บทความวิจัย

การปรับปรุงขั้นตอนวิธีค้นหาแบบแมลงหวี่เพื่อแก้ปัญหาการจัดตารางการเก็บเกี่ยวพืช

ปวีร์ ศิริรักษ์* และ อภิชา โคตรค้างพู
สาขาวิชาวิศวกรรมอุตสาหการ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

* ผู้นิพนธ์ประสานงาน โทรศัพท์ 09 6152 5165 อีเมล: pavee@g.sut.ac.th
DOI: 10.14416/j.kmutnb.2024.10.012
รับเมื่อ 16 ตุลาคม 2566 แก้ไขเมื่อ 29 มกราคม 2567 ตอบรับเมื่อ 14 กุมภาพันธ์ 2567 เผยแพร่ออนไลน์ 7 ตุลาคม 2567
© 2024 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอการพัฒนาอัลกอริทึมด้วยวิธีค้นหาแบบแมลงหวี่ เพื่อใช้ในการแก้ปัญหาการจัดตารางการเก็บเกี่ยวพืชของบริษัทแปรรูปสินค้าเกษตรให้มีผลกำไรตอบแทนต่อปีสูงสุด ขั้นตอนวิธีค้นหาแบบแมลงหวี่ถูกปรับให้เหมาะสมในรูปแบบของเมทริกซ์ เพื่อกำหนดพื้นที่เก็บเกี่ยวในแต่ละเดือนให้มีปริมาณผลผลิตสูงสุดต่อปี ตัวอย่างแบบจำลองทางคณิตศาสตร์สำหรับปัญหาการเก็บเกี่ยวพืชจำนวน 12 ถึง 60 แปลง นำมาใช้เป็นกรณีศึกษาเพื่อตรวจสอบความสามารถของอัลกอริทึมในปัญหาขนาดเล็ก ขนาดกลาง และขนาดใหญ่ เมทริกซ์คำตอบได้รับการปรับปรุงโดยการพิจารณาจุดตัดเพื่อป้องกันการติดคำตอบท้องถิ่น และเพิ่มขีดความสามารถในการค้นหาคำตอบ ผลการทดลองพบว่า วิธีการค้นหาแบบแมลงหวี่สามารถทำให้บริษัทได้รับปริมาณผลผลิตเพียงพอต่อการแปรรูปในแต่ละเดือน รวมทั้งได้รับผลกำไรตอบแทนต่อปีสูงสุด เทียบเท่ากับการจำลองปัญหาผ่านโปรแกรมสำเร็จรูป GAMS ที่สามารถหาคำตอบที่ดีที่สุดได้ในทุก ๆ ขนาดปัญหา

คำสำคัญ: วิธีค้นหาแบบแมลงหวี่ การจัดตารางการเก็บเกี่ยว เมตะฮิวริสติก

การอ้างอิงบทความ: ปวีร์ ศิริรักษ์ และ อภิชา โคตรค้างพู, "การปรับปรุงขั้นตอนวิธีค้นหาแบบแมลงหวี่เพื่อแก้ปัญหาการจัดตารางการเก็บเกี่ยวพืช," *วารสารวิชาการพระจอมเกล้าพระนครเหนือ*, ปีที่ 35, ฉบับที่ 3, หน้า 1–16, เลขที่บทความ 253-047377, ก.ค.–ก.ย. 2568.

Research Article

The Modified Fruit Fly Optimization Algorithm for Solving Harvest Scheduling

Pavee Siriruk* and Apicha Kotekangpoo

Department of Industrial Engineering, Institute of Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand

* Corresponding Author, Tel. 09–6152–5165, E–mail: pavee@g.sut.ac.th
DOI: 10.14416/j.kmutnb.2024.10.012
Received 16 October 2023; Revised 29 January 2024; Accepted 14 February 2024; Published online: 7 October 2024
© 2024 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

Abstract

This research develops the Fruit Fly Optimization Algorithm (FOA) to achieve the objectives of maximizing annual profits in a processing company's crop harvest scheduling problem. The fruit fly algorithm is adapted as a matrix to determine the monthly harvest area with the highest annual yield. The crop harvesting problems ranging from 12–60 plots are used to verify the algorithm's capability in small, medium, and large-scale problems. The matrix intersection technique is used to test different problem sizes in order to escape local optima and develop better answers. The results show that the fruit fly optimization algorithm allows the company to obtain sufficient monthly crops for processing. In addition, the highest annual profit has the same optimal value as formulating mathematical models through GAMS in all problem sizes.

Keywords: Fruit Fly Optimization Algorithm, Harvest Scheduling, Meta-heuristics

Please cite this article as: P. Siriruk and A. Kotekangpoo, "The modified fruit fly optimization algorithm for solving harvest scheduling," *The Journal of KMUTNB*, vol. 35, no. 3, pp. 1–16, ID. 253-047377, Jul.–Sep. 2025 (in Thai).

1. บทนำ

การเก็บเกี่ยวพืชควรทำในช่วงเวลาที่เหมาะสมตามวัตถุประสงค์ของการนำไปใช้ประโยชน์ เช่น การบริโภคสดหรือการแปรรูปเป็นผลิตภัณฑ์ ซึ่งจะทำให้ได้ผลผลิตที่มีคุณภาพ คุณค่าทางอาหาร และความสดสูงสุด โดยทั่วไปการทยอยเก็บเกี่ยวพืชตลอดฤดูการเพาะปลูกจะให้ผลดีกว่าการเก็บเกี่ยวหมดในครั้งเดียว อย่างไรก็ตามเกษตรกรที่เน้นปลูกพืชเพื่อการค้ามักเก็บเกี่ยวผลผลิตเพียงหนึ่งครั้ง เพื่อประหยัดแรงงานและใช้ประโยชน์ที่ดินให้คุ้มค่ามากที่สุด ในระหว่างการเก็บเกี่ยวพืชนั้นจำเป็นต้องมีความระมัดระวังอย่างสูง เพราะอาจส่งผลให้เกิดการสูญเสียปริมาณและคุณภาพของผลผลิตได้ [1] ในภาคการผลิตขนาดใหญ่ ปริมาณผลผลิตพืชจะถูกจำกัดด้วยกำลังการผลิตของโรงงานเพื่อแปรรูปเป็นผลิตภัณฑ์ต่าง ๆ สำหรับการบริโภค การกำหนดตารางการเก็บเกี่ยวพืชเพื่อการแปรรูปจะสามารถช่วยลดความสูญเสียจากการผลิตมากหรือน้อยเกินไป เนื่องจากเป็นการควบคุมการจัดหาปริมาณวัตถุดิบให้เกิดความสม่ำเสมอในแต่ละช่วงเวลา ซึ่งส่งผลให้ต้นทุนด้านแรงงานและเครื่องจักรลดลง [2]

ในการหาคำตอบที่ดีที่สุดสำหรับปัญหาการจัดตารางการผลิตมักมีความยุ่งยากซับซ้อน เช่น ปัญหาการจัดตารางการผลิตแบบตามสั่ง (Job Shop Scheduling) ซึ่งเป็นปัญหาแบบ NP-hard ใช้เวลานานในการค้นหาคำตอบที่ดีที่สุด (Optimal Solution) [3] วิธีการเมตะฮิวริสติก (Meta-heuristics Method) จึงถูกพัฒนาขึ้นเพื่อแก้ไขปัญหาที่มีความซับซ้อนและค้นหาคำตอบอย่างรวดเร็ว โดยนิยมใช้เลขสุ่มค้นหาพื้นที่คำตอบที่เป็นไปได้ [4] ในการแก้ปัญหาการจัดตารางทางการเกษตรมีผู้เสนอวิธีการต่าง ๆ ไว้มากมาย เช่น Richards และ Gunn [5] เสนอแบบจำลองการกำหนดการในการเก็บเกี่ยวไม้และการก่อสร้างถนนให้มีต้นทุนต่ำสุด ใน ค.ศ. 2000 โดยใช้วิธีการค้นหาแบบทาบู (TS) ต่อมาใน ค.ศ. 2010 Pant และคณะ [6] แสดงให้เห็นว่าวิธีกลุ่มอนุภาค (PSO) วิธีวิวัฒนาการโดยใช้ผลต่าง (DE) และวิธีเชิงวิวัฒนาการ (EP) สามารถให้ประสิทธิภาพเทียบเท่าหรือดีกว่าการคำนวณจากโปรแกรมสำเร็จรูป โดยนำไปประยุกต์เพื่อแก้ปัญหาการวางแผนพืชผลที่พิจารณาการใช้ประโยชน์จากชลประทาน ใน ค.ศ. 2012 Stray และคณะ [7] พัฒนาระบบการเก็บเกี่ยวอ้อยในประเทศแอฟริกาใต้บนพื้นที่แปลงขนาดใหญ่ซึ่งมีเงื่อนไขในการปลูกพืชเปลี่ยนแปลงบ่อยครั้ง โดยใช้วิธีการค้นหาแบบทาบู (TS) ที่สามารถหาคำตอบที่เป็นไปได้ภายในระยะเวลาที่เหมาะสม Filho และคณะ [8] นำเสนออัลกอริทึมด้วยขั้นตอนวิธีเชิงพันธุกรรม (GA) เพื่อแก้ปัญหาการปลูกพืชหมุนเวียนที่คำนึงถึงการไถกลบพืชสด ระยะเวลาการปล่อยแปลงว่าง รวมทั้งข้อจำกัดด้านอุปสงค์เพื่อทำกำไรสูงสุด อัลกอริทึมนี้สามารถแสดงผลคำตอบที่เป็นไปได้ในเวลาที่สมเหตุสมผล ต่อมาใน ค.ศ. 2015 Neungmatcha และคณะ [9] วางแผนเส้นทางสำหรับรถเก็บเกี่ยวและขนส่งอ้อยโดยใช้วิธีกลุ่มอนุภาค (PSO) ทำให้สามารถประหยัดต้นทุนค่าขนส่งและช่วยให้โรงงานได้รับปริมาณผลผลิตอ้อยอย่างสม่ำเสมอ อีกสองปีถัดมา Rath และคณะ [10] ได้ประยุกต์ใช้การค้นหาแบบนกกาเหว่า (CS) สำหรับการวางแผนการปลูกพืชในฤดูฝน โดยคำนึงถึงความต้องการน้ำ ระบบชลประทาน และความพร้อมของพื้นที่ เมื่อทดสอบเปรียบเทียบกับโปรแกรมสำเร็จรูปพบว่า วิธีการค้นหาแบบนกกาเหว่าให้ผลกำไรมากกว่า 1.9 เปอร์เซ็นต์ ใน ค.ศ. 2018 Afifah และคณะ [11] พัฒนาอัลกอริทึมการอบเหนียวจำลอง (SA) ในการจัดตารางการเก็บเกี่ยวอ้อยเพื่อจำกัดปริมาณการเก็บเกี่ยวอ้อยให้สอดคล้องกับกำลังการผลิตของโรงงาน อัลกอริทึมนี้ยังสามารถลดค่าเฉลี่ยจำนวนรถบรรทุกที่ต้องใช้ในการเก็บเกี่ยวอ้อยในแต่ละวัน นอกจากนี้ Ketsripongsa [12] สร้างอัลกอริทึมสำหรับการวางแผนเลือกพื้นที่ที่เหมาะสมในเพาะปลูกพืชเศรษฐกิจด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) เพื่อเพิ่มกำไรสูงสุดแก่เกษตรกร ผลลัพธ์บ่งชี้ให้เห็นว่าวิธีการ DE สามารถคำนวณได้ในระยะเวลาที่เร็วกว่า แม้ว่าจะให้ค่าผลกำไรเท่ากันเมื่อทดสอบปัญหาด้วยซอฟท์แวร์สำเร็จรูป และล่าสุดใน ค.ศ. 2023 Kommadath และคณะ [13] ใช้วิธีวิวัฒนาการโดยใช้ผลต่างด้วยกลยุทธ์การกลายพันธุ์ที่มีประชากรหลายกลุ่ม (MPEDE) แก้ปัญหาการจัดตารางการแปรรูปผักและการบรรจุภัณฑ์ เพื่อลดต้นทุนและลดเวลาการผลิตโดยรวมให้ต่ำที่สุด

วิธีการแบบไฮบริด (Hybrid Algorithm) เป็นวิธีการ

ที่รวมข้อดีของหลาย ๆ อัลกอริทึมไว้ด้วยกัน มีวัตถุประสงค์เพื่อปรับปรุงการค้นหาคำตอบให้มีประสิทธิภาพทั้งในแง่ของความเร็ว และความแม่นยำในการคำนวณมากยิ่งขึ้น [14] สามารถพบได้ในหลายงานวิจัย เช่น ใน ค.ศ. 2019 Mogale และคณะ [15] นำเสนอวิธี NK Hybrid Genetic Algorithm (NKHGA) มาใช้ในปัญหาการจัดกลุ่มหมู่บ้านเพื่อพัฒนานโยบายจัดซื้อธัญพืชในอินเดีย ผลการวิเคราะห์บ่งชี้ว่า ผู้กำหนดนโยบายควรมุ่งเน้นการสร้างจุดรับซื้อให้เพียงพอก่อนเริ่มฤดูกาลเก็บเกี่ยว ซึ่งจะช่วยให้เกษตรกรได้รับการประกันราคาที่สมเหตุสมผล ต่อมาใน ค.ศ. 2022 Chouhan และคณะ [16] ได้ทำการพิจารณาของเสียและผลิตภัณฑ์พลอยได้ของอุตสาหกรรมอ้อยเพื่อลดต้นทุนรวมของบริษัท สำหรับการทดสอบปัญหาขนาดใหญ่พบว่า วิธี Genetic-Social Engineering Optimizer (GASEO) ให้ผลลัพธ์ดีกว่าอัลกอริทึม Genetic-Simulated Annealing (GASA) และในปีเดียวกัน Florentino และคณะ [17] ใช้วิธีเชิงพันธุกรรม (GA) ร่วมกับการค้นหาพื้นที่ใกล้เคียงแบบแปรผัน (VNS) แก้ปัญหาการคัดเลือกพันธุ์อ้อย และกำหนดช่วงเวลาในการเพาะปลูกและเก็บเกี่ยวอ้อยที่เหมาะสม วิธีการแบบไฮบริดนี้ส่งผลให้เวลาคำนวณโดยเฉลี่ยลดลง เมื่อเทียบกับวิธีการหาคำตอบที่ดีที่สุด (Exact Method)

จากการทบทวนวรรณกรรมเพิ่มเติมพบว่า วิธีค้นหาแบบแมลงหวี่ (Fruit Fly Optimization Algorithm; FOA) เป็นวิธีการที่อยู่ในกลุ่มเมตาฮิวริสติกใหม่ ซึ่งมีจุดเด่น คือ พารามิเตอร์ไม่ซับซ้อน มีหลักการชัดเจน และใช้งานง่าย [18] จึงถูกนำไปประยุกต์ใช้ในการหาค่าที่ดีสุดหลายปัญหา เช่น Tapao and Cheerarot [19] ศึกษาการออกแบบฐานรากคอนกรีตเสริมเหล็กด้วยวิธี FOA โดยผลการออกแบบส่งผลให้ประหยัดต้นทุนมากกว่าวิธีทั่วไปถึง 23 เปอร์เซ็นต์ Babaoğlu [20] ใช้ FOA แก้ไขปัญหาการบรรจุสินค้าทรงสี่เหลี่ยมผืนผ้าลงในกล่องแบบสองมิติ เพื่อลดความสูงในการบรรจุให้น้อยที่สุด FOA ได้พิสูจน์ว่าเป็นหนึ่งในวิธีที่มีประสิทธิภาพด้านคำตอบเมื่อมีการจัดเรียงแบบมุมล่างซ้าย Iscan และ Gunduz [21] นำวิธี FOA ไปแก้ไขปัญหาการเดินทางของพนักงานขายให้มีระยะทางสั้นที่สุด โดย FOA สามารถให้คำตอบในปัญหาขนาดเล็กและกลาง อีกทั้ง Wang และคณะ [22] สนับสนุนว่าวิธี FOA สามารถแก้ปัญหาขนาดใหญ่ได้ โดยนำไปประยุกต์ใช้กับปัญหาถุงกระสอบแบบหลายมิติที่ต้องบรรจุสินค้าได้อย่างเพียงพอและสร้างผลกำไรสูงสุด นอกจากนี้ การปรับปรุงวิธีค้นหาแบบแมลงหวี่สามารถเพิ่มประสิทธิภาพในการค้นหาคำตอบปัญหาที่มีความซับซ้อนได้ เช่น Mhuadthongon [23] ใช้วิธีค้นหาแบบแมลงหวี่ที่ถูกปรับปรุงด้วยลำดับเคโอติคแบบ Tent ในการออกแบบสายอากาศแถวลำดับเชิงเส้น ผลลัพธ์แสดงให้เห็นถึงประสิทธิภาพในการลู่เข้าของคำตอบ และการค้นหาคำตอบที่มีความแม่นยำ Zhang และคณะ [24] นำเสนออัลกอริทึมแมลงหวี่แบบไม่ต่อเนื่อง ร่วมกับการค้นหาย่านคำตอบข้างเคียง และ Variable Neighborhood Descent (VND) เพื่อแก้ไขปัญหาการจัดตารางผลิตแบบไหลเลื่อนให้มีต้นทุนการผลิตต่ำสุด วิธีการนี้สามารถป้องกันการติดคำตอบท้องถิ่นและแก้ปัญหาขนาดใหญ่ได้

จากการศึกษางานวิจัยทั้งหมดสามารถสรุปได้ว่า ปัญหาการจัดตารางทางการเกษตร เช่น การจัดตารางการเก็บเกี่ยวพืช สามารถใช้วิธีการทางเมตะฮิวริสติกในการหาผลเฉลยได้ โดยเฉพาะเมื่อปัญหามีขนาดใหญ่หรือมีเงื่อนไขซับซ้อนเพิ่มมากขึ้น ซึ่งจะทำให้ได้คำตอบที่มีคุณภาพและสอดคล้องกับเงื่อนไขต่าง ๆ แม้ว่าวิธีค้นหาแบบแมลงหวี่ เป็นหนึ่งในวิธีการเมตะฮิวริสติกใหม่ ที่มีความสามารถในการค้นหาคำตอบที่เหมาะสม แต่ยังไม่พบงานวิจัยที่นำวิธีการนี้มาประยุกต์ใช้ในการแก้ปัญหาการจัดตารางการเก็บเกี่ยวพืช งานวิจัยครั้งนี้ จึงพัฒนาอัลกอริทึมโดยใช้วิธีค้นหาแบบแมลงหวี่ (Fruit-Fly Optimization Algorithm; FOA) ให้มีความสามารถในการค้นหาคำตอบที่ดีที่สุด (Optimal Solution) เพื่อสร้างแผนการจัดตารางการเก็บเกี่ยวพืชให้โรงงานแปรรูปได้รับปริมาณผลผลิตจากเกษตรกรในพื้นที่ต่าง ๆ ต่อปีสูงสุด และเพียงพอต่อการแปรรูปในแต่ละเดือน ซึ่งจะส่งผลให้บริษัทได้รับผลกำไรตอบแทนต่อปีสูงสุดด้วย ผู้วิจัยได้ศึกษาค่าตัวแปรที่เหมาะสม และวัดประสิทธิภาพของอัลกอริทึมด้วยการเปรียบเทียบคำตอบที่ได้กับโปรแกรมสำเร็จรูป GAMS โดยใช้ตัวอย่างแบบจำลองการเก็บเกี่ยวพืชตั้งแต่ 12 ถึง 60 แปลง เป็นกรณีศึกษา

2. วัสดุ อุปกรณ์และวิธีการวิจัย

2.1 วิธีค้นหาแบบแมลงหวี่ (FOA)

วิธีค้นหาแบบแมลงหวี่ หรือ FOA เป็นอัลกอริทึมการคำนวณเชิงวิวัฒนาการแบบใหม่ที่จำลองกิจกรรมการหาอาหารของแมลงหวี่ เพื่อค้นหาผลลัพธ์ที่เหมาะสมที่สุดของฟังก์ชันวัตถุประสงค์ ในกระบวนการหาอาหารนั้น แมลงหวี่จะบินหาอาหารตามความเข้มข้นของกลิ่นอาหาร และอาศัยการมองเห็นในการบินเข้าใกล้ตำแหน่งของแหล่งอาหาร ซึ่งข้อมูลของกลิ่นอาหารรวมทั้งพิกัดที่เหมาะสมจะถูกส่งผ่านไปยังแมลงหวี่ที่อยู่รอบ ๆ เพื่อค้นหาตำแหน่งของแหล่งอาหารที่ดีที่สุด [18], [25] โดยวิธีค้นหาแบบแมลงหวี่มีขั้นตอนดังต่อไปนี้

1) เริ่มต้นการทำงานของอัลกอริทึม กำหนดให้ $IR_h = 1,\ldots, IR_{max}$ โดยที่ IR_{max} คือ จำนวนรอบการทำงานสูงสุด สำหรับการคำนวณรอบแรก กำหนดตำแหน่งพิกัดเริ่มต้นของประชากรแมลงหวี่ ($X\text{-}axis$ และ $Y\text{-}axis$) ให้เป็นแบบสุ่ม โดยที่ a คือ ค่าพารามิเตอร์ปรับค่าได้ เป็นจำนวนเต็มมีค่ามากกว่า 0

$$X\text{-}axis = a * rand\,(0,1) \tag{1}$$

$$Y\text{-}axis = a * rand\,(0,1) \tag{2}$$

2) แมลงหวี่แต่ละตัวมีทิศทางการบินเป็นแบบสุ่มจากตำแหน่งเดิม เพื่อการค้นหาอาหารตามกลิ่นในพื้นที่ที่แตกต่างกัน กำหนดให้ $i = 1,\ldots., PN_{max}$ โดยที่ PN_{max} คือ จำนวนแมลงหวี่สูงสุด เป็นจำนวนเต็มมีค่ามากกว่า 0

$$X_i = X\text{-}axis + b * rand\,(0,1) - c \tag{3}$$

$$Y_i = Y\text{-}axis + b * rand\,(0,1) - c \tag{4}$$

b และ c คือ ค่าพารามิเตอร์ปรับค่าได้ เป็นจำนวนเต็มมีค่ามากกว่า 0

3) คำนวณระยะทางจากตำแหน่งเริ่มต้น ($Dist_i$) และหาค่าความเข้มข้นของกลิ่น (S_i) ของแมลงหวี่แต่ละตัว

$$Dist_i = \sqrt{x_i^2 + y_i^2} \tag{5}$$

$$S_i = \frac{1}{Dist_i} \tag{6}$$

4) คำนวณค่าฟิตเนสฟังก์ชัน (Fitness Function) จากความเข้มข้นของกลิ่น (S_i) เพื่อเก็บค่าไว้ใน $Smell_i$

$$Smell_i = Function(S_i) \tag{7}$$

5) แมลงหวี่ตัวใดที่ให้ค่า $Smell_i$ มากที่สุดจะถูกเก็บค่าไว้ใน $bestSmell$ และเก็บค่าตำแหน่งไว้ใน $bestIndex$ เพื่อพัฒนาการค้นหาแหล่งอาหารของกลุ่มแมลงหวี่ในรอบถัดไป

$$[bestSmell, bestIndex] = Max(Smell_i) \tag{8}$$

6) เก็บค่า $bestSmell$ ไว้ในตัวแปรใหม่ คือ $smellBest$ รวมทั้งอัพเดตตำแหน่งใหม่เพื่อใช้คำนวณในรอบถัดไป

$$smellBest = bestSmell \tag{9}$$

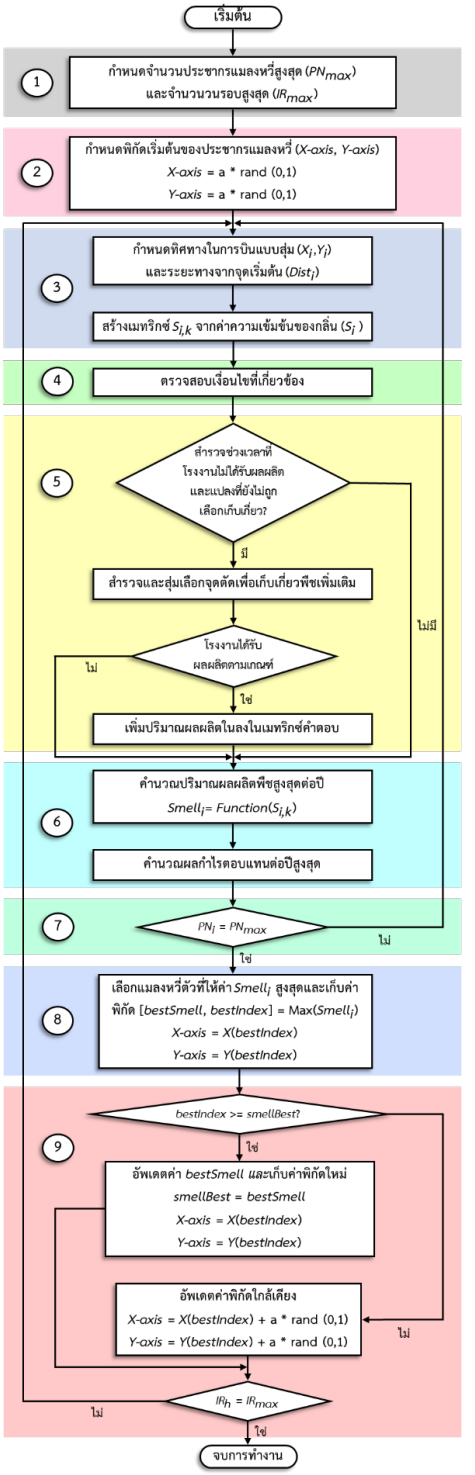
$$X\text{-}axis = X(bestIndex) \tag{10}$$

$$Y\text{-}axis = Y(bestIndex) \tag{11}$$

7) ทำซ้ำในขั้นตอนที่ 2) ถึง 5) หากในรอบการคำนวณใด พบค่า $bestSmell$ เพิ่มขึ้นจากรอบก่อนหน้า ให้กลับไปอัพเดตค่าต่าง ๆ ในขั้นตอนที่ 6 โดยกระบวนการทำซ้ำนี้จะจบลงเมื่อครบจำนวนรอบการทำงานสูงสุด

2.2 การปรับปรุงขั้นตอนวิธีค้นหา แบบแมลงหวี่ (Modified Fruit Fly Algorithm)

อัลกอริทึมวิธีค้นหาแบบแมลงหวี่ ได้รับการปรับปรุง



รูปที่ 1 รูปแบบอัลกอริทึมสำหรับการวางแผนการจัดตารางเก็บเกี่ยวพืชด้วยวิธีค้นหาแบบแมลงหวี่ (FOA)

เพื่อให้สามารถประยุกต์ใช้กับการแก้ไขปัญหาการจัดตารางการเก็บเกี่ยวพืชได้หลายประเภท รูปแบบอัลกอริทึมสำหรับการวางแผนการจัดตารางเก็บเกี่ยวพืช ด้วยวิธีค้นหาแบบแมลงหวี่ แสดงดังรูปที่ 1 ซึ่งมีรายละเอียดดังต่อไปนี้

1) กำหนดพารามิเตอร์ที่จำเป็น ได้แก่ จำนวนประชากรแมลงหวี่ i สูงสุด (PN_{max}) และจำนวนรอบการทำงาน h สูงสุด (IR_{max})

2) สร้างคำตอบเริ่มต้นในรอบที่ 1 โดย 1 คำตอบ ได้รับจากแมลงหวี่ 1 ตัว กระบวนการเริ่มต้นจากการกำหนดตำแหน่งพิกัดเริ่มต้น ($X\text{-}axis$ และ $Y\text{-}axis$) ของแมลงหวี่ทุกตัว ตามสมการที่ (1) และ (2)

3) กำหนดทิศทางในการบินแบบสุ่มของแมลงหวี่ตามสมการที่ (3) และ (4) จากนั้นคำนวณระยะทางจากตำแหน่งเริ่มต้น ($Dist_i$) และคำนวณหาค่าความเข้มข้นของกลิ่น (S_i) ตามสมการที่ (5) และ (6) ตามลำดับ ในการเชื่อมโยงวิธีค้นหาแบบแมลงหวี่กับปัญหาการจัดตารางเก็บเกี่ยวพืชนั้น ค่า S_i จะถูกปรับเปลี่ยนให้อยู่ในรูปแบบของเมทริกซ์ ที่ประกอบด้วยค่าสุ่ม $S_{i,k}$ แตกต่างกัน โดยการคำนวณแสดงดังสมการที่ 12 ซึ่งกำหนดให้ $k = 1,\ldots, p$ โดยที่ p คือ จำนวนแปลงพืชทั้งหมด และ $j = 1,\ldots, r$ โดยที่ r คือ ช่วงเวลาในการเก็บเกี่ยวทั้งหมด (เดือน)

$$S_{i,k} = S_i * rand(0,1) \tag{12}$$

ในแต่ละเดือน j ที่มีการเก็บเกี่ยวพืชจะต้องสร้างเมทริกซ์ที่ประกอบด้วยค่า $S_{i,k}$ ตามจำนวนแปลงพืชที่พิจารณา โดยค่า $S_{i,k}$ ดังกล่าว จะเป็นตัวกำหนดการสับเปลี่ยนหรือจัดลำดับความสำคัญในการเก็บเกี่ยวของแปลงพืชของแมลงหวี่ตัวที่ i ในแปลงที่ k โดยแปลงพืชที่มีค่า $S_{i,k}$ มากจะมีโอกาสในการเก็บเกี่ยวมากกว่าแปลงที่มีค่า $S_{i,k}$ น้อย ตัวอย่างการจัดลำดับพืชจำนวน 12 แปลง ($k = 12$) ของแมลงหวี่ตัวที่ 1 ($i = 1$) ในเดือนที่ 1 ($j = 1$) แสดงดังตารางที่ 1 โดยพืชแปลงที่ 8 ได้รับค่าสุ่ม $S_{i,k}$ มากที่สุด จึงถูกเก็บเกี่ยวเป็นแปลงแรกในปริมาณ 1,000 ตัน ส่วนพืชแปลงที่ 7 จะถูกเก็บเกี่ยวเป็นแปลงสุดท้าย เนื่องจากได้รับสุ่ม $S_{i,k}$ น้อยที่สุด

ตารางที่ 1 ตัวอย่างผลจัดลำดับเมทริกซ์ของพืช 12 แปลง

	$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	$S_{1,4}$	$S_{1,5}$	$S_{1,6}$	$S_{1,7}$	$S_{1,8}$	$S_{1,9}$	$S_{1,10}$	$S_{1,11}$	$S_{1,12}$
ค่าสุ่ม	0.34	0.17	0.22	0.11	0.56	0.38	0.09	0.96	0.41	0.84	0.28	0.77
ปริมาณ	1000	0	500	0	500	1000	0	1000	0	500	0	0
ลำดับ	7	10	9	11	4	6	12	1	5	2	8	3

ตารางที่ 2 ตัวอย่างการสำรวจจุดตัดที่เป็นไปได้

	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$	$k=8$	$k=9$	$k=10$	$k=11$	$k=12$
$j=1$	0	0	0	0	1	0	0	1	0	1	0	0
$j=2$	0	0	0	0	0	0	0	0	0	0	0	0
$j=3$	1	0	1	0	0	1	0	0	1	0	0	1

ตารางที่ 3 การสุ่มเลือกแปลงเก็บเกี่ยวเพิ่มเติม

	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$	$k=8$	$k=9$	$k=10$	$k=11$	$k=12$
$j=1$	0	0	0	0	1	0	0	1	0	1	0	0
$j=2$	0	0	0	1	0	0	0	0	0	0	1	0
$j=3$	1	0	1	0	0	1	0	0	1	0	0	1

4) เมื่อได้ผลการจัดเรียงจากเมทริกซ์ $S_{i,k}$ ข้างต้นครบทุกเดือนแล้ว แมลงหวี่ถูกตรวจสอบเงื่อนไขที่เกี่ยวข้องกับการเก็บเกี่ยวจนกระทั่งผ่านครบทุกเงื่อนไข ผลคำตอบที่ได้จะอยู่ในรูปแบบของเมทริกซ์คำตอบที่ระบุแปลงพืชและปริมาณเก็บเกี่ยวในแต่ละเดือน

5) แม้ว่าเมทริกซ์คำตอบจากขั้นตอนที่ 4) จะผ่านครบทุกเงื่อนไข แต่สามารถตรวจพบได้ว่าโรงงานยังไม่สามารถจัดหาผลผลิตได้ครบทุกเดือน ขั้นตอนการปรับปรุงเมทริกซ์คำตอบจึงเกิดขึ้น โดยผลเมทริกซ์คำตอบที่ได้จะถูกดึงมาสำรวจเดือน j ที่โรงงานยังไม่ได้รับปริมาณผลผลิตในแนวแถว และแปลง k ที่ยังไม่ถูกเลือกในการเก็บเกี่ยวในแนวคอลัมป์ เพื่อกำหนดตำแหน่งจุดตัดของคู่เดือนและแปลงพืช กำหนดให้ $d = 1,\ldots, IN_{max}$ โดยที่ IN_{max} คือ จุดตัดที่เป็นไปได้สูงสุดที่อาจถูกสุ่มค่าเป็น 1 เพื่อรับซื้อผลผลิตเพิ่มเติม ตารางที่ 2 แสดงตัวอย่างที่มีจุดตัดสูงสุด 4 ตำแหน่ง ($IN_{max} = 4$) ที่ถูกค้นพบในเดือนที่ 2($j = 2$) (สีแดง)

จุดตัดสีแดงทั้งหมด 4 ตำแหน่งนี้อาจถูกสุ่มค่าเป็น 1 บางส่วนหรือทั้งหมดก็ได้ โดยที่แต่ละจุดตัดนั้นมีโอกาสที่สุ่มเป็น 0 หรือ 1 เท่า ๆ กัน ตารางที่ 3 แสดงตัวอย่างการสุ่มเลือกแปลงเก็บเกี่ยวเพิ่มเติม 2 จุดตัด (สีน้ำเงิน)

จากตารางที่ 3 ในเดือนที่ 2 ($j = 2$) มีการสุ่มค่าเป็น 1 ในแปลงที่ 4 ($k = 4$) และแปลงที่ 11 ($k = 11$) โดยขั้นตอนการปรับปรุงเมทริกซ์นี้ดัดแปลงมาจากวิธีของ Diaz และ Fernandez [27] ซึ่งเป็นการพัฒนาผลลัพธ์เมทริกซ์คำตอบด้วยวิธีการแทรก (Insertion Move Algorithm) ที่มีการสุ่มค่าและเปลี่ยนตำแหน่งของแปลงพืช k ภายในจุดตัดที่เป็นไปได้ ให้สอดคล้องกับเกณฑ์ปริมาณรับซื้อที่กำหนดไว้ วิธีการนี้เป็นการเพิ่มโอกาสให้โรงงานได้รับซื้อผลผลิตจากแปลงของเกษตรกรครบทุกเดือน รวมทั้งเป็นป้องกันการติดคำตอบท้องถิ่นด้วย เมื่อเมทริกซ์คำตอบได้รับการปรับปรุงแล้ว ปริมาณผลผลิตพืชจากแปลงที่ 4 และ 11 ถูกดึงออกมา เพื่อประเมินว่าผลรวมของปริมาณพืชทั้งสองแปลงนี้ยังอยู่ในเกณฑ์ที่โรงงานกำหนดไว้หรือไม่ หากอยู่ในเกณฑ์ที่กำหนด ปริมาณผลผลิตจากทั้งสองแปลงจะถูกบันทึก

เพิ่มเติมลงในเมทริกซ์ของคำตอบเดิม และถ้าหากไม่อยู่ในเกณฑ์ที่กำหนด ปริมาณผลผลิตจากทั้งสองไม่ถูกบันทึกลงในเมทริกซ์ของคำตอบเดิม

6) จากสมการที่ (7) ค่าฟิตเนสฟังก์ชันเป็นผลจากการใช้ค่าสุ่ม $S_{i,k}$ ชักนำให้เกิดการจัดเรียงเมทริกซ์ในรูปแบบต่าง ๆ ซึ่งจะทำให้เกิดการตัดสินใจในการคัดเลือกแปลง k ในเดือน j ที่แตกต่างกัน กำหนดให้ค่าฟิตเนสฟังก์ชันเก็บไว้ในตัวแปร $Smell_i$ เป็นปริมาณผลผลิตพืชสูงสุดทั้งหมดที่โรงงานสามารถรับซื้อจากเกษตรกรทั้งปี ซึ่งเกิดจากผลรวมของปริมาณพืชที่ต้องเก็บเกี่ยวสูงสุดในแต่ละเดือน และเมื่อได้ผลคำนวณปริมาณผลผลิตสูงสุดต่อปีแล้ว จะสามารถประเมินผลกำไรตอบแทนต่อปีสูงสุดของโรงงานได้

7) ทำซ้ำในขั้นตอนที่ 2) ถึง 6) จนกระทั่งครบจำนวนแมลงหวี่สูงสุด (PN_{max}) ดังนั้น แมลงหวี่ 1 ตัว จะมีผลคำนวณปริมาณผลผลิตต่อปีสูงสุด ($Smell_i$) 1 คำตอบ และผลกำไรตอบแทนต่อปีสูงสุด 1 คำตอบ

8) เมื่อได้คำตอบจากแมลงหวี่ทุกตัวแล้ว เพื่อบรรลุวัตถุประสงค์ในการจัดตารางการเก็บเกี่ยวพืชของโรงงานแปรรูปให้ได้รับปริมาณผลผลิตจากเกษตรกรในพื้นที่ต่าง ๆ ต่อปีสูงสุด และเพียงพอต่อการแปรรูปในแต่ละเดือน ให้ตรวจสอบว่าแมลงหวี่ตัวใดมีค่า $Smell_i$ สูงสุด และเก็บค่านั้นไว้ในตัวแปร *bestSmell* ตามสมการที่ (8) สำหรับการเปรียบเทียบในรอบถัดไป

9) สำหรับรอบการคำนวณแรก ให้เก็บค่า *bestSmell* ไว้ในตัวแปร *smellBest* ตามสมการที่ (9) และเก็บตำแหน่งที่ดีที่สุดไว้ใน *bestIndex* ตามสมการที่ (10) และ (11) เสมอ สำหรับรอบการคำนวณถัดไป ทำซ้ำในขั้นตอนที่ 3) ถึง 8) โดยหากรอบการคำนวณใดพบค่า *bestSmell* เพิ่มขึ้นจากรอบก่อนหน้า ให้เก็บค่านี้ในตัวแปร *smellBest* รวมทั้งเก็บตำแหน่งที่ดีกว่าไว้ใน *bestIndex* และในกรณีที่รอบการคำนวณใดพบค่า *bestSmell* ลดลงจากรอบก่อนหน้า ให้สุ่มค่าตำแหน่งพิกัดใหม่ที่ใกล้เคียงกับตำแหน่งพิกัดเดิมเก็บไว้ใช้ในรอบถัดไป ตามสมการที่ (13) และ (14)

$$X\text{-}axis = X(bestIndex) + a * rand\,(0,1) \qquad (13)$$

$$Y\text{-}axis = Y(bestIndex) + a * rand\,(0,1) \qquad (14)$$

2.3 แบบจำลองและตัวอย่าง

ในแต่ละรอบปีการผลิต โรงงานแปรรูปจะวางแผนการเก็บเกี่ยวและรับซื้อพืชจากเกษตรกร โดยเริ่มจากการประเมินปริมาณผลผลิตของแปลงพืชที่กระจายตัวอยู่ในพื้นที่ต่าง ๆ เพื่อนำข้อมูลมาวิเคราะห์และสร้างแผนกำหนดการรายเดือนออกมาในรูปแบบของตารางให้มีผลกำไรตอบแทนต่อปีสูงสุด ในแต่ละเดือนโรงงานจะมีการคัดเลือกแปลงพืชที่ต้องเก็บเกี่ยว โดยคำนึงถึงปริมาณผลผลิต เงื่อนไขในการเก็บเกี่ยวของเกษตรกร รวมทั้งกำลังการผลิตของโรงงาน เมื่อได้แผนการจัดตารางการเก็บเกี่ยวมาแล้ว เกษตรกรแต่ละรายจะต้องเก็บเกี่ยวผลผลิตตามช่วงเวลาที่โรงงานได้กำหนดไว้ โรงงานจะมีรถบรรทุกออกไปรับผลผลิตยังแปลงพืชของเกษตรกรโดยตรง โดยต้นทุนค่าขนส่งนั้นจะขึ้นอยู่กับระยะทางจากแปลงพืชไปยังโรงงาน เมื่อกระบวนการจัดหาผลผลิตเสร็จสมบูรณ์แล้ว โรงงานจะจ่ายค่ารับซื้อผลผลิตแก่เกษตรกรเป็นค่าตอบแทน

แบบจำลองทางคณิตศาสตร์ต่อไปนี้ประยุกต์มาจากงานวิจัยของ Fikry และคณะ [26] ซึ่งได้รับการตรวจสอบความถูกต้องและความเหมาะสมของแบบจำลองแล้ว ผู้วิจัยจึงนำมาใช้เป็นกรณีศึกษาในการทดสอบอัลกอริทึมในการจัดตารางการเก็บเกี่ยวพืชด้วยวิธีค้นหาแบบแมลงหวี่ ซึ่งมีสมมติฐานดังต่อไปนี้

1) กำหนดให้มีการปลูกพืชหนึ่งชนิด โดยมีการปลูกเพียงหนึ่งครั้งต่อปี

2) พืชจากทุกแปลงมีลักษณะและคุณภาพเหมือนกัน

3) เกษตรกรสามารถเก็บเกี่ยวพืชในปริมาณคงที่ได้ภายในระยะเวลาสี่เดือนหลังโตเต็มที่

4) ปริมาณผลผลิตพืชที่เก็บเกี่ยวมาจะถูกขนส่งไปยังโรงงานแปรรูปโดยรถบรรทุก โดยการขนส่งจะดำเนินการภายในระยะเวลาที่จำกัดเพื่อลดการเสื่อมคุณภาพ

5) ปริมาณผลผลิตพืชที่รับซื้อมาจะถูกแปรสภาพเป็นสินค้าแปรรูปในปริมาณเท่ากัน

6) กำหนดให้มีโรงงานแปรรูป 1 แห่ง

แบบจำลองนี้เป็นแบบจำลองกำหนดการเชิงเส้นจำนวนเต็มไบนารี (BILP) ประกอบด้วย เซต และดัชนี พารามิเตอร์ ตัวแปรตัดสินใจ สมการเป้าหมาย และเงื่อนไข ดังต่อไปนี้

เซตและดัชนี

j คือ เซตของช่วงเวลา (เดือน)

k คือ เซตของแปลงพืช

f คือ เซตของโรงงานแปรรูป

v คือ เซตของยานพาหนะ

พารามิเตอร์

$PC_{min\,jf}$ คือ ปริมาณพืชที่รองรับกำลังการผลิตต่ำสุดในเดือน j ของโรงงาน f (ตัน)

$PC_{max\,jf}$ คือ ปริมาณพืชที่รองรับกำลังการผลิตสูงสุดในเดือน j ของโรงงาน f (ตัน)

$DIST_{kf}$ คือ ระยะทางระหว่างแปลง k ไปยังโรงงาน f (กิโลเมตร)

TC_v คือ ต้นทุนค่าขนส่งพืชที่บรรทุกโดยยานพาหนะ v (บาท/กิโลเมตร·ตัน)

PH_{fj} คือ ต้นทุนการรับซื้อพืชของโรงงาน f ในเดือน j (บาท/ตัน)

TP_{jk} คือ ปริมาณผลผลิตพืชทั้งหมดที่สามารถเก็บเกี่ยวได้ในเดือน j จากแปลง k (ตัน)

SP_{fj} คือ ราคาขายสินค้าของโรงงาน f ซึ่งแปรรูปจากพืชที่ถูกเก็บเกี่ยวในเดือน j (บาทต่อตัน)

ตัวแปรตัดสินใจ

$SQ_{jkfv} \geq 0$ คือ ปริมาณการขนส่งพืชที่ถูกเก็บเกี่ยวในเดือน j จากแปลง k ไปยังโรงงาน f โดยยานพาหนะ v (ตัน)

$HQ_{jk} \geq 0$ คือ ปริมาณการเก็บเกี่ยวพืชในเดือน j จากแปลง k (ตัน)

$$X_{jk}\begin{cases}1, & \text{พืชจะถูกเก็บเกี่ยวในเดือน } j \text{ จากแปลง } k\\ 0, & \text{พืชจะไม่ถูกเก็บเกี่ยวในเดือน } j \text{ จากแปลง } k\end{cases}$$

สมการเป้าหมาย

$$\begin{aligned}Maximize = &\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{f=1}^{F} SP_{fj} HQ_{jk} \\ &-\sum_{j=1}^{J}\sum_{k=1}^{K}\sum_{f=1}^{F}\sum_{v=1}^{V} DIST_{kf} TC_v HQ_{jk} \\ &-\sum_{j=1}^{J}\sum_{k=1}^{K} HQ_{jk} PH_{fj} \end{aligned} \quad (15)$$

เงื่อนไข

$$\sum_{j=1}^{J} X_{jk} = 1 \qquad ;\forall k \in K \quad (16)$$

$$HQ_{jk} = TP_{jk} X_{jk} \quad ;\forall j \in J, \forall k \in K \quad (17)$$

$$HQ_{jk} = SQ_{jkfv} \;;\forall j \in J, \forall k \in K, \forall f \in F, \forall v \in V \quad (18)$$

$$\sum_{k=1}^{K} SQ_{jkfv} \geq PC_{\min jf} \;;\forall j \in J, \forall f \in F, \forall v \in V \quad (19)$$

$$\sum_{k=1}^{K} SQ_{jkfv} \leq PC_{\max jf} \;;\forall j \in J, \forall f \in F, \forall v \in V \quad (20)$$

สมการที่ (15) เป็นสมการเป้าหมายมีวัตถุประสงค์เพื่อให้โรงงานแปรรูปได้รับผลกำไรตอบแทนต่อปีสูงสุด โดยพิจารณาจากรายได้จากการจำหน่ายสินค้าที่แปรรูปจากพืช รวมทั้งพิจารณาต้นทุนค่าขนส่งรวม และต้นทุนในการรับซื้อ ซึ่งเปรียบเสมือนค่าฟิตเนสฟังก์ชันตามสมการที่ (7) โดยสมการเงื่อนไขต่าง ๆ ได้แก่ สมการที่ (16) รับประกันว่าในแต่ละแปลงสามารถเก็บเกี่ยวพืชได้เพียงหนึ่งครั้งเท่านั้น สมการที่ (17) กำหนดให้ปริมาณการเก็บเกี่ยวพืชเท่ากับปริมาณผลผลิตทั้งหมดที่มีอยู่ในแปลงนั้น สมการที่ (18) ปริมาณพืชที่ถูกเก็บเกี่ยวทั้งหมดจะถูกขนส่งจากแปลงพืชไปยังโรงงานแปรรูปโดยตรง สมการที่ (19) และ (20) กำหนดช่วงปริมาณความต้องการในการรับซื้อพืชให้สอดคล้องกับกำลังการผลิตของโรงงาน โดยปริมาณรวมผลผลิตพืชที่ถูกจัดส่งไปยังโรงงานแปรรูปในแต่ละเดือนนั้นต้องมีค่าไม่น้อยกว่าปริมาณที่รองรับกำลังการผลิตต่ำสุด และไม่มากกว่าปริมาณที่รองรับกำลังการผลิตสูงสุด ในการทดสอบความถูกต้องของอัลกอริทึมจะใช้ตัวอย่างข้อมูลขนาด

ตารางที่ 4 ตัวอย่างข้อมูลแปลงพืชทั้ง 18 แปลง

แปลง	ระยะทาง	เดือน											
		1	2	3	4	5	6	7	8	9	10	11	12
1	175	2000	2000	2000	2000								
2	278		1000	1000	1000	1000							
3	143			1500	1500	1500	1500						
4	149				2000	2000	2000	2000					
5	103			1500	1500	1500	1500						
6	179				1000	1000	1000	1000					
7	202					2000	2000	2000	2000				
8	99								1500	1500	1500	1500	
9	151					1000	1000	1000	1000				
10	186									1500	1500	1500	1500
11	170							1000	1000	1000	1000		
12	122									2000	2000	2000	2000
13	248		2000	2000	2000	2000							
14	313				1000	1000	1000	1000					
15	272						500	500	500	500			
16	204					1000	1000	1000	1000				
17	316	1000	1000	1000	1000								
18	521									1500	1500	1500	1500

เล็กของแปลงพืชชนิดหนึ่งจำนวน 18 แปลง ข้อมูลระยะทางจากแปลงพืชต่าง ๆ ไปยังโรงงานแปรรูป 1 แห่ง ($DIST_{kf}$) และข้อมูลปริมาณผลผลิตพืชในแต่ละแปลงที่สามารถเลือกเก็บเกี่ยวได้ในแต่ละเดือน (TP_{jk}) แสดงดังตารางที่ 4 โดยเกษตรกรในแต่ละแปลงจะสามารถทำการเก็บเกี่ยวพืชในปริมาณคงที่ได้ภายในระยะเวลาสี่เดือน หลังจากที่พืชโตเต็มที่แล้ว เช่น เกษตรกรในแปลงที่ 16 อาจเลือกเก็บเกี่ยวพืชหนึ่งครั้งจากเดือนที่ 5 6 7 หรือ 8 ได้ในปริมาณที่เท่ากัน 1,000 ตัน

ผู้วิจัยได้แบ่งขนาดปัญหาออกเป็น 3 กลุ่ม ได้แก่ ปัญหาขนาดเล็ก มีจำนวนแปลงพืช 12 และ 18 แปลง ปัญหาขนาดกลาง มีจำนวนแปลงพืช 24 30 และ 36 แปลง และปัญหาขนาดใหญ่ มีจำนวนแปลงพืช 45 50 และ 60 แปลง ซึ่งค่าพารามิเตอร์ ประกอบด้วย ต้นทุนค่าขนส่ง (TC_v) 1.6 บาทต่อกิโลเมตรต่อตัน ต้นทุนในการรับซื้อ (PH_j) 5,500 บาทต่อตัน ราคาขายของสินค้าแปรรูป (SP_{jk}) 8,000 บาทต่อตัน กำลังการผลิตต่ำสุด ($PC_{min\ jf}$) 1,000 ตัน และกำลังการผลิตสูงสุด ($PC_{max\ jf}$) 2,000 ตัน สำหรับการทดสอบอัลกอริทึมได้ศึกษาภายใต้กรอบเวลา ทั้งหมด 12 เดือน ($j = 12$) กับพื้นที่เกษตรกรตั้งแต่ 12 ถึง 60 แปลง ($k = 12,\ldots, 60$) และโรงงานแปรรูป 1 แห่ง ($f = 1$) โดยโรงงานจะมีรถบรรทุกขนส่งผลผลิตพืช 1 คัน ($v = 1$)

3. ผลการทดลอง

ในการทดสอบอัลกอริทึมด้วยวิธีค้นหาแบบแมลงหวี่ (FOA) กับตัวอย่างการจัดตารางการเก็บเกี่ยวพืช ผู้วิจัยได้เขียนโปรแกรมภาษาไพทอน (Python) บน Jupyter Notebook โดยใช้คอมพิวเตอร์แบบพกพา AMD Ryzen 7 3700U CPU @ 2.30 กิกะเฮิรตซ์ หน่วยความจำแรม 8 กิกะไบต์ บนระบบปฏิบัติการวินโดว์ 11 แบบ 64 บิต

ตารางที่ 5 แผนการจัดตารางการเก็บเกี่ยวพืช 18 แปลง โดยใช้โปรแกรม GAMS

แปลง	เดือน											
	1	2	3	4	5	6	7	8	9	10	11	12
1	2000											
2				1000								
3												
4							2000					
5			1500									
6				1000								
7					2000							
8										1500		
9						1000						
10									1500			
11								1000				
12												2000
13		2000										
14						1000						
15									500			
16								1000				
17												

ผลคำตอบที่ได้จากอัลกอริทึม ด้วยวิธีค้นหาแบบแมลงหวี่นั้น จะถูกเปรียบเทียบกับผลคำตอบที่แก้ปัญหาผ่านโปรแกรมสำเร็จรูป GAMS version 40.1.0 Solver CPLEX และเมื่อทำการทดสอบแบบจำลองโดยใช้โปรแกรมสำเร็จรูป GAMS กับตัวอย่างข้อมูลแปลงพืช 18 แปลง ผลการจัดตารางที่ดีที่สุด แสดงดังตารางที่ 5

จากตารางที่ 5 โปรแกรมสำเร็จรูป GAMS สามารถค้นหาคำตอบที่ดีที่สุด (Optimal Solution) ได้ โดยโรงงานแปรรูปจะได้รับผลผลิตพืชในทุก ๆ เดือน จากพื้นที่เกษตรกรทั้งหมด 16 จาก 18 แปลง ตามเกณฑ์ที่โรงงานกำหนดไว้ คือ ระหว่าง 1,000 ถึง 2,000 ตัน และมีผลผลิตพืชสูงสุดโดยรวมต่อปี 22,500 ตัน โดยมีต้นทุนการรับซื้อ 123,750,000 บาท ต้นทุนค่าขนส่งรวม 7,338,400 บาท และผลกำไรตอบแทนต่อปีสูงสุด 48,911,600 บาท

ในการทดสอบอัลกอริทึมด้วยวิธีค้นหาแบบแมลงหวี่กับตัวอย่างข้อมูลแปลงพืชจำนวน 18 แปลง มีการปรับค่าพารามิเตอร์ คือ ค่า a และ c เท่ากับ 1 และค่า b เท่ากับ 2 โดยที่จำนวนประชากรแมลงหวี่สูงสุด (PN_{max}) เท่ากับ 5 และจำนวนรอบการทำงานสูงสุด (IR_{max}) เท่ากับ 5,000 ผลการทดสอบอัลกอริทึมพบว่า ปริมาณผลผลิตพืชสูงสุดที่โรงงานสามารถรับซื้อต่อปีซึ่งเก็บค่าไว้ในตัวแปร *smellBest* พบการลู่เข้าของคำตอบที่เหมาะสม ซึ่งเปรียบเทียบได้กับพฤติกรรมของแมลงหวี่ที่บินค้นหาแหล่งอาหารจนกระทั่งค้นพบแหล่งอาหารที่ดีที่สุด ตัวแปรดังกล่าวนี้จะมีค่าเพิ่มขึ้นเมื่อจำนวนรอบการทำงานเพิ่มขึ้น อันเป็นผลมาจากการปรับปรุงประสิทธิภาพของเมทริกซ์คำตอบซึ่งทำหน้าที่ป้องกันการติดคำตอบท้องถิ่น และทำให้โรงงานได้รับซื้อผลผลิตครบทุกเดือน และส่งผลให้ได้รับผลกำไรตอบแทนต่อปีสูงสุด ผลการจัดตารางที่ดีที่สุด แสดงดังตารางที่ 6

ตารางที่ 6 สามารถสรุปได้ว่า ในทุก ๆ เดือนโรงงานแปรรูปจะได้รับผลผลิตพืชจากพื้นที่เกษตรกรทั้งหมด 16 จาก 18 แปลง ตามเกณฑ์ปริมาณที่ได้กำหนดไว้ โดยโรงงาน

ตารางที่ 6 แผนการจัดตารางการเก็บเกี่ยวพืช 18 แปลง ด้วยวิธีค้นหาแบบแมลงหวี่ (FOA) รูปแบบที่ 1

แปลง	เดือน											
	1	2	3	4	5	6	7	8	9	10	11	12
1	2000											
2				1000								
3												
4							2000					
5			1500									
6				1000								
7					2000							
8											1500	
9						1000						
10									1500			
11								1000				
12												2000
13		2000										
14						1000						
15									500			
16								1000				
17												
18										1500		

จะได้รับผลผลิตสูงสุดโดยรวมต่อปี 22,500 ตัน โดยมีต้นทุนค่าขนส่งรวม 7,338,400 บาท และต้นทุนการรับซื้อ 123,750,000 บาท ส่งผลให้ผลกำไรตอบแทนต่อปีสูงสุดอยู่ที่ 48,911,600 บาท นอกจากนี้ ยังพบผลการจัดตารางในรูปแบบอื่น ๆ ที่คำนวณให้ค่าปริมาณผลผลิตและผลกำไรรวมสูงสุดต่อปีเท่ากัน ดังแสดงในตารางที่ 7

ในการทดสอบตัวอย่างปัญหาการจัดตารางการเก็บเกี่ยวพืชในพื้นที่แปลงที่ขนาดต่าง ๆ จะใช้ค่าพารามิเตอร์เดียวกันกับตัวอย่างข้างต้น กำหนดให้จำนวนรอบการทำงานสูงสุด (IR_{max}) ในปัญหาขนาดเล็ก เท่ากับ 5,000 รอบ ปัญหาขนาดกลาง 10,000 รอบ และปัญหาขนาดใหญ่ 20,000 รอบ ผลการทดสอบปัญหาการจัดตารางการเก็บเกี่ยวพืชขนาดต่าง ๆ โดยใช้วิธีค้นหาแบบแมลงหวี่และวิธีโปรแกรมสำเร็จรูป GAMS แสดงดังตารางที่ 8

เมื่อเปรียบเทียบผลลัพธ์จากการทดสอบตัวอย่างปัญหาการจัดตารางการเก็บเกี่ยวพืชในขนาดต่าง ๆ ด้วยวิธีค้นหาแบบแมลงหวี่และโปรแกรมสำเร็จรูป GAMS พบว่าวิธีค้นหาแบบแมลงหวี่เป็นไปตามเงื่อนไขต่าง ๆ ที่ได้กำหนดไว้ ซึ่งโรงงานจะได้รับซื้อผลผลิตพืชจากพื้นที่แปลงเกษตรกรต่าง ๆ ทุก ๆ เดือน ในปริมาณที่สอดคล้องกับกำลังการผลิตของโรงงาน โดยผลคำตอบปริมาณผลผลิตสูงสุดต่อปีมีค่าเทียบเท่ากับผลคำตอบที่ดีที่สุด เมื่อใช้โปรแกรมสำเร็จรูป GAMS แก้ไขปัญหา สำหรับผลการจัดตารางการเก็บเกี่ยวของทั้งสองวิธีนั้น พบจำนวนแปลงเก็บเกี่ยวที่เหมือนกันในจำนวนที่เท่ากัน ดังในตัวอย่างข้างต้น ที่มีการคัดเลือกแปลงพืชทั้งหมด 16 จาก 18 แปลง ซึ่งได้มีการปฏิเสธการเก็บเกี่ยวและรับซื้อจากแปลงที่ 3 และ 17 เช่นเดียวกัน จึงทำให้ต้นทุนค่าขนส่งรวมจากแปลงพืชต่าง ๆ ไปยังโรงงานและต้นทุนการรับซื้อของทั้งสองวิธีนั้นเท่ากัน และส่งผลให้ผลกำไรตอบแทนต่อปีสูงสุดเท่ากันด้วย

ตารางที่ 7 การจัดตารางการเก็บเกี่ยวพืช 18 แปลง ด้วยวิธีค้นหาแบบแมลงหวี่ (FOA) รูปแบบที่ 2

แปลง	เดือน											
	1	2	3	4	5	6	7	8	9	10	11	12
1	2000											
2				1000								
3												
4					2000							
5			1500									
6				1000								
7							2000					
8										1500		
9								1000				
10									1500			
11								1000				
12												2000
13		2000										
14						1000						
15									500			
16						1000						
17												
18											1500	

ตารางที่ 8 ผลการทดสอบด้วยวิธีค้นหาแบบแมลงหวี่เปรียบเทียบกับโปรแกรม GAMS

กลุ่มปัญหา	จำนวนแปลง	วิธีการทดสอบ								% ความต่างของกำไร
		GAMS			FOA					
		จำนวนแปลงเก็บเกี่ยว	ปริมาณผลผลิตสูงสุดต่อปี (ตัน)	ผลกำไรตอบแทนต่อปีสูงสุด (บาท)	จำนวนแผนเก็บเกี่ยว	จำนวนแปลงเก็บเกี่ยว	ปริมาณผลผลิตสูงสุดต่อปี (ตัน)	ผลกำไรตอบแทนต่อปีสูงสุด (บาท)	เวลาคำนวณ (นาที)	
ขนาดเล็ก	12	12	18,000	40,407,200	4	12	18,000	40,407,200	00:01:12	0.00
	18	16	22,500	48,911,600	4	16	22,500	48,911,600	00:02:33	0.00
ขนาดกลาง	24	18	23,500	52,893,200	2	18	23,500	52,893,200	00:10:07	0.00
	30	18	24,000	54,816,800	4	18	24,000	54,816,800	00:11:37	0.00
	36	18	24,000	55,118,400	2	18	24,000	55,118,400	00:12:39	0.00
ขนาดใหญ่	45	20	24,000	55,242,400	2	20	24,000	55,242,400	00:14:53	0.00
	50	19	24,000	55,436,800	5	19	24,000	55,436,800	00:17:07	0.00
	60	20	24,000	55,503,200	4	20	24,000	55,503,200	00:22:20	0.00

อย่างไรก็ตามวิธีการทั้งสองนี้สามารถสร้างแผนการจัดตารางการเก็บเกี่ยวที่แตกต่างกัน เนื่องจากผลการตัดสินใจในแต่ละเดือนมีการกำหนดแปลงพืชที่ต้องเก็บเกี่ยวแตกต่างกัน เช่น ในเดือนที่ 8 หากใช้โปรแกรมสำเร็จรูปจะต้องเก็บเกี่ยวพืชจากแปลงที่ 11 และ 16 และหากใช้วิธีค้นหาแบบแมลงหวี่จะต้องเก็บเกี่ยวพืชจากแปลงที่ 9 และ 11 โดยปกติแล้วโปรแกรมสำเร็จรูป GAMS จะสามารถสร้างแผนการจัดตารางที่ดีที่สุดได้เพียงหนึ่งรูปแบบเท่านั้น ในขณะที่วิธีค้นหาแบบแมลงหวี่ สามารถสร้างแผนการจัดตารางการเก็บเกี่ยวได้หลากหลายรูปแบบ ซึ่งอาจมีหลายแผนคำตอบที่สามารถคำนวณปริมาณผลผลิตสูงสุดและผลกำไรรวมสูงสุดต่อปีเทียบเท่ากับการใช้โปรแกรมสำเร็จรูป เนื่องด้วยวิธีการอัลกอริทึมที่นำเสนอนี้มีความยืดหยุ่นในการค้นหาคำตอบที่วงกว้างขึ้น ซึ่งจะเป็นประโยชน์ต่อโรงงานแปรรูปเมื่อมีการวางแผนและสร้างทางเลือกในการทำข้อตกลงกับเกษตรกรที่จะต้องเก็บเกี่ยวผลผลิตมาให้โรงงานในช่วงเวลาที่เหมาะสม

4. อภิปรายผลและสรุป

งานวิจัยนี้ได้พัฒนาอัลกอริทึมในการแก้ปัญหาการจัดตารางการเก็บเกี่ยวพืชด้วยวิธีค้นหาแบบแมลงหวี่ (FOA) ให้สามารถค้นหาคำตอบที่ดีที่สุด (Optimal Solution) ได้ โดยมีวัตถุประสงค์ คือ การสร้างผลกำไรตอบแทนต่อปีสูงสุดแก่โรงงานแปรรูป การทดสอบอัลกอริทึมขนาดต่าง ๆ ได้ใช้ตัวอย่างข้อมูลพืชชนิดหนึ่งจากพื้นที่แปลงขนาด 12 ถึง 60 แปลง เพื่อคำนวณและสร้างตารางการเก็บเกี่ยวพืชให้มีปริมาณผลผลิตสูงสุดต่อปี รวมทั้งมีผลผลิตเพียงพอต่อการแปรรูปในแต่ละเดือน ขั้นตอนการปรับปรุงเมทริกซ์คำตอบถูกนำมาใช้เพื่อป้องกันการติดคำตอบท้องถิ่นและพัฒนาคำตอบให้ดีขึ้น อัลกอริทึมที่เสนอวิธีการค้นหาแบบแมลงหวี่นี้แสดงให้เห็นถึงประสิทธิภาพในการค้นหาคำตอบที่ดีที่สุดมากกว่าหนึ่งแผนการเก็บเกี่ยว โดยมีความสามารถให้ผลกำไรตอบแทนสูงสุดต่อปีเทียบเท่ากับการใช้โปรแกรมสำเร็จรูป GAMS ในทุก ๆ ขนาดปัญหา ดังนั้นอัลกอริทึมที่ประยุกต์วิธีการค้นหาแบบแมลงหวี่จึงเป็นหนึ่งในวิธีที่สามารถช่วยให้โรงงานแปรรูปได้ผลการจัดตารางเก็บเกี่ยวพืชอย่างเหมาะสม

ในการต่อยอดงานวิจัยในอนาคต สามารถนำอัลกอริทึมดังกล่าวประยุกต์ใช้ในการแก้ปัญหาแบบจำลองทางคณิตศาสตร์ที่มีความซับซ้อนมากยิ่งขึ้นในระดับ NP hard โดยอาจเพิ่มเงื่อนไขที่เกี่ยวข้องกับกิจกรรมอื่น ๆ ในห่วงโซ่อุปทาน เช่น การจัดเก็บ การขนส่ง และการกระจายสินค้า รวมทั้งเปรียบเทียบผลลัพธ์กับอัลกอริทึมอื่น ๆ เพื่อให้เห็นประสิทธิภาพของวิธีการที่นำเสนอมากขึ้น

เอกสารอ้างอิง

[1] Songkhla Rajabhat University (SKRU). [Online] (in Thai). Available: http://oservice.skru.ac.th/ebookft/ 374/chapter8.pdf.

[2] C. S. Chen, “Scheduling of planting and harvesting programmes for processing vegetables,” *Journal of Agricultural Engineering Research*, vol. 19, no. 1, pp. 51– 57, 1974.

[3] M. R. Garey, D. S. Johnson, and R. Sethi, “The Complexity of Flowshop and Jobshop Scheduling,” *Mathematics of Operations Research*, vol. 1, no. 2, pp. 117–129, 1976.

[4] S. Bandaru and K. Deb, “Metaheuristic techniques,” in *Decision Sciences*, Boca Raton, FL: CRC Press, 2016, pp. 693–750.

[5] E. W. Richards and E. A. Gunn, “A model and tabu search method to optimize stand harvest and road construction schedules,” *Forest Science*, vol. 46, no. 2, pp. 188–203, 2000.

[6] M. Pant, R. Thangaraj, D. Rani, A. Abraham, and D. Srivastava, “Estimation of optimal crop plan using nature inspired meta- heuristics,” *World Journal of Modelling and Simulation*, vol. 6, pp. 97–109, 2010.

[7] B. J. Stray, J. H. van Vuuren, and C. N. Bezuidenhout,

"An optimisation-based seasonal sugarcane harvest scheduling decision support system for commercial growers in South Africa," *Computers and Electronics in Agriculture*, vol. 83, pp. 21–31, 2012.

[8] A. Filho, H. Florentino, and M. Pato, "A genetic algorithm for crop rotation," in *Proceedings ICORES*, 2012, pp. 454–457.

[9] W. Neungmatcha and K. Sethanan, "Optimal mechanical harvester route planning for sugarcane field operations using particle swarm optimization," *Engineering and Applied Science Research*, vol. 42, no. 2, pp. 125–133, 2015.

[10] A. Rath, S. Biswal, S. Samantaray, and P. Swain, "Derivation of optimal cropping pattern in part of hirakud command using cuckoo search," in *Proceedings ICMAEM*, 2017.

[11] E. Afifah, A. Alamsyah, and E. Sugiharti, "Scheduling optimization of sugarcane harvest using simulated annealing algorithm," *Scientific Journal of Informatics*, vol. 5, pp. 138–147, 2018.

[12] U. Ketsripongsa, R. Pitakaso, K. Sethanan, and T. Srivarapongse, "An improvement differential evolution algorithms for the crop planning in the northeastern region of Thailand," *Mathematical and Computational Applications*, vol. 23, no. 3, 2018.

[13] R. Kommadath, D. Maharana, R. Anandalakshmi, and P. Kotecha, "Multi- objective scheduling in the vegetable processing and packaging facility using metaheuristic-based framework," *Food and Bioproducts Processing*, vol. 137, pp. 1–19, 2023.

[14] S. Bhupender and K. Ashwani, "Introduction to Hybrid algorithm," *International Journal of Advance Research and Innovative Ideas in Education*, vol. 8, no. 3, 2022.

[15] D. Mogale, D. A. Ghadge, S. Kumar, and M. Tiwari, "Modelling supply chain network for procurement of food grains in India," *International Journal of Production Research*, vol. 58, 2019.

[16] V. Chouhan, B. S. H. Khan, and M. Hajiaghaei-Keshteli, "Sustainable planning and decision-making model for sugarcane mills considering environmental issues," *Journal of Environmental Management*, vol. 303, 2022.

[17] H. d. O. Florentino, D. F. Jones, C. A. Irawan, D. Ouelhadj, B. Khosravi, and D. R. Cantane, "An optimization model for combined selecting, planting and harvesting sugarcane varieties," *Annals of Operations Research*, vol. 314, no. 2, pp. 451–469, 2022.

[18] Y. Li and M. Han, "Improved fruit fly algorithm on structural optimization," *Brain informatics*, vol. 7, no. 1, 2020.

[19] A. Tapao and R. Cheerarot, "Optimum design of reinforced concrete foundations by fruit fly optimization algorithm," *The Journal of KMUTNB*, vol. 31, no. 1, pp. 16–24, 2021 (in Thai).

[20] İ. Babaoğlu, "Solving 2D strip packing problem using fruit fly optimization algorithm," *Procedia Computer Science*, vol. 111, pp. 52–57, 2017.

[21] H. Iscan and M. Gunduz, "An application of fruit fly optimization algorithm for traveling salesman problem," *Procedia Computer Science*, vol. 111, pp. 58–63, 2017.

[22] L. Wang, X.-l. Zheng, and S.-y. Wang, "A novel binary fruit fly optimization algorithm for solving the multidimensional knapsack problem," *Knowledge-Based Systems*, vol. 48, pp. 17–23, 2013.

[23] N. Mhuadthongon, "Linear Array Pattern Synthesis Using Chaotic Sequence Enhanced FOA Method," *Ladkrabang Engineering Journal*, vol. 35, no. 3–4, pp. 47–54, 2018 (in Thai).

[24] X. Zhang, X. Liu, S. Tang, G. Królczyk, and Z. Li, "Solving scheduling problem in a distributed manufacturing system using a discrete fruit fly optimization algorithm," *Energies*, vol. 12, no. 17, 2019.

[25] F. Xu and Y. Tao, "The Improvement of Fruit Fly Optimization Algorithm-Using Bivariable Function as Example," *Advanced Materials Research*, vol. 756–759, pp. 2952–2957, 2013.

[26] I. Fikry, M. Gheith, and A. Eltawil, "An integrated production-logistics-crop rotation planning model for sugar beet supply chains," *Computers & Industrial Engineering*, vol. 157, pp. 1–13, 2021.

[27] J. A. Diaz and E. Fernandez, "A tabu search heuristic for the generalized assignment problem," *European Journal of Operational Research*, vol. 132, no. 1, pp. 22–38, 2001