

การจัดลำดับการผลิตรถยนต์แบบหลายวัตถุประสงค์บนสายการ

ประกอบผลิตภัณฑ์ผสมแบบสองด้านด้วยอัลกอริทึม

การบรรจบร่วมกับฟัซซี่ลอจิก

Multi-Objective Car Sequencing Problem On Mixed-Model Two-Sided Assembly Lines With Combinatorial Optimization With Coincidence With Fuzzy Logic

วัชรวิทย์ ถนันทอง* และ ปารเมศ ชุตินา

Watcharawit Tanontong* and Parames Chutima

ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ถ.พญาไท เขตปทุมวัน กทม. 10330

Department of Industrial Engineering, Faculty of Engineering, Chulalongkorn University,
Phayathai Road, Patumwan, Bangkok 10330 Thailand.

*E-mail: newwatcharawit@gmail.com Tel: 6683-6509351

บทคัดย่อ

การจัดลำดับการผลิตรถยนต์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน เพื่อให้เกิดประสิทธิภาพสูงสุดนั้น จัดเป็นปัญหาที่มีความยุ่งยากและสลับซับซ้อน เนื่องจากเป็นปัญหาประเภท Non-deterministic Polynomial Hard: NP-Hard ซึ่งสามารถแก้ปัญหาได้ด้วยวิธีการทางฮิวริสติก (Heuristic) โดยปัญหาการจัดลำดับการผลิตรถยนต์แบบผลิตภัณฑ์ผสมบนสายการประกอบแบบสองด้านนี้ ได้พิจารณาฟังก์ชันวัตถุประสงค์ 3 ฟังก์ชัน คือ จำนวนครั้ง การเปลี่ยนแปลงสีน้อยที่สุด ปริมาณงานที่ทำไม่เสร็จน้อยที่สุด และจำนวนรถยนต์ที่ละเมิดรวมน้อยที่สุด และนำเสนอ อัลกอริทึมการบรรจบร่วมกับฟัซซี่ลอจิก (Combinatorial Optimization with Coincidence with Fuzzy Logic: COIN-F) ซึ่งเป็นอัลกอริทึมที่ประยุกต์มาจาก COIN มาใช้ในการแก้ปัญหา โดยทำการเปรียบเทียบกับอัลกอริทึม COIN-E ที่เป็นที่ยอมรับในการแก้ปัญหาการจัดลำดับการผลิต ผลจากการเปรียบเทียบพบว่า COIN-F มีสมรรถนะในการแก้ปัญหาที่ดีกว่า COIN-E ทั้งในดัชนีการเข้าสู่คำตอบที่เหมาะสมที่สุดแบบพาราโด ดัชนีการกระจายตัวของกลุ่มคำตอบ ดัชนีด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่หาได้เทียบกับกลุ่มคำตอบที่แท้จริง แต่ใช้เวลาในการหาคำตอบนานกว่า เนื่องจากการเพิ่มการทำงานให้กับอัลกอริทึม

ABSTRACT

Multi-objective car sequencing problem on mixed-model with two-sided assembly lines to solve maximum production efficiency is very complicated. This problem is a kind of non-deterministic polynomial hard: NP Hard which only be solved by heuristic method. In this paper, three objective functions are considered including 1) decreasing the number of color changes, 2) minimizing utility work and 3) reducing the number of violation. The Combinatorial Optimization with Coincidence with Fuzzy logic (COIN-F) algorithm developed from its original version (i.e. COIN), is used to solve the problem and compare with that from Combinatorial Optimization with Coincidence Expand (COIN-E)

which is a well-known algorithm. The experimental results indicate that COIN-F gave better performances than these of COIN-E, in terms of convergence to the Pareto-optimal set, spread index of solutions, ratio of non-dominated solution, but the computation time to get solution is more than COIN-E.

1. บทนำ

ปัจจุบันอุตสาหกรรมยานยนต์มีแนวโน้มการแข่งขันที่สูงขึ้นอย่างรวดเร็ว และความต้องการของลูกค้าที่มีความหลากหลายและเปลี่ยนแปลงตลอดเวลา จึงจำเป็นจะต้องมีการปรับปรุงประสิทธิภาพด้านการผลิต และพัฒนาทางด้านเทคโนโลยี ซึ่งสายการประกอบผลิตภัณฑ์เดี่ยว (Single-Model Assembly Lines) อาจจะไม่สามารถตอบสนองได้อย่างเหมาะสม จึงมีการพัฒนาสายการประกอบผลิตภัณฑ์ผสม (Mixed-Model Assembly Lines) มาประยุกต์ใช้กับสายการประกอบแบบสองด้าน (Two-sided assembly line) เพื่อใช้ในการปรับปรุงประสิทธิภาพการผลิตในอุตสาหกรรมยานยนต์

การจัดลำดับการผลิตบนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน มีเป้าหมายคือ การจัดลำดับการผลิตให้มีประสิทธิภาพสูงสุดตามวัตถุประสงค์ที่โรงงานอุตสาหกรรมต้องการ ซึ่งวัตถุประสงค์จะแตกต่างกันออกไปขึ้นอยู่กับลักษณะและความเหมาะสมของอุตสาหกรรมนั้นๆ เช่น ปริมาณงานที่ทำไม่เสร็จในกระบวนการผลิตน้อยที่สุด เวลาในการปรับตั้งเครื่องจักรน้อยที่สุด อัตราความผันแปรของการผลิตน้อยที่สุด ความสม่ำเสมอของการใช้ชิ้นส่วนประกอบ ต้นทุน/เวลาในการปรับตั้งเครื่องน้อยที่สุด ความเสี่ยงในการหยุดของสายการประกอบมีค่าน้อยที่สุด ฯลฯ [1] ซึ่งในการผลิตมีความจำเป็นต้องพิจารณาหลายวัตถุประสงค์ไปพร้อมๆกัน เพื่อให้เกิดประสิทธิภาพสูงสุดบนสายการประกอบ ทำให้ปัญหาในการจัดลำดับในสายการประกอบจึงค่อนข้างมีซับซ้อนและยุ่งยาก ใช้เวลาในการหาคำตอบนาน ซึ่งเป็นปัญหาที่เรียกว่าปัญหา NP-Hard [2] ปัจจุบันจึงมีการนำวิธีฮิวริสติก (Heuristic) เข้ามาช่วยในการแก้ปัญหาประเภทนี้ ทำให้ง่ายต่อการคำนวณมากขึ้น ได้คำตอบค่อนข้างดี ยอมรับได้ และใช้เวลาในการหาคำตอบไม่นานมาก [3] เช่น การประยุกต์ใช้เทคนิคอัลกอริทึม (GA)

[4] การประยุกต์ใช้วิธีการหาค่าเหมาะสมแบบการกระจายของสิ่งมีชีวิตตามภูมิศาสตร์ (BBO) [5] ถึงแม้ว่าคำตอบที่ได้นั้น ไม่อาจจะรับประกันได้ว่าเป็นคำตอบที่ดีที่สุด แต่ผู้ออกแบบสายการประกอบสามารถนำผลที่ได้จากวิธีฮิวริสติกไปปรับปรุงคำตอบจนกว่าจะได้ค่าที่เหมาะสมที่สุดที่จะใช้งานได้ในสภาพแวดล้อมนั้น ๆ [6]

2. สายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน

สายการประกอบผลิตภัณฑ์ผสมแบบสองด้านถูกออกแบบมาเพื่อจัดการสำหรับการผลิตผลิตภัณฑ์ที่มีลักษณะใหญ่และมีการผลิตจำนวนมาก เช่น รถยนต์ รถบรรทุก เป็นต้น โดยจะมีสถานีงานอยู่สองด้านของสายการประกอบ [6] ขนานกันทางด้านซ้าย (L) และทางด้านขวา (R) โดยที่สถานีงานที่อยู่กันจะเรียกว่า Mated-Station โดยมีพนักงานอยู่ในแต่ละด้านและทำการประกอบชิ้นส่วนต่างๆเข้ากับตัวถังรถยนต์ที่จะเคลื่อนผ่านไปในแต่ละสถานีงาน และเพื่อเป็นการตอบสนองความต้องการของลูกค้าที่มีหลากหลายมากขึ้น รูปแบบของสายการผลิตรถยนต์จึงจำเป็นต้องมีความยืดหยุ่น สามารถปรับเปลี่ยนได้ตามความต้องการ โดยที่ผลิตภัณฑ์แต่ละชนิดสามารถเข้าสู่สายการประกอบได้อย่างอิสระ ซึ่งจะแตกต่างจากสายการประกอบแบบหลายผลิตภัณฑ์ที่จะต้องผลิตผลิตภัณฑ์ทีละชนิด ซึ่งสายการประกอบแบบสองด้านมีประโยชน์ต่อกระบวนการผลิต คือ ความยาวสายการประกอบแบบสองด้านจะสั้นกว่าสายการประกอบแบบด้านเดียว ลดความสูญเสียเปล่า เช่น ลดระยะทางการเคลื่อนที่ของพนักงาน ลดเวลาปรับตั้งเครื่องและเวลาผลิต และลดต้นทุนค่าเครื่องมือและของใช้ประจำตัวของพนักงานที่สามารถใช้ร่วมกันได้ในคู่สถานีงาน [7] โดยที่สายการประกอบแบบสองด้านจะแบ่งประเภทชิ้นงานที่จัดสรรลงในสถานีงาน 2 แบบ คือ ชิ้นงานที่สามารถจัดสรรลงในสถานีงานได้เพียงทางด้านเดียวเท่านั้น คือ

ทางด้านซ้าย (L-type) ทางด้านขวา (R-type) ของสายการประกอบ และชิ้นงานที่สามารถจัดสรรลงในสถานีงานได้ทั้งสองด้านของสายการประกอบ (E-type) [8]

3. วิธีการแก้ปัญหาการหาค่าเหมาะสมที่สุดที่มีหลายวัตถุประสงค์

การพิจารณาแก้ปัญหาสำหรับหลายวัตถุประสงค์ให้มีประสิทธิภาพที่ดีไปพร้อมกัน เป็นเรื่องที่แทบจะเป็นไปไม่ได้ยาก เนื่องจากวัตถุประสงค์นั้นอาจมีความขัดแย้งระหว่างกันในการหาค่าน้อยที่สุดหรือค่าที่มากที่สุดของแต่ละวัตถุประสงค์ จึงยากที่จะบอกได้ว่าเป็นคำตอบใดเป็นคำตอบที่ดีที่สุด [9] สำหรับงานวิจัยนี้ได้นำวิธีเชิงกลุ่มที่ดีที่สุด (Pareto-based Approach) มาใช้ในการค้นหาคำตอบ ซึ่งสิ่งสำคัญในการแก้ปัญหาการหาค่าเหมาะสมที่มีหลายวัตถุประสงค์ คือ การกำหนดค่าความแข็งแรง (Fitness Assignment) ให้กับสมาชิก โดยนำเทคนิคการกำหนดค่าความแข็งแรงด้วยวิธีเชิงกลุ่มที่ดีที่สุดที่มีการจัดอันดับคำตอบแบบพารेटอ (Pareto Ranking Approach) มาใช้ คำตอบที่ได้จะอยู่ในรูปของกลุ่มคำตอบที่ดีที่สุด (Pareto Optimal Set) ซึ่งเป็นกลุ่มคำตอบที่ไม่สามารถบ่งบอกได้ว่าคำตอบใดเป็นคำตอบที่ดีกว่ากันหรือไม่ถูกรอบงำ (Non-dominated Solution) จากคำตอบอื่น [10] ในงานวิจัยนี้เป็นการหาค่าเหมาะสมที่สุดสำหรับการหาค่าน้อยที่สุดของแต่ละฟังก์ชันวัตถุประสงค์ โดยพิจารณาไปพร้อมกันทั้งหมด ดังสมการต่อไปนี้

$$\text{Minimize } \{f_1(\bar{x}), f_2(\bar{x}), \dots, f_k(\bar{x})\} \quad (1)$$

โดย \bar{x} คือเวกเตอร์ของตัวแปรในการตัดสินใจ $f_i(\bar{x})$ คือฟังก์ชันวัตถุประสงค์ในแต่ละเรื่อง ถ้าเวกเตอร์ของตัวแปรตัดสินใจ x ให้คำตอบที่ดีกว่าตัวแปรตัดสินใจ y แล้ว จะได้ว่า $f_i(x) \leq f_i(y)$ สำหรับทุกค่าของ $i \in \{1, 2, \dots, k\}$ และ $f_i(x) < f_i(y)$ อย่างน้อย 1 ค่าของ $i \in \{1, 2, \dots, k\}$

3.1 จำนวนครั้งการเปลี่ยนแปลงสี้น้อยที่สุด

ลำดับในการผลิตรถยนต์จะมีผลอย่างมากต่อการผันสีรถยนต์ เพราะทุกครั้งที่มีการผันสีที่แตกต่างกันในลำดับการผลิตต่อกันมีความจำเป็นจะต้องล้างหัวปืนผันสี ดังนั้นหากลำดับการผลิตรถยนต์ที่มีสีเดียวกันติดต่อกันก็ไม่ต้องมีความจำเป็นจะต้องทำความสะอาดหัวปืนผันสีและเปลี่ยนน้ำยาในการผัน ซึ่งเป็นการลดต้นทุนในการผลิต แต่หากมีการผันสีเดียวกันติดต่อกันหลายลำดับเกินค่าที่กำหนดนั้นจำเป็นที่จะต้องมีการล้างทำความสะอาดหัวปืนผันสี เนื่องจากหากไม่ทำความสะอาดหัวปืนผันสีจะมีการอุดตันและสีที่ผันจะไม่มีคุณภาพ สูตรการคำนวณจำนวนครั้งการเปลี่ยนแปลงสี [11] ดังสมการที่ 2

$$f_1(x) = \sum_{k=0}^{nc} \delta_{COL_{k,k+1}} \quad (2)$$

เมื่อ $f_1(x)$ คือ จำนวนการเปลี่ยนแปลงสี $\delta_{COL_{k,k+1}}$ คือ การเปลี่ยนสีผันรถยนต์ ณ ตำแหน่ง k และ $k+1$ โดยที่ $\delta_{COL_{k,k+1}} = 1$ เมื่อมีการเปลี่ยนสีผันรถยนต์ และ $\delta_{COL_{k,k+1}} = 0$ เมื่อไม่มีการเปลี่ยนสีผันรถยนต์

k คือ ตำแหน่งใดๆ ในลำดับการผลิต โดยที่ $k \in [1, nc]$ เมื่อพิจารณาลำดับการผลิตในรอบการผลิตปัจจุบัน $k \in [-\infty, 0]$ เมื่อพิจารณาลำดับการผลิตในรอบการผลิตก่อนหน้า และ $k \in [nc + 1, \infty]$ เมื่อพิจารณาลำดับการผลิตในรอบการผลิตถัดไป nc คือ ผลรวมสัดส่วนการผลิต (MPS) หรือจำนวนชิ้นผลิตภัณฑ์ทั้งหมดที่จะเข้าสู่สายการประกอบ

3.2 ปริมาณงานที่ไม่เสร็จในการผลิตน้อยที่สุด

ปริมาณงานที่ไม่เสร็จคือ ผลรวมของปริมาณงานของทุกผลิตภัณฑ์ที่ไม่เสร็จในรอบเวลาการทำงานนั้นๆ ในแต่ละชิ้นงานบนสายการประกอบ ยิ่งปริมาณงานที่ไม่เสร็จในการผลิตมีค่าน้อยจะทำให้ประหยัดเวลาที่ใช้ลดต้นทุนในด้าน Utility Worker และสายการประกอบจะมีความไหลลื่นและมีประสิทธิภาพที่มากยิ่งขึ้น สูตรการคำนวณ [12] ดังสมการที่ 3, 4 และ 5

$$f_2(x) = \sum_{n_m=1}^{N_M} \left(\sum_{i=1}^I U_{i,n_m} + Z_{(i+1),n_m} / v_c \right) \quad (3)$$

$$U_{i,n_m} = \left\{ \max \left[0, \frac{(Z_{i,n_m} + v_c \sum_{m=1}^M x_{i,m} (t_{2n_m-1,m} + Y_{2n_m-1,m}) - L_{n_m})}{v_c} \right], \max \left[0, \frac{(Z_{i,n_m} + v_c \sum_{m=1}^M x_{i,m} (t_{2n_m,m} + Y_{2n_m,m}) - L_{n_m})}{v_c} \right] \right\} \quad (4)$$

$$Z_{(i+1),n_m} = \max \left\{ \max \left[0, \min \left(Z_{i,n_m} + v_c \sum_{m=1}^M X_{i,m} (t_{2n_m-1,m} + Y_{2n_m-1,m}) - \gamma_c, L_{n_m} - \gamma_c \right) \right], \max \left[0, \min \left(Z_{i,n_m} + v_c \sum_{m=1}^M X_{i,m} (t_{2n_m,m} + Y_{2n_m,m}) - \gamma_c, L_{n_m} - \gamma_c \right) \right] \right\} \quad (5)$$

- เมื่อ $f_2(x)$ คือ ปริมาณงานที่ไม่เสร็จในการผลิต
- N_M คือ คู่สถานีงาน โดย $n_m = 1, 2, \dots, N_M$
- U_{i,n_m} คือ ปริมาณงานที่ไม่เสร็จในลำดับการผลิตที่ i ในคู่สถานีงาน n_m
- $Z_{(i+1),n_m}$ คือ เวลาเริ่มงานในลำดับการผลิตที่ i
- v_c คือ ความเร็วของสายการผลิต
- $t_{2n_m-1,m}$ คือ เวลาดำเนินการทั้งหมดของผลิตภัณฑ์ m ที่สถานีงานด้านซ้ายในคู่สถานีงาน n_m
- $t_{2n_m,m}$ คือ เวลาดำเนินการทั้งหมดของผลิตภัณฑ์ m ที่สถานีงานด้านขวาในคู่สถานีงาน n_m
- $Y_{2n_m-1,m}$ คือ เวลาเดินเปล่าทั้งหมดของผลิตภัณฑ์ m ที่สถานีงานด้านซ้ายในคู่สถานีงาน n_m
- $Y_{2n_m,m}$ คือ เวลาเดินเปล่าทั้งหมดของผลิตภัณฑ์ m ที่สถานีงานด้านขวาในคู่สถานีงาน n_m
- γ คือ ช่วงเวลาการปล่อยผลิตภัณฑ์เข้าสู่สายการผลิต (Launch Interval)
- L_{n_m} คือ ความยาวของคู่สถานีงาน n_m

3.3 จำนวนรถยนต์ที่ละเมิดรวมน้อยที่สุด

รถยนต์แต่ละรุ่นจะมีตัวเลือกที่ต่างกัน โดยในการจัดลำดับการผลิตรถยนต์นั้นจำเป็นต้องพิจารณาตัวเลือกของรถยนต์พยายามไม่ให้ผลิตเกินค่าที่กำหนด โดยพิจารณาจากค่า อัตราส่วน p/q (เมื่อ p คือ ความต้องการตัวเลือก และ q คือ จำนวนลำดับการผลิตย่อย) เช่น อัตราส่วนการประกอบระบบเบรก ABS เท่ากับ 1 ต่อ 3 จะหมายความว่า ในการผลิตรถยนต์ 3 คันจะมีรถยนต์ที่มีตัวเลือกนี้ได้ไม่เกิน 1 คัน ถ้ามีการเกินจะนับจำนวนที่เกินเป็นจำนวนการละเมิดที่เกิดขึ้น สูตรการคำนวณรถยนต์ที่ละเมิดรวม [11] ดังสมการที่ 6

$$f_3(x) = \sum_{o \in O} \left\{ NPV_{o;i \in day J} \left\| \bigcup_{i=q_o+1}^{mc} PV_o(i, \dots, i+q_o-1) \right\| \right\} \quad (6)$$

เมื่อ $f_3(x)$ คือ จำนวนรถยนต์ที่ละเมิดน้อยที่สุด

- o คือ ตัวเลือก (Options) เมื่อ $o=0, 1, 2, \dots, O$
- $NPV_{o;i \in day J}$ คือ เซตของตัวเลือกที่มีการละเมิดเกิดขึ้นจากตำแหน่งที่ i ถึง $i+q_o-1$
- $PV_o(i, \dots, i+q_o-1)$ คือ การนับจำนวนที่มีการละเมิดของตัวเลือก o ในวันที่ J
- \cup คือ Union operator

ในคู่สถานีงาน

4. การหาค่าที่เหมาะสมที่สุดแบบวิธีการบรรจบร่วมกับฟัซซี่ลอจิก

อัลกอริทึมแบบวิธีการบรรจบร่วมกับฟัซซี่ลอจิก ได้พัฒนามาจากอัลกอริทึมบรรจบ (COIN) โดยอาศัยแนวคิดวิธีการบรรจบแบบเดิมคือ การให้รางวัลแก่สตริงคำตอบที่ดี และทำการลงโทษแก่สตริงคำตอบที่ไม่ดี เพื่อเป็นตัวกำหนดทิศทางของคำตอบสุดท้าย โดยมีการประยุกต์ใช้หลักการตรรกศาสตร์คลุมเครือ (Fuzzy Logic) ที่ตั้งอยู่บนพื้นฐานของเหตุผลง่ายๆ คือ “ถ้า X และ Y ดังนั้น Z” ซึ่งเป็นเครื่องมือพื้นฐานทางคณิตศาสตร์ที่ช่วยในการจัดการกับข้อมูลซึ่งไม่มีความชัดเจนคลุมเครือ [13] สำหรับการแก้ปัญหาการจัดลำดับผลิตภัณฑ์เข้าสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านได้มีการประยุกต์ใช้ฟัซซี่ลอจิกในการปรับปรุงค่าพารามิเตอร์ k ที่ใช้ในการให้รางวัลและลงโทษ ให้มีความเหมาะสมกับสถานการณ์ที่แตกต่างกันไปในแต่ละรอบการทำงาน เพื่อให้ได้คำตอบที่เข้าใกล้คำตอบที่แท้จริง และเพิ่มการกระจายตัวของกลุ่มคำตอบให้ดีขึ้น

4.1 ขั้นตอนของอัลกอริทึม COIN-F

ขั้นที่ 1 สร้างประชากรเบื้องต้น โดยการสุ่มจากตารางเมตริกซ์ความน่าจะเป็นร่วม ดังตารางที่ 1 โดยที่ความน่าจะเป็นร่วมจะมีค่าเท่ากับ $\frac{1}{n-1}$ เมื่อ n คือผลรวมสัดส่วนความต้องการผลิตภัณฑ์ทั้งหมด (MPS) ดังนั้นความน่าจะเป็นร่วมเบื้องต้น จะมีค่าเท่ากับ $\frac{1}{12-1} = 0.091$

ขั้นที่ 2 ประเมินค่าฟังก์ชันวัตถุประสงค์และกำหนดค่าความแข็งแรง (fitness) ของแต่ละสตริงคำตอบ ดังตารางที่ 2

ขั้นที่ 3 คัดเลือกสตริงคำตอบที่ดีและไม่ดี โดยที่สตริงคำตอบที่ดีจะมีค่าความแข็งแรงน้อยที่สุด และสตริง

คำตอบที่ไม่ดีจะมีค่าความแข็งแรงมากที่สุด และปรับปรุงค่าความน่าจะเป็นจากตารางความน่าจะเป็นเริ่มต้น โดยกำหนดให้ k คือค่าพารามิเตอร์ในการให้รางวัล/ลงโทษ ดังสมการที่ 7 และ 8 ตามลำดับ (กำหนดให้ $k = 0.1$)

$$x = \frac{k}{n-1} \quad (7)$$

$$x = \frac{k}{(n-1)^2} \quad (8)$$

จากตารางที่ 2 จะเห็นได้ว่าสตริงที่ 1 และ 2 เป็นสตริงคำตอบที่ดีเนื่องจากมีค่า fitness เท่ากับ 1 ดังนั้นจะทำการให้รางวัลแก่สตริงที่ 1 และ 2 ทีละสตริง โดยการเพิ่มความน่าจะเป็นเท่ากับ $\frac{1}{12-1} = 0.091$ ให้กับคู่อันดับ (11, 12)

และลดความน่าจะเป็นของคู่อันดับทั้งหมดในแถวเดียวกัน คือ (11, 1), (11, 2), (11, 3), ..., (11, 12) ด้วยค่าความน่าจะเป็นเท่ากับ $\frac{1}{(12-1)^2} = 0.0082$ จากนั้นพิจารณา

ให้รางวัลคู่อันดับถัดไป คือ (12, 3) และลดความน่าจะเป็นของคู่อันดับทั้งหมดในแถวเดียวกัน คือ (12, 1), (12, 2), (12, 3), ..., (12, 11) ทำเช่นนี้ไปจนทุกคู่อันดับทั้ง 2 สตริงคำตอบ และในสตริงคำตอบที่ 3 ซึ่งเป็นสตริงคำตอบที่ไม่ดีจะทำการลดความน่าจะเป็นในคู่อันดับ (11, 2) และเพิ่มความน่าจะเป็นของคู่อันดับทั้งหมดในแถวเดียวกัน คือ (11, 1), (11, 2), (11, 3), ..., (11, 12) ทำเช่นนี้จนครบทุกคู่อันดับจะได้ผลดังตารางที่ 3

ขั้นที่ 4 เทคนิคการเก็บค่าที่ดีที่สุด โดยการนำสตริงคำตอบที่ดีที่สุดในรอบก่อนหน้ามารวมกับรอบปัจจุบัน แล้วนำมาเรียงลำดับด้วยวิธี Non-dominated Sorting เพื่อเก็บค่าสตริงที่ให้ค่าความแข็งแรงเท่ากับ 1 ซึ่งเป็นผลลัพธ์ที่ดีที่สุดในรอบนั้นๆ ดังตารางที่ 4

ขั้นที่ 5 กลับไปทำขั้นที่ 2 ซ้ำจนครบจำนวนรอบ

ตารางที่ 1 ความน่าจะเป็นร่วมเริ่มต้น

| From/To | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 11 | 11 | 12 |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 |
| 2 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 |
| 3 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 |
| 4 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 |
| 5 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 |
| 6 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 |
| 7 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 |
| 8 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 | 0.091 |
| 9 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 | 0.091 |
| 10 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 | 0.091 |
| 11 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 | 0.091 |
| 12 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.091 | 0.000 |

จากตารางที่ 1 คือค่าความน่าจะเป็นร่วมเริ่มต้นที่จะนำไปใช้ในการสุ่มประชากร (สตริงคำตอบ) เริ่มต้นจำนวน 3 ประชากร โดยเริ่มการสุ่มค่าตัวเลข 1 - 12 มาหนึ่งค่าจากนั้นสุ่มค่าถัดไปด้วยค่าความน่าจะเป็นดัง

ค่าในตารางของแต่ละลำดับ โดยเริ่มต้นจะเห็นได้ว่าทุกลำดับมีโอกาสหรือค่าความน่าจะเป็นในการถูกสุ่มเท่ากันทั้งหมดจากการคำนวณจากสมการในขั้นตอนที่ 1

ตารางที่ 2 ค่าฟังก์ชันวัตถุประสงค์และค่าความแข็งแรงของแต่ละสตริง

| String | Model sequence | f(1) | f(2) | f(3) | Fitness |
|--------|----------------------------|------|----------|------|---------|
| 1 | 11 12 3 6 5 2 1 7 9 10 8 4 | 10 | 65.37500 | 5 | 1 |
| 2 | 1 2 8 7 12 10 6 3 9 11 4 5 | 8 | 69.18750 | 6 | 1 |
| 3 | 11 2 12 4 8 6 1 9 3 10 7 5 | 10 | 69.03125 | 5 | 2 |

จากตารางที่ 2 คือผลจากการสุ่มประชากรเริ่มต้นจากตารางที่ 1 และทำการประเมินค่าฟังก์ชันวัตถุประสงค์ตามสมการที่ 2 สมการที่ 3 และสมการที่ 4 ตามลำดับของแต่ละสตริงคำตอบ พร้อมทั้งกำหนดค่าความแข็งแรงด้วย

วิธี Non-dominated Sorting ซึ่งจะใช้ในการพิจารณาคำตอบที่ดีและไม่ดี เพื่อทำการปรับปรุงความน่าจะเป็นต่อไป สำหรับสตริงคำตอบที่ดีจะมีค่า Fitness น้อยและสตริงคำตอบที่ไม่ดีจะมีค่า Fitness มาก

ตารางที่ 3 ความน่าจะเป็นร่วมหลังการปรับปรุง

| From/To | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 11 | 11 | 12 |
|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 0.000 | 0.099 | 0.090 | 0.090 | 0.090 | 0.090 | 0.099 | 0.090 | 0.081 | 0.090 | 0.090 | 0.090 |
| 2 | 0.099 | 0.000 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.099 | 0.090 | 0.090 | 0.090 | 0.081 |
| 3 | 0.090 | 0.090 | 0.000 | 0.090 | 0.090 | 0.099 | 0.090 | 0.090 | 0.099 | 0.081 | 0.090 | 0.090 |
| 4 | 0.090 | 0.090 | 0.090 | 0.000 | 0.099 | 0.090 | 0.090 | 0.081 | 0.090 | 0.090 | 0.099 | 0.090 |
| 5 | 0.099 | 0.099 | 0.090 | 0.090 | 0.000 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.081 | 0.090 |
| 6 | 0.081 | 0.090 | 0.099 | 0.090 | 0.099 | 0.000 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 |
| 7 | 0.090 | 0.090 | 0.090 | 0.090 | 0.081 | 0.090 | 0.000 | 0.090 | 0.099 | 0.090 | 0.090 | 0.099 |
| 8 | 0.090 | 0.090 | 0.090 | 0.099 | 0.090 | 0.081 | 0.099 | 0.000 | 0.090 | 0.090 | 0.090 | 0.090 |
| 9 | 0.090 | 0.090 | 0.081 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.000 | 0.099 | 0.099 | 0.090 |
| 10 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.099 | 0.081 | 0.099 | 0.090 | 0.000 | 0.090 | 0.090 |
| 11 | 0.090 | 0.081 | 0.090 | 0.099 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.000 | 0.099 |
| 12 | 0.090 | 0.090 | 0.099 | 0.081 | 0.090 | 0.090 | 0.090 | 0.090 | 0.090 | 0.099 | 0.090 | 0.000 |

จากตารางที่ 3 คือผลจากการให้รางวัลโดยการเพิ่มความน่าจะเป็นให้กับสตริงคำตอบที่ดี และการลงโทษโดยการลดความน่าจะเป็นแก่คำตอบที่ไม่ดี เพื่อให้คู่อันดับที่ดีที่อยู่ติดกันมีความน่าจะเป็นสูงขึ้น และคู่อันดับที่ไม่ดี

ที่อยู่ติดกันมีความน่าจะเป็นลดลง ทำให้โอกาสในการถูกสุ่มในรอบต่อไปคู่อันดับที่ดีจะมีโอกาสในการถูกสุ่มที่มากขึ้น และโอกาสในการถูกสุ่มสำหรับคู่อันดับที่ไม่ดีจะมีโอกาสน้อยลง

ตารางที่ 4 เทคนิคการเก็บค่าที่ดีที่สุด

| String | Model sequence | f(1) | f(2) | f(3) | Fitness |
|-------------|----------------------------|------|----------|------|---------|
| รอบก่อนหน้า | 11 12 3 6 5 2 1 7 9 10 8 4 | 10 | 65.37500 | 5 | 2 |
| | 1 2 8 7 12 10 6 3 9 11 4 5 | 8 | 69.18750 | 6 | 1 |
| รอบปัจจุบัน | 2 11 7 5 9 3 1 6 10 8 4 12 | 10 | 63.46875 | 4 | 1 |

จากตารางที่ 4 คือการนำสตริงคำตอบที่ดีที่สุดในรอบก่อนหน้ามารวมกับสตริงที่ดีที่สุดในรอบปัจจุบัน แล้วทำการกำหนดค่าความแข็งแรงใหม่ด้วยวิธี Non-dominated Sorting เพื่อหาคำตอบที่ดีที่สุดเมื่อจบรอบ

การทำงานนั้นๆ ซึ่งจะเห็นได้ว่ามีเฉพาะสตริงคำตอบที่ 2 ในรอบก่อนหน้า และสตริงคำตอบที่ 1 ในรอบปัจจุบันที่ยังคงถูกเก็บไว้เป็นสตริงคำตอบที่ดีที่สุดที่จะนำไปรวมกับสตริงคำตอบที่ดีที่สุดในรอบการทำงานต่อไป

ตารางที่ 5 ผลจากการสุ่มประชากรรอบที่ 5

| String | Model sequence | f(1) | f(2) | f(3) | Fitness |
|--------|----------------------------|------|----------|------|---------|
| 1 | 7 2 11 10 5 3 8 6 9 4 1 12 | 8 | 71.43750 | 4 | 1 |
| 2 | 7 11 5 8 1 10 6 12 4 3 2 9 | 10 | 65.71875 | 5 | 1 |
| 3 | 12 11 7 4 10 6 3 9 5 8 1 2 | 10 | 63.06250 | 7 | 1 |

จากตารางที่ 5 คือตัวอย่างสตริงคำตอบที่ได้จากการสุ่มประชากรในรอบที่ 5 เพื่อใช้เป็นตัวอย่างในการคำนวณค่าการลู่เข้ากลุ่มคำตอบ (Progressive Convergence : PC) จากรอบการทำงานที่ 1 ถึงรอบการทำงานที่ 5 เพื่อใช้พิจารณาว่าคำตอบมีการพัฒนาไปไกลมากน้อยเพียงใด

มาจากรอบที่ 1 ถึงรอบที่ 5 ก่อนข้างดี และ Number of NC = 0 นั่นคือหมายความว่าในทุก ๆ รอบคำตอบมีการพัฒนาไปในทางที่ดีขึ้น หรือหมายถึงทุกๆรอบสามารถเจอคำตอบที่ดีกว่า (มีค่า fitness น้อยกว่า) คำตอบในรอบก่อนหน้าทั้ง 5 รอบ

ขั้นที่ 6 ใช้ฟิชชี่ลอจิกในการปรับปรุงค่าพารามิเตอร์ในการให้รางวัลและลงโทษ (k) ในทุกๆรอบการทำงาน ของฟิชชี่ลอจิกที่กำหนดไว้ โดยกำหนดให้ช่วง $Convergence = 0.7$ และจำนวนรอบในการใช้ฟิชชี่ลอจิก $r = 5$ (จากการทดสอบพารามิเตอร์)

$$PC = \frac{1}{|S^*|} \sum_{y \in S^*} \min \{d_{xy} | x \in S_j\} \quad (9)$$

ขั้นที่ 6.1 คำนวณค่าการลู่เข้าของกลุ่มคำตอบ (Progressive Convergence : PC) [12] จากสมการที่ 9 ระหว่างคำตอบจากรอบที่ 1 ถึงรอบที่ 5 และพิจารณาจำนวนรอบที่คำตอบไม่มีการพัฒนา (Number of NC) จากสมการที่ 10 ซึ่งจากตัวอย่างข้างต้นมีค่า Progressive Convergence = 0.4937 นั่นคือคำตอบมีการพัฒนา

โดยที่ $d_{xy} = \sqrt{\sum_{i=1}^k \frac{(f_i(x) - f_i(y))^2}{f_i^{\max} - f_i^{\min}}}$

เมื่อ PC คือ ค่าการลู่เข้ากลุ่มคำตอบจากรอบที่ 1 ถึงรอบที่ 5 หรือค่าระยะห่างระหว่างคำตอบโดยเฉลี่ยจากรอบที่ 1 ถึงรอบที่ 5

S_j คือ กลุ่มคำตอบที่ดีที่สุดจากกลุ่มที่ j เมื่อ $j = 1, 2, \dots, J$

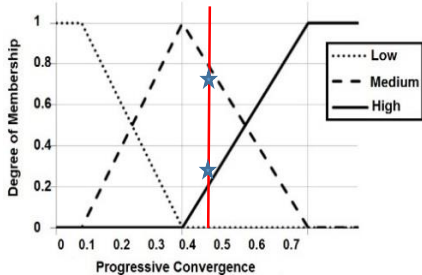
d_{xy} คือ ระยะทางระหว่างคำตอบที่ได้ x กับคำตอบที่แท้จริง y

$$NC = \sum_{i=1}^r R_i^* \quad (10)$$

เมื่อ NC คือ จำนวนรอบที่คำตอบไม่มีการพัฒนา

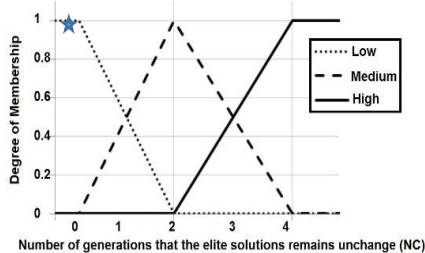
R_i^* คือ รอบการทำงานใดๆที่คำตอบไม่มีการพัฒนาให้ได้คำตอบที่ดีขึ้น

r คือ จำนวนรอบการทำงานซ้ำของฟิชชี่ลอจิก
 ชั้นที่ 6.2 สร้างกราฟของข้อมูลนำเข้า (input) แต่ละพารามิเตอร์ และกำหนดระดับค่าความเป็นสมาชิก (μ)



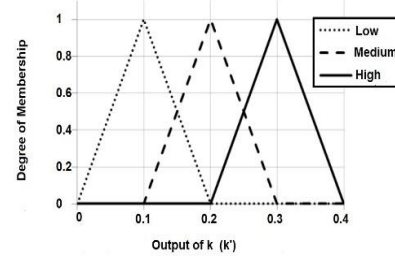
รูปที่ 1 กราฟข้อมูลนำเข้าของค่า Convergence

จากรูปที่ 1 จะเห็นได้ว่าเมื่อค่า PC ค่า 0.4937 จะมีค่าความเป็น High = 0.2635 ค่าความเป็น Medium = 0.7365 และค่าความเป็น Low = 0



รูปที่ 2 กราฟข้อมูลนำเข้าของค่า Number of NC

จากรูปที่ 2 จะเห็นได้ว่าเมื่อค่า NC ค่า 0 จะมีค่าความเป็น Low = 1 ค่าความเป็น Medium และ High = 0



รูปที่ 3 กราฟค่าพารามิเตอร์ที่ใช้ในการให้รางวัลและลงโทษ (Output)

จากรูปที่ 3 จะเห็นได้ว่าค่ากึ่งกลางความเป็น Low = 0.1 ค่ากึ่งกลางความเป็น Medium = 0.2 และค่ากึ่งกลางความเป็น High = 0.3

ชั้นที่ 6.3 สร้างกฎทางฟิชชี่ (กฎ 9 ข้อทางฟิชชี่ลอจิก) โดยอาศัยพื้นฐานของตรรกศาสตร์คลุมเครือที่ว่า “ถ้า x และ y ดังนั้น z ” หรือ ถ้า PC มีค่า x NC มีค่า y ดังนั้น k ควรมีค่า z เพื่อใช้เป็นเกณฑ์ในการกำหนดระดับของค่าพารามิเตอร์ k ที่ใช้ในการให้รางวัลและลงโทษ ดังตารางที่ 6

ตารางที่ 6 กฎทางฟิชชี่

| Rule | Progressive Convergence | Number of NC | Parameter :k |
|------|-------------------------|--------------|--------------|
| 1 | Low | Low | Medium |
| 2 | Low | Medium | High |
| 3 | Low | High | High |
| 4 | Medium | Low | Medium |
| 5 | Medium | Medium | Medium |
| 6 | Medium | High | Medium |
| 7 | High | Low | Low |
| 8 | High | Medium | Low |
| 9 | High | High | Medium |

จากตารางที่ 6 คือกฎทางฟิชชี่ที่ใช้กำหนดระดับของค่าพารามิเตอร์ k จากค่า PC และ NC เช่น เมื่อค่า PC เป็น Low นั้นหมายความว่าระยะห่างระหว่างคำตอบรอบที่ 1 ถึงรอบที่ 5 มีค่าน้อยหรือคำตอบมีการพัฒนาได้น้อย และ NC เป็น High นั้นหมายความว่าจำนวนรอบคำตอบที่ไม่มีการพัฒนามีค่ามาก ดังนั้น หาก PC เป็น Low และ NC เป็น High จึงควรกำหนดให้ค่าพารามิเตอร์ k เป็น High เพื่อเป็นการเพิ่มค่าความน่าจะเป็นในการให้รางวัลและ

ลงโทษเพื่อให้คำตอบหลุดจากกลุ่มคำตอบเฉพาะที่และมีการพัฒนาคำตอบที่ดีขึ้น

ขั้นที่ 6.4 ใช้กฎทางฟuzzy ในการกำหนดระดับของค่าพารามิเตอร์ (k) โดยพิจารณาจาก

$$k = \min(\mu_{con}, \mu_{NC}) \quad (11)$$

ตารางที่ 7 กฎทางฟuzzy และการกำหนดค่าระดับความเป็นสมาชิก

| Rule | Antecedent | | Consequent |
|------|-------------------------|------------------|---------------|
| | Progressive Convergence | The number of NC | Parameter : k |
| 1 | Low | Low | Medium |
| | 0 | 1 | 0 |
| 2 | Low | Medium | High |
| | 0 | 0 | 0 |
| 3 | Low | High | High |
| | 0 | 0 | 0 |
| 4 | Medium | Low | Medium |
| | 0.7365 | 1 | 0.7365 |
| 5 | Medium | Medium | Medium |
| | 0.7365 | 0 | 0 |
| 6 | Medium | High | Medium |
| | 0.7365 | 0 | 0 |
| 7 | High | Low | Low |
| | 0.2635 | 1 | 0.2635 |
| 8 | High | Medium | Low |
| | 0.2635 | 0 | 0 |
| 9 | High | High | Medium |
| | 0.2635 | 0 | 0 |

จากตารางที่ 7 เป็นการกำหนดค่าความเป็นสมาชิกจากตารางกฎทางฟuzzy โดยใช้ค่าจากกราฟข้อมูลการนำเข้าของค่า PC และ NC และพิจารณาเลือกพารามิเตอร์ k ตามสมการที่ 11 เพื่อนำไปใช้คำนวณค่าพารามิเตอร์ k ต่อไป

ขั้นที่ 6.5 การแปลงค่าผลลัพธ์แบบฟuzzy กลับเป็นค่าที่แน่นอน (Defuzzyfication) โดยในงานวิจัยชิ้นนี้เลือกใช้วิธี Weighted Average Method เนื่องจากฟังก์ชันการเป็นสมาชิกมีความสมมาตรและมีมากกว่า 1 รูป โดยนำค่าความเป็นสมาชิกที่ได้จากการแปลค่าจากกฎทางฟuzzy ของแต่ละระดับคูณกับจุดที่มีค่าความเป็นสมาชิกสูงสุดของกราฟแต่ละรูป แล้วนำมาคำนวณหาค่าเฉลี่ยของค่าความน่าจะเป็นที่ใช้ในการให้รางวัลและลงโทษ ดังสมการที่ 12 [14]

$$Low = \sqrt{R_7^2 + R_8^2}$$

$$Low = \sqrt{0.2635^2 + 0^2} = 0.2635$$

$$Medium = \sqrt{R_1^2 + R_4^2 + R_5^2 + R_6^2 + R_9^2}$$

$$Medium = \sqrt{0^2 + 0.7365^2 + 0^2 + 0^2 + 0^2} = 0.7365$$

$$High = \sqrt{R_2^2 + R_3^2}$$

$$High = \sqrt{0^2 + 0^2} = 0$$

$$k' = \frac{(Low * CL) + (Medium * CM) + (High * CH)}{Low + Medium + High} \quad (12)$$

เมื่อ CL คือ ค่ากลางของค่า Low

CM คือ ค่ากลางของค่า Medium

CH คือ ค่ากลางของค่า High

ดังนั้น

$$k' = \frac{(0.2635 * 0.1) + (0.7365 * 0.2) + (0 * 0.3)}{0.2635 + 0.7365}$$

$$k' = 0.1736$$

ดังนั้น $k' = 0.1736$ คือ ค่าในการให้รางวัลและลงโทษที่จะนำไปใช้ในการหาค่าตอบรอบต่อไป

ขั้นที่ 7 กลับไปทำขั้นที่ 2 เพื่อเริ่มกระบวนการค้นหาคำตอบรอบถัดไป และวนซ้ำจนครบจำนวนรอบที่กำหนดไว้ โดยใช้ค่า k' ที่ปรับแล้วในการให้รางวัลและลงโทษแก่สตริงที่ดีและไม่ดีต่อไป

5. การทดลองทางคอมพิวเตอร์

ในงานวิจัยฉบับนี้เป็นการทดลองแก้ตัวอย่างปัญหาผ่าน โปรแกรม Visual Studio 2013 ภาษา C++ ประมวลผลด้วย คอมพิวเตอร์โน้ตบุ๊ก Intel Core i5-3210M CPU 2.5 GHz RAM 8.0 GB โดยใช้ อัลกอริทึมการหาค่าที่เหมาะสมที่สุดแบบวิธีการบรรจบ

ร่วมกับพีชชีลอจิก (COIN-F) และวิธีการบรรจบแบบขยาย (COIN-E) ในการแก้ปัญหา

5.1 การออกแบบการทดลอง

งานวิจัยฉบับนี้ได้ทำการแก้ปัญหาด้อย่างทั้งสิ้น 5 ปัญหา [12] โดยแบ่งเป็นปัญหามิติเล็ก คือ เซตปัญหาที่ 1 และ 2 ปัญหามิติกลาง คือ เซตปัญหาที่ 3 และ 4 และปัญหามิติใหญ่ คือ เซตปัญหาที่ 5 โดยแต่ละปัญหาจะทำการทดลองซ้ำ 2 รอบการทดลอง ทำให้การทดลองย่อยในงานวิจัยนี้มีขนาด 20 การทดลองต่อ 1 อัลกอริทึม ดังตารางที่ 8 และกำหนดค่าพารามิเตอร์ที่ใช้ดังตารางที่ 9 และดังตารางที่ 10

ตารางที่ 8 เซตปัญหาที่ใช้ในการทดลอง [12]

| ปัญหาการทดลอง | รอบการทำซ้ำ | จำนวนรุ่นรถยนต์ | สัดส่วนผลิตภัณฑ์ (MPS) | จำนวนสี | จำนวนตัวเลือก |
|---------------|-------------|-------------------------------------|------------------------|---------|---------------|
| Set 1.1 | 1 | 5:3:2:1:1 | 12 | 4 | 2 |
| | 2 | 5:3:2:1:1 | 12 | 4 | 2 |
| Set 1.2 | 1 | 4:4:2:1:1 | 12 | 4 | 2 |
| | 2 | 4:4:2:1:1 | 12 | 4 | 2 |
| Set 2.1 | 1 | 7:3:2:2:1 | 15 | 10 | 4 |
| | 2 | 7:3:2:2:1 | 15 | 10 | 4 |
| Set 2.2 | 1 | 4:3:3:3:2 | 15 | 10 | 4 |
| | 2 | 4:3:3:3:2 | 15 | 10 | 4 |
| Set 3.1 | 1 | 8:7:2:2:1 | 20 | 13 | 7 |
| | 2 | 8:7:2:2:1 | 20 | 13 | 7 |
| Set 3.2 | 1 | 5:4:4:4:3 | 20 | 13 | 7 |
| | 2 | 5:4:4:4:3 | 20 | 13 | 7 |
| Set 4.1 | 1 | 7:5:1:1:1:1:1:1:1 | 20 | 15 | 8 |
| | 2 | 7:5:1:1:1:1:1:1:1 | 20 | 15 | 8 |
| Set 4.2 | 1 | 4:4:4:2:1:1:1:1:1 | 20 | 15 | 8 |
| | 2 | 4:4:4:2:1:1:1:1:1 | 20 | 15 | 8 |
| Set 5.1 | 1 | 20:20:20:15:15:1:1:1:1:1:1:1:1 | 100 | 17 | 11 |
| | 2 | 20:20:20:15:15:1:1:1:1:1:1:1:1 | 100 | 17 | 11 |
| Set 5.2 | 1 | 15:15:10:10:10:10:10:10:4:1:1:1:1:1 | 100 | 17 | 11 |
| | 2 | 15:15:10:10:10:10:10:10:4:1:1:1:1:1 | 100 | 17 | 11 |

ตารางที่ 9 การกำหนดค่าพารามิเตอร์ของ COIN-F

| COIN-F | |
|-------------------|------|
| Population size | 100 |
| Reward/Punish | 0.1 |
| จำนวนรอบพีชชีลจิก | 5 |
| จำนวนรอบในการรัน | 2000 |

จากตารางที่ 9 คือค่าพารามิเตอร์ที่ใช้ในการทดลองของ COIN-F ซึ่งได้มาจากการทดสอบค่าพารามิเตอร์

ตารางที่ 10 การกำหนดค่าพารามิเตอร์ของ COIN-E

| COIN-E | |
|------------------------|-------------|
| Population size | 100 |
| Reward/Punish | 0.1 |
| สัดส่วนการสุ่มจากตาราง | 70:10:10:10 |
| จำนวนรอบในการรัน | 2000 |

จากตารางที่ 10 คือค่าพารามิเตอร์ที่ใช้ในการทดลองของ COIN-E

5.2 ตัวชี้วัดสมรรถนะของอัลกอริทึม

งานวิจัยฉบับนี้อาศัยตัวชี้วัดสมรรถนะของอัลกอริทึมตามรูปแบบของปัญหาหลายวัตถุประสงค์ทั้งสิ้น 5 ตัวชี้วัดในการเปรียบเทียบผลลัพธ์ของอัลกอริทึมได้แก่ ตัวชี้วัดด้านการเข้าสู่กลุ่มคำตอบที่เหมาะสมที่สุดแบบพารето (Convergence to the Pareto-optimal Set) ใช้ประเมินว่า กลุ่มคำตอบที่อัลกอริทึมหามาได้มีความใกล้เคียงกับกลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริงมากน้อยเพียงใด หากค่าการเข้าสู่กลุ่มคำตอบที่เหมาะสมที่สุดแบบพารетоมีค่าเข้าใกล้ 0 หมายความว่ากลุ่มคำตอบที่อัลกอริทึมหามาได้มีความใกล้เคียงกับกลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริงมาก ตัวชี้วัดด้านการกระจายตัวของกลุ่มคำตอบ (Spread of Solution) ใช้บ่งชี้ว่ากลุ่มคำตอบมีการกระจายตัวกว้างและสม่ำเสมอเพียงใด หากค่าการกระจายตัวของกลุ่มคำตอบมีค่าเข้าใกล้ 0 หมายความว่ากลุ่มคำตอบที่อัลกอริทึมหามาได้มีการกระจายตัวกว้างและสม่ำเสมอดี ตัวชี้วัดด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่หาได้เทียบเท่ากับกลุ่มคำตอบที่แท้จริง I (Ratio

of Non-dominated Solution I) ใช้บ่งชี้ว่ากลุ่มคำตอบที่หาได้ทั้งหมดเป็นคำตอบที่ตรงกับคำตอบที่เหมาะสมที่สุดที่แท้จริงคิดเป็นอัตราส่วนเท่าใด ตัวชี้วัดด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่หาได้เทียบเท่ากับกลุ่มคำตอบที่แท้จริง II (Ratio of Non-dominated Solution II) ใช้แสดงว่ากลุ่มคำตอบที่หาได้ทั้งหมดที่เป็นคำตอบที่ตรงกับกลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริงเทียบกับจำนวนคำตอบที่อัลกอริทึมหามาได้คิดเป็นอัตราส่วนเท่าใด ซึ่งหากค่าอัตราส่วนของจำนวนกลุ่มคำตอบที่หาได้เทียบเท่ากับกลุ่มคำตอบที่แท้จริงยิ่งเข้าใกล้ 1 หมายความว่ากลุ่มคำตอบที่อัลกอริทึมหามาได้คือกลุ่มคำตอบที่แท้จริงมากเท่านั้น และตัวชี้วัดด้านระยะเวลาที่ใช้ในการคำนวณหาคำตอบ (Computer Time) ยิ่งใช้เวลาในการหาคำตอบน้อยยิ่งดี

5.3 ผลการทดลอง

จากการทดลองแก้ปัญหาตัวอย่าง ผลการทดลองที่ได้อยู่ในรูปตัวชี้วัดต่างๆ ดังตารางที่ 11 จะเห็นได้ว่าตัวชี้วัดด้านการเข้าสู่กลุ่มคำตอบที่แท้จริง (Convergence) ส่วนใหญ่นั้น COIN-F จะให้ผลลัพธ์ที่ดีกว่า COIN-E อย่างชัดเจนในทุกๆ เซตปัญหา สำหรับตัวชี้วัดด้านการกระจายตัวของกลุ่มคำตอบ (Spread) พบว่า COIN-F และ COIN-E จะให้ผลลัพธ์อยู่ในระดับที่ใกล้เคียงกัน โดยในเซตปัญหาขนาดเล็กและขนาดใหญ่ COIN-E จะให้ผลลัพธ์ที่ดีกว่า แต่ในเซตปัญหาขนาดกลาง COIN-F จะให้ผลลัพธ์ที่ดีกว่า สำหรับตัวชี้วัดด้าน RNDS-I, RNDS-II พบว่า COIN-F จะให้ผลลัพธ์ที่ดีกว่าอย่างชัดเจนในทุกๆเซตปัญหา และตัวชี้วัดด้าน CPU Time พบว่าในเซตปัญหาเล็ก COIN-F จะใช้เวลาในการหาคำตอบใกล้เคียงกับ COIN-E แต่เมื่อปัญหามีขนาดใหญ่ขึ้น COIN-F จะใช้เวลาในการหาคำตอบมากกว่า COIN-E อย่างชัดเจน จากการวิเคราะห์ผลการทดลองข้างต้นจึงสามารถสรุปได้ว่า COIN-F นั้นมีสมรรถนะในการแก้ปัญหาเหนือกว่า COIN-E ในทุกขนาดปัญหา และถือว่าอัลกอริทึมการบรรจบร่วมกับพีชชีลจิกเป็นอีกอัลกอริทึมหนึ่งที่สามารถแก้ปัญหาได้อย่างมีประสิทธิภาพ

ตารางที่ 11 ผลการทดลองแก้ปัญหาตัวอย่าง

| Problem | Convergence | | Spread | | RNDS-I | | RNDS-II | | CPU-Time | |
|---------|-------------|--------|--------|--------|--------|--------|---------|--------|----------|--------|
| | COIN-F | COIN-E | COIN-F | COIN-E | COIN-F | COIN-E | COIN-F | COIN-E | COIN-F | COIN-E |
| 1 | 0.0203 | 0.0832 | 0.4525 | 0.4392 | 1.0000 | 0.3636 | 0.8571 | 0.2857 | 5.725 | 3.307 |
| 2 | 0.0455 | 0.1439 | 0.4603 | 0.4574 | 0.7500 | 0.1818 | 0.8182 | 0.1818 | 5.616 | 3.244 |
| 3 | 0.0457 | 0.0855 | 0.3789 | 0.1772 | 0.7273 | 0.4167 | 0.6667 | 0.4167 | 5.397 | 4.788 |
| 4 | 0.1038 | 0.0820 | 0.4514 | 0.4371 | 0.6667 | 0.4615 | 0.5000 | 0.5000 | 5.116 | 5.002 |
| 5 | 0.0096 | 0.1104 | 0.6128 | 0.7205 | 0.9310 | 0.1000 | 0.9000 | 0.1000 | 6.988 | 4.414 |
| 6 | 0.0090 | 0.1283 | 0.5824 | 0.6633 | 0.9677 | 0.0625 | 0.9375 | 0.0625 | 7.660 | 4.383 |
| 7 | 0.0664 | 0.0630 | 0.5750 | 0.5886 | 0.5652 | 0.4615 | 0.5200 | 0.4800 | 6.848 | 4.383 |
| 8 | 0.0285 | 0.1850 | 0.5511 | 0.5944 | 1.0000 | 0.0870 | 0.9091 | 0.0909 | 7.400 | 4.430 |
| 9 | 0.0405 | 0.0550 | 0.6988 | 0.7056 | 0.9714 | 0.7391 | 0.4762 | 0.4348 | 101.30 | 43.742 |
| 10 | 0.0514 | 0.0614 | 0.5020 | 0.6633 | 0.7619 | 0.4667 | 0.6154 | 0.5385 | 100.19 | 42.868 |
| 11 | 0.0168 | 0.0962 | 0.7866 | 0.6822 | 0.9512 | 0.1818 | 0.8667 | 0.1778 | 91.696 | 38.984 |
| 12 | 0.0287 | 0.0648 | 0.7329 | 0.7854 | 0.8718 | 0.3721 | 0.7727 | 0.3636 | 93.85 | 39.140 |
| 13 | 0.0639 | 0.0461 | 0.5942 | 0.6716 | 0.6774 | 0.4444 | 0.5122 | 0.4878 | 63.351 | 27.393 |
| 14 | 0.0253 | 0.0869 | 0.5705 | 0.6629 | 0.8158 | 0.3143 | 0.7381 | 0.2619 | 63.679 | 27.721 |
| 15 | 0.0415 | 0.0736 | 0.7033 | 0.6388 | 0.7561 | 0.2979 | 0.6889 | 0.3111 | 63.242 | 30.286 |
| 16 | 0.0351 | 0.1061 | 0.6070 | 0.5468 | 0.6129 | 0.5652 | 0.6333 | 0.4333 | 63.336 | 26.832 |
| 17 | 0.0360 | 0.0679 | 0.5657 | 0.5955 | 0.8077 | 0.4615 | 0.6364 | 0.3636 | 381.74 | 171.36 |
| 18 | 0.0268 | 0.0665 | 0.5599 | 0.4922 | 0.8214 | 0.4667 | 0.6866 | 0.3134 | 388.81 | 179.86 |
| 19 | 0.0345 | 0.0506 | 0.6017 | 0.6930 | 0.5690 | 0.3768 | 0.5593 | 0.4407 | 414.98 | 202.94 |
| 20 | 0.0434 | 0.0536 | 0.6759 | 0.5787 | 0.6786 | 0.5667 | 0.5278 | 0.4722 | 432.18 | 206.43 |

6. สรุป

งานวิจัยนี้นำเสนอการแก้ปัญหาการจัดลำดับการผลิตที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน เพื่อตอบสนองความต้องการของผู้บริโภคที่มีความหลากหลายและมีการเปลี่ยนแปลงที่รวดเร็ว โดยมีวัตถุประสงค์ในการจัดลำดับการผลิตที่ให้ความสำคัญเท่ากันทั้ง 3 วัตถุประสงค์ คือ จำนวนครั้งการเปลี่ยนแปลงสิน้อยที่สุด ปริมาณงานที่ไม่เสร็จน้อยที่สุด และจำนวนรถยนต์ที่ละเมิดร่น้อยที่สุด และได้ทำการนำเสนออัลกอริทึมการบรรจบร่วมกับพีชชีลอจิก (Combinatorial Optimization with Fuzzy Logic: COIN-F) เพื่อใช้ในการแก้ปัญหาดังกล่าว ผลจากการทดลองในการแก้ปัญหาตัวอย่าง พบว่าอัลกอริทึมการบรรจบร่วมกับพีชชีลอจิก มีสมรรถนะที่เหนือกว่าอัลกอริทึมการบรรจบแบบขยายอย่างเห็นได้ชัดเจน คือ

ด้านการเข้าสู่คำตอบที่เหมาะสมที่สุดแบบพารโตพบว่าผลลัพธ์จาก COIN-F ให้ค่าที่ดีกว่า COIN-E คิดเป็นร้อยละ 85 ของทั้งหมด ด้านการกระจายตัวของกลุ่มคำตอบพบว่าผลลัพธ์จาก COIN-F ให้ค่าที่ดีกว่า COIN-E คิดเป็นร้อยละ 55 ของทั้งหมด ด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่หาได้เทียบเท่ากับกลุ่มคำตอบที่แท้จริงพบว่าผลลัพธ์จาก COIN-F มีค่าที่ดีกว่า COIN-E คิดเป็นร้อยละ 100 แต่ด้านเวลาที่ใช้ในการหาคำตอบพบว่า COIN-F ใช้เวลาในการหาคำตอบมากกว่า COIN-E คิดเป็นร้อยละ 100 ซึ่งเป็นผลมาจากการเพิ่มการทำงานให้กับอัลกอริทึมเพื่อให้ได้คำตอบที่ดีขึ้น จึงสามารถสรุปได้ว่า COIN-F เป็นอัลกอริทึมหนึ่งที่สามารถแก้ปัญหาการจัดลำดับการผลิตประเภทนี้ได้เป็นอย่างดีมีประสิทธิภาพ

เอกสารอ้างอิง

- [1] Akgunduz, O. S. and Tunali, S. A review of the current applications of genetic algorithms in mixed-model assembly line sequencing. *International Journal of Production Research*, 2011; 49: 4483-4503.
- [2] Tsai, L. H. Mixed-Model sequencing to minimize utility work and the risk of conveyor stoppage. *Management Science*, 1995; 41: 485-495.
- [3] Hyun, C. J., Kim, Y., and Kim, Y. K. A genetic algorithm for multiple objective sequencing problems in mixed model assembly lines. *Computers & Operations Research*, 1998; 25: 675-690.
- [4] ปารเมศ ชุตินา. การประยุกต์เทคนิคการจัดตารางในอุตสาหกรรม. พิมพ์ครั้งที่ 2, กรุงเทพมหานคร: สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย, 2008.
- [5] วันวิสา นฤมิตวงศ์. การจัดลำดับการผลิตที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านภายใต้ผลกระทบจากการเรียนรู้. วิทยานิพนธ์, จุฬาลงกรณ์มหาวิทยาลัย, 2555.
- [6] Bartholdi, J. J. Balancing 2 -sided assembly lines - a case-study. *International Journal of Production Research*, 1993; 31: 2447-2461.
- [7] Ozcan, U. and Toklu, B. Balancing of mixed-model two-sided assembly lines. *Computers & Industrial Engineering*, 2009; 57: 217-227.
- [8] Kim, Y. K., Kim, Y. H., and Kim, Y. J. Two-sided assembly line balancing: a genetic algorithm approach. *Production Planning & Control*, 2000; 11: 44-53.
- [9] Konak, A., Coit, D. W., and Smith, A. E. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety*, 2006; 91: 992-1007.
- [10] นพพล คำภิรมย์. การประยุกต์ใช้เมมเมติกอัลกอริทึมสำหรับปัญหาการจัดลำดับผลิตภัณฑ์ที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบลักษณะตัวยูในระบบผลิตแบบทันเวลาพอดี. วิทยานิพนธ์, จุฬาลงกรณ์มหาวิทยาลัย, 2551.
- [11] Chutima, P. and Olarnviwatchai, S. Multi-objective car sequencing problem on mixed-model two-sided assembly lines. *Journal on Intelligent Manufacturing*, 2016; 1-20.
- [12] สดาพร โอฟารวิวัฒน์ชัย. การจัดลำดับการผลิตรถยนต์แบบหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน. วิทยานิพนธ์, จุฬาลงกรณ์มหาวิทยาลัย, 2556.
- [13] Djam, X. Y., Wajiga, G. M., Kimbi, Y. H., and Blamah, N. V. A fuzzy expert system for the management of malaria. *International Journal of Pure and Applied Sciences and Technology*, 2011; 5: 84-108.
- [14] Hellendoorn, H. and Thomas, C. Defuzzification in fuzzy controllers. *Journal of Intelligent and Fuzzy Systems*, 1993; 1(2): 109-123.