

ตัวแบบสินค้าคงคลังสองระดับเมื่อมีความต้องการสินค้าไม่แน่นอน

ศิริประภา มโนมัยย์¹

คณะวิทยาศาสตร์ประยุกต์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

1518 ถนนประชากรราษฎร์ 1 แขวงวงศ์สว่าง เขตบางซื่อ กรุงเทพฯ 10800

Received: 15 September 2021; Revised: 27 November 2021; Accepted: 28 December 2021

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาตัวแบบสินค้าคงคลังสองระดับกรณีที่มีความต้องการสินค้าไม่แน่นอนและเปรียบเทียบวิธีการหาค่าตอบที่เหมาะสมที่สุดของตัวแบบโดยใช้วิธีการค้นหาคำตอบแบบขั้นตอนวิธีแมลงปอ (Dragonfly Algorithms; DA) วิธีการหาค่าที่เหมาะสมที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization ; PSO) และขั้นตอนวิธีค้างคาว (Bat Algorithm ; BA) ศึกษาภายใต้สถานการณ์สินค้าสองรายการมีความต้องการสินค้าที่เท่ากัน และไม่เท่ากัน โดยมีพารามิเตอร์ที่ศึกษาได้แก่ รอบของการค้นหาคำตอบ จำนวนสินค้ารายการที่บรรจุในแต่ละแพ็คเกจ ราคาสินค้าต่อหน่วย ค่าใช้จ่ายต่อหน่วยในการจัดส่งสินค้า ค่าใช้จ่ายต่อหน่วยเนื่องจากสินค้าไม่พอจำหน่าย ค่าใช้จ่ายในการจัดซื้อ ค่าใช้จ่ายในการผลิต ค่าใช้จ่ายในการถือครองสินค้าต่อหน่วยต่อหน่วยเวลาของผู้ขาย ค่าใช้จ่ายในการถือครองสินค้าต่อหน่วยต่อหน่วยเวลาของผู้ซื้อ ประสิทธิภาพของวิธีการค้นหาคำตอบพิจารณาจากการเข้าสู่หาค่าตอบที่เหมาะสมที่สุดของแต่ละวิธี ผลการวิจัยพบว่า วิธี DA ให้คำตอบที่ดีกว่าวิธีอื่นและเข้าใกล้คำตอบที่ดีที่สุดเมื่อจำนวนรอบการค้นหาคำตอบมากกว่าหรือเท่ากับ 50 ในขณะที่วิธี PSO เข้าใกล้คำตอบที่ดีที่สุดและเริ่มเสถียรที่จำนวนรอบของการค้นหาคำตอบมากกว่า 500 สำหรับวิธี BA ให้คำตอบที่ด้อยกว่าวิธีอื่นทุกกรณี

คำสำคัญ: ตัวแบบสินค้าคงคลังสองระดับ, ความต้องการสินค้าไม่แน่นอน, เมตาฮิวริสติก

* Corresponding author. E-mail: sirapapa.m@sci.kmutnb.ac.th

¹ ผู้ช่วยศาสตราจารย์ ภาควิชาสถิติประยุกต์ คณะวิทยาศาสตร์ประยุกต์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

Two-echelon Inventory Model with Uncertainty Demand

Siraprapa Manomat¹

Faculty of Applied Science, King Mongkut's University of Technology North Bangkok (KMUTNB),
1518 Pracharat 1 Rd., Wongsawang, Bangsue, Bangkok 10800, Thailand

Received: 15 September 2021; Revised: 27 November 2021; Accepted: 28 December 2021

Abstract

This research aims to develop a two-echelon inventory model under uncertain demand and to compare three meta heuristic method such as a Dragonfly Algorithm (DA), Particle Swarm Optimization (PSO) and Bat Algorithm (BA). Two products with equal and unequal demand are studied in various parameters such as iteration, the number of products in each package, purchasing price per unit, backorder cost per unit, shortage cost for each unit that is lost sale, constant cost per order, buyer's constant product cost for each setup, vendor's holding cost per unit time per unit and buyer's holding cost per unit time per unit parameters. Minimum total inventory costs are investigated. The results showed that the DA method approach to the optimal solution when the iterations are greater than or equal to 50, while the PSO method approach to the optimal solution when the iterations are greater than 500. The BA method is inferior answers in all cases.

Keywords: Two-echelon inventory model, Uncertainty demand, Meta heuristic

* Corresponding author. E-mail: siraprapa.m@sci.kmutnb.ac.th

¹ Assistant Professor in Faculty of applied science, King Mongkut's University of Technology North Bangkok

1. บทนำ

การจัดการสินค้าคงคลังเป็นกิจกรรมหนึ่งในกระบวนการจัดการโลจิสติกส์และโซ่อุปทาน (Logistic and supply chain) ซึ่งการจัดการสินค้าคงคลังที่มีประสิทธิภาพสามารถช่วยลดต้นทุนรวมของระบบได้ โดยปัจจัยที่ส่งผลต่อการบริหารจัดการสินค้าคงคลังได้แก่ ลักษณะความต้องการสินค้า ระบบการเติมเต็มสินค้า และนโยบายสินค้าคงคลัง มีผู้พัฒนาตัวแบบที่ใช้ในการแก้ปัญหาในระบบสินค้าคงคลังที่แตกต่างกันไป ซึ่งส่วนใหญ่จะศึกษาภายใต้ความต้องการสินค้าคงคลังที่ (Deterministic demand) หรือเป็นระบบสินค้าคงคลังภายใต้คลังเดียว (Single-echelon inventory system) อย่างไรก็ตามระบบการกระจายสินค้าในปัจจุบันมิใช่เป็นลักษณะคลังสินค้าระดับเดียว อาจมีศูนย์กระจายสินค้าก่อนจัดส่งถึงร้านค้าปลีก เพื่อความรวดเร็วและประหยัดค่าใช้จ่ายในการขนส่ง รูปแบบดังกล่าวเรียกว่าระบบสินค้าคงคลัง 2 ระดับ (Two-echelon inventory systems) โดยระดับที่ 1 คือคลังสินค้าของร้านค้าปลีก และระดับที่ 2 คือคลังสินค้าของศูนย์กระจายสินค้า ทำหน้าที่เติมสินค้าให้กับร้านค้าปลีก ซึ่งมีผู้พัฒนาและออกแบบนโยบายสำหรับแก้ปัญหาสินค้าคงคลังสองระดับอย่างต่อเนื่อง เช่น วราภรณ์ พกนนท์ และคณะ [1] ได้ศึกษานโยบายการเติมเต็มร่วม สำหรับระบบสินค้าแบบสองระดับ ภายใต้การศึกษาระยะเวลานำ, จารุมาศ แสงสว่าง และวุฒิชัย ศรีโสดาลพ [2] ได้พัฒนาตัวแบบสินค้าคงคลังแบบสองระดับที่มีเวลานำและมีการคืนสินค้าที่ขึ้นอยู่กับความต้องการสินค้า, Yang และ Zeng [3] ได้พัฒนาตัวแบบและวิธีการหาตำแหน่งการตั้งคลังสินค้าแบบสองระดับ, Sadjady และ Davoudpour [4] ได้ออกแบบระบบโซ่อุปทานที่มีคลังสินค้าแบบสองระดับสำหรับสินค้าหลายรายการ Mahmoodi และคณะ [5] ได้พัฒนาตัวแบบสินค้าคงคลังสองระดับที่มีการเติมเต็มสินค้าและมีนโยบายยอมให้เกิดการขาดแคลนสินค้า

จากงานวิจัยที่กล่าวมาข้างต้นนั้น จะเห็นได้ว่าในการใช้ตัวแบบแทนระบบของปัญหา ตัวแบบจะแตกต่างกันไปตามข้อตกลงเบื้องต้น และเงื่อนไขต่าง ๆ ที่แตกต่างกัน ทำให้เกิดตัวแบบสินค้าคงคลังมากมาย การวิเคราะห์ปัญหาสินค้าคงคลังนั้นจึงต้องเลือกตัวแบบให้เหมาะสมกับสภาพปัญหาในสถานการณ์จริง ตามปัจจัยที่กล่าวไปแล้ว นอกจากนี้

การเลือกตัวแบบที่เหมาะสมแล้ว การเลือกวิธีการหาคำตอบที่เหมาะสมก็เป็นประเด็นสำคัญที่ต้องทำการศึกษาต่อไป เนื่องจากตัวแบบสินค้าคงคลังตั้งแต่สองระดับขึ้นไปมีความซับซ้อนและมีตัวแปรที่ต้องหาคำตอบจำนวนมาก โดยทั่วไปจำนวนคำตอบที่เป็นไปได้จะมากจนไม่สามารถใช้วิธีการทางคณิตศาสตร์ธรรมดาแก้ปัญหาเพื่อหาคำตอบที่เหมาะสมได้ วิธีการหาคำตอบที่นิยมใช้ในกรณีนี้ได้แก่วิธีฮิวริสติก (Heuristic) และวิธีเมตาฮิวริสติก (Meta heuristic) ซึ่งมีการพัฒนาอย่างต่อเนื่อง วิธีฮิวริสติกเป็นวิธีการประมาณการคำตอบอย่างมีหลักการ ส่วนวิธีเมตาฮิวริสติกเป็นวิธีการประมาณการคำตอบที่พัฒนาภายหลังและใช้แนวคิดจากปรากฏการณ์ทางธรรมชาติ โดยเฉพาะพฤติกรรมของสัตว์สังคม เช่น มด ผึ้ง ค้างคาว หรือแมลง เป็นต้น วิธีการประมาณคำตอบทั้งสองแบบนี้มีได้รับประกันว่าคำตอบที่ได้จะเป็นคำตอบที่ดีที่สุด แต่เป็นวิธีที่ลดระยะเวลาในการคำนวณคำตอบได้อย่างมาก

การนำวิธีเมตาฮิวริสติกมาแก้ปัญหาตัวแบบสินค้าคงคลังได้มีการพัฒนาอย่างต่อเนื่อง เช่น Samani และ Hosseimi-Motlagh [6] ได้ใช้ขั้นตอนวิธีแบบผสมผสาน (Hybrid Algorithm) แก้ปัญหาการวางตำแหน่งคลังสินค้าแบบสองระดับกรณีที่มีความต้องการสินค้าเป็นแบบฟัซซี (Fuzzy demand), Seifbarghy และคณะ [7] ได้นำเสนอตัวแบบสินค้าสำหรับปัญหาที่มีผู้ผลิตรายเดียวแต่มีผู้ซื้อหลายราย ซึ่งผู้ผลิตต้องผลิตและส่งสินค้าไปยังผู้ซื้อ การคำนวณยึดหลักการของการสั่งซื้อที่ประหยัด และใช้วิธีการหาคำตอบตามวิธีการหาค่าเหมาะที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization; PSO) ศิริรัตน์ กะการดี และ ศิริประภา มโนมัยย์ [8] ได้ศึกษาตัวแบบสินค้าคงคลังสามระดับสำหรับสินค้ารายการเดียวโดยเปรียบเทียบวิธีการหาคำตอบด้วยวิธี PSO และขั้นตอนวิธีค้างคาว (Bat Algorithm; BA) ซึ่งผลการทดลองพบว่าวิธี PSO มีประสิทธิภาพในการหาคำตอบดีกว่าวิธี BA แต่ใช้เวลาประมวลผลสูงกว่าร้อยละ 49.64

ในปี 2016 Mirjalili [9] พัฒนาวิธีการค้นหาคำตอบแบบขั้นตอนวิธีแมงปอ (Dragonfly Algorithms; DA) โดยการเลียนแบบการจับกลุ่มของแมงปอในธรรมชาติ, ต่อมาในปี 2020 Rahman และ Rashid [10] ได้ศึกษาและนำวิธี DA ไปใช้ในการหาคำตอบในงานวิศวกรรมเครื่องกลและ

วิศวกรรมไฟฟ้า โดยเปรียบเทียบกับวิธีเมตาฮิวริสติกอื่น ๆ ซึ่งพบว่าวิธี DA มีประสิทธิภาพในการหาคำตอบได้ดีและรู้เข้าหาคำตอบได้เร็ว ด้วยเหตุนี้จึงทำให้ผู้วิจัยศึกษาการนำวิธี DA มาใช้ในการหาคำตอบของตัวแบบสินค้าคงคลังสองระดับ โดยงานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาตัวแบบสินค้าคงคลังสองระดับกรณีที่มีความต้องการสินค้าไม่แน่นอนและเปรียบเทียบวิธีการหาคำตอบที่เหมาะสมที่สุดของตัวแบบโดยใช้วิธีการค้นหาคำตอบแบบ DA, วิธี PSO และวิธี BA

2. วิธีการดำเนินงาน

2.1 กำหนดข้อตกลงเบื้องต้นของตัวแบบที่ศึกษา

ทำการสร้างแบบจำลองทางคณิตศาสตร์ที่แทนปัญหาสินค้าคงคลังสองระดับ เมื่อกำหนดให้ความต้องการสินค้าไม่แน่นอน และหาคำตอบด้วยวิธีเมตาฮิวริสติก โดยมีข้อตกลงเบื้องต้นของตัวแบบที่ศึกษาดังนี้

2.1.1 การกำหนดรูปแบบปัญหาสินค้าคงคลังสองระดับที่มีลักษณะผู้ขายหรือผู้ผลิตหนึ่งรายส่งสินค้าหลายรายการไปยังร้านค้า 1 แห่ง

2.1.2 รูปแบบปัญหาที่ศึกษาอาจเกิดสินค้าขาดมือ

2.1.3 ความต้องการสินค้าแต่รายการเป็นอิสระกัน และมีลักษณะไม่แน่นอน (Uncertainty) โดยกำหนดให้ความต้องการสินค้ารายการที่ i มีการแจกแจงแบบเอกรูป (Uniform distribution) อยู่ในช่วง (D_i^{\min}, D_i^{\max})

2.2 กำหนดตัวแปรตัดสินใจและพารามิเตอร์ของตัวแบบ

กำหนดให้ i แทนรายการสินค้าที่จัดส่งจากผู้ขาย (Vendor) ไปยังผู้ซื้อ (Buyer) มีจำนวนทั้งหมด p รายการ ($i=1, 2, \dots, p$)

2.2.1 ตัวแปรตัดสินใจ

ให้ n_i^o = จำนวนครั้งของการจัดส่งสินค้ารายการที่ i จากผู้ขายไปยังผู้ซื้อ

M_i = จำนวนของแพ็คเกจสำหรับการส่งสินค้ารายการที่ i

n_i = จำนวนสินค้ารายการที่ i ที่บรรจุในแต่ละแพ็คเกจ

r_i = จุดสั่งซื้อสินค้ารายการที่ i

Q_i = จำนวนการสั่งซื้อสินค้ารายการที่ i โดยเฉลี่ย ซึ่งคำนวณจาก $n_i M_i$

n_i^o, M_i, r_i และ Q_i เป็นจำนวนเต็มที่มากกว่าหรือเท่ากับศูนย์

2.2.2 พารามิเตอร์ที่ใช้ในตัวแบบ

Y_i เป็นตัวแปรสุ่มที่แทนความต้องการสินค้ารายการที่ i ซึ่งมีความไม่แน่นอน (Uncertainty Demand) มีลักษณะเป็นการแจกแจงแบบเอกรูปไม่ต่อเนื่อง ที่มีค่าต่ำสุดเป็น D_i^{\min} และมีค่าสูงสุดเป็น D_i^{\max} ($Y_i \sim U(D_i^{\min}, D_i^{\max})$) โดยค่าคาดหวังของความต้องการสินค้ารายการที่ i เขียนแทนด้วย D_i คำนวณได้จากสูตร

$$D_i = \frac{(D_i^{\min} + D_i^{\max})}{2} \quad (1)$$

RD_i แทนพิสัยของความต้องการสินค้ารายการที่ i ($RD_i = D_i^{\max} - D_i^{\min}$)

P_i เป็นอัตราการผลิตสินค้ารายการที่ i โดยอัตราการผลิตสินค้ารายการที่ i จะน้อยกว่าความต้องการสินค้า จึงจะมีสินค้าคงคลัง ($P_i \geq D_i$)

SL_i เป็นระดับการบริการลูกค้าสำหรับสินค้ารายการที่ i หรือความน่าจะเป็นที่ความต้องการสินค้าของลูกค้าไม่เกินกว่าปริมาณปริมาณสินค้าคงคลังที่มีอยู่

C_i เป็นราคาสินค้าต่อหน่วย (Purchasing price per unit) สำหรับสินค้ารายการที่ i

A_i เป็นค่าใช้จ่ายในการจัดซื้อ (ต่อหนึ่งคำสั่งซื้อ) สำหรับสินค้ารายการที่ i

A_i^t เป็นค่าใช้จ่ายในการขนส่งสินค้าของผู้ซื้อ (ต่อหนึ่งเที่ยว) สำหรับสินค้ารายการที่ i

A_i^p เป็นค่าใช้จ่ายในการส่งผลิตของผู้ซื้อสำหรับการตั้งคำสั่งรายการที่ i แต่ละครั้ง

h_i^v เป็นค่าใช้จ่ายในการถือครองสินค้าต่อหน่วยต่อหน่วยเวลาของผู้ขาย (Vendor's holding cost) สำหรับสินค้ารายการที่ i

h_i^b เป็นค่าใช้จ่ายในการถือครองสินค้าต่อหน่วยต่อหน่วยเวลาของผู้ซื้อ (Buyer's holding cost) สำหรับสินค้ารายการที่ i

B_i เป็นอัตราร้อยละของความต้องการสินค้ารายการที่ i ที่ไม่ได้รับการตอบสนองที่ถูกจัดส่งในภายหลัง (Back order)

s_i เป็นค่าใช้จ่ายต่อหน่วยในการจัดส่งสินค้ารายการที่ i ภายหลัง (Back ordered cost)

\tilde{s}_i เป็นค่าใช้จ่ายต่อหน่วยเนื่องจากสินค้ารายการที่ i ไม่พอจำหน่าย เกิดการสูญเสียการขาย (Lost sale)

LT_i เป็นเวลานำการผลิตสำหรับสินค้ารายการที่ i (the i^{th} Product lead time) คำนวณจาก

$$LT_i = \frac{Q_i}{P_i} + \gamma_i \quad (2)$$

เมื่อ γ_i คือเวลาที่เกิดการล่าช้าจากการขนส่งหรือผลิต

2.3 สร้างตัวแบบสินค้าคลังสองระดับ

ตัวแบบสินค้าคลังสองระดับพัฒนาจากตัวแบบสินค้าคลังสองระดับของ Taleizadeh และคณะ [11] งานวิจัยนี้กำหนดให้มีผู้ขาย 1 ราย ผู้ซื้อ 1 ราย แต่มีสินค้าที่ซื้อขายหลายรายการ ภายใต้ความต้องการสินค้ารายการที่ i มีการแจกแจงแบบเอกรูป ช่วงความต้องการสินค้าอยู่ระหว่าง D_i^{min} และ D_i^{max} โดยค่าใช้จ่ายรวมของระบบสินค้าคลังสองระดับประกอบด้วยค่าใช้จ่ายคาดหวังรวมของทางผู้ซื้อต่อหน่วยเวลา (Buyer's total expected cost per unit time) และค่าใช้จ่ายรวมของทางผู้ขายสำหรับสินค้ารายการที่ i (Vendor's total cost for the i^{th} product) โดยทางด้านผู้ซื้อจะมีค่าใช้จ่ายในการสั่งซื้อสินค้า ค่าใช้จ่ายในการเก็บสินค้า ค่าใช้จ่ายที่จัดส่งสินค้าที่หลังกรณีมีสินค้าไม่พอ (Back order cost) ค่าใช้จ่ายที่เกิดการสูญเสียการขายกรณีมีสินค้าไม่พอ (Lost sale cost) และค่าใช้จ่ายในการไปรับสินค้า สำหรับทางด้านผู้ขาย มีค่าใช้จ่ายในการเปลี่ยนรุ่นการผลิตหรือค่าติดตั้งเครื่องจักร (Setup cost) และค่าใช้จ่ายในการจัดเก็บสินค้าของผู้ขาย ซึ่งมีสมการเป้าหมายและสมการข้อจำกัดของตัวแบบดังนี้

2.3.1 สมการเป้าหมาย

$$\text{Min TC} = \sum_{i=1}^p \frac{(A_i + A_i^p)D_i}{n_i^0} + \sum_{i=1}^p h_i^b \left[\frac{n_i M_i}{2} + r_i \right.$$

$$\left. - D_i \left(\frac{n_i M_i}{P_i} + \gamma_i \right) + \frac{(1 - B_i)(D_i^{\text{Max}} - r_i)^2}{2(D_i^{\text{Max}} - D_i^{\text{Min}})} \right] + \sum_{i=1}^p \frac{(s_i B_i + \tilde{s}_i(1 - B_i)D_i(D_i^{\text{Max}} - r_i)^2}{2n_i^0 n_i M_i (D_i^{\text{Max}} - D_i^{\text{Min}})} + \sum_{i=1}^p \frac{D_i A_i^t}{n_i M_i} + \sum_{i=1}^p h_i^v \frac{n_i M_i}{2} \left[n_i^0 \left(1 - \frac{D_i}{P_i} \right) - 1 + \frac{2D_i}{P_i} \right] \quad (3)$$

2.3.2 สมการข้อจำกัด

ณ จุดสั่งซื้อต้องมีสินค้ามากกว่าหรือเท่ากับความต้องการสินค้าแต่ละรายการที่ระดับบริการ SL_i และจำนวนการสั่งซื้อต้องเท่ากับจำนวนสินค้าในแต่ละแพ็คเกจ (n_i) คูณกับจำนวนแพ็คเกจ (M_i) แสดงดังสมการต่อไปนี้

$$r_i \geq \left(\frac{n_i M_i}{P_i} + \gamma_i \right) + \left[(D_i^{\text{Max}} - D_i^{\text{Min}})SL_i + D_i^{\text{Max}} \right] \quad (4)$$

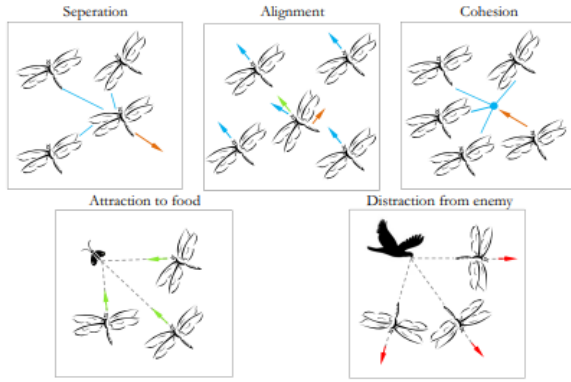
$$Q_i = n_i M_i \quad (5)$$

$$n_i^0, M_i, r_i, Q_i \geq 0 \quad \forall i; i = 1, 2, \dots, p \quad (6)$$

2.4 ศึกษาวิธีเมตาฮิวริสติกสำหรับการแก้ปัญหา

2.4.1 ขั้นตอนวิธีแมลงปอ (Dragonfly Algorithm; DA)

Mirjalili พัฒนาวิธี DA ซึ่งเป็นวิธีเมตาฮิวริสติก โดยเลียนแบบพฤติกรรมการจับกลุ่มของฝูงแมลงปอที่มีการจับกลุ่มแบบกลุ่มใหญ่แบบคงที่และสร้างกลุ่มย่อยบินผ่านพื้นที่ต่าง ๆ (static and dynamic swarming behaviours) เพื่อบินสำรวจและหาอาหาร โดยแมลงปอจะบินเป็นฝูงใหญ่และไปในทิศทางเดียวกันเพื่อหาอาหาร [9] อัลกอริทึมนี้ออกแบบให้เป็น 2 ระยะเลียนแบบพฤติกรรมของแมลงปอ โดยระยะแรกจะรวมกลุ่มแบบคงที่ และระยะที่สองแมลงปอจะสร้างฝูงย่อยเพื่อออกสำรวจและค้นหาอาหาร โดยไปในทิศทางเดียวเพื่อเจอตำแหน่งที่เหมาะสมที่สุด การเคลื่อนที่ของแมลงปอแบ่งได้เป็น 5 ลักษณะ ได้แก่ การสร้างฝูงย่อย (Separation) การจัดทิศทาง (Alignment) การติดต่อกัน (Cohesion) การบินเข้าหาอาหาร (Attraction to food) และการบินหนีจากศัตรู (Distraction from enemy) แสดงดังรูปที่ 1



รูปที่ 1 ลักษณะการเคลื่อนที่ของแมลงปอ ที่มา: [9]

จากพฤติกรรมเคลื่อนที่ของแมลงปอ กำหนดให้ P แทนตำแหน่งปัจจุบันของแมลงปอตัวที่สนใจ (ตัวที่ i) P_j แทนตำแหน่งของตัวที่ j ที่อยู่ในตำแหน่งข้างเคียง และ m แทนจำนวนแมลงปอที่อยู่ข้างเคียง สมการที่ใช้แทนการเคลื่อนที่แสดงได้ดังนี้

2.4.1.1 การสร้างฝูงย่อยแทนด้วยสมการ

$$S_i = - \sum_{j=1}^m P - P_j \quad (7)$$

2.4.1.2 การจัดทิศทางแทนด้วยสมการ

$$A_i = \frac{\sum_{j=1}^m V_j}{m} \quad (8)$$

2.4.1.3 การติดต่อกันแทนด้วยสมการ

$$C_i = \frac{\sum_{j=1}^m P_j}{m} - P \quad (9)$$

2.4.1.4 การบินเข้าหาอาหารแทนด้วยสมการ

$$F_i = F_p - P \quad (10)$$

2.4.1.5 การบินหนีจากศัตรูแทนด้วยสมการ

$$E_i = E_p - P \quad (11)$$

รหัสเทียม (Pseudo-codes) ของ DA [12] แสดงดังต่อไปนี้

ขั้นที่ 1 กำหนดตำแหน่งเริ่มต้นของแมลงปอแต่ละตัว P_i เมื่อ

$$i = 1, 2, \dots, m$$

ขั้นที่ 2 กำหนด step vectors เริ่มต้น (ΔP_i เมื่อ $i = 1,$

$$2, \dots, m)$$

ขั้นที่ 3 ถ้า $t <$ ค่าสูงสุดของจำนวนรอบการค้นหาคำตอบ ให้ทำขั้นตอนที่ 4 ถึง 13

ขั้นที่ 4 คำนวณค่าฟิตเนสของแมลงปอแต่ละตัว $f(P_i)$

ขั้นที่ 5 ปรับปรุงตำแหน่งของอาหารและศัตรู

ขั้นที่ 6 ปรับปรุงค่า s, a, c, f, e และ w ซึ่งเป็นค่าถ่วงน้ำหนักซึ่งมีผลต่อการลู่เข้าของคำตอบ

ขั้นที่ 7 คำนวณค่า S, A, C, F, E ตามสมการ (7) ถึง (11)

ขั้นที่ 8 ปรับปรุงรัศมีของตำแหน่งข้างเคียง

ขั้นที่ 9 ถ้ามีแมลงปออยู่ใกล้เคียงอย่างน้อย 1 ตัวให้ปรับปรุงเวกเตอร์ความเร็ว ตามขั้นตอนที่ 10 และปรับปรุงตำแหน่งตามขั้นตอนที่ 11 ถ้าไม่เป็นตามขั้นที่ 9 ให้ข้ามไปทำขั้นตอนที่ 12

ขั้นที่ 10 ปรับปรุง ΔP_i^{t+1} ตามสมการที่ (12)

$$\Delta P_i^{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + w\Delta P_i^t \quad (12)$$

ขั้นที่ 11 ปรับปรุงเวกเตอร์ตำแหน่งในตามสมการที่ (13)

$$P_i^{t+1} = P_i^t + \Delta P_i^{t+1} \quad (13)$$

ขั้นที่ 12 ถ้าไม่เป็นตามขั้นตอนที่ 9 ให้ปรับปรุงค่าตำแหน่งด้วยสมการที่ (14)

$$P_i^{t+1} = P_i^t + 0.01 \left(\frac{r_1}{|r_2|^\beta} \right) \sigma \times P_i^t \quad (14)$$

เมื่อ r_1, r_2 เป็นเวกเตอร์สุ่มที่มีการแจกแจงแบบเอกรูปในช่วง $[0,1]$ β เป็นค่าคงที่ และ σ คำนวณจากสมการที่ (15)

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\Gamma\beta}{2}\right)}{\Gamma \times 2 \left(\frac{\beta-1}{2}\right) \times \beta \times \left(\frac{1+\beta}{2}\right)} \right)^{\frac{1}{\beta}} \quad (15)$$

$$\Gamma(a) = \int_0^\infty (t^{a-1} e^{-t}) dt \quad (16)$$

เมื่อ a เป็นจำนวนเต็ม จะได้ดังสมการที่ (17)

$$\Gamma(a) = (a-1)! \quad (17)$$

ขั้นที่ 13 ตรวจสอบและแก้ไขตำแหน่งใหม่ตามขอบเขตของตัวแปร

2.4.2 วิธีการหาค่าเหมาะที่สุดแบบกลุ่มอนุภาค (Particle Swarm Optimization; PSO)

Wilson นักวิจัยชาวอังกฤษ [13] ได้ศึกษาวิธีการหาอาหารของฝูงนก ซึ่งเป็นแนวคิดพื้นฐานในการพัฒนาวิธีการหาค่าเหมาะที่สุดแบบกลุ่มอนุภาค โดยศึกษาลักษณะการหาอาหารร่วมกันของฝูงนกที่หาอาหารของนกแต่ละตัวในฝูง และให้นกตัวอื่นสามารถเข้าถึงแหล่งอาหารได้โดยไม่เกิดการแย่งชิงอาหารระหว่างนกแต่ละตัวภายในฝูง

การหาค่าเหมาะที่สุดแบบกลุ่มอนุภาค (PSO) จะแทนค่าตอบที่เป็นไปได้ด้วยตำแหน่งของอนุภาค (นกแต่ละตัว) อนุภาคทุกตัวจะมีค่าฟิตเนสที่หาได้จากฟังก์ชันฟิตเนส (ฟังก์ชันวัตถุประสงค์) ที่ใช้ในการหาค่าเหมาะที่สุด ค่าความเร็วในการเดินทางที่จะเป็นตัวกำกับระยะการเดินทางของอนุภาคโดยอนุภาคทุกตัวจะเดินตามอนุภาคที่อยู่ใกล้ตำแหน่งที่เหมาะสมที่สุด (g_{best}) ซึ่งในการเคลื่อนที่แต่ละครั้งอนุภาคจะจดจำตำแหน่งที่ดีที่สุดของตนที่พบมา (p_{best}) และจะทำการปรับความเร็วด้วยค่าองค์ประกอบความเร่งจากนั้นนำความเร็วที่ปรับแล้วไปหาตำแหน่งที่ดีกว่าที่พบมา [14]

กำหนดให้ $x(t)$ คือ ตำแหน่งของอนุภาคหรือคำตอบที่เป็นไปได้ตัวที่ i^{th} มิติ k เมื่อ k เป็นจำนวนพารามิเตอร์ที่ต้องการหาค่าเหมาะที่สุด รอบที่ t โดย

$$x_{ik}(t) = [x_{i,1}(t), x_{i,2}(t), x_{i,3}(t), \dots, x_{i,k}(t)] \quad (18)$$

n คือ จำนวนประชากรหรือจำนวนอนุภาค

$v(t)$ คือ ความเร็วของกลุ่มอนุภาค ซึ่งความเร็วของอนุภาคตัวที่ i^{th} มิติ k ของรอบค้นหาที่ t สามารถเขียนได้เป็น $v_{ik}(t)$

p_{best} คือ ค่าที่ดีที่สุดของแต่ละอนุภาคจะมีการปรับค่าอยู่ตลอดเวลาในระหว่างการหาค่าตอบที่เหมาะสมที่สุด

g_{best} คือ ตำแหน่งที่ดีที่สุดของกลุ่มอนุภาคที่คัดมาจากค่าที่ดีที่สุดของอนุภาคทุกตัวที่ผ่านมาของการค้นหาค่าที่เหมาะสมที่สุด

t_{max} คือ จำนวนรอบในการค้นหาค่าตอบด้วยวิธี PSO

$[X_{min}, X_{max}]$ คือ ช่วงของอนุภาค (Particle range)

c_1, c_2 คือ องค์ประกอบการเร่ง

w คือ ค่าถ่วงน้ำหนัก

รหัสเทียม (Pseudo-codes) ของวิธี PSO แสดงดังต่อไปนี้

ขั้นที่ 1 กำหนดค่าเริ่มต้นโดยทำการสุ่มค่าพารามิเตอร์ตำแหน่งของอนุภาค ($x_{ik}(t)$) ที่อยู่ในช่วง $[X_{min}, X_{max}]$ ด้วยความเร็วของอนุภาค ($v_{ik}(t)$) และหาค่าฟิตเนส $f(x(t))$ ให้กับอนุภาคแต่ละตัว จากนั้นคำนวณหาค่า p_{best} และ g_{best}

ขั้นที่ 2 เพิ่มค่า $t = t+1$

ขั้นที่ 3 ปรับค่าความเร็วของอนุภาคแต่ละตัวจากค่า $v_{i,k}(t)$, $p_{best}(t)$ และ $g_{best}(t)$ ที่ได้จากขั้นตอนขั้นต้นตามสมการที่ (19)

$$v_{i,k}(t+1) = w \times v_{i,k}(t) + c_1 \times r_1 \times (p_{best}(t) - x_{i,k}(t)) + c_2 \times r_2 \times (g_{best}(t) - x_{i,k}(t)) \quad (19)$$

ขั้นที่ 4 : นำค่าความเร็วที่ได้มาปรับค่าตำแหน่งของอนุภาคแต่ละตัวดังสมการที่ (20)

$$x_{i,k}(t+1) = x_{i,k}(t) + v_{i,k}(t+1) \quad (20)$$

ขั้นที่ 5 : คำนวณค่าฟิตเนส $f(x_{i,k}(t+1))$ และหาค่า $p_{best}(t+1)$ และ $g_{best}(t+1)$

ขั้นที่ 6 : ทำซ้ำตั้งแต่ขั้นที่ 2 ถึง 5 จนกว่า $t = t_{max}$ แล้วจึงยุติการค้นหา

2.4.3 ขั้นตอนวิธีค้างคาว (Bat Algorithm; BA)

วิธี BA มีแนวคิดจากพฤติกรรมการหาอาหารของค้างคาว ซึ่งเป็นสัตว์เลี้ยงลูกด้วยนมที่เล็กที่สุด โดยใช้วิธีปล่อยคลื่นเสียงที่มีความถี่สูงของค้างคาว (Sonar) ไปกระทบเหยื่อ แล้วเกิดการสะท้อนมายังหูของค้างคาว ทำให้สามารถรับรู้ตำแหน่งและระยะห่างของสิ่งกีดขวางเพื่อนำทางไปหาเหยื่อ และหลีกเลี่ยงอุปสรรคต่าง ๆ [15] มีลักษณะการทำงานดังต่อไปนี้

2.4.3.1 ค้างคาวค้นหาตำแหน่งของเหยื่อหรือวัตถุ

โดยใช้เสียงสะท้อนกลับเพื่อรับรู้ถึงระยะทางและความแตกต่างระหว่างอาหาร เหยื่อ และสิ่งกีดขวาง

2.4.3.2 ค้างคาวบินอย่างไร้แบบแผน (Randomly)

ด้วยความเร็วที่เกิดจากการเปลี่ยนแปลงความถี่ของสัญญาณชีพ (Pulse) ที่ปล่อยออกมาเรียกว่าความถี่พัลส์ และความ

ดังของเสียงค้างคาว ซึ่งการปล่อยความถี่พัลส์จะเพิ่มขึ้นเรื่อยๆ เมื่อใกล้เหยื่อมากขึ้น

2.4.3.3 ความดังของเสียงค้างคาว (Loudness, A) สามารถเปลี่ยนแปลงได้หลายรูปแบบ ซึ่งความดังจะลดลงเรื่อยๆ เมื่อค้างคาวเข้าใกล้เหยื่อเพิ่มมากขึ้น

2.4.3.4 พารามิเตอร์ของวิธี BA

x_i^t คือ ค่าตอบของค้างคาวตัวที่ i ในรอบการค้นหา ค่าตอบเท่ากับ t ซึ่งมีมิติเท่ากับ k เมื่อ k เป็นจำนวนพารามิเตอร์ที่ต้องการหาค่าเหมาะสมที่สุด

v_i^t คือ ความเร็วของค้างคาวตัวที่ i ในรอบการค้นหา ค่าตอบเท่ากับ t

f_i คือ ความถี่พัลส์ของค้างคาวตัวที่ i

A_i คือ ความดังของเสียงค้างคาวตัวที่ i

r_i คือ อัตราการปล่อยพัลส์ของค้างคาวตัวที่ i

n คือ จำนวนค้างคาวทั้งหมด

t คือ จำนวนรอบในการค้นหาค่าตอบ

ซึ่งพารามิเตอร์ต่างๆสามารถคำนวณได้จากสมการดังต่อไปนี้

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \quad (21)$$

เมื่อ $\beta \in [0,1]$ ถูกสุ่มมาจากการแจกแจงแบบยูนิฟอร์ม

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_*)f_i \quad (22)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (23)$$

$$A_i^{t+1} = \alpha A_i^t \quad (24)$$

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)]; r \in [0,1] \quad (25)$$

รหัสเทียม (Pseudo-codes) ของวิธี BA แสดงดังต่อไปนี้

ขั้นที่ 1 กำหนดค่าพารามิเตอร์ n , t_{\max} , α และ γ

ขั้นที่ 2 สุ่มพารามิเตอร์เริ่มต้นให้กับ x_i^t, v_i^t, r_i, f_i และ A_i

ขั้นที่ 3 ปรับค่าพัลส์ คำนวณค่าความเร็ว และค่าตำแหน่งตามสมการที่ (21), (22) และ (23) ตามลำดับ

ขั้นที่ 4 คำนวณค่าฟิตเนสและหาค่าตอบ

ขั้นที่ 5 สร้างเลขสุ่ม rand1 ($0 < \text{rand1} < 1$)

ขั้นที่ 6 ถ้าเลขสุ่มน้อยกว่า r_i^t คำนวณค่าฟิตเนสใหม่โดย

$$x_{\text{new}} = x_{\text{old}} + \epsilon A_i^t \quad (26)$$

และข้ามไปทำขั้นที่ 8

ขั้นที่ 7 ถ้าไม่เป็นตามขั้นที่ 6 สร้างเลขสุ่มใหม่ ($0 < \text{rand2} < 1$)

ถ้าเลขสุ่มใหม่น้อยกว่า A_i และ $f(x_i(t)) < f(x_0)$ ให้ปรับค่าความดังของเสียงและอัตราการปล่อยพัลส์ของค้างคาวตามสมการที่ (24) และ (25) ตามลำดับ

ขั้นที่ 8 หาค่าตอบที่เหมาะสมสำหรับรอบการค้นหาค่าตอบที่ t จากค่าฟิตเนสที่ต่ำสุด ทำจนครบจำนวนรอบที่กำหนด

สำหรับการคำนวณด้วยวิธีเมตาฮีริสติกส์ในงานวิจัยนี้ได้ใช้แพ็คเกจ ‘metaheuristicOpt’ [16] ในโปรแกรม R

2.5 จำลองสถานการณ์สินค้าคงคลังสองระดับและคำนวณค่าใช้จ่ายรวมของระบบ

ในงานวิจัยนี้ได้สร้างสถานการณ์ของสินค้าคงคลังสองระดับขึ้นมา โดยได้ออกแบบการทดลองให้มีผู้ขาย 1 ราย ผู้ซื้อ 1 ราย โดยกำหนดลักษณะความต้องการสินค้าแต่ละรายการต่อปีไม่แน่นอนมีลักษณะการแจกแจงแบบเอกรูปอยู่ในช่วง (D_i^{\min}, D_i^{\max}) กำหนดจำนวนรายการสินค้าที่ซื้อขายเป็น 2 รายการเนื่องจากเป็นกรณีให้เห็นการเปลี่ยนแปลงของค่าใช้จ่ายได้น้อยที่สุด ซึ่งแบ่งเป็น 4 กรณีดังนี้

กรณีที่ 1 สินค้าทั้งสองรายการมีความต้องการเฉลี่ยและพิสัยเท่ากัน

กรณีที่ 2 สินค้าทั้งสองรายการมีความต้องการเฉลี่ยเท่ากันแต่พิสัยต่างกันเล็กน้อย (RD_i ต่างกัน 500)

กรณีที่ 3 สินค้าทั้งสองรายการมีความต้องการเฉลี่ยเท่ากันแต่พิสัยแตกต่างกันมาก (RD_i ต่างกัน 2,000)

กรณีที่ 4 สินค้าทั้งสองรายการมีความต้องการเฉลี่ยและพิสัยแตกต่างกัน

ความต้องการสินค้าต่ำสุด (D_i^{\min}) ความต้องการสินค้าสูงสุด (D_i^{\max}) ความต้องการสินค้าเฉลี่ย (D_i) และพิสัยของความต้องการสินค้า (RD_i) แสดงดังตารางที่ 1

กำหนดให้เวลาที่เกิดการล่าช้าจากการขนส่งหรือผลิตเป็น 1 สัปดาห์ ($\gamma_i = 0.02$) อัตราการผลิตสินค้ารายการที่ i เป็น 15,000 ($P_i = 15,000$ ซึ่ง $P_i \geq D_i$) ระดับการให้บริการเป็นร้อยละ 90 $A_i^t = 1,000$ และ $B_i = 0.9$

ในแต่ละกรณีที่ศึกษาแบ่งออกเป็นกรณีย่อยอีก 2 กรณี คือ

กรณีย่อยที่ 1 ค่าพารามิเตอร์ $n_i, C_i, s_i, \xi_i, A_i, A_i^P, h_i^V$ และ h_i^b ของสินค้ารายการที่ 1 และรายการที่ 2 เท่ากัน โดย $n_i = 10, C_i = 50, s_i = 100, \xi_i = 120, A_i = 200, A_i^P = 150, h_i^V = 10$ และ $h_i^b = 15$

กรณีย่อยที่ 2 กำหนดให้ค่าพารามิเตอร์ของสินค้าทั้งสองรายการแตกต่างกัน สินค้ารายการที่ 1 $n_1 = 10, C_1 = 50, s_1 = 100, \xi_1 = 120, A_1 = 200, A_1^P = 150, h_1^V = 10$ และ $h_1^b = 15$ สินค้ารายการที่ 2 กำหนดให้ $n_2 = 12, C_2 = 100, s_2 = 120, \xi_2 = 150, A_2 = 250, A_2^P = 170, h_2^V = 15$ และ $h_2^b = 25$

ตารางที่ 1 ความต้องการสินค้าแต่ละกรณีการศึกษา

กรณีที่ 1				
รายการที่	D_i^{min}	D_i^{max}	D_i	RD_i
1	10,000	10,500	10,250	500
2	10,000	10,500	10,250	500
กรณีที่ 2				
รายการที่	D_i^{min}	D_i^{max}	D_i	RD_i
1	10,000	11,000	10,500	1,000
2	9,750	11,250	10,500	1,500
กรณีที่ 3				
รายการที่	D_i^{min}	D_i^{max}	D_i	RD_i
1	10,000	13,000	11,500	3,000
2	9,000	14,000	11,500	5,000

ตารางที่ 1 (ต่อ)

กรณีที่ 4				
รายการที่	D_i^{min}	D_i^{max}	D_i	RD_i
1	10,000	11,000	10,500	1,000
2	10,000	13,000	11,500	3,000

การกำหนดค่าพารามิเตอร์สำหรับแต่ละวิธี วิธี DA กำหนด s และ $a = 0.1, c = 0.7, f$ และ $e = 1$ ตามงานวิจัยของ Mirjalili [9] กับ Huang และคณะ [17] ซึ่งเปรียบเทียบวิธี DA ที่ใช้พารามิเตอร์ดังกล่าวกับวิธี PSO ที่ใช้ $c_1 = 2, c_2 = 2, w = 1$ โดยสอดคล้องกับงานวิจัยของ Taleizadeh และคณะ [11] ที่ใช้วิธี PSO แก้ปัญหาสินค้าคงคลังที่มีผู้ซื้อและผู้ขาย 1 ราย ลักษณะปัญหาใกล้เคียงกับปัญหาที่ศึกษา โดยให้ขอบเขตที่เหมาะสมของค่า c_1, c_2 และ w อยู่ที่ $[1.5, 2], [2, 2.5]$ และ $[0.4, 1.4]$ ตามลำดับ ผู้วิจัยปรับค่าภายใต้ขอบเขตดังกล่าว พบว่าหลายกรณี $c_1 = 2, c_2 = 2.5, w = 1$ ให้คำตอบดีกว่าจึงกำหนดเป็นค่าพารามิเตอร์สำหรับวิธี PSO ส่วนวิธี BA กำหนดค่าตาม ศิริรัตน กะการดี และ ศิริประภา มโนมัย [8] ได้ค่า $f_{max} = 100, f_{min} = 0.1, \alpha = 0.1$ และ $\gamma = 1$ ซึ่งค่า α จะอยู่ในช่วง $(0, 1)$ และ $\gamma \geq 1$ [16] สำหรับจำนวนประชากร (pop size) ทั้ง 3 วิธี กำหนดเป็น 500 ทำการทดลองภายใต้จำนวนรอบของการค้นหาคำตอบเป็น 10, 30, 50, 100, 300, 500 และ 1,000 ตามลำดับ สังเกตการเปลี่ยนแปลงของคำตอบ

3. ผลการทดลอง

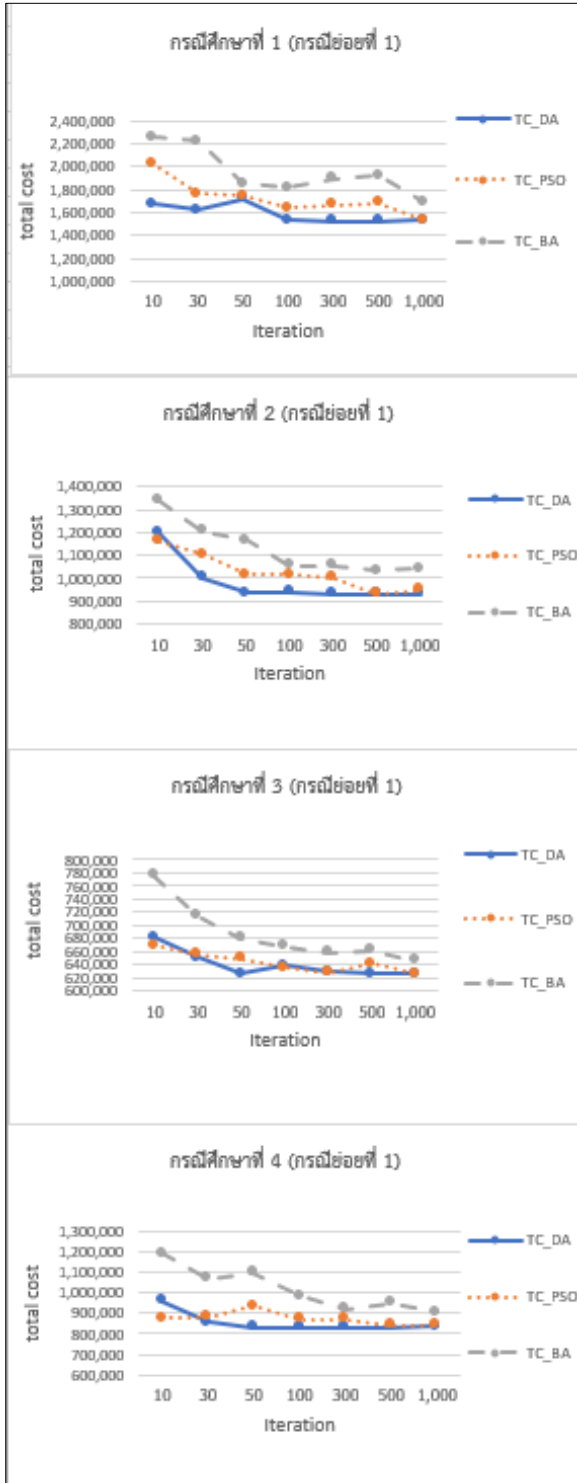
จากการจำลองสถานการณ์ที่ศึกษาและคำนวณคำตอบของตัวแบบที่ศึกษา ได้ผลการคำนวณทั้ง 4 กรณีศึกษา และ 2 กรณีย่อย แสดงดังรูปที่ 2 และรูปที่ 3

จากรูปที่ 2 และ 3 พบว่าการคำนวณด้วยวิธี DA จะเข้าสู่คำตอบที่เหมาะสมได้เร็วกว่าวิธี PSO และ BA โดยวิธี DA จะเริ่มเข้าสู่คำตอบที่เหมาะสมที่จำนวนรอบของการค้นหาคำตอบเป็น 50 ในขณะที่วิธี PSO จะเริ่มเข้าสู่คำตอบเมื่อจำนวนรอบของการค้นหาคำตอบสูงกว่า 500

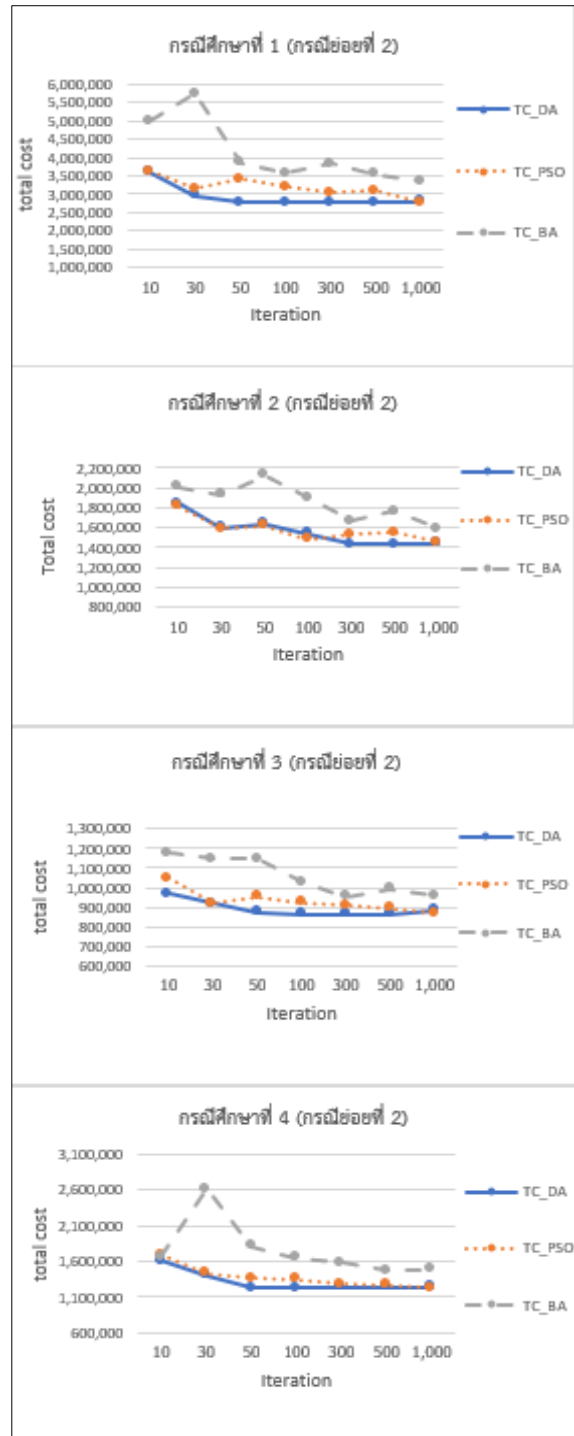
สำหรับวิธี BA มีประสิทธิภาพในการหาคำตอบน้อยกว่าวิธีอื่น ๆ (จำนวนรอบที่ 1,000 รอบ คำตอบของวิธี BA ยังไม่ให้ค่าที่ต่ำและไม่เสถียร) เมื่อนำวิธี PSO และวิธี BA มาเทียบกับวิธี DA โดยพิจารณาจากร้อยละของความแตกต่างระหว่างค่าใช้จ่ายรวม (TC) ที่ใช้วิธีการค้นหาคำตอบแบบ PSO หรือ BA เทียบกับ DA ตามสมการที่ (27)

$$\%diff = \frac{(TC_{PSO} \text{ or } TC_{BA} - TC_{DA})}{TC_{DA}} \times 100 \quad (27)$$

ผลการคำนวณแสดงดังตารางที่ 2



รูปที่ 2 การลู่ออกคำตอบกรณีศึกษาที่ 1 ถึง 4 (สำหรับกรณีย่อยที่ 1)



รูปที่ 3 การลู่ออกคำตอบกรณีศึกษาที่ 1 ถึง 4 (สำหรับกรณีย่อยที่ 2)

จากตารางที่ 2 พบว่าในบางกรณีที่จำนวนรอบของการค้นหาคำตอบเป็น 10 วิธี PSO จะให้คำตอบต่ำกว่าวิธี DA (ค่า %diff ติดลบ) แต่ยังไม่เป็นคำตอบที่เหมาะสมที่สุดซึ่งพบในกรณีที่ค่าเฉลี่ยและพิสัยของความต้องการสินค้าไม่เท่ากัน (กรณีศึกษาที่ 2-4) แต่เมื่อรอบของการค้นหาคำตอบมากกว่า 500 วิธี DA และ PSO มีประสิทธิภาพแตกต่างกันไม่มาก ในขณะที่วิธี BA เมื่อนำไปเทียบกับวิธี DA พบว่าคำตอบมีความแตกต่างกันมาก โดยมี %diff สูงสุดถึง 95.33% (กรณีศึกษาที่ 1 กรณีย่อยที่2)

ตารางที่ 2 การเปรียบเทียบวิธีการค้นหาคำตอบทั้ง 3 วิธี

t	กรณีที่1 (ย่อย1)		กรณีที่1 (ย่อย2)	
	PSO/DA	BA/DA	PSO/DA	BA/DA
10	20.87%	34.40%	0.79%	38.95%
30	8.74%	36.92%	6.41%	95.33%
50	1.25%	7.50%	23.15%	39.70%
100	7.16%	18.87%	15.04%	29.17%
300	9.17%	24.25%	9.72%	38.20%
500	9.97%	25.96%	11.64%	27.95%
1,000	0.03%	10.55%	0.44%	21.35%
t	กรณีที่2 (ย่อย1)		กรณีที่2 (ย่อย2)	
	PSO/DA	BA/DA	PSO/DA	BA/DA
10	-2.86%	12.18%	-1.55%	9.16%
30	10.24%	20.57%	-0.41%	21.01%
50	8.91%	24.74%	-1.06%	29.70%
100	8.06%	12.28%	-3.52%	23.73%
300	7.56%	13.37%	6.87%	16.67%
500	0.04%	11.36%	8.43%	23.13%
1,000	2.10%	12.23%	0.40%	9.70%
t	กรณีที่3 (ย่อย1)		กรณีที่3 (ย่อย2)	
	PSO/DA	BA/DA	PSO/DA	BA/DA
10	-1.82%	13.89%	8.30%	21.69%
30	0.76%	9.82%	-0.06%	24.93%
50	3.77%	8.79%	8.98%	31.60%
100	-0.29%	4.69%	7.09%	19.20%
300	-0.37%	4.58%	5.13%	10.51%
500	2.59%	5.70%	3.03%	14.60%

t	กรณีที่4 (ย่อย1)		กรณีที่4 (ย่อย2)	
	PSO/DA	BA/DA	PSO/DA	BA/DA
	1,000	-0.01%	3.26%	-1.87%
10	-8.89%	24.40%	5.13%	3.30%
30	2.95%	24.62%	1.56%	85.84%
50	11.81%	31.69%	10.98%	47.71%
100	4.74%	17.90%	10.15%	34.95%
300	4.80%	10.35%	3.20%	28.15%
500	0.77%	13.99%	2.60%	19.57%
1,000	0.65%	8.58%	0.26%	21.22%

4. สรุปผลและข้อเสนอแนะ

ตัวแบบสินค้าคงคลังสองระดับเป็นปัญหาที่ซับซ้อนซึ่งขึ้นอยู่กับลักษณะและเงื่อนไขของการจัดการสินค้าคงคลัง การหาคำตอบด้วยวิธีการทางคณิตศาสตร์ธรรมดาอาจทำได้ยาก จึงนำวิธีเมตาฮีริสติกมาใช้ในการหาคำตอบ งานวิจัยนี้แสดงให้เห็นว่าวิธี DA มีประสิทธิภาพและลู่เข้าสู่คำตอบที่เหมาะสมที่สุดโดยใช้จำนวนรอบการค้นหาคำตอบน้อยกว่าวิธี PSO และ BA โดยในตัวแบบที่ศึกษาจะเริ่มลู่เข้าสู่คำตอบที่จำนวนรอบของการค้นหาคำตอบเป็น 50 ในขณะที่วิธี PSO เริ่มลู่เข้าสู่คำตอบที่จำนวนรอบของการค้นหาคำตอบมากกว่า 500 สำหรับวิธี BA จำนวนรอบของการค้นหาคำตอบเป็น 1,000 ยังไม่ลู่เข้าสู่คำตอบที่เหมาะสมที่สุด อย่างไรก็ตามประสิทธิภาพของวิธีขึ้นกับค่าพารามิเตอร์ที่กำหนด ผู้วิจัยได้ศึกษาและทำการทดลองปรับค่าพารามิเตอร์ของตัวแบบสินค้าคงคลังที่คาดว่ามีความสำคัญ ถึงแม้ว่าค่าของคำตอบที่คำนวณได้ไม่เท่ากันแต่ทิศทางของคำตอบส่วนใหญ่เป็นไปในลักษณะเดียวกัน กล่าวคือ คำตอบที่คำนวณจากวิธี DA จะเริ่มเสถียรเมื่อใช้จำนวนรอบของการค้นหาคำตอบตั้งแต่ 50 รอบขึ้นไป ซึ่งผลจากการวิจัยสามารถนำไปเป็นแนวทางในการเลือกวิธีการหาคำตอบของปัญหาสินค้าคงคลังสองระดับที่มีลักษณะความต้องการไม่แน่นอน เพื่อใช้ในการตัดสินใจอย่างมีประสิทธิภาพได้

5. กิตติกรรมประกาศ

งานวิจัยนี้ได้รับทุนสนับสนุนการวิจัย จากคณะวิทยาศาสตร์ประยุกต์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้า

พระนครเหนือ สัญญาทุนเลขที่ 6345103 ผู้วิจัย
ขอขอบพระคุณมา ณ ที่นี้ด้วย

6. เอกสารอ้างอิง

- [1] วราภรณ์ พกนนท์, ปวีณา เขาวลิตวงศ์ และนระเกณท์ พุ่มชูศรี, “นโยบายการเติมเต็มร่วมแบบสามารถจัดหาได้ สำหรับระบบแบบสองระดับ ภายใต้การศึกษาระยะเวลา,” *วารสารวิศวกรรมศาสตร์*, ปีที่ 5, ฉบับที่ 2, น. 27-47, 2557.
- [2] จารุมาศ แสงสว่าง และ วุฒิชัย ศรีโสตาพล, “ตัวแบบสินค้าคงคลังแบบสองระดับที่มีเวลานำและการคืนสินค้าขึ้นอยู่กับความต้องการสินค้า,” *วารสารวิทยาศาสตร์บูรพา*, ปีที่ 22, ฉบับที่ 2, น. 145-153, 2560.
- [3] P. Yang and L. Zeng, “Model and method for Two-echelon location routing problem with time constraints in city logistics,” *Mathematical Problem in Engineering*, 2018.
- [4] H. Sadjady and H. Davoudpour, “Two-echelon, multi-commodity supply chain network design with mode selection lead-times and inventory costs,” *Computers & Operations Research*, vol. 39, pp. 1345-1354, 2012.
- [5] A. Mahmoodi, A. Haji and R. Haji, “A Two-echelon inventory model with perishable items and lost sales,” *Scientia Iranica*, vol.23, no.5, pp. 2277-2286, 2016
- [6] M. G. Samani and S. M. Hosseimi-Motlahg, “A Hybrid algorithm for a Two-echelon location-routing problem with simultaneous pickup and delivery under fuzzy demand,” *International Journal of Transportation Engineering*, vol.5, no.1, pp. 59-85, 2017.
- [7] M. Seifbarghy, M. M. Kalani, M. Hemmati, “A Discrete Particle Swarm Optimization algorithm with local search for a production-based Two-echelon single-vendor multiple-buyer supply chain,” *Journal of Industrial Engineering International*, vol. 12, pp. 29-43, 2016.
- [8] ศิริรัตน์ กะการดี และ ศิริประภา มโนมัยย์, “การเปรียบเทียบประสิทธิภาพวิธีการหาค่าเหมาะสมสุดแบบกลุ่มอนุภาคและวิธีแบบสำหรับตัวแบบสินค้าคงคลัง 3 ระดับชั้น,” *วารสารไทยการวิจัยดำเนินงาน*, ปีที่ 8, ฉบับที่ 1, น. 11-19, 2563.
- [9] S. Mirjalili “Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems,” *Neural Computing and Applications*, vol. 27, pp. 1053-1073, 2016.
- [10] C. M. Rahman and T. A. Rashid, “A Survey on Dragonfly Algorithm and its applications in engineering,” [Online]. Available: 10.1093/jcde/qvaa037. [Accessed 27 May 2521].
- [11] A. A. Taleizadeh, S.T.A. Niaki, N. Shafii, R.G. Meibodi and A. Jabbarzadeh, “A Particle Swarm Optimization approach for constraint joint single buyer-single vendor inventory problem with changeable lead time and (r,Q) policy in supply chain,” *International Advanced Manufacturing Technology*, vol 51, pp.1209-1223, 2010.
- [12] Y. Meraihi, A. R. Cherif, D. Acheli and M. Mahseur, “Dragonfly Algorithm: a comprehensive review and applications,” *Neural Computing and Applications*, vol. 32, pp.16625-16646, 2020.
- [13] E. O. Wilson, *Sociobiology: The New Synthesis*, Cambridge: Belknap Press, 1975.
- [14] J. Kennedy, R. Eberhart, “Particle Swarm Optimization,” in *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia, pp. 1942-1945, 1995.
- [15] X. S. Yang, “A New Metaheuristic Bat-Inspired Algorithm in J.R. González, D.A. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor (eds.),” *Nature*

Inspired Cooperative Strategies for Optimization Studies in Computational Intelligence., vol. 284, pp. 65-74, 2010.

- [16] L. S. Riza, M. B. A. Prabowo, E. Junaeti, A. G. Abdullah and K. A. Fariza, "Development and experimentation of R package "metaheuristicOpt" on continuous optimization," *Journal of Engineering Science and Technology.*, vol. 16, no. 2, pp.1006–1018, 2021.
- [17] H. Huang, X. Feng, S. Zhou, J. Jiang, H. Chen, Y. Li and C. Li, "A new fruit fly optimization algorithm enhanced support vector machine for diagnosis of breast cancer based on high-level features," *BMC Bioinformatics.*, vol. 20, no. 290, 2019. doi:10.1186/s12859-019-2771-z