

วิธีการผ่อนปรนปัญหาแบบลากรางจ์สำหรับแบบจำลองการจัดเส้นทางยานพาหนะร่วมกับโดรน

LAGRANGIAN RELAXATION METHOD FOR INTEGRATED VEHICLES AND DRONES ROUTING MODEL

พีรวิญญู เจริญวุฒิ^{1,*} และ รศ.ดร.มานิช โลหะเตพานนท์²

¹ ภาควิชาวิศวกรรมโยธา, คณะวิศวกรรมศาสตร์, จุฬาลงกรณ์มหาวิทยาลัย, กรุงเทพมหานคร

² สถาบันการขนส่ง, จุฬาลงกรณ์มหาวิทยาลัย, กรุงเทพมหานคร, ประเทศไทย

*Corresponding author address: Perawit.cha@gmail.com

บทคัดย่อ

การขนส่งสินค้าปัจจุบันนี้มีความต้องการเพิ่มขึ้นเป็นอย่างมากเนื่องจากการมาของระบบการซื้อขายผ่านอินเทอร์เน็ตทำให้หลายบริษัทได้พัฒนาการขนส่งรูปแบบต่าง ๆ เพื่อตอบสนองตลาดและลดต้นทุน หนึ่งในนั้นคือการใช้โดรนไร้ผู้บังคับควบคู่กับยานพาหนะอื่นเช่นรถบรรทุก โดยโดรนสามารถออกจากทั้งคลังสินค้าหรือจากรถบรรทุกเพื่อไปส่งสินค้าแล้วกลับมาที่ฐานพัก เพื่อรอรถบรรทุกมารับกลับหรือบินกลับคลังสินค้าเอง ปัญหาการจัดเส้นทางของยานพาหนะร่วมกับโดรน (Vehicle Routing Problem with Drone) ที่ศึกษาเป็นปัญหาการจัดเส้นทางขนส่งสินค้าของรถบรรทุกและโดรนไปพร้อม ๆ กัน ซึ่งเป็นปัญหาที่มีขนาดใหญ่มากจึงไม่สามารถใช้วิธีการแก้ปัญหาแบบทั่วไปได้ งานวิจัยนี้ได้พัฒนาเทคนิคการผ่อนปรนปัญหาแบบลากรางจ์ (Lagrangian Relaxation) เพื่อนำมาใช้ควบคู่กับเทคนิค Branch-and-Price ในการคำนวณขอบเขตล่างของคำตอบในแต่ละขั้นของ Branch-and-Price ซึ่งขอบเขตล่างของลากรางจ์นั้นจะมีค่าใกล้เคียงคำตอบมากกว่าคำตอบจาก Column Generation ในขั้นนั้น ส่งผลให้เวลาที่ใช้ในการหาคำตอบลดลง

คำสำคัญ: การผ่อนปรนปัญหาแบบลากรางจ์, Branch-and-Price, ปัญหาการจัดเส้นทางของยานพาหนะร่วมกับโดรน

Abstract

Nowadays, e-commerce increases parcels transportation demand year by year. Many companies are seeking the way to improve performance of delivery service and cost efficiency. One among many solutions is using a drone attached with a truck for the last mile delivery. Drone can be launched from either depot or truck and returned to the depot or wait for the truck to take them back from the docking station around delivery area. The vehicle routing problem with drone considers truck- and drone-route simultaneously. It is an enormous problem that cannot be solved by normal linear programming method. This research proposes the Lagrangian relaxation technique, in conjunction with the Branch-and-Price technique, to estimate the lower bound of the solution in each iteration of the Branch-and-Price. The lower bound from the Lagrangian relaxation is tighter than the solution from the Column generation which can improve the total solution time.

Keywords: Lagrangian relaxation, Branch-and-Price, Vehicle Routing Problem with Drones

1. บทนำ

จาก “The Last-Mile Delivery Challenge” [1] ลู ก ค้า ให้ความสำคัญกับความเร็วในการจัดส่งเนื่องจากคำสั่งซื้อของพวกเขา มักจะเป็นอาหารสดหรือของจากร้านค้าปลีก ที่จำเป็นต้องถูกส่งในวันเดียวกันมากขึ้น นับเป็นความท้าทายที่ยิ่งใหญ่สำหรับบริษัทจัดส่งในการรักษาคุณภาพการบริการ (เช่นเวลารอของลูกค้า) ซึ่งส่งผลโดยตรงต่อความเชื่อมั่นของลูกค้า แต่ในขณะเดียวกันการส่งสินค้าที่เพิ่มขึ้นจะทำให้กำไรลดลง 26% ใน 3 ปี จากการที่จัดส่งระยะสุดท้าย (Last-mile delivery) มีค่าใช้จ่ายสูงถึง 50% ของค่าจัดส่งทั้งหมด ผู้ให้บริการขนส่งจึงมีแนวคิดมากมายเพื่อช่วยในการให้บริการเช่น Drop-shipping, Drop-off lockers, Autonomous Vehicles และDrones [2]

ในปี 2013 Jeff Bezos ประกาศว่า Amazon กำลังพัฒนา

อากาศยานไร้คนขับหรือ Drone สำหรับการจัดส่งสินค้าขนาดเล็ก เรียกว่า “Amazon Prime Air” อย่างน้อย 86% ของน้ำหนักสินค้าที่จัดส่งนั้นน้ำหนักต่ำกว่า 5 ปอนด์ซึ่งเป็นน้ำหนักบรรทุกที่จำกัดของโดรน ทำให้จะใช้เวลาเพียง 30 นาทีใน Last-mile delivery [3] และในปี 2559 Amazon ประสบความสำเร็จในเที่ยวบินส่งมอบครั้งแรกในซานเมอริโดแคลิฟอร์เนีย หลังจากนั้นทำให้บริษัทค้าปลีกและโลจิสติกส์หลายแห่งหันมาสนใจเทคโนโลยีนี้ หรือแม้แต่บริษัท Alphabet Inc. ซึ่งเป็นบริษัทแม่ของ Google ก็ยังพัฒนาโครงการโดรนที่เรียกว่า Wing

ในภูมิภาคเอเชีย บริษัท Rakuten และ JD.com ซึ่งเป็นบริษัทค้าปลีกของประเทศญี่ปุ่นและประเทศจีนได้ร่วมมือกันเริ่มโครงการโดรนในปี 2558 Rakuten เปิดตัว “Sora Raku” บริการจัดส่งอุปกรณ์กีฬาในสนามด้วยโดรน [4, 5]

ข้อดีของการใช้โดรนคือสามารถเข้าถึงได้ทุกที่โดยไม่ต้องกังวลถึงความปลอดภัยบนท้องถนน เส้นทางตรงของโดรนสามารถประหยัดเวลาได้มากเมื่อเทียบกับการเดินทางไปตามถนนที่คับคั่ง อย่างไรก็ตาม ข้อจำกัดหลักของประการหลักคือระยะเวลาและความเร็วในการบินถูกจำกัดโดยแบตเตอรี่ ประการที่สองคือน้ำหนักและขนาดที่บรรทุก และประการสุดท้ายคือสภาพอากาศ

การศึกษาในด้านการวิจัยดำเนินการหรือ Operation Research นั้นมีแบบจำลองทางคณิตศาสตร์และวิธีการในการแก้ปัญหาการกำหนดเส้นทางของรถบรรทุก-โดรนจำนวนมาก ซึ่งแบบจำลองเชิงเส้น (Linear Programing) ที่นำเสนอสามารถแบ่งออกเป็น 2 ประเภท ได้แก่ Arc-based model โดยมีเส้นทางระหว่าง Node เป็นตัวแปร และ Path-based model เป็นแบบจำลองที่ดัดแปลงจาก Arc-based model โดยตัวแปรคือ Arc ต่อกันเป็นเส้นทาง ผลลัพธ์ที่ได้คือ Constraints ที่ซับซ้อนจะถูกใช้ในขั้นตอนสร้างตัวแปร

สามารถใช้วิธีการแบบ Simplex แก้ปัญหานี้ได้แต่จะเปลี่ยนทรัพยากรคอมพิวเตอร์เนื่องจากมีตัวแปรจำนวนมาก งานวิจัยหลายชิ้นจึงใช้เทคนิค Column Generation มาช่วยจัดการ [6, 7] อย่างไรก็ตามการที่จะได้มาซึ่งคำตอบที่เป็นจำนวนเต็มนั้นจำเป็นต้องใช้เทคนิค Branch-and-Price [8] เป็นการทำงานร่วมกันระหว่าง Column Generation และ Branch-and-Bound [9]

ข้อเสียที่สำคัญของ Branch-and-Price คือ การยืนยัน Optimal solution ที่ช้า เนื่องจากการแตก Branch จำนวนมาก Lagrangian Relaxation สามารถคำนวณค่าที่ใกล้เคียง Optimal solution ของ Node ปัจจุบันของ Branch-and-Price ได้ ถ้าหากค่านี้มากกว่าหรือเท่ากับ Incumbent solution Node นั้นจะถูกหยุด [10]

งานวิจัยนี้มีวัตถุประสงค์เพื่อศึกษาการใช้ Lagrangian relaxation สำหรับเพิ่มความเร็วของ Branch-and-Price สำหรับ Tour-Based model of Vehicle Routing Problem with Drones

2. งานวิจัยที่เกี่ยวข้อง

2.1. TRAVELLING SALESMAN PROBLEM WITH DRONES AND VEHICLE ROUTING PROBLEM WITH DRONES

Travelling salesman problem (TSP) คือปัญหาของพนักงานขายที่เดินทางไปยังกลุ่มลูกค้าที่กำหนดโดยใช้ระยะทางที่สั้นที่สุดในขณะที่ Vehicle routing problem (VRP) กำหนดให้ลูกค้าจำนวนหนึ่ง แล้วจัดเส้นทางการเดินทางเพื่อลูกค้าจะได้รับบริการในแต่ละเส้นทางตามความต้องการ ทั้งหมดต้องไม่เกินขีดจำกัด ความจุของรถ มีวัตถุประสงค์เพื่อลดต้นทุนรวมในการจัดส่ง

ให้น้อยที่สุด

มีงานวิจัยมากกว่าหลายพันฉบับที่นำเสนอวิธีการแก้ปัญหาโดยใช้อัลกอริทึมที่ให้คำตอบที่ Exact solution และแบบ Heuristic ที่ให้คำตอบที่ใกล้เคียง Optimal [11]

ปัญหาการกำหนดเส้นทางยานพาหนะกับโดรน หรือ Vehicle routing problem with drone (VRPD) เป็นหนึ่งในส่วนขยายของ VRP ซึ่งมีความซับซ้อนมากขึ้น งานวิจัยฉบับแรกเกี่ยวกับปัญหานี้มาคือ “The flying sidekick traveling salesman problems” (FSTSP) [12] ซึ่งเป็นส่วนขยายของ TSP มุ่งเน้นเฉพาะการทำงานของรถบรรทุกคันเดียวที่ติดกับโดรนตัวเดียว ฟังก์ชันวัตถุประสงค์ของคือทำให้เวลาที่ยานพาหนะสุดท้ายน้อยที่สุด (ไม่ว่าจะเป็นรถบรรทุกหรือโดรน) โดยมีฟังก์ชันข้อจำกัดคือ การจำกัดตัวแปร, การไปหาลูกค้าได้เพียงครั้งเดียว, การปล่อย/รับของโดรนและระยะทางที่บินได้ของโดรน

TSP เป็นปัญหาแบบ NP-hard [13] ดังนั้นจึงสามารถสรุปได้ว่า FSTSP เป็นปัญหา NP-hard เช่นกัน Heuristic algorithm จึงถูกพัฒนาขึ้นเพื่อแก้ปัญหา คำตอบถูกเปรียบเทียบกับ คำตอบจากการใช้ซอฟต์แวร์แก้ปัญหา Mixed Integer Program, คำตอบจาก Nearest neighbor algorithm และ คำตอบจาก Sweep algorithm ได้คุณภาพคำตอบที่ดีกว่า Algorithm อื่น ๆ

ในปี 2562 [14] mTSPD ได้เพิ่มฟังก์ชันข้อจำกัดให้กับ FSTSP เพื่อให้โดรนสามารถโต้ตอบกับรถบรรทุกมากกว่าหนึ่งคัน แก้ปัญหาโดยเริ่มจากการใช้ Genetic algorithm, K-mean clustering / Nearest neighbor และ Random cluster ในการสร้าง mTSP tour จากนั้นใช้ Adaptive insertion heuristic ในการหาคำตอบ ผลการคำนวณพบว่า Genetic algorithm ให้คุณภาพของคำตอบได้ดีที่สุด นอกจากนี้การปรับจำนวนรถบรรทุกต่อการจัดส่งทำให้ได้ผลคือยิ่งจำนวนรถบรรทุกมาก จะใช้เวลาในกระบวนการจัดส่งน้อยลง แต่ก็ต้องแลกกับเวลาในการประมวลผลที่มากขึ้น

ผู้วิจัยได้ปรับปรุง mTSPD เรียกว่า Two Echelon Vehicle Routing Problem with Drone [14] งานวิจัยนี้ได้เพิ่มการพิจารณาความจุของยานพาหนะและโดรนสามารถบรรทุกกำหนดให้สามารถบริการลูกค้าได้มากกว่าหนึ่งราย เพื่อแก้ปัญหาขนาดใหญ่ Drone Truck Route Construction (DTRC) และ Large Neighborhood Search (LNS) ถูกนำมาใช้

VRPD [15] เป็นแบบจำลองที่สมจริงมากขึ้นโดยโดรนไม่สามารถกลับไปที่รถบรรทุกผ่าน Node ของลูกค้าได้ต้องเป็น Node เชื่อมต่อหรือโกดังสินค้า (Depot) เท่านั้น

Arc-based model ถูกนำเสนอขึ้นเพื่ออธิบายความซับซ้อนของปัญหา จากนั้นได้เสนอ Path-based model โดยมีความแตกต่าง คือ ตัวแปรของ Path-based model เป็นเส้นทางที่สมบูรณ์ที่เป็นไปได้ของยานพาหนะแต่ละชนิด โดยสร้างจากการใช้

เงื่อนไขเดียวกันกับฟังก์ชันข้อจำกัดของ Arc-Based model ดังนั้น
ในแบบจำลองจึงมีฟังก์ชันข้อจำกัดเพียงสี่ฟังก์ชันเท่านั้น

เนื่องจาก Path-based model เป็นแบบจำลองที่มีความ
ซับซ้อนน้อยแต่มีปริมาณตัวแปรมหาศาล ผู้วิจัยจึงใช้เทคนิค
Column generation และ Branch-and-Price Framework เพื่อ
แก้ปัญหาดังกล่าว ใช้เวลาสูงสุดประมาณ 4 ชั่วโมงครึ่งสำหรับ
ตัวอย่าง 15 Node สามารถประหยัดค่าจัดส่งได้ 20% โดยเฉลี่ยเมื่อ
เทียบกับการไม่ใช้โดรน

2.2. COLUMN GENERATION AND LAGRANGIAN RELAXATION

Column generation [16, 17] วิธีนี้ถูกเสนอเพื่อแก้ปัญหา
Linear relaxation of cutting stock problem ซึ่งเป็นปัญหาใน
การตัดชิ้นส่วนเป็นความยาวที่ต้องการ ตัวอย่างเช่นมีไม้ยาว 13
เมตร ต้องการตัดให้เป็น 3 เมตร 5 เมตรและ 7 เมตรในจำนวนที่
กำหนด ตัวแปรคือรูปแบบของการตัด (เช่น 4 ชิ้น 3 เมตร, 2 ชิ้น 5
เมตร หรือ 5 ชิ้น 7 เมตร) จำนวนตัวแปรที่ได้จากการจัดกลุ่มที่
เป็นไปได้ทั้งหมดนั้นมีจำนวนมากและไม่มีประสิทธิภาพในการ
แก้ปัญหา

Column generation เริ่มด้วยการแยกปัญหาหลักออกเป็น
Restricted master problem (RMP) และ Pricing sub-problem
โดยที่ RMP เป็นแบบจำลองที่เหมือนกับปัญหาดั้งเดิมแต่มีจำนวนตัว
แปรที่ถูกตัดทอนลง และในส่วนของ Pricing sub-problem เป็นการ
คำนวณหาตัวแปรใหม่ที่มี Reduced cost เป็นลบ (ในปัญหาแบบ
Minimization) ในกรณีของ the cutting stock problem Pricing
sub-problem คือ the knapsack problem กระบวนการมี
ดังต่อไปนี้: 1) เริ่มต้น RMP ด้วยตัวแปรจำนวนหนึ่ง 2) แก้ RMP
และรับค่า duals สำหรับ Pricing sub-problem 3) แก้ไขปัญหา
ย่อย คำตอบที่ได้จะเป็นตัวแปรใหม่ (Column) ของ RMP ในกรณีที่
Reduced cost เป็นลบ 4) หากพบ Column ที่มี Reduced cost
เป็นลบให้เพิ่มคอลัมน์ลงใน RMP และทำซ้ำขั้นตอนที่ 2 และ 3
จนกว่าจะไม่พบ Column ที่มี Reduced cost เป็นลบ และคำตอบ
ที่ได้จาก RMP สุดท้ายนั้นคือคำตอบที่ดีที่สุด

ใน Lagrangian relaxation ฟังก์ชันข้อจำกัดจะถูกย้ายไปใส่ไว้
ในฟังก์ชันวัตถุประสงค์โดยมีเวกเตอร์ μ (Lagrangian multiplier)
เป็นสัมประสิทธิ์

ค่า Objective value ของ Lagrangian relaxation จะเป็น
ขอบเขตล่างของปัญหาดั้งเดิม (Master problem) วิธีนี้สามารถ
เพิ่มความเร็วในการสร้าง Column หรือสามารถสร้าง Column
ขึ้นมาได้ [18]

3. การดำเนินการวิจัย

3.1. นิยามปัญหา

3.1.1. แบบจำลองการจัดการเส้นทางยานพาหนะ ร่วมกับโดรน (ARC-BASED/PATH-BASED)

แบบจำลองการจัดการเส้นทางยานพาหนะเป็นปัญหาเครือข่าย
การไหลของยานพาหนะที่กำหนดเป็นกราฟ $G = (N, A)$ ใน
เครือข่ายประกอบด้วย Node $N = \{1, 2, \dots, |N|\}$ และ Arc
 $A = \{(i, j) | i, j \in N, i \neq j\}$ ที่เป็นเส้นทางของยานพาหนะ
แต่ละ Arc มีต้นทุนการเดินทางซึ่งอาจเป็นระยะทางหรือเวลา
เดินทางหรือค่าเดินทาง สิ่งเหล่านี้แสดงด้วยเมทริกซ์สมมาตร
 $D = (D_{ij})$ ฟังก์ชันข้อจำกัดที่สำคัญมีดังต่อไปนี้ 1) สามารถไป
แต่ละ Node ของลูกค้าเพียงครั้งเดียว และต้องตอบสนองความ
ต้องการของลูกค้า 2) เส้นทางรถทั้งหมดเริ่มต้นและสิ้นสุดที่
คลังสินค้า 3) รถแต่ละคันมีความจุที่จำกัด

VRPD สามารถกำหนดให้เป็นกราฟ G เหมือนกับ VRP โดยเซต
ของ Node จะมีข้อเชื่อมต่อ $D = \{1, 2, \dots, |N|\}$ ถูกรวมอยู่
ใน ข้อเชื่อมต่อมีไว้สำหรับการลงจอดของโดรนและรถบรรทุกต้อง
รวบรวมโดรนกลับไปคลังหรือปล่อยโดรนตัวใหม่เพื่อไปส่งสินค้า
ซึ่งทั้งรถและโดรนมีขีดจำกัดของความจุ แต่โดรนจะมีระยะเวลาบิน
สูงสุดเป็นข้อจำกัดด้วย [15]

ยานพาหนะทั้งหมดจะต้องเริ่มต้นและสิ้นสุดที่คลังสินค้า โดรน
สามารถให้บริการลูกค้าในระยะที่อยู่รอบ ๆ คลังในระยะบินได้
สำหรับลูกค้าที่เหลือ รถบรรทุกจะออกจากคลังและให้บริการด้วย
ตัวเองหรือปล่อยโดรนและให้บริการ เมื่อโดรนเสร็จสิ้นการทำงานโด
รนจะลงจอดที่ข้อเชื่อมต่อและถูกเก็บโดยรถบรรทุก อย่างไรก็ตาม
รถบรรทุกสามารถปล่อยโดรนสำรองได้หากโดรนที่พวกเขาปล่อย
ก่อนจะลงจอดหลังจากที่รถบรรทุกมาถึงข้อเชื่อมต่อ โดย
ตั้งสมมติฐานว่าเวลาของการเปลี่ยนแบตเตอรี่และการโหลดพัสดุไป
ยังโดรนนั้นไม่นำมาพิจารณา

ตามที่กล่าวไว้ในบทที่ 2 โมเดลนี้เป็นแบบ NP-Hard ในการ
แก้ปัญหานี้กับลูกค้าจำนวนมากจำเป็นต้องใช้
ทรัพยากรคอมพิวเตอร์จำนวนมาก เนื่องจากฟังก์ชันข้อจำกัดมี
สมบัติ Weak linear relaxation

เพื่อให้ได้ประสิทธิภาพที่ดีขึ้นจึงได้ปรับรูปแบบเป็น Path-
based model ที่ตัวแปรมีสองประเภท: 1) Truck path เป็น
เส้นทางที่สมบูรณ์จากคลังสินค้าไปยังลูกค้า และ/หรือ ไปยังข้อ
เชื่อมต่อจากนั้นกลับไปคลัง 2) Drone path เช่นเดียวกันกับ
Truck path แต่ Drone สามารถเดินทางด้วยตัวเองหรือติดไปกับ
รถบรรทุกได้ ซึ่งคำตอบที่ดีที่สุดคือการผสมผสานระหว่างเส้นทาง
บรรทุกและเส้นทางโดรนที่มีต้นทุนรวมต่ำที่สุด

ลักษณะเฉพาะของแบบจำลองเส้นทางคือตัวแปร (Column)
จำนวนมาก จำเป็นต้องใช้ Column generation และ Branch-
and-Price [8] เพื่อลดการสร้างตัวแปรที่ไม่จำเป็นลงเพื่อให้ได้
คำตอบที่เป็นจำนวนเต็ม

3.1.2. แบบจำลองการจัดการเส้นทางยานพาหนะ ร่วมกับโดรน (TOUR-BASED)

จาก [19] ได้เสนอแบบจำลอง Tour-Based ซึ่งพัฒนามาจากแบบจำลอง Path-Based [15] ตัวแปรแต่ละตัวแสดงถึงเส้นทางที่เป็นไปได้ของรถบรรทุกและโดรนเริ่มจากคลังไปยังลูกค้าทั้งหมดและกลับไปคลังอีกครั้ง ตามวิธีการนี้ VRPD เปลี่ยนเป็นปัญหาแบบ Set covering [20] จำนวนตัวแปรของแบบจำลองนี้มีมากกว่า Arc-based และ Path-based model อย่างมาก แต่ฟังก์ชันข้อกำหนดมีคุณสมบัติ Tight linear relaxation ซึ่งเป็นประโยชน์ใน Branch-and-Bound tree

สมมติฐานของแบบจำลองมีดังต่อไปนี้

1. โดรนสามารถดำเนินการจัดส่งได้เพียงรอบเดียวต่อทัวร์
2. จำนวนรถบรรทุกสูงสุดต่อทัวร์คือ 2 คัน
3. โดรนสามารถลงจอดบนฮับเชื่อมต่อหรือคลังสินค้าเท่านั้น
4. ไม่พิจารณาเวลาในการชาร์จและเปลี่ยนแบตเตอรี่ของโดรน
5. โดรนสามารถออกจากคลังสินค้าหรือจากรถบรรทุกที่ Node ใดก็ได้ หลังจากเสร็จสิ้นการจัดส่ง โดรนสามารถบินกลับไปที่คลังสินค้าหรือลงจอดบนฮับเชื่อมต่อได้
6. ทัวร์จะต้องไม่ละเมิดเงื่อนไขความจุของยานพาหนะทั้งสองและขีดจำกัดระยะทางสูงสุดของโดรน

3.2. แบบจำลองทางคณิตศาสตร์

กำหนดให้กราฟ $G = (N, A)$ ให้ N เป็นเซตของ Node คลังสินค้า o_s และ o_t สำหรับจุดเริ่มและจุดสิ้นสุดของทัวร์ ตามลำดับ Node ฮับเชื่อมต่อ $O = \{o_1, o_2, \dots, o_m\}$ และ Node ลูกค้า $C = \{c_1, c_2, \dots, c_n\}$ กำหนดให้ $A = \{(i, j) | i, j \in N, i \neq j\}$ เป็นเซตของ $arc(i, j)$ แทนเส้นทางของยานพาหนะจาก Node i ไปยัง Node j

ให้ R_c เป็นเซตของทัวร์ที่เป็นไปได้ทั้งหมดของรถบรรทุกและโดรน ให้ δ_i^r เป็น 1 ถ้า $r \in R_c$ มียานพาหนะใด ๆ ไปที่ Node $i \in C$ และเป็น 0 หากตรงกันข้าม ให้ θ_r เป็นค่าขนส่งของทัวร์ r

ให้ F^T เป็นต้นทุนขนส่งของรถบรรทุกและจำนวนรถบรรทุก และ K, k_r^T แทนจำนวนรถบรรทุกที่สามารถใช้ได้ทั้งหมดและจำนวนรถบรรทุกที่ใช้ในทัวร์ตามลำดับ สุดท้ายให้ x_r เป็นตัวแปรไบนารีแทนการเลือกทัวร์ $r \in R_c$

$$\min \sum_{r \in R_c} (c_r + F^T) x_r \quad (1)$$

$$\text{s.t. } \sum_{r \in R_c} \delta_i^r x_r = 1; \forall i \in C \quad (2)$$

$$\sum_{r \in R_c} k_r^T x_r \leq |K| \quad (3)$$

$$x_r \in \{0, 1\}, \forall r \in R_c \quad (4)$$

ฟังก์ชันวัตถุประสงค์ (1) เป็น Minimization เพื่อหาต้นทุน

รวมที่น้อยที่สุด ฟังก์ชันข้อจำกัด (2) กำหนดให้ลูกค้าทุกคนได้รับบริการเพียงครั้งเดียว และฟังก์ชันข้อจำกัด (3) คือจำนวนรถบรรทุกที่ใช้

3.3. COLUMN GENERATION และ BRANCH-AND-PRICE

จาก [19] กำหนดให้ $\tilde{G} = (N, A)$ ให้ N เป็นเซตของ Node คลังสินค้า Node ฮับเชื่อมต่อ O และ Node ลูกค้า C Node ซึ่งมี 3 เลเยอร์ได้แก่ เลเยอร์ของลูกค้าที่รับของจากหนึ่งใน 2 รถบรรทุกหรือโดรน โดย $T^1 = \{c_1, c_2, \dots, c_n\}$ และ $T^2 = \{c_1, c_2, \dots, c_n\}$ ถูกกำหนดเป็นเซตของ Node ลูกค้าที่ให้บริการโดยรถบรรทุกคันแรกและคันที่สอง และ $\Omega = \{c_1, c_2, \dots, c_n\}$ กำหนดเป็นเซตของ Node ลูกค้าที่ให้บริการโดยโดรน

Node ฮับเชื่อมต่อและคลังสินค้ามีเหมือนกันกับ Master problem กำหนดให้ $A_{(i,j)}$ มี 3 เลเยอร์ ได้แก่ arc บรรทุก $a_{(i,j)}^{T^n} : n \in \{1, 2\}$ และ Arc โดรน $a_{(i,j)}^D$ Arc ของรถบรรทุกแต่ละเลเยอร์แสดงถึง Arc ที่เดินทางโดยรถบรรทุกแต่ละคันที่เป็นไปได้ $a_{(i,j)}^{T^n}$ คือเส้นทางที่เป็นไปได้ทั้งหมดของสอง Node จาก $\{o_s, o_t\} \cup O \cup T^n$ โดยไม่ต้องเชื่อมต่อระหว่างเลเยอร์ และ Arc ของโดรนแสดงถึง Arc ที่เดินทางโดยโดรนเอง

จากคำจำกัดความของทัวร์ โดรนสามารถไปยัง Node ได้สามประเภท ได้แก่ Node ที่ปล่อยโดรน, Node ส่งสินค้า และ Node ที่โดรนลงจอด เนื่องจากโดรนสามารถออกจากรถบรรทุกหรือคลังสินค้าได้ด้วยตัวเอง Node เริ่มต้น i ของ Arc โดรนอาจเป็น Node ประเภทใดก็ได้ใน $\bigcup_{n=1}^2 T^n \cup O \cup \{o_s\}$ จากนั้นจึงเป็นการไปส่งของที่ Node ลูกค้า แล้วลงจอดที่ฮับเชื่อมต่อหรือคลังสินค้า เนื่องจากโดรนไม่สามารถลงจอดบน Node ของลูกค้าปลายทางของโดรน j อาจเป็น $(D \cup O \cup \{o_t\})$ แต่ไม่ใช่ T^n ดังนั้นเซตของ Arc รถบรรทุกและ Arc ของโดรนที่เป็นไปได้ทั้งหมดจะสามารถกำหนดได้จาก A^{T^n} คือเซตของ Arc ที่เป็นไปได้ทั้งหมดสำหรับรถบรรทุก n^{th} A^T คือเซตของ Arc รถบรรทุกที่เป็นไปได้ทั้งหมด A^D คือเซตของ Arc โดรนที่เป็นไปได้ทั้งหมด

$$A^{T^n} = \{(i, j) | i \neq j, i \in T^n \cup O \cup \{o_s\}, j \in T^n \cup O \cup \{o_t\}\}; n \in \{1, 2\} \quad (5)$$

$$A^T = \bigcup_{n=1}^2 A^{T^n} \quad (6)$$

$$A^D = \{(i, j) | i \neq j, i \in \bigcup_{n=1}^2 T^n \cup O \cup \{o_s\}, j \in D \cup O \cup \{o_t\}\} \quad (7)$$

ตัวแปรและพารามิเตอร์นิยามไว้ดังนี้

$a_{(i,j)}^{Tn}$: เท่ากับ 1 ถ้ารถบรรทุกผ่าน Arc $(i,j) \in A^{Tn}$ และเป็น 0

$a_{(i,j)}^D$: เท่ากับ 1 ถ้าโดรนผ่าน Arc $(i,j) \in A^D$ และเป็น 0

θ_i : จำนวน Arc จาก Node $i \in N$

β_j : จำนวน Arc ไปยัง Node $j \in N$

$\square_{ij}$: หน่วยน้ำหนักสะสมของรถบรรทุก n^{th} ที่ Node $i \in \{o_s, o_t\} \cup O \cup T^n$

g_i : ระยะทางสะสมที่เดินทางโดยรถบรรทุกที่ $i \in \{o_s, o_t\} \cup O \cup T^n$

z_i : หน่วยน้ำหนักสะสมของโดรนที่ Node $i \in N$

v_i : ระยะบินสะสมของโดรนที่ Node $i \in N$

α_i^n : เท่ากับ 1 ถ้ารถบรรทุกคันที่ n ผ่าน Node $i \in \{o_s, o_t\} \cup O \cup T^n$ และเท่ากับ 0

γ_i^n : ตัวแปรไบนารีสำหรับ if-then constraint สำหรับ $i \in O \cup T^n$

q_i : ความต้องการของลูกค้าของแต่ละ Node $i \in N$ (คลังสินค้าและฮับเชื่อมต่อกับลูกค้าเป็น 0)

$a_{(i,j)}^T$: ระยะทางของ Arc รถบรรทุก $(i,j) \in A^{Tn}$

$a_{(i,j)}^D$: ระยะทางของ Arc โดรน $(i,j) \in A^D$

$c_{(i,j)}^T$: ต้นทุนของ Arc รถบรรทุก $(i,j) \in A^{Tn}$

$c_{(i,j)}^D$: ต้นทุนของ Arc โดรน $(i,j) \in A^D$

L_t : ความจุสำหรับรถบรรทุก

L_d : ความจุสำหรับโดรน

L_r : จำนวนโดรนสูงสุดที่รถบรรทุกสามารถบรรทุกได้ในเส้นทางเดียว

D_d : ระยะทางบินของโดรน

ฟังก์ชันวัตถุประสงค์ของปัญหาย่อยคือการหาเส้นทางที่ทำให้ได้ Reduced cost ที่น้อยที่สุด ให้ $\square$ และ $\square$ เป็นตัวแปร Dual ของฟังก์ชันข้อจำกัด (2) และ (3) ตามลำดับ $\square_{ij}$ และ $\square_{ij}$ แสดงถึงดัชนีของ Node ลูกค้าที่รถบรรทุกและโดรนเข้าพบตามลำดับ

ให้ $\tilde{c}_r$ เป็น Reduced cost ของเส้นทาง

$$\tilde{c}_r = c_r - \sum_{i \in C} \pi_i (\theta_{i_D} + \sum_{n=1}^2 \alpha_i^n) - \sigma k_r^T ; r \in R_c \quad (8)$$

$$\min \tilde{c}_r \quad (9)$$

$$\text{s.t.} \quad \sum_{(i,j) \in A^{Tn}} a_{(i,j)}^T - \sum_{(j,k) \in A^{Tn}} a_{(j,k)}^T = 0; \forall j \in T^n \cup O, \forall n \in \{1,2\} \quad (10)$$

$$y_j^n \geq y_i^n + q_j a_{(i,j)}^T - M(1 - a_{(i,j)}^T); \forall (i,j) \in A^{Tn}, \forall n \in \{1,2\} \quad (11)$$

$$g_j^n \geq g_i^n + d_{(i,j)}^T a_{(i,j)}^T - M(1 - a_{(i,j)}^T); \forall (i,j) \in$$

$$A^{Tn}, \forall n \in \{1,2\} \quad (12)$$

$$y_j^n \leq L_t; \forall i \in \{o_t\} \cup T^n \cup O, \forall n \in \{1,2\} \quad (13)$$

$$\theta_i = \sum_{(i,j) \in A^T} a_{(i,j)}^T + \sum_{(i,j) \in A^D} a_{(i,j)}^D; \forall i \in \{o_s\} \cup \bigcup_{n=1}^2 T^n \cup O \cup D \quad (14)$$

$$\beta_j = \sum_{(i,j) \in A^T} a_{(i,j)}^T + \sum_{(i,j) \in A^D} a_{(i,j)}^D; \forall j \in \{o_t\} \cup \bigcup_{n=1}^2 T^n \cup O \cup D \quad (15)$$

$$\alpha_i^n = \sum_{(i,j) \in A^{Tn}} a_{(i,j)}^T; \forall i \in \{o_s\} \cup T^n \cup O, \forall n \in \{1,2\} \quad (16)$$

$$\sum_{(i,j) \in A^T} a_{(i,j)}^T \geq 1; \forall i \in \{o_s\} \quad (17)$$

$$\sum_{(i,j) \in A^D} a_{(i,j)}^D - \sum_{(j,k) \in A^D} a_{(j,k)}^D = 0; \forall j \in D \quad (18)$$

$$z_j \geq z_i + q_j a_{(i,j)}^D - M(1 - a_{(i,j)}^D); \forall (i,j) \in A^D \quad (19)$$

$$v_j \geq v_i + d_{(i,j)}^D a_{(i,j)}^D - M(1 - a_{(i,j)}^D); \forall (i,j) \in A^D \quad (20)$$

$$z_i \leq L_D; \forall i \in D \quad (21)$$

$$v_i \leq D_D; \forall i \in D \cup O \cup \{o_t\} \quad (22)$$

$$\theta_i \leq 1; \forall i \in D \quad (23)$$

$$\sum_{i \in T^n} (\theta_i - \alpha_i^n) \leq L_R; \forall n \in \{1,2\} \quad (24)$$

$$\sum_{i \in O} (\beta_i - \sum_{n=1}^2 \alpha_i^n) \leq L_R \quad (25)$$

$$\sum_{(i,j) \in A^D} a_{(i,j)}^D \leq M \alpha_i^n; \forall i \in T^n, n \in \{1,2\} \quad (26)$$

$$\sum_{(i,j) \in A^D} a_{(i,j)}^D \leq M (\sum_{n=1}^2 \alpha_i^n); \forall i \in O \quad (27)$$

$$\sum_{(i,j) \in A^D} a_{(i,j)}^D \leq M (1 - \sum_{n=1}^2 \alpha_k^n); \forall (j,k) \in \{(j,k) | j = k\} \quad (28)$$

$$\sum_{(i,j) \in A^D} a_{(i,j)}^D \leq M (\sum_{n=1}^2 \alpha_i^n); \forall j \in O \quad (29)$$

$$\sum_{(i,j) \in A^D} a_{(i,j)}^D \leq M (1 - \gamma_i^n); \forall i \in T^n \cup O, \forall n \in \{1,2\} \quad (30)$$

$$g_j^n - g_i^n \leq M \gamma_i^n; \forall (i,j) \in \{(i,j) | i \neq j, i \in T^n \cup O, j \in O\}, \forall n \in \{1,2\} \quad (31)$$

ฟังก์ชันข้อจำกัด (10) คือการกำหนดให้จำนวนรถบรรทุกเข้าและออกจาก Node เท่ากัน ฟังก์ชันข้อจำกัด (11) และ (12) ตอบสนองความเป็นไปได้ของความสามารถในการบรรทุกและระยะทางสะสมซึ่งตัวแปรน้ำหนักสะสม y_j และตัวแปรระยะทางสะสม g_j ต้องมากกว่าหรือเท่ากับ Node ก่อนหน้า ฟังก์ชันข้อจำกัด (13) จำกัดความจุของรถบรรทุกแต่ละคัน

ฟังก์ชันข้อจำกัด (14) - (16) กำหนดตัวแปร θ, β และ α ฟังก์ชันข้อจำกัด (17) จำกัดว่าแต่ละเส้นทางต้องปล่อยรถบรรทุกอย่างน้อยหนึ่งคัน

ฟังก์ชันข้อจำกัด (18) คือการอนุรักษ์ฟังก์ชันข้อจำกัดการไหลเพื่อให้แน่ใจว่าจำนวนโดรนเข้าและออกจาก Node เท่ากัน ฟังก์ชันข้อจำกัด (19), (20) จำกัดความจุของโดรนและระยะทางการบินเทียบกับ Node ก่อนหน้า โดยตัวแปรน้ำหนักสะสม Z_j และตัวแปรระยะบินสะสม V_j ต้องสูงกว่าหรือเท่ากับของ Node ก่อนหน้า ฟังก์ชันข้อจำกัด (21) จำกัดความจุของโดรนแต่ละตัว ฟังก์ชันข้อจำกัด (22) จำกัดระยะทางบินของโดรนแต่ละตัว ฟังก์ชันข้อจำกัด (23) กำหนดให้ลูกค้าแต่ละรายจะได้รับบริการด้วยโดรนเพียงครั้งเดียวและป้องกันไม่ให้โดรนบินไปซ้ำ ฟังก์ชันข้อจำกัด (24), (25) ตรวจสอบให้แน่ใจว่ารถบรรทุกไม่สามารถบรรทุกโดรนเกินกว่าที่จำกัดไว้

ฟังก์ชันข้อจำกัด (26) - (27) จำกัดความเป็นไปได้ในการออกและลงจอดของโดรนในทัวร์ หากโดรนออกจาก Node รถบรรทุกหรือ Node ฮับเชื่อมต่อรถบรรทุกจะต้องไปที่ Node นั้นด้วย ฟังก์ชันข้อจำกัด (28) ป้องกันไม่ให้โดรนและรถบรรทุกเข้าซ้ำ ฟังก์ชันข้อจำกัด (29) ช่วยให้เห็นใจได้ว่าโดรนบนฮับเชื่อมต่อจำเป็นต้องมีรถบรรทุกอยู่ด้วย ฟังก์ชันข้อจำกัด (30), (31) ป้องกันไม่ให้เกิดการเดินทางที่ไม่สามารถทำได้ โดยโดรนลงจอดบนฮับเชื่อมต่อที่รถบรรทุกขับผ่านไปแล้ว Node ที่โดรนออกจากต้องมีระยะทางสะสมของรถบรรทุกน้อยกว่าหรือเท่ากับระยะทางสะสมของรถบรรทุกใน Node ฮับเชื่อมต่อ

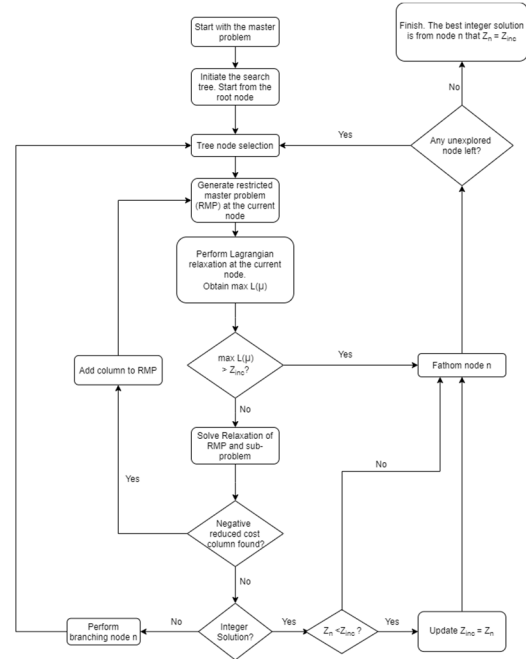
เพื่อให้ได้คำตอบเป็นจำนวนเต็ม จำเป็นต้องใช้ Branch-and-Price Framework ดังแสดงไว้ในรูปที่ 1

ในรูปที่ 1 Z_n คือคำตอบของ Node และ Z_{inc} เป็นคำตอบที่ดีที่สุดในปัจจุบัน โดยเริ่มแรก Z_{inc} มีค่าเป็นอนันต์

Branching rule ถูกนำเสนอโดย [21] จากนั้น [8] นำมาใช้ใน Branch-and-Price คำตอบที่ไม่เป็นจำนวนเต็มจะถูกแปลงเป็น Arc-based model จากนั้นจึงระบุ Arc ที่ไม่เป็นจำนวนเต็มจากฟังก์ชันข้อจำกัด แล้วจึง Branch โดยบังคับให้ Flow ใน Arc นั้นเป็น 0 หรือ 1

Lagrangian relaxation ใช้เพื่อลดการเกิด Tailing-off effect จาก Column generation โดยการสร้างขอบเขตล่าง (หรือขอบเขตบนในกรณีของ Maximization) อ้างอิงจากทฤษฎีบทต่อไปนี้ [22]

พิจารณาแบบจำลองเชิงเส้นต่อไปนี้:



รูปที่ 1 Branch-and-Price with Lagrangian relaxation

$$z^* = \min cx \quad (32)$$

$$s. t. \quad Ax = b \quad (33)$$

$$x \in X \quad (34)$$

“ทฤษฎีบท 1 (Lagrangian Bounding Principle) สำหรับเวกเตอร์ μ ของตัวคูณ Lagrangian ค่า $L(\mu)$ ของฟังก์ชัน Lagrangian คือขอบเขตล่างของค่าฟังก์ชันวัตถุประสงค์ที่เหมาะสมของ Master problem (P) ”

“พิสูจน์ เนื่องจาก $Ax = b$ สำหรับทุกวิธีแก้ปัญหาที่เป็นไปได้ไปยัง (P) สำหรับเวกเตอร์ μ ของตัวคูณ Lagrangian ใด ๆ $z^* = \min\{cx: Ax = b, x \in X\}$ เนื่องจากการลบฟังก์ชันข้อจำกัด $Ax = b$ ไม่สามารถนำไปสู่การเพิ่มค่าของฟังก์ชันวัตถุประสงค์ได้ $(z^* \geq \min\{cx + \mu(Ax = b): x \in X\} = L(\mu))$ ”

จากนั้นเพื่อให้เข้าใกล้ Z^* ให้มากที่สุดจำเป็นต้องใช้การแก้ปัญหา Lagrangian multiplier โดย Gradient search

$$L^* = \max_{\mu} L(\mu) \quad (35)$$

จาก Path-based model สามารถกำหนดฟังก์ชันวัตถุประสงค์ของ Lagrangian ได้ดังต่อไปนี้

$$L(\mu) = \min \sum_{r \in R_c} (c_r + F^T) x_r - \sum_{i \in C} \mu_i \sum_{r \in R_c} (\delta_i^r x_r - 1) \quad (36)$$

ภายใต้ฟังก์ชันข้อจำกัด (3) และ (4)

จาก (36)

$$L(\mu) = \min \sum_{i \in C} \sum_{r \in R_c} (c_r + F^T - \delta_i^r \mu_i) x_r - \sum_{i \in C} \mu_i \quad (37)$$

ภายใต้ฟังก์ชันข้อจำกัด เรื่อง (3) และ (4)

กำหนดค่าเริ่มต้นของเวกเตอร์ μ (ในงานวิจัยนี้คือ 1) และ
แก้ปัญหา Lagrangian multiplier

ในการวนซ้ำแต่ละครั้งจะเปลี่ยนค่า μ ด้วยการใช้พารามิเตอร์
 $\theta^k = 1/k$ ร่วมกับ (38) โดย k แทนครั้งของการวนซ้ำ

$$\mu^{k+1} = \mu^k + \theta^k (\sum_{r \in R_c} \delta_r^T x_r - 1) \quad (38)$$

หลังจากการทำซ้ำ 100 ครั้งแล้วจึงเก็บค่าสูงสุดของ $L(\mu)$
เพื่อเป็นขอบเขตล่างของ Node นั้น ๆ ใน Branch-and-Price tree

4. การทดลองและผลการทดลอง

งานวิจัยนี้จะทดสอบกับตัวอย่างที่มี Customer node จำนวน
8 และ 10 Node แบบละ 10 ตัวอย่าง โดยแต่ละตัวอย่างจะมี
จำนวน Node เชื่อมต่อและคลังสินค้าตัวอย่างละ 1 Node โดยใช้
การสุ่มตำแหน่งและอุปสงค์ของแต่ละ Node ซึ่งอุปสงค์จะอยู่ในช่วง
10 ถึง 50 กิโลกรัม

กำหนดให้ความเร็วของรถบรรทุกและโดรนเท่ากับ 40
กิโลเมตรต่อชั่วโมง มีต้นทุนการเดินทางเท่ากับ 2.50 และ 0.65
บาทต่อนาที่ตามลำดับ โดยรถบรรทุกมีต้นทุนคงที่เท่ากับ 600 บาท
ต่อคัน

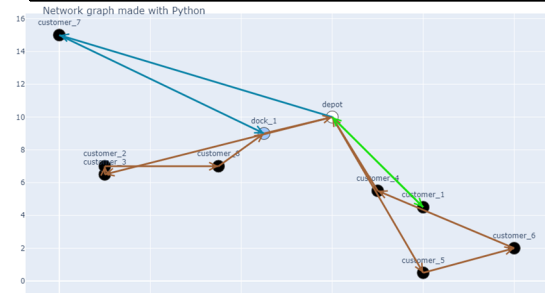
รถบรรทุกสามารถบรรทุกน้ำหนักได้สูงสุด 100 กิโลกรัม
บรรทุกโดรนได้ 5 ลำ ในขณะที่โดรนบรรทุกน้ำหนักได้สูงสุด 20
กิโลกรัม บินได้ไกลสูงสุด 20 กิโลเมตร

ผลการทดลองดังตารางที่ 1 ใช้ภาษา Python เขียน Algorithm
ที่นำเสนอไว้ข้างต้นบนคอมพิวเตอร์ที่มีชิปประมวลผล Ryzen 7
4800H Ram 16 GB โดยมี Gurobi 9.1 เป็น MIP Solver

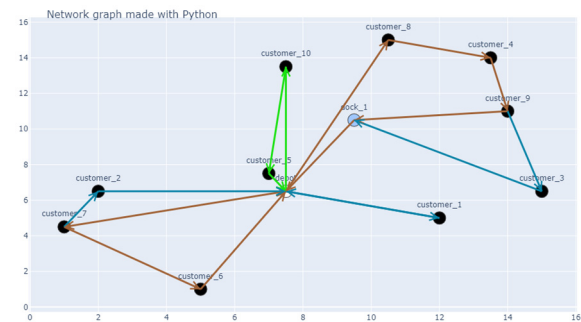
ตารางที่ 1 ผลการทดลอง

Customer	Node Explored (Without LG, With LG)	Time (s) (Without LG, With LG)	%Saving
1	8	5, 5	19.96, 19.68
2	8	11, 7	38.31, 25.62
3	8	5, 5	27.27, 24.68
4	8	17, 17	48.39, 36.45
5	8	7, 3	131.33, 73.90
6	8	9, 9	76.64, 68.94
7	8	26, 25	74.05, 61.76
8	8	57, 53	126.68, 99.43
9	8	19, 15	101.90, 74.76
10	8	11, 11	56.57, 45.35
11	10	23, 21	368.59, 221.74
12	10	13, 13	355.19, 323.98
13	10	61, 39	294.58, 198.27
14	10	13, 3	586.48, 227.83

Customer	Node Explored (Without LG, With LG)	Time (s) (Without LG, With LG)	%Saving
15	10	7, 7	345.35, 322.63
16	10	33, 31	1065.70, 1015.87
17	10	5, 5	430.45, 274.86
18	10	29, 13	2972.84, 2000.68
19	10	3, 3	254.79, 197.89
20	10	59, 47	1466.05, 1061.07



รูปที่ 2 คำตอบของตัวอย่างที่ 2



รูปที่ 3 คำตอบของตัวอย่างที่ 13

เส้นทางที่ดีที่สุดจากตัวอย่างที่มี 8 และ 10 Customer node
แสดงที่รูปที่ 2 และ 3 ตามลำดับ โดยเส้นสีน้ำตาลและเขียวแทน
เส้นทางของรถบรรทุกและสีน้ำเงินแทนเส้นทางของโดรน

5. สรุปผลการวิจัยและข้อเสนอแนะ

งานวิจัยนี้เสนอการใช้ Lagrangian relaxation เพื่อหา
 $\max L(\mu)$ ค่าที่ใกล้เคียง Optimal solution ในแต่ละ Node
ของ Branch-and-Price สำหรับปัญหา Tour-based model of
vehicle routing problem with drones เทคนิคนี้สามารถลด
เวลาการคำนวณได้จากการที่ไม่จำเป็นต้องทำ Column
generation เมื่อ $\max L(\mu)$ ของ Node นั้นมีค่าเกิน
Incumbent solution และสามารถลดจำนวน Node ที่ต้อง
คำนวณได้ เนื่องจากไม่จำเป็นต้องแตก Branch เพิ่ม

สำหรับตัวอย่าง 8 และ 10 Customer node ใช้เวลาในการ
คำนวณโดยไม่ใช้ Lagrangian relaxation ตั้งแต่ 19.96 วินาที ถึง
131.33 วินาที และ 254.79 วินาที ถึง 2972.84 วินาที ตามลำดับ

ซึ่งการใช้ Lagrangian relaxation สามารถลดเวลาการคำนวณได้ตั้งแต่ร้อยละ 1.4 ถึง 61.15

จากการทดสอบในแต่ละตัวอย่างพบว่าในเวลาที่ใช้ในการทำ Column generation เป็นเวลาส่วนใหญ่ของทั้งกระบวนการการหาคำตอบ สำหรับงานวิจัยในอนาคตสามารถนำเทคนิคการ Stabilization เช่น Primal-Dual strategy, Weighted Dantzig-Wolfe decomposition มาเพื่อเพิ่มประสิทธิภาพแบบจำลองนี้

6. รายการอ้างอิง

- [1] K. Jacobs, et al., *The last-mile delivery challenge*. 2019.
- [2] Michat, M., et al., *Five Forces Transforming Transport & Logistics* PwC CEE Transport & Logistics Trend Book 2019. 2019.
- [3] Popper, B. *Drones could make Amazon's dream of free delivery profitable*. 2015 Jun 3, 2015 [cited 2019 Nov 18]; Available from: <https://www.theverge.com/2015/6/3/8719659/amazon-prime-air-drone-delivery-profit-free-shipping-small-items>.
- [4] Rakuten, I. *Rakuten to Launch "Sora Raku" Delivery Service Utilizing Drones*. 2016 Apr 25, 2016 [cited 2019 Nov 18]; Available from: https://global.rakuten.com/corp/news/press/2016/0425_01.html.
- [5] Rakuten, I. *Rakuten and JD.com to Collaborate to Create Unmanned Delivery Solutions in Japan*. 2019 [cited 2019 Dec 02, 2019]; Available from: https://global.rakuten.com/corp/news/press/2019/0221_04.html.
- [6] Feillet, D., A tutorial on column generation and branch-and-price for vehicle routing problems. *4or*, 2010. 8(4): p. 407-424.
- [7] Baldacci, R., P. Toth, and D. Vigo, *Exact algorithms for routing problems under vehicle capacity constraints*. *Annals of Operations Research*, 2010. 175(1): p. 213-245.
- [8] Barnhart, C., et al., *Branch-and-price: Column generation for solving huge integer programs*. *Operations research*, 1998. 46(3): p. 316-329.
- [9] Ryan, D. and E. Foster, *An Integer Programming Approach to Scheduling*. 1981.
- [10] Lübbecke, M.E. and J. Desrosiers, *Selected topics in column generation*. *Operations research*, 2005. 53(6): p. 1007-1023.
- [11] Golden, B.L., S. Raghavan, and E.A. Wasil, *The vehicle routing problem: latest advances and new challenges*. Vol. 43. 2008: Springer Science & Business Media.
- [12] Murray, C.C. and A.G. Chu, *The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery*. *Transportation Research Part C: Emerging Technologies*, 2015. 54: p. 86-109.
- [13] Kumar, S.N. and R. Panneerselvam, *A survey on the vehicle routing problem and its variants*. *Intelligent Information Management*, 2012. 4(03): p. 66.
- [14] Kitjacharoenchai, P., B.-C. Min, and S. Lee, *Two echelon vehicle routing problem with drones in last mile delivery*. *International Journal of Production Economics*, 2019: p. 107598.
- [15] Wang, Z. and J.-B. Sheu, *Vehicle routing problem with drones*. *Transportation Research Part B: Methodological*, 2019. 122: p. 350-364.
- [16] Dantzig, G.B. and P. Wolfe, *Decomposition principle for linear programs*. *Operations research*, 1960. 8(1): p. 101-111.
- [17] Gilmore, P.C. and R.E. Gomory, *A linear programming approach to the cutting-stock problem*. *Operations research*, 1961. 9(6): p. 849-859.
- [18] Huisman, D., et al., *Combining Column Generation and Lagrangian Relaxation*. 1970. p. 247-270.
- [19] Pichayavet, R., P. Charoenwut, and M. Lohatepanont, *A Tour-based Model for Solving Vehicle Routing Problem with Drones*. 2019.
- [20] Winston, W.L. and J.B. Goldberg, *Operations Research: Applications and Algorithms*. 2004: Thomson Brooks/Cole.
- [21] Ryan, D. and B. Foster, In A. Wren (Ed.), *An integer programming approach to scheduling*, *Computer scheduling of public transport urban passenger vehicle and crew scheduling* (pp. 269-280). 1981, North Holland, Amsterdam.
- [22] Ahuja, R.K., T.L. Magnanti, and J.B. Orlin, *Network flows: theory, algorithms, and applications*. 1993: Prentice Hall.