การจัดลำดับการผลิตรถยนต์แบบหลายวัตถุประสงค์ บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน

สถาพร โอฬารวิวัฒน์ชัย[1] และ ปารเมศ ชุติมา[2]*

บทคัดย่อ

การจัดลำดับการผลิตรถยนต์บนสายการประกอบแบบสองด้านมีความสำคัญอย่างยิ่งสำหรับใช้ในการแก้ปัญหาสายการประกอบที่มีหลายผลิตภัณฑ์ให้เกิดประสิทธิภาพสูงสุด ซึ่งปัญหาชนิดนี้มีความยุ่งยากและสลับซับซ้อน เนื่องจากเป็นปัญหาแบบ Non-deterministic Polynomial Hard: NP-Hard โดยปัญหาการจัดลำดับการผลิตรถยนต์แบบผลิตภัณฑ์ผสมบนสายการประกอบแบบสองด้านนี้ ได้พิจารณาฟังก์ชันวัตถุประสงค์ 3 ฟังก์ชันในงานวิจัยคือ ปริมาณงานที่ทำไม่เสร็จน้อยที่สุด จำนวนรถยนต์ที่ละเมิดรวมน้อยที่สุด และจำนวนครั้งการเปลี่ยนแปลงสีน้อยที่สุด และนำเสนออัลกอริทึมการบรรจวบแบบขยาย (Combinatorial Optimization with Coincidence Expand: COIN-E) ซึ่งเป็นอัลกอริทึมที่ประยุกต์มาจาก COIN มาใช้ในการแก้ปัญหา โดยทำการเปรียบเทียบกับอัลกอริทึมที่ยอมรับในการแก้ปัญหาการจัดลำดับการผลิต ได้แก่ NSGA-II, DPSO, BBO และ COIN ผลจากการเปรียบเทียบพบว่า COIN-E มีประสิทธิภาพด้านการลู่เข้าสู่กลุ่มคำตอบ ด้านการกระจายกลุ่มคำตอบและด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบกับกลุ่มคำตอบที่แท้จริงเท่ากับ 91.85, 51.08 และ 57.48 ตามลำดับ ซึ่งจากตัวชี้วัดสมรรถนะของทั้ง 3 ชนิดจะพบว่า COIN-E มีประสิทธิภาพในการใช้การแก้ปัญหาได้ดีกว่า NSGAII, DPSO, BBO และ COIN

คำสำคัญ: อัลกอริทึมการบรรจวบแบบขยาย การจัดลำดับการผลิตรถยนต์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน

[1] นิสิต ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

[2] รองศาสตราจารย์ ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

* ผู้นิพนธ์ประสานงาน โทรศัพท์ 0-2218-6847 อีเมล: cparames@chula.ac.th

รับเมื่อ 5 มิถุนายน 2556 ตอบรับเมื่อ 12 กันยายน 2556

Multi-objective Car Sequencing Problem on Mixed-model Two-sided Assembly Lines

Sathaporn Olanviwatchai[1] and Parames Chutima[2]*

Abstract

Car Sequencing on two-sided assembly line is an important problem in an automotive industry. Researchers and practitioners have attempted several approaches to solve this problem aiming at maximum production efficiency. The problem is considered as an "NP-Hard problem". In this paper, three objective functions are considered including 1) minimizing utility work, 2) reducing the number of violation and 3) decreasing the number of color changes. The expansion of Combinatorial Optimization with Coincidence (COIN-E) algorithm is developed from its original version (i.e. COIN). Several well-known algorithms are compared in solving this problem including Non-dominated Sorting Genetic Algorithms (NSGA-II), Discrete Particle Swarm Optimization (DPSO), Biogeography-based Optimization (BBO) and (COIN). The experimental results indicate that COIN-E is efficient and it obtains the values of convergence = 91.85%, spread = 51.08% and ratio = 57.48%, which are significantly superior to NSGA-II, DPSO, BBO and COIN.

Keywords: Expanded Combinatorial Optimization with Coincidence, Car Sequencing, Mixed-model Two-sided Assembly Lines

[1] Student, Department of Industrial Engineering, Faculty of Engineering, Chulalongkorn University.

[2] Associate Professor, Department of Industrial Engineering, Faculty of Engineering, Chulalongkorn University.

* Corresponding Author, Tel. 0-2218-6847, E-mail: cparames@chula.ac.th

Received 5 June 2013; Accepted 12 September 2013

1. บทนำ

สภาพในยุคปัจจุบันนี้โรงงานอุตสาหกรรมมีการออกแบบสายการประกอบสำหรับหลายผลิตภัณฑ์เข้าสู่สายการประกอบพร้อมๆ กัน และต่อเนื่องกัน [1] รวมไปถึงอุตสาหกรรมรถยนต์ที่มีความต้องการจำนวนของรถยนต์สำหรับการผลิตในแต่ละวันที่มีจำนวนแตกต่างกันขึ้นอยู่กับใบสั่งซื้อของลูกค้า โดยการประกอบรถยนต์จะแบ่งออกเป็น 3 ส่วนคือ ส่วนของโครงรถยนต์ (Body) ส่วนของพ่นสี (Paint) และส่วนของการประกอบ (Assembly) ซึ่งรถยนต์แต่ละคันมีเอกลักษณ์เฉพาะของแต่ละตัวเลือก (Option) ที่แตกต่างกัน เช่น รถยนต์รุ่นเดียวกันแต่คันหนึ่งจะสามารถเปิด-ปิดหลังคารถยนต์ได้ แต่อีกคันไม่สามารถเปิด-ปิดหลังคารถยนต์ได้ ดังนั้นการติดตั้งชุดของตัวเลือก จะต้องออกแบบมาเพื่อจัดการไม่ให้จำนวนรถยนต์เกินจำนวนของแต่ละตัวเลือกในลำดับการผลิตย่อยของแต่ละครั้งการผลิต [2] ปัญหาในเรื่องการผลิตสำหรับหลายผลิตภัณฑ์บนสายการประกอบ ก็คือการทำให้ลำดับการผลิตมีประสิทธิภาพสูงสุด โดยขึ้นอยู่กับแต่ละโรงงานอุตสาหกรรมที่จะให้ความสำคัญในเรื่องใด เช่น ต้นทุน/เวลาการปรับตั้งเครื่องน้อยที่สุด อัตราความผันแปรของการผลิตน้อยที่สุด ปริมาณงานที่ทำไม่เสร็จน้อยที่สุดเป็นต้น [1] ในภาคปฏิบัติจริงในอุตสาหกรรมจะมีการพิจารณากับหลายวัตถุประสงค์ ซึ่งในแต่ละวัตถุประสงค์ที่พิจารณานั้นบ่อยครั้งที่วัตถุประสงค์ที่กำลังพิจารณามีความขัดแย้งกันเองเกิดขึ้น [3]

ปัญหาการจัดลำดับการผลิตจึงจัดเป็นปัญหาที่มีความยุ่งยากซับซ้อนและใช้เวลาในการหาคำตอบที่ยาวนาน จึงจำเป็นที่จะต้องนำฮิวริสติกเข้ามาช่วยให้การแก้ปัญหา เนื่องจากฮิวริสติกจะใช้เวลาการหาคำตอบที่ไม่ยาวนานและคำตอบที่ได้ก็เป็นคำตอบที่มีเหมาะสมสำหรับการนำไปใช้ โดยฮิวริสติกที่นิยมกันใช้และเป็นวิธีที่มีการยอมรับว่าสามารถแก้ปัญหาได้ เช่น วิธีเจนเนติกอัลกอริทึม (Genetic Algorithms: GAs) เป็นวิธีที่นิยมนำมาใช้แก้ปัญหาในการจัดลำดับการผลิตที่มีความซับซ้อน โดย GA จะเป็นอัลกอริทึม ที่มีแนวคิดมาจากการตัดต่อทางพันธุกรรมของมนุษย์ โดยอัลกอริทึมนี้ลดเวลาการค้นหาคำตอบ และให้คำตอบที่ใกล้เคียงกับคำตอบที่ดีที่สุดเมื่อเทียบกับอัลกอริทึมอื่นๆ Simon [4] ได้มีการนำเสนออัลกอริทึมใหม่ที่มีชื่อว่า วิธีการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์ (Biogeography Based Optimization: BBO) โดยใช้แนวคิดการย้ายถิ่นฐานของสิ่งมีชีวิตที่มีความต้องการที่จะอยู่อาศัยในที่ที่มีความอุดมสมบูรณ์ แต่ถ้าพื้นที่ที่อยู่นั้นมีความหนาแน่น สิ่งมีชีวิตของจะเริ่มทำการย้ายที่อยู่อาศัยไปสู่ที่อื่นที่มีความอุดมสมบูรณ์น้อยกว่า และพัฒนาที่อยู่ใหม่ให้มีความอุดมสมบูรณ์ โดยวิธีการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์นี้ได้มีการเปรียบเทียบคำตอบที่ได้จากปัญหาในภาคปฏิบัติ ซึ่งพบว่าคำตอบที่ได้จึงมีประสิทธิภาพดีกว่าอัลกอริทึมหลายตัว

Solnon [5] ได้นำเสนอการจัดการแข่งขันปัญหาการจัดลำดับการผลิตรถยนต์เมื่อปี 2005 โดยใช้ข้อมูลจริงจากบริษัทผู้ผลิตรถยนต์ (RENAULT) ในประเทศฝรั่งเศส ในการแข่งขันนั้นมีผู้เข้าร่วมการแข่งขันจะเสนอวิธีการต่างๆในการแก้ปัญหาโดยได้กำหนดวัตถุประสงค์ไว้ 3 วัตถุประสงค์คือ จำนวนการเปลี่ยนแปลงสี จำนวนการละเมิดตัวเลือกความสำคัญสูง และจำนวนการละเมิดตัวเลือกความสำคัญต่ำที่มีค่าน้อยที่สุด โดยผู้ชนะได้เลือกเอาวิธีการ Local Search มาใช้ในการแก้ปัญหาโดยผลที่ได้จากการแข่งขันนี้ได้นำไปสู่งานวิจัยต่อๆ ไปในการใช้เปรียบเทียบผลที่ได้จากการแข่งขันนี้ Knausz [6] ได้ศึกษาการจัดลำดับการผลิตรถยนต์สำหรับผลิตภัณฑ์ผสมที่มีหลายวัตถุประสงค์ โดยนำวิธีการ Parallel Multi Neighbourhood-Order Variable Neighbourhood Search (PMNOVNS) เปรียบเทียบกับ Parallel Efficiency Guided Variable Neighbourhood Search (PEGVNS) และ Time Restricted Randomised Variable Neighbourhood Search (TRRVNS) โดยศึกษา 3 วัตถุประสงค์ คือ จำนวนการเปลี่ยนแปลงสี จำนวนการละเมิดตัวเลือกความสำคัญสูง และจำนวนการละเมิดตัวเลือกความสำคัญต่ำที่มีค่า

น้อยที่สุด ผลสรุปจากงานวิจัยนี้คือ วิธีการ PEGVNS มีประสิทธิภาพมากที่สุดเมื่อเปรียบกับอีกสองวิธีการที่ใช้ในการแก้ปัญหางานวิจัย

Malte Fliedner และ Boysen [7] ได้นำอัลกอริทึม Branch & Bound มาใช้ในการแก้ปัญหาการจัดลำดับในอุตสาหกรรมรถยนต์เพื่อแก้ปัญหาภาระงานเกินในสถานีงาน โดยงานวิจัยนี้ได้นำเสนอวิธีการนับจำนวนการละเมิดแบบใหม่จากเดิมที่ใช้การนับที่เรียกว่า Sliding Window ซึ่งการนับแบบนี้จะมีข้อเสียคือ เกิดการนับจำนวนการละเมิดที่ซ้ำเกิดขึ้นในแต่ละลำดับย่อยที่พิจารณา จึงทำให้ได้จำนวนการละเมิดที่มากเกินความเป็นจริง งานวิจัยนี้จึงได้นำเสนอการนับจำนวนการละเมิดแบบใหม่ขึ้น โดยจะนับจำนวนการละเมิดก็ในลำดับย่อยมีจำนวนตัวเลือกที่เกินจำนวนความต้องการและถ้าได้มีการนับไปแล้วในลำดับย่อยก่อนหน้านี้จะไม่ถูกนับในลำดับย่อยที่กำลังพิจารณาอยู่ ซึ่งการนับนี้จะป้องกันการนับซ้ำ Zinflou [8] ได้นำอัลกอริทึม PMS^{MO} (Pareto Memetic Strategy for Multiple Objective) มาเปรียบเทียบกับ NSGA-II มาใช้ในการแก้ปัญหาการจัดลำดับรถยนต์สำหรับหลายวัตถุประสงค์ สรุปว่าอัลกอริทึม PMS^{MO} เป็นวิธีการที่มีประสิทธิภาพในการแก้ปัญหามากกว่าวิธี NSGA-II Zinflou [9] ได้ศึกษาการจัดลำดับการผลิตผลิตภัณฑ์ผสมที่มีหลายวัตถุประสงค์ โดยนำอัลกอริทึม GISMOO มาเปรียบเทียบกับ NSGA-II กับ PMS^{MO} โดยศึกษา 3 วัตถุประสงค์ คือ จำนวนการเปลี่ยนแปลงสี จำนวนการละเมิดตัวเลือกความสำคัญสูง และจำนวนการละเมิดตัวเลือกความสำคัญต่ำที่มีค่าน้อยที่สุด โดยวิธีการ GISMOO เป็นลูกผสมระหว่าง GA และระบบภูมิคุ้มกันเทียมมาใช้ในการแก้ปัญหา สรุปว่าอัลกอริทึม GISMOO เป็นวิธีการมีประสิทธิภาพในการแก้ปัญหา เนื่องจากวิธีการนี้แก้ปัญหาช่องว่างระหว่างทฤษฎีกับแนวทางการปฏิบัติในปัญหาที่เกิดขึ้นจริงในอุตสาหกรรมรถยนต์ เพราะฉะนั้นในการพัฒนาอัลกอริทึม ควรมีการพัฒนาประสิทธิภาพทั้งในการหาคำตอบและการนำไปใช้งานจริงควบคู่ไปกับเวลาที่ใช้หาคำตอบ นพพล [10] ได้นำเสนออัลกอริทึมการบรรจวบ (Combinatorial Optimization with Coincidence: COIN) ในการแก้ปัญหาการจัดลำดับการผลิตภัณฑ์ที่มีหลายวัตถุประสงค์บนสายการประกอบแบบตัวยู โดยวัตถุประสงค์ที่ใช้ คือ ค่าใช้จ่ายการปรับตั้งเครื่องจักรและความผันแปรของภาระงานในระบบการผลิตน้อยที่สุด ผลจากงานวิจัยพบว่า COIN จะมีประสิทธิภาพก็ต่อเมื่อได้มีการรวมกับเมมเมติกอัลกอริทึมแบบ NSGA-II สามารถนำมาใช้ในการแก้ปัญหาการจัดลำดับผลิตภัณฑ์บนสายการประกอบลักษณะตัวยูได้มีประสิทธิภาพสำหรับปัญหาขนาดใหญ่ จากงานวิจัยที่กล่าวมา พบว่า COIN มีประสิทธิภาพในการค้นหาคำตอบที่รวดเร็ว เนื่องจากมีการจดจำลักษณะตำแหน่งของคู่ผลิตภัณฑ์ที่อยู่ติดกันและมีการจดจำลักษณะตำแหน่งของผลิตภัณฑ์ที่ดีและไม่ดี

ในปัญหาการจัดลำดับการผลิตรถยนต์จะมีสิ่งที่มีผลกระทบต่อการหาคำตอบการจัดลำดับการผลิตรถยนต์ก็คือ ตัวเลือก สี และสายการประกอบ ดังนั้นผู้วิจัยจึงนำเสนออัลกอริทึมการบรรจวบแบบขยาย (Combinatorial Optimization with Coincidence Expand: COIN-E) ซึ่งเป็นอัลกอริทึมที่ประยุกต์มาจาก COIN มาใช้ในการแก้ปัญหาการจัดลำดับการผลิตรถยนต์ที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน และทำการเปรียบเทียบประสิทธิภาพในการค้นหาคำตอบของ COIN-E กับอัลกอริทึมที่ได้รับการยอมรับว่าสามารถแก้ปัญหาประเภทนี้ได้ ได้แก่ NSGA-II, DPSO, BBO และ COIN [6],]10], [15]

2. สายการประกอบแบบสองด้าน

สายการประกอบแบบสองด้าน (Two–Sided Assembly Line) เป็นสายการประกอบที่มีลักษณะเป็นเส้นตรง ในการจัดสรรขั้นงานลงในสถานีงานบนสายการประกอบแบบสองด้าน มีข้อจำกัดในการจัดสรรขั้นงาน 2 แบบ คือ ขั้นงานที่สามารถจัดสรรลงในสถานีงานได้เพียงด้านเดียวเท่านั้น คือด้านซ้าย (L-type) หรือด้านขวา (R-type) ของสายการประกอบ และขั้นงานที่

สามารถจัดสรรลงในสถานีงานด้านใดด้านหนึ่งของสายการประกอบก็ได้ (E-type) สถานีงานที่อยู่คู่กัน เรียกว่า Mated-station ในรูปที่ 1 Mated-station ที่ 1 คือ สถานีงานที่ 1 กับสถานีงานที่ 2 และ Mated-station ที่ 2 คือ สถานีงานที่ 3 กับสถานีงานที่ 4 [11]

สายการประกอบแบบสองด้านจะเป็นสายการประกอบที่เหมาะสำหรับการผลิตรถยนต์เนื่องจากให้การประกอบชิ้นส่วนรถยนต์บางขั้นงานจะมีข้อจำกัดที่ต้องประกอบชิ้นส่วนทางด้านซ้ายหรือด้านขวาเพียงอย่างเดียว เช่น ขั้นงานของการประกอบพวงมาลัยรถยนต์ในประเทศไทย ก็จะมีการประกอบทางด้านขวาด้านเดียว จึงต้องจัดขั้นงานนี้ในสถานีงานทางด้านขวา โดยสายการประกอบนี้ ทำให้ความยาวของสายการประกอบสั้นลงเป็นผลให้ช่วยลดระยะในการเคลื่อนที่ของพนักงาน ลดเวลาในการปรับตั้งเครื่องจักรและลดเวลาในการผลิตให้สั้นลง และเครื่องมืออุปกรณ์ในสถานีงานประเภทนี้สามารถใช้ร่วมกันได้จึงทำให้ต้นทุนในการผลิตลดลง [12]

3. ปัญหาการจัดลำดับการผลิตรถยนต์

ในการแข่งขันการจัดลำดับรถยนต์ในปี 2005 ของบริษัทผู้ผลิตรถยนต์ในประเทศฝรั่งเศส ได้กำหนดปัญหาการจัดลำดับการผลิตที่สำคัญที่จะใช้ในการพิจารณาคือ ประการแรกคือ ตัวเลือกของรถยนต์ โดยในรถยนต์แต่ละคันจะมีตัวเลือกที่ต่างกัน เช่น ถุงลมนิรภัยรถคันหนึ่งจะมีตัวเลือกนี้ แต่อีกคันจะไม่มีตัวเลือกนี้ โดยในการจัดลำดับการผลิตรถยนต์นั้นจึงจำเป็นต้องพิจารณาตัวเลือกของรถยนต์พยายามไม่ให้ผลิตเกินค่าที่กำหนด โดยพิจารณาจากค่า อัตราส่วน p/q (เมื่อ p คือ ความต้องการตัวเลือก และ q คือ จำนวนลำดับการผลิตย่อย) เช่น ถ้าอัตราส่วนเท่ากับ 1 ต่อ 3 จะหมายความว่า ในการผลิตรถยนต์ 3 คันจะมีรถยนต์ที่มีตัวเลือกนี้ได้ไม่เกิน 1 คัน ถ้ามีการเกินจะนับจำนวนที่เกินเป็นจำนวนการละเมิดที่เกินขึ้น โดยจะพิจารณาไปเรื่อยๆ จนครบลำดับการผลิตรถยนต์ เพื่อทำให้การจัดลำดับการผลิตรถยนต์เกิดการละเมิดตัวเลือกรถยนต์นี้น้อยที่สุด โดยในการจัดลำดับการผลิตรถยนต์ในแต่ละวันจำเป็นที่จะต้องมีการนำข้อมูลลำดับการผลิตจากวันก่อนมาพิจารณาและจำเป็นต้องมีการวางแผนลำดับการผลิตไว้ในวันถัดไป เพื่อนำมาใช้ในการจัดลำดับการผลิตของวันปัจจุบัน เพื่อจะทำให้ลำดับการผลิตนั้นมีประสิทธิภาพตรงตามวัตถุประสงค์ ดังตารางที่ 1 ประการที่สองก็คือ เรื่องสีของรถยนต์เนื่องจากให้การพ่นสีให้รถยนต์แต่ละคันนั้น ลำดับการผลิตรถยนต์จะมีผลอย่างมากให้การพ่นสีเพราะ ถ้าลำดับการผลิตรถยนต์มีสีเดียวกันที่ติดต่อกันก็จะทำให้ไม่ต้องมีการทำความสะอาดหัวปืนพ่นสีและเปลี่ยนน้ำยาในการพ่น ทำให้เป็นการลดต้นทุนให้การผลิต แต่ถ้าพ่นสีที่เป็นสีเดียวกันติดต่อกันหลายลำดับนั้นจำเป็นที่จะต้องมีการล้างทำความสะอาดหัวปืนพ่นสี เนื่องจากถ้าไม่ทำความสะอาดหัวปืนพ่นสีจะมีการอุดตันและสีที่พ่นออกมาก็จะไม่มีคุณภาพในการผลิต ดังนั้นจึงต้องมีการกำหนดสีเดียวกันที่สามารถพ่นสีติดต่อกันได้ไม่เกินต่อคัน

ตารางที่ 1 อัตราส่วนความต้องการตัวเลือกของรถยนต์

ตัวเลือก	อัตราส่วนข้อกำหนด (p/q)	รุ่นของรถยนต์		
		A	B	C
Option#1	1/ 2	✓	✓	
Option#2	1/ 4		✓	✓
Option#3	2/ 3	✓		✓
Option#4	1/ 3	✓		✓

ตัวอย่างการจัดลำดับการผลิตรถยนต์ A, B และ C ลำดับการผลิตของวันก่อนหน้าคือ C-A-C สีที่พ่นในลำดับสุดท้ายของวันก่อนหน้าคือ สีที่ 1 และลำดับผลิตของวันถัดไปคือ B-A-C โดยกำหนดให้มีสัดส่วนผลิตภัณฑ์ (MPS) = 1:2:3 แสดงว่าในการจัดลำดับการผลิตนี้จะมีรถยนต์ A จำนวน 1 คัน รถยนต์ B จำนวน 2 คัน และรถยนต์ C จำนวน 3 คัน เข้าไปในสายการประกอบ โดยมีการจัดลำดับการผลิต (Model Sequence) เป็น ABCBCC โดยจะเริ่มจากการผลิตรถยนต์ A ก่อน ต่อรถยนต์ B และ C ไปเรื่อยๆ จนครบลำดับการผลิต

Option#1 ที่มีข้อกำหนดอัตราส่วนการผลิต 1/2 ของแต่ละลำดับย่อย จากลำดับการผลิตคือ ABCBCC โดยจะนับจำนวนการละเมิดทุกๆ 2 ลำดับ โดยจะนับจำนวนการละเมิดก็ต่อเมื่อในลำดับย่อยมีจำนวนตัวเลือกที่เกินจำนวนที่ต้องการ แต่ถ้าลำดับของตัวเลือกได้ถูกนับการละเมิดไปแล้วในลำดับย่อยที่ผ่านๆ มาแล้วจะไม่นับเป็นการละเมิด เนื่องจากจะเกิดปัญหาของการนับซ้ำ ซึ่งการนับแบบนี้จะป้องกันปัญหาการนับซ้ำที่เกิดในแต่ละรอบของการนับจำนวนการละเมิดในแต่ละลำดับย่อย ตัวอย่างการนับการละเมิด เริ่มที่ลำดับการผลิตที่ -1 และ 1 เนื่องจากการนับจำนวนการละเมิดจะต้องพิจารณาวันก่อนหน้านี้โดยจะพิจารณาลำดับย่อยนั้นจะขึ้นอยู่กับจำนวนลำดับย่อยของตัวเลือกนั้นซึ่งเท่ากับ 2 (q=2) โดยมีรถยนต์ที่ต้องผลิตคือ C และ A ซึ่งจะเห็นว่ารถยนต์ C ไม่ต้องทำการประกอบตัวเลือกนี้ แต่รถยนต์ A ต้องทำการประกอบตัวเลือกนี้ จึงทำให้ลำดับการผลิตที่ -1 และ 1 ไม่ผลิตเกินข้อกำหนดอัตราส่วนการผลิต 1 ต่อ 2 จึงไม่นับเป็นจำนวนการละเมิด พิจารณาแบบนี้ไปเรื่อยๆ ไปจนครบลำดับการผลิตทั้งหมด และจะพบว่าจะมีจำนวนการละเมิดรวมของทุกตัวเลือกทั้งหมดเท่ากับ 11 จำนวน

ส่วนเรื่องของสีที่พ่นให้กับรถยนต์นั้นจะพิจารณาดูว่ามีการเปลี่ยนแปลงสีไปทั้งหมดกี่ครั้ง ตัวอย่างเช่น ลำดับการผลิตคือ ABCBCC มาจัดวางลำดับโดยที่จะนำเอาลำดับสุดท้ายของวันที่ผ่านมามาวางในตำแหน่งที่ -1 เพื่อเป็นการพิจารณาว่าจะต้องมีการทำความสะอาดหัวปืนพ่นสีก่อนหรือไม่ โดยเมื่อมีการทำความสะอาดหัวปืนพ่นสีจะนับการเปลี่ยนแปลงสีเป็นหนึ่งครั้งแต่ถ้ามีการพ่นสีเดียวกันติดต่อกันเกิน 2 คันติดต่อกันก็จะต้องมีการทำความสะอาดหัวปืนพ่นสี จึงจำเป็นต้องนับเป็นหนึ่งครั้ง เนื่องจากถ้าไม่ทำความสะอาดหัวปืนพ่นสีจะทำให้ได้สีที่ไม่มีคุณภาพ โดยพิจารณาแบบนี้ไปเรื่อยๆ จนครบลำดับการผลิตของวันนั้น ซึ่งจะได้จำนวนการเปลี่ยนแปลงสีดังตารางที่ 2

ตารางที่ 2 จำนวนการเปลี่ยนแปลงของลำดับ ABCBCC

ลำดับ	-1	1	2	3	4	5	6
รถยนต์	C	A	B	C	B	C	C
สี	1	3	1	4	2	1	2
จำนวนการเปลี่ยนแปลงสี	-	1	1	1	1	1	1
รวม							6

4. การหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

ในการพิจารณาแก้ปัญหาที่มีหลายวัตถุประสงค์ให้มีประสิทธิภาพดีไปพร้อมกัน เป็นเรื่องที่แทบจะเป็นไปไม่ได้เพราะจะไม่มีคำตอบที่ดีที่สุด[13], [14] จึงมีการนำเทคนิคในการค้นหาเซตคำตอบที่อยู่ในพื้นที่ของคำตอบที่เป็นไปได้ทั้งหมด (Feasible Region) มาแก้ปัญหา ในงานวิจัยนี้เป็นการหาค่าเหมาะสมที่สุดเพื่อหาค่าที่น้อยที่สุด ดังสมการที่ 5

$$Minimize\{f_1(\bar{x}), f_2(\bar{x}), ..., f_k(\bar{x})\} \quad (5)$$

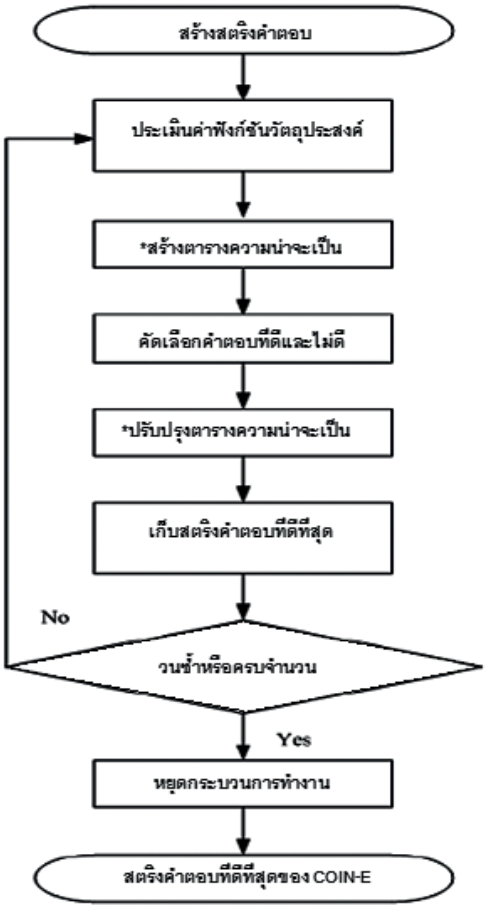
โดย

$\bar{x}$ คือเวกเตอร์ของตัวแปรตัดสินใจ

$f_i(\bar{x})$ คือฟังก์ชันวัตถุประสงค์ที่

ในงานวิจัยนี้พิจารณา 3 ฟังก์ชันวัตถุประสงค์ดังนี้ 1) ปริมาณงานที่ทำไม่เสร็จที่น้อยที่สุด (Minimize Utility Work)[11] 2) จำนวนรถยนต์ที่ละเมิดรวมน้อยที่สุด (Minimize the Number of Violations) 3) จำนวนครั้งการเปลี่ยนแปลงสีน้อยที่สุด (Minimize the Number of Color Changes)

โดยได้นำเทคนิคการกำหนดค่าความแข็งแรงด้วยวิธีเชิงกลุ่มที่ดีที่สุด (Pareto-based Approach) ที่มีการจัดอันดับคำตอบแบบพาเรโต (Pareto Ranking Approach) มาใช้ คำตอบที่ได้จะอยู่ในรูปของกลุ่มคำตอบที่ดีที่สุด (Pareto Optimal Set หรือ Pareto Optimal Front) ถ้าเวกเตอร์ของตัวแปรตัดสินใจ x ให้คำตอบที่ดีกว่าเวกเตอร์ของตัวแปรตัดสินใจ y แล้ว จะได้ว่า $f_i(x) \le f_i(y)$ สำหรับทุกค่า $i \in \{1,2,...,k\}$ และ $f_i(x) \le f_i(y)$ มีอย่างน้อย 1 ค่าของ $i \in \{1,2,...,k\}$



รูปที่ 1 ขั้นตอนการทำงานอัลกอริทึม COIN-E

5. อัลกอริทึม COIN-E

COIN มีแนวคิดหลักนั้นเป็นการศึกษาแนวทางของคำตอบที่ดี (Good) และคำตอบที่ไม่ดี (Not Good) ที่เกิดขึ้นในเวลาเดียวกันเพื่อเป็นตัวกำหนดทิศทางของคำตอบสุดท้าย โดยมีการสร้างตารางความน่าจะเป็นขึ้นมาแล้วสุ่มเลือกมาสร้างประชากรเริ่มต้น โดยมีการปรับปรุงตารางความน่าจะเป็นตลอดเวลาในแต่ละรอบ โดยปรับปรุงตามคำตอบที่ดีและคำตอบที่ไม่ดี ถ้าเกิดเป็นคำตอบที่ดีค่าความน่าจะเป็นก็จะมากขึ้นเพื่อใช้ในการสุ่มรอบถัดไป แต่ถ้าคำตอบไม่ดีค่าความน่าจะเป็นก็จะถูกลดลงเพื่อจะได้สุ่มเจอน้อยลงในรอบถัดไป [10] โดยมีขั้นตอนการทำงานดังรูปที่ 1

งานวิจัยนี้ได้นำแนวคิดของ COIN มาประยุกต์ในการหาคำตอบที่เหมาะสมเป็นอัลกอริทึม COIN-E โดยในขั้นตอนการสร้างตารางความน่าจะเป็นร่วมจะแตกต่างกันจากเดิมวิธีการ COIN จะใช้ตารางความน่าจะเป็นร่วมเพียงตารางเดียวในการเลือกเดินเพื่อนำไปสู่คำตอบที่เหมาะสม แต่ของ COIN-E จะเพิ่มตารางความน่าจะเป็นร่วมเพิ่มมาอีก 3 ตารางที่ได้มาจากการหาคำตอบของแต่ละวัตถุประสงค์ โดยนำคำตอบของแต่ละวัตถุประสงค์ที่ดีและไม่ดีมากระทำในขั้นตอนการสร้างตารางความน่าจะเป็น ดังนั้น COIN-E จะมีตารางความน่าจะเป็นร่วมรวมทั้งหมด 4 ตารางที่เกิดจากการขั้นตอนการปรับปรุงตารางความน่าจะเป็นคือ 1) ตารางวิธีการแบบ COIN 2) ตารางจากวัตถุประสงค์แรก 3) ตารางจากวัตถุประสงค์สอง 4) ตารางจากวัตถุประสงค์สาม เพื่อจะทำให้ได้คำตอบที่เหมาะสมและคำตอบที่ได้ก็จะมีการกระจายตัวของคำตอบมากกว่าวิธีการ COIN แบบเดิม ซึ่ง COIN-E จะแบ่งการสุ่มในแต่ละรอบคือจากตาราง COIN มาร้อยละ 70 ของประชากรทั้งหมด และสุ่มจากตารางคำตอบของแต่ละวัตถุประสงค์รวมกันร้อยละ 30 ของประชากรทั้งหมด โดยเป็นร้อยละที่เหมาะสมที่ได้จากการทำการทดลองโดยใช้วิธี Pilot Run ในการหาร้อยละความเหมาะสมที่สุดที่จะทำให้อัลกอริทึมของ COIN-E มีประสิทธิภาพสูงสุด

ขั้นตอนของการทำงาน COIN-E ที่ใช้ในงานวิจัยมีดังนี้

ขั้นที่ 1 สร้างสตริงคำตอบ

การสร้างสตริงคำตอบเบื้องต้นในรอบแรกจะใช้วิธีการสุ่มด้วยความน่าจะเป็นที่เท่ากัน แต่ในรอบถัดไปจะสุ่มจากตารางความน่าจะเป็นร่วมจากตาราง COIN มาร้อยละ 70 ของประชากรทั้งหมด และอีกร้อยละ 30 จะมาจากตารางความน่าจะเป็นร่วมของแต่ละวัตถุประสงค์ ปัญหาตัวอย่างได้กำหนดค่า MPS = 1:2:3 แสดงว่าในการจัดลำดับการผลิตนี้จะมีรถยนต์ A จำนวน 1 คัน รถยนต์ B จำนวน 2 คัน และรถยนต์ C จำนวน 3 คัน จะมีความยาวของลำดับการผลิตเท่ากับ 6 จากนั้นก็ทำการให้รหัสประจำตัวของรถแต่ละคันและกำหนดสีให้รถยนต์แต่ละคัน

ดังตารางที่ 3 โดยลำดับการผลิตของวันก่อนหน้าคือ C-A-C สีที่พ่นในลำดับสุดท้ายของวัดก่อนหน้าคือ สีที่ 1 และลำดับการผลิตของวันถัดไปคือ B-A-C แล้วทำการสุ่มสตริงคำตอบเริ่มต้นมาจำนวน 5 สตริงคำตอบดังตารางที่ 4 แล้วทำการคัดลอกสตริงคำตอบไว้ในสตริงคำตอบชั่วคราว

ตารางที่ 3 รหัสงานผลิตภัณฑ์

Model Sequence	A	B	B	C	C	C
String	1	2	3	4	5	6
สี	3	1	2	4	1	2

ตารางที่ 4 สตริงคำตอบเริ่มต้นที่ได้จากการสุ่ม

Strings	**Model Sequences**					
1	1	3	4	2	6	5
2	2	5	4	1	3	6
3	1	6	5	3	2	4
4	5	1	3	2	4	6
5	3	2	6	1	4	5

ขั้นที่ 2 ประเมินค่าฟังก์ชันวัตถุประสงค์

การนำเอาสตริงคำตอบที่ได้จากการสุ่มมาหาค่าฟังก์ชันวัตถุประสงค์ กำหนดค่าความแข็งแรงของสตริงคำตอบด้วยวิธี Non-Dominated Sorting และกำหนดค่าความหนาแน่นด้วยวิธี Crowding Distance [16] ได้ค่าดังตารางที่ 5

ตารางที่ 5 ค่าความหนาแน่นของสตริงคำตอบ

Strings	**Utility Work**	**The Number of Violations**	**Color Changes**	**Fitness**	**Crowding Distance**
1	23.6596	11	6	2	Infinity
2	23.6883	11	4	1	3.0000
3	23.3394	12	6	1	Infinity
4	23.8388	12	5	2	Infinity
5	23.3681	11	6	1	3.0000

ขั้นที่ 3 สร้างตารางความน่าจะเป็น

การสร้างตารางความน่าจะเป็นร่วมจะสร้างขึ้นมาโดยพิจารณาจากรถยนต์ที่ต้องผลิต ซึ่งในที่นี้ MPS เท่ากับ 1:2:3 หรือความยาวเท่ากับ 6 หน่วย ดังนั้นขนาดของตารางเมทริกซ์ความน่าจะเป็นร่วมคือ 6×6 โดยการกำหนดค่าความน่าจะเป็นในตารางเมทริกซ์เริ่มแรก โดยที่การเริ่มต้นค่าความน่าจะเป็นในการถูกสุ่มเลือกเท่ากันทั้งหมดซึ่งมีค่าเท่ากับ $n = \frac{1}{6-1} = \frac{1}{6_{-1}} = 0.2$ เมื่อ n คือจำนวนผลิตภัณฑ์ทั้งหมด โดยจะมีทั้งหมด 4 ตาราง คือ 1) ตารางวิธีการแบบ COIN คือ พิจารณาจากค่า Fitness ในการปรับปรุงตารางความน่าจะเป็นร่วม 2) ตารางจากวัตถุประสงค์แรกคือ การนำคำตอบของวัตถุประสงค์แรกมาเรียงจากน้อยไปหามากที่สุด 3) ตารางจากวัตถุประสงค์ที่สองคือ การนำคำตอบของวัตถุประสงค์ที่สองมาเรียงจากน้อยไปหามากที่สุด 4) ตารางจากวัตถุประสงค์ที่สามคือ การนำคำตอบของวัตถุประสงค์ที่สามมาเรียงจากน้อยไปหามากที่สุด

ขั้นที่ 4 คัดเลือกสตริงคำตอบที่ดีและไม่ดี

สตริงคำตอบที่ดีจะทำการเพิ่มค่าความน่าจะเป็นร่วมในทางตรงกันข้ามสตริงคำตอบที่ไม่ดีจะทำการลดค่าความน่าจะเป็นร่วม

ขั้นที่ 5 ปรับปรุงตารางค่าความน่าจะเป็น

การปรับปรุงความน่าจะเป็นในตารางความน่าจะเป็นร่วมเป็นขั้นตอนที่สำคัญที่สุด เพื่อให้รางวัล (Reward) หรือเพิ่มโอกาสค่าความน่าจะเป็นในการสุ่มเลือกผลิตภัณฑ์ในรอบถัดไป ในทางตรงกันข้ามถ้าลงโทษ (Punish) หรือลดค่าความน่าจะเป็นในการสุ่มเลือกผลิตภัณฑ์ในรอบถัดไป

ปรับปรุงตารางความน่าจะเป็นร่วมแบบ COIN

เริ่มต้นจะพิจารณาปรับปรุงจากตารางแรกคือ ตารางค่าความน่าจะเป็นร่วมโดยพิจารณาจากค่า Fitness จากตารางที่ 5 จะพบว่ามีสตริงคำตอบที่ดีหรือมีค่า Fitness น้อยที่สุดอยู่ 3 สตริงคำตอบ จึงทำการให้รางวัลหรือเพิ่มโอกาสค่าความน่าจะเป็น ในทางตรงกันข้ามจะพบว่ามีสตริงคำตอบที่ไม่ดีหรือมีค่า Fitness มากที่สุดอยู่ 2 สตริงคำตอบจึงทำการลงโทษหรือลงโอกาสในการสุ่มเจอในรอบถัดไป

กรณีสตริงคำตอบที่มีคำตอบที่ดีจะพิจารณาปรับปรุงไปทีละสตริงคำตอบ โดยเริ่มที่สตริงคำตอบที่ 2 คือ 2, 5, 4, 1, 3, 6 ทำการปรับปรุงตารางความน่าจะเป็นร่วม ซึ่งรถยนต์อันดับแรกที่ถูกเลือกลงในสตริงคำตอบที่ คือ รหัส 2 (รถยนต์รุ่น B) และรถยนต์ลำดับที่สองที่ถูกเลือกลงในสตริงคำตอบที่ 2 คือ รหัส 5 (รถยนต์รุ่น C) ทำการให้รางวัลหรือเพิ่มค่าความน่าจะเป็นร่วมที่ตำแหน่ง (รถยนต์คนแรกที่สุ่ม,รถยนต์คันที่สองที่สุ่มเจอ) คือคู่ลำดับ (2,5) ในตารางเมทริกซ์ความน่าจะเป็นร่วม โดยเพิ่มค่าความน่าจะเป็นเท่ากับ โดยเพิ่มค่าความน่าจะเป็นเท่ากับ $x=\frac{k}{n-1}=\frac{0.1}{6-1}=0.02$ (เมื่อ k คือค่าพารามิเตอร์ในการให้รางวัลที่กำหนดไว้เบื้องต้น) และลดค่าความน่าจะเป็นของคู่ลำดับทั้งหมด (รวมทั้งคู่ลำดับ (2, 5)) ในแถวเดียวกัน คือ (2, 1), (2, 3)… (2, 6) โดยลดค่าความน่าจะเป็น $x=\frac{k}{(n-1)^2}=\frac{0.1}{(6-1)^2}=0.004$ ซึ่งค่าดังกล่าวจะมีค่าเท่ากับค่าความน่าจะเป็นที่จะเพิ่มให้คู่ลำดับ (2, 5) นั้นเอง

ต่อจากนั้นผลิตภัณฑ์อันดับสามที่ถูกเลือกลงสตริงคำตอบที่ดีตัวแรกต่อจากรหัส 5 (รถยนต์รุ่น C) คือ รหัส 4 (รถยนต์รุ่น C) ดังนั้นจะทำการให้รางวัล (Reward) หรือเพิ่มค่าความน่าจะเป็นในตารางเมทริกซ์ความน่าจะเป็นร่วมที่ตำแหน่ง พิจารณาแบบนี้ไปจนครบสตริงคำตอบที่ดี จะได้ตารางความน่าจะเป็นร่วมที่ได้ปรับปรุง

กรณีสตริงคำตอบที่มีคำตอบที่ไม่ดีจะพิจารณาปรับปรุงไปทีละสตริงคำตอบ โดยเริ่มที่สตริงคำตอบที่ 1 คือ 1, 3, 4, 2, 6, 5 ทำการปรับปรุงตารางความน่าจะเป็นร่วม ซึ่งรถยนต์อันดับแรกที่ถูกเลือกลงในสตริงคำตอบที่ 1 คือ รหัส 1 (รถยนต์รุ่น A) และรถยนต์ลำดับที่สองที่ถูกเลือกลงในสตริงคำตอบที่ 1 คือ รหัส 3 (รถยนต์รุ่น B) ทำการลงโทษหรือลดค่าความน่าจะเป็นร่วมที่ตำแหน่ง (รถยนต์คนแรกที่สุ่ม,รถยนต์คันที่สองที่สุ่มเจอ) คือคู่ลำดับ (1,3) ในตารางเมทริกซ์ความน่าจะเป็นร่วมได้โดยลดค่าความน่าจะเป็นเท่ากับ $x=\frac{k}{n-1}=\frac{0.1}{6-1}=0.02$ และเพิ่มค่าความน่าจะเป็นของคู่ลำดับทั้งหมด (รวมทั้งคู่ลำดับ (1, 3)) ในแถวเดียวกันคือ (1, 2), (1, 3) … (1, 6) โดยเพิ่มค่าความน่าจะเป็นคู่ลำดับเท่ากับค่าความน่าจะเป็นที่ลดจากคู่ลำดับเท่ากับ $x=\frac{k}{(n-1)^2}=\frac{0.1}{(6-1)^2}=0.004$ ซึ่งค่าดังกล่าวจะมีค่าเท่ากับค่าความน่าจะเป็นที่ลดจากคู่ลำดับ (1, 3) นั้นเอง โดยค่าความน่าจะเป็นของคู่ลำดับต่างๆ อ้างอิงจากตารางเมทริกซ์ความน่าจะเป็นร่วมหลังการปรับปรุงจากสตริงคำตอบที่มีคำตอบที่ดี ซึ่งพิจารณาแบบนี้ไปเรื่อยๆ จนครบสตริงคำตอบที่ไม่ดี ก็จะได้ตารางค่าความน่าจะเป็นร่วมที่ได้ผ่านการให้รางวัลและลงโทษสตริงคำตอบที่ดีและไม่ดี ได้ผลดังตารางที่ 6

ตารางที่ 6 ตารางเมทริกซ์ความน่าจะเป็นร่วมหลังจากการปรับปรุงจากตารางความน่าจะเป็นแบบ COIN

From/To	1	2	3	4	5	6
1	0	0.196	0.176	0.216	0.196	0.216
2	0.196	0	0.196	0.196	0.216	0.196
3	0.196	0.216	0	0.176	0.196	0.216
4	0.220	0.180	0.200	0	0.220	0.180
5	0.176	0.196	0.216	0.216	0	0.196
6	0.216	0.196	0.196	0.196	0.196	0

ปรับปรุงตารางของแต่ละวัตถุประสงค์

ตารางวัตถุประสงค์ที่ 1 คือ วัตถุประสงค์ของปริมาณงานที่ทำไม่เสร็จ โดยนำค่าวัตถุประสงค์ที่ 1 มาเรียงจากค่าน้อยที่สุดไปยังมากที่สุด จากตารางที่ 5 จะเรียงลำดับสตริงคำตอบที่มีค่าน้อยที่สุดไปมากที่สุดคือ สตริงคำตอบที่ 3, 5, 1, 2 และ 4 มีค่าเท่ากับ 23.3394, 23.3681, 23.6596, 23.6883 และ 23.8388 ตามลำดับ จะเห็นว่าสตริงคำตอบที่อยู่ในกลุ่มคำตอบที่ดีคือ สตริงคำตอบที่ 3 โดยมีลำดับการผลิตเท่ากับ 1, 6, 5, 3, 2, 4 จึงนำสตริงคำตอบนี้ไปทำการให้รางวัลหรือเพิ่มโอกาสค่าความน่าจะเป็น ในทางตรงกันข้ามสตริงคำตอบที่มีกลุ่มคำตอบที่ไม่ดีคือ สตริงคำตอบที่ 4 โดยมีลำดับการผลิตเท่ากับ 5, 1, 3, 2, 4, 6 ไปทำการลงโทษหรือลดค่าความน่าจะเป็น โดยการคำนวณปรับปรุงค่าตารางความน่าจะเป็นจะมีวิธีการคำนวณเหมือนตารางความน่าจะเป็นร่วมจากค่า Fitness ที่ผ่านมา หลังจากปรับปรุงตารางความน่าจะเป็นของกลุ่มคำตอบที่ดีและไม่ดีจากวัตถุประสงค์ที่ 1 ได้ค่าดังตารางที่ 7 ในทำนอง

เดียวกันในอีกสองวัตถุประสงค์คือ ตารางวัตถุประสงค์ที่ 2 คือ วัตถุประสงค์ของจำนวนรถยนต์ที่ละเมิดรวมของตัวเลือก โดยนำค่าวัตถุประสงค์ที่ 2 มาเรียงจากค่าน้อยที่สุดไปยังมากที่สุด และตารางวัตถุประสงค์ที่ 3 คือ วัตถุประสงค์ของจำนวนการเปลี่ยนแปลงสี โดยนำค่าวัตถุประสงค์ที่ 3 มาเรียงจากค่าน้อยที่สุดไปยังมากที่สุด หลังจากปรับปรุงตารางความน่าจะเป็นของกลุ่มคำตอบที่ดีและไม่ดีจากวัตถุประสงค์ที่ 2 และ 3 ได้ค่าดังตารางที่ 8 และ 9

ตารางที่ 7 ตารางเมทริกซ์ความน่าจะเป็นร่วมหลังจากการปรับปรุงจากตารางความน่าจะเป็นของวัตถุประสงค์ที่ 1

From/To	1	2	3	4	5	6
1	0	0.200	0.180	0.200	0.200	0.220
2	0.200	0	0.200	0.200	0.200	0.200
3	0.200	0.200	0	0.200	0.200	0.200
4	0.204	0.204	0.204	0	0.204	0.184
5	0.180	0.200	0.220	0.200	0	0.200
6	0.196	0.196	0.196	0.196	0.216	0

ตารางที่ 8 ตารางเมทริกซ์ความน่าจะเป็นร่วมหลังจากการปรับปรุงจากตารางความน่าจะเป็นของวัตถุประสงค์ที่ 2

From/To	1	2	3	4	5	6
1	0	0.196	0.216	0.216	0.196	0.176
2	0.196	0	0.196	0.156	0.216	0.236
3	0.196	0.176	0	0.216	0.196	0.216
4	0.212	0.212	0.192	0	0.212	0.172
5	0.184	0.204	0.184	0.224	0	0.204
6	0.216	0.196	0.196	0.196	0.196	0

ตารางที่ 9 ตารางเมทริกซ์ความน่าจะเป็นร่วมหลังจากการปรับปรุงจากตารางความน่าจะเป็นของวัตถุประสงค์ที่ 3

From/To	1	2	3	4	5	6
1	0	0.208	0.208	0.188	0.208	0.188
2	0.208	0	0.208	0.188	0.228	0.168
3	0.208	0.168	0	0.188	0.208	0.228
4	0.224	0.184	0.204	0	0.184	0.204
5	0.200	0.200	0.180	0.220	0	0.200
6	0.192	0.212	0.212	0.212	0.172	0

ขั้นที่ 6 เก็บสตริงคำตอบที่ดีที่สุด

ใช้เทคนิคการเก็บค่าที่ดีที่สุดคัดเลือกกลุ่มสตริงคำตอบที่ดีที่สุดในรอบการทำงานปัจจุบัน และนำไปรวมกับกลุ่มสตริงคำตอบที่ดีที่สุดในรอบก่อนหน้า แล้วนำไปจัดอันดับสตริงคำตอบด้วยวิธี Non-dominated Sorting ทำการเก็บสตริงคำตอบที่มีค่าความแข็งแรงมากที่สุดเข้าสู่กลุ่มสตริงคำตอบที่ดีที่สุดแทนกลุ่มในรอบก่อนหน้า หลังจากนั้นสร้างประชากรในรอบถัดไป โดยเลือกจากตารางของ COIN จะสุ่มเลือกมาร้อยละ 70 ของประชากรทั้งหมด และจากตารางความน่าจะเป็นของแต่ละวัตถุประสงค์อีกร้อยละ 30 ของประชากรทั้งหมด

ขั้นที่ 7 การวนซ้ำและหยุดกระบวนการทำงาน

ทำการวนซ้ำตั้งแต่ขั้นตอนที่ 2 จนกระทั่งครบตามจำนวนรอบการทำงานที่กำหนด

6. การทดลอง

ในการทดลองจะใช้ปัญหามาตรฐานในการจัดลำดับการผลิตของ Mcmullen [17] และ Mansouri [18] มีรายละเอียดดังแสดงในตารางที่ 10 ทั้งหมด 11 ปัญหา โดยจะแบ่งเป็นปัญหาขนาดเล็ก ปัญหาขนาดกลาง และปัญหาขนาดใหญ่ มีความแตกต่างกันในเรื่องของขั้นงาน 16, 24, 65, 70, 148 และ 205 ขั้นงาน มีจำนวนผลิตภัณฑ์ 5, 10 และ 15 ผลิตภัณฑ์ มีตัวเลือกของรถยนต์ตั้งแต่ 2 จนถึง 10 ตัวเลือกรถยนต์ โดยอัตราส่วน p/q จะอ้างอิงจากงานวิจัยก่อนหน้า [5] ดังตารางที่ 11 มีสีเริ่มแรก ลำดับการผลิตก่อนหน้าและถัดไป ดังตารางที่ 12 มีจำนวนสี 4, 8, 10 และ 15 สี และมีความแตกต่างของความต้องการสัดส่วนผลิตภัณฑ์ที่แตกต่างกัน โดยงานวิจัยนี้มีการนำอัลกอริทึม NSGA-II, DPSO, BBO และ COIN มาเปรียบเทียบกับอัลกอริทึม COIN-E

6.1 ตัววัดสมรรถนะ

ในการเปรียบเทียบสมรรถนะของอัลกอริทึม COIN-E COIN BBO DPSO และ NSGA-II โดยจะใช้ตัวแปรตอบสนอง 4 ตัวในการวิเคราะห์ คือ การลู่เข้าสู่กลุ่มคำตอบ

ที่แท้จริง (Convergence to the Pareto Optimal Set) [19] การกระจายตัวของกลุ่มคำตอบที่ค้นพบ (Spread Measurement) [20] อัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบกับกลุ่มคำตอบที่แท้จริง (Raito of Non-dominate Solution) [20] และเวลาที่ใช้ในการคำนวณ (Computation Time to Solution)

6.2 การกำหนดค่าพารามิเตอร์

เนื่องจากปัญหาที่ใช้ในการวิจัยมีความหลากหลายและแตกต่างกัน 11 ปัญหา จึงต้องมีการทำการทดสอบค่าพารามิเตอร์ที่เหมาะสมสำหรับแต่ละปัญหาเพื่อทำให้ได้คำตอบที่มีประสิทธิภาพในการหาคำตอบ ซึ่งในแต่ละอัลกอริทึมมีการทำการทดสอบพารามิเตอร์โดยทำการออกแบบการทดลองแบบ Full Factorial Design ทำซ้ำ 2 Replicate ปัญหาการทดลอง Set 1-4 จำนวนรอบการทำงาน 1000 รอบ และปัญหาการทดลอง Set 5 และ Set 6 จำนวนรอบการทำงาน 2000 รอบ อัลกอริทึม NSGA-II กำหนดค่าพารามิเตอร์จากการอ้างอิงงานวิจัยที่ผ่านมา อัลกอริทึม DPSO มีการทำการทดสอบพารามิเตอร์ของจำนวนฝูงและจำนวนอนุภาคในฝูง แบบเป็น 3 ระดับ คือ ระดับที่ 1 (4-25) คือมีจำนวนฝูง 4 ฝูงและ 25 อนุภาคในฝูง ระดับที่ 2 (5-20) และระดับที่ 3 (10-10) อัลกอริทึม BBO มีการทำการทดสอบพารามิเตอร์ของรูปแบบการอพยพแบบเป็น 2 แบบ คือ Linear และ Sinusoidal และมีกำหนดจากค่าพารามิเตอร์จากการอ้างอิงงานวิจัยที่ผ่านมา อัลกอริทึม COIN มีการทำการทดสอบพารามิเตอร์ของความน่าจะเป็นที่ใช้ในการปรับปรุงตารางความน่าจะเป็นร่วม แบ่งเป็น 3 ระดับคือ ระดับที่ 1 ความน่าจะเป็นในการปรับปรุงเท่ากับ 0.010 ระดับที่ 2 เท่ากับ 0.125 และระดับที่ 3 เท่ากับ 0.150 อัลกอริทึม COIN-E จะมีพารามิเตอร์ของความน่าจะเป็นที่ใช้ในการปรับปรุงตารางความน่าจะเป็นโดยจะใช้การอ้างอิงจากอัลกอริทึม COIN และพารามิเตอร์ที่จะใช้สุ่มจำนวนประชากรจากตารางความน่าจะเป็นของแต่ละวัตถุประสงค์ แบ่งเป็น 3 ระดับคือ ระดับที่ 1 (50-25-25) คือ ในแต่ละสตริงคำตอบจะสุ่มจากตารางวัตถุประสงค์ที่ 1 มาร้อยละ 50 จากตารางวัตถุประสงค์ที่ 2 ร้อยละ 25 และจากตารางวัตถุประสงค์ที่ 3 ร้อยละ 25 จนครบจำนวนสตริงคำตอบทั้งสิ้นร้อยละ 30 ของประชากรทั้งหมด ระดับที่ 2 (25-50-25) และระดับที่ 3 (25-25-50) ซึ่งมีรายละเอียดดังแสดงในตารางที่ 13 โดยการรันโปรแกรม Matlab 2009a บนคอมพิวเตอร์ Intel(R) CoreTM i7-2670QM CPU@ 2.2GHz Ram 4 GB Window7

ตารางที่ 10 ปัญหาที่ใช้ในการวิจัย

ปัญหาการทดลอง		จำนวนรุ่นรถยนต์	สัดส่วนผลิตภัณฑ์ (MPS)	ความยาวสตริง	จำนวนสีรถยนต์ (ซ้ำไม่เกิน)	จำนวนตัวเลือก Options
Set 1	1	5	5:3:2:1:1	12	4(2)	2
	2	5	4:4:2:1:1	12	4(2)	2
Set 2	3	5	7:3:2:2:1	15	4(2)	4
	4	5	4:3:3:3:2	15	4(2)	4
Set 3	5	5	8:7:2:2:1	20	4(2)	7
	6	5	5:4:4:4:3	20	4(2)	7
Set 4	7	10	7:5:1:1:1:1:1:1:1:1	20	8(4)	8
	8	10	4:4:4:2:1:1:1:1:1:1	20	8(4)	8
Set 5	9	15	20:20:20:15:15:1:1:1:1:1:1:1:1:1:1	100	10(5)	10
	10	15	15:15:10:10:10:10:10:10:4:1:1:1:1:1:1	100	10(5)	10
Set 6		15	20:20:20:15:15:10:20:10:5:20:5:10:10:10:10	200	15(7)	10

ตารางที่ 11 ตัวเลือกรถยนต์และอัตราส่วนการผลิตในลำดับการผลิตย่อย

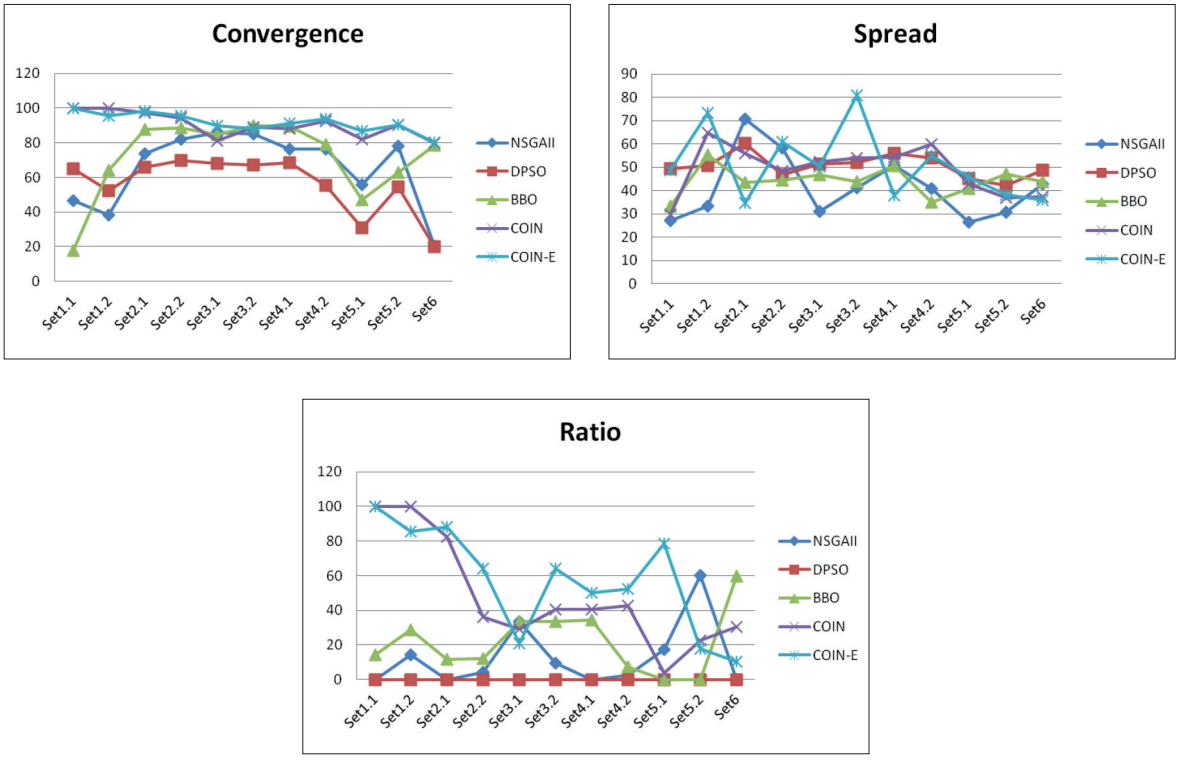
Option	p/q	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Option#1	2/3	✓			✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
Option#2	1/2			✓	✓		✓		✓	✓		✓	✓	✓	✓	
Option#3	17/30				✓	✓	✓		✓	✓		✓	✓	✓	✓	✓
Option#4	1/2				✓	✓	✓			✓		✓		✓	✓	✓
Option#5	3/5	✓			✓	✓	✓		✓	✓		✓	✓	✓	✓	✓
Option#6	39/43	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	
Option#7	10/15	✓			✓				✓	✓	✓	✓			✓	
Option#8	6/30								✓							
Option#9	2/15								✓	✓						
Option#10	4/9				✓		✓		✓	✓		✓	✓		✓	

ตารางที่ 12 สี ลำดับการผลิตก่อนหน้าและลำดับถัดไป

Set	สีเริ่มแรก	รหัสรถยนต์ที่ผลิตลำดับก่อนหน้า	รหัสรถยนต์ที่ผลิตลำดับถัดไป
1	1	4,1	1,5
2	2	2,1	4,1
3	1	1,5,1,2,4,3,4,2,2	4,3,1,3,4,3,2,5,5
4	5	3,1,10,7,4,7,8,4,7,3,10,8,2,5	10,6,7,7,10,2,5,4,7,9,5,9,4,6
5และ6	2	5,11,9,6,6,4,13,2,6,2,14,1,4,6,14,8,4,5,14,1,3,12,3,8, 5,14,13,15,6,8,12,6,14,7,1,3,6,1,13,14,1,4	4,3,4,15,15,15,5,1,12,2,2,14,6,7,1,12,10,1,7,3,3,8,14, 11,2,9,5,11,8,9,9,5,15,2,2,12,5,15,4,9,1,13

ตารางที่ 13 ค่าพารามิเตอร์ของอัลกอริทึม

Set	NSGA-II		DPSO	BBO		COIN	COIN-E
	P_c	P_m	Swarm (Particles)	Migration	Mutation	Prob	Prob(Ramdom)
Set 1.1	1.0	0.01	4(25)	Linear	0.01	0.150	0.150(25-50-25)
Set 1.2			10(10)	Sinusoidal		0.100	0.100(50-25-25)
Set 2.1			5(20)	Sinusoidal		0.150	0.150(50-25-25)
Set 2.2			10(10)	Sinusoidal		0.100	0.100(50-25-25)
Set 3.1			10(10)	Linear		0.150	0.150(25-25-50)
Set 3.2			5(20)	Sinusoidal		0.100	0.100(50-25-25)
Set 4.1			4(25)	Linear		0.125	0.125(25-25-50)
Set 4.2			4(25)	Linear		0.150	0.150(25-25-50)
Set 5.1			10(10)	Sinusoidal		0.125	0.125(25-50-25)
Set 5.2			10(10)	Linear		0.100	0.100(50-25-25)
Set 6			5(20)	Linear		0.125	0.125(25-50-25)



รูปที่ 2 ผลการทดลอง

6.3 ผลการทดลอง

จากผลการทดลองดังแสดงในตารางที่ 14 เมื่อพิจารณาโดยการนำมาเปรียบเทียบเป็นค่าเปอร์เซ็นต์ดังตารางที่ 15 และรูปที่ 2 ผลการทดลองตัวชี้วัดสมรรถนะด้านการลู่เข้าสู่กลุ่มคำตอบที่แท้จริง พบว่าส่วนใหญ่ของทุกปัญหา COIN-E นั้นมีค่าการลู่เข้าสู่กลุ่มคำตอบที่แท้จริงเข้าใกล้ 0 มากที่สุด 91.85 เปอร์เซ็นต์ ค่าเบี่ยงเบนมาตรฐานเท่ากับ 5.39 ส่วน DPSO เป็นวิธีการที่มีค่าการลู่เข้าสู่คำตอบที่แท้จริงน้อยที่สุดมีค่าเท่ากับ 56.11 เปอร์เซ็นต์ ค่าเบี่ยงเบนมาตรฐานเท่ากับ 15.85 ส่วนการพิจารณาตัวชี้วัดสมรรถนะด้านการกระจายของกลุ่มคำตอบที่ค้นพบ พบว่าทุกอัลกอริทึมมีการกระจายของกลุ่มคำตอบที่ไม่แตกต่างกัน โดย COIN-E มีการกระจายของกลุ่มคำตอบที่เข้าใกล้ 0 ที่มากที่สุดเท่ากับ 51.08 เปอร์เซ็นต์ ค่าเบี่ยงเบนมาตรฐานเท่ากับ 14.66 ส่วน NSGA-II มีการกระจายของกลุ่มคำตอบที่น้อยที่สุดเท่ากับ 41.17 เปอร์เซ็นต์ ค่าเบี่ยงเบนมาตรฐานเท่ากับ 13.28 ในการพิจารณาด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบเท่ากลุ่มคำตอบที่แท้จริง พบว่าวิธี COIN-E มีค่าอัตราส่วนเข้าใกล้ค่า 1 มากที่สุดเท่ากับ 57.48 เปอร์เซ็นต์ ค่าเบี่ยงเบนมาตรฐานเท่ากับ 29.09 ส่วน DPSO มีค่าอัตราส่วนน้อยที่สุดเท่ากับ 0 เปอร์เซ็นต์ ค่าเบี่ยงเบนมาตรฐานเท่ากับ 0 และเมื่อพิจารณาด้านเวลาในการค้นหาคำตอบ พบว่าวิธี DPSO ใช้เวลาในการค้นหาคำตอบน้อยที่สุด ส่วน NSGA-II นั้นจะใช้เวลาในการค้นหาคำตอบมากที่สุด จากการเปรียบเทียบตัวชี้วัดสมรรถนะของทั้ง 3 อัลกอริทึมแล้ว พบว่าวิธี COIN-E มีประสิทธิภาพด้านการลู่เข้าสู่กลุ่มคำตอบ ด้านการกระจายกลุ่มคำตอบ และด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบเท่ากับกลุ่มคำตอบที่แท้จริงดีกว่า NSGA-II, DPSO, BBO และ COIN

ตารางที่ 14 ผลการทดลอง

Efficiency	Algorithm	Problems										
		Set 1.1	**Set 1.2**	**Set 2.1**	**Set 2.2**	**Set 3.1**	**Set 3.2**	**Set 4.1**	**Set 4.2**	**Set 5.1**	**Set 5.2**	**Set 6**
Convergence	NSGAII	0.5344	0.6158	0.2646	0.1804	0.1416	0.1484	0.2367	0.2352	0.4439	0.2200	0.7945
	DPSO	0.3480	0.4761	0.3393	0.3019	0.3208	0.3283	0.3150	0.4478	0.6914	0.4568	0.8026
	BBO	0.8206	0.3586	0.1216	0.1144	0.1537	0.1025	0.1042	0.2112	0.5284	0.3727	0.2146
	COIN	0.0000	0.0000	0.0279	0.0553	0.1867	0.1102	0.1187	0.0739	0.1779	0.0947	0.2000
	COIN-E	0.0000	0.0438	0.0166	0.0455	0.1011	0.1175	0.0871	0.0602	0.1328	0.0947	0.1975
Spread	NSGAII	0.7299	0.6667	0.2939	0.4176	0.6892	0.5882	0.4938	0.5927	0.7351	0.6927	0.5715
	DPSO	0.5066	0.4915	0.3989	0.5327	0.4854	0.4802	0.4414	0.4615	0.5502	0.5793	0.5119
	BBO	0.6667	0.4479	0.5664	0.5564	0.5323	0.5612	0.4927	0.6513	0.5898	0.5289	0.5607
	COIN	0.7016	0.3528	0.4391	0.5144	0.4767	0.4604	0.4560	0.3999	0.5723	0.6294	0.6258
	COIN-E	0.5142	0.2667	0.6527	0.3903	0.4951	0.1905	0.6215	0.4505	0.5430	0.6165	0.6394
Ratio	NSGAII	0.0000	0.1429	0.0000	0.0400	0.3333	0.0952	0.0000	0.0238	0.1750	0.6008	0.0000
	DPSO	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	BBO	0.1429	0.2857	0.1176	0.1200	0.3333	0.3333	0.3438	0.0714	0.0000	0.0000	0.5976
	COIN	1.0000	1.0000	0.8235	0.3600	0.2917	0.4048	0.4063	0.4286	0.0375	0.2253	0.3028
	COIN-E	1.0000	0.8571	0.8824	0.6400	0.2083	0.6417	0.5000	0.5238	0.7875	0.1779	0.1036
Time(s)	NSGAII	291.45	134.00	753.09	691.92	1653.92	2026.50	2364.12	3040.57	392825.6	456531.1	727161.8
	DPSO	122.21	82.02	203.40	155.21	414.76	334.67	667.35	291.36	6176.7	10055.9	16251.6
	BBO	197.57	221.75	473.99	702.30	682.40	2335.69	2004.88	1831.17	43565.6	48088.2	574821.9
	COIN	163.33	151.06	339.30	695.70	1188.52	1665.67	2251.20	3106.74	297650.1	393347.7	619731.3
	COIN-E	125.49	148.62	367.49	453.71	1067.93	1215.84	2193.88	2557.13	271015.0	274804.9	688429.6

ตารางที่ 15 ผลการเปรียบเทียบการทดลองของแต่ละอัลกอริทึม

Efficiency	Algorithm	Problems												
		Set 1.1	**Set 1.2**	**Set 2.1**	**Set 2.2**	**Set 3.1**	**Set 3.2**	**Set 4.1**	**Set 4.2**	**Set 5.1**	**Set 5.2**	**Set 6**	**%**	**Std.**
Convergence	NSGAII	46.56	38.42	73.54	81.96	85.84	85.16	76.33	76.48	55.61	78	20.55	65.31	19.87
	DPSO	65.20	52.39	66.07	69.81	67.92	67.17	68.5	55.22	30.86	54.32	19.74	56.11	15.85
	BBO	17.94	64.14	87.84	88.56	84.63	89.75	89.58	78.88	47.16	62.73	78.54	71.79	21.50
	COIN	100	100	97.21	94.47	81.33	88.98	88.13	92.61	82.21	90.53	80	90.50	6.85
	COIN-E	100	95.62	98.34	95.45	89.89	88.25	91.29	93.98	86.72	90.53	80.25	91.85	5.39
Spread	NSGAII	27.01	33.33	70.61	58.24	31.08	41.18	50.62	40.73	26.49	30.73	42.85	41.17	13.28
	DPSO	49.34	50.85	60.11	46.73	51.46	51.98	55.86	53.85	44.98	42.07	48.81	50.55	4.82
	BBO	33.33	55.21	43.36	44.36	46.77	43.88	50.73	34.87	41.02	47.11	43.93	44.05	5.99
	COIN	29.84	64.72	56.09	48.56	52.33	53.96	54.4	60.01	42.77	37.06	37.42	48.83	10.29
	COIN-E	48.58	73.33	34.73	60.97	50.49	80.95	37.85	54.95	45.7	38.35	36.06	51.08	14.65
Ratio	NSGAII	0	14.29	0	4.00	33.33	9.52	0	2.38	17.5	60.08	0	12.83	17.92
	DPSO	0	0	0	0	0	0	0	0	0	0	0	0	0
	BBO	14.29	28.57	11.76	12.00	33.33	33.33	34.38	7.14	0	0	59.76	21.32	17.39
	COIN	100	100	82.35	36.00	29.17	40.48	40.63	42.86	3.75	22.53	30.28	48.00	30.37
	COIN-E	100	85.71	88.24	64.00	20.83	64.17	50.00	52.38	78.75	17.79	10.36	57.48	29.09

7. สรุป

การแก้ปัญหาการจัดลำดับการผลิตรถยนต์ที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านโดยการนำเอา อัลกอริทึม COIN-E เข้ามาประยุกต์ใช้ พบว่า ตัวชี้วัดสมรรถนะของคำตอบที่ได้จากอัลกอริทึม COIN-E มีประสิทธิภาพที่ดีกว่า NSGA-II, DPSO, BBO และ COIN ทั้งในด้านการลู่เข้าของกลุ่มคำตอบที่แท้จริง และอัตราส่วนของจำนวนกลุ่มคำตอบที่ค้นพบเทียบกับกลุ่มคำตอบที่แท้จริง ส่วนด้านเวลาจะเห็นว่า COIN-E ส่วนใหญ่จะใช้เวลาในการหาคำตอบที่น้อยกว่าเมื่อเทียบ COIN และยังแสดงให้เห็นว่า มีความเหมาะสมในการแก้ปัญหาในงานวิจัยนี้

เอกสารอ้างอิง

[1] O. Akgunduz and S. Tunal, "A review of the current applications of genetic algorithms in mixed-model assembly line sequencing," *International Journal of Production Research*, pp. 1-21, 2010.

[2] A. Zinflou, C. Gagne, and M. Gravel, "Design of an efficient genetic algorithm to solve the industrial car sequencing problem," *Advances in Evolutionary Algorithms, I-Tech Education and Publishing*, pp. 377-400, 2008.

[3] C.J. Hyun, Y. Kim, and Y.K. Kim, "A genetic algorithm for multiple objective sequencing problems in mixed model assembly lines," *Computers and Operations Research*, vol. 25, pp. 675-690, 1998.

[4] D. Simon, "Biogeography-Based Optimization," *IEEE transactions on evolutionary computation*, vol. 12, pp. 702-713, 2008.

[5] C. Solnon, V.D. Cung, A. Nguyen, and C. Artigues, "The car sequeincing problem: overview of state-of-the-art methods and industrial case-study of the ROADEF'2005 challenge problem," *European Journal of Operational Research*, pp.191, 912-927, 2008.

[6] M. Knausz, "Parallel Variable Neighbourhood Search for the Car Sequencing Problem," *Fakultat fur Informatik der Technischen Universitat Wien*, 2008.

[7] M. Fliedner and N. Boysen, "Solving the car sequencing problem via Branch & Bound," *European Journal of Operational Research*, pp. 1023-1042, 2008.

[8] A. Zinflou, C. Gagne, and M. Gravel, "Solving the industrial car sequencing problem in a Pareto sense," *Proceedings of the 12th International Workshop on Nature Inspired Distributed Computing*, 2009.

[9] A. Zinflou, C. Gagne, and M. Gravel, "Tackling the industrial car sequencing problem using GISMOO algorithm," *Assembly Line - Theory and Practice*, DOI: 10.5772/21113, pp. 85-106, 2011.

[10] Noppon Kampirom, "Application of Combinatorial Optimization with Coincidence for Multi-objective Sequencing Problems on Mixed-model U-lines in JIT Production Systems," A Thesis for the Degree of Master of Engineering Program in Industrial Engineering, Chulalongkorn University, 2008.

[11] Y.K. Kim, Y. H. Kim, and Y. J. Kim, "Two- sided assembly line balancing: a genetic algorithm approach," *Production Planning & Control*, vol. 11, pp. 44-53, 2000.

[12] U. Ozcan and B. Toklu, "Balancing of mixed-model two-sided assembly lines," *Computers & Industrial Engineering*, pp.217-227, 2009.

[13] A. Konak, D. W. Coit, and E. S. Alice, "Multi-objective optimization using genetic algorithms: A tutorial," *Reliability Engineering and System Safety 91*, pp. 992-1007, 2006.

[14] N. Srinivas and K. Deb, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algrorithms," *The Journal of Evolutionaty Computation*, pp. 221-248, 2002.

[15] P. Chutima and P. Chimklai, "Multi-objective two-sided mixed-model assembly line balancing using particle swarm optimisation with negative knowledge," *Computers & Industrial Engineering*, pp. 39-55, 2012.

[16] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transations on Evolutionary Computation*, pp. 182-197, 2002.

[17] P.R. McMullen, "An efficient frontier approach to addressing JIT sequencing problems with setups via search heuristics," *Computers and Industrial Engineering*, vol. 41, pp. 335-353, 2001.

[18] S.A. Mansouri, "A multi-objective genetic algorithm for mixed-model sequencing on JIT assembly lines," *European Journal of Operational Research*, vol. 167, pp. 696-716, 2005.

[19] R. Kumar, and P.K. Singh, "Pareto evolutionary algorithm hybridized with local search for bi-objective TSP," *Studies in Computational Intelligence*, pp. 361-398, 2007.

[20] M. Li, and J. Zheng, *Spread assessment for evolutionary multi-objective optimization*, Springer-Verlag Berlin Heidelberg, 2009, pp. 216-230.

[21] Y.K. Kim, C. J. Hyun, and Y. Kim, "Sequencing in mixed model assembly lines: A genetic algorithm approach," *Computers and Operations Research*, vol. 23, no.12, pp. 1131-1145, 1996.

[22] P.R. McMullen, P. Tarasewich, and G.V. Frazier, "Using genetic algorithms to solve the multi-product JIT sequencing problem with set-ups," *International Journal of Production Research*, vol. 38, pp. 2653-2670, 2000.