

การแก้ปัญหการจัดสมดุลและจัดสรรพนักงานหลายทักษะบนสายการประกอบ ผลิตภัณฑ์ผสมแบบขนานที่มากวัตถุประสงค์

Solving Line Balancing and Allocation Multi-Skilled Workers Problem on Parallel Assembly Line under Many-Objective

ชินวิชญ์ สินธุเดชากุล¹ ปารเมศ ชูติมา^{2*}

¹นิสิตปริญญาโท ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
254 ถนนพญาไท แขวงวังใหม่ เขตปทุมวัน กรุงเทพมหานคร 10310

²ศาสตราจารย์ ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
254 ถนนพญาไท แขวงวังใหม่ เขตปทุมวัน กรุงเทพมหานคร 10310

Chinnawich Sintudachakul¹ Parames Chutima^{2*}

¹ Master's Degree, Department of Industrial Engineering, Faculty of Engineering,
Chulalongkorn University, 254 Phayathai Road, Patumwan, Bangkok 10330

² Professor, Department of Industrial Engineering, Faculty of Engineering,
Chulalongkorn University, 254 Phayathai Road, Patumwan, Bangkok 10330

*Corresponding author: Email: parames.c@chula.ac.th

บทคัดย่อ

วิธีการหาค่าที่เหมาะสมด้วยวิธีการเชิงวิวัฒนาการแบบหลายวัตถุประสงค์โดยยึดหลักการจำแนก (A Multi-Objective Evolutionary Algorithm based on Decomposition : MOEA/D) เป็นฮิวริสติกที่มีแนวคิดในการแก้ปัญหาที่มากวัตถุประสงค์ไปพร้อมกันโดยจะจำแนกปัญหาออกเป็นปัญหาย่อยๆ บทความนี้จะนำเสนออัลกอริทึม MOEA/D ร่วมกับวิธี Bisection เพื่อใช้แก้ปัญหการจัดสมดุลและจัดสรรพนักงานหลายทักษะบนสายการประกอบผลิตภัณฑ์ผสมแบบขนาน ซึ่งทักษะที่หลากหลายเกิดจากความหลากหลาย โดยมีวัตถุประสงค์ที่พิจารณา 4 วัตถุประสงค์ ได้แก่ รอบเวลาดำเนินการน้อยที่สุด จำนวนสถานีน้อยที่สุด ความแตกต่างของภาระงานระหว่างสถานีงานน้อยที่สุด และความไม่เกี่ยวข้องกันของชิ้นงานน้อยที่สุด ผลที่ได้จากการทดลองคือ MOEA/D มีสมรรถนะในการแก้ปัญหาที่ดีกว่าอัลกอริทึมการบรรจบ (Combinatorial Optimization with Coincidence : COIN) ในด้านการใช้เวลาหาคำตอบที่แท้จริง ซึ่งเป็นตัวชี้วัดที่สำคัญที่สุดในการเปรียบเทียบอัลกอริทึม และสามารถค้นหาคำตอบได้ดีกว่าทุกวัตถุประสงค์ ถึงแม้การกระจายตัวของกลุ่มคำตอบจะไม่ดีเท่า และใช้เวลานานกว่า COIN

คำสำคัญ: สายการประกอบผสมแบบขนาน คนทุพพลภาพ การจัดสมดุลและจัดสรรพนักงาน วิธีการเชิงวิวัฒนาการแบบหลายวัตถุประสงค์โดยยึดหลักการจำแนก

ABSTRACT

A multi-objective evolutionary algorithm based on decomposition (MOEA/D) is an evolutionary meta-heuristic which uses the concept of solving problems with many objectives simultaneously by classification problem into various subproblems. This paper applies MOEA/D algorithm in conjunction

with the bisection method for solving line balancing and allocation problems of multi-skilled workers whom some are disable on parallel assembly lines under four objectives, i.e. minimize cycle time, minimize the number of stations, minimize different workload between workstations, and minimize index of task-unrelated. The experiment results show that MOEA/D obtains better performance than the Combinatorial Optimization with Coincidence (COIN) algorithm in terms of convergence which is the main concern of algorithm comparison. Although, its spread and CPU time are inferior to COIN.

Keyword: Mixed-Model Parallel Assembly line, Disable Worker, Line Balancing and Worker Allocation, MOEA/D

1. บทนำ

สายการประกอบคือ กระบวนการผลิตประเภทหนึ่ง ที่ชิ้นงานในแต่ละชิ้นงานถูกนำมาประกอบเข้าไปกับ ชิ้นงานก่อนหน้าตามลำดับของขั้นตอนการประกอบ (Task Sequence) โดยต้องไม่ผิดเงื่อนไขของลำดับ ก่อนหลังของชิ้นงาน (Precedence Constraint) และ หลังจากทำการประกอบชิ้นงานทั้งหมดที่ได้รับมอบหมาย บนแต่ละสถานีงานเสร็จตามรอบเวลาดำเนินการ (Cycle Time) ชิ้นงานประกอบนี้จะถูกเคลื่อนย้ายไปยังสถานี งานที่อยู่ในลำดับถัดไปด้วยอุปกรณ์ขนถ่ายวัสดุอัตโนมัติ เช่น สายพานลำเลียง หุ่นยนต์ เป็นต้น การทำงานเช่นนี้ จะเกิดขึ้นจนกระทั่งการประกอบในขั้นตอนสุดท้ายเสร็จ สิ้น ก็จะกลายเป็นผลิตภัณฑ์สำเร็จรูปในที่สุด ทั้งนี้เราจะ เรียกการจัดสรรชิ้นงานต่างๆ ให้กับแต่ละสถานีงานในสาย การประกอบ โดยต้องไม่ละเมิดต่อข้อจำกัดต่างๆ ซึ่งจะ เรียกกระบวนการทั้งหมดนี้ว่า การจัดสมดุลสายการผลิต (Line Balancing)

ความแปรผันของเวลาการดำเนินการที่เกิดขึ้นกับ ปัญหาการจัดสมดุลสายการประกอบ สามารถแบ่งออกได้ เป็น 3 ประเภท [1] ดังนี้ เวลาดำเนินงานเชิงกำหนด (Deterministic Task Time) คือเวลาดำเนินงานของแต่ละ ชิ้นงานจะมีค่าคงที่, เวลาดำเนินงานเชิงพลวัต (Dynamic Task Time) คือเวลาดำเนินงานของแต่ละ ชิ้นงานมีค่าลดลงอย่างเป็นระบบอันเกิดจากเส้นโค้งการ เรียนรู้ (Learning Curve) และ เวลาดำเนินงานเชิงสุ่ม (Stochastic Task Time Problem) คือเวลา

ดำเนินงานของแต่ละชิ้นงานไม่คงที่โดยเกิดจากความผัน แปรของเวลาดำเนินงาน ซึ่งมีผลมาจากหลายสาเหตุ เช่น เครื่องจักรเสีย สูญเสียแรงจูงใจ สิ่งแวดล้อมในการทำงาน และทักษะของพนักงาน [2] ซึ่งประเภทที่ผู้วิจัยจะนำมา ศึกษาในบทความนี้ คือการแก้ปัญหาการจัดสมดุลสายการ ประกอบแบบเวลาดำเนินงานเชิงสุ่ม (Balancing Stochastic Assembly Lines) ที่เกิดจากทักษะของ พนักงานที่เกิดความทุพพลภาพและความเชี่ยวชาญของ พนักงาน โดยทั่วไปแล้วจะสามารถจำแนกปัญหาดังกล่าว ได้ 2 รูปแบบ คือ ปัญหาการจัดสมดุลสายการประกอบ แบบเวลาดำเนินงานเชิงสุ่ม และ ปัญหาการจัดสมดุล และการจัดสรรพนักงานที่มีหลายทักษะ

ในบทความนี้จะศึกษาในรูปแบบที่ 2 ซึ่งจะมองว่า คนทุพพลภาพนั้นเป็นคนที่มีทักษะต่ำ (Low-Skill Worker) ซึ่งสามารถทำงานบางงานได้ช้ากว่าคนปกติและ ไม่สามารถทำงานบางงานได้เลย ดังนั้นจะทำให้เกิด ข้อจำกัดและความซับซ้อนในการจัดสมดุลสายการ ประกอบที่มากขึ้น รวมถึงเวลาที่ใช้ในการจัดสมดุลสาย การประกอบ แต่ถ้าสามารถแก้ปัญหาการจัดสมดุลสาย การประกอบที่มีพนักงานทุพพลภาพให้เกิดประสิทธิภาพที่ ดีขึ้นได้ จะสามารถลดต้นทุนและเพิ่มโอกาสที่ผู้ทุพพล ภาพจะมีงานทำ ซึ่งถือว่าเป็นการช่วยสังคมได้อีกทางหนึ่ง ด้วย เพราะในปัจจุบันมีจำนวนผู้ทุพพลภาพเพิ่มขึ้นเท่ากับ 10% ของประชากรโลก ซึ่งบุคคลประเภทนี้จะมีความ ลำบากมากในการหางานทำ และส่วนใหญ่ผู้ทุพพลภาพมี แนวโน้มที่จะถูกจ้างให้ทำงานเพียงชั่วคราวเท่านั้น

ในประเทศไทยมีจำนวนผู้ทุพพลภาพประมาณ 2.2% ของประชากรทั้งประเทศ และปัจจุบันได้มี พ.ร.บ. ส่งเสริมพัฒนาคุณภาพชีวิตคนทุพพลภาพ พ.ศ.2550 โดยได้เพิ่มสภาพบังคับให้กับระบบสัดส่วนการจ้างงานคนทุพพลภาพด้วยการกำหนดให้มีการจ่ายเงินเข้ากองทุนส่งเสริมและพัฒนาคุณภาพชีวิตคนทุพพลภาพล่าช้าต้องเสียดอกเบี้ยร้อยละ 7.5 ต่อปี โดยกำหนดสัดส่วนการจ้างงานบุคคลปกติต่อคนทุพพลภาพ 100 : 1

ปัญหาการจัดสมดุลและจัดสรรพนักงานที่มีหลายทักษะบนสายการประกอบผลิตภัณฑ์ผสมแบบขนานที่มากวัตถุประสงคฺภายใต้ปัญหาประเภทที่ 2 หมายถึง การหาเวลาดำเนินการที่น้อยที่สุดโดยจะกำหนดจำนวนสถานีงาน (Workstation) หรือกำหนดจำนวนพนักงานมาให้ ซึ่งถือเป็นปัญหาที่เป็นไปได้ยากที่จะได้คำตอบที่ดีที่สุดในระยะเวลาอันสั้น (NP-Hard) จากการสำรวจงานวิจัยที่ผ่านมาวิธีที่นำมาใช้ในการแก้ไขประเภทนี้บนสายการประกอบประเภทต่างๆ คือวิธีการทางฮิวริสติก ซึ่งคำตอบที่ได้ถือว่าเป็นคำตอบที่ดี (Best Solution) ที่ยอมรับได้ในการแก้ปัญหาเหล่านี้ เช่น Kilincci [3] ได้นำเสนอ Petri Net-Based (PN) Heuristic Chaves และคณะ [4] ได้นำเสนอวิธีการหาคำตอบโดย Clustering Search (CS) และ Zhang และคณะ [5] ได้นำเสนอ Random Key-Based Genetic Algorithm (rkGA) ร่วมกับ Fuzzy Logic Controller (FLC) ซึ่งงานวิจัยที่กล่าวมาเป็นเพียงสายการประกอบเส้นตรง Araújo และคณะ [6] ได้นำเสนอ Tabu Search และ Biased Random-Key Genetic Algorithm (BRKGA) ในการแก้ปัญหาบนสายการประกอบขนานที่ไม่มีความเกี่ยวเนื่องกัน

ในงานวิจัยนี้จึงจะทำการศึกษาการจัดสมดุลและการจัดสรรพนักงานที่มีหลายทักษะบนสายการประกอบผลิตภัณฑ์ผสมบนสายการประกอบแบบเส้นตรงขนานกัน เพราะด้วยลักษณะเฉพาะตัวของสายการประกอบเส้นตรงแบบขนาน จะทำให้เราสามารถลดจำนวนสถานีงาน ลดพนักงาน และลดเวลาสูญเสียได้ [7] รวมถึงอาจจะเพิ่มประสิทธิภาพและผลิตภาพของสายการประกอบที่มี

พนักงานเป็นผู้ทุพพลภาพ

จากที่มาข้างต้น บทความนี้จึงนำเสนอวิธีการแก้ปัญหาด้วยวิธีการเชิงวิวัฒนาการแบบหลายวัตถุประสงค์โดยยึดหลักการจำแนก (Multi-Objective Evolutionary Algorithm Based On Decomposition: MOEA/D) ที่เป็นวิธีที่มีประสิทธิภาพในการแก้ปัญหาแบบมากวัตถุประสงคฺและเหมาะกับปัญหาที่เป็น Scalar Optimization ซึ่งถูกพัฒนาโดย Zhang และ Li [8] พบว่าสามารถแก้ปัญหา 0-1 Knapsack ได้อย่างมีประสิทธิภาพเหนือกว่าอัลกอริทึมอื่นๆที่เป็นที่นิยมร่วมกับวิธี Bisection มาประยุกต์ใช้ในการแก้ปัญหาการจัดสมดุลและจัดสรรพนักงานหลายทักษะบนสายการประกอบผลิตภัณฑ์ผสมแบบขนานที่มากวัตถุประสงคฺภายใต้ปัญหาประเภทที่ 2 โดยมีจำนวนวัตถุประสงคฺทั้งสิ้น 4 วัตถุประสงคฺ ซึ่งจะพิจารณาค่าที่เหมาะสมที่สุดไปพร้อมๆกัน ได้แก่ รอบเวลาดำเนินการน้อยที่สุด จำนวนสถานีงานน้อยที่สุด ความแตกต่างของปริมาณภาระงานระหว่างสถานีงานมีค่าน้อยที่สุด และความไม่เกี่ยวเนื่องกันของงานในแต่ละสถานีงานมีค่าน้อย

2. การจัดสมดุลและการจัดสรรพนักงานหลายทักษะให้กับสถานีงานบนสายการประกอบผลิตภัณฑ์ผสมแบบขนานภายใต้ปัญหาประเภทที่ 2

ปัญหาประเภทนี้จะมีลักษณะคล้ายคลึงกับปัญหาการจัดสมดุลสายการประกอบผลิตภัณฑ์ผสมแบบขนานในด้านของลักษณะของสายการประกอบ โดยสายการประกอบประเภทนี้จะมีลักษณะการจัดวางในรูปของสถานี (Station) ที่จัดเรียงกันอย่างต่อเนื่อง [9] โดยในแต่ละสถานีอาจประกอบด้วยสถานีงานแยก (Separate Workstation) ที่ถูกจัดสรรชิ้นงานจากเพียงสายการประกอบเดียว อย่างมาก 2 สถานีงาน แต่ต้องเป็นสถานีแยกที่อยู่คนละสายการประกอบ หรือสถานีงานร่วม (Common Workstation) ที่ถูกจัดสรรชิ้นงานจากทั้งสองสายการประกอบ และส่วนที่ต่างออกไป คือลักษณะของ

โจทย์ปัญหา สดริงคำตอบ การถอดรหัสสดริงคำตอบ ซึ่งสามารถอธิบายได้ดังนี้

2.1 ลักษณะของโจทย์ปัญหา

ในบทความนี้จะเน้นปัญหาการจัดสมดุลและการจัดสรรพนักงานหลายทักษะให้กับสถานงานบนสายการประกอบผลิตภัณฑ์ผสมแบบขนาน ดังนั้นต้องมีการปรับแต่งโจทย์ปัญหาเพื่อประยุกต์ให้เข้ากับปัญหาที่กำลังวิจัย โดยหนึ่งชิ้นงานจะประกอบด้วยเวลาของพนักงาน 3 ระดับ คือ ทักษะสูง ทักษะปกติ และทักษะต่ำ ซึ่งเวลาของชิ้นงานที่ได้มาจากโจทย์ปัญหาตัวอย่างผู้วิจัยจะกำหนดให้เป็นเวลาของพนักงานทักษะปกติ ส่วนพนักงานทักษะสูงนั้นจะมีบางชิ้นงานที่สามารถทำได้เร็วกว่าพนักงานทักษะปกติ ส่วนพนักงานทักษะต่ำนั้นจะมีบางชิ้นงานที่สามารถทำได้ช้ากว่าพนักงานทักษะปกติและบางชิ้นงานไม่สามารถทำได้เลย ดังแสดงตัวอย่างในตารางที่ 1

ตารางที่ 1 เวลาในแต่ละชิ้นงานของพนักงานที่ปรับแต่ง

Task	Task Time Problem	Modified Task Time		
		High	Medium	Low
1	6	5	6	8
2	2	2	2	3
3	5	4	5	-
4	7	6	7	-
5	1	1	1	3
6	2	2	2	4
7	3	3	3	5
8	6	4	6	7
9	5	5	5	7
10	5	4	5	6
11	4	3	4	5

2.2 ลักษณะของสดริงคำตอบ

สดริงคำตอบ ที่ปรากฏในบทความนี้จะอยู่ในรูปของลำดับความสำคัญในการจัดสรรซึ่งจะแบ่งออกเป็น 2 สดริงคำตอบย่อย คือ สดริงคำตอบของชิ้นงาน และสดริงคำตอบของพนักงาน ดังตารางที่ 2 และ 3 โดยแต่ละสดริงคำตอบของชิ้นงานและพนักงานนั้นจะมีลำดับความสำคัญที่ไม่ซ้ำกัน ซึ่งค่าน้อยจะมีโอกาสจัดสรรก่อน

2.3 การถอดรหัสสดริงคำตอบ

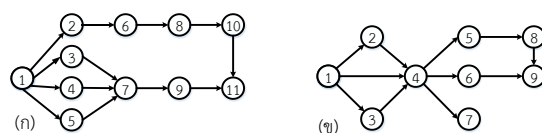
ขั้นตอนในการถอดรหัสสดริงคำตอบของการจัดสมดุลและการจัดสรรพนักงานหลายทักษะให้กับสถานงานบนสายการประกอบผลิตภัณฑ์ผสมแบบขนานภายใต้ปัญหาประเภทที่ 2 [5] มี 4 ขั้นตอนหลัก ดังนี้

ในที่นี้จะยกตัวอย่างกรณีสายการประกอบผลิตภัณฑ์ผสม A และ B ที่ขนานกันอยู่ที่มีพนักงานอยู่ 3 ระดับ โดยใช้ปัญหาของ Jackson [10] และ Jaeschke [10] ซึ่งมีจำนวนชิ้นงานทั้งหมด 11 ชิ้นงาน และ 9 ชิ้นงานตามลำดับ โดยสายการประกอบ A และ สายการประกอบ B มีการผลิตสินค้าจำนวน 2 รุ่น ซึ่งมีสัดส่วนการผลิตเป็น 1:1 และ 2:1 ตามลำดับ

ขั้นที่ 1 ทำการรวมรุ่นของผลิตภัณฑ์ในแต่ละสายการประกอบ โดยสามารถแบ่งออกเป็นขั้นตอนย่อยๆ ได้ 2 ขั้นตอนคือ

1. การสร้างแผนภาพความสัมพันธ์ก่อนหลังร่วม (Combined Precedence Diagram) ตามรูปที่ 1

2. การคำนวณหาเวลาชิ้นงานร่วม (Combined Task Time) ในแต่ละชิ้นงานมีค่าดังตารางที่ 4 โดยคำนวณจากสัดส่วนการผลิตของผลิตภัณฑ์ในแต่ละรุ่น ซึ่งบทความนี้จะมีพนักงานอยู่ 3 ระดับ ทำให้วิธีการคำนวณจะต่างจากวิธีของ Sparling และ Miltenburg [11] ซึ่งสามารถคำนวณได้จาก $t_{iw} = \sum_{m=1}^M q_{ml} t_{iwm}$ โดยที่ t_{iwm} คือ เวลาชิ้นงาน i ที่มีพนักงานทักษะ w ($w = 1, 2, 3$) เป็นผู้ปฏิบัติของผลิตภัณฑ์รุ่น m และ $q_{ml} = \frac{D_m}{\sum_{m=1}^M D_m}$ คือ สัดส่วนจำนวนผลิตภัณฑ์รุ่น m ในสายการประกอบที่ l ($l = 1, 2$)



รูปที่ 1 แผนภาพความสัมพันธ์ก่อนหลังร่วม (ก) สายการประกอบผลิตภัณฑ์ A (ข) สายการประกอบผลิตภัณฑ์ B

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561

ตารางที่ 2 ตัวอย่างสตรึงคำตอบของชิ้นงาน

Task id	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	B1	B2	B3	B4	B5	B6	B7	B8	B9
Task priority	3	10	13	9	20	16	12	18	5	17	4	8	19	7	11	1	15	14	2	6

ตารางที่ 3 ตัวอย่างสตรึงคำตอบของพนักงาน

Worker id	H	H	M	M	M	L
Worker priority	6	3	2	1	4	5

ตารางที่ 4 เวลาชิ้นงานร่วมในแต่ละชิ้นงานของพนักงานในแต่ละทักษะ

skill	Task time for each skill worker																			
	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	B1	B2	B3	B4	B5	B6	B7	B8	B9
H	4	3	4	4.5	2	2	3.5	2.5	4	3.5	2.5	4.33	3	2.67	3	2	3.67	1.33	3	4.33
M	5	3	5	6	2	2	4	4	4	5	3	5.33	3	4	4.33	3.33	5	1.67	4.33	5.33
L	7	4.5	-	-	3.5	4.5	5.5	5.5	6	6.5	4.5	7	3	-	4.67	4.67	5.67	2.67	5	5.67

ขั้นที่ 2 กำหนดลำดับของชิ้นงานและพนักงาน ซึ่งมี 2 ขั้นตอนย่อยๆ คือ

1. สร้างลำดับความสำคัญของชิ้นงานและของพนักงานโดยอัลกอริทึมที่ทำการวิจัย
2. กำหนดลำดับของชิ้นงาน โดยพิจารณาจากความสำคัญของชิ้นงานและความสัมพันธ์ก่อนหลังร่วมของผลิตภัณฑ์ และกำหนดลำดับของพนักงานโดยพิจารณาจากลำดับความสำคัญของพนักงาน

ขั้นที่ 3 มอบหมายชิ้นงานและพนักงานให้กับสถานีนงาน ซึ่งมี 2 ขั้นตอนย่อยๆ คือ

1. คำนวณหาค่าขอบเขตล่าง และขอบเขตบนของรอบเวลาดำเนินการจากสมการที่ 1 และ 2 ตามลำดับ

$$C_{LB} = \frac{1}{N_w} \sum_{i=1}^n \min_{1 \leq w \leq 3} \{t_{iw}\} \quad (1)$$

$$C_{UB} = \left(\frac{1}{N_w} \sum_{i=1}^n \max_{1 \leq w \leq 3} \{t_{iw}\} \right) + \max_{1 \leq i \leq n} \{t_{iw}\} \quad (2)$$

2. หารอบเวลาดำเนินการที่เหมาะสม ด้วยวิธีแบ่งครึ่งช่วง (Bisection Searching) คือการหาเวลาดำเนินการโดยการคำนวณหาจุดกึ่งกลางของค่าขอบเขตบนและขอบเขตล่างของรอบเวลาดำเนินการ และทำการจัดสรรชิ้นงานและพนักงานให้กับสถานีนงานด้วยรอบเวลาดำเนินการที่คำนวณได้ ซึ่งมีข้อมูลนำเข้า (Input) อยู่ 8 ตัวแปร คือ 1. จำนวนของสถานีนงานทั้งหมด (N_w)

2. จำนวนชิ้นงานทั้งหมดของทั้ง 2 สายการประกอบ (n)
3. ลำดับของชิ้นงานที่จัดสรรลงสถานีนงาน (Task Sequence : V_2)

4. ลำดับของพนักงานที่จัดสรรลงสถานีนงาน (Worker Allocation : V_3)
5. ขอบเขตล่างของรอบเวลาดำเนินการ (C_{LB})
6. ขอบเขตบนของรอบเวลาดำเนินการ (C_{UB})
7. เวลาของชิ้นงาน i ที่ทำโดยพนักงานทักษะ w (t_{iw})
8. ความต่างระหว่างขอบเขตล่างและขอบเขตบนของรอบเวลาดำเนินการที่กำหนด (Dif_C) ซึ่งเป็นตัวกำหนดจำนวนรอบการทำงานของวิธีแบ่งครึ่งช่วง โดยถ้าค่าต่ำกว่าค่าที่กำหนดจะหยุดการทำงาน และมีขั้นตอนดังนี้

- 1) ให้คำนวณ $C_{UB} - C_{LB}$ ถ้ามากกว่าค่า Dif_C ให้ทำขั้นที่ 2-5 ต่อ แต่ถ้าน้อยกว่าให้หยุดการทำงานทันที

$$2) \text{ คำนวณค่า } C_T \text{ จาก } \frac{C_{LB} + C_{UB}}{2}$$

- 3) เริ่มจัดสรรชิ้นงานและพนักงานตามลำดับที่กำหนด ซึ่งใช้รอบเวลาดำเนินการที่เหมาะสม (C_T) โดยจัดสรรที่ละสถานีนงาน และจะขึ้นสถานีนงานใหม่เมื่อเวลาของชิ้นงานรวมเกินรอบเวลาดำเนินการที่เหมาะสม

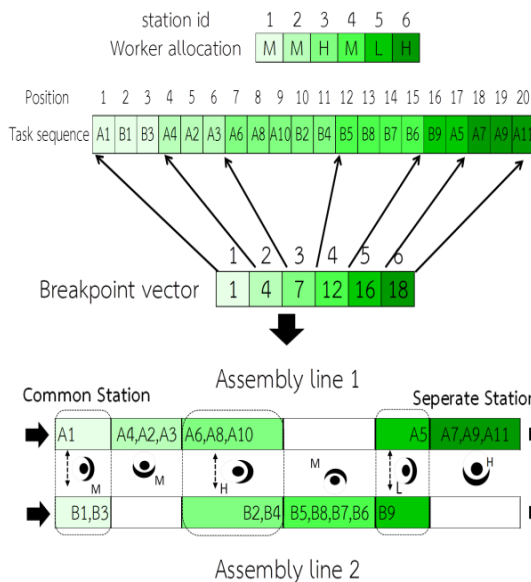
- 4) ปรับค่าขอบเขตบนและขอบเขตล่างของรอบเวลาดำเนินการและเก็บคำตอบตามเงื่อนไขที่กำหนดซึ่งจะไม่เหมือนกับของ Zhang และคณะ [5] ตามตารางที่ 5

- 5) ทำวนข้อที่ 1 -4 จนหยุดการทำงาน

ตารางที่ 5 เงื่อนไขการปรับปรุงขอบเขตล่างและขอบเขตบนของรอบเวลาดำเนินการและการเก็บคำตอบ

เงื่อนไข	สถานีงานหรือพนักงาน ถูกจัดสรรอย่างน้อย 1 ชิ้นงาน	รอบเวลาดำเนินการ พอหรือไม่	การปรับปรุง รอบเวลาดำเนินการ	การเก็บคำตอบ
1	☒	☒	C_{UB}	☒
2	☒	☒	C_{LB}	☒
3	☒	☒	C_{UB}	☒
4	☒	☒	C_{LB}	☒

ขั้นที่ 4 กำหนดผู้ปฏิบัติงานในแต่ละสถานีงานตามลำดับของพนักงานและกำหนดชิ้นงานในแต่ละสถานีงานโดยการถอดรหัสเวกเตอร์จุดแบ่งชิ้นงาน (Breakpoint Vector) ตามรูปที่ 2



รูปที่ 2 การกำหนดชิ้นงานลงในแต่ละสถานีงาน

3. ฟังก์ชันวัตถุประสงค์

ในบทความนี้จะนำเสนอการแก้ปัญหาการจัดสมดุลและการจัดสรรพนักงานหลายทักษะให้กับสถานีงานบนสายการประกอบผลิตภัณฑ์ผสมแบบขนาน ที่มีวัตถุประสงค์จำนวนทั้งสิ้น 4 วัตถุประสงค์ และจะพิจารณาค่าที่เหมาะสมที่สุดไปพร้อมๆ กัน โดยมีรายละเอียดดังนี้

3.1. รอบเวลาดำเนินการที่น้อยที่สุด

เพื่อให้สามารถประกอบผลิตภัณฑ์ต่อชิ้นได้เร็วที่สุดโดยมีค่าเท่ากับเวลารวมของชิ้นงานในแต่ละสถานีงาน

(Total Time of Each Workstation : TT_k) ที่มากที่สุด ดังสมการที่ 3 ซึ่งเป็นวัตถุประสงค์ด้านประสิทธิภาพของสายการประกอบที่สำคัญที่สุดในการจัดสมดุลสายการประกอบภายใต้ปัญหาประเภทที่ 2

$$\text{Minimum } C_T = \max_{1 \leq k \leq N_w} \{TT_k\} \quad (3)$$

เมื่อ N_w คือ จำนวนสถานีงานทั้งหมด

k คือ สัญลักษณ์กำหนดสถานีงาน

3.2. จำนวนสถานีงานน้อยที่สุด

เพื่อให้การจัดสมดุลสายการประกอบนั้นมีขนาดเส้นที่สุด หรือทำให้การสูญเสียพื้นที่ใช้สอยน้อยที่สุด [9] แต่ในงานวิจัยนี้ในบางคำตอบที่ได้นั้นอาจใช้พนักงานที่น้อยกว่ากำหนด ทำให้ต้องมีการปรับการคำนวณจำนวนสถานี โดยคิดจากจำนวนสถานีที่จัดได้จริง (Number of Station: N_s) รวมกับจำนวนสถานีที่ไม่ได้ใช้ (Number of Unused Stations : N_{un}) ตามสมการที่ 4

$$\text{Minimum Station} = N_s + N_{un} \quad (4)$$

3.3. ความแตกต่างของปริมาณภาระงานระหว่างสถานีงานมีค่าน้อยที่สุด

การพยายามทำให้ผลต่างของรอบเวลาดำเนินการกับเวลารวมของชิ้นงานในแต่ละสถานีงานให้มีการกระจายตัวแบบยูนิฟอร์ม [12] ซึ่งชี้วัดโดยค่าความแตกต่างของปริมาณภาระงานระหว่างสถานีงาน (Balancing Workload Between Workstations: B_b) ซึ่ง B_b จะมีค่าอยู่ระหว่าง 0 ถึง 1 โดยค่ายิ่งต่ำจะยิ่งแสดงให้เห็นว่า

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561

การกระจายภาระงานระหว่างสถานีงานมีความสมดุลกันมาก ซึ่งทำให้เกิดความเท่าเทียมกันระหว่างพนักงาน แต่ในงานวิจัยนี้บางคำตอบที่ได้นั้นอาจใช้พนักงานน้อยกว่าที่กำหนดไว้ ซึ่งสามารถคำนวณได้ตามสมการที่ 5

$$\text{Minimise } B_b = \frac{N_w}{N_u - 1} \sum_{k=1}^{N_u} \sum_{b=1}^2 \left(\frac{S_{kb}}{TAD} - \frac{1}{N_u} \right)^2 \quad (5)$$

$$TAD = \sum_{k=1}^{N_u} \sum_{b=1}^2 S_{kb} \quad (6)$$

เมื่อ N_w คือ จำนวนสถานีงานทั้งหมด

N_u คือ จำนวนสถานีงานที่ถูกจัดสรรงาน

k คือ สัญลักษณ์กำหนดสถานีงาน

b คือ สัญลักษณ์กำหนดประเภทของสถานี

เมื่อ $b = 1, 2$ โดย 1 = สถานีงานร่วม

2 = สถานีงานแยก

S_{kb} คือ ค่าเบี่ยงเบนสัมบูรณ์เฉลี่ยระหว่างรอบเวลา

การดำเนินการ (C_T) กับเวลาขึ้นงานรวมที่

สถานีงานที่ k ซึ่งเป็นสถานีงานประเภท b

TAD คือ ผลรวมค่าเบี่ยงเบนสัมบูรณ์ของทุกสถานีงาน

ซึ่งการคำนวณ S_{kb} ขึ้นอยู่กับค่า b ถ้า $b=1$ คือสถานีงานแยก สามารถคำนวณจากสมการที่ 7

$$\begin{aligned} S_{k,1} &= \sum_{m_l=1}^{M_l} q_{m_l} S_{k,1,m_l,l,w} \\ &= \sum_{m_l=1}^{M_l} q_{m_l} |C_T - TT_{k,1,m_l,l,w}| \quad \forall l \in \{1,2\} \quad (7) \end{aligned}$$

โดยที่ q_{m_l} คือสัดส่วนจำนวนผลิตภัณฑ์รุ่น m ที่จะทำการผลิตเทียบกับจำนวนผลิตภัณฑ์ที่จะผลิตทั้งหมดบนสายการประกอบ $l = \{1,2\}$ และ $TT_{k,b,m_l,l,w}$ คือ เวลาขึ้นงานรวมที่สถานีงานที่ k ซึ่งเป็นสถานีงานประเภท b ของผลิตภัณฑ์รุ่น m บนสายการประกอบ l ที่มีพนักงานทักษะ w

และ ถ้า $b = 2$ คือสถานีงานร่วมที่มีการทำงานบนทั้ง 2 สายการประกอบที่อยู่ติดกัน สามารถคำนวณตามสมการที่ 8 เมื่อ m_l และ m_{l+1} เป็นรุ่นของผลิตภัณฑ์ที่ผลิตบนสายการประกอบ l และ $l+1$ ที่ขึ้นงานกันตามลำดับ

$$\begin{aligned} S_{k,2} &= \sum_{m_l=1}^{M_l} \sum_{m_{l+1}=1}^{M_{l+1}} q_{m_l} q_{m_{l+1}} S_{k,2,m_{(l,l+1)},(l,l+1),w} \\ &= \sum_{m_l=1}^{M_l} q_{m_l} q_{m_{l+1}} |C_T - TT_{k,2,m_l,l,w} - \\ &\quad TT_{k,2,m_{l+1},l+1,w}| \quad \forall l \in \{1\} \quad (8) \end{aligned}$$

3.4. ความไม่เกี่ยวข้องกันของงานในแต่ละสถานีงานมีค่าน้อยที่สุด

การทำให้ขึ้นงานภายในสถานีงานมีความเกี่ยวเนื่องกันมากที่สุด จะทำให้การปฏิบัติงานมีประสิทธิภาพที่สูงขึ้นเพราะคิดว่าขึ้นงานที่เกี่ยวข้องกันนั้นจะเป็นงานประเภทเดียวกันซึ่งจะทำให้เกิดเชี่ยวชาญในด้านใดด้านหนึ่ง โดยขึ้นงานสองขึ้นงานใดๆจะถือว่ามีความเกี่ยวเนื่องกันก็ต่อเมื่อมีการเชื่อมโยงกันโดยตรงในแผนภาพความสัมพันธ์ก่อนหลัง [13] แต่เพื่อให้การพิจารณาวัตถุประสงคนี้เป็นไปในทิศทางเดียวกันกับวัตถุประสงค์อื่นๆ ดังนั้น วัตถุประสงคนี้จึงเป็นการทำค่าความไม่เกี่ยวข้องกันภายในสถานีงาน (Index of Task Unrelated :ITUR) มีค่าน้อยที่สุด สามารถคำนวณได้ตามสมการที่ 9 โดยจะมีค่าอยู่ระหว่าง 0 ถึง 1 ซึ่งยิ่งน้อยจะแสดงให้เห็นว่าขึ้นงานภายในสถานีงานมีความเกี่ยวเนื่องกันมาก

เมื่อ SN_k คือจำนวนเครือข่ายของขึ้นงานที่มีความสัมพันธ์ก่อนหลังกันโดยตรงในสถานีงาน แต่ในบทความนี้บางคำตอบที่ออกมานั้นอาจใช้พนักงานที่น้อยกว่าจำนวนที่กำหนด ทำให้ต้องมีดัดแปลงการคำนวณ SN_k โดยสถานีงานที่ไม่ถูกใช้งานจะให้ค่าเท่ากับจำนวนขึ้นงานเฉลี่ยในแต่ละสถานีงาน (Measures the Average Number of Activities per Station : WEST) โดยสามารถคำนวณได้ตามสมการที่ 10 ดังนี้

$$\text{Minimise ITUR} = 1 - \frac{N_u}{\sum_{k=1}^{N_u} SN_k} \quad (9)$$

$$\text{WEST} = \frac{n}{N_w} \quad (10)$$

ซึ่งหลักในการเลือกวัตถุประสงค์ของการจัดสมดุลและการจัดสรรพนักงานหลายทักษะบนสายการประกอบ

ผลิตภัณฑ์ผสมแบบขนาน (ประเภทที่ 2) สำหรับบทความนี้จะเลือกจากวัตถุประสงค์ที่มีความสัมพันธ์เชิงลบ คือมีคำตอบที่เป็นไปในทิศทางตรงกันข้าม และจะต้องมีทั้งวัตถุประสงค์ด้านประสิทธิภาพของสายการประกอบ (วัตถุประสงค์ที่ 1 2 และ 4) และวัตถุประสงค์ด้านสมดุลของภาระงาน (วัตถุประสงค์ที่ 3)

4. วิธีการเชิงวิวัฒนาการแบบหลายวัตถุประสงค์โดยยึดหลักการจำแนก (MOEA/D)

บทความนี้จะนำเอาวิธี MOEA/D ที่พัฒนาโดย Zhang และ Li [8] มาประยุกต์ใช้กับการแก้ปัญหาการจัดสมดุลและจัดสรรพนักงานหลายทักษะบนสายการประกอบ ผลิตภัณฑ์ผสมแบบขนานที่มากวัตถุประสงค์ ภายใต้ปัญหาประเภทที่ 2 ซึ่งมีแนวคิดในการแก้ปัญหาโดยการพัฒนาสตริงคำตอบแบบมาววัตถุประสงค์ไปพร้อมๆกัน โดยกำหนดค่าถ่วงน้ำหนักบนจุด p ของแต่ละวัตถุประสงค์ i (λ_{pi}) ให้กับแต่ละจุดของค่าถ่วงน้ำหนัก (Point of Weight : λ_p) พร้อมกับกำหนดจุดข้างเคียง (Neighborhood Point) เพื่อใช้ในการระบุหรือจำแนกปัญหาย่อย

การพัฒนาสตริงคำตอบของปัญหาย่อยในแต่ละเจเนอเรชันจะใช้การสลับสายพันธุ์ (Crossover) ซึ่งเป็นวิธีเชิงพันธุกรรม (Genetic Algorithm) และใช้วิธีเทบิเชฟฟ์ (Tchebycheff Approach) ในการคัดเลือกสตริงคำตอบแบบมาววัตถุประสงค์ และเมื่อครบตามเจเนอเรชันที่กำหนดจะนำสตริงคำตอบจากเจเนอเรชันสุดท้าย มาคัดเลือกหากลุ่มคำตอบที่ดีที่สุด (1^{st} Frontier) โดยใช้การกำหนดค่าความแข็งแรง (Fitness Assignment) ด้วยการจัดลำดับแบบไม่ถูกครอบงำ (Non-dominated sorting)

4.1. ขั้นตอนการทำงานของ MOEA/D

การแก้ปัญหาการจัดสมดุลและจัดสรรพนักงานหลายทักษะบนสายการประกอบผลิตภัณฑ์ผสมแบบขนานที่มากวัตถุประสงค์ ภายใต้ปัญหาประเภทที่ 2 ด้วยอัลกอริทึม MOEA/D ซึ่งมีแผนภูมิการไหลตามรูปที่ 3

ในที่นี้จะยกตัวอย่างการแก้ปัญหาการจัดสมดุลและการจัดสรรพนักงานหลายทักษะบนสายการประกอบ ผลิตภัณฑ์ผสมแบบขนาน ที่มี 4 ฟังก์ชันวัตถุประสงค์ และมีชิ้นงานของทั้ง 2 สายการประกอบเท่ากับ 20 ชิ้นงาน โดยมีพนักงาน 6 คน (สถานีงาน 6 สถานีงาน) ซึ่งมีพารามิเตอร์และขั้นตอนการทำงานดังนี้

พารามิเตอร์ของ MOEA/D ที่เลือกใช้ คือ

- จำนวนส่วนในการแบ่งค่าถ่วงน้ำหนักในแต่ละฟังก์ชันค่าวัตถุประสงค์ (H) เท่ากับ 2
 - จำนวนของจุดข้างเคียง (T) เท่ากับ 4
 - ร้อยละของจำนวนในการสุ่มเลือกจุดข้างเคียงในประชากรย่อยที่ทำการพิจารณาเท่ากับ 40
 - จำนวนวัตถุประสงค์ทั้งหมด (O) เท่ากับ 4
- ดังนั้นจำนวนจุดของค่าถ่วงน้ำหนักทั้งหมดหรือประชากร (P) คำนวณได้เท่ากับ 10 ดังสมการที่ 12

$$\lambda Set = C_{O-1}^{H+O-1} = C_{4-1}^{2+4-1} = 10 \quad (12)$$

ขั้นที่ 1 สร้างค่าถ่วงน้ำหนัก λ_{pi} ด้วยซิมเพล็กซ์แลตทิซดีไซน์ โดยใช้พารามิเตอร์ที่กำหนด จะได้ค่าตามตารางที่ 6 ซึ่ง $\sum_{i=1}^O \lambda_{pi} = 1 \forall p \in P$ และ $\lambda_{pi} \geq 0$ จะเห็นได้ว่าค่าถ่วงน้ำหนักในตารางนั้นจะมีเพียง 3 ค่า เนื่องจากค่าถ่วงน้ำหนักของแต่ละวัตถุประสงค์จะมีวิธีคิดจาก $\{0/H, \dots, H/H\}$ โดยจะมีค่าเท่ากับ 0 0.5 และ 1

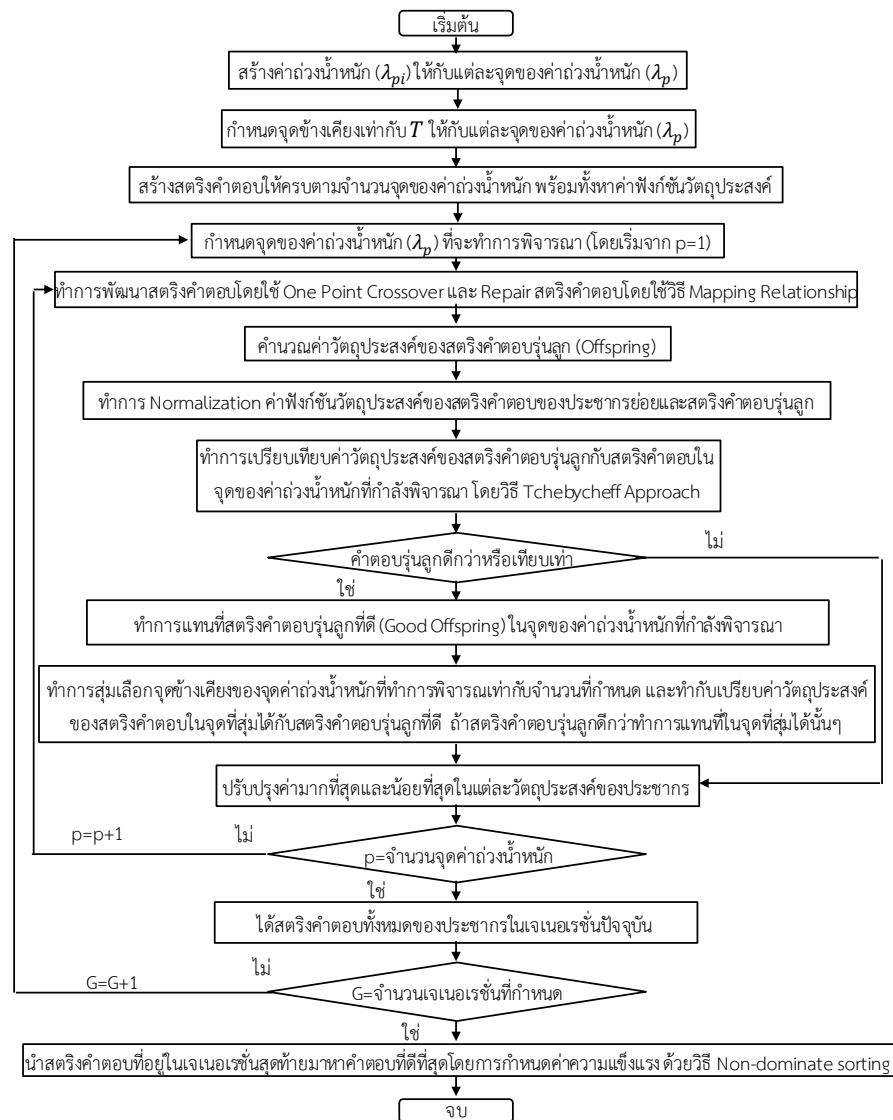
ขั้นที่ 2 เลือกจุดข้างเคียงให้กับแต่ละจุดค่าถ่วงน้ำหนัก โดยพิจารณาจากระยะห่างระหว่างจุดของค่าถ่วงน้ำหนักที่สนใจ (λ_p) กับจุดของค่าถ่วงน้ำหนักอื่นๆ (λ_j) ที่คำนวณจากสมการที่ 13

$$d_{pj} = \sqrt{\sum_{i=1}^O (\lambda_{pi} - \lambda_{ji})^2} \quad (13)$$

$$\forall p \in P, \forall j \in P, p \neq j$$

และเลือกจุดข้างเคียงจากค่าระยะห่างที่น้อยที่สุดจนครบตามจำนวนที่กำหนด ตามตารางที่ 7 ถ้าระยะห่างมีค่าเท่ากันและไม่สามารถเลือกได้หมดทุกจุด ให้สุ่มเลือกแบบอิสระ

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561



รูปที่ 3 ขั้นตอนการทำงาน MOEA/D

ตารางที่ 6 ค่าถ่วงน้ำหนัก λ_{pi} ในแต่ละจุดค่าถ่วงน้ำหนัก

λ_p	λ_{p1}	λ_{p2}	λ_{p3}	λ_{p4}
1	0.5	0	0	0.5
2	0	0	0	1
3	0	0	0.5	0.5
4	0	0.5	0.5	0
5	0	1	0	0
6	0	0	1	0
7	0.5	0.5	0	0
8	1	0	0	0
9	0.5	0	0.5	0
10	0	0.5	0	0.5

ขั้นที่ 3 สร้างสตริงคำตอบเริ่มต้นของชิ้นงานและพนักงานให้กับทุกจุดค่าถ่วงน้ำหนักโดยการสุ่มอย่างอิสระ พร้อมทั้งหาค่าฟังก์ชันวัตถุประสงค์ตามตารางที่ 8 และหาค่าน้อยที่สุดและมากที่สุดในแต่ละค่าฟังก์ชันวัตถุประสงค์

ตารางที่ 7 ค่าถ่วงน้ำหนักข้างเคียงของแต่ละจุดค่าถ่วงน้ำหนัก λ_p (ค่า T เท่ากับ 4)

T	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_9	λ_{10}
1	λ_2	λ_1	λ_2	λ_3	λ_3	λ_3	λ_1	λ_1	λ_1	λ_1
2	λ_3	λ_3	λ_4	λ_5	λ_4	λ_4	λ_4	λ_4	λ_6	λ_2
3	λ_7	λ_7	λ_6	λ_6	λ_7	λ_7	λ_5	λ_7	λ_7	λ_5
4	λ_9	λ_{10}	λ_9	λ_{10}	λ_9	λ_{10}	λ_8	λ_9	λ_8	λ_7

ขั้นที่ 4 กำหนดจุดของค่าถ่วงน้ำหนักที่ทำการพิจารณา ด้วยการกำหนดค่า p โดยค่า $p = \{1, \dots, 10\}$ เพื่อสร้างกลุ่มประชากรย่อย ตามตารางที่ 9

ขั้นที่ 5 ทำการสุ่มเลือก 2 สตริงคำตอบจากสตริงคำตอบที่อยู่ในประชากรย่อย โดยสตริงคำตอบนั้นต้องไม่ซ้ำกัน ซึ่งงานวิจัยนี้ต้องสุ่มทั้งสิ้น 4 สตริงคำตอบโดยแบ่งออกเป็นสตริงคำตอบของชิ้นงาน 2 สตริง และสตริงคำตอบของพนักงาน 2 สตริง

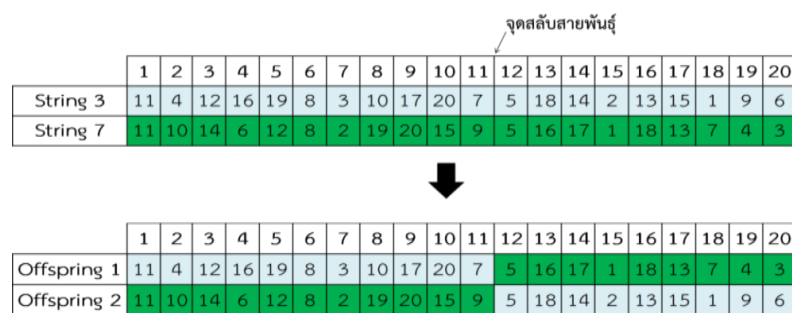
ขั้นที่ 6 ทำการพัฒนาสตริงคำตอบที่สุ่มได้โดยการการสลับสายพันธุระหว่าง 2 สตริงคำตอบแบบ 1 ตำแหน่ง (One-point Crossover) ซึ่งจะได้สตริงคำตอบรุ่นลูก (Offspring) 2 สตริง ตามรูปที่ 4 และทำการซ่อมแซมสตริงคำตอบรุ่นลูก ด้วยวิธี Mapping Relationship ของ Goldberg และ Ling [14] ตามรูปที่ 5 เพื่อไม่ให้ลำดับความสำคัญในสตริงคำตอบมีค่าซ้ำกัน โดยจะต้องทำทั้งสตริงคำตอบของชิ้นงานและพนักงาน

ตารางที่ 8 สตริงคำตอบเริ่มต้นของชิ้นงานและพนักงานของจุดค่าถ่วงน้ำหนักทั้งหมดพร้อมทั้งค่าฟังก์ชันวัตถุประสงค์

λ_p	String	ชิ้นงาน 20 ชิ้นงาน	พนักงาน 6 คน	f_1	f_2	f_3	f_4
λ_1	1	[3 10 13 9 20 16 12 18 5 17 4 8 19 7 11 1 15 14 2 6]	[6 3 2 1 4 5]	14.333	6	0.146	0.538
λ_2	2	[2 3 18 11 20 19 8 14 13 15 4 12 9 5 16 17 7 1 6 10]	[1 6 2 4 3 5]	15.833	6	0.067	0.538
λ_3	3	[11 4 12 16 19 8 3 10 17 20 7 5 18 14 2 13 15 1 9 6]	[5 6 3 1 4 2]	15.333	6	0.208	0.538
λ_4	4	[4 8 6 1 2 20 5 14 15 3 12 18 20 13 7 11 16 9 17 19]	[3 4 5 6 2 1]	14.833	5	0.165	0.571
λ_5	5	[5 11 1 17 14 12 9 3 18 8 7 15 4 20 10 2 19 16 6 13]	[3 6 1 4 2 5]	15.333	6	0.121	0.455
λ_6	6	[3 15 17 9 8 11 2 14 1 4 19 7 13 5 10 6 16 12 20 18]	[3 6 2 1 4 5]	15.333	5	0.164	0.500
λ_7	7	[11 10 14 6 12 8 2 19 20 15 9 5 16 17 1 18 13 7 4 3]	[2 3 1 5 4 6]	15.500	6	0.062	0.571
λ_8	8	[5 18 15 11 20 13 3 4 10 19 9 7 1 17 8 12 16 14 6 2]	[6 4 2 1 3 5]	15.000	6	0.030	0.455
λ_9	9	[14 8 7 15 2 13 6 10 17 12 5 3 9 18 16 19 20 11 1 4]	[3 2 5 6 4 1]	15.333	5	0.145	0.500
λ_{10}	10	[2 15 9 12 10 13 8 6 7 3 18 5 4 17 1 20 19 16 11 14]	[3 1 5 2 4 6]	15.333	6	0.132	0.500

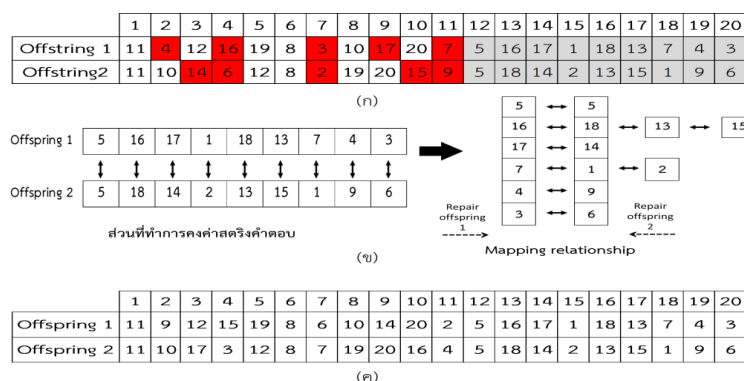
ตารางที่ 9 กลุ่มประชากรย่อยที่จะทำการพัฒนาคำตอบ เมื่อ p เท่ากับ 1

λ_p	$\{\lambda_{p1}, \lambda_{p2}, \lambda_{p3}, \lambda_{p4}\}$	String	ชิ้นงาน 20 ชิ้นงาน	พนักงาน 6 คน
λ_1	[0.5 0 0 0.5]	1	[3 10 13 9 20 16 12 18 5 17 4 8 19 7 11 1 15 14 2 6]	[6 3 2 1 4 5]
λ_2	[0 0 0 1]	2	[11 4 12 16 19 8 3 10 17 20 7 5 18 14 2 13 15 1 9 6]	[5 6 3 1 4 2]
λ_3	[0 0 0.5 0.5]	3	[4 8 6 1 2 20 5 14 15 3 12 18 20 13 7 11 16 9 17 19]	[3 4 5 6 2 1]
λ_7	[0.5 0.5 0 0]	7	[3 15 17 9 8 11 2 14 1 4 19 7 13 5 10 6 16 12 20 18]	[3 6 2 1 4 5]
λ_9	[0 0.5 0 0.5]	9	[11 10 14 6 12 8 2 19 20 15 9 5 16 17 1 18 13 7 4 3]	[2 3 1 5 4 6]



รูปที่ 4 ขั้นตอนพัฒนาสตริงคำตอบโดยวิธี One-point Cross Over

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561



รูปที่ 5 ขั้นตอนการซ่อมแซมสตริงคำตอบ (ก) การสุ่มช่วงที่จะทำการซ่อมแซม
(ข) วิธีสร้าง Mapping relationship (ค) สตริงคำตอบหลังซ่อมแซม

ขั้นที่ 7 คำนวณค่าฟังก์ชันวัตถุประสงค์ของสตริงคำตอบรุ่นลูก (Offspring) ทั้ง 2 สตริงคำตอบที่ทำการซ่อมแซมเป็นที่เรียบร้อยแล้ว และทำการปรับปรุงค่ามากที่สุดและน้อยที่สุดในแต่ละวัตถุประสงค์ของประชากรโดยเทียบค่าในแต่ละวัตถุประสงค์ของสตริงคำตอบรุ่นลูก (Offspring) กับค่ามากที่สุดและน้อยที่สุดในแต่ละวัตถุประสงค์เดิมว่ามีการเปลี่ยนแปลงหรือไม่

ขั้นที่ 8 ทำการนอร์มัลไล (Normalization) ค่าฟังก์ชันวัตถุประสงค์ของสตริงคำตอบของกลุ่มประชากรย่อยและสตริงคำตอบรุ่นลูก (Offspring) ด้วยสมการที่ 14

$$\bar{f}_i = \frac{f_i - z_i^*}{z_i^{nad} - z_i^*} \quad (14)$$

เมื่อ z_i^* คือ $\min\{f_i(x) | x \in P\}$

z_i^{nad} คือ $\max\{f_i(x) | x \in P\}$

โดยที่ $z_i^* = (z_1^*, \dots, z_0^*), z_i^{nad} = (z_1^{nad}, \dots, z_0^{nad})$

0 คือ จำนวนวัตถุประสงค์ทั้งหมด

ขั้นที่ 9 ทำการเปรียบเทียบค่าฟังก์ชันวัตถุประสงค์ของสตริงคำตอบรุ่นลูก (Offspring) กับค่าฟังก์ชันวัตถุประสงค์ที่อยู่ในจุดค่าถ่วงน้ำหนัก λ_p ที่กำลังพิจารณาโดยการกำหนดค่าความแข็งแรง (Fitness Assignment) ด้วยวิธีเทบิเชฟฟ์ (Tchebycheff Approach) ที่สามารถคำนวณจากสมการที่ 15 เมื่อ z_i^* คือ ค่าเป้าหมายของวัตถุประสงค์ i โดย $i = 1, \dots, O$

$$g(x|\lambda_p, z^*) = \max_{1 \leq i \leq O} \{\lambda_{pi} | f_i(x) - z_i^* | \} \quad (15)$$

โดยที่ $(x|\lambda_p, z^*)$ คือ ผลต่างที่มีค่ามากที่สุดของค่าฟังก์ชันวัตถุประสงค์ i กับค่าเป้าหมายของวัตถุประสงค์ i ที่ทำการคูณค่าถ่วงน้ำหนักบนจุด p ของวัตถุประสงค์

โดยมีขั้นตอนการเปรียบเทียบและแทนที่ดังต่อไปนี้

1. นำค่าฟังก์ชันวัตถุประสงค์ของสตริงคำตอบรุ่นลูก (Offspring) กับสตริงคำตอบที่อยู่ในจุดค่าถ่วงน้ำหนัก λ_p ที่กำลังพิจารณาและผ่านการนอร์มัลไลแล้วในแต่ละวัตถุประสงค์ไปคูณกับค่าถ่วงน้ำหนักของแต่ละวัตถุประสงค์ในจุดค่าถ่วงน้ำหนัก λ_p ที่กำลังพิจารณา

2. หาค่าฟังก์ชันเทบิเชฟฟ์ $g(x|\lambda_p, z^*)$ โดยที่ $g(x|\lambda_p, z^*) = \max_{1 \leq i \leq 4} \{\lambda_{pi} \times \bar{f}_i\}$ ของสตริงคำตอบรุ่นลูก (offspring) ทั้ง 2 สตริง และสตริงคำตอบที่อยู่ในจุดค่าถ่วงน้ำหนัก λ_p ที่กำลังพิจารณา

3. พิจารณาค่าฟังก์ชันเทบิเชฟฟ์ $g(x|\lambda_p, z^*)$ ในข้อที่ 2 ของสตริงคำตอบรุ่นลูก (Offspring) ทั้ง 2 สตริง กับสตริงคำตอบที่อยู่ในจุดค่าถ่วงน้ำหนัก λ_p ที่กำลังพิจารณาว่าดีกว่าหรือไม่ ในบทความนี้คำว่าดีกว่าคือคำตอบนั้นต้องมีค่าน้อยกว่าหรือเทียบเท่า ถ้าดีกว่าทุกสตริงคำตอบให้เลือกสตริงคำตอบที่ดีที่สุด แต่ถ้ามีค่าเท่ากันให้สุ่มเลือกแบบอิสระ ตามตารางที่ 10

ตารางที่ 10 การพิจารณาคำตอบที่ดีที่สุด

λ_p	String	$g(x \lambda_p, z^*)$	Selection
λ_1	1	1	
λ_1	Offspring 1	1 (เท่ากับ)	Good
λ_1	Offspring 2	0 (น้อยกว่า)	Best

4. ทำการแทนที่สตริงคำตอบและค่าฟังก์ชันวัตถุประสงค์ของสตริงคำตอบรุ่นลูกที่ดีที่สุด (Best Offspring) ลงในจุดค่าถ่วงน้ำหนัก (λ_p) ที่กำลังพิจารณาตามตารางที่ 11 และทำขั้นที่ 10 แต่ถ้าไม่ดีกว่า จะไม่เกิดการแทนที่สตริงคำตอบและให้ข้ามไปขั้นที่ 11 เลย

ขั้นที่ 10 จะทำการคำนวณจำนวนจุดข้างเคียงที่ต้องสุ่ม จากพารามิเตอร์ร้อยละของจำนวนในการสุ่มเลือกจุดข้างเคียงในกลุ่มประชากรย่อย จากนั้นสุ่มเลือกจุดข้างเคียงในกลุ่มประชากรย่อยที่กำลังพิจารณาเท่ากับจำนวนที่คำนวณได้ หลังจากนั้นทำการเปรียบเทียบเหมือนกับขั้นที่ 9 แต่จะทำการเปรียบเทียบค่าวัตถุประสงค์ของสตริงคำตอบในจุดที่สุ่มได้กับค่าวัตถุประสงค์ของสตริงคำตอบรุ่นลูกที่ดี ถ้าสตริงคำตอบรุ่นลูกดีกว่าตามตารางที่ 12 จะทำการแทนที่สตริงคำตอบและค่าฟังก์ชันวัตถุประสงค์ของสตริงคำตอบรุ่นลูก (Offspring) ที่ดีในจุดค่าถ่วงน้ำหนักที่สุ่มได้นั้นๆ ทำการปรับปรุงค่ามากที่สุดและน้อยสุดในแต่ละวัตถุประสงค์ของประชากรทั้งหมด

ขั้นที่ 11 พิจารณาว่า $p = P$ หรือไม่ ถ้าไม่เท่า ให้กำหนด $p = p + 1$ เพื่อเปลี่ยนไปพัฒนาสตริงคำตอบในปัญหาย่อยถัดไป โดยกลับไปทำซ้ำในขั้นที่ 4 แต่ถ้าเท่ากัน ให้ไปทำในขั้นที่ 12 ต่อ

ขั้นที่ 12 เมื่อทำการพัฒนาสตริงคำตอบครบตามจำนวนปัญหาย่อยจะได้รับสตริงคำตอบทั้งหมดของประชากรในเจเนอเรชันปัจจุบัน ต่อจากนั้นพิจารณาว่า $G = Genaretion$ หรือไม่ ถ้าไม่ให้กำหนด $G = G + 1, p = 1$ และกลับไปทำซ้ำในขั้นที่ 4 แต่ถ้า $G = Genaretion$ ให้ข้ามไปทำในขั้นที่ 13 ต่อ

ขั้นที่ 13 นำสตริงคำตอบทั้งหมดในเจเนอเรชันสุดท้ายมาหาคำตอบที่ดีที่สุดโดยการกำหนดค่าค่าแข็งแรง ด้วยการการจัดลำดับแบบไม่ถูกครอบงำ ซึ่งจะถือว่าเป็นค่าแข็งแรงไม่แท้จริง (Dummy Fitness Value) หรือการเปรียบเทียบค่าฟังก์ชันในแต่ละวัตถุประสงค์ว่าถูกครอบงำหรือไม่ คำว่าถูกครอบงำคือทุกค่าฟังก์ชันวัตถุประสงค์เป็นคำตอบที่แย่กว่าหรือเทียบเท่า ไม่มีค่าฟังก์ชันวัตถุประสงค์ไหนเลยที่ดีกว่า โดยสตริงคำตอบนั้นจะถือว่าเป็นสตริงคำตอบที่ถูกครอบงำและอยู่ลำดับที่สูงกว่า โดยคำตอบที่ดีที่สุดจะอยู่ลำดับที่ 1 (1st Frontier) เสมอ แต่ถ้าคำตอบนั้นไม่ถูกครอบงำจะถูกจัดอยู่ในลำดับเดียวกันตามตารางที่ 13

ตารางที่ 11 การแทนที่สตริงคำตอบและฟังก์ชันวัตถุประสงค์รุ่นลูกสตริงที่ 1 (Offspring 2) ในจุดค่าถ่วงน้ำหนัก λ_1

λ_p	String	ชิ้นงาน 20 ชิ้นงาน	พนักงาน 6 คน	f_1	f_2	f_3	f_4
λ_1	Off 2	[11 10 17 3 12 8 7 19 20 16 4 5 18 14 2 13 15 1 9 6]	[3 4 5 1 6 2]	16.83	5	0.092	0.68
λ_2	2	[11 4 12 16 19 8 3 10 17 20 7 5 18 14 2 13 15 1 9 6]	[5 6 3 1 4 2]	15.33	6	0.208	0.53
λ_3	3	[4 8 6 1 2 20 5 14 15 3 12 18 20 13 7 11 16 9 17 19]	[3 4 5 6 2 1]	14.83	5	0.165	0.57
λ_7	7	[3 15 17 9 8 11 2 14 1 4 19 7 13 5 10 6 16 12 20 18]	[3 6 2 1 4 5]	15.33	5	0.164	0.50
λ_9	9	[11 10 14 6 12 8 2 19 20 15 9 5 16 17 1 18 13 7 4 3]	[2 3 1 5 4 6]	15.50	6	0.062	0.57

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561

ตารางที่ 12 การเปรียบเทียบโดยใช้จุดค่าถ่วงน้ำหนักที่สุ่มได้ (สมมติว่าสุ่มได้ λ_3)

λ_p	String	$g(x \lambda_p, z^*)$	Selection
λ_3	3	1	
λ_3	Offspring 2	1 (เทียบเท่า)	Good

ตารางที่ 13 ตัวอย่างการคิดค่าแข็งแรง ด้วยการการจัดลำดับแบบไม่ถูกครอบงำ

String	Objective				Dummy Fitness
	f_1	f_2	f_3	f_4	
Offspring(1.1)	14.333	6	0.066	0.538	2
Offspring(2.1)	14.333	6	0.085	0.538	3
String 3	14.000	5	0.084	0.538	1
Offspring(4.1)	14.333	5	0.057	0.500	1

ตารางที่ 14 ปัญหาตัวอย่างที่ใช้ในการทดลอง

No.	% Unavailable	Line1			Line2			Total Worker	Worker ratio (High : Medium : Low)
		Problem	Model	MPS	Problem	Model	MPS		
P1	5-15	Buxey 29	2	1:3	Sawyer 30	2	2:3	13	(4:8:1) , (2:8:3)
								18	(5:11:2) , (3:11:4)
P2	5-15	Tonge 70	3	2:3:1	Warnecke 58	3	2:1:1	20	(6:13:1) , (5:13:2)
								29	(9:19:1) , (6:19:4)
P3	5-15	Arcus 83	3	2:1:2	Arcus 111	3	1:1:3	33	(10:21:2) , (7:21:5)
								48	(14:31:3) , (12:31:5)
P4	20-30	Scholl 297	2	1:1	Arcus 83	3	1:1:3	30	(9:20:1) , (7:20:3)
								43	(13:28:2) , (11:28:4)

5. การทดลองทางคอมพิวเตอร์

5.1. การออกแบบการทดลอง

เพื่อเป็นการประเมินสมรรถนะของ MOEA/D ในการแก้ปัญหาการจัดสมมูลที่นำเสนอ ผู้วิจัยได้ทดลองใช้ MOEA/D ในการแก้ปัญหาตัวอย่างจำนวนทั้งสิ้น 4 ปัญหา โดย P1 จะเป็นปัญหขนาดเล็ก P2 จะเป็นปัญหขนาดกลาง ส่วน P3 และ P4 จะเป็นปัญหขนาดใหญ่ แต่มีความแตกต่างที่ร้อยละจำนวนของชิ้นงานที่พนักงานทักษะต่ำไม่สามารถทำงานได้ ซึ่งในแต่ละปัญหาแบ่งออกเป็น 4 ปัญหาย่อยตามจำนวนสถานีนงาน และอัตราส่วนของพนักงานในแต่ละทักษะ ดังตารางที่ 14 (รวมทั้งสิ้น มี 16 ปัญหาย่อย) ซึ่งจะทำการเปรียบเทียบกับ อัลกอริทึมการบรรจบ (Combinatorial Optimization with Coincidence : COIN) พัฒนาโดย ปารเมศ ชูติมา และคณะ [15] ซึ่งเป็นอีกอัลกอริทึมหนึ่ง

ที่นิยมใช้และมีประสิทธิภาพในการหาคำตอบที่แท้จริงสำหรับการแก้ปัญหาการจัดสมมูลสายการประกอบแบบหลายวัตถุประสงค์ที่ดีกว่าอัลกอริทึมหลายอัลกอริทึม [15],[16]ซึ่งเป็นอัลกอริทึมที่ได้รับความนิยม และมีการนำไปใช้การแก้ปัญหาการจัดลำดับการผลิตรถยนต์แบบหลายวัตถุประสงค์อีกด้วย [17]

เพื่อให้แน่ใจว่าอัลกอริทึมทั้งหมดที่ทำการศึกษาคำตอบได้ทำงานตามสมรรถนะที่ดีที่สุด ดังนั้นพารามิเตอร์ต่างๆของแต่ละอัลกอริทึมจะถูกกำหนดอย่างมีหลักการโดยอ้างอิงจากงานวิจัยที่ผ่านมา เช่น จำนวนประชากรที่ใช้ในการทดลองเท่ากับ 100 ประชากร [18] แต่ในอัลกอริทึม MOEA/D จำนวนของประชากรจะขึ้นอยู่กับจำนวนของวัตถุประสงค์และสัดส่วนในการแบ่งค่าถ่วงน้ำหนัก จึงไม่สามารถกำหนดให้ค่าเท่ากับ 100 ได้ ดังนั้นจึงใช้ $120 (C_{O-1}^{H+O-1}, O = 4, H = 7)$ ทั้ง 2 อัลกอริทึม

ส่วนพารามิเตอร์อื่น เช่น 1. จำนวนของจุดข้างเคียง (MOEA/D) 2. ร้อยละของจำนวนในการสุ่มเลือกจุดอื่นในประชากรย่อยที่ทำการพิจารณา (MOEA/D) 3. ค่าความน่าจะเป็นในการให้รางวัลและลงโทษ (COIN) 4. ร้อยละในการกำหนดจำนวนสตริงคำตอบที่จะนำมาปรับปรุงตารางความน่าจะเป็น (COIN) 5. จำนวนเจเนอเรชัน (MOEA/D และ COIN) จะผ่านการวิเคราะห์ผลทางสถิติเพื่อให้ได้ค่าที่เหมาะสมตามตารางที่ 15

ตารางที่ 15 พารามิเตอร์ของอัลกอริทึม

Population size		120
Generation		P1 = 500 , P2 P3 = 1500, P4 = 2000
MOEA/D	Neighborhood	P1 P3 P4 : 4 , P2 : 9
	%Replace neighbor	P1 P3 P4 : 0.3 , P2 : 0.5
COIN	Reward/Punish	P1- P3 : 0.2 , P4 : 0.3
	Selection rate	P1 -P4 : 0.5

5.2. ตัวชี้วัดสมรรถนะของอัลกอริทึม

การเปรียบเทียบสมรรถนะในการหาคำตอบระหว่าง MOEA/D กับ COIN นั้นจะพิจารณาค่าฟังก์ชันวัตถุประสงค์ไปพร้อมกันๆ ซึ่งแต่ละอัลกอริทึมอาจค้นพบคำตอบที่ดีที่สุดหลายคำตอบ การจะเปรียบเทียบว่ากลุ่มคำตอบของอัลกอริทึมใดดีกว่าจึงต้องอาศัยตัวชี้วัดตามรูปแบบของปัญหาแบบมัลติวัตถุประสงค์ โดยตัวชี้วัดสมรรถนะที่ใช้ในงานวิจัยนี้จะอ้างอิงมาจาก [19],[20] รวมทั้งหมด 6 ตัวชี้วัด โดยจะเรียงตามลำดับความสำคัญจากมากไปหาน้อย คือ

1. การลู่เข้าสู่กลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริง (Convergence) เป็นตัวชี้วัดที่ประเมินว่ากลุ่มคำตอบในแต่ละอัลกอริทึมที่ได้มีความใกล้เคียงกับกลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริง (True Pareto) มากน้อยเพียงใด ซึ่งถือว่าเป็นตัวชี้วัดที่สำคัญที่สุดในบทความนี้

2. การกระจายตัว (Spread) เป็นตัวชี้วัดในการประเมินกลุ่มคำตอบที่ได้มีการกระจายที่สม่ำเสมอเพียงใด

3. อัตราส่วนของคำตอบที่ไม่ถูกรอรับเทียบกับกลุ่มคำตอบที่อัลกอริทึมหาค่าได้ (R_{NDS1}) ซึ่งใช้แสดงว่าในคำตอบที่ค้นพบทั้งหมดเป็นคำตอบที่อยู่ในคำตอบที่เหมาะสมที่สุดที่แท้จริงคิดเป็นอัตราส่วนเท่าใดเมื่อเทียบกับคำตอบที่หามาได้

4. อัตราส่วนของคำตอบที่ไม่ถูกรอรับเทียบกับกลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริง (R_{NDS2}) ซึ่งใช้แสดงว่าในคำตอบที่ค้นพบทั้งหมดเป็นคำตอบที่อยู่ในคำตอบที่เหมาะสมที่สุดที่แท้จริงคิดเป็นอัตราส่วนเท่าใดเมื่อเทียบกับคำตอบที่ดีที่สุดที่แท้จริง

5. จำนวนคำตอบที่ไม่ถูกรอรับ (N_{NDS}) คือจำนวนคำตอบที่ดีที่สุดที่เหมาะสมที่สุด (1^{st} Frontier) ในแต่ละอัลกอริทึม

6. เวลาในการค้นหาคำตอบ (CPU Time)

5.3. ผลการทดลอง

ในการทดลองแก้ปัญหาการจัดสมมูลและจัดสรรพนักงานหลายทักษะบนสายการประกอบผลิตภัณฑ์ผสมลักษณะขนานแบบมัลติวัตถุประสงค์ภายใต้ปัญหาประเภทที่ 2 โดยการทำให้โจทย์ละ 3 รอบนั้น เมื่อได้คำตอบที่เหมาะสมที่สุดเชิงพาเรโตของแต่ละอัลกอริทึมแล้ว จะนำคำตอบทั้งหมดของทั้ง 2 อัลกอริทึมในโจทย์ตัวอย่างนั้นมาทำการเปรียบเทียบตัวชี้วัดสมรรถนะด้านต่างๆ โดยกำหนดค่าความแข็งแกร่งร่วมกันด้วยวิธี Non-Dominated Sorting โดยจะถือว่ากลุ่มคำตอบที่มีค่าความแข็งแกร่งเท่ากับ 1 (1^{st} Frontier) เป็นกลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริงและทำการประเมินอัลกอริทึมด้วยตัวชี้วัดที่กำหนดไว้เพื่อเปรียบเทียบสมรรถนะของอัลกอริทึมนั้นๆ

การทดลองแก้ปัญหาตัวอย่างเป็นการทดลองผ่านโปรแกรมที่เขียนภาษา C# ประมวลผลด้วยคอมพิวเตอร์ Intel® Core™ i7-4710HQ CPU@2.50GHz RAM 8.00 GB โดยผลการทดลองที่ได้อยู่ในตารางที่ 16 พบว่า MOEA/D สามารถค้นพบกลุ่มคำตอบที่มีค่าต่ำที่สุดที่ต่ำกว่า (ดีกว่า) หรือเทียบเท่าของทุกฟังก์ชันวัตถุประสงค์ ในเกือบทุกปัญหา ยกเว้น ปัญหา P4 กรณีคำนวณทั้งหมดเท่ากับ 43 ซึ่งค่าฟังก์ชันวัตถุประสงค์ที่ 1 (C_7) ที่สูงกว่า

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561

(แย่กว่า) และ ปัญหา P2 กรณีคนงานทั้งหมดเท่ากับ 20 ซึ่งค่าฟังก์ชันวัตถุประสงค์ที่ 3 (B_b) ที่สูงกว่า

ดังนั้นจะเห็นได้ว่าการหาคำตอบเบื้องต้นของ MOEA/D นั้นดีกว่า COIN ซึ่งจะสอดคล้องกับตัวชี้วัด Convergence ของ MOEA/D มีค่าน้อยกว่า (ยิ่งเข้าใกล้ 0 ยิ่งดี) ซึ่งหมายความว่ากลุ่มคำตอบที่ได้จาก MOEA/D มีความใกล้เคียงกับกลุ่มคำตอบที่เหมาะสมที่สุดที่แท้จริงมากกว่า และไปทิศทางเดียวกับตัวชี้วัด R_{NDS1} และ R_{NDS2} ของ MOEA/D ที่มีค่ามากกว่า (ยิ่งเข้าใกล้ 1 ยิ่งดี) ซึ่งหมายความว่ากลุ่มคำตอบที่ได้จาก MOEA/D มีสัดส่วนคำตอบที่อยู่ในคำตอบที่เหมาะสมที่สุดที่แท้จริงเมื่อเทียบกับจำนวนคำตอบที่หาได้ (R_{NDS1}) และเทียบกับจำนวนคำตอบที่เหมาะสมที่สุดที่แท้จริง (R_{NDS2}) ที่มากกว่า เมื่อเทียบกับ COIN

ถึงแม้ตัวชี้วัด Spread ของ MOEA/D มีค่ามากกว่า ซึ่งหมายความว่ากลุ่มคำตอบที่ได้จาก MOEA/D มีความกระจายตัวกันที่น้อยกว่า COIN ซึ่งอาจส่งผลมาจากจำนวนคำตอบที่ไม่ถูกรอบจำ (N_{NDS}) ที่มีค่าน้อยกว่ามาก และในด้านระยะเวลาที่ใช้ในการหาคำตอบ พบว่า MOEA/D จะใช้เวลาที่นานกว่า COIN ประมาณ 2 เท่าในทุกปัญหา

จากการวิเคราะห์ผลการทดลองข้างต้นจึงสามารถสรุปได้ว่า MOEA/D นั้นมีสมรรถนะในการแก้ปัญหาที่นำเสนอได้ดีกว่า COIN เพราะตัวชี้วัดในการหาคำตอบที่แท้จริง (Convergence, R_{NDS1} , R_{NDS2}) มีค่าที่ดีกว่า ซึ่งจะถือว่าอัลกอริทึมนี้สามารถค้นหาคำตอบได้ดีกว่า แม้ว่าตัวชี้วัดในการกระจายตัวของคำตอบและจำนวนคำตอบที่ไม่ถูกรอบจำจะมีค่าแย่กว่าก็ตาม เพราะเป็นตัวชี้วัดที่แสดงให้เห็นว่า มีจำนวนและการกระจายตัวของคำตอบให้ผู้แก้ปัญหาได้เลือกนำไปใช้ แต่ถ้ามีจำนวนของคำตอบให้เลือกเยอะแต่เป็นคำตอบที่ไม่ดีก็จะถือว่าเป็นอัลกอริทึมที่ไม่มีประสิทธิภาพเช่นกัน และรวมไปถึงเวลาที่ใช้จะนานกว่า แต่ผู้แก้ปัญหาย่อมพึงพอใจกับอัลกอริทึมที่สามารถค้นหาคำตอบที่ดีได้ภายในระยะเวลาที่ไม่นานจนเกินไป

6. บทสรุป

บทความนี้นำเสนอวิธีการแก้ปัญหาด้วยวิธีการเชิงวิวัฒนาการแบบหลายวัตถุประสงค์โดยยึดหลักการจำแนก (Multi-Objective Evolutionary Algorithm Based On Decomposition: MOEA/D) ร่วมกับวิธี Bisection มาประยุกต์ใช้ในการแก้ปัญหาการจัดสมดุลและจัดสรรพนักงานหลายทักษะบนสายการประกอบผลิตภัณฑ์แบบขนานที่มากวัตถุประสงค์ภายใต้ปัญหาประเภทที่ 2 ซึ่งทักษะที่หลากหลายเกิดจากความหลากหลายและความสามารถของพนักงาน โดยมีจำนวนวัตถุประสงค์ทั้งสิ้น 4 วัตถุประสงค์ ซึ่งจะพิจารณาค่าที่เหมาะสมที่สุดไปพร้อมๆ กัน ได้แก่ รอบเวลาดำเนินการน้อยที่สุด จำนวนสถานีน้อยที่สุด ความแตกต่างของปริมาณภาระงานระหว่างสถานีนงานมีค่าน้อยที่สุด และความไม่เกี่ยวเนื่องกันของงานในแต่ละสถานีนงานมีค่าน้อยที่สุด

ผลการทดลองใช้ MOEA/D ในการแก้ปัญหาตัวอย่างเพื่อทำการเปรียบเทียบกับอัลกอริทึมการบรรจบ (COIN) ซึ่งเป็นอีกอัลกอริทึมหนึ่งที่นิยม และมีประสิทธิภาพที่ดีในการแก้ปัญหาลักษณะนี้ พบว่า MOEA/D สามารถค้นพบกลุ่มคำตอบที่มีค่าต่ำที่สุดที่ต่ำกว่า (ดีกว่า) หรือเทียบเท่าของทุกฟังก์ชันวัตถุประสงค์ส่วนในด้านของตัวชี้วัดนั้น MOEA/D มีสมรรถนะที่เหนือกว่าในด้านการหาคำตอบที่เหมาะสมที่สุด อัตราส่วนของคำตอบที่ไม่ถูกรอบจำเทียบกับกลุ่มคำตอบที่อัลกอริทึมหาได้ และอัตราส่วนของคำตอบที่ไม่ถูกรอบจำเทียบกับกลุ่มคำตอบที่เหมาะสมที่แท้จริง แต่ในด้านการกระจายตัวของกลุ่มคำตอบ จำนวนคำตอบที่ไม่ถูกรอบจำ และเวลาในการค้นหาคำตอบนั้นมีสมรรถนะที่ดีไม่เท่า

แต่เมื่อพิจารณาตามลำดับความสำคัญของตัวชี้วัดนั้นจะเห็นว่า MOEA/D สามารถค้นพบกลุ่มคำตอบที่ดี เพราะมีการหาคำตอบที่แท้จริงมากกว่าซึ่งเป็นสิ่งสำคัญที่สุดในการค้นหาคำตอบโดยอัลกอริทึม เพราะผู้แก้ปัญหาส่วนใหญ่น่าจะอยากได้กลุ่มคำตอบที่ดีกว่าถึงแม้จะมีคำตอบให้เลือกน้อยและการกระจายตัวของกลุ่มคำตอบ

ไม่ดีก็ตาม รวมถึงเวลาในการค้นหาคำตอบนั้นผู้แก้ปัญหา ย่อมพึงพอใจกับอัลกอริทึมที่สามารถค้นหาคำตอบที่ดี ถึงแม้ว่าจะใช้เวลา นานกว่า ซึ่งในบทความนี้เวลาของ MOEA/D อาจจะนานกว่า COIN ถึง 2 เท่า แต่เมื่อ พิจารณาถึงเวลาที่ใช้จริงนั้นเวลาที่ใช้ในการค้นหาคำตอบ ที่นานที่สุดของ MOEA/D คือ 594 วินาที หรือประมาณ 10 นาทีจึงเป็นเวลา ที่ผู้แก้ปัญหายอมรับได้: ดังนั้นจึงสรุป ได้ว่า MOEA/D เป็นอัลกอริทึมหนึ่งที่ใช้ในการแก้ปัญหา การจัดสมดุลและจัดสรรพนักงานหลายทักษะบนสายการ ประกอบผลิตภัณฑ์ผสมแบบขนานที่มากวัตถุประสงค์ ภายใต้อุปสรรคประเภทที่ 2 (Type II) ได้อย่างมี ประสิทธิภาพ

จากการนำเสนอวิธีการแก้ปัญหาการจัดสมดุลและ จัดสรรพนักงานในบทความนี้สามารถนำไปประยุกต์ใช้กับ การแก้ปัญหาจริงในสถานประกอบการที่ต้องการปรับปรุง สายการประกอบที่มีเวลาดำเนินการไม่คงที่ขึ้นอยู่กับ พนักงาน ถึงแม้จะไม่มีพนักงานทุพพลภาพร่วมอยู่ด้วย เพราะการแก้ปัญหาแบบประเภทที่ 2 นั้นจะเหมาะกับการ ที่มีกำหนดจำนวนสถานงานหรือพนักงานไว้แล้ว แต่ ต้องการทำให้รอบเวลาดำเนินการน้อยที่สุด (วัตถุประสงค์ ที่ 1) รวมถึงวัตถุประสงค์อื่นๆของบทความนี้อาจจะช่วย เพิ่มประสิทธิภาพของสายการประกอบได้อีกด้วย

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561

ตารางที่ 16 ผลการทดลองแก้ปัญหาตัวอย่าง

Problem	P1				P2				P3				P4																																																			
Total worker	13		18		20		29		33		48		30		43																																																	
Worker ratio	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2																																																
Min Objective																																																																
Cycle time (C_T)	MOEA/D	48.6	50.3	36.2	38.2	243	249	175.2	179	6873	7202	5663	5663	4708	4884	3417	3528																																															
	COIN	48.55	51	36.4	38.25	245.5	249.9	175.2	182	7000	7275	5663	5663	4770	4886	3413	3517																																															
Number of station (N_s)	MOEA/D	8	8	12	11	15	14	22	21	27	26	40	38	23	24	35	36																																															
	COIN	9	8	12	12	16	16	22	23	27	27	41	41	25	25	36	37																																															
Balancing Workload Between Workstations (B_w)	MOEA/D	0.003	0.006	0.004	0.005	0.005	0.004	0.004	0.004	0.005	0.004	0.005	0.004	0.004	0.005	0.005	0.005																																															
	COIN	0.005	0.007	0.005	0.008	0.004	0.006	0.005	0.006	0.006	0.005	0.006	0.006	0.006	0.008	0.007	0.008																																															
Index of Task Unrelated (ITUR)	MOEA/D	0.350	0.409	0.217	0.250	0.630	0.623	0.532	0.540	0.588	0.548	0.448	0.415	0.773	0.779	0.713	0.705																																															
	COIN	0.458	0.435	0.280	0.308	0.652	0.667	0.574	0.567	0.607	0.607	0.495	0.500	0.797	0.796	0.726	0.724																																															
Convergence																																																																
MOEA/D	0.050				0.053				0.018				0.010				0.024				0.032				0.024				0.074				0.001				0.056				0.042																							
COIN	1.425				1.165				1.006				0.386				0.334				0.566				0.963				1.117				1.424				2.940				2.074				2.324				2.060				1.500				3.865				1.586			
Spread																																																																
MOEA/D	0.851				0.602				0.628				0.706				0.845				0.820				0.794				0.660				0.664				0.770				0.683				0.717				0.850				0.832				0.770				0.814			
COIN	0.526				0.541				0.483				0.443				0.621				0.726				0.528				0.456				0.584				0.465				0.534				1.123				0.679				0.710				0.585				0.445			
R_{ND51}																																																																
MOEA/D	0.738				0.747				0.857				0.831				0.964				0.702				0.792				0.771				0.710				0.516				0.851				0.727				0.761				0.970				0.640				0.741			
COIN	0.035				0.039				0.019				0.083				0.100				0.008				0.043				0.004				0				0.012				0				0				0				0.004				0.008							
R_{ND52}																																																																
MOEA/D	0.385				0.409				0.450				0.349				0.391				0.491				0.405				0.488				0.500				0.444				0.500				0.500				0.500				0.500				0.486				0.485			
COIN	0.115				0.091				0.050				0.151				0.109				0.009				0.095				0.012				0				0.056				0				0				0				0.014				0.015							
N_{ND5}																																																																
MOEA/D	21				27				21				28				31				62				24				29				42				36				26				44				38				31				37							
COIN	89				102				101				75				108				81				107				79				108				85				99				66				102				74				98				79			
CPU Time (s)																																																																
MOEA/D	41.21				43.18				41.58				44.32				134.3				132.2				167.9				190.3				314.2				315.5				320.7				331.6				540.7				543.5				593.8				578.6			
COIN	19.82				20.25				20.97				23.99				70.55				63.59				78.21				75.24				158.1				156.4				168.8				167.9				330.3				360.3				396.7				392.8			

7. เอกสารอ้างอิง

- [1] S. Ghosh and R. J. Gagnon, "A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems," *Int. J. Prod. Res.*, vol. 27, no. 4, pp. 637-670, 1989.
- [2] U. Özcan, "Balancing stochastic two-sided assembly lines: A chance-constrained, piecewise-linear, mixed integer program and a simulated annealing algorithm," *Eur. J. Oper. Res.*, vol. 205, no. 1, pp. 81-97, 2010.
- [3] O. Kilincci, "A Petri net-based heuristic for simple assembly line balancing problem of type 2," *Int. J. Adv. Manuf. Technol.*, vol. 46, no. 1-4, pp. 329-338, 2009.
- [4] A. A. Chaves, L. A. N. Lorena, and C. Miralles, "Hybrid Metaheuristic for the Assembly Line Worker Assignment and Balancing Problem," in *Hybrid Metaheuristics: 6th International Workshop, HM 2009, Udine, Italy, October 16-17, 2009. Proceedings*, M. J. Blesa, C. Blum, L. Di Gaspero, A. Roli, M. Sampels, and A. Schaerf, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1-14.
- [5] W. Zhang, M. Gen, and L. Lin, "A multiobjective genetic algorithm for Assembly Line Balancing problem with worker allocation," *IEEE Trans. Syst. Man Cybern.*, pp. 3026-3033, 2008.
- [6] F. F. B. Araújo, A. M. Costa, and C. Miralles, "Balancing parallel assembly lines with disabled workers," *Eur. J. Ind. Eng.*, vol. 9, no. 3, p. 344, 2015.
- [7] H. Gökçen, K. Ağpak, and R. Benzer, "Balancing of parallel assembly lines," *Int. J. Prod. Econ.*, vol. 103, no. 2, pp. 600-609, 2006.
- [8] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712-731, 2007.
- [9] A. Scholl and N. Boysen, "Designing parallel assembly lines with split workplaces: Model and optimization procedure," *Int. J. Prod. Econ.*, vol. 119, no. 1, pp. 90-100, 2009.
- [10] A. Scholl, "Data of assembly line balancing problems," Darmstadt Technical University, Department of Business Administration, Economics and Law, Institute for Business Studies (BWL) 1995.
- [11] D. Sparling and J. Miltenburg, "The mixed-model u-line balancing problem," *Int. J. Prod. Res.*, vol. 36, no. 2, pp. 485-501, 1998.
- [12] Y. K. Kim, Y. Kim, and Y. J. Kim, "Two-sided assembly line balancing: A genetic algorithm approach," *Prod. Plann. Contr.*, vol. 11, no. 1, pp. 44-53, 2000.
- [13] T. O. Lee, Y. Kim, and Y. K. Kim, "Two-sided assembly line balancing to maximize work relatedness and

ปีที่ 13 ฉบับที่ 1 เดือนมกราคม – เมษายน พ.ศ. 2561

- slackness," *Comput. Ind. Eng.*, vol. 40, no. 3, pp. 273-292, 2001.
- [14] D. E. Goldberg and R. Lingle, "Alleles, loci, and the traveling salesman problem," in *Proceedings of an international conference on genetic algorithms and their applications*, 1985, vol. 154, pp. 154-159: Lawrence Erlbaum, Hillsdale, NJ.
- [15] ปารเมศ ชูติมา, ภาณุวัฒน์ โอฬารวิวัฒน์ชัย, วรินทร์ วัฒนพรพรหม, และ ประภาส จงสฤษดิ์วัฒนา, "การประยุกต์ใช้อัลกอริทึมการบรรจบสำหรับปัญหาการจัดสมดุลสายการประกอบแบบลักษณะตัวยูที่ทำการผลิตผลิตภัณฑ์ผสมที่มีหลายวัตถุดิบประสมในระบบผลิตแบบทันเวลาพอดี," การประชุมทางวิชาการของมหาวิทยาลัยเกษตรศาสตร์ ครั้งที่ 47, 2552.
- [16] P. Chutima and P. Olanviwatchai, "Mixed-Model U-Shaped Assembly Line Balancing Problems with Coincidence Memetic Algorithm," *JSEA*, vol. 3, no. 4, pp. 347-363, 2010.
- [17] สถาพร โอฬารวิวัฒน์ชัย และ ปารเมศ ชูติมา, "การจัดลำดับการผลิตแบบหลายวัตถุดิบประสมสายการประกอบผลิตภัณฑ์ผสมแบบสองด้าน," *วารสารวิชาการพระจอมเกล้าพระนครเหนือ*, vol. 24, no. 1, pp. 87-102, 2557.
- [18] R. K. Hwang, H. Katayama, and M. Gen, "U-shaped assembly line balancing problem with genetic algorithm," *Int. J. Prod. Res.*, vol. 46, no. 16, pp. 4637-4649, 2008.
- [19] R. Kumar and P. K. Singh, "Pareto Evolutionary Algorithm Hybridized with Local Search for Biobjective TSP," in *Hybrid Evolutionary Algorithms*, A. Abraham, C. Grosan, and H. Ishibuchi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 361-398.
- [20] วัชรวิทย์ ถนันทอง, "การจัดลำดับผลิตรถยนต์แบบหลายวัตถุดิบสายการประกอบผลิตภัณฑ์ผสมแบบสองด้านด้วยอัลกอริทึมการบรรจบร่วมกับฟิชชี," วิศวกรรมศาสตรมหาบัณฑิต, สาขาวิชาวิศวกรรมอุตสาหการ ภาควิชาวิศวกรรมอุตสาหการ, จุฬาลงกรณ์มหาวิทยาลัย, 2558.