู้ที่ต้องการเข้าเชื่อมต่อข้อมูลเข้ากับเว็ปไซต์ได้แบบไม่มีค่าใช้จ่าย

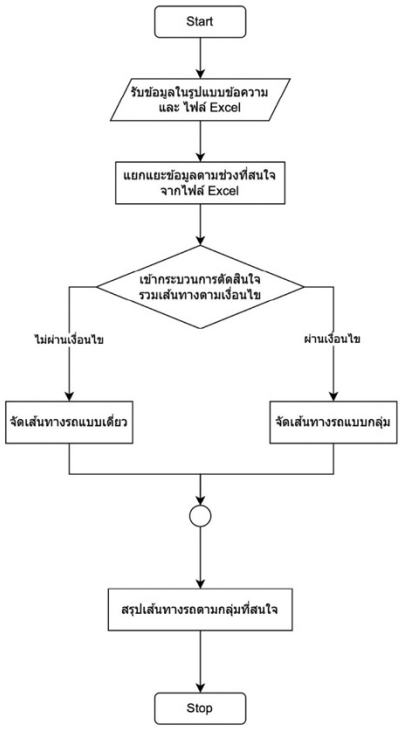
4. ผลการดำเนินงานวิจัย

4.1 Flowchart สำหรับโปรแกรมจัดเส้นทางการกระจายสินค้า

สำหรับโปรแกรมจัดเส้นทางกระจายสินค้าจะมีการใส่ข้อมูล (input) ในรูปแบบของตาราง Microsoft excel ซึ่งประกอบไปด้วยข้อมูลตามรูปแบบที่กำหนด, ชื่อจุดกระจายสินค้า และ วันที่ที่ต้องการจะจัดเส้นทาง แล้วจากนั้นจะนำข้อมูล ที่ได้ไปผ่านกระบวนการจัดเส้นทางเพื่อแสดงผลออกมาในรูปแบบของตารางบนหน้า Platform และ ตาราง Microsoft excel ซึ่งจะกล่าวถึงรายละเอียดของหัวข้อในหัวข้อย่อยต่อไป



รูปที่ 4 Flowchart สำหรับ Algorithm ที่ใช้ในโปรแกรมจัดเส้นทางกระจายสินค้า



รูปที่ 5 Flowchart สำหรับโปรแกรมจัดเส้นทางกระจายสินค้า

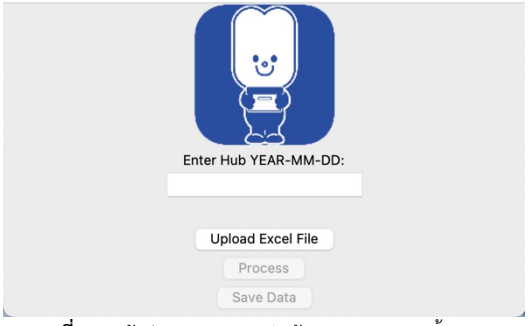
4.2 ข้อมูลที่จะต้อง input ใส่เข้ามาในโปรแกรม

4.2.1 ข้อมูลที่ต้องใส่บนหน้าต่าง platform

ช่อง Upload an Excel file : ต้องทำการเลือกไฟล์ตาราง Microsoft excel ที่มีข้อมูลตามรูปแบบที่กำหนด

ช่อง Enter Hub YEAR-MM-DD : ต้องกรอกชื่อของจุดกระจายสินค้าตามด้วย ปี-เดือน-วันที่ ยกตัวอย่างเช่น B 2022-12-16

จากนั้นกดที่ปุ่ม Process เพื่อทำการแยกข้อมูลตามประเภท และเตรียมจัดข้อมูลใส่ใน list หลังจากนำข้อมูลใส่ใน list แล้ว จะดำเนินการจัดเส้นทางการกระจายสินค้าต่อไป



รูปที่ 6 หน้าต่าง Platform จัดเส้นทางกระจายสินค้า

4.2.2 ข้อมูลไฟล์ตาราง Microsoft excel

จากที่กล่าวไว้ในหัวข้อข้างต้น รูปแบบของไฟล์ตาราง Microsoft excel ต้องมีข้อมูลและการจัดเรียงข้อมูลที่กำหนดเอาไว้ เพื่อสามารถดำเนินการต่อไปได้ จากภาพที่ 4.3 จะเห็นถึงตัวอย่างของไฟล์ตาราง Microsoft excel ที่ถูกต้อง โดยจะต้องมีรูปแบบของตารางดังนี้

Column A: delivery_date คือ วันที่ที่ต้องการส่งสินค้า

Column B: box_area คือ รหัสของศูนย์กระจายสินค้า ซึ่งจะต่างกันออกไปตามแต่ละพื้นที่จุดส่ง โดยต้องมีชื่อของศูนย์กระจายสินค้าเริ่มต้นเสมอ เช่น BN หมายถึงรหัสของศูนย์กระจายสินค้า B

Column C: Sum of Box คือ จำนวนกล่องที่รายการนั้นๆ ต้องการจะจัดส่ง เช่น 19 หมายถึง รายการนั้นมีพัสดุจัดส่ง 19 กล่อง

Column D: dest_location คือ พิกัดละติจูด-ลองจิจูดของจุดหมายปลายทาง

Column E: delivered_time_slot คือ เวลาในการจัดส่งสินค้าที่เร็วที่สุด โดยสามารถจัดส่งได้ภายในเวลาที่กำหนดไว้ใน Column I เช่น Column E =14 และ Column I = 1 หมายถึงสามารถเริ่มจัดส่งสินค้ารายการนี้ในเวลา 14.00 น. และสามารถส่งสินค้าได้ถึง 15.00 น.

Column F: Hub คือพิกัดละติจูด-ลองจิจูดของจุดกระจายสินค้า

Column G: HUB_NAME คือชื่อของจุดกระจายสินค้า โดยที่ชื่อของจุดกระจายสินค้า ต้องอยู่ใน row เดียวกันกับ พิกัดละติจูด-ลองจิจูดของจุดกระจายสินค้านั้น

Column H: CARCAP คือ จำนวนกล่องที่รถ สามารถบรรทุกได้ เช่น 50 หมายถึง รถคันนั้นสามารถบันทุกกล่องได้มากที่สุด 50 กล่อง

Column I: TIME(HR) คือ ช่วงเวลาในหน่วยชั่วโมง ที่สามารถขนส่งได้หลัง delivered_time_slot (เวลาจัดส่งสินค้าที่เร็วที่สุด) เช่น Column E =14 และ Column I = 1 หมายถึงต้องจัดส่งสินค้ารายการนี้ภายใน 1 ชั่วโมง ซึ่งคือ 14.00 น. ถึง 15.00 น.

Column J: DROP คือ จำนวนจุดปลายทางที่สามารถไปส่งสินค้าได้มากที่สุด เช่น 3 หมายถึง ใน 1 รอบการเดินรถ สามารถไปส่งสินค้าได้มากที่สุด 3 จุด แล้วจะต้องกลับมาที่จุดกระจายสินค้า

4.3 ขั้นตอนการสร้างลำดับงาน - เส้นทางการกระจายสินค้า

ขั้นตอนที่ 1: หลังจากผ่านการรับข้อมูลตามรูปแบบที่กำหนดและทำการแยกประเภทข้อมูลแล้ว หลังจากนั้นจะนำ Circuity factor ที่ได้คำนวณเอาไว้ ซึ่งได้ค่าเท่ากับ 1.4655 มาใช้ในการหาระยะทาง และนำข้อมูลของแต่ละรายการมาเรียงเป็น list โดยวน loop ไปจนครบทุกรายการโดยมีรูปแบบดังนี้

[slot เวลาของจุดปลายทาง, ระยะทางจากศูนย์กระจายสินค้าไปยังจุดปลายทาง, ลำดับของรายการ, Car capacity, พิกัดละติจูด-ลองจิจูดของจุดปลายทาง, ระยะทางจากจุดปลายทาง ไปยังศูนย์กระจายสินค้า, slot เวลาของจุดปลายทาง]

แล้วนำข้อมูลทั้งหมดที่ทำการจัดรูปแบบแล้ว ใส่ไว้ใน list อีกที

ขั้นตอนที่ 2: ทำการเรียงลำดับ slot เวลาของจุดปลายทาง และระยะทางจากศูนย์กระจายสินค้าไปยังจุดปลายทาง จากน้อยไปมาก

ขั้นตอนที่ 3: นำรายการที่เรียงลำดับได้เป็นลำดับแรก มาเป็นจุดตั้งต้นกำหนดให้เป็นจุด D1

ขั้นตอนที่ 4: จับคู่จากจุด D1 ไปยังทุกจุดหมายปลายทางอื่นทุกจุดทีละจุด (กำหนดให้จุดหมายปลายทางอื่น แทนด้วย D2) โดยมีเงื่อนไขดังนี้

- ปริมาณกล่องสินค้า ของ D1 และ D2 รวมกัน ต้องไม่เกินที่กำหนด
- ระยะทางของ D1 และ D2 รวมกัน ต้องไม่เกินที่กำหนด
- Time slot เดียวกันทั้ง D1 และ D2
- จำนวนของจุดส่งสินค้า ต้องไม่เกินที่กำหนดไว้

จากนั้นทำการสร้าง list ใหม่ที่ประกอบไปด้วย list ของข้อมูล D1 และ D2 แต่ละจุดที่จับคู่กัน โดยมีรูปแบบดังนี้

[ระยะทางจากจุด D1 ไปยัง D2, ลำดับรายการของ D2, ปริมาณกล่องสินค้าของ D1+D2, Slot เวลาของ D1, Slot เวลาของ D2, พิกัดละติดจูด-ลองจิจูดของจุด D2, ปริมาณกล่องสินค้าของ D2)

ขั้นตอนที่ 5: ทำการเรียงลำดับระยะทาง จากจุด D1 ไปยัง D2 แต่ละจุดจากน้อยไปหามาก

ขั้นตอนที่ 6: นำคู่ของรายการ D1D2 ที่เรียงลำดับได้เป็นลำดับแรก มาเป็นจุดตั้งต้นใหม่ กำหนดให้เป็นจุด D1D2A

ขั้นตอนที่ 7: ทำการจับคู่และสร้าง list ใหม่จากจุด D1D2A ไปยังทุกจุดหมายปลายทางอื่นทุกจุดทีละจุด โดยมีเงื่อนไขเดียวกันกับในขั้นตอนที่ 4

ขั้นตอนที่ 8: ทำขั้นตอนที่ 4-7 ไปจนกว่าจะไม่ผ่านเงื่อนไขทั้งหมด จะถือว่าได้เส้นทางกระจายสินค้า 1 รอบการเดินรถเป็นที่เรียบร้อย

ขั้นตอนที่ 9: นำเอารายการส่งสินค้าที่ได้มีการจับคู่เป็นที่เรียบร้อยแล้วออกจาก list เพื่อให้ไม่เกิดการจับคู่ซ้อนทับกัน และเริ่มทำการเรียงลำดับและจับคู่ใหม่ ตั้งแต่ขั้นตอนที่ 1 ไปจนครบรายการส่งสินค้าทั้งหมดที่มี

4.4 การแสดงผลเส้นทางกระจายสินค้า

	A	B	C	D	E	F
1	Driver	timeslot	carcap	location	Distance	Link
2	1				-	-
3	1				6.447	https://ww
4	2				13.891	https://ww
5	3				15.277	https://ww
6	4				8.953	https://ww
7	5				4.901	https://ww
8	6				-	-
9	6				2.56	https://ww
10	7				14.839	https://ww
11	8				15.36	https://ww
12	9				-	-
13	9				8.733	https://ww
14	10				-	-
15	10				5.368	https://ww

รูปที่ 7 ตัวอย่างภาพการแสดงผลบนตาราง Microsoft excel

Column A: Driver คือ รอบการเดินรถ

Column B: timeslot คือ เวลาเริ่มรอบการเดินรถนั้นๆ ที่ต้องไปส่งภายในเวลา time window ที่กำหนด

Column C: Number of box คือ จำนวนกล่องพัสดุของรอบการเดินรถนั้นๆ เป็นแบบสะสม (cumulative) ในรอบการเดินรถนั้น

Column D: location คือ พิกัดละติจูด-ลองจิจูดของจุดปลายทาง

Column E: Dist คือ ระยะทางรวมของรอบการเดินรถนั้นๆ ถ้าหากแสดงผลว่า – หมายถึงยังไม่ถึงจุดสุดท้ายของรอบการเดินรถนี้ เช่น ใน row ที่ 3 แสดงผลว่า – หมายความว่ายังไม่ถึงจุดส่งสินค้าสุดท้าย ในทางกลับกัน ใน row ที่ 4 แสดงผลว่า 10.844 หมายความว่าเป็นจุดส่งสินค้าสุดท้ายของรอบการเดินรถนี้ และ มีระยะทางรวมทั้งสิ้น 10.844 กม. เป็นต้น

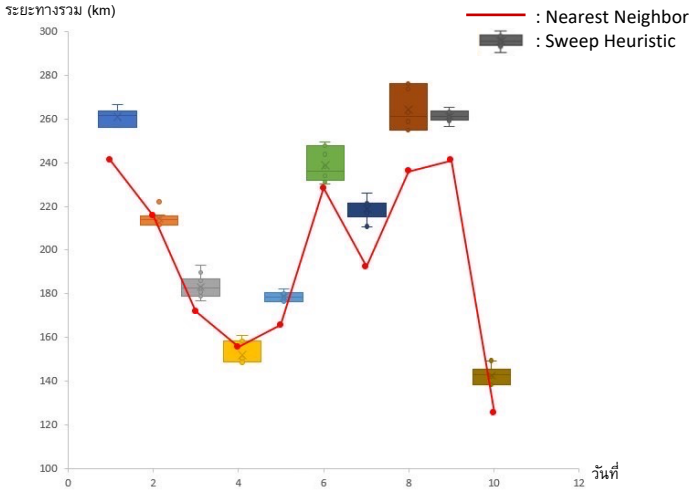
Column F: Link คือ link ที่จะนำไปสู่เว็ปไซต์ Google map ที่แนะนำเส้นทางของรอบการเดินรถนั้นตั้งแต่จากจุดกระจายสินค้า ไปยังจุดส่งสินค้าอื่นๆ และ กลับมายังจุดกระจายสินค้าเดิมเช่นเดียวกับใน Column E

4.5 การเปรียบเทียบผลลัพธ์

1. เหตุผลที่เลือกใช้หลักการของ Nearest Neighbor Algorithm ร่วมกับการดึงข้อมูลแผนที่จาก OpenStreetMap และการดัดแปลง Nearest Neighbor ให้เหมาะสม เนื่องจาก OpenStreetMap เป็นฐานข้อมูลแผนที่ที่เชื่อถือได้ สามารถทำงานได้ดีกับการเขียนโปรแกรมภาษา Python และ เนื่องจากการโปรแกรมการจัดเส้นทางกระจายสินค้านี้เป็นเพียงต้นแบบ ทำให้ยังไม่ได้มีการลงทุนด้านงบประมาณในการดึงข้อมูลแผนที่ที่มีค่าใช้จ่าย ดังนั้นด้วยเหตุผลเหล่านี้ OpenStreetMap จึงเหมาะสมสำหรับการพัฒนาโปรแกรมกระจายสินค้า

2. การเปรียบเทียบผลลัพธ์ของการจัดเส้นทางด้วย Saving Algorithm และ Sweep Algorithm จะได้ผลลัพธ์ว่า การจัดเส้นทางการกระจายสินค้าด้วยวิธีประยุกต์มาจาก Nearest Neighbor Algorithm สามารถลดจำนวนการออกรอบของรถยนต์ไฟฟ้า ได้มากกว่าวิธี Sweep Heuristic

ต่อไปเป็นการเปรียบเทียบระยะทางรวมของการจัดเส้นทางแบบ Sweep Heuristic และ Nearest Neighbor ใน 10 วันแรกของเดือนธันวาคม พ.ศ.2565 โดยทำการเปรียบเทียบในรูปแบบของกราฟ ซึ่งได้มีการทดสอบวิธี Sweep Heuristic เป็นจำนวน 10ครั้ง/1ศูนย์กระจายสินค้า และ ทำการ plot ออกมาเป็นกราฟ Box and Whisker เพื่อให้เห็นถึงผลลัพธ์ที่แตกต่างกันในแต่ละรอบของการทดลอง



รูปที่ 8 กราฟแสดงตัวอย่างการเปรียบเทียบระยะทางรวมของการจัดเส้นทางด้วยวิธี Nearest Neighbor และ Sweep Heuristic (Hub3)

จากกราฟแสดงตัวอย่างการเปรียบเทียบระยะทางรวมในแต่ละวันของการจัดเส้นทาง ด้วยวิธี Nearest Neighbor และ Sweep Heuristic จำนวนคำตอบที่ได้จากวิธี Nearest Neighbor นั้นมีระยะทางรวมที่น้อยกว่า Sweep Heuristic อยู่ถึง 74%, มีจำนวนคำตอบที่ใกล้เคียงกับ Sweep Heuristic อยู่ 18% และ มีจำนวนคำตอบที่มีระยะทางรวมมากกว่า Sweep Heuristic อยู่เพียง 8% ซึ่งการจัดเส้นทางด้วยวิธี Sweep Heuristic นั้นจะได้ระยะทางรวมที่ไม่เท่ากันในแต่ละรอบการทดลอง เนื่องจากในวิธีนี้จะทำการสุ่มเลือกจุดเริ่มต้นและกวาดรวมจุดในทิศทางทวนเข็มนาฬิกา พร้อมพิจารณาความต้องการไปด้วย ทำให้มีความไม่แน่นอนในการจัดเส้นทาง

โดยสรุปจากการเปรียบเทียบจำนวนรอบเที่ยวของรถยนต์ไฟฟ้าที่ต้องใช้นั้น น้อยกว่าในทุกศูนย์กระจายสินค้า และจากการวิเคราะห์กราฟแสดงตัวอย่างการเปรียบเทียบระยะทางรวมในแต่ละวันของการจัดเส้นทาง ด้วยวิธี Nearest Neighbor และ Sweep Heuristic คำตอบที่ได้จากวิธี Nearest Neighbor Algorithm มีแนวโน้มที่จะให้เส้นทางที่มีระยะทางที่สั้นกว่า ทำให้การกระจายสินค้าด้วย Nearest Neighbor Algorithm มีความประหยัดทรัพยากร และ มีประสิทธิภาพมากกว่าการจัดเส้นทางกระจายสินค้าด้วย Sweep Heuristic ด้วยเหตุผลเช่นนี้ ทางผู้จัดทำจึงตัดสินใจเลือกใช้ Nearest Neighbor ร่วมกับการดึงข้อมูลแผนที่จาก OpenStreetMap เป็นแผนที่อ้างอิงในการคำนวณระยะทาง

5. สรุปผลการวิจัย

5.1 สรุปผล

จากการศึกษานี้ผู้ศึกษาได้ทำการออกแบบ Platform เพื่อจัดเส้นทางการกระจายสินค้า โดยใช้หลักการของ Nearest Neighbor ที่มีการดัดแปลงให้เหมาะสมกับข้อกำหนดของการกระจายสินค้า และ ใช้การดึงข้อมูลแผนที่จาก OpenStreetMap (OSM) เป็นฐานข้อมูลแผนที่ที่สามารถดึงข้อมูลแผนที่ได้อย่างถูกต้องและรวดเร็ว ด้วยการเขียนโปรแกรมด้วยภาษา Python และ Library ที่หลากหลาย ทำให้เส้นทางการกระจายสินค้าที่ได้รับการจัดเส้นทางนั้นสามารถจัดเส้นทางได้อย่างมีประสิทธิภาพ

ในการจัดเส้นทางการกระจายสินค้านี้มีข้อกำหนดที่ต้องระบุมากมาย คือ ซึ่งการศึกษาออกแบบ Platform นี้ ทำให้การจัดเส้นทางกระจายสินค้าร่วมกันหลายรายการเกิดประโยชน์มากมาย ไม่ว่าจะเป็นการลดค่าใช้จ่ายในการกระจายสินค้า ประหยัดเวลา และ เพิ่มประสิทธิภาพในการจัดการส่งสินค้า โดย Platform นี้เป็นตัวอย่างที่ดีของการนำเทคโนโลยีและการพัฒนาซอฟต์แวร์มาใช้เพื่อเพิ่มประสิทธิภาพในการจัดการ และยังช่วยลดเวลาในการจัดเส้นทางและการกระจายสินค้าอีกด้วย

โดยจากการเปรียบเทียบการจัดเส้นทางการกระจายสินค้าด้วยหลักการ Nearest Neighbor Algorithm และ หลักการ Sweep Heuristic จะได้ผลลัพธ์ว่า จำนวนรอบการเดินรถของรถยนต์ไฟฟ้าที่จัดด้วย Nearest Neighbor Algorithm นั้นน้อยกว่า Sweep Heuristic ในทุกศูนย์กระจายสินค้า และจากการวิเคราะห์กราฟแสดงตัวอย่างการเปรียบเทียบระยะทางรวมในแต่ละวันของการจัดเส้นทาง คำตอบที่ได้จากวิธี Nearest Neighbor Algorithm มีแนวโน้มที่จะให้เส้นทางที่มีระยะทางที่สั้นกว่า ทำให้การจัดเส้นทางกระจายสินค้าด้วย Nearest Neighbor Algorithm มีความประหยัดทรัพยากร และ มีประสิทธิภาพมากกว่าการจัดเส้นทางกระจายสินค้าด้วย Sweep Heuristic ทางผู้จัดทำจึงตัดสินใจเลือกใช้ Nearest Neighbor Algorithm ร่วมกับการดึงข้อมูลแผนที่จาก OpenStreetMap เป็นข้อมูลแผนที่อ้างอิงในการคำนวณระยะทางด้วยเหตุผลเรื่องของความน่าเชื่อถือ ความสามารถในการทำงาน และ เรื่องของงบประมาณ

5.2 ข้อเสนอแนะ

จากการทำโปรแกรมการจัดเส้นทางการกระจายสินค้าโดยใช้หลักการประยุกต์มาจาก Saving Algorithm จำเป็นจะต้องมีการกรอกข้อมูลตามรูปแบบที่กำหนด เพื่อให้โปรแกรมทำงานได้อย่างมีประสิทธิภาพสูงสุด นอกจากนี้โปรแกรมยังได้มีการใช้แผนที่อ้างอิงมาจาก OpenStreetMap ซึ่งอาจจะมีความผิดพลาดในบางจุดเนื่องจาก เป็นแผนที่ที่มีลักษณะในการให้บุคคลภายนอกเข้ามามีส่วนช่วยพัฒนาด้วย ทำให้อาจจะไม่ได้รับข้อมูลที่ถูกต้อง ครบถ้วน และยังไม่ได้มีการอัพเดตแบบเรียลไทม์ ซึ่งถ้าหากสามารถจัดหางบประมาณในการพัฒนาโปรแกรมนี้ได้ แนะนำให้ใช้ API ของทาง Google Map จะทำให้ได้โปรแกรมที่เสถียรและแม่นยำมากยิ่งขึ้น

นอกจากนี้ ถ้าหากต้องการจะเพิ่มความประหยัดให้ได้มากขึ้นกว่าเดิม อาจทำได้โดยการเปลี่ยนขนาดของรถยนต์ไฟฟ้าให้สามารถขนส่งสินค้าได้มากขึ้น จะทำให้สามารถจับคู่จุดหมายปลายทางได้มากขึ้น ทำให้ลดระยะทางการเดินทางได้ เนื่องจากการจัดเส้นทางด้วยข้อจำกัดด้านความจุของรถยนต์ไฟฟ้าในปัจจุบัน เป็นปัญหาหลักที่ทำให้ไม่สามารถเพิ่มจุดหมายปลายทางได้ เพราะฉะนั้น การเพิ่มความจุให้แก่รถยนต์ไฟฟ้า หรือปรับเปลี่ยนชนิดของรถยนต์ไฟฟ้าจะสามารถช่วยในส่วนนี้ให้มีความประหยัดมากยิ่งขึ้น

เอกสารอ้างอิง

[1] Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford (2001). "Section 24.3: Dijkstra's algorithm". *Introduction to Algorithms* (Second ed.). MIT Press and McGraw–Hill. pp. 595–601. ISBN.

[2] Cormen, Thomas H. (22 March 2022). *Introduction to Algorithms*(4th ed.). MIT Press. ISBN 9780262046305. Retrieved August 19, 2022.Khalifa, Y., Salem, O. and Shahin, A. (2006), Cutting Stock Waste Reduction Using Genetic Algorithms, pp.1675-1680.

[3] J Brimley *et al.* (1992). *A new distance function for modeling travel distances in a transportation network*. Transportation Science.

[4] เบญจมาศ ปัญญางาม. (2017). *โครงสร้างข้อมูล บทที่ 11 กราฟ*. หน้า 149-180. ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่.

[5] *Shortest Path Problem* แหล่งที่มา: https://home.kku.ac.th/urachart/algo/pdf/short.pdf.

[6] ดรุณี หันวิสัย (2022). *อัลกอริทึมของ Dijkstra สำหรับแก้ปัญหาเส้นทางสั้นที่สุดบนเครือข่ายภายใต้ตัวเลขฟัซซีสี่เหลี่ยมคางหมู*. หน้า 30-41. สาขาวิชาคณิตศาสตร์ประยุกต์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏวไลยอลงกรณ์ ในพระบรมราชูปถัมภ์.

[7] ขวัญชนก วิเชียรวรรณ. (2019). *การจัดเส้นทางการขนส่งสินค้าแบบเปิดขนส่งหน้าตู้ กรณีศึกษา บริษัทไดนามิค ทรานสปอร์ต จำกัด*. วิศวกรรมศาสตรมหาบัณฑิต สาขาการจัดการทางวิศวกรรม มหาวิทยาลัยธุรกิจบัณฑิตย์.

[8] ปัญญาวัฒน์ จันทร์ชัยภักดิ์. (2018). *การแก้ปัญหาการจัดเส้นทางเดินรถรับส่งนักเรียน: กรณีศึกษาโรงเรียนประสิทธิ์ศึกษาสงเคราะห์*. วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยอุบลราชธานี.

[9] คลอเคลีย วจนะวิชากร; กนกกาญจน์ ศรีสุรินทร์. (2018). วิธีการหาคำตอบสำหรับปัญหาการจัดเส้นทางรถเก็บขยะมูลฝอย กรณีศึกษาเทศบาลตำบลอุบล จังหวัดอุบลราชธานี. คณะวิศวกรรมศาสตร์ มหาวิยาลัยอุบลราชธานี.

[10] *ประวัติความเป็นมา OpenStreetMap*. (ออนไลน์). (2021). แหล่งที่มา: https://www.mindphp.com/บทความ/277-openstreetmap/7910-opensteetmap.