วิธีการวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู เพื่อหาค่ารอบเวลาที่ดีที่สุด กรณีทราบค่าจำนวนสถานีงาน

วิทยานิพนธ์

ของ

กฤต จันทรสมัย

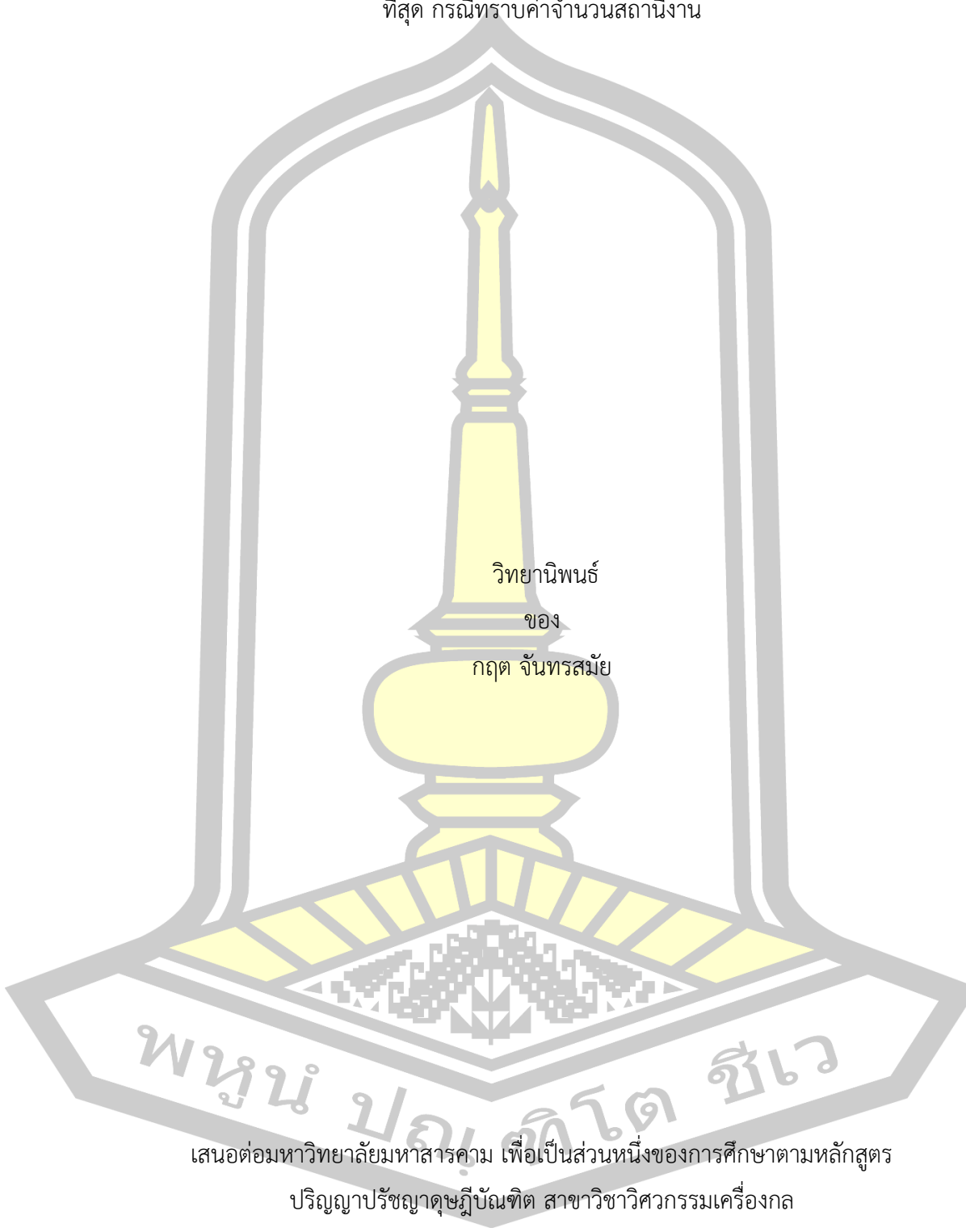
เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมเครื่องกล

มกราคม 2564

ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม

วิธีการวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู เพื่อหาค่ารอบเวลาที่ดีที่สุด กรณีทราบค่าจำนวนสถานีงาน



วิทยานิพนธ์

ของ

กฤต จันทรสมัย

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

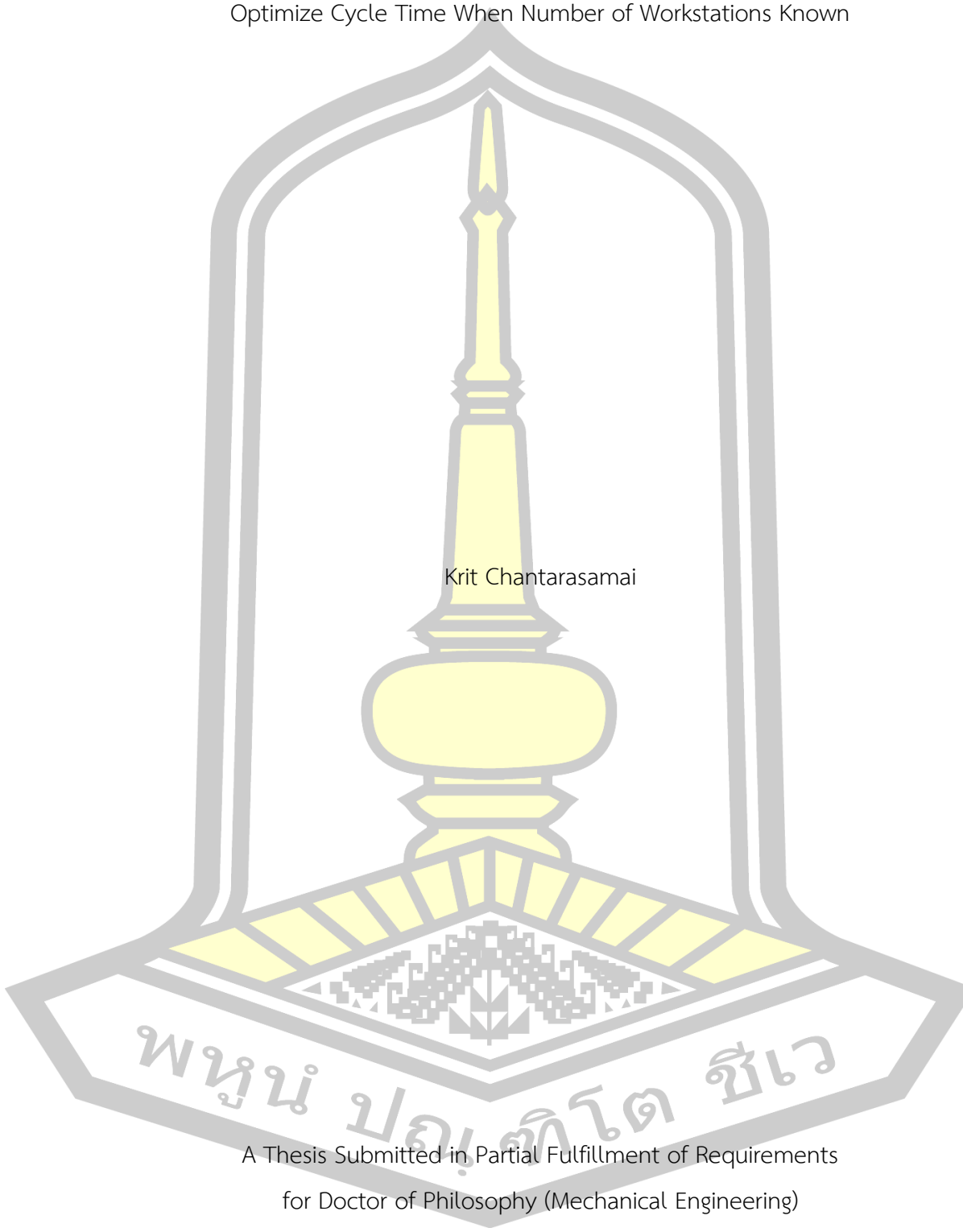
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมเครื่องกล

มกราคม 2564

ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม

A Differential Evolution Algorithm for U – Shaped Assembly Line Balancing to Optimize Cycle Time When Number of Workstations Known

Krit Chantarasamai



A Thesis Submitted in Partial Fulfillment of Requirements

for Doctor of Philosophy (Mechanical Engineering)

January 2021

Copyright of Mahasarakham University



คณะกรรมการสอบวิทยานิพนธ์ ได้พิจารณาวิทยานิพนธ์ของนายกฤต จันทรสมัย แล้ว เห็นสมควรรับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมเครื่องกล ของมหาวิทยาลัยมหาสารคาม

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(ศ. ดร. ระพีพันธ์ ปิตาคะโส)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(ผศ. ดร. อรอุมา ลาสุนนท์)

.....กรรมการ

(ผศ. ดร. ยศฐา ศรีเทพ)

.....กรรมการ

(ผศ. ดร. ละมุล วิเศษ)

.....กรรมการ

(รศ. ดร. เกียรติศักดิ์ ศรีประทีป)

มหาวิทยาลัยอนุมัติให้รับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา ปรัชญาดุษฎีบัณฑิต สาขาวิชาวิศวกรรมเครื่องกล ของมหาวิทยาลัยมหาสารคาม

.....	.....
(รศ. ดร. เกียรติศักดิ์ ศรีประทีป)	(รศ. ดร. กริสน์ ชัยมูล)
คณบดีคณะวิศวกรรมศาสตร์	คณบดีบัณฑิตวิทยาลัย

ชื่อเรื่อง วิธีการวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู เพื่อหาค่ารอบเวลาที่ดีที่สุด กรณีทราบค่าจำนวนสถานีงาน

ผู้วิจัย กฤต จันทรสมัย

อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร. อรอุมา ลาสุนนท์

ปริญญา ปรัชญาดุษฎีบัณฑิต **สาขาวิชา** วิศวกรรมเครื่องกล

มหาวิทยาลัย มหาวิทยาลัยมหาสารคาม **ปีที่พิมพ์** 2564

บทคัดย่อ

งานวิจัยนี้นำเสนอวิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithm: DE) สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 2 (U-Shaped assembly line balancing type 2: UALBP-2) ที่มีความสำคัญต่อการผลิตแบบทันเวลา (Just In Time Production) เพื่อหารอบเวลาการผลิต (cycle time) ที่ต่ำที่สุด สำหรับการผลิตสินค้าชนิดเดียว กรณีทราบจำนวนสถานีงานที่แน่นอน โดยในการศึกษาได้พัฒนาวิธีการค้นหาคำตอบและทำการทดสอบกับปัญหาการจัดสมดุลสายการประกอบจำนวน 15 ชุดปัญหา (101 ตัวอย่าง) แบ่งเป็นปัญหาขนาดกลางที่มีจำนวนงาน 21-58 งาน จำนวน 10 ชุดปัญหา (50 ตัวอย่าง) และปัญหาขนาดใหญ่ ที่มีจำนวนงาน 70-297 งาน จำนวน 5 ชุดปัญหา (51 ตัวอย่าง) จากนั้นทำการเปรียบเทียบผลกับวิธี Two Rules และวิธี Three Rules ของ Rules-based heuristic พบว่า ในปัญหาขนาดกลางวิธีการวิวัฒนาการโดยใช้ผลต่างสามารถหาคำตอบได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules 39 ปัญหา (ร้อยละ 78) และดีกว่าวิธี Three Rules 14 ปัญหา (ร้อยละ 28) โดยใช้เวลาเฉลี่ยในการคำนวณอยู่ระหว่าง 0.02-0.40 วินาที/ปัญหา ส่วนในปัญหาขนาดใหญ่วิธีการวิวัฒนาการโดยใช้ผลต่างสามารถหาคำตอบได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules 38 ปัญหา (ร้อยละ 76) และวิธี Three Rules 3 ปัญหา (ร้อยละ 6) โดยใช้เวลาเฉลี่ยในการคำนวณอยู่ระหว่าง 0.39-40 นาที/ปัญหา ดังนั้นวิธีวิวัฒนาการโดยใช้ผลต่างจึงเป็นวิธีที่สามารถใช้ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 2 ได้อย่างมีประสิทธิภาพ

คำสำคัญ : การจัดสมดุลสายการประกอบแบบตัวยู, วิธีวิวัฒนาการโดยใช้ผลต่าง, การผลิตแบบทันเวลา

TITLE A Differential Evolution Algorithm for U – Shaped Assembly Line Balancing to Optimize Cycle Time When Number of Workstations Known

AUTHOR Krit Chantarasamai

ADVISORS Assistant Professor On-uma Lasunon , Ph.D.

DEGREE Doctor of Philosophy **MAJOR** Mechanical Engineering

UNIVERSITY Mahasarakham University **YEAR** 2021

ABSTRACT

This research presents a Differential Evolution (DE) algorithm for solving U-shaped assembly line balancing type 2 problems. The minimum cycle time in a just-in-time production line for producing a single product with a certain number of workstations was investigated by developing solution methods and testing with 15 problem sets (101 instances). The problems were divided into 10 medium-scale problem sets with 21-58 tasks (50 instances) and 5 large-scale problem sets with 70-297 tasks (51 instances). The results were compared with those obtained from the rules-based heuristic; two rules and three rules. In medium-scale problems, the DE algorithm generated better cycle time than two rules and three rules for 39 instances (78%) and 14 instances (28%), respectively. The average computational time (CPU) was between 0.02-0.40 second/instance. For the large-scale problems, the DE algorithm provided 38 better solutions (76%) compared with two rules and 3 better solutions (6%) compared with three rules. The average computational time was between 0.39-40 second/instance. Therefore, the DE algorithm is a potential and effective approach to solve the U-shape assembly line balancing type 2 problems.

Keyword : U-shaped assembly line balancing, Differential evolution algorithm, Just-in-time

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยความกรุณาและความช่วยเหลือเป็นอย่างดีจาก ผู้ช่วยศาสตราจารย์ ดร.อรอุมา ลาสุนนท์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผู้ซึ่งกรุณาให้ความรู้ คำแนะนำ ข้อคิดเห็น ให้คำปรึกษา เป็นอย่างดีตลอดมา จึงขอกราบขอบพระคุณท่านเป็นอย่างสูง นอกจากนี้ขอขอบพระคุณอาจารย์สาขาวิชาวิศวกรรมการผลิต และสาขาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหาสารคาม ทุกท่านที่ได้ให้ความรู้ และให้โอกาสเข้ารับการศึกษา ณ มหาวิทยาลัยแห่งนี้

ขอขอบพระคุณท่านคณะกรรมการสอบวิทยานิพนธ์ทุกท่านที่สละเวลาอันมีค่าของท่านมาเป็นคณะกรรมการสอบวิทยานิพนธ์ ได้แก่ ศาสตราจารย์ ดร.ระพีพันธ์ ปิตาคะโส รองศาสตราจารย์ ดร. เกียรติศักดิ์ ศรีประทีป ผู้ช่วยศาสตราจารย์ ดร.ยศฐา ศรีเทพ และผู้ช่วยศาสตราจารย์ ดร. ละมุล วิเศษ ที่ช่วยกรุณาตรวจสอบวิทยานิพนธ์และให้คำแนะนำที่ดีเสมอมารวมทั้งขอขอบพระคุณเจ้าหน้าที่ของทางคณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหาสารคามทุกท่านที่คอยให้กำลังใจและให้คำแนะนำในการทำเอกสารวิทยานิพนธ์ในครั้งนี้

สุดท้ายนี้ ผู้วิจัยขอกราบขอบพระคุณครอบครัวผู้อยู่เบื้องหลังที่คอยสนับสนุนมาโดยตลอดตลอดจนทุกคนที่คอยให้กำลังใจแก่ผู้วิจัยเสมอมา และนอกจากนี้ขอขอบคุณบุคคลที่เกี่ยวข้องทุกท่านที่ไม่ได้กล่าวนามที่คอยช่วยเหลือและเป็นกำลังใจจนประสบความสำเร็จ ผู้วิจัยขอขอบพระคุณทุกท่านไว้ ณ โอกาสนี้ด้วย

กฤต จันทรสมัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ฌ

สารบัญภาพประกอบ ..... ญ

คำอธิบายสัญลักษณ์ ..... ฐ

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ 3

1.3 สมมติฐานของการวิจัย 3

1.4 ขอบเขตของการวิจัย 4

1.5 ขั้นตอนการดำเนินการวิจัย 4

1.6 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย 4

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 6

2.1 ระบบการผลิต (Production system) 6

2.2 หลักการและทฤษฎีที่เกี่ยวข้อง 10

2.3 งานวิจัยที่เกี่ยวข้องกับการจัดสมดุลสายการประกอบ 19

บทที่ 3 วิธีการดำเนินการวิจัย 32

3.1 ขั้นตอนการดำเนินงานวิจัย 32

บทที่ 4 ผลการวิจัย 46

4.1 การพัฒนาโปรแกรมการจัดสมดุลสายการประกอบ 46
4.2 ชุดปัญหาของการทดสอบการจัดสมดุลสายการประกอบ 47
4.3 การจัดสมดุลสายการประกอบ 48
4.4 ผลของการจัดสมดุลสายการประกอบและการเปรียบเทียบผล 50
4.5 สรุปผลการวิจัย 62
บทที่ 5 สรุปผลการวิจัย 69
5.1 สรุปผลการวิจัย 69
5.2 ปัญหาและอุปสรรคของงานวิจัย 71
5.3 ข้อเสนอแนะและแนวทางศึกษาการวิจัยในอนาคต 72
บรรณานุกรม 73
ภาคผนวก 79
ภาคผนวก ก ชุดโจทย์ปัญหาตัวอย่าง (Data set) ของ Armin Scholl 80
ภาคผนวก ข อธิบายขั้นตอนการกรอกข้อมูลและการแสดงผลของโปรแกรม DE2 93
ภาคผนวก ค ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 101
ประวัติผู้เขียน 149

พหูน ปณ ฑิโต ชีเว

สารบัญตาราง

หน้า

ตาราง 1 สรุปปัญหา SALBP ของแต่ละประเภท 13

ตาราง 2 สรุปปัญหา UALBP ของแต่ละประเภท 14

ตาราง 3 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู 23

ตาราง 4 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบประเภทที่ 2 27

ตาราง 5 สรุปงานวิจัยที่เกี่ยวข้องกับการจัดสมดุลสายการประกอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง 30

ตาราง 6 ค่า Target Vector ของคำตอบเริ่มต้น 40

ตาราง 7 ผลการสร้างคำตอบเริ่มต้นของปัญหาเท่ากับจำนวน *NP* 40

ตาราง 8 ผลการมิวแทนเวกเตอร์ของเวกเตอร์เป้าหมาย 1 42

ตาราง 9 การไทรอัลเวกเตอร์ของเวกเตอร์เป้าหมาย 1 43

ตาราง 10 การจัดงานเข้าสู่สถานีงานของเวกเตอร์เป้าหมาย 1 43

ตาราง 11 ชุดปัญหาในการทดสอบการจัดสมดุลสายการประกอบ 44

ตาราง 12 ตารางแสดงข้อมูลของชุดปัญหาที่ใช้ในการทดสอบ 47

ตาราง 13 การเปรียบเทียบผลของปัญหาขนาดกลาง (21-58 งาน) กับวิธี Two Rules 50

ตาราง 14 การเปรียบเทียบผลของปัญหาขนาดใหญ่ (70-297 งาน) กับวิธี Two Rules 53

ตาราง 15 การเปรียบเทียบผลของปัญหาขนาดกลาง (21-58 งาน) กับวิธี Three Rules 56

ตาราง 16 การเปรียบเทียบผลของปัญหาขนาดใหญ่ (70-297 งาน) กับ Three Rules 59

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง 102

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ 116

สารบัญภาพประกอบ

หน้า

ภาพประกอบ 1 ประเภทของปัญหาการจัดสมดุลสายการประกอบ [1] 2

ภาพประกอบ 2 ระบบการผลิต 6

ภาพประกอบ 3 สายการประกอบที่แบ่งกระบวนการผลิตตามจำนวนชนิดของสินค้า [20] 8

ภาพประกอบ 4 สายการประกอบเย็บเสื้อผ้าโรงงานกรณีศึกษาเป็นการผลิตแบบสินค้าชนิดเดียว 9

ภาพประกอบ 5 สายการประกอบชิ้นส่วนเครื่องปรับอากาศเป็นการผลิตแบบสินค้าหลายชนิด 9

ภาพประกอบ 6 สายการประกอบเครื่องปรับอากาศที่มีความหลากหลายของรุ่น 10

ภาพประกอบ 7 แผนภาพลำดับงานก่อน-หลังของปัญหา Bowman [21] 11

ภาพประกอบ 8 ประเภทของปัญหาการจัดสมดุลสายการประกอบ [1] 12

ภาพประกอบ 9 ขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง [22] 16

ภาพประกอบ 10 ลำดับขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง [22] 16

ภาพประกอบ 11 ลำดับขั้นตอนดำเนินงานวิจัย 32

ภาพประกอบ 12 ลำดับขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง 34

ภาพประกอบ 13 แสดงแผนผังลำดับขั้นตอนงานของปัญหา Jackson [58] 39

ภาพประกอบ 14 กระบวนการพัฒนาโปรแกรม DE2 46

ภาพประกอบ 15 แสดงการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 49

ภาพประกอบ 16 การเปรียบเทียบรอบเวลาการผลิตในปัญหาขนาดกลาง 63

ภาพประกอบ 17 การเปรียบเทียบของจำนวนปัญหาในปัญหาขนาดกลาง 64

ภาพประกอบ 18 การเปรียบเทียบรอบเวลาการผลิตในปัญหาขนาดใหญ่ 64

ภาพประกอบ 19 การเปรียบเทียบของจำนวนปัญหาในปัญหาขนาดใหญ่ 65

ภาพประกอบ 20 การเปรียบเทียบรอบเวลาการผลิตในปัญหาขนาดกลาง 66

ภาพประกอบ 21 การเปรียบเทียบเวลาในการค้นหาคำตอบ 67

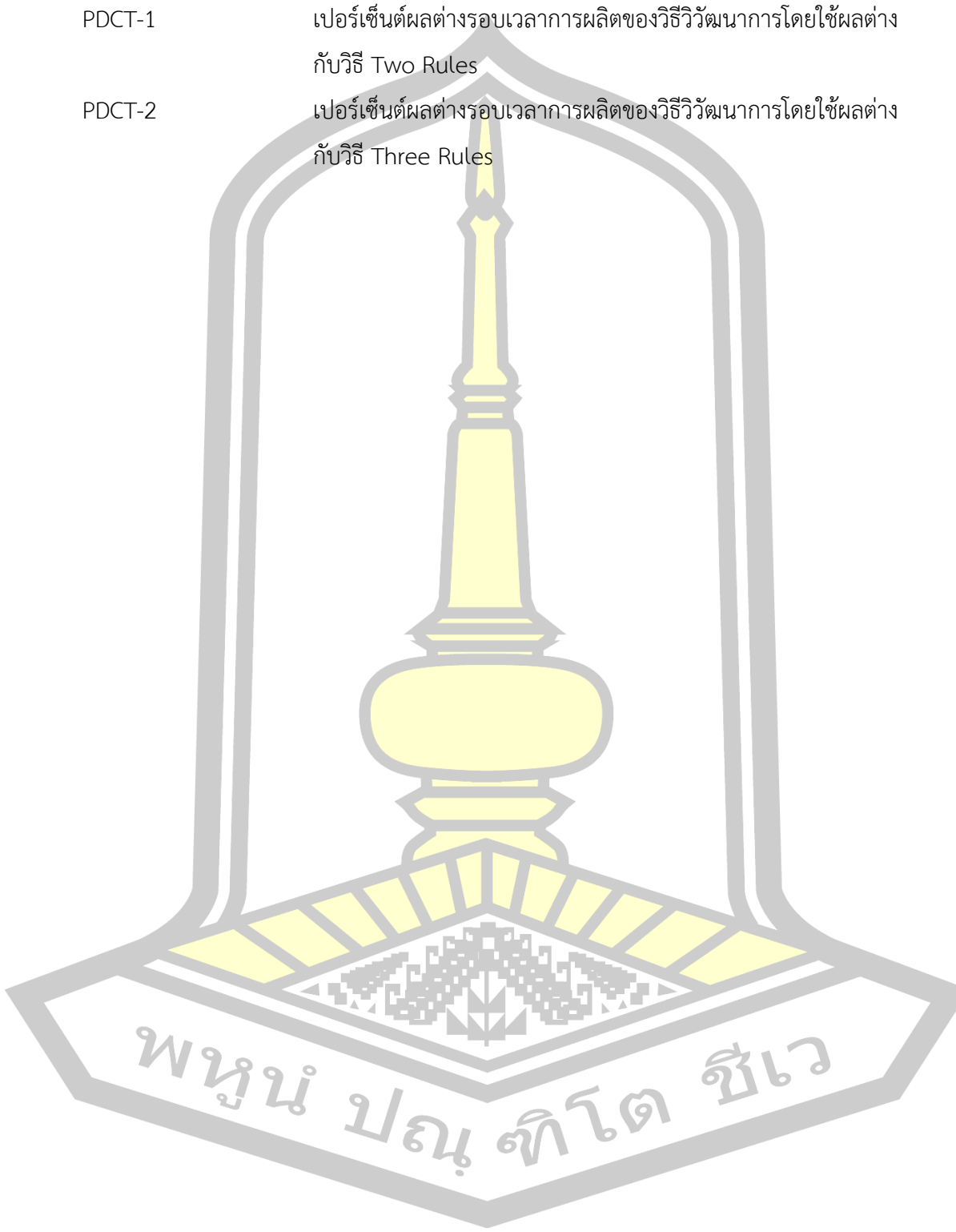
ภาพประกอบ 22 การเปรียบเทียบผลของรอบเวลาการผลิต 70
ภาพประกอบ 23 ชุดปัญหาของ Mitchell ที่มีจำนวน 21 งาน 81
ภาพประกอบ 24 ชุดปัญหาของ Rosenberg ที่มีจำนวน 25 งาน 81
ภาพประกอบ 25 ชุดปัญหาของ Heskiaoff ที่มีจำนวน 28 งาน 82
ภาพประกอบ 26 ชุดปัญหาของ Buxey ที่มีจำนวน 29 งาน 83
ภาพประกอบ 27 ชุดปัญหาของ Sawyer ที่มีจำนวน 30 งาน 83
ภาพประกอบ 28 ชุดปัญหาของ Lutz1 ที่มีจำนวน 32 งาน 84
ภาพประกอบ 29 ชุดปัญหาของ Gunther ที่มีจำนวน 35 งาน 84
ภาพประกอบ 30 ชุดปัญหาของ Kilbridge&Wester ที่มีจำนวน 45 งาน 85
ภาพประกอบ 31 ชุดปัญหาของ Hahm ที่มีจำนวน 53 งาน 85
ภาพประกอบ 32 ชุดปัญหาของ Warnecke ที่มีจำนวน 58 งาน 86
ภาพประกอบ 33 ชุดปัญหาของ Tonge ที่มีจำนวน 70 งาน 87
ภาพประกอบ 34 ชุดปัญหาของ Lutz2 ที่มีจำนวน 89 งาน 88
ภาพประกอบ 35 ชุดปัญหาของ Mukherjee ที่มีจำนวน 94 งาน 89
ภาพประกอบ 36 ชุดปัญหาของ Bartholdi ที่มีจำนวน 148 งาน 90
ภาพประกอบ 37 ชุดปัญหาของ Scholl ที่มีจำนวน 297 งาน 91
ภาพประกอบ 38 ชุดปัญหาของ Scholl ที่มีจำนวน 297 งาน (ต่อ) 92
ภาพประกอบ 39 ขั้นตอนการกรอกพารามิเตอร์ที่เกี่ยวข้องกับการคำนวณโดยใช้วิธี 94
ภาพประกอบ 40 ขั้นตอนการกรอกสูตรการคำนวณวิธีวัฒนาการโดยใช้ผลต่าง 95
ภาพประกอบ 41 ขั้นตอนการกรอกข้อมูลของชุดปัญหาการจัดสมดุลสายการประกอบ 96
ภาพประกอบ 42 ขั้นตอนการโหลดข้อมูลของชุดปัญหาที่ทำการบันทึกไว้ 97
ภาพประกอบ 43 แสดงข้อมูลของชุดปัญหาที่จะทำการทดสอบ 98
ภาพประกอบ 44 ขั้นตอนของการรันผลของโปรแกรม 98
ภาพประกอบ 45 แสดงผลการคำนวณ 99

ภาพประกอบ 46 แสดงผลการจัดสมดุลสายการประกอบแบบ..... 99

คำอธิบายสัญลักษณ์

สัญลักษณ์	ความหมาย
JIT	ระบบทันเวลาพอดี
DE	วิธีวิวัฒนาการโดยใช้ผลต่าง
SALBP	ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรง
SALBP-1	ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงหาสถานีงานต่ำที่สุด
SALBP-2	ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงหารอบเวลาต่ำที่สุด
SALBP-E	ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงหาประสิทธภาพสูงที่สุด
SALBP-F	ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงหาความเป็นปได้
GALBP	ปัญหาการจัดสมดุลสายการประกอบแบบทั่วไป
UALBP	ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู
UALBP-1	ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูหาสถานีงานต่ำที่สุด
UALBP-2	ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูหารอบเวลาต่ำที่สุด
UALBP-E	ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูหาประสิทธภาพสูงที่สุด
MALBP	ปัญหาการจัดสมดุลสายการประกอบที่มีผลิตภัณฑ์แบบผสม
MSP	ปัญหาการจัดสมดุลสายการประกอบที่มีการจัดลำดับการผลิต
m	จำนวนสถานีงาน
c	รอบเวลาการผลิต
E	ประสิทธิภาพสายการผลิต
BD	การสูญเสียความสมดุล
ID	เวลาสูญเปล่า
F	แฟคเตอร์ในการขยายผลต่าง
CR	อัตราการแลกเปลี่ยนค่าในพิกัด
NP	จำนวนขั้นตอนงาน
R	จำนวนรอบการทดลอง
D	พิกัดในเวกเตอร์
G	ประชากรในแต่ละรุ่น
OPT	รอบเวลาการผลิตที่เหมาะสมที่สุด
CPU	เวลาในการค้นหาคำตอบ (second)

MCPU	เวลาในการค้นหาคำตอบโดยเฉลี่ย
PDCT-1	เปอร์เซ็นต์ผลต่างรอบเวลาการผลิตของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Two Rules
PDCT-2	เปอร์เซ็นต์ผลต่างรอบเวลาการผลิตของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Three Rules



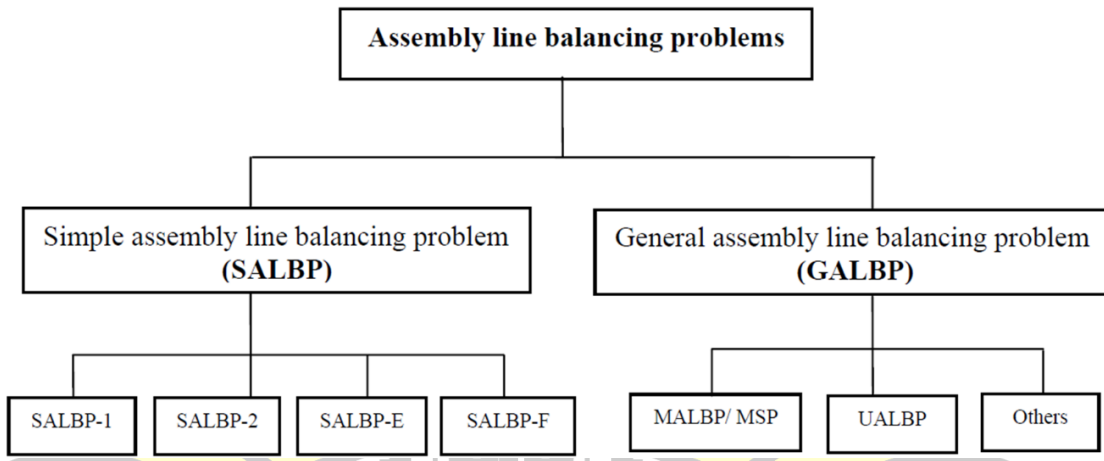
บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบันสภาวะของธุรกิจอุตสาหกรรมทั่วโลก มีการตื่นตัวเพิ่มมากขึ้นมีเทคโนโลยีนำสมัยเข้ามาพัฒนาระบบการผลิตเพื่อที่จะให้บริษัทของตนเองอยู่รอดและแข็งแกร่งเมื่อเปรียบเทียบกับคู่แข่งรายอื่น ๆ ทำให้บริษัทของตนเองต้องมีปรับตัวเพื่อที่จะตอบสนองความต้องการของลูกค้ามากที่สุดไม่ว่าจะเป็นการส่งมอบ วัตถุดิบ สินค้า หรือบริการ ให้ถึงมือลูกค้าทันเวลาที่ต้องการมากที่สุด หรือที่เรียกว่า ระบบทันเวลาพอดี (Just-in-time: JIT) ถ้าบริษัทไหนมีการตอบสนองต่อลูกค้าได้ดีและต่อเนื่องก็จะทำให้บริษัทนั้นคงอยู่ได้นานและมั่นคงมากยิ่งขึ้น ดังนั้นการบริหารการจัดการที่ดีจึงมีความสำคัญต่อบริษัท โดยในปัจจุบันแต่ละบริษัทได้มีการพัฒนาและปรับปรุงผลิตภัณฑ์อย่างต่อเนื่องอยู่ตลอดเวลา ซึ่งเมื่อมีการพัฒนาและปรับปรุงผลิตภัณฑ์จะทำให้ระบบการผลิตไม่ว่าจะเป็นขั้นงาน รอบเวลาการผลิต (Cycle time) ลำดับงานก่อน-หลัง (Precedence diagram) เปลี่ยน เมื่อระบบการผลิตเปลี่ยนการจัดสมดุลสายการประกอบ (Assembly line balancing) ก็ต้องมีการปรับใหม่เช่นกัน ซึ่งปัญหาการจัดสมดุลสายการประกอบ เป็นปัญหาที่สำคัญต่ออุตสาหกรรมที่มีการประกอบชิ้นส่วน เช่น รถยนต์ แอร์ เป็นต้น

การจัดสมดุลสายการประกอบ คือรูปแบบหนึ่งของการวางแผนการผลิตที่ใช้สำหรับการมอบหมายงานให้กับแต่ละสถานีภายใต้เงื่อนไขของรอบเวลาการผลิตและลำดับงานก่อนหลัง เพื่อให้แต่ละสถานีงานทำงานโดยใช้เวลาในการผลิตเฉลี่ยเท่ากันและทำให้ระบบไหลของกระบวนการมีความยืดหยุ่นและขจัดความล่าช้าหรือคอขวดเพื่อให้สามารถผลิตสินค้าได้อย่างถูกต้องและขจัดข้อผิดพลาดระหว่างการผลิต [1-2] ซึ่งปัญหาการจัดสมดุลสายการประกอบสามารถแบ่งออกเป็น 2 ประเภท คือ ปัญหาการจัดสมดุลสายการประกอบแบบอย่างง่าย (Simple assembly line balancing problem: SALBP) และปัญหาการจัดสมดุลสายการประกอบแบบทั่วไป (General assembly line balancing problem: GALBP) แสดงดังภาพประกอบ 1 โดยปัญหา SALBP สามารถบ่งออกเป็น 4 ประเภทคือ ปัญหา SALBP-1 หาจำนวนสถานีงานที่ต่ำที่สุด กรณีทราบรอบเวลาการผลิต, ปัญหา SALBP-2 หารอบเวลาการผลิตที่ต่ำที่สุด กรณีทราบค่าจำนวนสถานีงาน, SALBP–E หาประสิทธิภาพสูงที่สุด กรณีทราบรอบเวลาการผลิตและจำนวนสถานีงาน และ SALBP–F เป็นปัญหาที่สามารถหาคำตอบได้ กรณีทราบรอบเวลาการผลิตและจำนวนสถานีงาน ส่วนในปัญหา GALBP สามารถบ่งออกเป็น 3 ประเภทคือ ปัญหา

MALBP/MSP เป็นหาการผลิตภัณฑ์แบบผสม มีเวลาการทำงานที่แน่นอน, ปัญหา UALBP เป็นปัญหาการจัดสมดุลสายการประกอบแบบตัวยูที่สมารถแยกย่อยออกเป็น UALBP-1 หาจำนวนสถานีงานที่ต่ำที่สุด กรณีทราบรอบเวลาการผลิต, UALBP-2 หารอบเวลาการผลิตที่ต่ำที่สุด กรณีทราบจำนวนสถานีงานและ UALBP-E หาประสิทธิภาพสูงที่สุด กรณีทราบรอบเวลาการผลิตและจำนวนสถานีงาน และปัญหาอื่นๆ จะเป็นปัญหาที่มีขอบเขตที่กว้าง ซึ่งอาจจะพิจารณาตามเงื่อนไขอื่นๆที่เกี่ยวข้อง เช่น การเลือกเครื่องมือและอุปกรณ์ (Equipment selection), การผลิตในแบบอื่นๆ (Processing alternatives) และข้อจำกัดของการกำหนดงาน (Assignment restrictions) โดยในอุตสาหกรรมหลายๆประเภทมีการใช้ SALBP ในรูปแบบเส้นตรง (Straight line) และ GALBP ในรูปแบบตัวยู (U-Shaped line) แต่เมื่อเปรียบเทียบรูปแบบการจัดสายการประกอบทั้ง 2 รูปแบบ พบว่ารูปแบบตัวยูมีข้อดีมากกว่ารูปแบบเส้นตรงทั้งในเรื่องของประสิทธิภาพ (Efficiency) จัดสมดุลสายการประกอบ จำนวนสถานีงานที่น้อยกว่า มีความยืดหยุ่นสูงกว่า และมีรอบการผลิตที่ต่ำกว่า เป็นต้น รูปแบบตัวยู จึงเป็นรูปแบบที่น่าสนใจสำหรับการจัดสมดุลสายการประกอบ



ภาพประกอบ 1 ประเภทของปัญหาการจัดสมดุลสายการประกอบ [1]

การจัดสมดุลสายการประกอบแบบประเภทที่ 2 เป็นปัญหาชนิดหนึ่งของการจัดสมดุลสายการประกอบที่มีความสำคัญต่อ JIT ที่มีวัตถุประสงค์หารอบเวลาการผลิตที่ต่ำที่สุด เมื่อทราบจำนวนสถานีงาน โดยถูกนำเสนอครั้งแรกในปี 1996 สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบอย่างง่าย ประเภทที่ 2 (Simple assembly line balancing type 2: SALBP-2) ด้วยวิธี Branch and bound [3] หลังจากนั้นก็ได้มีนักวิจัยหลายๆท่านได้นำเสนอวิธีฮิวริสติกและเมตาฮิวริสติกต่างๆสำหรับแก้ไขปัญหาทั้งในรูปแบบเส้นตรง เช่น วิธี Two bi-

directional heuristics [4-5], วิธีการจำลองการอบเหนียว (Simulated annealing Algorithm: SA) [6], Variable neighborhood strategy adaptive search (VNS) [2], วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential evolution algorithm: DE) [7] และรูปแบบตัวยู เช่น วิธีเชิงพันธุกรรม (Genetic algorithm: GA) [8], วิธี Rules-based heuristic [9] เมื่อทำการทบทวนวรรณกรรมของวิธีฮิวริสติกและเมตาฮิวริสติกส์สำหรับแก้ปัญหาการจัดสมดุลสายการประกอบแบบประเภทที่ 2 พบว่าวิธีการวิวัฒนาการโดยใช้ผลต่าง (DE) เป็นวิธีเมตาฮิวริสติกที่น่าสนสำหรับการใช้แก้ไขปัญหาเนื่องจากหาคำตอบของปัญหาได้ดีที่สุด (Optimal) โดยใช้เวลาค้นหาคำตอบอย่างรวดเร็ว [10] โดยวิธีการวิวัฒนาการโดยใช้ผลต่าง (DE) ถูกนำเสนอครั้งแรกในปี 1997 [11] และถูกใช้ในงานวิจัยในปัญหาต่างๆ เช่น ปัญหาการการมอบหมายงาน (Assignment problem) [12], ปัญหาการขนส่ง (Transportation problem) [13-14], ปัญหาทำเลที่ตั้ง (Location Routing problem) [15], การจัดสมดุลสายการประกอบแบบเส้นตรงประเภทที่ 1 (SALBP-1) [7], [16-17] และการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 (UALBP-1) [18] แต่ยังไม่มีผู้วิจัยท่านใดนำวิธีการวิวัฒนาการโดยใช้ผลต่าง มาใช้สำหรับแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 2 (UALBP-2)

ในการวิจัยนี้ จึงได้เล็งเห็นถึงความสำคัญของปัญหาการจัดสมดุลสายการประกอบ เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด ที่มีความสำคัญต่อระบบทันเวลาพอดี (JIT) ด้วยการพัฒนาวิธีการเมตาฮิวริสติกโดยใช้วิธีการวิวัฒนาการโดยใช้ผลต่าง (DE) เข้ามาช่วยในการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2

1.2 วัตถุประสงค์

เพื่อพัฒนาวิธีการเมตาฮิวริสติก โดยใช้วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential evolution algorithm: DE) สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (U-Shaped assembly balancing problem type 2: UALBP-2)

1.3 สมมติฐานของการวิจัย

1.3.1 สายการผลิตที่พิจารณาจะเป็นสายการผลิตแบบต่อเนื่อง เป็นสินค้าชนิดเดียวกันที่มีหลายขั้นงาน

1.3.2 จำนวนสถานีงาน (Number of workstation) ทราบค่าที่แน่นอน

1.3.3 เวลาการทำงานทราบค่าเวลาแน่นอน

1.3.4 ไม่มีข้อจำกัดของการทำงาน นอกจากเงื่อนไขของลำดับงานก่อน-หลัง

1.3.5 ไม่พิจารณาการจัดสถานีงานแบบขนานกัน (Parallel workstation)

1.4 ขอบเขตของการวิจัย

ปัญหาที่ศึกษาเป็นปัญหาการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 2 ที่มีการผลิตสินค้าชนิดเดียว กรณีทราบค่าจำนวนสถานีงาน (m) และมีเป้าหมายเพื่อหารอบเวลาการผลิตที่น้อยที่สุด (c) ด้วยการพัฒนากระบวนการของวิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential evolution algorithm: DE) เพื่อช่วยหาคำตอบที่ดีที่สุด ด้วยโปรแกรมภาษา Java จากนั้นทำการทดสอบโดยใช้ชุดปัญหาของ scholl [19] ที่มีจำนวนงานตั้งแต่ 21-297 งาน จำนวน 15 ชุดปัญหา (101 ปัญหา) จาก www.assembly-line-balancing.de ที่สามารถแบ่งออกเป็นปัญหาขนาดกลางที่มีจำนวนงานตั้งแต่ 21-58 งาน จำนวน 10 ชุดปัญหา (50 ปัญหา) และปัญหาขนาดใหญ่ที่มีจำนวนงานตั้งแต่ 70-297 งาน จำนวน 5 ชุดปัญหา (51 ปัญหา) แล้วเปรียบเทียบผลกับวิธี Rules–based Heuristic ของ Li et al. [9]

1.5 ขั้นตอนการดำเนินการวิจัย

โดยมีขั้นตอนการดำเนินงานดังนี้

1.5.1 ศึกษางานวิจัยและค้นคว้าข้อมูลที่เกี่ยวข้องกับการจัดสมดุลสายการประกอบ

1.5.2 ศึกษาหลักการ และทฤษฎีที่เกี่ยวข้องกับวิธีการวิวัฒนาการโดยใช้ผลต่าง

1.5.3 พัฒนาหลักการเพื่อแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 โดยพัฒนาวิธีการวิวัฒนาการโดยใช้ผลต่าง เพื่อหารอบเวลาการผลิตที่น้อยที่สุด

1.5.4 วิเคราะห์และทำการเปรียบเทียบผลการวิจัย

1.5.5 สรุปและเสนอแนะ

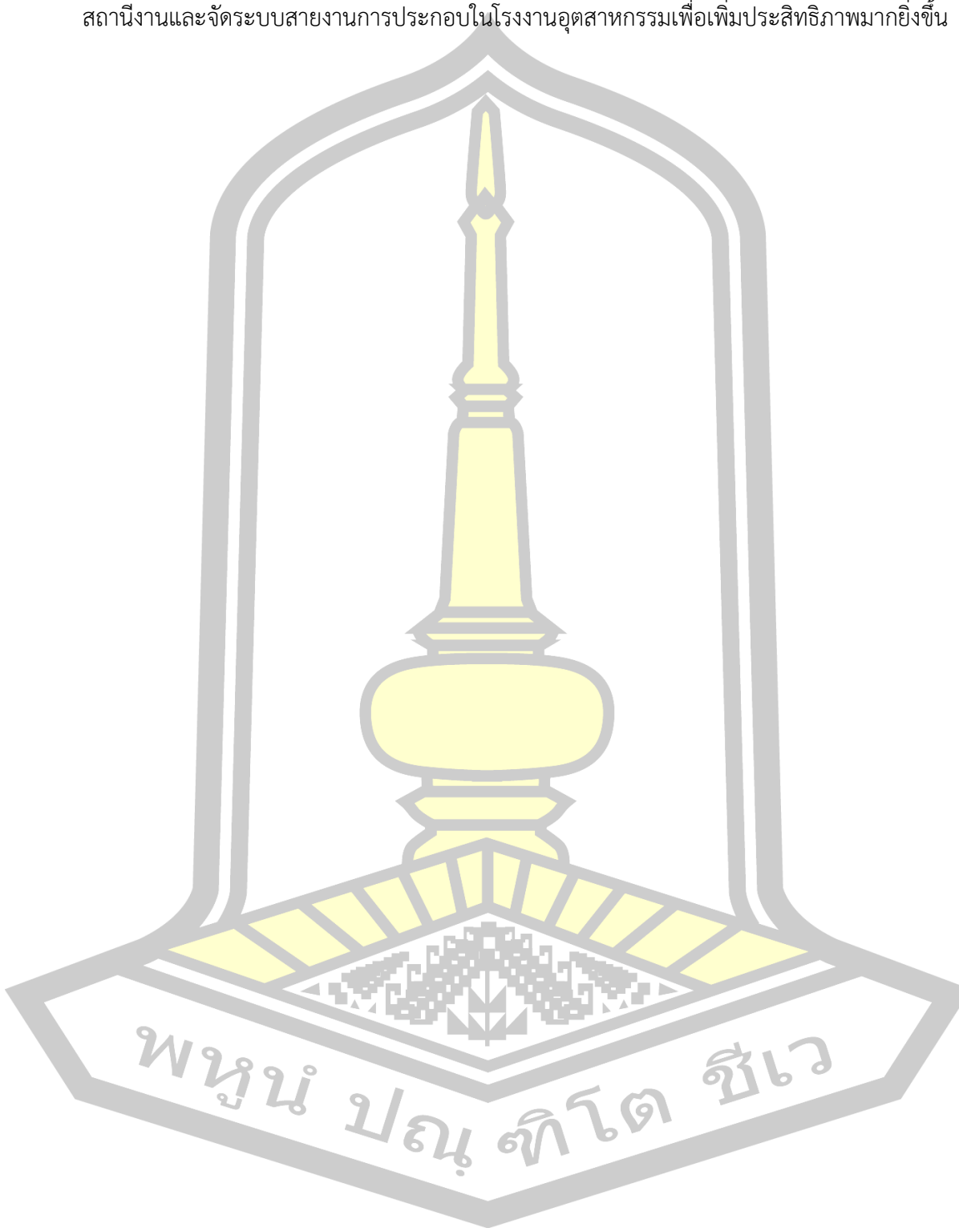
1.5.6 จัดทำรูปเล่มวิทยานิพนธ์ และนำเสนองานวิจัย

1.6 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย

1.6.1 สามารถวิเคราะห์และนำวิธีการวิวัฒนาการโดยใช้ผลต่าง ไปใช้กับลักษณะของปัญหาที่มีความซับซ้อนและขนาดที่ต่างกันได้อย่างเหมาะสม

1.6.2 เป็นการบูรณาการ วิธีการวิวัฒนาการโดยใช้ผลต่าง ในการพัฒนาหาคำตอบแบบใหม่

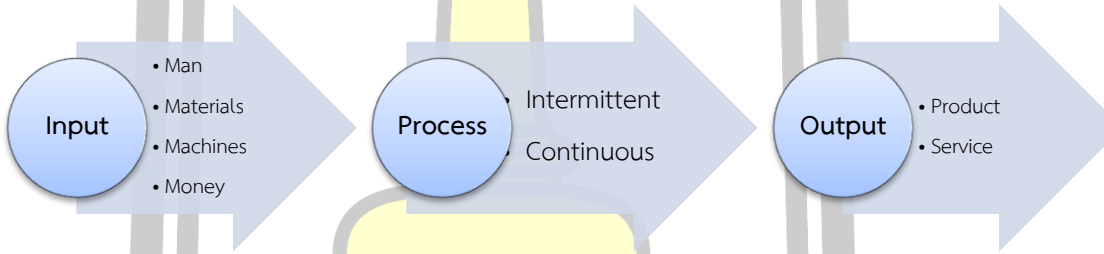
1.6.3 เป็นแนวทางในการประยุกต์วิธีการวิวัฒนาการโดยใช้ผลต่าง สำหรับการตัดสินใจจัดสถานีงานและจัดระบบสายงานการประกอบในโรงงานอุตสาหกรรมเพื่อเพิ่มประสิทธิภาพมากยิ่งขึ้น



บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ระบบการผลิต (Production system)

ระบบการผลิตโดยทั่วไป มีกลไกลพื้นฐาน เหมือนกับระบบทั่วๆไป มี 3 ส่วน ด้วยกัน คือ ปัจจัยการผลิต (Input) ได้แก่ คน (Man) วัตถุดิบต่างๆ (Materials) เครื่องจักร (Machines) เงิน (Money) ส่วน กระบวนการผลิต (Process) ได้แก่ การเตรียมวัตถุดิบต่างๆ การนำส่วนประกอบต่างๆ เข้าด้วยกัน การสร้างรูปทรง การตกแต่ง และส่วนที่เป็นผลผลิต (Output) ได้แก่ ผลิตภัณฑ์สำเร็จรูป (Products) ซึ่งผลผลิตจะออกมา ในรูปของสินค้า หรือบริการ (Service) ดังแสดงในภาพประกอบ 2



ภาพประกอบ 2 ระบบการผลิต

ระบบการผลิตสามารถแบ่งตามลักษณะของการผลิตได้เป็น 2 ประเภท ดังนี้

2.1.1 แบ่งประเภทตามลักษณะระบบการผลิต

2.1.1.1 การผลิตแบบไม่ต่อเนื่อง (Intermittent production) เป็นการผลิตที่วัตถุดิบไม่เลื่อนไหลไปตามสายการผลิต การผลิตจะผลิตเป็นช่วง ๆ สามารถจะผลิตสินค้าได้หลายชนิด โดยจะจัดซื้อเครื่องจักร เครื่องมือเป็นกลุ่ม ๆ เช่น งานกลึงจะอยู่กลุ่มงานกลึง งานเชื่อมก็จะอยู่กลุ่มของงานเชื่อม เป็นต้น การผลิตแบบไม่ต่อเนื่องจะผลิตสินค้าครั้งละมาก ๆ ก็ได้ ไม่มากก็ได้ ผลิตภัณฑ์จะทำเป็นชิ้น ๆ จะทำทีละขั้นตอน เมื่อเสร็จเรียบร้อยในแต่ละขั้นตอนแล้วก็นำชิ้นส่วนมาประกอบกันเป็นผลิตภัณฑ์ เช่น ตู้ เตียง เก้าอี้ การซ่อมรถยนต์ งานก่อสร้างต่าง ๆ เป็นต้น ระบบการผลิตจะเป็นไปตามความเหมาะสมของผู้ดำเนินงาน การติดตั้งเครื่องจักรก็จะติดตั้งตามกรรมวิธีการผลิต จึงเป็นผลทำให้มีความต้องการการใช้พื้นที่ในการเก็บวัสดุในการผลิตมากขึ้น ทั้งนี้เพราะการผลิตแบบไม่

ต่อเนื่องนี้มีจุดพักชิ้นงานหลายจุด ฉะนั้นผู้ผลิตจะต้องกำหนดแนวทางการวางผังโรงงานให้สอดคล้องกับระบบการผลิต ตลอดจนกำหนดวิธีการขนย้ายวัสดุให้เหมาะสมจึงจะทำให้การผลิตมีประสิทธิภาพ

2.1.1.2 การผลิตแบบต่อเนื่อง (Continuous production) เป็นกระบวนการผลิตสินค้าที่มีจำนวนมาก มีกระบวนการผลิตที่ต่อเนื่องและค่อนข้างแน่นอน การผลิตจะเริ่มจากการป้อนวัตถุดิบเข้าไปในสายการผลิต และตำแหน่งของการทำงานต่างๆ จะถูกกำหนดตามลำดับขั้นตอนเป็นสายการผลิต ที่แบ่งเป็นจุดการทำงานหรือหลายๆสถานีต่อเนื่องกันตามลำดับ จนเป็นผลิตภัณฑ์ ทำให้ระยะเวลาระหว่างการผลิตสั้นลง ต่างจากการผลิตแบบไม่ต่อเนื่อง ซึ่งผลิตเป็นรุ่น รุ่นละจำนวนมากๆ จึงทำให้ใช้เวลาระหว่างการผลิตจนออกมาเป็นผลิตภัณฑ์ยาวนานกว่า

ซึ่งการผลิตแบบต่อเนื่องยังสามารถแบ่งกระบวนการผลิตตามชนิดของการผลิตและตามจำนวนชนิดของสินค้า ดังภาพประกอบ 3 ได้ดังนี้ [1]

1) แบ่งประเภทตามลักษณะกระบวนการผลิต

(1) สายการผลิตแบบส่งถ่าย (Transfer line)

จะอาศัยการทางานของเครื่องจักรอัตโนมัติ เพื่อส่งถ่ายวัสดุและชิ้นงาน แรงงานคนที่ใช้ส่วนใหญ่จะเป็นการควบคุมการทำงานของเครื่องจักร ไม่ว่าแผนการผลิตจะมีการเปลี่ยนแปลงอย่างไร สายการผลิตจะไม่มีการเปลี่ยนแปลง กำลังการผลิตจะคงที่และผลผลิตขึ้นอยู่กับชั่วโมงการทำงานของเครื่องจักร เช่น การผลิตสุรา การกลั่นน้ำมัน เป็นต้น

(2) สายการผลิตแบบงานประกอบ (Assembly line)

สายการผลิตแบบงานประกอบ ส่วนใหญ่จะใช้แรงงานคนเป็นหลักในการประกอบชิ้นส่วนต่างๆเข้าด้วยกัน โดยใช้คนประจำในสถานีต่างๆและสามารถปรับเปลี่ยนชนิดของเครื่องจักรในแต่ละสถานีงานได้ตามลักษณะการทำงานและพื้นที่ในการทำงาน ซึ่งการผลิตแบบนี้สามารถเปลี่ยนแปลงสายการผลิตในสถานีต่างๆ เพื่อให้สอดคล้องกับแผนการผลิต เช่น โรงงานตัดเย็บเสื้อผ้า เครื่องใช้ไฟฟ้า เป็นต้น

2) แบ่งประเภทตามจำนวนชนิดของสินค้า

(1) การผลิตแบบสินค้าชนิดเดียว (Single model line)

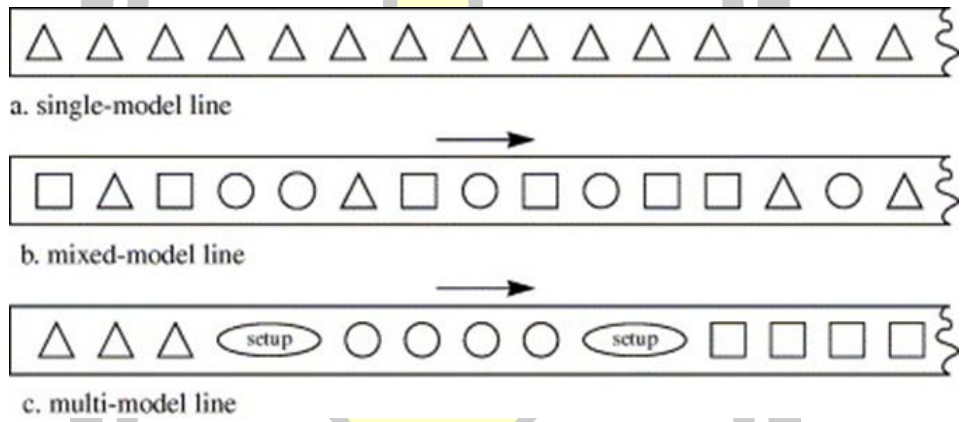
จะเป็นสายการผลิตผลิตภัณฑ์ชนิดใดชนิดหนึ่งเพียงสินค้าเดียวโดยเฉพาะและเป็นผลิตภัณฑ์รูปแบบเดียว ซึ่งเป็นสินค้าที่มีความต้องการจากลูกค้ามากจึงทำให้สายการผลิตสามารถผลิตสินค้าชนิดเดียวได้ตลอดเวลา แสดงดังภาพประกอบ 3

(2) การผลิตแบบสินค้าแบบผสม (Mixed model line)

เป็นสายการผลิตที่ผลิตผลิตภัณฑ์ตั้งแต่ 2 ชนิดขึ้นไปเหมือนกับการผลิตแบบสินค้าหลายชนิด แต่จะต่างกันตรงที่วัตถุดิบต่างๆจะเข้าไปในสายการผลิตพร้อมๆกันและจะผลิตสินค้าต่างชนิดพร้อมๆกัน ไม่มีการแบ่งว่าต้องทำผลิตภัณฑ์ชุดไหนก่อน และระหว่างการผลิตจะไม่มีการปรับเปลี่ยนสายการผลิตใหม่ แสดงดังภาพประกอบ 4

(3) การผลิตแบบสินค้าแบบหลายชนิด (Multi model line)

เป็นสายการผลิตที่ผลิตผลิตภัณฑ์ตั้งแต่ 2 ชนิดขึ้นไป ซึ่งผลิตภัณฑ์แต่ละชนิดจะมีกระบวนการผลิตที่ใกล้เคียงกัน สามารถผลิตบนสายงานประกอบเดียวกันได้ โดยในการผลิตจะผลิตที่ละชุดผลิตภัณฑ์ และในช่วงที่มีการเปลี่ยนผลิตภัณฑ์อาจจะต้องมีการปรับเปลี่ยนสายการผลิตใหม่ แสดงดังภาพประกอบ 5



ภาพประกอบ 3 สายการประกอบที่แบ่งกระบวนการผลิตตามจำนวนชนิดของสินค้า [20]



ภาพประกอบ 4 สายการประกอบเย็บเสื้อผ้าโรงงานกรณีศึกษาเป็นการผลิตแบบสินค้าชนิดเดียว



ภาพประกอบ 5 สายการประกอบชิ้นส่วนเครื่องปรับอากาศเป็นการผลิตแบบสินค้าหลายชนิด



ภาพประกอบ 6 สายการประกอบเครื่องปรับอากาศที่มีความหลากหลายของรุ่นเป็นการผลิตสินค้าแบบผสม

2.2 หลักการและทฤษฎีที่เกี่ยวข้อง

2.2.1 การจัดสมดุลสายการประกอบ (Assembly line balancing)

การจัดสมดุลสายการประกอบ หมายถึง การจัดรูปแบบสถานีงานในระบบสายงานการประกอบแบบต่อเนื่อง ชิ้นส่วนผลิตภัณฑ์ที่จะทำการประกอบจะเคลื่อนย้ายตามสถานีงานต่างๆ เมื่อชิ้นส่วนผลิตภัณฑ์ดังกล่าวเข้าสู่สถานีงานและมีการประกอบในสถานีนั้นเกิดขึ้นตามลำดับงานก่อนหลัง เมื่อหมดขั้นตอนการประกอบในสถานีนั้นแล้วชิ้นส่วนนั้นจะเคลื่อนย้ายไปยังสถานีถัดไป ในขณะเดียวกันที่สถานีเดิมก็จะมีชิ้นส่วนผลิตภัณฑ์จากหน่วยก่อนหน้าเข้ามาแทน

ดังนั้นการจัดสมดุลสายการประกอบจึงเป็นการพยายามจัดสถานีงานต่างๆให้มีอัตราการผลิตให้ใกล้เคียงกันมากที่สุด ซึ่งอัตราการผลิตนั้นจะถูกกำหนดโดยรอบเวลาการผลิต (Cycle time) ซึ่งหมายถึง เวลาที่ผลิตสินค้าเสร็จออกมาในแต่ละชิ้น โดยในบางครั้งการกำหนดรอบการผลิตสามารถกำหนดจากความต้องการของลูกค้าได้ เช่น ลูกค้าต้องการ 200 หน่วย/วัน และ 1 วันทำงาน 8 ชั่วโมง ดังนั้นจะต้องผลิตสินค้าออกมา 2.4 นาทีต่อหน่วย และเมื่อนำเวลารอบการผลิตที่ได้ไปจัดสมดุลสายการประกอบ พบว่า แต่ละสถานีงานที่ได้ใช้เวลาน้อยกว่ารอบเวลาการผลิต ซึ่งผลต่างของเวลาในแต่ละสถานีงานกับรอบเวลาการผลิต เรียกว่า เวลาว่างงาน (Idle time : ID) เป็นต้น

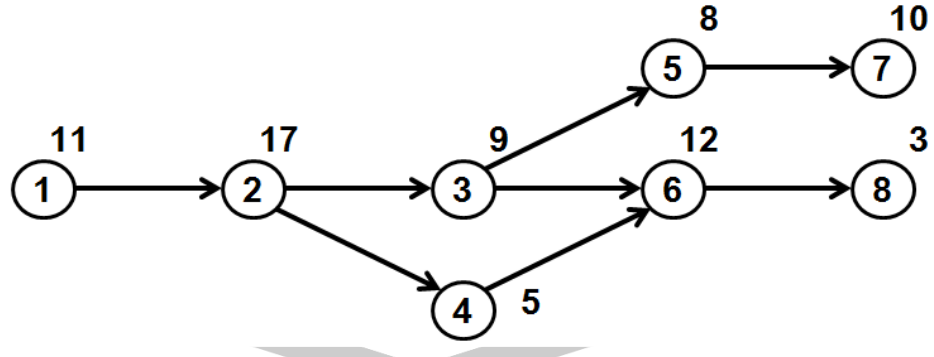
ในการจัดสมดุลสายการประกอบ จะเป็นการพยายามจัดสถานีงานต่างๆให้มีอัตราการทำงานหรือเวลาที่ใช้ในการผลิตสินค้าแต่ละชิ้นเท่าๆกันมากที่สุด ซึ่งจะเริ่มต้นด้วยการกำหนดรอบเวลาการผลิตตามลำดับขั้นงานต่างๆ และเวลาการทำงานของแต่ละขั้นตอน จากนั้นจะพยายามรวมขั้นงานเข้าด้วยกันให้เป็นสถานีงาน โดยจัดให้แต่ละสถานีงานมีเวลาการทำงานไม่เกินรอบเวลาการผลิตและมีเวลาว่างงานน้อยที่สุด ถ้าสามารถจัดให้แต่ละสถานีงานมีความสมดุลกันก็ส่งผลให้สายการประกอบนั้นมีประสิทธิภาพสูง ซึ่งการจัดสมดุลสายการประกอบจะมีองค์ประกอบต่างๆดังต่อไปนี้

2.2.1.1 งาน (Task) หรือ การดำเนินงาน (Operation) คือ ส่วนหนึ่งของเนื้องานทั้งหมดในกระบวนการประกอบ เวลาที่ใช้ไปในการทำแต่ละงานเรียกว่า เวลางาน (Task time) หรือเวลาดำเนินงาน (Operation time) แต่ละงานถือเป็นเนื้องานที่ไม่สามารถแบ่งย่อยลงไปได้อีก โดยปราศจากการทำงานเพิ่มเติมที่ไม่จำเป็น

2.2.1.2 สถานีงาน (Station หรือ Workstation) คือ ส่วนหนึ่งของสายการประกอบที่จะรวมงาน (Task) ตั้งแต่งานเดี่ยว หรือหลายๆงานมาอยู่ร่วมกัน ซึ่งจะรับผิดชอบในการทำงานประกอบ (เซตของงาน) ที่กำหนดให้จำนวนหนึ่ง หรือที่เรียกกันว่า ภาระงานของสถานี (Workstation load)

2.2.1.4 รอบเวลาการผลิต (Cycle time) คือ ระยะเวลาที่มากที่สุดที่สถานีงานจะใช้ในการทำการประกอบหรือทำเซตของงานที่กำหนดให้

2.2.1.5 แผนภาพลำดับขั้นตอนการทำงาน (Precedence diagram) คือ แผนภาพแสดงลำดับขั้นตอนการทำงาน ซึ่งจะสามารถบอกลำดับขั้นตอนงานก่อน-หลัง โดยมีลูกศรกำหนดทิศทางการดำเนินงานและวงกลมหมายถึงงานซึ่งการดำเนินงานจะเริ่มจากด้านซ้ายสุดของผังผ่านกระบวนการประกอบต่างๆจนเป็นผลิตภัณฑ์ที่อยู่ด้านขวามือ ดังตัวอย่างภาพประกอบ 7

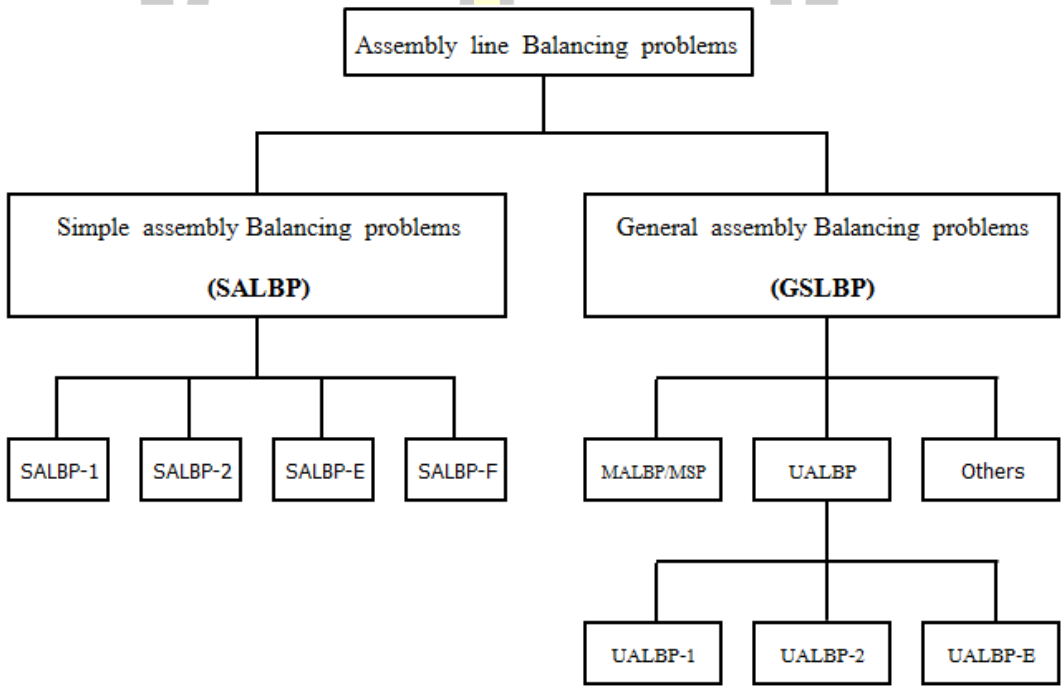


ภาพประกอบ 7 แผนภาพลำดับงานก่อน-หลังของปัญหา Bowman [21]

หมายเลขในวงกลมหมายถึงชื่อของแต่ละงานมีทั้งหมดจำนวน 8 งาน และหมายเลขด้านนอกบนวงกลมหมายถึงเวลาของแต่ละงาน

2.2.2 ประเภทของปัญหาการจัดสมดุลสายการประกอบ (Classification of assembly line balancing problem)

สามารถแบ่งประเภทของปัญหาการจัดสมดุลสายการประกอบ [1] ได้แสดงดังภาพประกอบ 8



ภาพประกอบ 8 ประเภทของปัญหาการจัดสมดุลสายการประกอบ [1]

2.2.2.1 ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรง (Simple assembly line balancing problem: SALBP)

เป็นปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงที่มีการผลิตสินค้าชนิดเดียวและผลิตคราวละมากๆมีหลายขั้นตอนงาน มีรอบเวลาการผลิตคงที่และมีเวลาการทำงานแน่นอนหรือมีเวลามาตรฐานของงานนั้นๆ ไม่มีข้อจำกัดของการกำหนดงานนอกจากเงื่อนไขลำดับงานก่อนหลัง สายการผลิตเป็นแบบต่อเนื่อง ทุกสถานีงานจะมีจำนวนคนงานและเครื่องจักรที่เท่าๆกัน มีอัตราการปล่อยชิ้นงานคงที่หรือทราบรอบเวลาการผลิตที่แน่นอน ซึ่งสามารถแบ่งประเภทของปัญหา

SALBP ได้อีก 4 ประเภทคือ SALBP–1, SALBP–2, SALBP–E และ SALBP–F ซึ่งแต่ละประเภทมีวัตถุประสงค์แตกต่างกันสามารถสรุปได้ดังตาราง 1

ตาราง 1 สรุปปัญหา SALBP ของแต่ละประเภท

ประเภท	ทราบค่า	วัตถุประสงค์
SALBP–1	รอบเวลาการผลิต (c)	จำนวนสถานีงาน (m) ต่ำที่ที่สุด
SALBP–2	จำนวนสถานีงาน (m)	รอบเวลาการผลิต (c) ต่ำที่ที่สุด
SALBP–E	รอบเวลาการผลิต (c) และจำนวนสถานีงาน (m)	ประสิทธิภาพสูงที่สุด (E)
SALBP–F	รอบเวลาการผลิต (c) และจำนวนสถานีงาน (m)	ถ้า feasible แสดงว่าเป็นปัญหาที่สามารถหาคำตอบได้

2.2.2.2 ปัญหาการจัดสมดุลสายการประกอบแบบทั่วไป (General assembly line balancing problem: GALBP)

เป็นปัญหาที่มีความยุ่งยากซับซ้อนขึ้นและสามารถแบ่งออกได้ 3 ชนิด ดังนี้

1) ปัญหา Mixed model assembly line balancing problem and mixed model sequencing problem (MALBP/MSP)

ปัญหา MALBP คือ ปัญหาการจัดสมดุลสายการประกอบที่มีการผลิตผลิตภัณฑ์แบบผสม มีเวลาการทำงานที่แน่นอน โดยการจัดงานลงสถานีงานจะทำการพิจารณาเวลาที่ต่างกันสำหรับสินค้าแต่ละอย่าง เพื่อหาสถานีงานหรือรอบเวลาการผลิตที่เหมาะสมที่สุด

ปัญหา MSP คือ ปัญหาการจัดสมดุลสายการประกอบที่มีการจัดลำดับการผลิตให้สินค้าทุกรุ่นที่จะทำการผลิตให้เกิดความเหมาะสมที่สุด เพื่อลดเหตุการณ์ต่างๆที่ไม่มีประสิทธิภาพในสายการผลิต เช่น การหยุดซ่อม เป็นต้น

2) ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู (U-line assembly line balancing problem: UALBP)

เป็นปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ที่มีการผลิตสินค้าชนิดเดียว สายการประกอบจะถูกจัดเป็นรูปตัวยู โดยสามารถจัดงานเริ่มต้นและงานสุดท้ายของแผนภูมิลำดับงานก่อน-หลัง ลงในสถานีเดียวกันได้ และจะมีการทำงานในสถานีทั้งสองฝั่ง ทำให้จำนวนของสถานีงานมีจำนวนน้อยกว่าหรือเท่ากับ SALBP เมื่อเปรียบเทียบค่าที่เหมาะสมของปัญหา

UALBP และ SALBP จะได้ว่า UALBP นั้นจะมีประสิทธิภาพของสายการประกอบ (Efficiency line) ที่ดีกว่าหรือเท่ากับ SALBP ซึ่งในปัญหาการจัดสมดุลสายการประกอบแบบตัวยู สามารถแบ่งปัญหาออกได้ 3 ประเภท คือ UALBP-1, UALBP-2 และ UALBP-E ซึ่งแต่ละประเภทมีวัตถุประสงค์แตกต่างกันสามารถสรุปได้ดังตาราง 2

ตาราง 2 สรุปปัญหา UALBP ของแต่ละประเภท

ประเภท	ทราบค่า	วัตถุประสงค์
UALBP-1	รอบเวลาการผลิต (c)	จำนวนสถานีงาน (m) ต่ำที่สุด
UALBP-2	จำนวนสถานีงาน (m)	รอบเวลาการผลิต (c) ต่ำที่สุด
UALBP-E	รอบเวลาการผลิต (c) และจำนวนสถานีงาน (m)	ประสิทธิภาพสูงที่สุด (E)

3) ปัญหาอื่นๆ

ในปัญหา GALBP จะเป็นปัญหาที่มีขอบเขตที่กว้างกว่าในปัญหาแบบอื่น ซึ่งอาจจะพิจารณาตามเงื่อนไขอื่นๆที่เกี่ยวข้อง เช่น การเลือกเครื่องมือและอุปกรณ์ (Equipment selection), การผลิตในแบบอื่นๆ (Processing alternatives) และข้อจำกัดของการกำหนดงาน (Assignment restrictions) ฯลฯ [20]

2.2.3 วิธีการการวัดการแก้ปัญหาการจัดสมดุลสายการประกอบ

2.2.3.1 ประสิทธิภาพสายการผลิต (Efficiency of line balance: E)

ประสิทธิภาพของสายการผลิต จะเป็นดัชนีที่แสดงให้เห็นความสามารถของการจัดงานลงในสถานีงานเพื่อให้เกิดเวลาสูญเปล่าน้อยที่สุด

$$E = (\sum t/mc) \times 100 \quad (2.1)$$

โดย E = เป็นค่าเปอร์เซ็นต์, $\sum t$ = เวลารวมขั้นงาน, m = จำนวนสถานีงาน, c = รอบเวลาการผลิต

2.2.3.2 การสูญเสียความสมดุล (Balance delay: BD)

เป็นเครื่องชี้ถึงประสิทธิภาพที่เกิดความสูญเสียไม่สามารถทำงานได้เต็มประสิทธิภาพ โดยพิจารณาที่เวลาสูญเปล่า (Idle time) ของการจัดงานลงสถานีงาน

$$BD = \left((mc - \sum t)/mc\right) \times 100 \quad (2.2)$$

หรือ $$BD = 100 - E \quad (2.3)$$

โดย BD = เป็นค่าเปอร์เซ็นต์, $\sum t$ = เวลารวมขั้นงาน, m = จำนวนสถานีงาน, c = รอบเวลาการผลิต

2.2.3.3 เวลาสูญเปล่า (Idle time: ID)

$$ID = mc - \sum t \tag{2.4}$$

โดย ID = เวลาสูญเปล่า, $\sum t$ = เวลารวมขั้นงาน, m = จำนวนสถานีงาน, c = รอบเวลาการผลิต

2.2.4 วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential evolution algorithm: DE)

วิธีการวิวัฒนาการโดยใช้ผลต่างถูกนำเสนอโดย Storn and Price [11] วิธีการนี้มีกระบวนการทำงานคล้ายๆ กับวิธีการวิวัฒนาการแบบอื่นๆ โดยวิธีการนี้เป็นวิธีการที่ใช้ในการหาคำค่าที่ดีที่สุดโดยใช้ประชากรในแต่ละรุ่นในการหาคำตอบและมีการพัฒนาคำตอบจากรุ่นสู่รุ่น ในกระบวนการการวิวัฒนาการโดยใช้ผลต่างนี้จะใช้เลขจำนวนจริงในการแทนคำตอบในแต่ละรุ่นของสมาชิกของประชากรในแต่ละรุ่น ในวิธีการวิวัฒนาการโดยใช้ผลต่างจะเรียกสมาชิกของประชากรในแต่ละรุ่นว่าเวคเตอร์ และ เรียกส่วนประกอบในเวคเตอร์ว่าพิกัด และจำนวนพิกัดในแต่ละเวคเตอร์จะเรียกว่าขนาดของเวคเตอร์ (Dimension)

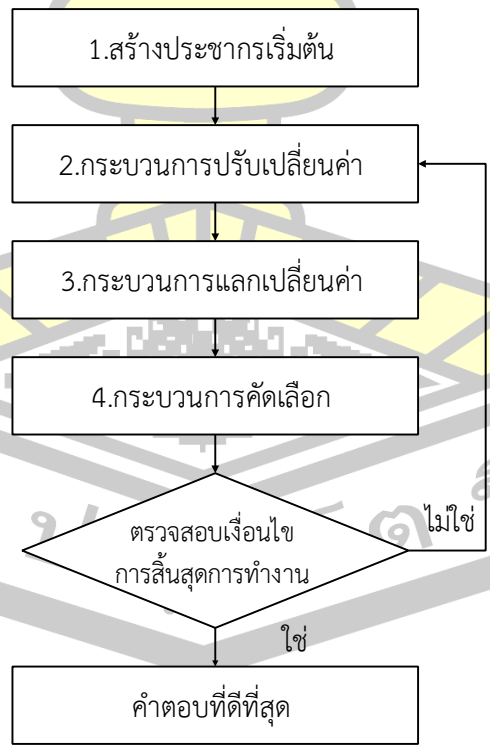
2.2.4.1 กระบวนการขั้นตอนวิธีวิวัฒนาการโดยใช้ผลต่างอย่างง่าย (Simple differential evolution)

วิธีวิวัฒนาการโดยใช้ผลต่างเป็นวิธีการหาคำตอบที่มีโครงสร้างที่ไม่ซับซ้อน สามารถหาคำตอบได้อย่างรวดเร็ว มีประสิทธิภาพ และจะใช้ค่าจริง (Floating point number) แทนในการคำนวณ ซึ่งต่างจากขั้นตอนวิธีเชิงพันธุกรรมที่ต้องแปลงตัวแปรการตัดสินใจ (Decision variables) ให้เป็นรหัสเลขฐานสอง (binary code) โดยในการพัฒนาคำตอบของวิธีวิวัฒนาการโดยใช้ผลต่างนั้น จะใช้วิธีการปรับเปลี่ยนค่าในพิกัด (Mutation) ซึ่งสามารถดำเนินการได้ด้วยการหาผลต่างของเวกเตอร์ที่สุ่มมา 2 เวกเตอร์ของประชากรและนำผลต่างนี้ไปคูณกับปัจจัยขยายผลต่าง (Scaling factor: *F*) แล้วจึงนำไปบวกกับเวกเตอร์ที่สุ่มมาอีกเวกเตอร์หนึ่ง โดยประชากรรุ่นที่กำลังจะถูกปรับเปลี่ยนค่าในพิกัดเรียกว่า เวกเตอร์เป้าหมาย (Target vector) และประชากรที่ถูกปรับเปลี่ยนแล้วจะเรียกว่า มิวแทนต์เวกเตอร์ เมื่อประชากรถูกปรับเปลี่ยนจนครบจำนวนประชากรที่ตั้งไว้ล่วงหน้า (Number of population: *NP*) จากนั้นจะใช้วิธีการแลกเปลี่ยนค่าในพิกัด (Crossover หรือ Recombination) ระหว่างเวกเตอร์เป้าหมายและมิวแทนต์เวกเตอร์โดยใช้ความน่าจะเป็นแลกเปลี่ยนค่าในพิกัดซึ่งเป็นตัวแปรที่ทราบค่าที่ควบคุมได้เรียกว่าอัตราการปรับเปลี่ยน (Crossover rate: *CR*) เวกเตอร์ที่ผ่านการแลกเปลี่ยนค่าในพิกัดแล้วจะถูกเรียนว่าไทรอัลเวกเตอร์ (Trial vector)

จากนั้นจะนำไทรอัลเวกเตอร์ที่ได้ไปหาค่าฟิตเนสฟังก์ชันหรือฟังก์ชันเป้าหมาย แล้วเปรียบเทียบค่าฟิตเนสฟังก์ชันของไทรอัลเวกเตอร์กับเวกเตอร์เป้าหมาย เวกเตอร์ใดดีกว่าจะถูกใช้เป็นเวกเตอร์เป้าหมายในรอบถัดไป [22] โดยได้แสดงขั้นตอนวิธีแสดงอัลกอริทึมของวิธีวิวัฒนาการโดยใช้ผลต่างและลำดับขั้นตอนของวิธีวิวัฒนาการโดยใช้ผลต่างในภาพประกอบ 9 และ 10

1. เริ่มต้น (begin)
2. t ⟵ 0;
3. สร้างประชากรเริ่มต้น (Initial population)
4. ขณะที่ (while) ยังไม่ผ่านเงื่อนไขการหยุด (Not termination); วนลูป
5. การปรับเปลี่ยนค่าพิกัด (Mutation);
6. การแลกเปลี่ยนพิกัด (Crossover หรือ Recombination);
7. คัดเลือก (Selection);
8. t ⟵ t+1;
9. จบกระบวนการวนลูป (end while)
10. จบกระบวนการ (End)

ภาพประกอบ 9 ขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง [22]



ภาพประกอบ 10 ลำดับขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง [22]

1) การสร้างประชากรเริ่มต้น (Initial population)

การสร้างประชากรของคำตอบเริ่มต้นสามารถทำได้โดยการสุ่มค่าประชากรหรือเวกเตอร์ตั้งต้น (Initial population) โดยกำหนดขอบเขตของตัวแปรการตัดสินใจเพื่อให้แน่ใจว่าขอบเขตของตัวแปรการตัดสินใจจะคลอบคลุมถึงคำตอบที่ดีที่สุด และกำหนดให้โอกาสที่จะถูกเลือกของคำตอบที่มีค่าสม่ำเสมอ (Uniform probability distribution) ที่มีมิติเท่ากับ *D* ซึ่งอาจหมายถึงพิกัดหรือตำแหน่ง และจำนวนคำตอบที่เป็นไปได้เริ่มต้นเท่ากับ *NP* โดยแต่ละคำตอบหรือแต่ละเวกเตอร์เป้าหมายเรียกว่า เวกเตอร์การตัดสินใจ (Decision vector) ซึ่งจะนำมาคำนวณเพื่อหาค่าคำตอบของสมการเป้าหมายของเวกเตอร์เป้าหมาย (Target Vector) หรือค่าฟิตเนส (Fitness value)

2) การปรับเปลี่ยนค่าในพิกัด (Mutation)

การปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากร โดยมีจุดประสงค์หลักเพื่อแลกเปลี่ยนพิกัดภายในเวกเตอร์ ซึ่งในการปรับเปลี่ยนค่าพิกัดนั้นสามารถทำได้โดยทำการสุ่มเวกเตอร์มา 3 เวกเตอร์ (ตัวแทนผลลัพธ์) $X_{r1,G}$, $X_{r2,G}$ และ $X_{r3,G}$ มาดำเนินการปรับเปลี่ยนพิกัด โดยเวกเตอร์ที่ถูกปรับเปลี่ยนค่าพิกัดไปแล้วจะถูกเรียกว่า มิวแทนเวกเตอร์ (Mutant vector) โดยที่ค่ามิวแทนเวกเตอร์ $(V_{i,G+1})$ จะมีค่าเท่ากับ $X_{r1,G}$ รวมกับผลต่างของเวกเตอร์ $X_{r2,G}$ และ $X_{r3,G}$ คูณด้วยค่าแฟกเตอร์ในการขยายผลต่าง (Scaling factor: *F*) โดยสามารถอธิบายการปรับค่าในพิกัดของเวกเตอร์ได้ดังสมการ 2.5

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \quad (2.5)$$

เมื่อ $V_{i,G+1}$ = มิวแทนเวกเตอร์ (Mutant vector)

$X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$ = เวกเตอร์สุ่ม (Random vector) คือเวกเตอร์ที่ทำการสุ่มเลือกมา 3 เวกเตอร์

F = แฟกเตอร์ในการขยายผลต่าง (Scaling factor) คือจำนวนจริงที่มีค่าคงที่และมีค่าระหว่าง 0 ถึง 2

3) การแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (Crossover หรือ Recombination)

การแลกเปลี่ยนค่าในพิกัดเวกเตอร์เป็นการเพิ่มความหลากหลายในการหาคำตอบของปัญหาโดยขั้นตอนนี้จะเป็นการสร้างไทรอัลเวกเตอร์ (Trial vector) โดยใช้สมการ 2.6 ซึ่งจะใช้ในการสุ่มตัวเลขที่อยู่ระหว่าง 0-1 โดยมีขนาดตำแหน่งเท่ากับ *D* และจำนวนประชากรเท่ากับ *NP* หากตัวเลขสุ่มของทุกเวกเตอร์และทุกตำแหน่ง มีค่าน้อยกว่าหรือเท่ากับค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (*CR*) ค่าในตำแหน่งพิกัดของเวกเตอร์นั้นจะมีค่าเท่ากับค่าของมิวแทนต์เวกเตอร์ใน

ตำแหน่งและพิกัดเดียวกัน และในกรณีที่เป็นอย่างอื่น (มีค่ามากกว่าค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (*CR*)) ค่าในตำแหน่งและพิกัดนั้นๆจะมีค่าเท่ากับเวกเตอร์เป้าหมาย (Target vector)

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{if } (rand\ (j)\ [0,1) \le CR) \\ X_{ji,G+1} & \text{if } (rand\ (j) > CR) \end{cases} \quad (2.6)$$

เมื่อ $U_{ji,G+1}$ = ไทรอัลเวกเตอร์ (Trial vector)

$V_{ji,G+1}$ = มิวแทนเวกเตอร์ (Mutant vector)

$X_{ji,G+1}$ = เวกเตอร์เป้าหมาย (Target vector)

$rand\ (j)$ = การสุ่มเลขจำนวนจริงที่มีค่าตั้งแต่ 0-1 ให้กับทุกตำแหน่ง j; j = 1,2,...,D

CR = อัตราแลกเปลี่ยนตำแหน่งพิกัดที่มีค่าเป็นเลขจำนวนจริงระหว่าง 0-1

4) การคัดเลือก (Selection)

การคัดเลือกเป็นขั้นตอนของการเลือกเวกเตอร์เป้าหมายในรอบถัดไปหรือการเลือกประชากรสำหรับการดำเนินในรุ่นต่อไป โดยเวกเตอร์ที่ให้คำตอบที่ดีกว่าจะถูกเลือกโดยใช้วิธีการเปรียบเทียบค่าคำตอบของสมการเป้าหมาย (Objective value) ที่ได้จากเวกเตอร์เป้าหมาย (Target vector) กับค่าคำตอบที่ได้จากไทรอัลเวกเตอร์ (Trial vector) ซึ่งถ้าหากค่าคำตอบที่ได้จากไทรอัลเวกเตอร์มีค่าน้อยกว่าหรือเท่ากับค่าคำตอบที่ได้จากเวกเตอร์เป้าหมาย จะเลือกไทรอัลเวกเตอร์เป็นประชากรในรุ่นถัดไป และในกรณีที่เป็นอย่างอื่น (ค่าคำตอบที่ได้จากเวกเตอร์เป้าหมายมีค่ามากกว่าค่าคำตอบที่ได้จากไทรอัลเวกเตอร์) จะทำการเลือกเวกเตอร์เป้าหมายเป็นประชากรรุ่นถัดไป ดังสมการที่ 2.7

$$X_{ji,G+1} = \begin{cases} U_{ji,G+1} & \text{if } f(U_{ji,G+1}) \le f(X_{ji,G+1}) \\ X_{ji,G} & \text{otherwise} \end{cases} \quad (2.7)$$

เมื่อ $X_{ji,G+1}$ = เวกเตอร์เป้าหมาย (Target vector)

$U_{ji,G+1}$ = ไทรอัลเวกเตอร์ (Trial vector)

พหุปณฺฑิโต ชีเว

2.2.4.2 ตัวแปรที่ทราบค่าในวิธีวิวัฒนาการโดยใช้ผลต่าง (Differential evolution algorithm parameters)

ตัวแปรที่ทราบค่าในการแก้ปัญหาด้วยวิธีวิวัฒนาการโดยใช้ผลต่างนั้น จะมีตัวแปรที่ทราบค่าหลักที่เกี่ยวข้อง คือ แฟกเตอร์ในการขยายผลต่าง (*F*) และค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (*CR*) โดยมีรายละเอียดของแต่ละตัวแปรดังนี้ [22]

1) แฟกเตอร์ในการขยายผลต่าง (Scaling factor: F)

เป็นค่าที่จะทำให้เวกเตอร์แต่ละตัวเปลี่ยนแปลงไปอย่างช้าหรือรวดเร็ว โดยถ้าค่าแฟกเตอร์ในการขยายผลต่างมีค่าสูงจะทำให้มีความรวดเร็วในการหาคำตอบในขอบเขตที่กว้าง แต่อาจทำให้ลดประสิทธิภาพของการค้นหาคำตอบอย่างละเอียดในตำแหน่งใกล้เคียงกับเวกเตอร์เดิม ในทางตรงกันข้ามหากแฟกเตอร์ในการขยายผลต่างมีค่าน้อยจะทำให้การหาคำตอบช้า แต่จะสามารถเพิ่มความละเอียดในการหาคำตอบในพื้นที่ใกล้เคียงได้ดี ซึ่งค่าแฟกเตอร์ในการขยายผลต่างที่นิยมใช้กันจะมีค่าอยู่ในช่วง [0,2]

2) ค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (CR)

ในการหาคำตอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่างนั้น สิ่งที่มีผลที่ทำให้เกิดความหลากหลายของคำตอบคือการแลกเปลี่ยนตำแหน่งพิกัด โดยในการแลกเปลี่ยนตำแหน่งพิกัดระหว่างเวกเตอร์เป้าหมายและมิวแทนต์เวกเตอร์นั้นจะใช้ค่าสุ่มเพื่อเปรียบเทียบกับค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด โดยถ้าหากมีค่าอัตราแลกเปลี่ยนตำแหน่งพิกัดที่สูงก็จะทำให้มีการเปลี่ยนแปลงของเวกเตอร์ที่หลากหลายยิ่งขึ้น โดยทั่วไปจะนิยมใช้อัตราแลกเปลี่ยนตำแหน่งพิกัดที่ 0.8

2.3 งานวิจัยที่เกี่ยวข้องกับการจัดสมดุลสายการประกอบ

ในช่วงครึ่งศตวรรษที่ผ่านมาได้มีผู้ศึกษางานวิจัยทางด้านการจัดสายการประกอบแบบตัวยู, วิธีวิวัฒนาการโดยใช้ผลต่างและไว้ทั้งภายในประเทศและต่างประเทศเป็นจำนวนมาก ซึ่งมีรายละเอียดดังนี้

2.3.1 งานวิจัยที่เกี่ยวกับการจัดสมดุลสายการประกอบแบบตัวยู

งานวิจัยในปัญหาการสมดุลสายการประกอบแบบตัวยู ได้เริ่มมีการวิจัยเป็นครั้งแรกเมื่อปี ค.ศ. 1994 โดย Miltenburg and Wijngaard [23] ซึ่งได้ทำวิจัยเรื่อง "The U-line line balancing problem" ที่เป็นการนำรูปแบบทางคณิตศาสตร์ของปัญหาโดยใช้กำหนดการพลวัต (Dynamic programming) สำหรับปัญหา U-line ที่ผลิตผลิตภัณฑ์ชนิดเดียวเพื่อลดจำนวนสถานี

งานให้น้อยที่สุด และได้พัฒนาวิธีการ RPWT (Ranked positional weight technique) ในการแก้ปัญหาของการสมดุลสายการประกอบแบบ U-line ในปัญหาขนาดใหญ่ที่มีงานถึง 111 งาน ซึ่งการจัดแบบ U-line จะมีความซับซ้อนมากกว่าปัญหาแบบเดิม เนื่องจากงานสามารถถูกจัดกลุ่มจากข้างหน้าไปข้างหลัง (Forward), และจากข้างหลังไปข้างหน้า (Backward) หรือทั้งสองทิศทางพร้อมๆ กันตามแผนผังลำดับงานก่อนหลัง และจากนั้นในปีถัดมา Miltenburg and Sparling [24] ได้พัฒนาวิธีการทางคณิตศาสตร์ (Exact methods) 3 วิธีการ ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู โดยวิธีกำหนดการพลวัต (Dynamic programming formulation; DP), วิธี Breadth-1 และ วิธี Depth-first branch and bound (B&B) ตามลำดับ ซึ่งผลการทดสอบแสดงให้เห็นว่าวิธี B&B มีประสิทธิภาพที่ดีกว่าวิธี DP-based algorithm และวิธี Breadth-first จะใช้เวลาคำนวณน้อยกว่าวิธี Depth-first แต่วิธี Depth-first จะเจอคำตอบเหมาะสมที่สุดเร็วกว่า ต่อมา Sparling and Miltenburg [25] ได้พัฒนาวิธีฮิวริสติกเพื่อแก้ปัญหาการจัดสมดุลแบบตัวยูที่ผลิตสินค้าแบบผสม (Mixed- model) ที่มีการรวมผังแสดงลำดับงาน (Precedence raph) ของแต่ละรุ่นผลิตภัณฑ์เข้าด้วยกันที่มีงาน 25 งาน และในปีเดียวกัน Urban [26] นำเสนอตัวแบบกำหนดการแบบเลขจำนวนเต็ม (Integer programming; IP) เพื่อหาค่าสมดุลที่เหมาะสุดสำหรับปัญหาการสมดุลสายการประกอบแบบ U-line ซึ่งตัวแบบนี้สามารถแก้ปัญหาขนาดใหญ่ได้ดีกว่าวิธีการเดิม ต่อจากนั้น Sparling and Miltenburg [25] ได้นำเสนอเกี่ยวกับรูปแบบทางคณิตศาสตร์ของปัญหา โดยใช้กำหนดการพลวัต (Dynamic programming) สำหรับปัญหาที่มีหลาย U-lines และมีจำนวนงานไม่เกิน 22 งาน โดยมีวัตถุประสงค์เพื่อจัดงานลงสถานีงานในลักษณะต่างๆ ให้มีจำนวนสถานีงานและลดเวลาสูญเปล่าให้น้อยสุดจากหลักการที่เสนอแนะก็เป็นวิธีที่มีประสิทธิภาพในการจัดสมดุลการประกอบ และการสมดุลสายการประกอบใหม่ (Re-balancing) ของปัญหาแบบตัวยูได้

ในปี ค.ศ. 1999 Chanahan [27] ได้ทำวิทยานิพนธ์เรื่อง "Incorporating ergonomics criteria into assembly line balancing ได้พัฒนาวิธีฮิวริสติก 3 วิธี ได้แก่ วิธี Multiple ranking heuristic, วิธี Combinatorial GA และ วิธี A Problem-space GA ตามลำดับ โดยนำเอาเกณฑ์การออกแบบ ทางการยศาสตร์ (Ergonomic) มาพิจารณาประกอบสำหรับแก้ปัญหา ALBP ทั้งแบบ I-Shaped และ U-Shaped line เพื่อลดรอบเวลาการผลิต และความสูญเสียของแรงที่ใช้ในการจับ (Grip strength) เนื่องจากความเมื่อยล้าในระหว่างสถานีงาน ให้มีค่าน้อยสุด ผลที่ได้จากงานวิจัยทำให้โรงงานอุตสาหกรรมได้รับประโยชน์ ทั้งในด้านการผลิตและทางการยศาสตร์ ต่อจากนั้น Scholl and Klein [28] ได้ทำวิจัยเรื่อง "ULINO-optimally balancing U-shaped JIT assembly line" โดยใช้วิธีการ Branch and Bound (B&B) ที่พัฒนามาจากวิธี SALOME ที่ใช้ในการแก้ปัญหาแบบเส้นตรง มาใช้สำหรับแก้ปัญหาแบบตัวยูที่มี 297 ขั้นงาน ซึ่งวิธีการใหม่ คือ ULINO (U-line optimizer) ที่มีพื้นฐานมาจากวิธี depth-first branch and bound และ dominance rules

เพื่อลดจำนวนสถานีงาน หรือ รอบเวลาผลิต หรือ ทั้งสองอย่างให้น้อยสุด และในปี ค.ศ. 2001 Erel et al. [29] ได้พัฒนาวิธีการ Simulated annealing (SA) มาใช้ในการแก้ปัญหา UALBP ซึ่งกำลังเป็นที่นิยมในปัจจุบัน เป้าหมายเพื่อลดจำนวนสถานีงานลงให้น้อยที่สุด (UALBP-1) โดยประสิทธิภาพของหลักการนี้จะถูกวัดจากการหาคำตอบในปัญหาที่มีขนาดใหญ่ และเปรียบเทียบผล กับ วิธี ULINO (U-line optimizer; เป็น B&B based heuristic procedure) ของ Scholl and Klein [28] นอกจากนี้ นักวิจัยยังได้เสนอแนะทิศทางงานวิจัยที่จะทำต่อไปคือ 1) สามารถใช้วิธีการ SA ในโจทย์ปัญหาที่ยากขึ้น เช่น Mixed/Multi model line, Stochastic task time และ U-line ในลักษณะอื่นๆ 2) นำเอาหลักการ Metaheuristic วิธีอื่นๆ เพื่อแก้ปัญหาทาง UALBP 3) พัฒนาวิธีการ Exact method เพื่อแก้ปัญหา U- line ที่เป็นปัญหาขนาดใหญ่ 4) ใช้หลักการ Metaheuristic วิธีอื่นๆ เพื่อแก้ปัญหาแบบ Type-2 (min. c, given m)

ปี ค.ศ. 2003 Chen [30] ได้พัฒนาวิธีการเพื่อหาผลลัพธ์ที่ดีที่สุดของการจัดสมดุลสายการประกอบแบบตัวยู ที่มีสถานีงานคู่ขนาน โดยใช้วิธี Multiple lower bounding และวิธีฮิวริสติกแบบใหม่ เพื่อหา Upper bounding ผลของวิธีการที่พัฒนาขึ้นทั้ง 2 วิธีแสดงให้เห็นถึงการปรับปรุงที่ดีขึ้นกว่าวิธีการเดิม จากนั้นในปีต่อมา Martinez and Duff [31] ได้ประยุกต์วิธีการเจเนติกอัลกอริทึม (GA) ที่ใช้ในปัญหาการสมดุลสายการประกอบแบบเส้นตรง (SALBP-1) มาใช้แก้ปัญหาการสมดุลงานแบบตัวยู ประเภทที่ 1 (UALBP-1) แล้วทำการเปรียบเทียบกับค่าคำตอบที่ดีที่สุดของงานวิจัยที่ผ่านมา ผลจากการวิจัยแสดงให้เห็นว่า วิธีการ GA สามารถหาผลลัพธ์ที่ดีที่สุด หรือคำตอบที่ใกล้เคียงได้ในรอบแรกๆของการทดลอง จากนั้น Gökçen et al. [32] นำเสนอวิธีเส้นทางที่สั้นที่สุด (Shortest route formulation) สำหรับหารจัดสมดุลสายการประกอบแบบตัวยู และในปีต่อมา Gökçen and Ağpak [33] ได้พัฒนารูปแบบของสมการเป้าหมายสำหรับปัญหาการจัดสมดุลงานแบบตัวยู ที่มีพื้นฐานมาจากรูปแบบกำหนดการเชิงจำนวนเต็มที่พัฒนาโดย Urban [26] สำหรับปัญหารูปแบบตัวยู และรูปแบบสมการเป้าหมายของ Deckro and Rangachari [34] ในปัญหารูปแบบเส้นตรง โดยรูปแบบที่นำเสนอจะเป็นหลักการที่มีความยืดหยุ่นที่ช่วยในการตัดสินใจในปัญหาการสมดุลงานแบบตัวยู กรณีที่มีหลายเงื่อนไข และในปีเดียวกัน Chiang and Urban [35] ที่ได้อธิบายถึงเหตุผลที่สายการผลิตแบบตัวยูเป็นที่นิยม เนื่องจากระบบการผลิตแบบ JIT ได้เข้ามามีบทบาท ดังนั้นสายการประกอบแบบ U-shaped ก็ได้มาแทนการจัดสายการประกอบแบบเดิม คือแบบเส้นตรง (Straight assembly lines) จุดที่คำนึงถึงในปัญหาของการจัดสมดุลสายการประกอบแบบ U-shaped คือ ความไม่แน่นอนของเวลางาน (Task time) เนื่องจากพนักงานหรือปัจจัยอื่นๆ ที่เวลาเป็นแบบไม่แน่นอนโดยมีการพัฒนาวิธีการฮิวริสติก 2 ส่วน คือ การหาผลลัพธ์เบื้องต้นและการปรับปรุงคำตอบ ผลจากการคำนวณแสดงให้เห็นถึงประสิทธิภาพของวิธีการที่นำมาใช้

ต่อมาได้มีนักวิจัยหลายท่านได้นำวิธีเมตาฮิวริสติกมาใช้ในการแก้ปัญหาแบบตัวยู เช่น Kriengkorakot [36] ได้นำเสนอวิธีระบบมดแม็กมิน (Max-Min ant system; MMAS) วิธี Max. task time และ Min. task time ร่วมกับวิธีโลคอลเสิร์ช ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงและตัวยู สำหรับการจัดสมดุลสายการ ประกอบแบบเส้นตรง ได้ทำการทดลองในชุดปัญหาตัวอย่างที่เป็นปัญหาขนาดใหญ่มี 45-111 งาน และชุดปัญหาตัวอย่างของ Lapierre [37] โดยวิธีทาบูเสิร์ช ซึ่งเป็นปัญหาขนาดใหญ่มี 297 งาน ส่วนในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ได้ทำการทดลองในชุดปัญหาตัวอย่าง ในการทดลองโดยวิธี Max. RPW กับปัญหาขนาดกลางที่มี 21-45 งาน และชุดปัญหาตัวอย่าง ข้อมูลบางส่วนที่เป็นปัญหาขนาดใหญ่มี 75-297 งาน จากการวิจัยสรุปผลได้ว่าวิธีระบบมดแม็กมิน ที่พัฒนาขึ้นเป็นวิธีที่มีประสิทธิภาพดีที่สุดเมื่อเปรียบเทียบกับวิธีฮิวริสติก และวิธี Max. RPW ของ Miltenburg และ Wijngaard [23] โดยจากการทดลองวิธี Min. task time เป็นวิธีที่ให้ค่าคำตอบที่แย่ที่สุด เมื่อเปรียบเทียบกับวิธีการอื่นๆและเมื่อมีการเปลี่ยนแปลงของค่ารอบเวลาการผลิตที่เพิ่มขึ้นผล ปรากฏว่า ไม่มีผลต่อความสามารถของการหาคำตอบของวิธีระบบมดแบบแม็กมิน ดังนั้นสรุปได้ว่า วิธีการที่พัฒนาขึ้นเป็นวิธีที่มีประสิทธิภาพมากวิธีหนึ่ง จากนั้นผู้วิจัยท่านอื่นได้ศึกษาการแก้ปัญหาสายการประกอบแบบตัวยูโดยวิธีอื่นๆ จากนั้น S. Seyed-Alagheband et al. [6] ได้นำเสนออัลกอริทึมทางพันธุกรรมแบบ (GA) เพื่อแก้ปัญหาสมดุลสายการผลิตแบบตัวยู (UALBP) โดยนำหลักการผลิตแบบ (JIT) มาใช้งานแก้ปัญหาร่วมด้วยในสายการผลิตแบบตัวยู ผลการทดลองแสดงพบว่าวิธีที่นำเสนอมีประสิทธิภาพรวมกลุ่มเวิร์กสเตชันได้ดีขึ้นและปรับปรุงรูปแบบของภาระงานดีขึ้น ในปีต่อมา Jirachai and Chutima [38] นำเสนออัลกอริทึมในการแก้ปัญหาการจัดสมดุลสายการประกอบลักษณะตัวยู ที่มีการผลิตผลิตภัณฑ์ผสมที่มีหลายวัตถุประสงค์ในระบบผลิตแบบทันเวลาพอดี ภายใต้ปัญหาประเภทที่ 1 โดยจะพิจารณาฟังก์ชันวัตถุประสงค์ทั้งหมด 4 วัตถุประสงค์พร้อมกันคือ จำนวนสถานีงานมีจำนวนน้อยที่สุด งานมีผลต่างความสัมพันธ์ใน สถานีงานมีค่าน้อยที่สุดและความผันแปรของเวลาในสถานีงานทั้งหมดมีค่าน้อยที่สุด ซึ่งจะทำการ เปรียบเทียบประสิทธิภาพของอัลกอริทึมที่นำเสนอกับอัลกอริทึมที่เป็นที่นิยมและได้คำตอบที่ดีใน ขณะนั้น คือ Non-dominated sorting genetic algorithm II (NSGA-II) ผลที่ได้จากการเปรียบเทียบ พบว่าอัลกอริทึมที่นำเสนอเป็นอัลกอริทึมที่มีประสิทธิภาพที่ดีกว่า อัลกอริทึม NSGA-II ในด้านคำตอบที่มีการลู่เข้าใกล้กลุ่มคำตอบที่ดีที่สุดที่แท้จริงได้คิดเป็น 45.145% ด้านอัตราส่วนของจำนวนกลุ่มคำตอบที่หาได้เทียบเท่ากับกลุ่มคำตอบที่แท้จริง 100 % และด้านเวลาในการประมวลผลที่เร็วกว่าอัลกอริทึม NSGA-II จากนั้นผู้วิจัยท่านอื่นได้ศึกษาการแก้ปัญหาสายการประกอบแบบตัวยูโดยวิธีอื่นๆ เช่น Sa-nguansin and Kunadilok [39] ได้วิจัยการจัดสมดุลสายงานการประกอบรูปทรงตัวยู ด้วยวิธีเจเนติกอัลกอริทึม (Genetic algorithm; GA) ร่วมกับระบบมดแม็ก-มิน (Max-Min ant system; MMAS) วิธีการที่นำเสนอนี้ GA ทำหน้าที่กำหนดจำนวนขั้นงานที่

เหมาะสม ให้กับแต่ละสถานีงาน ในขณะที่ MMAS ทำหน้าที่ปรับละเอียดลำดับขั้นงานที่ทำให้ค่าความแปรปรวนของภาระงานของสายงานการประกอบมีค่าน้อยที่สุด โดยสอดคล้องกับเงื่อนไขความสัมพันธ์ก่อน-หลังของขั้นงาน ค่าความแปรปรวนของภาระงาน ถูกกำหนดให้เป็นฟังก์ชันเป้าหมายของกระบวนการค้นหาด้วยวิธีเจเนติกอัลกอริทึม วิธีการที่นำเสนอนี้ถูกทดสอบกับปัญหา UALBP จำนวน 24 ปัญหาที่ได้จากการรวบรวมของ Scholl [37] จากผลการทดสอบพบว่า วิธีการที่นำเสนอนี้สามารถลดค่าความแปรปรวนของภาระงานในทุกปัญหา เมื่อเปรียบเทียบกับวิธี COMSOAL และวิธีระบบมดแม็ก-มิน ร่วมกับเทคนิคแบบสุ่ม มีค่าค่าความแปรปรวนของภาระงานมีค่าเฉลี่ยเท่ากับ 81.95% และ 73.16% ตามลำดับ ซึ่งจากการทบทวนงานวิจัยสายการประกอบแบบตัวยูที่ผ่านมาสามารถสรุปได้ดังในตาราง 3

ตาราง 3 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู

ผู้วิจัย	ปี	วิธีการแก้ปัญหา	รูปแบบปัญหา	เป้าหมาย	จำนวนงานในโจทย์ปัญหา
Miltenburg and Wijngaard	1994	DP formulation RPWT-based heuristic	Single model	*m*	Up to 111 tasks
Miltenburg and Wijngaard	1995	DP-based exact alg. Depth-first and breadth-first B&B	Single model	*m*	up to 40 tasks
Sparling and Miltenburg	1998	Heuristic	Mixed model	*m*	up to 25 tasks
Urban	1998	IP formulation	Single model	*m*	up to 45 tasks
Miltenburg	1998	DP-based exact alg.	U-line facility with several individual U-lines	*m* and idle time in a single st.	Individual U-line with up to 22 tasks
Scholl and Klein	1999	Branch and bound	Single model	*m, c* and both	Up to 297 tasks

ตาราง 3 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู (ต่อ)

ผู้วิจัย	ปี	วิธีการแก้ปัญหา	รูปแบบปัญหา	เป้าหมาย	จำนวนงานในโจทย์ปัญหา
Erel et al.	2001	Simulated annealing	Single model	*m*	Up to 297 tasks
Martinez and Duff	2004	Heuristic rules and Genetic Algorithm	Single model	*m*	Up to 21 tasks
Gokcen and Agpak	2005	Shortest route formulation (network programming)	Single model	*m*	Up to 7 tasks
Gokcen and Agpak	2006	IP formulation and GP	Single model	*m, c* and No. Tasks/st	Up to 30 tasks
Chaing and Urban	2006	Hybrid heuristic	Single model stochastic	*m*	Up to 111 tasks
Hwang et al.	2008	Genetic Algorithm	Single model	*m*	Up to 111 tasks
Kriengkorakot	2007	Ant colony	Single model	*m*	21-297 tasks
Parames et al.	2009	Combinatorial Optimization with Coincidence: COIN	Mixed model	*m*	Up to 111 tasks
Amonpong and Jakrawarn	2014	Genetic Algorithm with Max-Min Ant System	Single model	*m*	Up to 297 tasks

2.3.2 งานวิจัยที่เกี่ยวกับการจัดสมดุลสายการประกอบประเภทที่ 2

งานวิจัยเกี่ยวกับการจัดสมดุลสายการประกอบประเภทที่ 2 ตั้งแต่อดีตจนถึงปัจจุบัน มีผู้ทำวิจัยอยู่เป็นจำนวนมาก เริ่มตั้งแต่ Klein and Scholl [3] นำเสนอวิธี Branch and bound ที่มีการกำหนดขอบเขตของรอบเวลาการผลิตด้วยวิธี Local lower bound method ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรง ประเภทที่ 2 (SALBP-2) เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด ซึ่งผลการวิจัยพบว่า วิธีการที่นำเสนอให้ประสิทธิภาพมากในการจัดสมดุลสายการประกอบ ต่อมา Liu et al. [4] ได้นำเสนอ Two bi-directional heuristics ที่สร้างคำตอบเริ่มต้นโดยการจัดงานเข้าสถานีด้วยวิธีของ Hoffmann's heuristic แล้วทำการปรับปรุงคำตอบโดยการสลับสับเปลี่ยนงานด้วยวิธี TA&TT heuristic สำหรับการจัดสมดุลสายการประกอบแบบเส้นตรง ประเภทที่ 2 (SALBP-2) แล้วทำการทดสอบกับ 302 ตัวอย่าง และ 1440 ตัวอย่าง เพื่อวัดค่าเบี่ยงเบนโดยเฉลี่ยของรอบเวลาการผลิต (Average deviation from the optimal of cycle time: ADOC) และวัดค่า Workload smoothing ด้วยค่าเฉลี่ยของค่าสัมบูรณ์ของความคลาดเคลื่อน (Mean absolution deviation: MAD) ผลการวิจัยพบว่า Two bi-directional heuristics มีประสิทธิภาพทั้งในการลดรอบเวลาการผลิตและค่าเฉลี่ยของค่าสัมบูรณ์ของความคลาดเคลื่อน และในปี 2005 Chiang and Urban [35] นำเอา Two bi-directional heuristics ที่ได้นำเสนอก่อนหน้านี้ มาใช้ในการจัดสมดุลสายการประกอบแบบเส้นตรง ประเภทที่ 2 (SALBP-2) ที่มีเวลางานเป็นแบบสุ่ม (Stochastic) แล้วทำการทดสอบปัญหาที่มี 30 งาน, 45 งาน และ 70 งาน เพื่อเปรียบเทียบกับ Moodie and Young's algorithm ของ Moodie and Young [40] ผลการวิจัยพบว่า Bidirectional heuristics มีประสิทธิภาพในการลดรอบเวลาการผลิตในเวลาที่รวดเร็วถึง 70 งาน สำหรับปัญหาแบบ single-model stochastic assembly line balancing problem ต่อมา Seyed-Alagheband et al. [6] ได้ประยุกต์ใช้วิธีการจำลองการอบเหนียว (SA) สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบแบบทั่วไปด้วยเวลาติดตั้งระหว่างงานในแบบประเภทที่ 2 (General assembly line balancing problem with setups (GALBPS-2) เพื่อหารอบเวลาการผลิตที่น้อยที่สุด ซึ่งในกระบวนการของวิธี SA ค่าพารามิเตอร์ผู้วิจัยได้ Taguchi Method ออกแบบการทดลองเพื่อหาค่าที่เหมาะสม แล้วทดสอบกับชุดปัญหา 4 ชุดที่มีงานตั้งแต่ 11-111 งาน แล้ววัดค่า ค่าเฉลี่ยค่าเบี่ยงเบนสัมพัทธ์ (Average relative deviation: ARD) และ ค่าเบี่ยงเบนสัมพัทธ์ของค่าที่ดีที่สุดที่ได้จากวิธี SA (Relative deviation of the best obtained solutions: RDB) ผลการวิจัยพบว่าวิธี SA มีประสิทธิภาพในการหาคำตอบในด้านเวลาในการประมวลผล Attique et al. [41] นำเสนอการจัดสมดุลสายการประกอบประเภทที่ 2 ในกรณีที่ใช้ตัวดำเนินการหลายแบบที่ไม่เท่ากัน โดยใช้ Tomlab modeling platform ใน Matlab และใช้ข้อมูลของโรงงานผลิตรถยนต์ในประเทศปากีสถาน ที่มีจำนวน 6 สถานี, รอบเวลาการผลิต 410 นาทีและมีประสิทธิภาพสายการประกอบ 74.10% ในการ

ทดสอบ ซึ่งผลการวิจัยพบว่า สามารถลดรอบเวลาการผลิตลงเหลือ 251 นาที และเพิ่มประสิทธิภาพเป็น 94.62%

ในปี ค.ศ. 2014 Venkatesh and Balaji [42] ได้ประยุกต์ใช้วิธีเชิงพันธุกรรม (GA) ในการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) เพื่อหารอบเวลาการผลิตที่น้อยที่สุด โดยทดสอบกับชุดปัญหา 6 ชุดที่มีงานตั้งแต่ 11-70 งาน พบว่าวิธี GA ให้ผลเฉลยคำตอบของการจัดสมดุลสายการประกอบที่ดี ในปีเดียวกัน Choudhary and Agrawal [43] นำเสนอการจัดสมดุลสายการประกอบแบบเส้นตรง ประเภทที่ 2 (SALBP-2) สำหรับพนักงานที่เพิ่งได้รับการคัดเลือกเข้าสายการประกอบ ด้วยวิธี Branch and Bound และประมวลผลผ่านโปรแกรม Lingo 10.0 โดยมี 10 งานที่ทำงาน 4 คนและมี 4 สถานีงาน พบว่า สามารถจัดรอบเวลาการผลิตได้น้อยที่สุดที่ 39 วินาทีแต่ต้องมีการฝึกพนักงานที่ดีด้วย ในปีต่อมา Pakeeza et al. [44] นำเสนอวิธีฮิวริสติก (Heuristic) สำหรับการจัดสมดุลสายการประกอบแบบเส้นตรง ประเภทที่ 2 (SALBP-2) โดยทำการทดลองกับปัญหาของ Gunther จาก Scholl [19] ที่มีงาน 35 งานและ 6 สถานีงาน แล้วจึงนำไปใช้ในงานจริงกับ XYZ Company ที่มี 24 งาน และ 5 สถานีงาน ผลการการทดสอบ พบว่า สามารถจัดสมดุลสายการประกอบได้รอบเวลาการผลิตน้อยที่สุดที่ 17.45 min. ถัดมา Li et al. [9] จัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) ด้วย วิธี Rules-based heuristic ที่มีขั้นตอนการจัดสมดุลสายการประกอบ 3 ขั้นตอน คือ การเลือกงาน (Task selection rule), การมอบหมายงาน (Task assignment rule) และกฎการแลกเปลี่ยนงาน (Task exchange rule) ที่แบ่งเป็นวิธี Two Rules และวิธี Three Rules หลังจากนั้นนำไปทดสอบกับปัญหาที่มีงานตั้งแต่ 8-297 งาน ซึ่งเมื่อเปรียบเทียบผลกับการเขียนโปรแกรมจำนวนเต็ม (Integer programming) ที่ 63 ตัวอย่างที่มีงานตั้งแต่ 8-58 งาน พบว่าวิธี Three Rules ให้ผลที่ดีเทียบเท่ากับการเขียนโปรแกรมจำนวนเต็มมากว่ากว่า วิธี Two Rules และวิธีที่นำเสนอใช้เวลาในการประมวลผลที่เร็วกว่าการเขียนโปรแกรมจำนวนเต็ม และเมื่อเปรียบเทียบวิธี Two Rules กับ วิธี Three Rules ที่จำนวนงานตั้งแต่ 58-297 งาน พบว่า วิธี Three Rules ให้ผลที่ดีกว่า วิธี Two Rules และ Polat et al. [45] นำเสนอ Mathematical model สำหรับการจัดสมดุลสายการประกอบประเภทที่ 2 ภายใต้เงื่อนไขของการยศาสตร์ (Ergonomic) ภาระงาน (Workload) ของแต่ละสถานีงาน เพื่อให้แต่ละสถานีงานมีภาระงานไม่เกินตามหลักการยศาสตร์ ที่หาด้วยวิธีการประเมินความเสี่ยงทั่วทั้งร่างกาย (Rapid entire body assessment: REBA) เพื่อกำหนดภาระงาน (Workload) ที่เหมาะสม แล้วค่อยจัดสมดุลสายการประกอบ ซึ่งในงานวิจัยนี้ผู้วิจัยได้นำปัญหาของ Killbridge ที่มี 45 งาน จาก Scholl [46] มาสมมุติค่าภาระงาน (workload: W) ของแต่ละงาน และกำหนดให้มี 8 สถานีงาน ประมวลผลด้วย IBM ILOG CPLEX Optimization Studio 12.7.1 ผลการทดสอบพบว่าแต่ละสถานีงานมีภาระงานที่

สมดุลมากขึ้น ซึ่งจะช่วยลดความเสี่ยงของการเจ็บป่วยของพนักงานที่ทำงานในสถานีงานนั้นได้ โดยงานวิจัยที่เกี่ยวกับการจัดสมดุลสายการประกอบประเภทที่ 2 สามารถสรุปได้ดังตาราง 4

ตาราง 4 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบประเภทที่ 2

ผู้วิจัย	ปี	วิธีการแก้ปัญหา	รูปแบบปัญหา	เป้าหมาย	จำนวนงานในโจทย์ปัญหา
Klein and Scholl	1996	Branch and bound	SALBP-2	*c*	-
Liu et al.	2003	Bi-directional	SALBP-2	*c*	302 instance and 1440 instance
Liu et al.	2005	Bi-directional	SALBP-2	*c*	30 tasks, 45 tasks and 70 tasks
Seyed-Alagheband et al.	2011	Simulated annealing	SALBP-2	*c*	11-111 tasks
Attique et al.	2013	Heuristic	SALBP-2	*c*	107 tasks
Jonnalagedda and Balaji	2014	Genetic algorithm	UALBP-2	*c*	11-70 tasks
Choudhar and Agrawal	2014	Branch and Bound	SALBP-2	*c*	10 tasks
Pakeeza et al.	2015	Heuristic	SALBP-2	*c*	24 tasks
Li et al.	2017	Rules-based heuristic	UALBP-2	*c*	8-297 tasks
Polat et al.	2018	Mathematical model	SALBP-2	*c* and workload	45 tasks

2.3.3 งานวิจัยที่เกี่ยวกับการจัดสมดุลสายการประกอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง

งานวิจัยที่มีการจัดสมดุลสายการประกอบโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างกับปัญหาการจัดสมดุลสายการประกอบมีผู้วิจัยทั้งในและต่างประเทศเริ่มตั้งแต่ Nearchou [7] เป็นผู้ริเริ่มนำวิธีการวิวัฒนาการโดยใช้ผลต่าง (DE) มาใช้ในการแก้ปัญหาการจัดสมดุลสายการประกอบที่เป็นแบบเส้นตรง (SALBP) ที่มีวัตถุประสงค์เพื่อที่จะหาจำนวนสถานีงานน้อยที่สุด โดยที่เวลาของแต่ละสถานีจะต้องไม่เกินรอบเวลาการผลิต และเป็นไปตามเงื่อนไขลำดับงานก่อน ซึ่งวิธีการวิวัฒนาการโดยใช้ผลต่างมีกระบวนการในการหาคำตอบมี 4 ขั้นตอน คือ การสร้างประชากรเริ่มต้น (Initial population), การปรับเปลี่ยนค่าในพิกัด (Mutation), การแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (Crossover หรือ Recombination) และการคัดเลือก (Selection) โดยในการทดสอบวิธี DE ผู้วิจัยได้กำหนดค่าพารามิเตอร์ population size=10 individuals, crossover rate=0.4, mutation scale factor F=0.3, maximum number of generations=100×n (n=number of tasks) และใช้ตัวอย่างปัญหาที่มาจาก http://www.assembly-line-balancing.de ที่มีขั้นงาน 7-111 ขั้นงาน ในการทดสอบสอบ ผลการทดสอบพบว่าวิธีการวิวัฒนาการโดยใช้ผลต่างให้ประสิทธิภาพของคำตอบที่ดีเมื่อเปรียบเทียบกับคำตอบที่ได้จากตัวอย่างปัญหา Jirasirilerd and Pitakaso [47] การประยุกต์ใช้วิธีการวิวัฒนาการโดยใช้ผลต่างในการจัดสมดุลสายงานการประกอบแบบเส้นตรงประเภทที่ 1 โดยมีการพิจารณาประเภทเครื่องจักรเป็นเงื่อนไขประกอบในการจัดสมดุลสายงานการประกอบเย็บ ของอุตสาหกรรมผลิตเสื้อผ้าสำเร็จรูป เพื่อหาจำนวนสถานีงานที่น้อยที่สุด สำหรับปัญหาการจัดสมดุลสายงานการประกอบเย็บของเสื้อแขนสั้นรุ่น 518729 และทำการเปรียบเทียบผล ซึ่งวิธีการวิวัฒนาการโดยใช้ผลต่าง จะทำหน้าที่ในการสร้างเวคเตอร์เริ่มต้นด้วยการสุ่มจากจำนวนจริงและปรับปรุงคำตอบด้วยวิธีการปรับเปลี่ยนค่าในพิกัดด้วยวิธี DE/rand/1, DE/best/2 และ DE/rand/2 แล้วทำการแลกเปลี่ยนพิกัดด้วยวิธี Exponential crossover 1 Position จึงทำการคัดเลือกตามลำดับ จากผลการทดลอง พบว่าวิธีการวิวัฒนาการโดยใช้ผลต่างโดยการใช้ DE/rand/1 ในการปรับเปลี่ยนค่าในพิกัดสามารถลดจำนวนสถานีงานจากสายงานการประกอบเย็บ ปัจจุบันจาก 23 สถานี เป็น 17 สถานี และมีประสิทธิภาพของสายงานการประกอบเย็บเสื้อแขนสั้นรุ่น 518729 เพิ่มขึ้นเท่ากับ 56 เปอร์เซ็นต์ และใช้เวลาประมวลผลได้อย่างรวดเร็ว และในปีเดียวกัน Parawech [48] ได้ใช้วิธีการวิวัฒนาการโดยใช้ผลต่าง (DE) เพื่อเพิ่มประสิทธิภาพสายการประกอบ แบบเส้นตรงประเภทที่ 1 (SALBP-1) ที่มีวัตถุประสงค์เพื่อหาจำนวนสถานีงานน้อยที่สุด ซึ่งวิธีการ DE จะทำการสร้างเวกเตอร์เริ่มต้นโดยการสุ่มจำนวนจริง จากนั้นจะทำการปรับปรุงคำตอบโดยการผ่าเหล่า (Mutation), การข้ามสายพันธุ์ (Crossover) และทำการคัดเลือก (Selection) ซึ่งได้ประยุกต์ใช้วิธี Mutation 3 วิธีคือ DE/Rand, DE/Best/1 และ DE/Rand-to-Best/1 มาใช้ร่วมกับวิธี Crossover 2 วิธีคือ Binomial Crossover และ Exponential Crossover 2 Position ในการจัดสมดุลสายการ

ประกอบแบบเส้นตรงประเภทที่ 1 โดยใช้ตัวอย่างปัญหาของ Scholl [19] ที่มีงาน 7-111 งาน จำนวน 64 ปัญหา จากการทดลองพบว่าการผ่าเหล่า (Mutation) ด้วยวิธี DE/Best/1 ใช้ร่วมกับวิธี Exponential Crossover 2 Position ให้คำตอบที่ที่สุดในเวลาที่รวดเร็วที่สุด จำนวน 59 ปัญหา ซึ่งถือได้ว่าเป็นวิธีการปรับปรุงคำตอบที่มีประสิทธิภาพมากวิธีหนึ่งเมื่อเปรียบเทียบกับการใช้วิธี DE/Rand ร่วมกับ Binomial Crossover, DE/Best/1 ร่วมกับ Binomial Crossover, DE/Rand-to-Best/1 ร่วมกับ Binomial Crossover , DE/Rand ร่วมกับ Exponential Crossover 2 Position และ DE/Rand-to-Best/1 ร่วมกับ Exponential Crossover 2 Position ที่สามารถให้คำตอบที่ดีที่สุดในเวลาที่รวดเร็วที่สุด จำนวน 27, 34 , 31, 45 และ 54 ปัญหาตามลำดับ

ถัดมา Parawech and Paitakaso [49] นำวิธีการวิวัฒนาการโดยใช้ผลต่าง (DE) ไปจัดเป็นปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงประเภทที่ 1 เพื่อเพิ่มประสิทธิภาพสายการประกอบเย็บในโรงงานผลิตเสื้อผ้าสำเร็จรูปแห่งหนึ่งในจังหวัดอุบลราชธานี ที่มีผลิตภัณฑ์ชนิดเดียว มีจำนวนสถานีงาน 19 สถานี ประสิทธิภาพสายการประกอบเท่ากับ 65.39 % เพื่อหาจำนวนสถานีงานน้อยที่สุด โดยมีเงื่อนไขความสัมพันธ์ก่อนหลัง มีการพิจารณาให้แต่ละสถานีงานมีเวลารวมไม่เกินรอบเวลาการผลิตและสามารถมีเครื่องจักรได้ไม่เกิน 2 ประเภท ซึ่งวิธีการวิวัฒนาการโดยใช้ผลต่าง จะทำการสร้างโครโมโซมเริ่มต้นโดยการสุ่มจานวนจริงแล้วปรับปรุงคำตอบโดยการกลายพันธ์และทำการผสมสายพันธุ์ จากนั้นทำการเลือกผลที่ดีกว่าเพื่อเป็นประชากรในรุ่นถัดไป ซึ่งได้เปรียบเทียบกับวิธีฮิวริสติกพื้นฐาน ได้แก่ วิธี Maximum task time, Kilbridge and Weter's method (KWM) และวิธี Ranked positional weights method (RPW) ที่สามารถลดสถานีงานเหลือ 14, 15 และ 16 สถานี เพิ่มประสิทธิภาพเป็น 88.75 %, 82.83 % และ 77.66 % ตามลำดับ ผลการทดลองพบว่าวิธีการวิวัฒนาการโดยใช้ผลต่าง ให้คาตอบที่ดีกว่าสายการผลิตปัจจุบันและวิธีฮิวริสติกทั้ง 3 วิธี โดยวิธีการวิวัฒนาการโดยใช้ผลต่าง สามารถลดสถานีงานเหลือ 13 สถานี และเพิ่มประสิทธิภาพสายการประกอบเป็น 95.58 % ปีต่อมา Pitakaso [50] ใช้วิธีวิวัฒนาการของโดยใช้ผลต่าง (DE) ในการแก้ปัญหาสมดุลสายการประกอบแบบที่ 1 (SALBP-1) ปัญหาที่นำเสนอประกอบด้วย 4 ขั้นตอนหลัก ได้แก่ (1) การเริ่มต้น (2) การผ่าเหล่า (Mutation), (3) การข้ามสายพันธุ์ (Crossover) (4) กระบวนการคัดเลือก โดยงานวิจัยนี้ได้มีการคิดการผสมผสานกันระหว่างกระบวนการ การผ่าเหล่า (Mutation) ที่ใช้สมการวิธี DE/rand/1, DE/best/1, DE/rand-to-best/1, DE/best/2 และ DE/rand/2 ตามลำดับ และ การข้ามสายพันธุ์ (Crossover) ที่ใช้วิธี Binomial Crossover, วิธี Exponential Crossover 1 Position และวิธี Exponential Crossover 2 Position จึงทำให้ได้วิธีการใหม่เกิดขึ้น ซึ่งผลในการให้การคำนวณจากการทดสอบเปรียบเทียบกับวิธี Genetic Algorithm (GA) [51], วิธี Tabu Search [52], [37], วิธี Differential Evolution (DE) [7] และวิธี Firing sequences backward algorithm [53] แสดงให้เห็นว่าอัลกอริธึม DE ที่ใช้

สามารถแก้ปัญหา SALPB-1 ได้ในระดับดีและปะมวลผลได้อย่างรวดเร็ว และในปีเดียวกัน Pitakaso and Sethanan [16] ได้ใช้วิธีวิวัฒนาการของโดยใช้ผลต่าง (DE) ในการแก้ปัญหาสมดุลสายการประกอบแบบที่ 1 (SALBP-1) ที่มีเงื่อนไขของจำนวนประเภทเครื่องจักรที่อยู่ในสถานีงานต้องไม่เหมือนกัน (SALBP-1M) แล้วทำการประมวลผลแล้วเปรียบเทียบกับวิธี Genetic algorithm [51], [54], วิธี Tabu search algorithms [37], [52] ผลการวิจัยพบว่า วิธีวิวัฒนาการของโดยใช้ผลต่าง (DE) เป็นให้คำตอบที่ดีเมื่อเปรียบเทียบกับวิธีอื่นๆ ต่อมา Nearchou and Omirou [17] นำเสนอการจัดสมดุลสายการประกอบแบบเส้นตรง ประเภทที่ 1 (SALBP-1) เพื่อหาจำนวนสถานีงานที่น้อยที่สุด และประเภทที่ 2 (SALBP-2) ที่หารอบเวลาการผลิตที่น้อยที่สุด ด้วยวิธีวิวัฒนาการของโดยใช้ผลต่าง (DE) แล้วใช้ตัวอย่างปัญหาของ Talbot [55] และ Scholl [19] ที่มีงานตั้งแต่ 7-111 งานในการทดสอบและเปรียบเทียบผล ซึ่งวิธี DE มีประสิทธิภาพสามารถหาคำตอบได้ดีเท่ากับชุดปัญหาและ Sresracoo et al. [18] ได้ใช้วิธีการวิวัฒนาการโดยใช้ผลต่าง (DE) ในการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 (UALBP-1) กับการแก้ไขปัญหาขนาดกลางที่มีจำนวนงาน 11-45 งาน โดยกำหนดค่าพารามิเตอร์ F=0.8, NP=30 และ CR=0.8 และประมวลผลเทียบกับวิธี Integer programming solution [26], วิธี Ant Colony [56] และ วิธี Max-Min Colony [36] ผลการทดสอบพบว่า วิธีการวิวัฒนาการโดยใช้ผลต่างสามารหาคำตอบได้กว่าและใช้เวลาในการค้นหาได้อย่างรวดเร็ว โดยงานวิจัยที่เกี่ยวกับการจัดสมดุลสายการประกอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่างสามารถสรุปได้ดังตาราง 5

ตาราง 5 สรุปงานวิจัยที่เกี่ยวข้องกับการจัดสมดุลสายการประกอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง

ผู้วิจัย	ปี	รูปแบบปัญหา	เป้าหมาย	จำนวนงานในโจทย์ปัญหา
Nearchou	2005	SALBP-1	*m*	7-111 tasks
Jirasirilerd and Pitakaso	2013	SALBP-1	*m*	36 tasks in Garment
Parawech and Paitakaso	2013	SALBP-1	*m*	7-111 tasks
Parawech and Paitakaso	2014	SALBP-1	*m*	52 tasks in Garment
Pitakaso	2015	SALBP-1	*m*	-
Rapeepan and Kanchana	2015	SALBP-1M	*m*	-
Nearchou and Omirou	2017	SALBP-1 SALBP-2	*m, c*	7-111 tasks
Sresracoo et al.	2018	UALBP-1	*m*	11-45 tasks

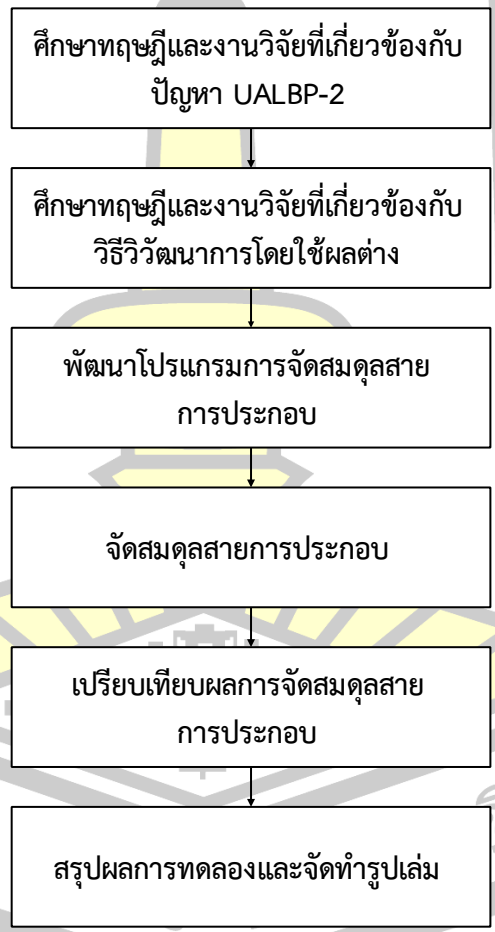
จากที่ผู้วิจัยได้ทำการสรุปจากการทบทวนวรรณกรรมและงานวิจัยที่เกี่ยวข้องกับสมดุลสายการประกอบ พบว่า ในการจัดสมดุลสายการประกอบผู้วิจัยแต่ละท่านนำเสนอวิธีการแก้ปัญหาที่หลากหลายวิธีที่แตกต่างกัน ไม่ว่าจะเป็น วิธีอาณานิคมมด (Ant Colony), วิธีเชิงพันธุกรรม (Genetic Algorithm: GA), วิธีการค้นหาคำตอบแบบทาบู (Tabu search) และ วิธีวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) ซึ่งแต่ละวิธีก็มีกระบวนการหาคำตอบที่แตกต่างกันด้วย ซึ่งในปัญหาหารจัดสมดุลสายการประกอบวิธีเชิงพันธุกรรม (GA) เป็นวิธีที่มีผู้วิจัยได้นิยมนำมาใช้ในการแก้ปัญหาจำนวนมาก ซึ่งถือได้ว่าเป็นวิธีที่มีประสิทธิภาพมากวิธีหนึ่ง และเมื่อศึกษางานวิจัยที่เกี่ยวกับวิธีเชิงพันธุกรรม ก็พบว่ามีวิธีที่สามารถให้คำตอบได้เทียบเท่ากับวิธีเชิงพันธุกรรมและสามารถหาคำตอบที่เป็นไปได้และใกล้เคียงคำตอบที่ดีที่สุด ในเวลาที่สั้น คือ วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) เป็นวิธีที่สามารถที่จะค้นหาคำตอบที่ดีที่สุดสำหรับปัญหาขนาดใหญ่ที่มีความซับซ้อนของปัญหา ภายในพื้นที่คำตอบที่เป็นไปได้โดยใช้ระยะเวลาที่สั้นในการหาคำตอบเมื่อเปรียบกับวิธีเมตาฮิวริสติกวิธีอื่นๆ จากงานวิจัยของ Dervis and Selcuk [10] ได้อธิบายว่า วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithms :DEA) เป็นหนึ่งในวิธีการ (Evolutionary Algorithms :EA) ที่มีขั้นตอนวิธีการวิวัฒนาการที่เป็นเทคนิคใหม่ในการเพิ่มประสิทธิภาพ ความสามารถในการจัดการปัญหาประเภท Non-Differentiable, Non-Linear และ Multimodal Objective Functions โดยเฉพาะในปัญหาขนาดใหญ่ที่ซับซ้อน ก็พบว่าความเร็วในหาคำตอบของวิธีการวิวัฒนาการโดยใช้ผลต่างมีประสิทธิภาพที่ดีกว่า ดังนั้นในงานวิจัยนี้ผู้วิจัยจึงได้เลือกวิธีวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) สำหรับการแก้ปัญหาสายการประกอบแบบตัวยู ประเภทที 2 (UALBP-2) เพื่อหารอบการผลิตที่น้อยที่สุด

พหุน ปณ ฑิโต ชีเว

บทที่ 3
วิธีการดำเนินการวิจัย

3.1 ขั้นตอนการดำเนินงานวิจัย

ในการวางแผนการดำเนินงานการวิจัยในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิต (Cycle time) ที่ต่ำที่สุด ภายใต้เงื่อนไขข้อจำกัดของจำนวนสถานีงาน และเงื่อนไขของลำดับงานก่อนหลัง (Precedence diagram) สามารถกำหนดขั้นตอนในการดำเนินงานได้ดังแสดงในภาพประกอบ 11



ภาพประกอบ 11 ลำดับขั้นตอนดำเนินงานวิจัย

3.1.1 ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับปัญหา UALBP-2

ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP–2) กรณีทราบจำนวนสถานีงาน และหาจำนวนรอบเวลาที่ต่ำที่สุด โดยทำการทบทวนเอกสารงานวิจัยที่เกี่ยวข้องทั้งในประเทศและต่างประเทศ เพื่อที่จะกำหนดรูปแบบของปัญหาที่จะศึกษา วัตถุประสงค์ และขอบเขตการวิจัย

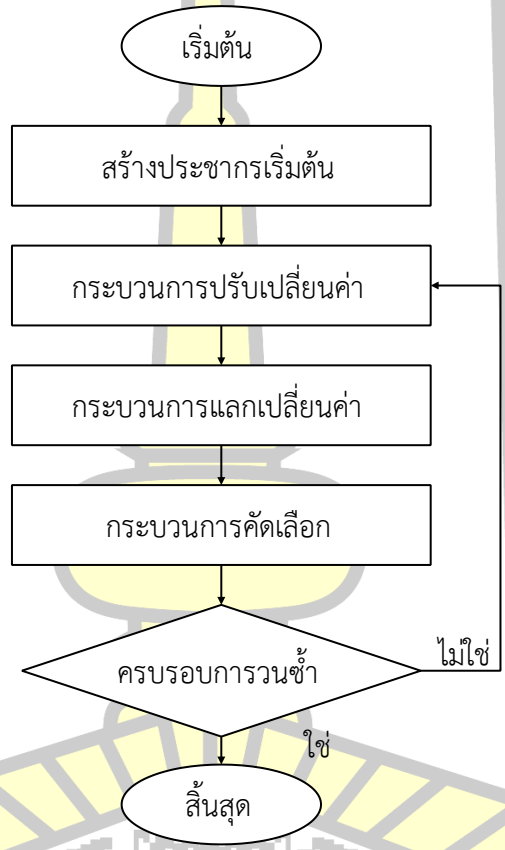
3.1.2 ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับวิธีวิวัฒนาการโดยใช้ผลต่าง

จากการที่ผู้วิจัยได้ทำการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับวิธีวิวัฒนาการโดยใช้ผลต่างที่ใช้ในการจัดสมดุลสายการประกอบ ที่มีผู้ที่ทำการวิจัยมาตั้งแต่อดีตจนถึงปัจจุบัน โดยทำการทบทวนเอกสารงานวิจัยที่เกี่ยวข้องทั้งในประเทศและต่างประเทศ พบว่าวิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential evolution algorithm: DE) เป็นวิธีที่สามารถค้นหาคำตอบที่ดีที่สุด สำหรับปัญหาขนาดใหญ่ที่มีความซับซ้อนของปัญหา ภายในพื้นที่คำตอบที่เป็นไปได้ โดยใช้ระยะเวลาที่สั้นในการหาคำตอบเมื่อเปรียบกับวิธีเมตาฮิวริสติกวิธีอื่นๆ โดยขั้นตอนวิธีการวิวัฒนาการโดยใช้ผลต่างเป็นเทคนิคใหม่ในการเพิ่มประสิทธิภาพและความสามารถในการจัดการสำหรับปัญหาขนาดใหญ่ที่ซับซ้อนในการหาคำตอบและใช้เวลาไม่นาน ดังนั้นวิธีการวิวัฒนาการโดยใช้ผลต่าง จะเป็นวิธีการที่นำมาใช้แก้ปัญหาและช่วยเพิ่มประสิทธิภาพทางในการหาคำตอบสำหรับปัญหาการจัดสมดุลสายการประกอบได้และอีกทั้งวิธีการนี้ยังไม่เคยมีนักวิจัยนำมาใช้ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) จึงน่าจะเป็นวิธีที่ใหม่ที่ช่วยในการค้นหาคำตอบสำหรับการแก้ปัญหาสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) ดังนั้นผู้วิจัยจึงได้เลือกวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) สำหรับการแก้ปัญหาสายการประกอบแบบตัวยู (UALBP-2) ในงานวิจัยครั้งนี้

3.1.2.1 วิธีวิวัฒนาการโดยใช้ผลต่าง (Differential evolution: DE)

วิธีวิวัฒนาการโดยใช้ผลต่างเป็นวิธีการหาคำตอบที่มีโครงสร้างที่ไม่ซับซ้อน สามารถหาคำตอบได้อย่างรวดเร็ว มีประสิทธิภาพ โดยในการพัฒนาคำตอบของวิธีวิวัฒนาการโดยใช้ผลต่างนั้น จะใช้วิธีการปรับเปลี่ยนค่าในพิกัด (Mutation) ซึ่งสามารถดำเนินการได้ด้วยการหาผลต่างของเวกเตอร์ที่สุ่มมา 2 เวกเตอร์ของประชากรและนำผลต่างนี้ไปคูณกับปัจจัยขยายผลต่าง (Scaling factor; *F*) แล้วจึงนำไปบวกกับเวกเตอร์ที่สุ่มมาอีกเวกเตอร์หนึ่ง โดยประชากรรุ่นที่กำลังจะถูกปรับเปลี่ยนค่าในพิกัดเรียกว่า เวกเตอร์เป้าหมาย (Target vector) และประชากรที่ถูกปรับเปลี่ยนแล้วจะเรียกว่า มิวแทนต์เวกเตอร์ เมื่อประชากรถูกปรับเปลี่ยนจนครบจำนวนประชากรที่ตั้งไว้ล่วงหน้า (Number of population; *NP*) จากนั้นจะใช้วิธีการแลกเปลี่ยนค่าในพิกัด (Crossover หรือ Recombination) ระหว่างเวกเตอร์เป้าหมายและมิวแทนต์เวกเตอร์โดยใช้ความน่าจะเป็น

แลกเปลี่ยนค่าในพิกัดซึ่งเป็นตัวแปรที่ทราบค่าที่ควบคุมได้เรียกว่าอัตราการปรับเปลี่ยน (Crossover Rate; *CR*) เวกเตอร์ที่ผ่านการแลกเปลี่ยนค่าในพิกัดแล้วจะถูกเรียนว่าไทรอัลเวกเตอร์ (Trial vector) จากนั้นจะนำไทรอัลเวกเตอร์ที่ได้ไปหาค่าฟิตเนสฟังก์ชันหรือฟังก์ชันเป้าหมาย แล้วเปรียบเทียบค่าฟิตเนสฟังก์ชันของไทรอัลเวกเตอร์กับเวกเตอร์เป้าหมาย เวกเตอร์ใดดีกว่าจะถูกใช้เป็นเวกเตอร์เป้าหมายในรอบถัดไป โดยได้แสดงขั้นตอนวิธีแสดงอัลกอริทึมของวิธีวิวัฒนาการโดยใช้ผลต่างและลำดับขั้นตอนของวิธีวิวัฒนาการโดยใช้ผลต่างในภาพประกอบ 12



ภาพประกอบ 12 ลำดับขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง

1) การสร้างประชากรเริ่มต้น (Initial population)

การสร้างประชากรของคำตอบเริ่มต้นสามารถทำได้โดยการสุ่มค่าประชากรหรือเวกเตอร์ตั้งต้น (Initial population) โดยกำหนดขอบเขตของตัวแปรการตัดสินใจเพื่อให้แน่ใจว่าขอบเขตของตัวแปรการตัดสินใจจะคลอบคลุมถึงคำตอบที่ดีที่สุด และกำหนดให้โอกาสที่จะถูกเลือก

ของคำตอบที่มีค่าสม่ำเสมอ (Uniform probability distribution) ที่มีมิติเท่ากับ *D* ซึ่งอาจหมายถึงพิกัดหรือตำแหน่ง และจำนวนคำตอบที่เป็นไปได้เริ่มต้นเท่ากับ *NP* โดยแต่ละคำตอบหรือแต่ละเวกเตอร์เป้าหมายเรียกว่า เวกเตอร์การตัดสินใจ (Decision vector) ซึ่งจะนำมาคำนวณเพื่อหาค่าคำตอบของสมการเป้าหมายของเวกเตอร์เป้าหมาย (Target Vector) หรือค่าฟิตเนส (Fitness value)

2) การปรับเปลี่ยนค่าในพิกัด (Mutation)

การปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากร เรียกว่า Mutation factor โดยมีจุดประสงค์หลักเพื่อแลกเปลี่ยนพิกัดภายในเวกเตอร์ ซึ่งในการปรับเปลี่ยนค่าพิกัดนั้นสามารถทำได้โดยทำการสุ่มเวกเตอร์มา 3 เวกเตอร์ (ตัวแทนผลลัพธ์) $X_{r1,G}$, $X_{r2,G}$ และ $X_{r3,G}$ มาดำเนินการปรับเปลี่ยนพิกัด โดยเวกเตอร์ที่ถูกปรับเปลี่ยนค่าพิกัดไปแล้วจะถูกเรียกว่า มิวแทนเวกเตอร์ (Mutant vector) โดยที่ค่ามิวแทนเวกเตอร์ $(V_{i,G+1})$ จะมีค่าเท่ากับ $X_{r1,G}$ รวมกับผลต่างของเวกเตอร์ $X_{r2,G}$ และ $X_{r3,G}$ คูณด้วย *F* โดยสามารถอธิบายการปรับค่าในพิกัดของเวกเตอร์ได้ดังสมการ 3.1

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \qquad (3.1)$$

เมื่อ $V_{i,G+1}$ = มิวแทนเวกเตอร์ (Mutant vector)

$X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$ = เวกเตอร์สุ่ม (Random vector) คือเวกเตอร์ที่ทำการสุ่มเลือกมา 3 เวกเตอร์

F = แฟกเตอร์ในการขยายผลต่าง (Scaling factor) คือจำนวนจริงที่มีค่าคงที่และมีค่าระหว่าง 0 ถึง 2

3) การแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (Crossover หรือ Recombination)

การแลกเปลี่ยนค่าในพิกัดเวกเตอร์เป็นการเพิ่มความหลากหลายในการหาคำตอบของปัญหาโดยขั้นตอนนี้จะเป็นการสร้างไทรอัลเวกเตอร์ (Trial vector) โดยใช้สมการ 3.2 ซึ่งจะใช้ในการสุ่มตัวเลขที่อยู่ระหว่าง 0-1 โดยมีขนาดตำแหน่งเท่ากับ *D* และจำนวนประชากรเท่ากับ *NP* หากตัวเลขสุ่มของทุกเวกเตอร์และทุกตำแหน่ง มีค่าน้อยกว่าหรือเท่ากับค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (*CR*) ค่าในตำแหน่งพิกัดของเวกเตอร์นั้นจะมีค่าเท่ากับค่าของมิวแทนต์เวกเตอร์ในตำแหน่งและพิกัดเดียวกัน และในกรณีที่เป็นอย่างอื่น (มีค่ามากกว่าค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (*CR*)) ค่าในตำแหน่งและพิกัดนั้นๆจะมีค่าเท่ากับเวกเตอร์เป้าหมาย (Target vector)

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{if } (rand\ (j)\ [0,1) \le CR) \\ X_{ji,G+1} & \text{if } (rand\ (j) > CR) \end{cases} \quad (3.2)$$

เมื่อ $U_{ji,G+1}$ = ไทรอัลเวกเตอร์ (Trial vector)

$V_{ji,G+1}$ = มิวแทนเวกเตอร์ (Mutant vector)

$X_{ji,G+1}$ = เวกเตอร์เป้าหมาย (Target vector)

$rand\ (j)$ = การสุ่มเลขจำนวนจริงที่มีค่าตั้งแต่ 0-1 ให้กับทุกตำแหน่ง j; j = 1,2,...,D

CR = อัตราแลกเปลี่ยนตำแหน่งพิกัดที่มีค่าเป็นเลขจำนวนจริงระหว่าง 0-1

4) การคัดเลือก (Selection)

การคัดเลือกเป็นขั้นตอนของการเลือกเวกเตอร์เป้าหมายในรอบถัดไปหรือการเลือกประชากรสำหรับการดำเนินในรุ่นต่อไป โดยเวกเตอร์ที่ให้คำตอบที่ดีกว่าจะถูกเลือกโดยใช้วิธีการเปรียบเทียบค่าคำตอบของสมการเป้าหมาย (Objective value) ที่ได้จากเวกเตอร์เป้าหมาย (Target vector) กับค่าคำตอบที่ได้จากไทรอัลเวกเตอร์ (Trial vector) ซึ่งถ้าหากค่าคำตอบที่ได้จาก ไทรอัลเวกเตอร์มีค่าน้อยกว่าหรือเท่ากับค่าคำตอบที่ได้จากเวกเตอร์เป้าหมาย จะเลือกไทรอัลเวกเตอร์เป็นประชากรในรุ่นถัดไป และในกรณีที่เป็นอย่างอื่น (ค่าคำตอบที่ได้จากเวกเตอร์เป้าหมายมีค่ามากกว่าค่าคำตอบที่ได้จากไทรอัลเวกเตอร์) จะทำการเลือกเวกเตอร์เป้าหมายเป็นประชากรรุ่นถัดไป ดังสมการ 3.3

$$X_{ji,G+1} = \begin{cases} U_{ji,G+1} & \text{if } f\ (U_{ji,G+1}) \le f\ (X_{ji,G+1}) \\ X_{ji,G} & \text{otherwise} \end{cases} \quad (3.3)$$

เมื่อ $X_{ji,G+1}$ = เวกเตอร์เป้าหมาย (Target vector)

$U_{ji,G+1}$ = ไทรอัลเวกเตอร์ (Trial vector)

3.1.2.2 การปรับปรุงวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2

จากขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง ที่แสดงดังหัวข้อ 3.1.2.1 ซึ่งเป็นกระบวนการหาคำตอบโดยทั่วไปของวิธีวิวัฒนาการโดยใช้ผลต่าง แต่เนื่องจากวิธีวิวัฒนาการโดยใช้ผลต่างนั้นได้รับความนิยมเมื่อไม่กี่ปีที่ผ่านมา ดังนั้นการปรับปรุงวิธีวิวัฒนาการโดย

ใช้ผลต่างจึงยังมีไม่มากนักเมื่อเปรียบเทียบกับวิธีอื่นๆ โดยในการปรับปรุงประสิทธิภาพในการหาคำตอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่างนั้น ผู้วิจัยจะเน้นการปรับเปลี่ยนวิธีการพัฒนาคำตอบของเวกเตอร์ทั้งในส่วนของการปรับเปลี่ยนพิกัด (Mutation) และการแลกเปลี่ยนพิกัดเวกเตอร์ (Recombination) เพื่อใช้ในการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 สามารถอธิบายได้ดังนี้

1) การปรับปรุงการปรับเปลี่ยนพิกัด (Mutation)

การปรับเปลี่ยนพิกัดโดยปกติแล้วด้วยวิธีวิวัฒนาการโดยใช้ผลต่างจะใช้สมการ 3.4 แต่ Qin et al. [57] ได้พัฒนาสมการในการปรับเปลี่ยน 5 สมการเพื่อเพิ่มประสิทธิภาพและสร้างทางเลือกในการหาคำตอบมากขึ้น คือ สมการ DE/rand/1, DE/best/1, DE/rand-to-best/1, DE/best/2 และ DE/rand/2 ตามลำดับ สามารถแสดงได้ดังนี้

DE/rand/1 $$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \quad (3.4)$$

DE/best/1 $$V_{i,G+1} = X_{best,G} + F(X_{r1,G} - X_{r2,G}) \quad (3.5)$$

DE/rand-to-best/1 $$V_{i,G+1} = X_{i,G} + F(X_{best,G} - X_{i,G}) + F(X_{r1,G} - X_{r2,G}) \quad (3.6)$$

DE/best/2 $$V_{i,G+1} = X_{best,G} + F(X_{r1,G} - X_{r2,G}) + F(X_{r3,G} - X_{r4,G}) \quad (3.7)$$

DE/rand/2 $$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) + F(X_{r4,G} - X_{r5,G}) \quad (3.8)$$

2) การปรับปรุงการแลกเปลี่ยนพิกัด (Recombination)

การแลกเปลี่ยนพิกัดโดยปกติแล้วด้วยวิธีวิวัฒนาการโดยใช้ผลต่างจะใช้สมการ 3.9 ซึ่งเป็นการแลกเปลี่ยนค่าในพิกัดแบบไบโนเมียล (Binomial crossover) ซึ่งต่อมา Pitakaso [50] ได้พัฒนาสมการในการแลกเปลี่ยนค่าในพิกัดแบบเอกซ์โพเนนเชียลแบบ 1 จุด (Exponential crossover 1 position) และเอกซ์โพเนนเชียลแบบ 2 จุด (Exponential crossover 2 position) ขึ้นมา ดังแสดงในสมการ 3.10 และ 3.11 โดยสมการแลกเปลี่ยนพิกัดสามารถแสดงได้ดังนี้

(1) การแลกเปลี่ยนค่าในพิกัดแบบไบโนเมียล (Binomial Crossover)

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{if } (rand\ (j)\ [0,1) \le CR) \\ X_{ji,G+1} & \text{if } (rand\ (j) > CR) \end{cases} \quad (3.9)$$

(2) การแลกเปลี่ยนค่าในพิกัดแบบเอกซ์โพเนนเชียลแบบ 1 จุด (Exponential Crossover 1 Position)

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{เมื่อ n มีค่าตั้งแต่ 1 ถึง R และ R ตำแหน่งแรกที่พบว่า (rand (j)<}CR\text{)} \\ X_{ji,G} & \text{ในกรณีอื่นๆ} \end{cases} \quad (3.10)$$

(3) การแลกเปลี่ยนค่าในพิกัดแบบเอกซ์โพเนนเชียลแบบ 2 จุด (Exponential Crossover 2 Position)

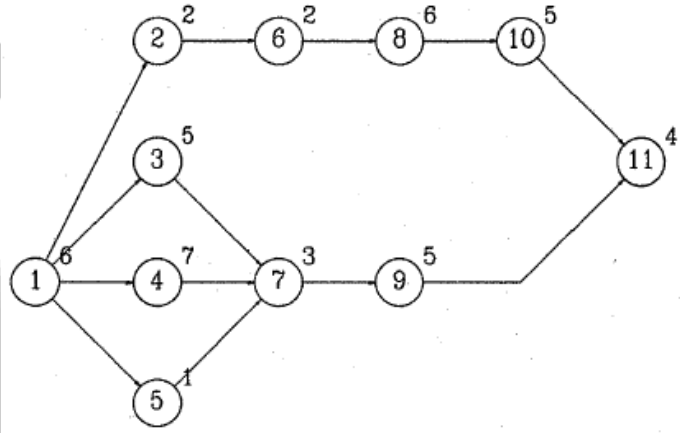
$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{เมื่อ n มีค่าตั้งแต่ 1 ถึง } R_1 \text{ และ } R_1 \text{ ตำแหน่งแรกที่พบว่า (rand(j)>}CR\text{) และตำแหน่ง} \\ & R_2 \text{ ถึง N เมื่อ } R_1 \text{ คือตำแหน่งที่พบว่า (rand(j)>}CR\text{) ที่อยู่ถัดจาก } R_1 \\ X_{ji,G+1} & \text{ในกรณีอื่นๆ} \end{cases} \quad (3.11)$$

จากการทบทวนงานวิจัยในอดีตพบว่า Sresracoo et al. [18] ได้นำวิธีการวิวัฒนาการโดยใช้ผลต่างไปแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 เพื่อหาจำนวนสถานีงานที่ต่ำที่สุด โดยทำการหาคำตอบด้วยการปรับปรุงการเปลี่ยนพิกัด (Mutation) ด้วย 5 สมการร่วมกับการปรับปรุงการแลกเปลี่ยนพิกัด (Recombination) 3 วิธีแล้วพบว่าการใช้สมการ DE/best/2 ในการปรับปรุงการปรับเปลี่ยนพิกัดร่วมกับการแลกเปลี่ยนพิกัดแบบเอกซ์โพเนนเชียลแบบ 1 จุด (Exponential crossover 1 position) โดยกำหนดพารามิเตอร์ของ R=30 รอบ, NP=30, F=0.8 และ CR=0.8 สามารถหาคำตอบที่ดีที่สุดได้มีประสิทธิภาพมากสุด ดังนั้นในการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 ของงานวิจัยนี้จึงได้กำหนดพารามิเตอร์ R=30 รอบ, NP=30, F=0.8 และ CR=0.8 และทำการปรับเปลี่ยนพิกัดด้วยสมการ DE/best/2 ร่วมกับการแลกเปลี่ยนค่า (Recombination) แบบเอกซ์โพเนนเชียลแบบ 1 จุด (Exponential crossover 1 Position) สามารถแสดงตัวอย่างการปรับปรุงวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 ได้ดังนี้

3.1.2.3 ตัวอย่างการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 2 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่าง

จากกระบวนการวิธีวิวัฒนาการโดยใช้ผลต่าง เริ่มต้นจากการสร้างประชากรเริ่มต้น (Initial population) และนำเวคเตอร์ที่ออกแบบไปผ่านกระบวนการปรับเปลี่ยนค่าในพิกัดของเวกเตอร์ (Mutation) แล้วนำเวกเตอร์ที่ได้เข้าสู่กระบวนการแลกเปลี่ยนค่าในพิกัดของเวคเตอร์

(Recombination) แล้วทำการคัดเลือกเวคเตอร์ (Selection) ที่ได้เป็นประชากรเริ่มต้นในรอบถัดไป ผู้วิจัยจะแสดงตัวอย่างการคำนวณโดยใช้ปัญหาของ Jackson [58] ซึ่งสามารถอธิบายลำดับขั้นตอนงานก่อนหลังดังภาพประกอบ 13



ภาพประกอบ 13 แสดงแผนผังลำดับขั้นตอนงานของปัญหา Jackson [58]

จากชุดปัญหาของ Jackson ดังภาพประกอบ 13 ได้แสดงแผนภาพลำดับงานก่อนหลังของปัญหาการจัดสมดุลสายการประกอบอย่างง่ายที่มี 11 งานตัวเลขในวงกลมแสดงชื่อของแต่ละงานและตัวเลขบนวงกลมแสดงเวลาของงานนั้นๆ ซึ่งก่อนที่จะทำการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) ต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยมีตัวแปรดังนี้ m= จำนวนสถานีงาน, R=Round (จำนวนรอบการทดลอง), NP=จำนวนคำตอบที่เป็นไปได้เริ่มต้น, F= แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR=อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณปัญหานี้กำหนดให้ m=4 สถานีงาน, R=1 รอบ, NP=3, F=0.8, CR=0.8 และ Scholl and Klein [28] LC = max $\{[t_{sum}/m], t_{max}\}$ จะได้ LC = max $\{12,7\}$ ดังนั้นจะได้ Cycle time=12 โดยได้แสดงวิธีการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างของการปรับเปลี่ยนพิกัด แบบพื้นฐาน “DE/best/2” ร่วมกับการแลกเปลี่ยนพิกัดแบบ Exponential Crossover 1 Position ได้ดังนี้

1) การสร้างคำตอบเริ่มต้น (Initial population) การสร้างคำตอบเริ่มต้น (Initial population) เริ่มจากการสุ่มจำนวนจริง ตั้งแต่ 0-1 ให้กับงานทั้ง 11 งาน ซึ่งเป็นการสร้างค่า Target vector หรือคำตอบเริ่มต้นเพื่อเป็นค่าตัดสินใจในการจัดงาน (task) ลงสถานีงาน (station) และเป็นค่าเริ่มต้นของเวกเตอร์ที่ใช้ในการ Mutation และ Crossover ในขั้นตอนถัดไป แสดงดังตาราง 6 โดยในการจัดงานลงสถานีจะทำการจัดงานที่มีค่า Target vector มากที่สุดลงสถานีงานก่อนโดยไม่

ขัดแย้งกับเงื่อนไขของลำดับงานก่อนหลัง (Precedence diagram) และจำนวนสถานีงาน (m) ผลการการสร้างคำตอบเริ่มต้นแสดงได้ดังตาราง 7

ตาราง 6 ค่า Target Vector ของคำตอบเริ่มต้น

Task	1	2	3	4	5	6	7	8	9	10	11
Vector 1	0.58	0.43	0.71	0.25	0.78	0.77	0.85	0.16	0.25	0.44	0.88
Vector 2	0.25	0.54	0.44	0.80	0.42	0.70	0.60	0.12	0.75	0.32	0.44
Vector 3	0.77	0.60	0.66	0.55	0.80	0.58	0.42	0.10	0.23	0.23	0.50
Vector 4	0.59	0.44	0.72	0.26	0.79	0.78	0.86	0.17	0.26	0.45	0.89
Vector 5	0.24	0.53	0.43	0.79	0.41	0.69	0.59	0.11	0.74	0.31	0.43

ตาราง 7 ผลการสร้างคำตอบเริ่มต้นของปัญหาเท่ากับจำนวน *NP*

Target Vector	Dimension	1	2	3	4	5	6	7	8	9	10	11
Vector 1	Station		1			2			3		4	
	Task	11	1	5	3	10	2	6	4	7	9	8
	Target Vector	0.88	0.58	0.78	0.71	0.44	0.43	0.77	0.25	0.85	0.25	0.16
Vector 2	Station		1		2			3			4	
	Task	11	9	7	4	3	5	10	1	2	6	8
	Target Vector	0.44	0.75	0.60	0.80	0.44	0.42	0.32	0.25	0.54	0.70	0.12
Vector 3	Station		1			2			3		4	
	Task	1	5	3	2	6	4	11	7	10	9	8
	Target Vector	0.77	0.80	0.66	0.60	0.58	0.55	0.50	0.42	0.23	0.23	0.10
Vector 4	Station		1			2			3		4	
	Task	11	1	5	3	10	2	6	4	7	9	8
	Target Vector	0.89	0.59	0.79	0.72	0.45	0.44	0.78	0.26	0.86	0.26	0.17
Vector 5	Station		1		2			3			4	
	Task	11	9	7	4	3	5	10	1	2	6	8
	Target Vector	0.43	0.74	0.59	0.79	0.43	0.41	0.31	0.24	0.53	0.69	0.11

2) การการปรับเปลี่ยนพิกัด (Mutation) เป็นการปรับเปลี่ยนค่าในพิกัดเวกเตอร์ (Dimension: *D*) เพื่อทำให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากประชากรเดิม โดยเวกเตอร์ที่ถูกปรับเปลี่ยนค่าพิกัดด้วยสมการ “DE/Best/2” แล้วจะถูกเรียกว่า มิวแทนเวกเตอร์ (Mutant vector)

$$V_{i,G+1} = X_{best,G} + F(X_{r1,G} - X_{r2,G}) + F(X_{r3,G} - X_{r4,G}) \quad (3.12)$$

เมื่อ $V_{i,G+1}$ = มิวแทนเวกเตอร์ (Mutant vector)

$X_{best,G}$ = เวกเตอร์ที่ดีที่สุดหรือเวกเตอร์ที่จัดงานลงสถานีงานและได้รอบเวลาการผลิตที่ต่ำที่สุดภายใต้เงื่อนไขที่กำหนด

$X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$, $X_{r4,G}$ = เวกเตอร์สุ่ม (Random vector) จากประชากร G

F = แฟกเตอร์ในการขยายผลต่าง (Scaling factor) คือ จำนวนจริงที่มีค่าคงที่และมีค่าระหว่าง 0-2

จากตาราง 7 เมื่อกำหนดให้ Target vector 1 คือเวกเตอร์ที่ดีที่สุด ($X_{best,G}$) และกำหนดให้ค่า F=0.8 เมื่อทำการปรับเปลี่ยนพิกัด (Mutation) เวกเตอร์เป้าหมาย 1 (Target vector 1) จะเริ่มจากการสุ่มเวกเตอร์จำนวน 4 เวกเตอร์ (r_1,r_2,r_3,r_4) ของแต่ละเวกเตอร์เป้าหมายจากประชากรตั้งต้น (NP) ได้เวกเตอร์ 3, 2, 5 และ 4 จากนั้นทำการคำนวณหา Mutant vector ($V_{i,G+1}$) โดยนำเวกเตอร์สุ่มที่ได้ไปแทนสมการที่ 3.12 จะได้ค่ามิวแทนเวกเตอร์ทุกพิกัดดังนี้ $V_{i,G+1}$ = 0.88 + 0.8x(0.77–0.44) + 0.8x(0.43–0.89) = 0.78

Dimension 1 $V_{i,G+1}$= 0.88 + 0.8x(0.77–0.44) + 0.8x(0.43–0.89) = 0.78

Dimension 2 $V_{i,G+1}$= 0.58 + 0.8x(0.80–0.75) + 0.8x(0.74–0.59) = 0.74

Dimension 3 $V_{i,G+1}$= 0.78 + 0.8x(0.66–0.60) + 0.8x(0.59–0.79) = 0.67

Dimension 4 $V_{i,G+1}$= 0.71 + 0.8x(0.60–0.80) + 0.8x(0.79–0.72) = 0.61

Dimension 5 $V_{i,G+1}$= 0.44 + 0.8x(0.58–0.44) + 0.8x(0.43–0.45) = 0.54

Dimension 6 $V_{i,G+1}$= 0.43 + 0.8x(0.55–0.42) + 0.8x(0.41–0.44) = 0.51

Dimension 7 $V_{i,G+1}$= 0.77 + 0.8x(0.50–0.32) + 0.8x(0.31–0.78) = 0.54

Dimension 8 $V_{i,G+1}$= 0.25 + 0.8x(0.42–0.25) + 0.8x(0.24–0.26) = 0.37

Dimension 9 $V_{i,G+1}$= 0.85 + 0.8x(0.23–0.54) + 0.8x(0.53–0.86) = 0.34

Dimension 10 $V_{i,G+1}$= 0.25 + 0.8x(0.23–0.70) + 0.8x(0.69–0.26) = 0.22

Dimension 11 $V_{i,G+1}$= 0.16 + 0.8x(0.10–0.12) + 0.8x(0.11–0.17) = 0.10

เมื่อทำการคำนวณจนครบทุกพิกัดแสดงค่ามิวแทนเวกเตอร์ดังตาราง 8

ตาราง 8 ผลการมิวแทนเวกเตอร์ของเวกเตอร์เป้าหมาย 1

	Dimension	1	2	3	4	5	6	7	8	9	10	11
Vector 1	Task	11	1	5	3	10	2	6	4	7	9	8
	Mutant Vector	0.78	0.74	0.67	0.61	0.54	0.51	0.54	0.37	0.34	0.22	0.10

3) การแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ (Crossover หรือ Recombination)

เป็นการเพิ่มความหลากหลายในการหาคำตอบของปัญหาด้วยการสร้างไทรอัลเวกเตอร์ (Trial vector) ด้วยวิธี Exponential crossover 1 position ซึ่งจะเริ่มจากการสุ่มตัวเลขที่อยู่ระหว่าง 0-1 ให้กับทุกพิกัด (D) และหากตัวเลขสุ่มของพิกัดแรกที่พบว่ามีค่าน้อยกว่าหรือเท่ากับค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (CR) ค่าในตำแหน่งตั้งแต่พิกัดแรกจนถึงพิกัดที่พบจะมีค่าเท่ากับค่าของมิวแทนต์เวกเตอร์ และในกรณีที่เป็นอย่างอื่น ค่าในตำแหน่งและพิกัดนั้นๆจะมีค่าเท่ากับเวกเตอร์เป้าหมาย (Target vector)

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{เมื่อ } n \text{ มีค่าตั้งแต่ 1 ถึง } R \text{ และ } R \text{ ตำแหน่งแรกที่พบว่า } (rand\ (j)<CR) \\ X_{ji,G} & \text{ในกรณีอื่นๆ} \end{cases} \quad (3.13)$$

เมื่อ $V_{ji,G+1}$ = มิวแทนเวกเตอร์ (Mutant vector)

$X_{ji,G+1}$ = เวกเตอร์เป้าหมาย (Target vector)

CR = อัตราการปรับเปลี่ยน (Crossover rate) จำนวนจริงระหว่าง 0-1

rand(j) = การสุ่มตัวเลขจำนวนจริงที่มีค่าระหว่าง 0-1 ครั้งที่ j; j = 1,2,3,...,D

จากผลของการมิวแทนเวกเตอร์ของเวกเตอร์เป้าหมาย 1 ตาราง 8 การแลกเปลี่ยนค่าในพิกัดเวกเตอร์ (Recombination) เมื่อกำหนดให้อัตราแลกเปลี่ยนตำแหน่งพิกัด CR=0.8 แล้วทำการสุ่มตัวเลขระหว่าง 0-1 ให้ทุกพิกัดได้ 0.81, 0.88, 0.87, 0.92, 0.81, 0.88, 0.95, 0.99, 069, 0.46, 0.77 และพบว่าพิกัดที่ 9 มีค่าตัวเลขสุ่มเท่ากับ 0.69 คือพิกัดแรกที่พบว่ามีค่าน้อยกว่าค่า CR ในการไทรอัลเวกเตอร์ ตั้งแต่พิกัดที่ 1-9 จะใช้ค่ามิวแทนเวกเตอร์ ส่วนในพิกัดที่ 10-11 จะใช้ค่าเวกเตอร์เป้าหมาย แสดงดังตาราง 9 และเมื่อได้ค่าไทรอัลเวกเตอร์ แล้วจะนำค่าที่ได้ไปจัดงานลงสู่สถานีงานโดยไม่ขัดแย้งกับเงื่อนไขของลำดับงานก่อนหลัง และจำนวนสถานีงาน (m) ได้ดังตาราง 10

ตาราง 9 การไทรอัลเวกเตอร์ของเวกเตอร์เป้าหมาย 1

	Dimension	1	2	3	4	5	6	7	8	9	10	11
Vector 1	Task	11	1	5	3	10	2	6	4	7	9	8
	Target Vector	0.88	0.58	0.78	0.71	0.44	0.43	0.77	0.25	0.85	0.25	0.16
	Mutant Vector	0.78	0.74	0.67	0.61	0.54	0.51	0.54	0.37	0.34	0.22	0.10
	rand(j)	0.81	0.88	0.87	0.92	0.81	0.88	0.95	0.99	0.69	0.46	0.77
	Trial Vector	0.78	0.74	0.67	0.61	0.54	0.51	0.54	0.37	0.34	0.25	0.16

ตาราง 10 การจัดงานเข้าสู่สถานีงานของเวกเตอร์เป้าหมาย 1

	Station	1			2			3			4	
Vector 1	Task	11	1	5	3	10	2	6	4	7	9	8
	Trial Vector	0.78	0.74	0.67	0.61	0.54	0.51	0.54	0.37	0.34	0.25	0.16

4) การคัดเลือก (Selection) เป็นขั้นตอนของการเลือกเวกเตอร์เป้าหมายในรอบถัดไปหรือการเลือกประชากรสำหรับการดำเนินในรุ่นต่อไป โดยเวกเตอร์ที่ให้คำตอบที่ดีกว่าจะถูกเลือกโดยใช้วิธีการเปรียบเทียบค่าคำตอบของสมการเป้าหมาย (Objective value) ที่ได้จากเวกเตอร์เป้าหมาย (Target vector) กับค่าคำตอบที่ได้จากไทรอัลเวกเตอร์ (Trial vector) ซึ่งถ้าหากค่าคำตอบที่ได้จาก ไทรอัลเวกเตอร์มีค่าน้อยกว่าหรือเท่ากับค่าคำตอบที่ได้จากเวกเตอร์เป้าหมาย จะเลือกไทรอัลเวกเตอร์เป็นประชากรในรุ่นถัดไป และในกรณีที่เป็นอย่างอื่น (ค่าคำตอบที่ได้จากเวกเตอร์เป้าหมายมีค่ามากกว่าค่าคำตอบที่ได้จากไทรอัลเวกเตอร์) จะทำการเลือกเวกเตอร์เป้าหมายเป็นประชากรรุ่นถัดไป ดังสมการ 3.14

$$X_{i,G+1} = \begin{cases} U_{i,G+1} \text{ if } (U_{i,G+1}) < f(X_{i,G+1}) \\ X_{i,G} \text{ Otherwise} \end{cases} \quad (3.14)$$

เมื่อ $U_{i,G+1}$ = ไทรอัลเวกเตอร์ (Trial vector)

$X_{i,G+1}$ = เวกเตอร์เป้าหมาย (Target vector) ในรุ่นถัดไป, i = 1,2,...n

3.1.3 พัฒนาโปรแกรมการจัดสมดุลสายการประกอบ

หลังจากที่ได้ศึกษาวิธีวิวัฒนาการโดยใช้ผลต่างที่ทำให้ทราบขั้นตอนการหาคำตอบของวิธีวิวัฒนาการโดยใช้ผลต่าง ที่มีขั้นตอนทั้งหมด 4 ขั้นตอน ได้แก่ สร้างประชากรเริ่มต้น (Initial population), การปรับเปลี่ยนค่าพิกัด (Mutation), การแลกเปลี่ยนพิกัด (Crossover หรือ Recombination) และการคัดเลือก (Selection) ภายใต้เงื่อนไขของจำนวนสถานีงาน (m) ผู้วิจัยจะขั้นตอนของวิธีวิวัฒนาการโดยใช้ผลต่างไปพัฒนาออกมาเป็นโปรแกรมโดยใช้ภาษา Java ในการเขียนโปรแกรม และเมื่อได้โปรแกรม ผู้วิจัยจะนำไปทดสอบโดยการจัดสมดุลสายการประกอบในลำดับต่อไป

3.1.4 จัดสมดุลสายการประกอบ

จากการศึกษาวิธีวิวัฒนาการโดยใช้ผลต่างและพัฒนาโปรแกรมการจัดสมดุลสายการประกอบ สำหรับการการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP–2) กรณีทราบจำนวนสถานีงาน (m) และหารอบการผลิต (c) ที่น้อยที่สุด แล้วนั้น ผู้วิจัยก็จะนำโปรแกรมที่ได้จากหัวข้อ 3.1.3 ไปทดสอบการจัดสมดุลสายการประกอบโดยใช้ชุดปัญหาของ Scholl [19] จาก https://assembly-line-balancing.de/ ที่มีจำนวนงานตั้งแต่ 21-297 งาน จำนวน 15 ชุดปัญหา หรือสามารถจำแนกออกได้เป็น 101 ปัญหา แล้วนำผลการจัดสมดุลสายการประกอบที่ได้มาเปรียบเทียบกับงานวิจัยท่านอื่นซึ่งจะอธิบายในขั้นตอนต่อไป

ตาราง 11 ชุดปัญหาในการทดสอบการจัดสมดุลสายการประกอบ

ชื่อชุดปัญหา	จำนวนงาน (n)	เวลางานต่ำสุด (t_{min})	เวลางานสูงสุด (t_{max})	เวลารวมของทุกงาน (t_{sum})
Mitchell	21	1	13	105
Rosenberg	25	1	13	125
Heskiaoff	28	1	108	1024
Buxey	29	1	25	324
Sawyer	30	1	25	324
Lutz 1	32	100	1400	14140
Gunther	35	1	40	483
Kilbridge&Wester	45	3	55	552
Hahn	53	40	1775	14026

ตาราง 11 ชุดปัญหาในการทดสอบการจัดสมดุลสายการประกอบ (ต่อ)

ชื่อชุดปัญหา	จำนวนงาน (n)	เวลางานต่ำสุด (t_{min})	เวลางานสูงสุด (t_{max})	เวลารวมของทุกงาน (t_{sum})
Warnecke	58	7	53	1548
Tonge	70	1	156	3510
Lutz 2	89	1	10	485
Mukherjee	94	8	158	4208
Bartholdi	148	3	383	5642
Scholl	297	5	1310	69655

3.1.5 เปรียบเทียบผลการจัดสมดุลสายการประกอบ

เมื่อได้ผลการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 จากวิธีวิวัฒนาการโดยใช้ผลต่าง จากโปรแกรมที่พัฒนาขึ้นมาแล้วนั้นผู้วิจัยจะนำผลการจัดสมดุลสายการประกอบที่ได้มาเปรียบเทียบผลกับงานวิจัยของ Li et al. [9] ที่ทำการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อรอบเวลาการผลิต (c) ที่ต่ำที่สุดด้วยวิธี Rules–based Heuristic เพื่อวัดประสิทธิภาพของวิธีการหาคำตอบและเวลาในการประมวลผล ของการจัดสมดุลสายการประกอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่างว่ามีความแตกต่างกับวิธี Rules–based Heuristic มากน้อยเพียงใด

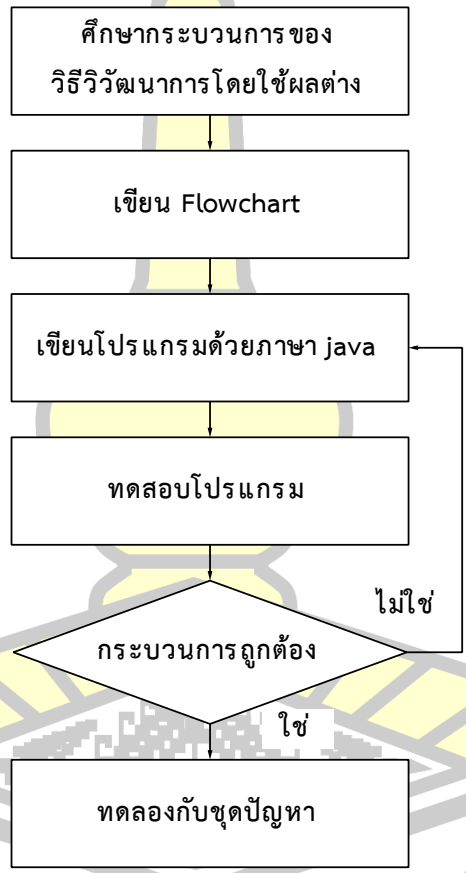
3.1.6 สรุปผลการทดลองและจัดทำรูปเล่ม

หลังจากที่ได้ทำการปรับปรุงวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 แล้วพัฒนาโปรแกรมเพื่อใช้ในการจัดสมดุลสายการประกอบโดยใช้ชุดปัญหาของ Scholl [19] 15 ชุดปัญหาเพื่อทำการเปรียบเทียบผลกับวิธี Rules–based Heuristic [9] ตลอดจนทำการเปรียบเทียบผลของรอบเวลาการผลิตและเวลาที่ใช้ในการประมวลผลที่กล่าวมาข้างต้นแล้ว ผู้วิจัยก็จะทำการวิเคราะห์และสรุปผลการวิจัยของวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2

บทที่ 4
ผลการวิจัย

4.1 การพัฒนาโปรแกรมการจัดสมดุลสายการประกอบ

สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด ด้วยการประยุกต์ใช้วิธีวิวัฒนาการโดยใช้ผลต่าง (Differential evolution algorithm: DE) สามารถอธิบายการพัฒนาโปรแกรม DE2 แสดงดังภาพประกอบ 14



ภาพประกอบ 14 กระบวนการพัฒนาโปรแกรม DE2

จากภาพประกอบ 14 แสดงการพัฒนาโปรแกรม DE2 โดยเริ่มจากการศึกษาวิธีวิวัฒนาการโดยใช้ผลต่างจากแหล่งข้อมูลต่างๆ แล้วกำหนดขั้นตอนการทำงานของจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 ออกมาเป็น Flowchart ตลอดจนได้กำหนดพารามิเตอร์สำหรับการทดสอบโดยกำหนด

R=30 รอบ, NP=30, F=0.8 และ CR=0.8 และทำการปรับเปลี่ยนพิกัดด้วยสมการ DE/best/2 ร่วมกับการแลกเปลี่ยนค่า (Recombination) แบบเอกซ์โพเนนเชียลแบบ 1 จุด (Exponential crossover 1 position) Sresracoo et al. [18] จากนั้นผู้วิจัยได้ทำการเขียนโปรแกรมด้วยภาษา Java บนโปรแกรม NetBeans IDE 8.2 ผ่านเครื่องคอมพิวเตอร์ Intel(R) Core(TM) i5-8500 CPU@3.0 GHz, 4GB RAM แล้วทำการทดสอบโปรแกรมด้วยการใช้ชุดปัญหาของ Jackson [58] เพื่อดูความถูกต้องของกระบวนการหาคำตอบ หลังจากที่ได้โปรแกรมที่มีกระบวนการหาค้นคำตอบตามวิธีวิวัฒนาการโดยใช้ผลต่าง ผู้วิจัยจะนำโปรแกรม DE2 ไปทำการทดสอบกับชุดปัญหาของ Scholl ต่อไป

4.2 ชุดปัญหาของการทดสอบการจัดสมดุลสายการประกอบ

การทดสอบวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP–2) เพื่อหาค่ารอบเวลาการผลิตที่ต่ำที่สุด กรณีทราบจำนวนสถานีงาน ผู้วิจัยได้นำชุดปัญหาของ Scholl [19] ที่มีจำนวนงานตั้งแต่ 21-297 งาน จำนวน 15 ชุดปัญหา (101 ปัญหา) จาก www.assembly-line-balancing.de ดังภาคผนวก ก โดยสามารถแบ่งออกเป็นปัญหาขนาดกลางที่มีจำนวนงานตั้งแต่ 21-58 งาน จำนวน 10 ชุดปัญหา (50 ปัญหา) และปัญหาขนาดใหญ่ที่มีจำนวนงานตั้งแต่ 70-297 งาน จำนวน 5 ชุดปัญหา (51 ปัญหา) แสดงข้อมูลดังตาราง 12

ตาราง 12 ตารางแสดงข้อมูลของชุดปัญหาที่ใช้ในการทดสอบ

ชื่อชุดปัญหา	จำนวนงาน (n)	เวลางานต่ำสุด (t_{min})	เวลางานสูงสุด (t_{max})	เวลารวมของทุกงาน (t_{sum})
Mitchell	21	1	13	105
Rosenberg	25	1	13	125
Heskiaoff	28	1	108	1024
Buxey	29	1	25	324
Sawyer	30	1	25	324
Lutz 1	32	100	1400	14140
Gunther	35	1	40	483
Kilbridge&Wester	45	3	55	552
Hahn	53	40	1775	14026
Warnecke	58	7	53	1548

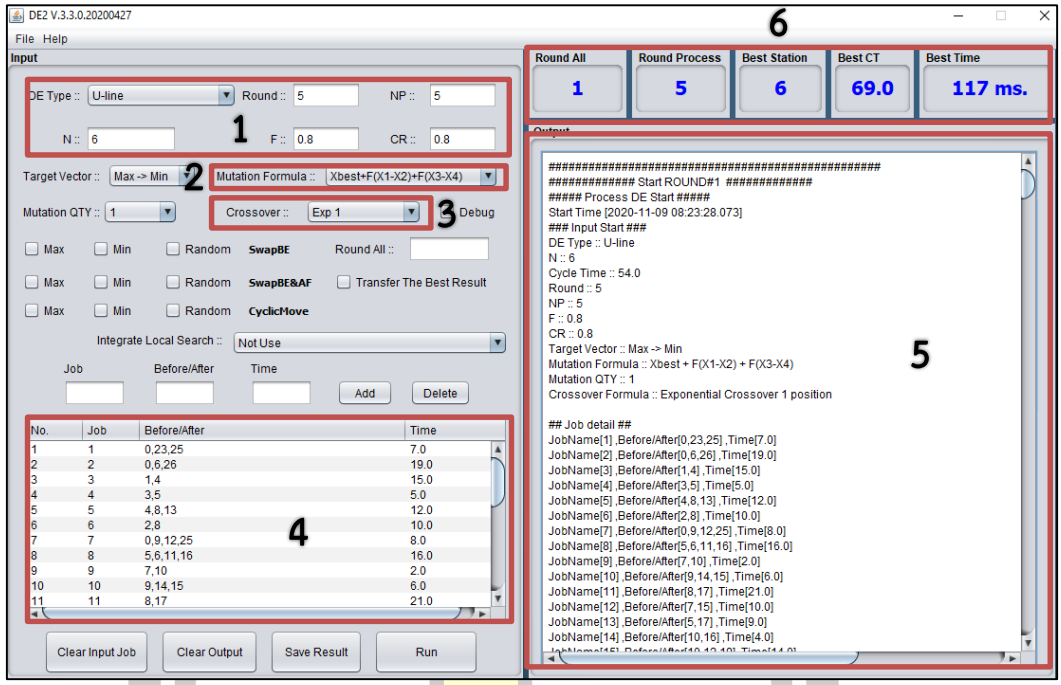
ตาราง 12 ตารางแสดงข้อมูลของชุดปัญหาที่ใช้ในการทดสอบ (ต่อ)

ชื่อชุดปัญหา	จำนวนงาน (n)	เวลางานต่ำสุด (t_{min})	เวลางานสูงสุด (t_{max})	เวลารวมของทุกงาน (t_{sum})
Tonge	70	1	156	3510
Lutz 2	89	1	10	485
Mukherjee	94	8	158	4208
Bartholdi	148	3	383	5642
Scholl	297	5	1310	69655

จากตาราง 12 แสดงถึงขนาดของปัญหาทั้ง 15 ปัญหาที่มีจำนวนงาน (n) ตั้งแต่ 21-297 งาน มีเวลางานต่ำสุด (t_{min}) เวลางานสูงสุด (t_{max}) และเวลารวมของทุกงาน (t_{sum}) ที่แตกต่างกัน ซึ่งจะเป็นข้อมูลที่ใช้ในการทดสอบประสิทธิภาพของวิธีวิธีวิวัฒนาการโดยใช้ผลต่างต่อไป

4.3 การจัดสมดุลสายการประกอบ

หลังจากที่ได้พัฒนาโปรแกรมที่ใช้ในการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP–2) และกำหนดชุดปัญหาที่ใช้ในการทดสอบวิธีวิวัฒนาการโดยใช้ผลต่าง ตามที่ได้กล่าวมาในหัวข้อ 4.1 และ 4.2 แล้วนั้น ในการจัดสมดุลสายการประกอบเพื่อหารอบเวลาของการผลิตที่ต่ำที่สุด ผู้วิจัยได้ทำการทดสอบโปรแกรม DE2 บนเครื่องคอมพิวเตอร์ Intel(R) Core(TM) i5-8500 CPU@3.0 GHz, 4GB RAM ดังภาพประกอบ 15



ภาพประกอบ 15 แสดงการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2

จากภาพประกอบ 15 สามารถอธิบายส่วนประกอบของโปรแกรมได้ดังนี้

1. ส่วนของอินพุต (Input) ข้อมูลพารามิเตอร์ จำนวนรอบ (round), จำนวนประชากรที่ตั้งไว้ล่วงหน้า (NP), จำนวนสถานีงาน (N), ปัจจัยขยายผลต่าง (F) และอัตราการปรับเปลี่ยน (CR)

2. สมการของการปรับเปลี่ยนค่าในพิกัด (mutation)

3. สมการของการแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (crossover หรือ recombination)

4. ส่วนของข้อมูลชุดปัญหาที่ใช้ในการทดสอบที่ประกอบไปด้วยงาน (task), ลำดับงานก่อน-หลัง (precedence diagram) และเวลาของงาน (time)

5. ส่วนของเอาต์พุต (Output) ที่แสดงผลลัพธ์ที่ได้จากการประมวลผล

6. ส่วนของการแสดงผลของการหาคำตอบที่ประกอบไปด้วยรอบในการทดสอบ (round All), รอบของกระบวนการ (round process), จำนวนสถานี (best station), รอบเวลาการผลิตที่ต่ำที่สุด (best CT) และเวลาในการประมวลผลของคำตอบ (best time) ที่แสดงเป็นหน่วยมิลลิวินาที (millisecond; ms) โดยที่ 1000 มิลลิวินาที (ms) จะเท่ากับ 1 วินาที (s)

4.4 ผลของการจัดสมดุลสายการประกอบและการเปรียบเทียบผล

หลังจากได้นำวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) ไปแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) เพื่อหารอบเวลาการผลิต (cycle time) ที่ต่ำที่สุด โดยใช้ข้อมูลจากชุดปัญหาของ Scholl จำนวน 15 ชุดปัญหา (101 ปัญหา) แบ่งเป็นปัญหาขนาดกลาง (21-58 งาน) 10 ชุดปัญหา (50 ปัญหา) และปัญหาขนาดใหญ่ (70-297 งาน) 5 ชุดปัญหา (51 ปัญหา) มาทำการทดสอบโปรแกรม DE2 และวัดประสิทธิภาพของวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) โดยทำการเปรียบเทียบผลของรอบเวลาการผลิต (Optimal cycle time; OPT), เวลาในการค้นหาคำตอบ (Computational time; CPU, second), เวลาในการค้นหาคำตอบโดยเฉลี่ย (Mean Computational Time; MCPU), เปอร์เซ็นต์ผลต่างรอบเวลาการผลิตของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Two Rules (Percentage difference cycle time; PDCT-1) และ เปอร์เซ็นต์ผลต่างรอบเวลาการผลิตของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Three Rules (Percentage difference cycle time; PDCT-2) กับ Rules-based heuristic ของ Li et al. [9] ที่ทำการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) โดยใช้ 3 Rules ประกอบไปด้วย Task selection rule, Task assignment rule และTask exchange rule ซึ่งในการหาคำตอบได้แบ่งการประมวลออกเป็น 2 วิธี คือ วิธี Two Rules (Task selection rule, Task assignment rule) และวิธี Three Rules (Task selection rule, Task assignment rule, Task exchange rule) สามารถแสดงผลของวิธี Two Rules ในปัญหาขนาดกลางและขนาดใหญ่ดังตาราง 13 และ 14 และแสดงผลของวิธี Three Rules ในปัญหาขนาดกลางและขนาดใหญ่ดังตาราง 15 และ 16

4.4.1 ผลการเปรียบเทียบการจัดสมดุลสายประกอบกับวิธี Two Rules

ตาราง 13 การเปรียบเทียบผลของปัญหาขนาดกลาง (21-58 งาน) กับวิธี Two Rules

Example	*m*	Rules-based heuristic Two Rules			Differential Evolution (DE)			
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-1
Mitchell-21	4	27	0.24	0.28	27	0.02	0.03	0.00
	5	22	0.30		**21**	0.03		-4.76
	6	18	0.31		18	0.03		0.00
Rosenberg-25	4	33	0.22	0.27	**32**	0.06	0.04	-3.13
	5	25	0.14		25	0.03		0.00
	6	23	0.19		**21**	0.03		-9.52

ตาราง 13 การเปรียบเทียบผลของปัญหาขนาดกลาง (21-58 งาน) กับวิธี Two Rules (ต่อ)

Example	*m*	Rules-based heuristic Two Rules			Differential Evolution (DE)			
		OPT	OPT	MCPU	OPT	CPU	MCPU	PDCT-1
Rosenberg-25	7	18	0.35		18	0.03		0.00
	8	16	0.43		16	0.04		0.00
Heskiaoff-28	6	171	1.16	0.64	171	0.03	0.04	0.00
	7	152	0.29		**147**	0.03		-3.40
	8	129	0.82		129	0.04		0.00
	9	117	0.63		**116**	0.05		-0.86
	10	109	0.27		**108**	0.06		-0.93
Buxey-29	5	69	0.23	0.25	**65**	0.02	0.03	-6.15
	6	56	0.34		**54**	0.03		-3.70
	7	50	0.24		**47**	0.03		-6.38
	8	44	0.25		**41**	0.03		-7.32
	9	39	0.16		**38**	0.05		-2.63
	10	35	0.29		**33**	0.03		-6.06
Sawyer-30	5	65	0.79	0.78	65	0.03	0.04	0.00
	6	55	0.80		**54**	0.03		-1.85
	7	48	0.82		**47**	0.04		-2.13
	8	41	0.71		41	0.05		0.00
Lutz1-32	6	2448	0.22	0.25	**2424**	0.03	0.05	-0.99
	7	2092	0.30		2092	0.04		0.00
	8	1830	0.18		**1816**	0.05		-0.77
	9	1644	0.27		**1624**	0.05		-1.23
	10	1474	0.30		1474	0.06		0.00
Gunther-35	5	98	0.97	0.72	**97**	0.05	0.05	-1.03
	6	83	0.56		**81**	0.05		-2.47
	7	71	0.65		**69**	0.05		-2.90
	8	62	0.70		**61**	0.05		-1.64
Kilbridge&Wester-45	6	95	0.61	0.81	**92**	0.08	0.11	-3.26
	7	83	0.88		**79**	0.10		-5.06
	8	73	0.89		**69**	0.11		-5.80

ตาราง 13 การเปรียบเทียบผลของปัญหาขนาดกลาง (21-58 งาน) กับวิธี Two Rules (ต่อ)

Example	*m*	Rules-based heuristic Two Rules			Differential Evolution (DE)			
		OPT	OPT	MCPU	OPT	CPU	MCPU	PDCT-1
Kilbridge&Wester-45	9	66	0.77		**62**	0.12		-6.45
	10	59	0.89		**56**	0.14		-5.36
Hahm-53	5	2948	0.23	0.33	**2820**	0.09	0.14	-4.54
	6	2420	0.36		**2416**	0.12		-0.17
	7	2082	0.51		**2010**	0.14		-3.58
	8	1840	0.44		**1775**	0.16		-3.66
	9	2193	0.11		**1775**	0.18		-23.55
Warnecke-58	10	162	0.92	1.01	**155**	0.24	0.32	-4.52
	11	149	0.96		**143**	0.26		-4.20
	12	136	0.76		**131**	0.28		-3.82
	13	127	0.95		**120**	0.31		-5.83
	14	115	1.27		**111**	0.33		-3.60
	15	110	0.88		**104**	0.35		-5.77
	16	103	1.03		**97**	0.37		-6.19
	17	96	1.34		**94**	0.40		-2.13

หมายเหตุ: **0** = ได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules

0 = ได้รอบเวลาการผลิตเท่ากับวิธี Two Rules

0* = ได้รอบเวลาการผลิตที่แย่กว่าวิธี Two Rules

จากตาราง 13 แสดงผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุดด้วยวิธีวิวัฒนาการโดยใช้ผลต่างและเปรียบเทียบผลกับวิธี Two Rules ของ Rules-based heuristic ในปัญหาขนาดกลาง (21-58 งาน) 10 ชุดปัญหา สามารถจำแนกได้ทั้งหมด 50 ปัญหา เมื่อเปรียบเทียบผลของรอบเวลาการผลิต (OPT) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างสามารถหารอบเวลาการผลิตที่ดีกว่าวิธี Two Rules จำนวน 39 ปัญหา, ได้รอบเวลาการผลิตที่เท่ากับวิธี Two Rules จำนวน 11 ปัญหา และไม่มีปัญหาใดที่ได้รอบเวลาการผลิตแย่กว่าวิธี Two Rules

เมื่อวิเคราะห์เวลาในการค้นหาคำตอบ (CPU) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบทั้ง 50 ปัญหาเท่ากับ 0.10 วินาทีต่อปัญหา ซึ่งน้อยกว่าวิธี Two Rules ที่ใช้เวลาใน

การค้นหาคำตอบทั้ง 50 ปัญหาเท่ากับ 0.56 วินาทีต่อปัญหา และเมื่อวิเคราะห์เวลาในการค้นหาคำตอบโดยเฉลี่ย (MCPU) ในแต่ละปัญหาจะเห็นได้ว่าวิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบโดยเฉลี่ยแต่ละชุดปัญหาทั้ง 10 ชุดปัญหาน้อยกว่าวิธี Two Rules

เมื่อวิเคราะห์เปอร์เซ็นต์ผลต่างรอบเวลาการผลิต (PDCT-1) ในแต่ละปัญหากับวิธี Two Rules ของ Rules-based heuristic จากตาราง 13 พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถหาคำตอบของทุกปัญหาได้ผลของรอบเวลาการผลิตเทียบเท่าและดีกว่าวิธี Two Rules ในทุกปัญหา ซึ่งในปัญหาที่สามารถหารอบเวลาการผลิตได้ดีกว่าที่มีผลต่างของรอบเวลาการผลิตมากที่สุดคือปัญหาของ Hahm-53-9 ที่ร้อยละ 23.55 และเมื่อนำวิธีวิวัฒนาการโดยใช้ผลต่างไปทำการทดลองการจัดสมดุลสายการประกอบในปัญหาขนาดใหญ่ (70-297 งาน) สามารถแสดงผลได้ดังตาราง 14

ตาราง 14 การเปรียบเทียบผลของปัญหาขนาดใหญ่ (70-297 งาน) กับวิธี Two Rules

Example	*m*	Rules-based heuristic			Differential Evolution (DE)			
		Two Rules						
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-1
Tonge-70	10	359	1.54	1.55	359	0.39	0.51	0.00
	11	323	2.03		**320**	0.44		-0.94
	12	298	2.19		**293**	0.46		-1.71
	13	275	1.14		275	0.54		0.00
	14	255	1.45		255	0.55		0.00
	15	237	1.26		237	0.60		0.00
	16	232	1.24		232	0.61		0.00
Lutz2-89	13	40	1.91	2.22	**38**	0.95	1.24	-5.26
	14	37	1.92		**35**	1.03		-5.71
	15	33	2.74		33	1.00		0.00
	16	31	2.57		31	1.17		0.00
	17	30	2.38		**29**	1.24		-3.45
	18	29	1.72		**27**	1.32		-7.41
	19	27	2.42		**26**	1.40		-3.85
	20	27	1.71		**25**	1.47		-8.00
	21	24	2.59		24	1.56		0.00
Mukherjee-94	14	307	3.67	3.50	307	0.66	0.89	0.00
	15	284	6.12		**283**	0.69		-0.35
	16	267	3.65		**265**	0.72		-0.75

ตาราง 14 การเปรียบเทียบผลของปัญหาขนาดใหญ่ (70-297 งาน) กับวิธี Two Rules (ต่อ)

Example	*m*	Rules-based heuristic Two Rules			Differential Evolution (DE)			
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-1
Mukherjee-94	17	252	3.01		**251**	0.79		-0.40
	18	237	4.55		**236**	0.85		-0.42
	19	224	4.03		224	0.90		0.00
	20	223	2.85		**221**	0.95		-0.90
	21	211	4.47		**210**	0.99		-0.48
	22	203	2.04		**201**	1.03		-1.00
	23	193	1.64		193	1.09		0.00
	24	186	2.47		186	1.14		0.00
Bartholdi-148	7	806	19.04	13.22	**805**	2.19	3.31	-0.12
	8	712	12.00		**705**	2.51		-0.99
	9	628	16.38		**626**	2.85		-0.32
	10	564	14.30		564	3.14		0.00
	11	520	10.27		**513**	3.46		-1.36
	12	475	10.03		**470**	3.79		-1.06
	13	437	11.12		**434**	4.10		-0.69
	14	405	12.58		**403**	4.46		-0.50
Scholl-297	21	3490	10.13	15.52	**3460**	20.10	29.74	-0.87
	23	3189	10.27		**3133**	21.65		-1.79
	25	2931	11.32		**2925**	22.57		-0.21
	27	2695	25.45		**2684**	24.26		-0.41
	29	2526	14.11		**2519**	25.46		-0.28
	31	2366	10.67		**2358**	25.25		-0.34
	33	2203	16.08		**2144**	27.55		-2.75
	35	2011	41.62		**2001**	28.10		-0.50
	37	1983	13.43		**1980**	29.22		-0.15
	39	1882	14.91		**1880**	31.02		-0.11
	41	1779	17.18		**1774**	32.22		-0.28

ตาราง 14 การเปรียบเทียบผลของปัญหาขนาดใหญ่ (70-297 งาน) กับวิธี Two Rules (ต่อ)

Example	*m*	Rules-based heuristic Two Rules			Differential Evolution (DE)			
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-1
Scholl-297	43	1706	13.87		**1697**	33.12		-0.53
	45	1620	12.98		**1616**	37.55		-0.25
	47	1562	8.67		**1544**	38.45		-1.17
	49	1496	15.44		**1479**	39.25		-1.15
	50	1458	12.13		**1421**	40.13		-2.60

หมายเหตุ: **0** = ได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules

0 = ได้รอบเวลาการผลิตเท่ากับวิธี Two Rules

0* = ได้รอบเวลาการผลิตที่แย่กว่าวิธี Two Rules

เมื่อนำวิธีวิวัฒนาการโดยใช้ผลต่างไปทำการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุดด้วยในปัญหาขนาดใหญ่ (70-297 งาน) 5 ชุดปัญหา สามารถจำแนกได้ทั้งหมด 51 ปัญหาและเปรียบเทียบผลกับวิธี Two Rules ของ Rules-based heuristic แสดงดังตาราง 14 เมื่อเปรียบเทียบผลของรอบเวลาการผลิต (OPT) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างสามารถหารอบเวลาการผลิตที่ดีกว่าวิธี Two Rules จำนวน 38 ปัญหา, ได้รอบเวลาการผลิตที่เท่ากับวิธี Two Rules จำนวน 13 ปัญหา และไม่มีปัญหาใดที่ได้รอบเวลาการผลิตแย่กว่าวิธี Two Rules

เมื่อวิเคราะห์เวลาในการค้นหาคำตอบ (CPU) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบทั้ง 51 ปัญหาเท่ากับ 10.33 วินาทีต่อปัญหา ซึ่งใช้เวลามากกว่าวิธี Two Rules ที่ใช้เวลาในการค้นหาคำตอบทั้ง 51 ปัญหาเท่ากับ 8.30 วินาทีต่อปัญหา และเมื่อวิเคราะห์เวลาในการค้นหาคำตอบโดยเฉลี่ย (MCPU) ในแต่ละปัญหาจะเห็นได้ว่าวิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบโดยเฉลี่ยน้อยกว่าวิธี Two Rules ใน 4 ชุดปัญหา แต่ในชุดปัญหาของ Scholl-297 จำนวน 16 ปัญหาที่ถึงแม้วิธีวิวัฒนาการโดยใช้ผลต่างจะสามารถค้นหาคำตอบได้ดีกว่าวิธี Three Rules แต่เมื่อเทียบเวลาในการค้นหาคำตอบโดยเฉลี่ยวิธีวิวัฒนาการโดยใช้ผลต่างยังคงใช้เวลาในการค้นหามากกว่าวิธี Two Rules

เมื่อวิเคราะห์เปอร์เซ็นต์ผลต่างรอบเวลาการผลิต (PDCT-1) ในแต่ละปัญหากับวิธี Two Rules ของ Rules-based heuristic จากตาราง 14 พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถหาคำตอบของทุกปัญหาได้ผลของรอบเวลาการผลิตเทียบเท่าและดีกว่าวิธี Three Rules ในทุกปัญหา

ซึ่งในปัญหาที่สามารถหารอบเวลาการผลิตได้ดีกว่าที่มีผลต่างของรอบเวลาการผลิตมากที่สุดคือปัญหาของ Lutz2-89-18 ที่ร้อยละ 7.41

4.4.2 ผลการเปรียบเทียบการจัดสมดุลสายประกอบกับวิธี Three Rules

ตาราง 15 การเปรียบเทียบผลของปัญหาขนาดกลาง (21-58 งาน) กับวิธี Three Rules

Example	*m*	Rules-based heuristic			Differential Evolution (DE)			
		Three Rules						
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-2
Mitchell-21	4	27	0.26	0.24	27	0.02	0.03	0.00
	5	21	0.10		21	0.03		0.00
	6	18	0.37		18	0.03		0.00
Rosenberg-25	4	33	0.27	0.35	**32**	0.06	0.04	-3.13
	5	25	0.11		25	0.03		0.00
	6	21	0.43		21	0.03		0.00
	7	18	0.43		18	0.03		0.00
	8	16	0.50		16	0.04		0.00
Heskiaoff-28	6	171	1.27	0.92	171	0.03	0.04	0.00
	7	147	1.18		147	0.03		0.00
	8	129	0.82		129	0.04		0.00
	9	116	0.58		116	0.05		0.00
	10	108	0.77		108	0.06		0.00
Buxey-29	5	66	0.50	0.54	**65**	0.02	0.03	-1.54
	6	55	0.60		**54**	0.03		-1.85
	7	48	0.55		**47**	0.03		-2.13
	8	41	0.59		41	0.03		0.00
	9	38	0.45		38	0.05		0.00
	10	34	0.57		**33**	0.03		-3.03
Sawyer-30	5	65	0.70	0.61	65	0.03	0.04	0.00
	6	54	0.17		54	0.03		0.00
	7	47	0.73		47	0.04		0.00
	8	41	0.84		41	0.05		0.00

ตาราง 15 การเปรียบเทียบผลของปัญหาขนาดกลาง (21-58 งาน) กับวิธี Three Rules (ต่อ)

Example	*m*	Rules-based heuristic Three Rules			Differential Evolution (DE)			
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-2
Lutz1-32	6	2404	0.49	0.34	2424*	0.03	0.05	0.83
	7	2092	0.37		2092	0.04		0.00
	8	1816	0.38		1816	0.05		0.00
	9	1622	0.16		1624*	0.05		0.12
	10	1474	0.31		1474	0.06		0.00
Gunther-35	5	97	1.09	0.85	97	0.05	0.05	0.00
	6	82	0.73		**81**	0.05		-1.23
	7	70	0.75		**69**	0.05		-1.45
	8	61	0.82		61	0.05		0.00
Kilbridge&Wester-45	6	92	0.41	1.22	92	0.08	0.11	0.00
	7	79	1.90		79	0.10		0.00
	8	69	0.39		69	0.11		0.00
	9	62	1.61		62	0.12		0.00
	10	56	1.77		56	0.14		0.00
Hahm-53	5	2823	1.57	1.17	**2820**	0.09	0.14	-0.11
	6	2416	1.21		2416	0.12		0.00
	7	2013	1.71		**2010**	0.14		-0.15
	8	1827	1.21		**1775**	0.16		-2.93
	9	1775	0.15		1775	0.18		0.00
Warnecke-58	10	155	2.66	2.28	155	0.24	0.32	0.00
	11	143	2.59		143	0.26		0.00
	12	131	2.29		131	0.28		0.00
	13	122	2.15		**120**	0.31		-1.67
	14	112	2.14		**111**	0.33		-0.90
	15	105	2.50		**104**	0.35		-0.96
	16	98	2.10		**97**	0.37		-1.03
	17	94	1.83		94	0.40		0.00

หมายเหตุ: **0** = ได้รอบเวลาการผลิตที่ดีกว่าวิธี Three Rules

0 = ได้รอบเวลาการผลิตเท่ากับวิธี Three Rules

0* = ได้รอบเวลาการผลิตที่แย่กว่าวิธี Three Rules

จากตาราง 15 แสดงผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุดด้วยวิธีวิวัฒนาการโดยใช้ผลต่างและเปรียบเทียบผลกับวิธี Three Rules ของ Rules-based heuristic ในปัญหาขนาดกลาง (21-58 งาน) 10 ชุดปัญหา สามารถจำแนกได้ทั้งหมด 50 ปัญหา เมื่อเปรียบเทียบผลของรอบเวลาการผลิต (OPT) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างสามารถหารอบเวลาการผลิตที่ดีกว่าวิธี Three Rules จำนวน 14 ปัญหา, ได้รอบเวลาการผลิตที่เท่ากับวิธี Three Rules จำนวน 34 ปัญหา และได้รอบเวลาการผลิตแย่กว่าวิธี Three Rules จำนวน 2 ปัญหา ซึ่งพบในปัญหาของ Lutz1-32-6 และ Lutz1-32-9 ที่ยังไม่สามารถหาคำตอบของรอบเวลาการผลิตที่ดีกว่าหรือเทียบเท่าวิธี Three Rules

เมื่อวิเคราะห์เวลาในการค้นหาคำตอบ (CPU) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบทั้ง 50 ปัญหาเท่ากับ 0.10 วินาทีต่อปัญหา ซึ่งน้อยกว่าวิธี Three Rules ที่ใช้เวลาในการค้นหาคำตอบทั้ง 50 ปัญหาเท่ากับ 0.96 วินาทีต่อปัญหา และเมื่อวิเคราะห์เวลาในการค้นหาคำตอบโดยเฉลี่ย (MCPU) ในแต่ละปัญหาจะเห็นได้ว่าวิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบโดยเฉลี่ยแต่ละชุดปัญหาทั้ง 10 ชุดปัญหาน้อยกว่าวิธี Three Rules

เมื่อวิเคราะห์เปอร์เซ็นต์ผลต่างรอบเวลาการผลิต (PDCT-2) ในแต่ละปัญหากับวิธี Three Rules ของ Rules-based heuristic จากตาราง 15 พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถหาคำตอบของทุกปัญหาได้ผลของรอบเวลาการผลิตเทียบเท่าและดีกว่าวิธี Three Rules เกือบทุกปัญหา ซึ่งในปัญหาที่สามารถหารอบเวลาการผลิตได้ดีกว่าที่มีผลต่างของรอบเวลาการผลิตมากที่สุดคือปัญหาของ Rosenberg-25-4 ที่ร้อยละ 3.13 แต่ในปัญหาของ Lutz1-32-6 และ Lutz1-32-9 วิธี DE ยังหาคำตอบได้รอบเวลาการผลิตที่มากกว่าวิธี Three Rules ที่ร้อยละ 0.83 และร้อยละ 0.12 ตามลำดับ และเมื่อนำวิธีวิวัฒนาการโดยใช้ผลต่างไปทำการทดลองการจัดสมดุลสายการประกอบในปัญหาขนาดใหญ่ (70-297 งาน) สามารถแสดงผลได้ดังตาราง 16

พหุน ปณฺฑิโต ชีเว

ตาราง 16 การเปรียบเทียบผลของปัญหาขนาดใหญ่ (70-297 งาน) กับ Three Rules

Example	*m*	Rules-based heuristic Three Rules			Differential Evolution (DE)			
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-2
Tonge-70	10	352	3.87	2.60	359*	0.39	0.51	1.95
	11	320	3.28		320	0.44		0.00
	12	295	2.90		**293**	0.46		-0.68
	13	273	2.15		275*	0.54		0.73
	14	252	2.46		255*	0.55		1.18
	15	236	2.00		237*	0.60		0.42
	16	222	1.52		232*	0.61		4.31
Lutz2-89	13	38	2.88	2.68	38	0.95	1.24	0.00
	14	35	2.89		35	1.03		0.00
	15	33	2.78		33	1.00		0.00
	16	31	2.62		31	1.17		0.00
	17	30	2.45		**29**	1.24		-3.45
	18	27	2.73		27	1.32		0.00
	19	27	2.32		**26**	1.40		-3.85
	20	25	2.65		25	1.47		0.00
	21	24	2.76		24	1.56		0.00
Mukherjee-94	14	302	8.12	6.36	307*	0.66	0.89	1.63
	15	283	6.92		283	0.69		0.00
	16	265	6.31		265	0.72		0.00
	17	251	5.71		251	0.79		0.00
	18	236	6.05		236	0.85		0.00
	19	224	4.44		224	0.90		0.00
	20	212	5.37		221*	0.95		4.07
	21	205	8.07		210*	0.99		2.38
	22	193	4.16		201*	1.03		3.98
	23	185	7.54		193*	1.09		4.15
	24	179	7.24		186*	1.14		3.76

ตาราง 16 การเปรียบเทียบผลของปัญหาขนาดใหญ่ (70-297 งาน) กับ Three Rules (ต่อ)

Example	*m*	Rules-based heuristic Three Rules			Differential Evolution (DE)			
		OPT	CPU	MCPU	OPT	CPU	MCPU	PDCT-2
Bartholdi-148	7	805	20.77	17.42	805	2.19	3.31	0.00
	8	705	20.23		705	2.51		0.00
	9	626	11.33		626	2.85		0.00
	10	564	15.87		564	3.14		0.00
	11	513	13.68		513	3.46		0.00
	12	470	10.51		470	3.79		0.00
	13	434	24.60		434	4.10		0.00
	14	403	22.35		403	4.46		0.00
Scholl-297	21	3328	35.65	48.09	3460*	20.10	29.74	3.82
	23	3034	53.13		3133*	21.65		3.16
	25	2790	47.36		2925*	22.57		4.62
	27	2582	69.36		2684*	24.26		3.80
	29	2406	48.90		2519*	25.46		4.49
	31	2256	43.26		2358*	25.25		4.33
	33	2114	56.53		2144*	27.55		1.40
	35	1993	62.17		2001*	28.10		0.40
	37	1888	45.60		1980*	29.22		4.65
	39	1792	40.66		1880*	31.02		4.68
	41	1702	58.57		1774*	32.22		4.06
	43	1623	36.72		1697*	33.12		4.36
	45	1551	53.30		1616*	37.55		4.02
	47	1488	42.92		1544*	38.45		3.63
	49	1438	34.72		1479*	39.25		2.77
	50	1398	40.51		1421*	40.13		1.62

หมายเหตุ: **0** = ได้รอบเวลาการผลิตที่ดีกว่าวิธี Three Rules

0 = ได้รอบเวลาการผลิตเท่ากับวิธี Three Rules

0* = ได้รอบเวลาการผลิตที่แย่กว่าวิธี Three Rules

เมื่อนำวิธีวิวัฒนาการโดยใช้ผลต่างไปทำการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุดด้วยในปัญหาขนาดใหญ่ (70-297 งาน) 5 ชุดปัญหา สามารถจำแนกได้ทั้งหมด 51 ปัญหาและเปรียบเทียบผลกับวิธี Three Rules ของ Rules-based heuristic แสดงดังตาราง 16 เมื่อเปรียบเทียบผลของรอบเวลาการผลิต (OPT) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างสามารถหารอบเวลาการผลิตที่ดีกว่าวิธี Three Rules จำนวน 3 ปัญหา, ได้รอบเวลาการผลิตที่เท่ากับวิธี Three Rules จำนวน 21 ปัญหา และได้รอบเวลาการผลิตแย่กว่าวิธี Three Rules จำนวน 27 ปัญหา ซึ่งพบในชุดปัญหาของ Tonge-70, Mukherjee-94 และ Scholl-297 ที่ยังไม่สามารถหาคำตอบของรอบเวลาการผลิตที่ดีกว่าหรือเทียบเท่าวิธี Three Rules

เมื่อวิเคราะห์เวลาในการค้นหาคำตอบ (CPU) พบว่า วิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบทั้ง 51 ปัญหาเท่ากับ 10.33 วินาทีต่อปัญหา ซึ่งใช้เวลาน้อยกว่าวิธี Three Rules ที่ใช้เวลาในการค้นหาคำตอบทั้ง 51 ปัญหาเท่ากับ 20.02 วินาทีต่อปัญหา และเมื่อวิเคราะห์เวลาในการค้นหาคำตอบโดยเฉลี่ย (MCPU) ในแต่ละปัญหาจะเห็นได้ว่าวิธีวิวัฒนาการโดยใช้ผลต่างใช้เวลาในการค้นหาคำตอบโดยเฉลี่ยแต่ละชุดปัญหาทั้ง 5 ชุดปัญหาน้อยกว่าวิธี Three Rules

เมื่อวิเคราะห์เปอร์เซ็นต์ผลต่างรอบเวลาการผลิต (PDCT-2) ในแต่ละปัญหากับวิธี Three Rules ของ Rules-based heuristic จากตาราง 16 พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างโดยส่วนมากยังสามารถหาคำตอบของรอบเวลาการผลิตยังได้ไม่ดีหรือเทียบเท่าวิธี Three Rules จำนวน 27 ปัญหาจาก 51 ปัญหา ซึ่งพบในชุดปัญหาปัญหาของ Tonge-70, Mukherjee-94 และ Scholl-297 ที่ยังไม่สามารถหาคำตอบของรอบเวลาการผลิตที่ดีกว่าหรือเทียบเท่าวิธี Three Rules

จากผลของการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างของปัญหาขนาดกลางที่มีจำนวน 21-58 งาน จำนวน 50 ปัญหา และปัญหาขนาดใหญ่ที่มีจำนวนงาน 70-297 งาน จำนวน 51 ปัญหา แสดงดังตาราง 13-16 และเมื่อนำผลของการจัดสมดุลสายการประกอบไปเปรียบเทียบผลกับ Rules-based heuristic ที่ทำการจัดสมดุลสายการประกอบออกเป็น 2 วิธี คือ วิธี Two Rules และวิธี Three Rules เมื่อเปรียบเทียบผลของรอบเวลาการผลิตที่ต่ำที่สุดกับวิธี Two Rules พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าและเทียบเท่าทุกปัญหาทั้งในปัญหาขนาดกลาง (21-58 งาน) และปัญหาขนาดใหญ่ (70-297 งาน) จำนวน 50 ปัญหา และเมื่อเปรียบเทียบกับวิธี Three Rules พบว่าในปัญหาขนาดกลาง (21-58 งาน) วิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าและเทียบเท่าจำนวน 48 ปัญหา แต่มีจำนวน 2 ปัญหาที่วิธีวิวัฒนาการโดยใช้ผลต่างยังไม่สามารถหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Three Rules ส่วนในปัญหาขนาดใหญ่ (70-297 งาน) วิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าและเทียบเท่าจำนวน 24 ปัญหา แต่มีจำนวน 27 ปัญหาที่วิธีวิวัฒนาการโดย

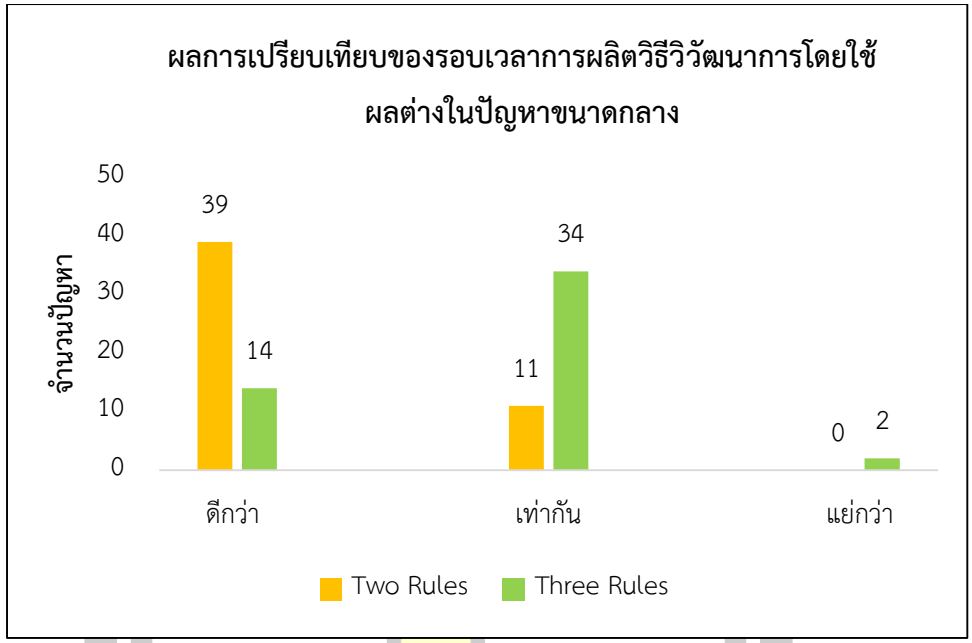
ใช้ผลต่างยังไม่สามารถหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Three Rules ซึ่งเมื่อทำการวิเคราะห์ในจำนวน 29 ปัญหาที่ยังไม่สามารถค้นหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Three Rules จะเห็นได้ว่าโดยส่วนมากเป็นปัญหาขนาดใหญ่ (70-297 งาน) ที่เมื่อมีจำนวนงานและมีเงื่อนไขของลำดับงานเพิ่มมากขึ้น ส่งผลให้กระบวนการสร้างคำตอบเริ่มต้น (Initial population) ของวิธีวิวัฒนาการโดยใช้ผลต่างในส่วนของค่าเวกเตอร์เป้าหมาย (Target Vector) ที่มาจากการสุ่มตัวเลข และเมื่อนำค่าเวกเตอร์เป้าหมายไปจัดงานลงสู่สถานีงานจึงทำให้สามารถสร้างคำตอบเริ่มต้นได้หลากหลาย ซึ่งอาจทำให้ได้ผลของคำตอบเริ่มต้นที่ห่างไกลจากคำตอบที่ดีที่สุด จึงทำให้เมื่อนำคำตอบเริ่มต้นผ่านกระบวนการปรับเปลี่ยนค่าในพิกัด (Mutation) และการแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (Recombination) ของวิธีวิวัฒนาการโดยใช้ผลต่างจนครบทุกระบวนการ ก็ยังทำให้ไม่ได้คำตอบที่ดีที่สุดที่ดีกว่าหรือเทียบเท่าวิธี Three Rules ในวิจัยในอนาคตจึงควรเพิ่มการค้นหาเฉพาะที่ (Local Search) หรือฮิวริสติก (Heuristic) เข้ามาปรับปรุงคำตอบที่ได้จากกระบวนการของวิธีวิวัฒนาการโดยใช้ผลต่างเพื่อทำให้ วิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหาคำตอบในปัญหาขนาดใหญ่ (70-297 งาน) ได้อย่างมีประสิทธิภาพมากขึ้น

4.5 สรุปผลการวิจัย

จากการทำการทดสอบการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด กรณีทราบจำนวนสถานีงาน ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างผ่านโปรแกรม DE2 บนเครื่องคอมพิวเตอร์ Intel(R) Core(TM) i5-8500 CPU@3.0 GHz, 4GB RAM โดยใช้ชุดปัญหาของ Scholl [19] จาก http://www.assembly-line-balancing.de จำนวน 15 ชุดปัญหา (101 ปัญหา) แบ่งเป็นปัญหาขนาดกลางที่มีจำนวน 21-58 งาน จำนวน 10 ชุดปัญหา (50 ปัญหา) และปัญหาขนาดใหญ่ที่มีจำนวนงาน 70-297 งาน จำนวน 5 ชุดปัญหา (51 ปัญหา) และนำผลของการจัดสมดุลสายการประกอบไปเปรียบเทียบผลกับว Rules-based heuristic ของ Li et al. [9] ที่ทำการจัดสมดุลสายการประกอบออกเป็น 2 วิธี คือ วิธี Two Rules และวิธี Three Rules สามรถสรุปผลของรอบเวลาการผลิตและผลของเวลาที่ใช้ในการหาคำตอบได้ดังนี้

4.5.1 ผลของรอบเวลาการผลิต

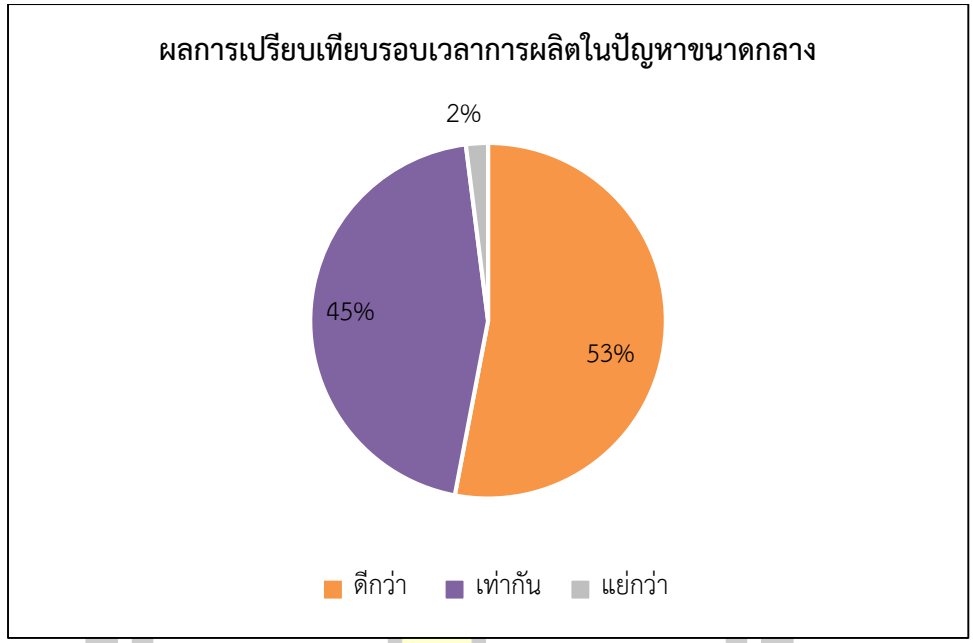
เมื่อทำการการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุดในปัญหาขนาดกลางที่มีจำนวน 21-58 งาน จำนวน 10 ชุดปัญหา (50 ปัญหา) และเมื่อเปรียบเทียบผลกับวิธี Two Rules และวิธี Three Rules ของ Rules-based heuristic สามารถแสดงดังภาพประกอบ 16



ภาพประกอบ 16 การเปรียบเทียบรอบเวลาการผลิตในปัญหาขนาดกลาง

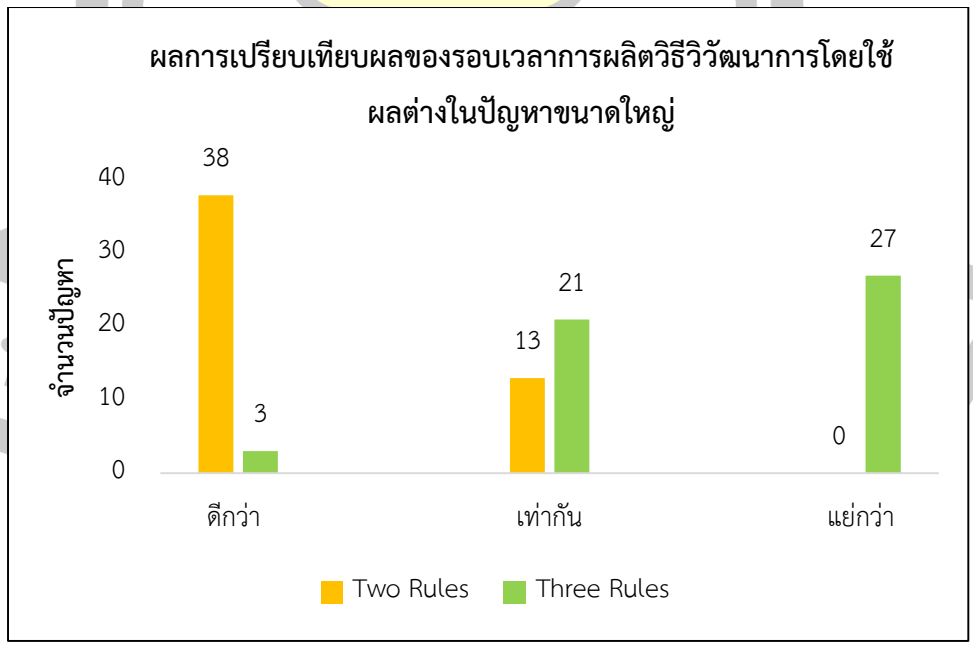
จากภาพประกอบ 16 แสดงผลการเปรียบเทียบของรอบเวลาการผลิตของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Two Rules และวิธี Three Rules ของ Rules-based heuristic ในปัญหาขนาดกลาง (21-58 งาน) จำนวน 50 ปัญหา พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules จำนวน 39 ปัญหา หรือคิดเป็นร้อยละ 78 และได้รอบการผลิตที่ดีกว่าวิธี Three Rules จำนวน 14 ปัญหา หรือคิดเป็นร้อยละ 28 ได้รอบเวลาการผลิตที่เท่ากันกับวิธี Two Rules จำนวน 11 หรือคิดเป็นร้อยละ 22 และได้รอบเวลาการผลิตที่เท่ากันกับวิธี Three Rules จำนวน 34 ปัญหา หรือคิดเป็นร้อยละ 68 ส่วนการจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่แย่กว่าพบเพียงในวิธี Three Rules จำนวน 2 ปัญหา หรือคิดเป็นร้อยละ 4

เมื่อวิเคราะห์ผลโดยรวมของการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 ด้วยวิธีวิวัฒนาการโดยผลต่างในการทดสอบกับวิธี Two Rules และวิธี Three Rules ซึ่งรวมเป็น 100 ปัญหา พบว่าวิธีวิวัฒนาการโดยผลต่างสามารถจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules และวิธี Three Rules จำนวน 53 ปัญหา หรือคิดเป็นร้อยละ 53 ได้รอบเวลาการผลิตที่เท่ากันจำนวน 45 ปัญหา หรือคิดเป็นร้อยละ 45 และได้รอบเวลาการผลิตที่แย่กว่าจำนวน 2 ปัญหา หรือคิดเป็นร้อยละ 2 สามารถแสดงดังภาพประกอบ 17



ภาพประกอบ 17 การเปรียบเทียบของจำนวนปัญหาในปัญหาขนาดกลาง

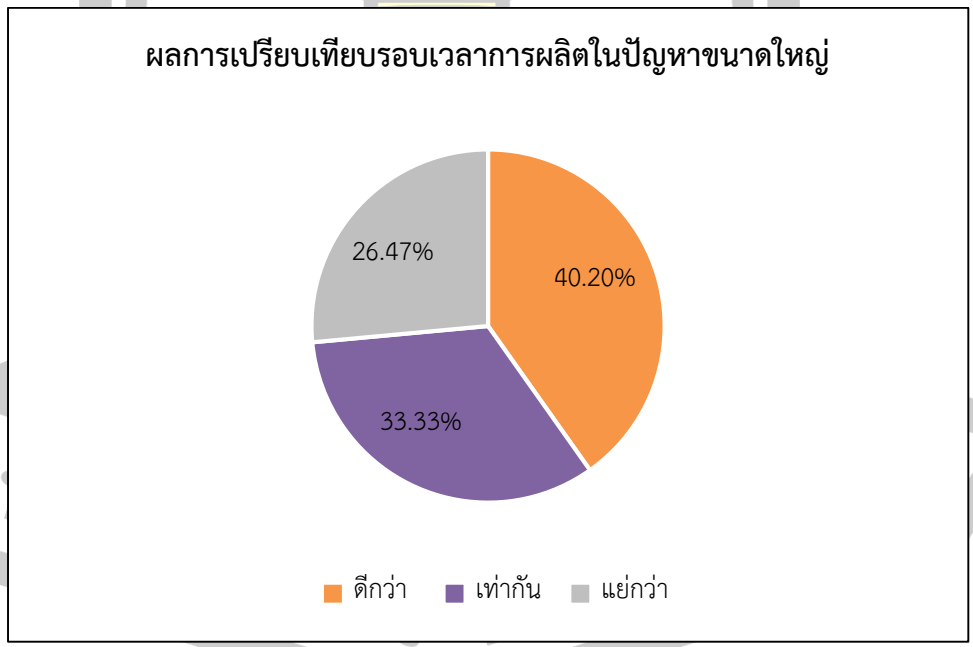
เมื่อทำการการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุดในปัญหาขนาดใหญ่ที่มีจำนวน 70-297 งาน จำนวน 5 ชุดปัญหา (51 ปัญหา) และเปรียบเทียบผลกับวิธี Two Rules และวิธี Three Rules ของ Rules-based heuristic สามารถแสดงได้ดังภาพประกอบ 18



ภาพประกอบ 18 การเปรียบเทียบรอบเวลาการผลิตในปัญหาขนาดใหญ่

จากภาพประกอบ 18 แสดงผลการเปรียบเทียบของรอบเวลาการผลิตของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Two Rules และวิธี Three Rules ของ Rules-based heuristic ในปัญหาขนาดใหญ่ (70-297 งาน) จำนวน 51 ปัญหา พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules จำนวน 38 ปัญหา หรือคิดเป็นร้อยละ 74.50 และได้รอบการผลิตที่ดีกว่าวิธี Three Rules จำนวน 3 ปัญหา หรือคิดเป็นร้อยละ 5.88 ได้รอบเวลาการผลิตที่เท่ากันกับวิธี Two Rules จำนวน 13 หรือคิดเป็นร้อยละ 5.88 และได้รอบเวลาการผลิตที่เท่ากันกับวิธี Three Rules จำนวน 21 ปัญหา หรือคิดเป็นร้อยละ 25.50 ส่วนการจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่แย่กว่าพบเพียงในวิธี Three Rules จำนวน 27 ปัญหา หรือคิดเป็นร้อยละ 53

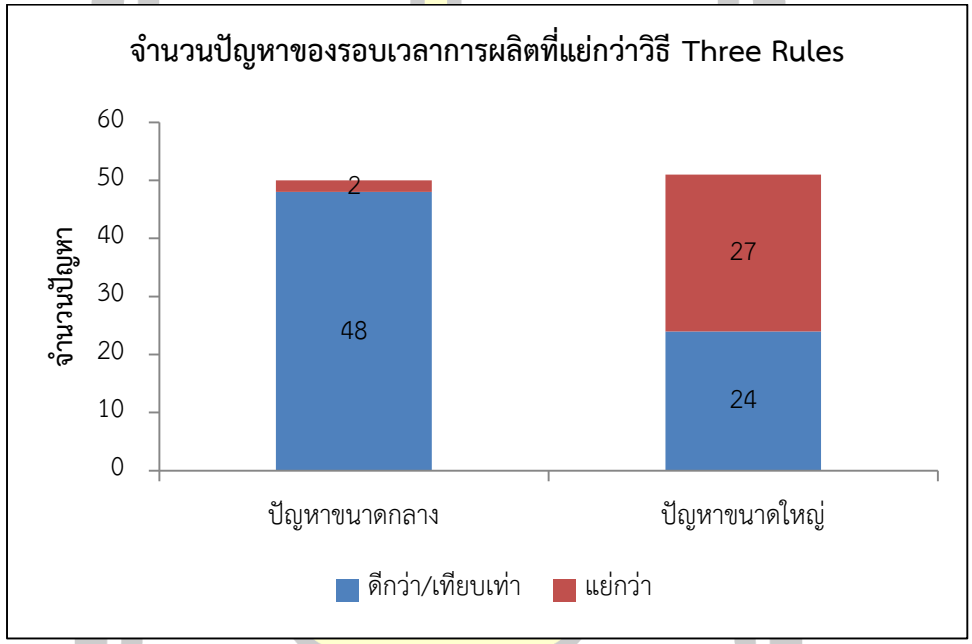
เมื่อวิเคราะห์ผลโดยรวมของการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 ด้วยวิธีวิวัฒนาการโดยผลต่างในการทดสอบกับวิธี Two Rules และวิธี Three Rules ซึ่งรวมเป็น 102 ปัญหา พบว่าวิธีวิวัฒนาการโดยผลต่างสามารถจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules และวิธี Three Rules จำนวน 41 ปัญหา หรือคิดเป็นร้อยละ 40.20 ได้รอบเวลาการผลิตที่เท่ากันจำนวน 34 ปัญหา หรือคิดเป็นร้อยละ 33.33 และได้รอบเวลาการผลิตที่แย่กว่าจำนวน 27 ปัญหา หรือคิดเป็นร้อยละ 26.47 สามารถแสดงดังภาพประกอบ 19



ภาพประกอบ 19 การเปรียบเทียบของจำนวนปัญหาในปัญหาขนาดใหญ่

จากผลการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างและทำการเปรียบเทียบผลกับวิธี Two Rules และวิธี

Three Rules ของ Rules-based heuristic ทั้งในปัญหาขนาดกลาง (21-58 งาน) จำนวน 50 ปัญหา และปัญหาขนาดใหญ่ (70-297 งาน) จำนวน 51 ปัญหา พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Two Rules ในทุกปัญหาแต่เมื่อเปรียบเทียบกับวิธี Three Rules วิธีวิวัฒนาการโดยใช้ผลต่างยังไม่สามารถจัดสมดุลสายการประกอบได้รอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าในปัญหาขนาดกลาง จำนวน 2 ปัญหา และในปัญหาขนาดใหญ่จำนวน 27 ปัญหา แสดงดังภาพประกอบ 20



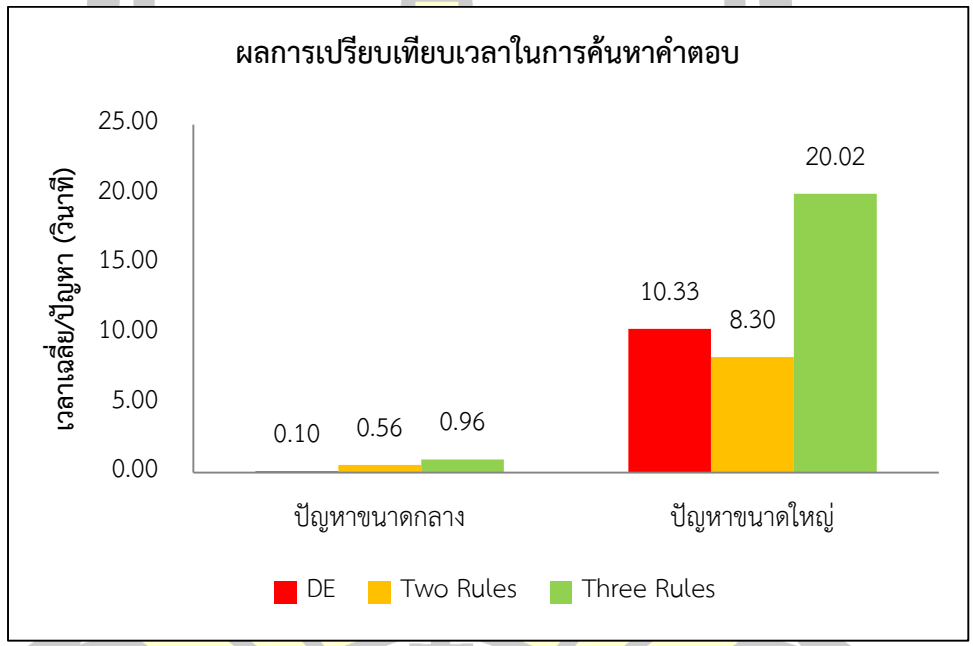
ภาพประกอบ 20 การเปรียบเทียบรอบเวลาการผลิตในปัญหาขนากลาง

จากภาพประกอบ 20 แสดงผลของการหารอบการผลิตที่ต่ำที่สุดของวิธีวิวัฒนาการโดยใช้ผลต่างที่เปรียบเทียบผลกับวิธี Three Rules ทั้งในปัญหาขนาดกลาง (21-58 งาน) และในปัญหาขนาดใหญ่ (70-297 งาน) ทั้งหมด 101 ปัญหาแล้วพบว่า วิธีวิวัฒนาการโดยใช้ผลต่างสามารถหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าจำนวน 72 ปัญหาหรือคิดเป็นร้อยละ 71.29 แต่ในจำนวน 29 ปัญหาหรือคิดเป็นร้อยละ 28.71 ที่วิธีวิวัฒนาการโดยใช้ผลต่างยังไม่สามารถหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Three Rules ซึ่งมาจากปัญหาขนาดใหญ่ (70-297 งาน) เป็นส่วนมาก ที่เมื่อขนาดปัญหาใหญ่ขึ้นจำนวนงานและเงื่อนไขของลำดับงานก็จะเพิ่มมากขึ้น ส่งผลให้กระบวนการของวิธีวิวัฒนาการโดยใช้ผลต่างในกระบวนการสร้างคำตอบเริ่มต้น (Initial population) การปรับเปลี่ยนค่าในพิกัด (Mutation) และการแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (Recombination) ยังไม่สามารถหาคำตอบที่ดีที่สุดที่ได้ดีกว่าหรือเทียบเท่าวิธี Three Rules ในการทำวิจัยในอนาคตจึงควรเพิ่มการค้นหาเฉพาะที่ (Local search) หรือฮิวริสติก

(Heuristic) เข้ามาปรับปรุงคำตอบที่ได้จากกระบวนการของวิธีวิวัฒนาการโดยใช้ผลต่างเพื่อทำให้ วิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหาคำตอบในปัญหาขนาดใหญ่ (70-297 งาน) ได้อย่างมีประสิทธิภาพมากขึ้น

4.5.2 ผลของเวลาที่ใช้ในการค้นหาคำตอบ

หลังจากทำการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 ด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) และเปรียบเทียบผลของรอบเวลาการผลิต (Cycle time) กับวิธี Two Rules และวิธี Three Rules ของ Rules-based heuristic แล้วนั้นผู้วิจัยจึงได้นำผลของรอบเวลาการผลิตที่ต่ำที่สุดที่ได้จากวิธีวิวัฒนาการโดยใช้ผลต่างในแต่ละปัญหามาวัดประสิทธิภาพของเวลาที่ใช้ในการค้นหาคำตอบ (CPU) สามารถแสดงดังภาพประกอบ 21



ภาพประกอบ 21 การเปรียบเทียบเวลาในการค้นหาคำตอบ

จากผลของเวลาในการค้นหาคำตอบจากภาพประกอบ 21 เมื่อนำเวลาในการค้นหาคำตอบที่ดีที่สุดในแต่ละปัญหาของปัญหาขนาดกลาง (21-58 งาน) จำนวน 50 ปัญหา และในปัญหาขนาดใหญ่ (70-297 งาน) จำนวน 51 ปัญหา พบว่าวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) ใช้เวลาเฉลี่ยในการค้นหาคำตอบอยู่ระหว่าง 0.02-0.40 วินาที/ปัญหา สำหรับปัญหาขนาดกลาง และในปัญหาขนาดใหญ่ใช้เวลาเฉลี่ยในการค้นหาคำตอบอยู่ระหว่าง 0.39-40 วินาที/ปัญหา เมื่อวิเคราะห์หาเวลาในการค้นหาโดยเฉลี่ยต่อปัญหาแล้วนำผลของเวลาโดยเฉลี่ยไปเปรียบเทียบกับวิธี Two Rules และวิธี Three Rules ของ Rules-based heuristic พบว่าในปัญหาขนาดกลางวิธีวิวัฒนาการโดยใช้ผลต่าง

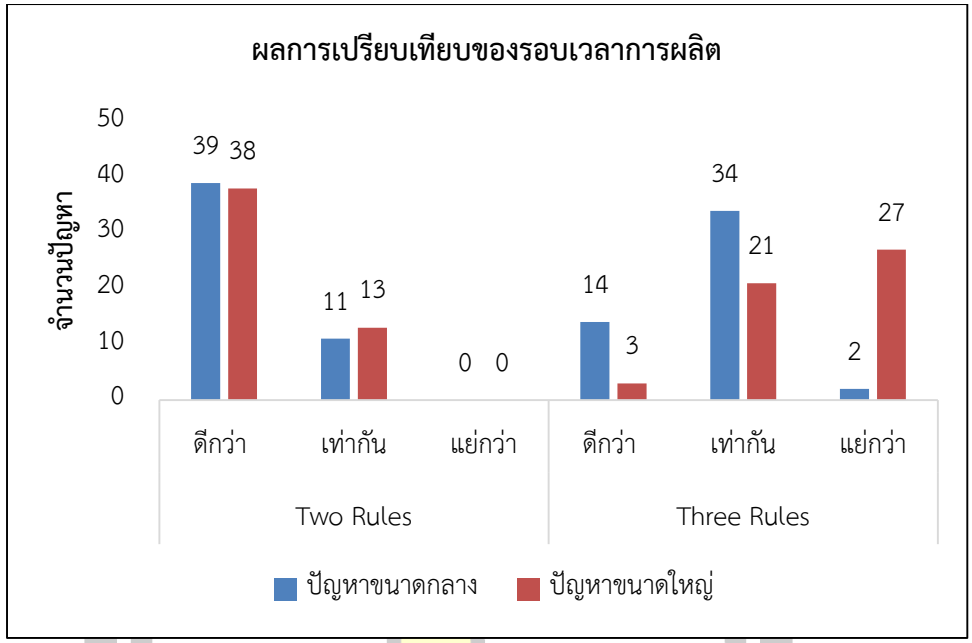
(DE) สามารถใช้เวลาในการค้นหาคำตอบโดยเฉลี่ยน้อยที่สุด 0.10 วินาที/ปัญหา เมื่อเทียบกับวิธี Two Rules และวิธี Three Rules ที่ใช้เวลาในการค้นหาคำตอบโดยเฉลี่ย 0.56 วินาที/ปัญหา และ 0.96 วินาที/ปัญหา ตามลำดับ และเมื่อเปรียบเทียบกับปัญหาขนาดใหญ่ (70-297 งาน) พบว่าวิธี Two Rules สามารถค้นหาคำตอบโดยเฉลี่ยน้อยที่สุด 8.30 วินาที/ปัญหา เมื่อเทียบกับวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) และวิธี Three Rules ที่ใช้เวลาในการค้นหาคำตอบโดยเฉลี่ย 10.33 วินาที/ปัญหา และ 20.02 วินาที/ปัญหา ตามลำดับ



บทที่ 5
สรุปผลการวิจัย

5.1 สรุปผลการวิจัย

งานวิจัยนี้เป็นการนำเสนอวิธีการแก้ปัญหาการจัดสมดุลสายประกอบประเภทที่ 2 (UALBP-2) เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด (c) กรณีทราบจำนวนสถานีงาน (m) ด้วยการประยุกต์ใช้วิธีวิวัฒนาการโดยใช้ผลต่าง (DE) โดยกำหนดพารามิเตอร์ R=30 รอบ, NP=30, F=0.8 และ CR=0.8 และทำการปรับเปลี่ยนพิกัดด้วยสมการ DE/best/2 ร่วมกับการแลกเปลี่ยนค่า (Recombination) แบบเอกซ์โพเนนเชียลแบบ 1 จุด (Exponential crossover 1 position) Sresracoo et al. [18] จากนั้นผู้วิจัยได้ทำการทดสอบโปรแกรม DE2 ที่พัฒนาขึ้นมาผ่านเครื่องคอมพิวเตอร์ Intel(R) Core(TM) i5-8500 CPU@3.0 GHz, 4GB RAM โดยใช้ชุดปัญหาของ Scholl [19] ที่มีจำนวนงานตั้งแต่ 21-297 งาน จำนวน 15 ชุดปัญหา (101 ปัญหา) จาก www.assembly-line-balancing.de ที่สามารถแบ่งออกเป็นปัญหาขนาดกลางที่มีจำนวนงานตั้งแต่ 21-58 งาน จำนวน 10 ชุดปัญหา (50 ปัญหา) และปัญหาขนาดใหญ่ที่มีจำนวนงานตั้งแต่ 70-297 งาน จำนวน 5 ชุดปัญหา (51 ปัญหา) มาทำการหารอบของเวลาการผลิตที่ต่ำที่สุด และวัดประสิทธิภาพของวิธีวิวัฒนาการโดยใช้ผลต่าง (DE) โดยทำการเปรียบเทียบผลของรอบเวลาการผลิต (OPT), เวลาในการค้นหาคำตอบ (CPU, second), เวลาในการค้นหาคำตอบโดยเฉลี่ย (MCPU), เปอร์เซ็นต์ผลต่างรอบเวลาการผลิตของวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Two Rules (PDCT-1) และ เปอร์เซ็นต์ผลต่างรอบเวลาการผลิตของวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธี Three Rules (PDCT-2) กับ Rules-based heuristic ของ Li et Al. [9] ที่ทำการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) โดยใช้ 3 Rules ประกอบไปด้วย Task selection rule, Task assignment rule และ Task exchange rule ซึ่งในการหาคำตอบได้แบ่งการประมวลออกเป็น 2 วิธี คือ วิธี Two Rules (Task selection rule, Task assignment rule) และวิธี Three Rules (Task selection rule, Task assignment rule, Task exchange rule) สามารถสรุปผลได้ดังภาพประกอบ 22



ภาพประกอบ 22 การเปรียบเทียบผลของรอบเวลาการผลิต

จากผลของการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 เพื่อหารอบเวลาการผลิตที่ต่ำที่สุด ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างของปัญหาขนาดกลางที่มีจำนวน 21-58 งาน จำนวน 50 ปัญหา และปัญหาขนาดใหญ่ที่มีจำนวนงาน 70-297 งาน จำนวน 51 ปัญหา แสดงดังภาพประกอบ 22 และเมื่อนำผลของการจัดสมดุลสายการประกอบไปเปรียบเทียบผลกับ Rules-based heuristic ที่แบ่งการจัดสมดุลสายการประกอบออกเป็น 2 วิธี คือ วิธี Two Rules และวิธี Three Rules เมื่อเปรียบเทียบผลของรอบเวลาการผลิตที่ต่ำที่สุดกับวิธี Two Rules พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าและเท่ากันในทุกปัญหากับวิธี Two Rules จำนวน 39 ปัญหา (ร้อยละ 78) และจำนวน 11 ปัญหา (ร้อยละ 22) สำหรับปัญหาขนาดกลาง (21-58 งาน) และจำนวน 38 ปัญหา (ร้อยละ 74.50) และจำนวน 13 ปัญหา (ร้อยละ 25.50) สำหรับปัญหาขนาดใหญ่ (70-297 งาน) และเมื่อเปรียบเทียบกับวิธี Three Rules พบว่าในปัญหาขนาดกลาง (21-58 งาน) วิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าและเทียบเท่ากับวิธี Three Rules จำนวน 14 ปัญหา (ร้อยละ 28) และจำนวน 34 ปัญหา (ร้อยละ 68) แต่มีจำนวน 2 ปัญหา (ร้อยละ 4) ที่วิธีวิวัฒนาการโดยใช้ผลต่างยังไม่สามารถหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Three Rules ส่วนในปัญหาขนาดใหญ่ (70-297 งาน) วิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าและเทียบเท่าจำนวน 3 ปัญหา (ร้อยละ 5.88) และจำนวน 21 ปัญหา (ร้อยละ 41.17) แต่มีจำนวน 27 (ร้อยละ 52.95) ปัญหาที่วิธีวิวัฒนาการโดยใช้ผลต่างยังไม่สามารถหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Three Rules

เมื่อวิเคราะห์ผลโดยรวมของการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 ด้วยวิธีวิวัฒนาการโดยผลต่างในการทดสอบกับวิธี Two Rules และวิธี Three Rules ในปัญหาขนาดกลาง (21-58 งาน) ซึ่งรวมเป็น 100 ปัญหา พบว่าวิธีวิวัฒนาการโดยผลต่างสามารถจัดสมดุลสายการประกอบโดยใช้เวลาเฉลี่ยในการค้นหาคำตอบอยู่ระหว่าง 0.02-0.40 วินาที/ปัญหา (เฉลี่ย 0.10 วินาที/ปัญหา) ที่ทำให้ได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules และวิธี Three Rules จำนวน 53 ปัญหา (ร้อยละ 53) ได้รอบเวลาการผลิตที่เท่ากันจำนวน 45 ปัญหา (ร้อยละ 45) และได้รอบเวลาการผลิตที่แย่กว่าจำนวน 2 ปัญหา (ร้อยละ 2) ส่วนในปัญหาขนาดใหญ่ (70-297 งาน) ซึ่งรวมเป็น 102 ปัญหา พบว่าวิธีวิวัฒนาการโดยผลต่างสามารถจัดสมดุลสายการประกอบที่ใช้เวลาเฉลี่ยในการค้นหาคำตอบอยู่ระหว่าง 0.39-40 วินาที/ปัญหา (เฉลี่ย 10.33 วินาที/ปัญหา) ที่ทำให้ได้รอบเวลาการผลิตที่ดีกว่าวิธี Two Rules และวิธี Three Rules จำนวน 41 ปัญหา (ร้อยละ 40.20) ได้รอบเวลาการผลิตที่เท่ากันจำนวน 34 ปัญหา (ร้อยละ 33.33) และได้รอบเวลาการผลิตที่แย่กว่าจำนวน 27 ปัญหา (ร้อยละ 26.47)

จากผลของการจัดสมดุลสายการประกอบด้วยวิธีวิวัฒนาการโดยใช้ผลต่างยังไม่สามารถหารอบเวลาการผลิตที่ต่ำที่สุดได้ดีกว่าหรือเทียบเท่าวิธี Two Rules (2 ปัญหา) และวิธี Three Rules (27 ปัญหา) จำนวน 29 ปัญหา โดยเมื่อทำการวิเคราะห์พบว่าส่วนมาเป็นปัญหาขนาดใหญ่ (70-297 งาน) ที่เมื่อขนาดปัญหาใหญ่ขึ้นจำนวนงานและเงื่อนไขของลำดับงานก็จะเพิ่มมากขึ้น ส่งผลให้กระบวนการของวิธีวิวัฒนาการโดยใช้ผลต่างในกระบวนการสร้างคำตอบเริ่มต้น (Initial population) การปรับเปลี่ยนค่าในพิกัด (Mutation) และการแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (Recombination) ยังไม่สามารถหาคำตอบที่ดีที่สุดที่ได้ดีกว่าหรือเทียบเท่าวิธี Three Rules ในการทำวิจัยในอนาคตจึงควรเพิ่มการค้นหาเฉพาะที่ (Local Search) หรือฮิวริสติก (Heuristic) เข้ามาปรับปรุงคำตอบที่ได้จากกระบวนการของวิธีวิวัฒนาการโดยใช้ผลต่างเพื่อทำให้ วิธีวิวัฒนาการโดยใช้ผลต่างสามารถค้นหาคำตอบในปัญหาขนาดใหญ่ (70-297 งาน) ได้อย่างมีประสิทธิภาพมากขึ้น

5.2 ปัญหาและอุปสรรคของงานวิจัย

5.2.1 เนื่องจากปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) ยังมีผู้ทำวิจัยยังไม่มาก ทำให้มีข้อมูลค่อนข้างน้อยที่ใช้ในการเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างกับวิธีฮิวริสติก (Heuristic) หรือเมตาฮิวริสติก (Metaheuristic) จากงานวิจัยอื่นๆ

5.2.2 เนื่องจากยังไม่มีผู้วิจัยท่านใดนำวิธีวิวัฒนาการโดยใช้ผลต่างมาใช้แก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (UALBP-2) ทำให้ใช้เวลาในการคิดอัลกอรึทึมและเขียนโปรแกรมค่อนข้างนาน

5.3 ข้อเสนอแนะและแนวทางศึกษาการวิจัยในอนาคต

5.3.1 ควรมีการเพิ่มวิธีการค้นหาเฉพาะที่ (Local search), ฮิวริสติก (Heuristic) หรือเมตาฮิวริสติก (Metaheuristic) เข้ามาช่วยปรับปรุงคำตอบที่ได้จากกระบวนการของวิธีวิวัฒนาการโดยใช้ผลต่างเพื่อให้สามารถแก้ไขในปัญหาขนาดใหญ่ได้มีประสิทธิภาพมากขึ้น

5.3.2 ควรมีการนำวิธีวิวัฒนาการโดยใช้ผลต่างไปใช้ในการแก้ไขโจทย์ปัญหาอื่นๆ เช่น การผลิตแบบสินค้าแบบผสม (Mixed model line), การผลิตแบบสินค้าแบบหลายชนิด (Multi model line) และการผลิตแบบสองด้าน (Parallel line) เป็นต้น

5.3.3 ควรมีการพัฒนาหลักการเมตาฮิวริสติกอื่นๆมาใช้ร่วมกับวิธีวิวัฒนาการโดยใช้ผลต่างเพื่อแก้ปัญหาการจัดสมดุลสายการประกอบในรูปแบบต่างๆ

พหูน ปณ ฑิโต ชีเว

บรรณานุกรม

[1] N. Kriengkorakot and N. Pianthong, “The U-line Assembly Line Balancing Problem,” *KKU Eng. J.*, vol. 34, no. May – June, pp. 267–274, 2007.

[2] G. Jirasirilerd, R. Pitakaso, K. Sethanan, S. Kaewman, W. Sirirak, and M. Kosacka-Olejnik, “Simple assembly line balancing problem type 2 by variable neighborhood strategy adaptive search: A case study garment industry,” *J. Open Innov. Technol. Mark. Complex.*, vol. 6, no. 1, pp. 1–22, 2020, doi: 10.3390/JOITMC6010021.

[3] R. Klein and A. Scholl, “Maximizing the production rate in simple assembly line balancing - A branch and bound procedure,” *Eur. J. Oper. Res.*, vol. 91, no. 2, pp. 367–385, 1996, doi: 10.1016/0377-2217(95)00047-X.

[4] S. B. Liu, H. L. Ong, and H. C. Huang, “Two bi-directional heuristics for the assembly line type II problem,” *Int. J. Adv. Manuf. Technol.*, vol. 22, no. 9, pp. 656–661, 2003, doi: 10.1007/s00170-002-1504-y.

[5] S. B. Liu, H. L. Ong, and H. C. Huang, “A bidirectional heuristic for stochastic assembly line balancing type II problem,” *Int. J. Adv. Manuf. Technol.*, vol. 25, no. 1, pp. 71–77, 2005, doi: 10.1007/s00170-003-1833-5.

[6] S. A. Seyed-Alagheband, S. M. T. F. Ghomi, and M. Zandieh, “A simulated annealing algorithm for balancing the assembly line type II problem with sequence-dependent setup times between tasks,” *Int. J. Prod. Res.*, vol. 49, no. 3, pp. 805–825, 2011, doi: 10.1080/00207540903471486.

[7] A. C. Nearchou, *A differential evolution algorithm for simple assembly line balancing*, vol. 16, no. 1. IFAC, 2005.

[8] 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, doi: 10.1080/00207540701247906.

[9] M. Li, Q. Tang, Q. Zheng, X. Xia, and C. A. Floudas, “Rules-based heuristic approach for the U-shaped assembly line balancing problem,” *Appl. Math. Model.*, vol. 48, pp. 423–439, 2017, doi: 10.1016/j.apm.2016.12.031.

[10] D. Karabo̍a and S. Ökdem, “A simple and global optimization algorithm for

engineering problems: Differential evolution algorithm," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 12, no. 1, pp. 53–60, 2004.

[11] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces," *J. ofGlobal Optim.*, vol. 11, pp. 341–359, 1997, doi: 10.1071/AP09004.

[12] S. Kaewman, T. Srivarapongse, C. Theeraviriya, and G. Jirasirilerd, "Differential Evolution Algorithm for Multilevel Assignment Problem: A Case Study in Chicken Transportation," *Math. Comput. Appl.*, vol. 23, no. 4, pp. 1–19, 2018, doi: 10.3390/mca23040055.

[13] U. Ketsripongsa, R. Pitakaso, K. Sethanan, and T. Srivarapongse, "An Improved Differential Evolution Algorithm for Crop Planning in the Northeastern Region of Thailand," *Math. Comput. Appl.*, vol. 23, no. 3, pp. 1–19, 2018, doi: 10.3390/mca23030040.

[14] R. Kamphukaew, K. Sethanan, T. Jamrus, and H. K. Wang, "Differential evolution algorithms with local search for the multi-products capacitated vehicle routing problem with time windows: A case study of the ice industry," *Eng. Appl. Sci. Res.*, vol. 45, no. 4, pp. 273–281, 2018, doi: 10.14456/easr.2018.37.

[15] R. Akararungruangkul and S. Kaewman, "Modified Differential Evolution Algorithm Solving the Special Case of Location Routing Problem," *Math. Comput. Appl.*, vol. 23, no. 3, pp. 1–16, 2018, doi: 10.3390/mca23030034.

[16] R. Pitakaso and K. Sethanan, "Modified differential evolution algorithm for simple assembly line balancing with a limit on the number of machine types," *Eng. Optim.*, vol. 48, no. 2, pp. 253–271, 2015, doi: 10.1080/0305215X.2015.1005082.

[17] A. C. Nearchou and S. L. Omirou, "Assembly Line Balancing Using Differential Evolution Models," *Cybern. Syst.*, vol. 48, no. 5, pp. 436–458, 2017, doi: 10.1080/01969722.2017.1319238.

[18] P. Sresracoo, N. Kriengkorakot, P. Kriengkorakot, and K. Chantarasamai, "U-Shaped Assembly Line Balancing by Using Differential Evolution Algorithm," *Math. Comput. Appl.*, vol. 23, no. 4, pp. 1–21, 2018, doi: 10.3390/mca23040079.

[19] A. Scholl, "Data of Assembly Line Balancing Problems," 1993. https://assembly-line-balancing.de/salbp/benchmark-data-sets-1993/ (accessed Nov. 15, 2019).

[20] C. Becker and A. Scholl, “A survey on problems and methods in generalized assembly line balancing,” *Eur. J. Oper. Res.*, vol. 168, no. 3, pp. 694–715, 2006, doi: 10.1016/j.ejor.2004.07.023.

[21] E. H. Bowman, “Assembly-Line Balancing by Linear Programming,” *Oper. Res.*, vol. 8, no. 3, pp. 385–389, 1960, doi: 10.1287/opre.8.3.385.

[22] K. Sethanan, *Metaheuristics and Appication for Industry*. Khon Kaen: Klungnana Vitthaya press, 2015.

[23] G. J. Miltenburg and J. Wijngaard, “U-line line balancing problem,” *Manage. Sci.*, vol. 40, no. 10, pp. 1378–1388, 1994, doi: 10.1287/mnsc.40.10.1378.

[24] D. Miltenburg, J. and Sparling, “Optimal solution algorithms for the U-line balancing problem,” Hamilton, 1995.

[25] D. Sparling and J. Miltenburg, “The mixed-model U-line balancing problem,” *Int. J. Prod. Res.*, vol. 36, no. 2, pp. 485–501, 1998, doi: 10.1080/002075498193859.

[26] T. L. Urban, “Note. Optimal Balancing of U-Shaped Assembly Lines,” *Manage. Sci.*, vol. 44, no. 5, pp. 738–741, 1998, doi: 10.1287/inte.7.1.57.

[27] B. J. Chanahan, “Incorporating Ergonomics Criteria into Assemble Line Balancing,” Pittsburgh, 1999.

[28] A. Scholl and R. Klein, “ULINO: Optimally balancing U-shaped JIT assembly lines,” *Int. J. Prod. Res.*, vol. 37, no. 4, pp. 721–736, 1999, doi: 10.1080/002075499191481.

[29] E. Erel, I. Sabuncuoglu, and B. A. Aksu, “Balancing of U-type assembly systems using simulated annealing,” *Int. J. Prod. Res.*, vol. 39, no. 13, pp. 3003–3015, 2001, doi: 10.1080/00207540110051905.

[30] S. Chen, “Just-In-Time U-Shaped Assembly Line Balancing,” Lehigh, 2003.

[31] U. Martinez and W. S. Duff, “Heuristic Approaches to Solve The U-Shaped Line Balancin Problem Augmented by Gennetic Algorithms,” in *Proceedings of the 2004 Systems and Information Engineering Design Symposium Matthew*, 2004, no. Consortium 2002, pp. 55–64.

[32] H. Gökçen, K. Ağpak, C. Gencer, and E. Kizilkaya, “A shortest route formulation of simple U-type assembly line balancing problem,” *Appl. Math. Model.*, vol. 29,

no. 4, pp. 373–380, 2005, doi: 10.1016/j.apm.2004.10.003.

[33] H. Gökçen and K. Ağpak, “A goal programming approach to simple U-line balancing problem,” *Eur. J. Oper. Res.*, vol. 171, no. 2, pp. 577–585, 2006, doi: 10.1016/j.ejor.2004.09.021.

[34] R. F. Deckro and S. Rangachari, “A goal approach to assembly line balancing,” *Comput. Oper. Res.*, vol. 17, no. 5, pp. 509–521, 1990, doi: 10.1016/0305-0548(90)90055-C.

[35] W. C. Chiang and T. L. Urban, “The stochastic U-line balancing problem: A heuristic procedure,” *Eur. J. Oper. Res.*, vol. 175, no. 3, pp. 1767–1781, 2006, doi: 10.1016/j.ejor.2004.10.031.

[36] N. Kriengkorakot, “Metaheuristic approach for assembly line balancing problem,” Ubonratchatani, 2007.

[37] S. D. Lapierre, A. Ruiz, and P. Soriano, “Balancing assembly lines with tabu search,” *Eur. J. Oper. Res.*, vol. 168, no. 3, pp. 826–837, 2006, doi: 10.1016/j.ejor.2004.07.031.

[38] P. Jirachai and P. Chutima, “Mixed-Model Parallel U-Shaped Assembly Line Balancing under Many-Objectives U sing a Multi-Objective Evolutionary Algorithm Based on Decomposition,” *SWU Eng. J.*, vol. 13, pp. 82–97, 2018.

[39] A. Sa-nguansin and J. Kunadilok, “Heuristic Method for Workload Variance Reduction In U-Shaped Assembly Line Balancing,” *Thai J. Oper. Res.*, vol. 2, pp. 11–21, 2014.

[40] C. L. Moodie and H. H. Young, “A Heuristic Method of Assembly Line Balancing for Assumptions of Constant or Variable Work Element Times,” *J. Ind. Eng.*, vol. 16, pp. 23–29, 1965, Accessed: Nov. 15, 2020. [Online]. Available: https://ci.nii.ac.jp/naid/10010869519/.

[41] U. Attique, A. Ghafoor, R. Ahmed, and S. Ikramullah, “Assembly line balancing type-2 using unequal multiple operators,” *Adv. Mater. Res.*, vol. 816–817, pp. 1169–1173, 2013, doi: 10.4028/www.scientific.net/AMR.816-817.1169.

[42] J. Venkatesh and D. Balaji, “Application of simple genetic algorithm to U-shaped Assembly Line Balancing problem of type II,” in *IFAC Proceedings Volumes (IFAC-*

PapersOnline), 2014, vol. 19, no. 3, pp. 6168–6173, doi: 10.3182/20140824-6-za-1003.01769.

[43] S. Choudhary and S. Agrawal, "Mixed-Model Assembly Line Balancing Problem : A Focus on Model Formulation," in *5th International & 26th All India Manufacturing Technology, Design and Research Conference (AIMTDR 2014)*, 2014, no. Aimtdr, pp. 12–15.

[44] B. Pakeeza, A. Riaz, and M. Umer, "A Heuristic Approach for Type 2 Assembly Line Balancing Problem," *Appl. Mech. Mater.*, vol. 789–790, pp. 1296–1300, 2015, doi: 10.4028/www.scientific.net/amm.789-790.1296.

[45] O. Polat, Ö. Mutlu, and E. Özgormus, "A Mathematical Model For Assembly Line Balancing Problem Type 2 Under Ergonomic Workload Constraint," *Ergon. Open J.*, vol. 11, no. 1, pp. 1–10, 2018, doi: 10.2174/1875934301811010001.

[46] A. Scholl, *Balancing and Sequencing of Assembly Lines*, 2nd ed. Physica-Verlag Heidelberg, 1995.

[47] G. Jirasirilerd and R. Pitakaso, "Application of The Differential Evolution Algorithm to Solve Assembly Line Balancing Problem Type 1: A Case Study of the Garment Manufacturing Industry," *Thai J. Oper. Res.*, vol. 2, pp. 39–50, 2013.

[48] P. Parawech, "Differential Evolution Algorithm to Solve Simple Assembly Line Balancing Problem Type 1," Ubonratchatani, 2013.

[49] P. Parawech, R. Paitakaso, and P. Mayachearw, "Solving an Assembly Line Balancing Problem by Differential Evolution: A Case Study of a Garment Factory," *Princess Naradhiwas Univ. J.*, vol. 6, no. 2, pp. 92–104, 2014.

[50] R. Pitakaso, "Differential evolution algorithm for simple assembly line balancing type 1 (SALBP-1)," *J. Ind. Prod. Eng.*, vol. 32, no. 2, pp. 104–114, 2015, doi: 10.1080/21681015.2015.1007094.

[51] I. Sabuncuoglu, E. Erel, and M. Tanyer, "Assembly line balancing using genetic algorithms," *J. Intell. Manuf.*, vol. 11, no. 3, pp. 295–310, 2000, doi: 10.1023/A:1008923410076.

[52] W. C. Chiang, "The application of a tabu search metaheuristic to the assembly line balancing problem," *Ann. Oper. Res.*, vol. 77, pp. 209–227, 1998, doi:

10.1023/a:1018925411397.

[53] O. Kilincci, “Firing sequences backward algorithm for simple assembly line balancing problem of type 1,” *Comput. Ind. Eng.*, vol. 60, no. 4, pp. 830–839, 2011, doi: 10.1016/j.cie.2011.02.001.

[54] Y. K. Kim, Y. J. Kim, and Y. Kim, “Genetic algorithms for assembly line balancing with various objectives,” *Comput. Ind. Eng.*, vol. 30, no. 3 SPEC. ISS., pp. 397–409, 1996, doi: 10.1016/0360-8352(96)00009-5.

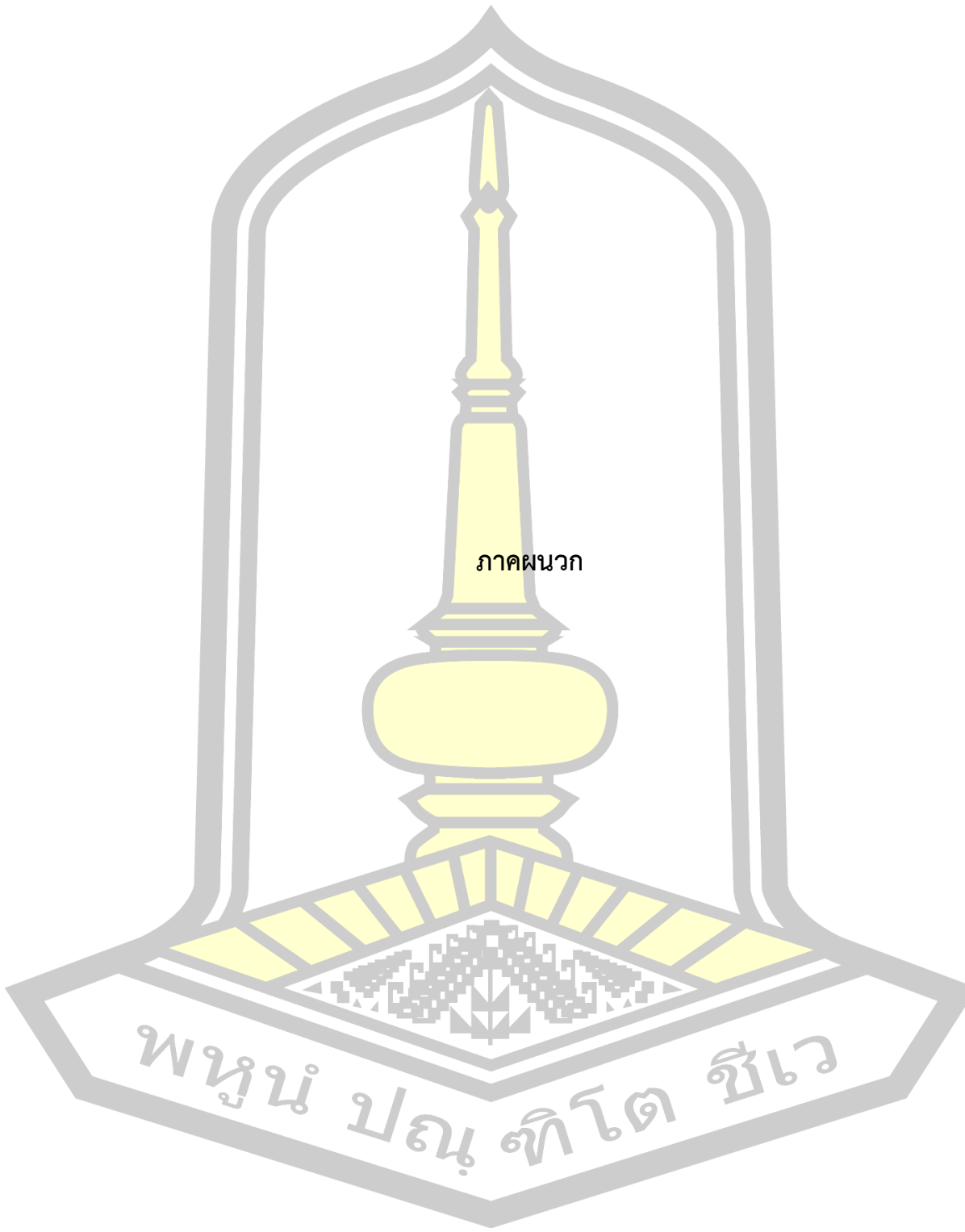
[55] F. B. Talbot, J. H. Patterson, and W. V Gehrlein, “A COMPARATIVE EVALUATION OF HEURISTIC LINE BALANCING TECHNIQUES *,” *Manage. Sci.*, vol. 32, no. 4, pp. 430–454, 1986.

[56] Z. Zeqiang, C. Wenming, C. Yue, and L. Jian, “A novel ant colony optimization algorithm for U-shaped line balancing problem,” *Proc. - 4th Int. Conf. Nat. Comput. ICNC 2008*, vol. 7, pp. 455–459, 2008, doi: 10.1109/ICNC.2008.18.

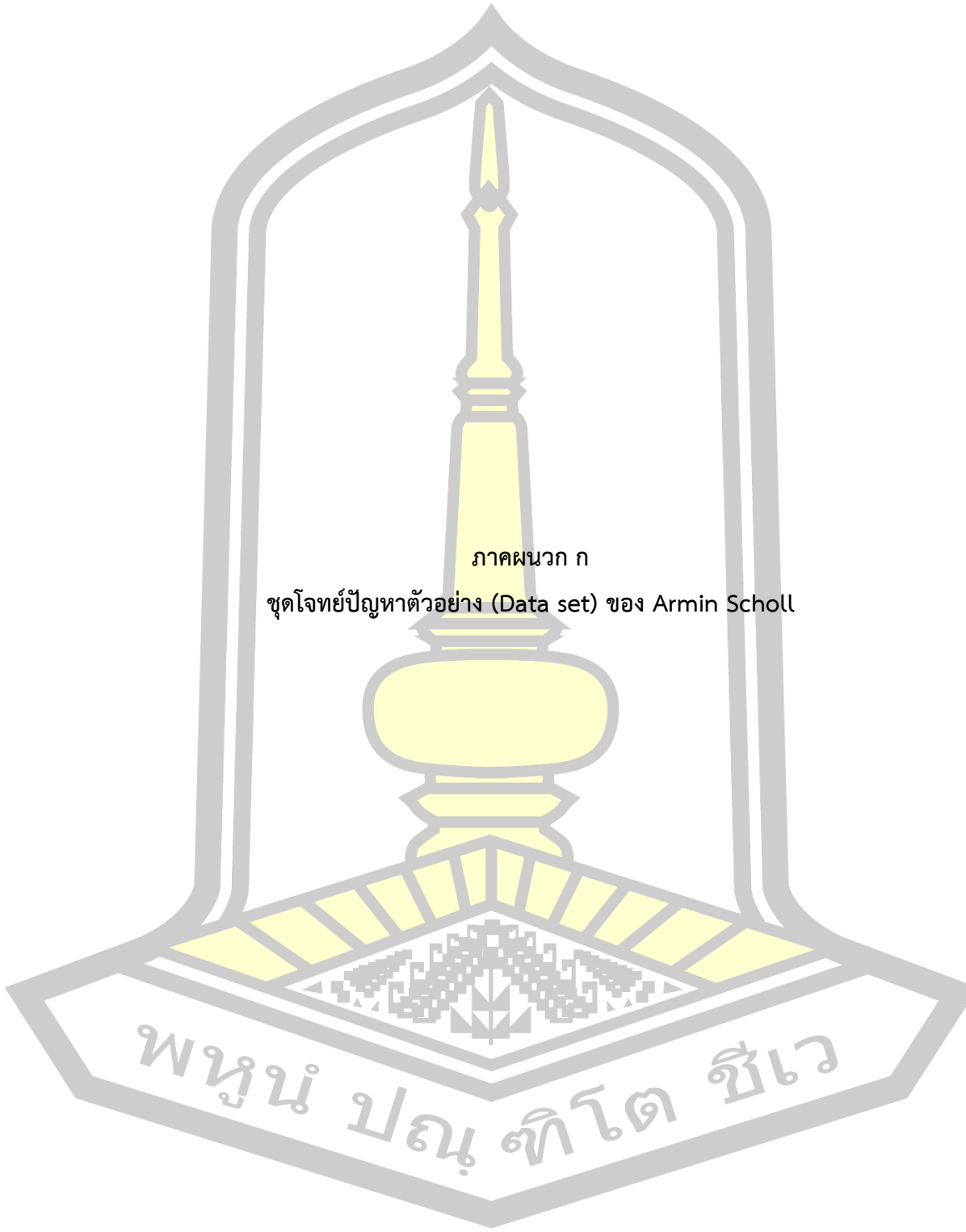
[57] A. K. Qin, V. L. Huang, and P. N. Suganthan, “Differential evolution algorithm with strategy adaptation for global numerical optimization,” *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, 2009, doi: 10.1109/TEVC.2008.927706.

[58] J. R. Jackson, “A Computing Procedure for a Line Balancing Problem,” *Manage. Sci.*, vol. 2, no. 3, pp. 261–271, 1956, doi: 10.1287/mnsc.2.3.261.

พหุน ปณ ฑิโต ชีเว

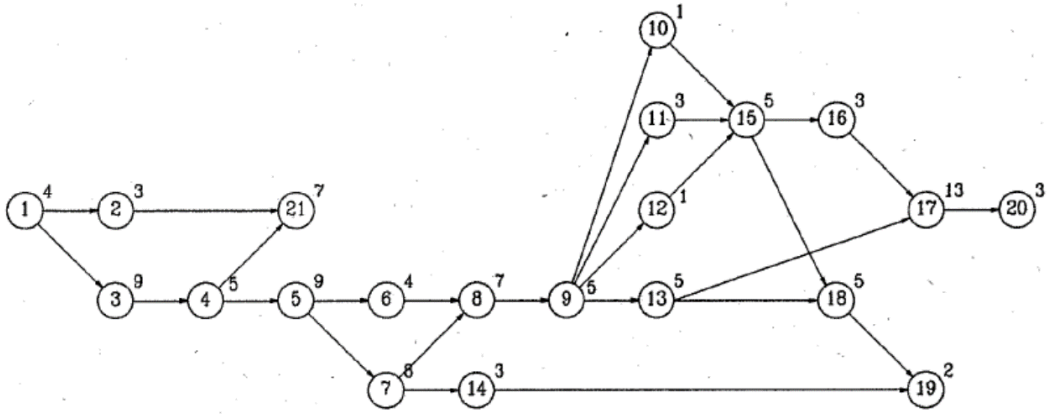


ภาคผนวก

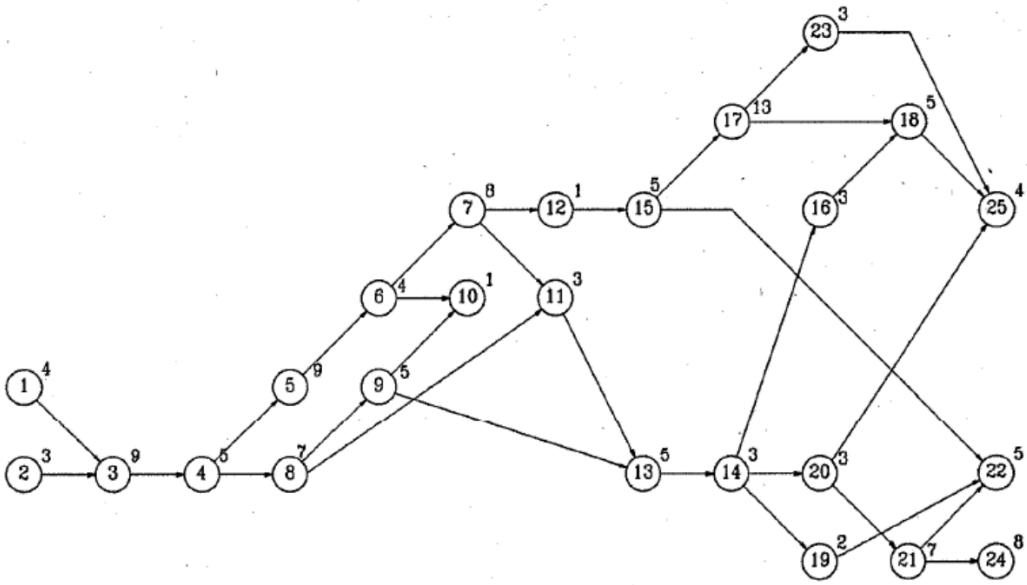


ภาคผนวก ก

ชุดโจทย์ปัญหาตัวอย่าง (Data set) ของ Armin Scholl

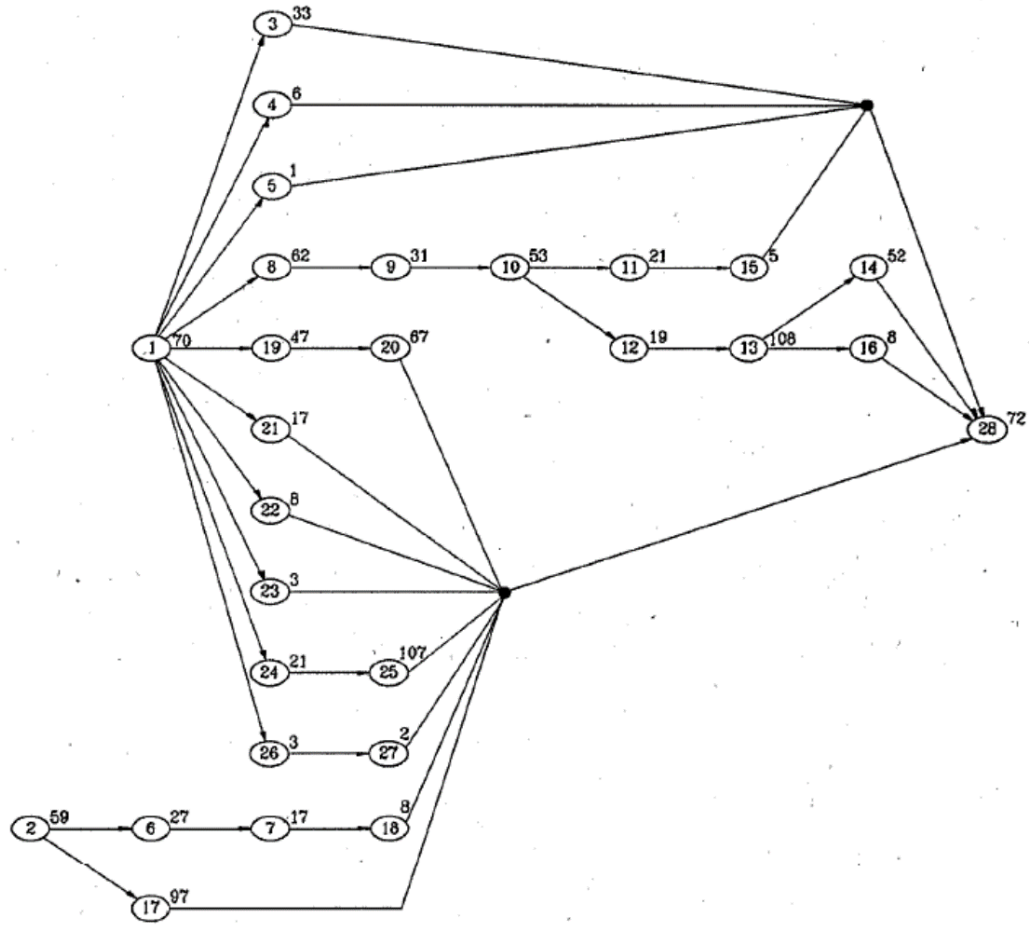


ภาพประกอบ 23 ชุดปัญหาของ Mitchell ที่มีจำนวน 21 งาน



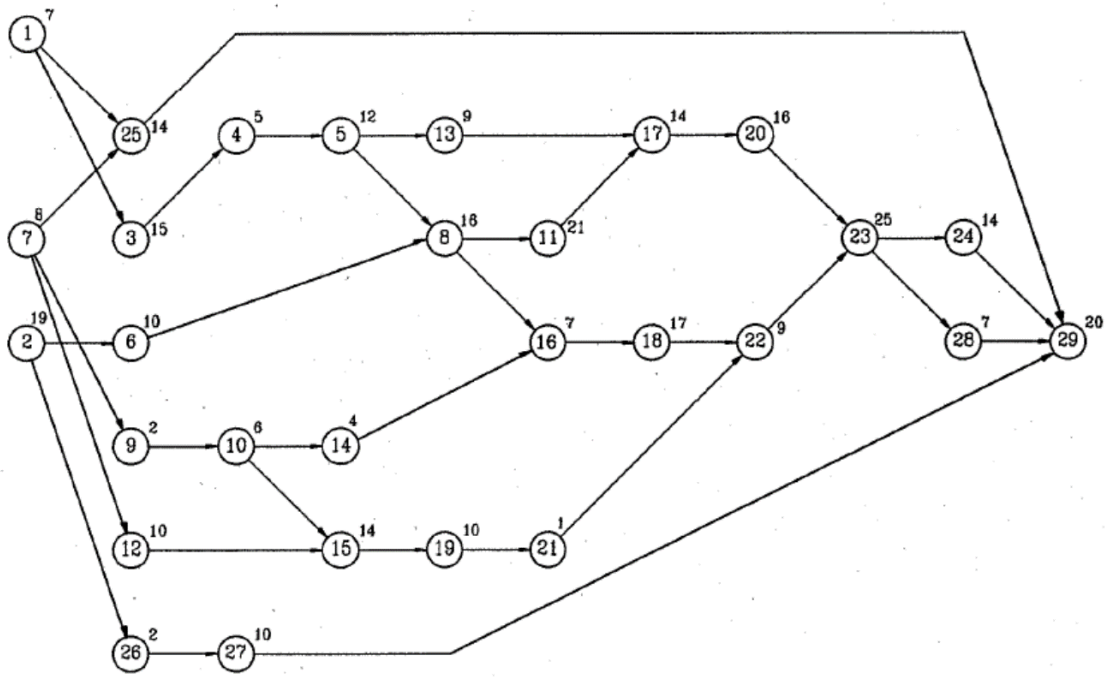
ภาพประกอบ 24 ชุดปัญหาของ Rosenberg ที่มีจำนวน 25 งาน

พหุน ปณ ฑิโต ชีเว

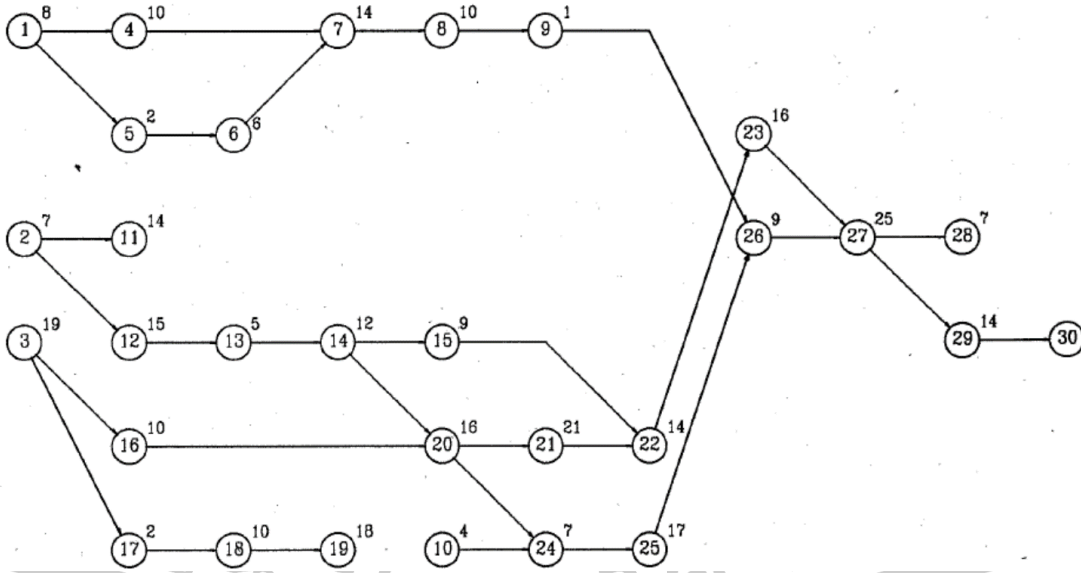


ภาพประกอบ 25 ชุดปัญหาของ Heskiaoff ที่มีจำนวน 28 งาน

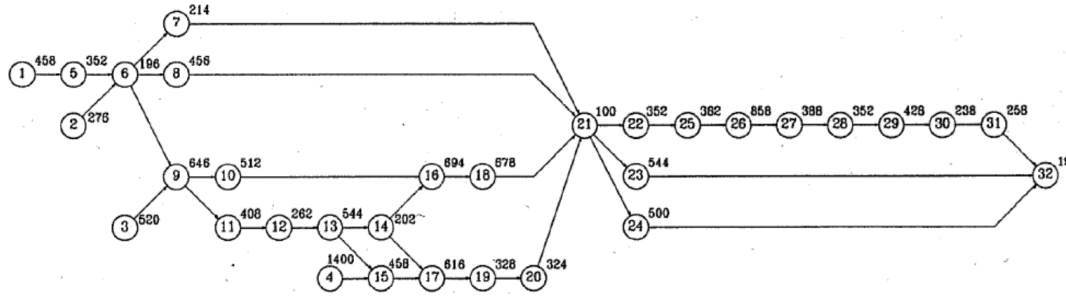
พหูน ปณ ฑิโต ชีเว



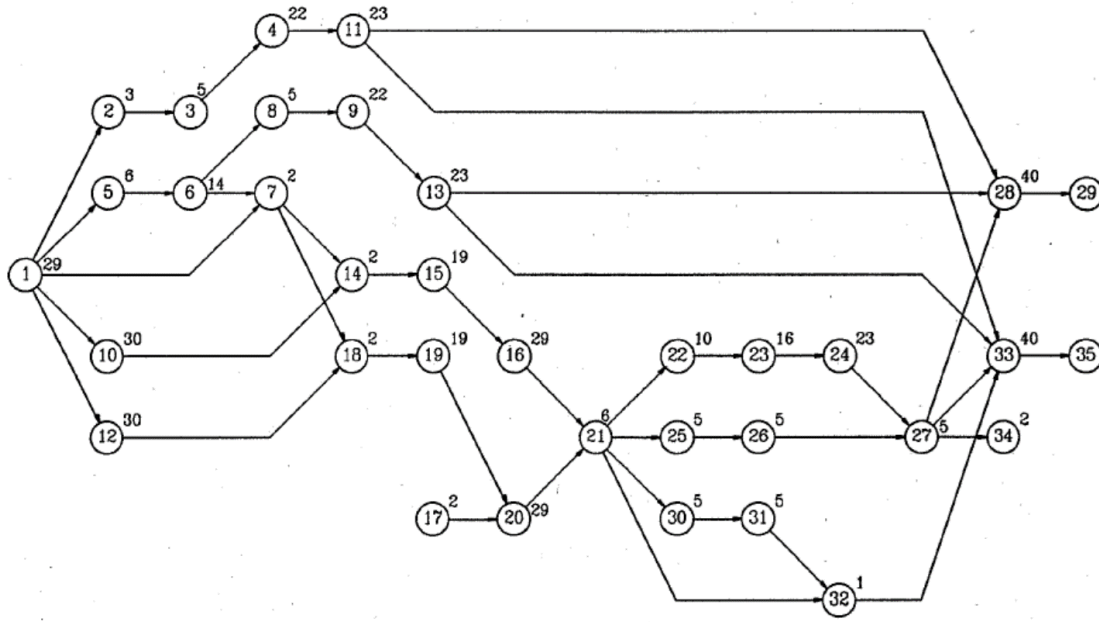
ภาพประกอบ 26 ชุดปัญหาของ Buxey ที่มีจำนวน 29 งาน



ภาพประกอบ 27 ชุดปัญหาของ Sawyer ที่มีจำนวน 30 งาน

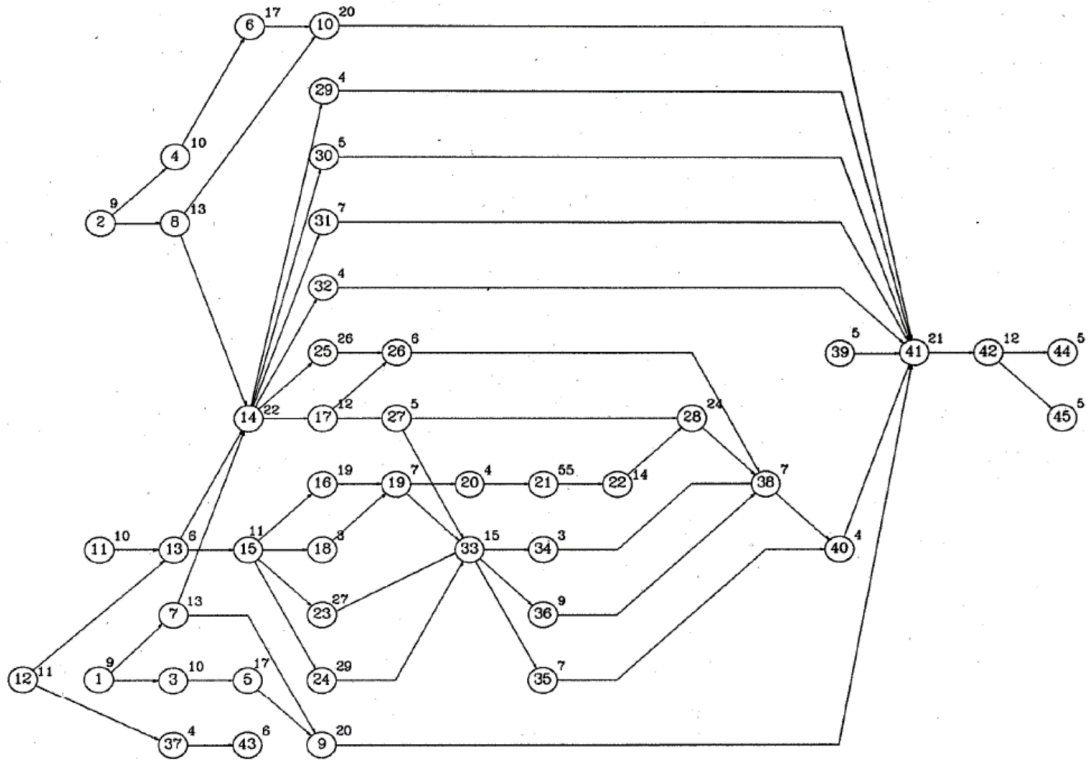


ภาพประกอบ 28 ชุดปัญหาของ Lutz1 ที่มีจำนวน 32 งาน

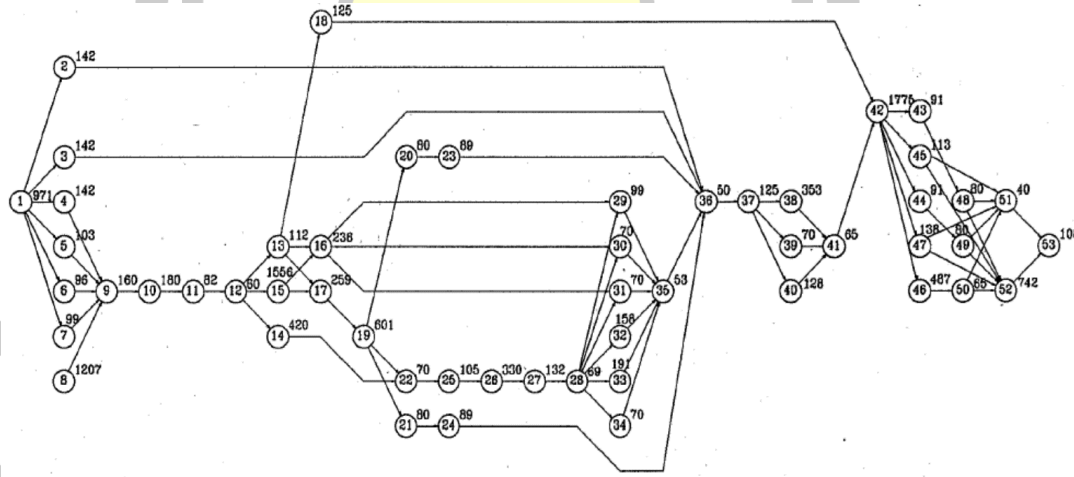


ภาพประกอบ 29 ชุดปัญหาของ Gunther ที่มีจำนวน 35 งาน

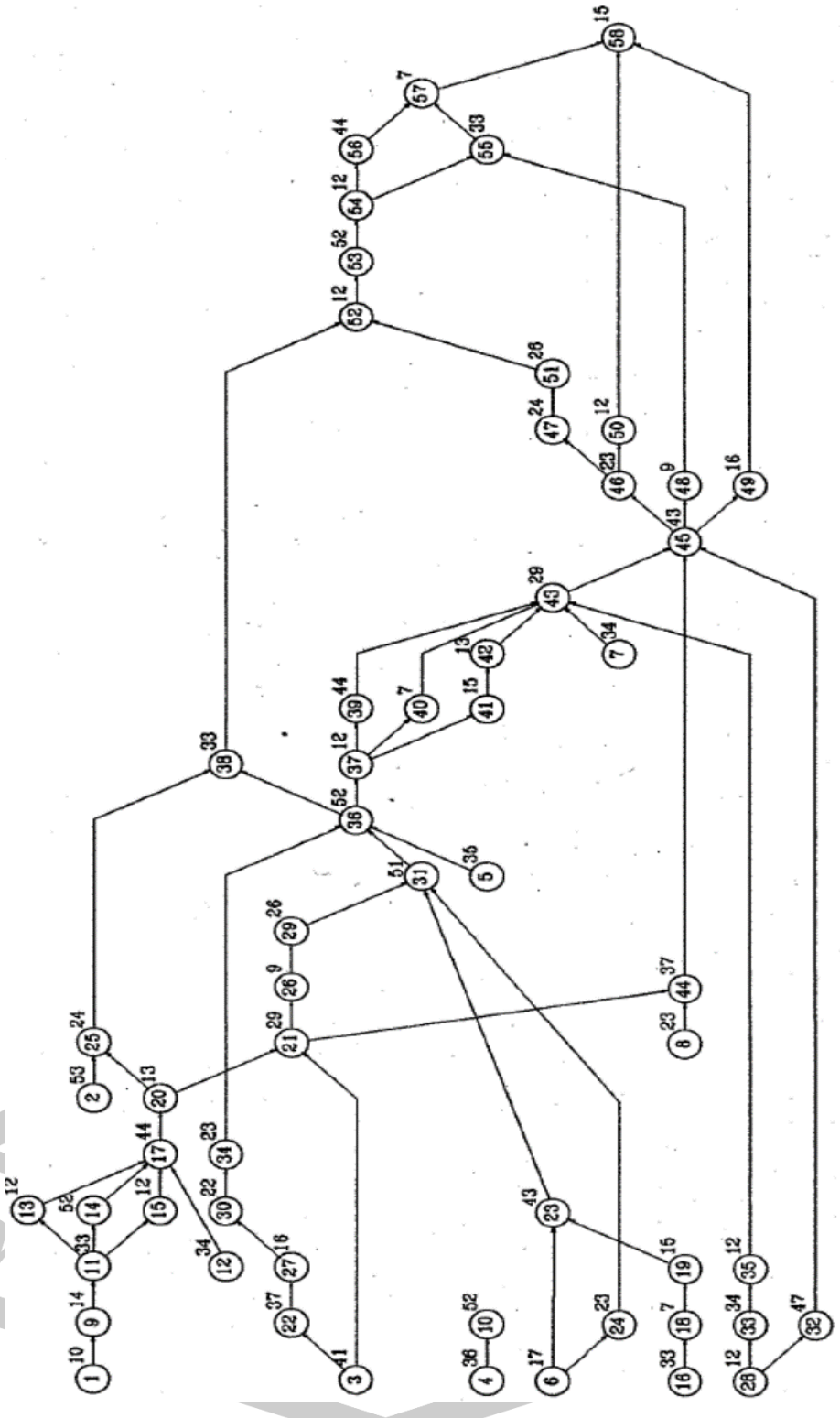
พหุน ปณ ฑิโต ชีเว



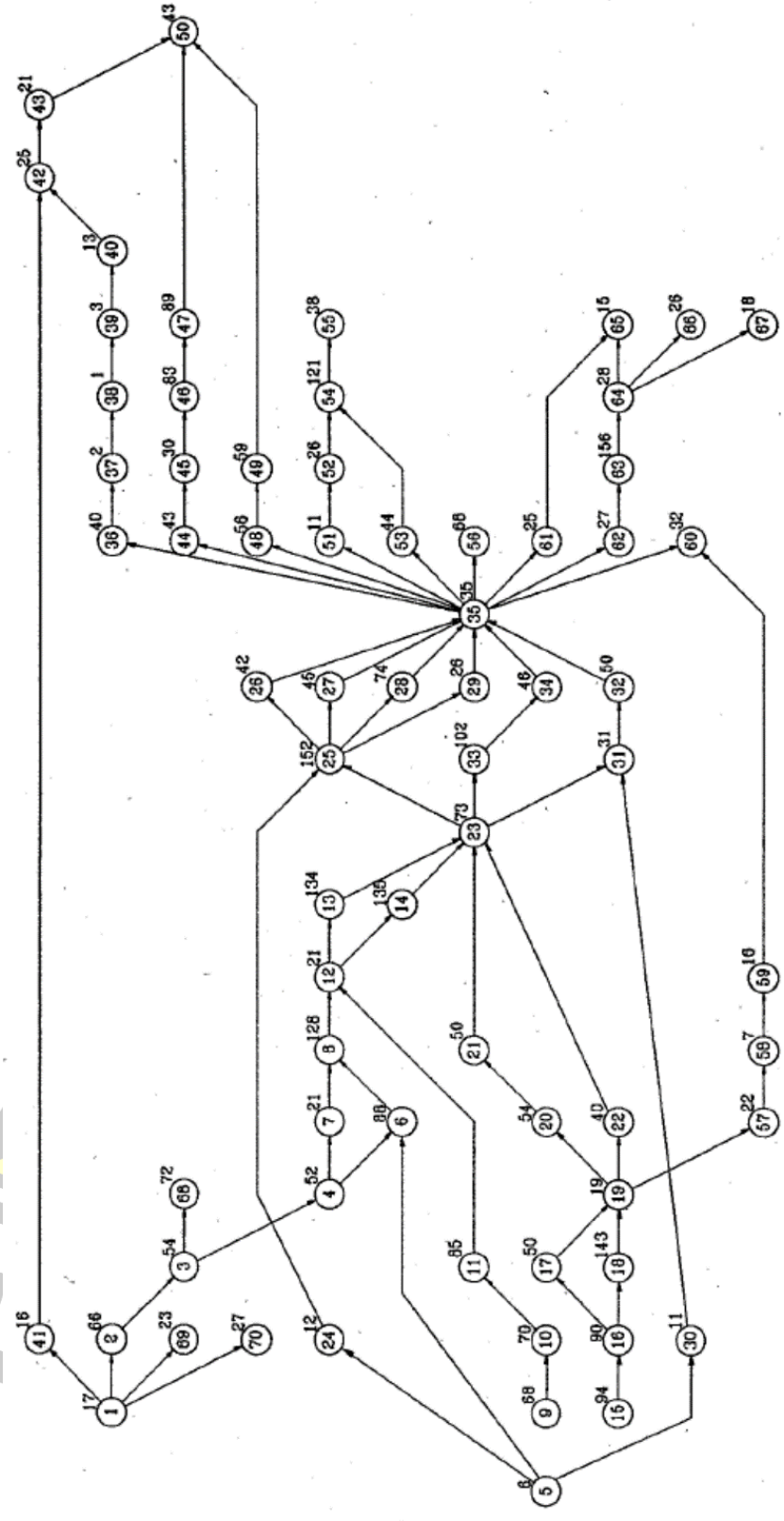
ภาพประกอบ 30 ชุดปัญหาของ Kilbridge&Wester ที่มีจำนวน 45 งาน



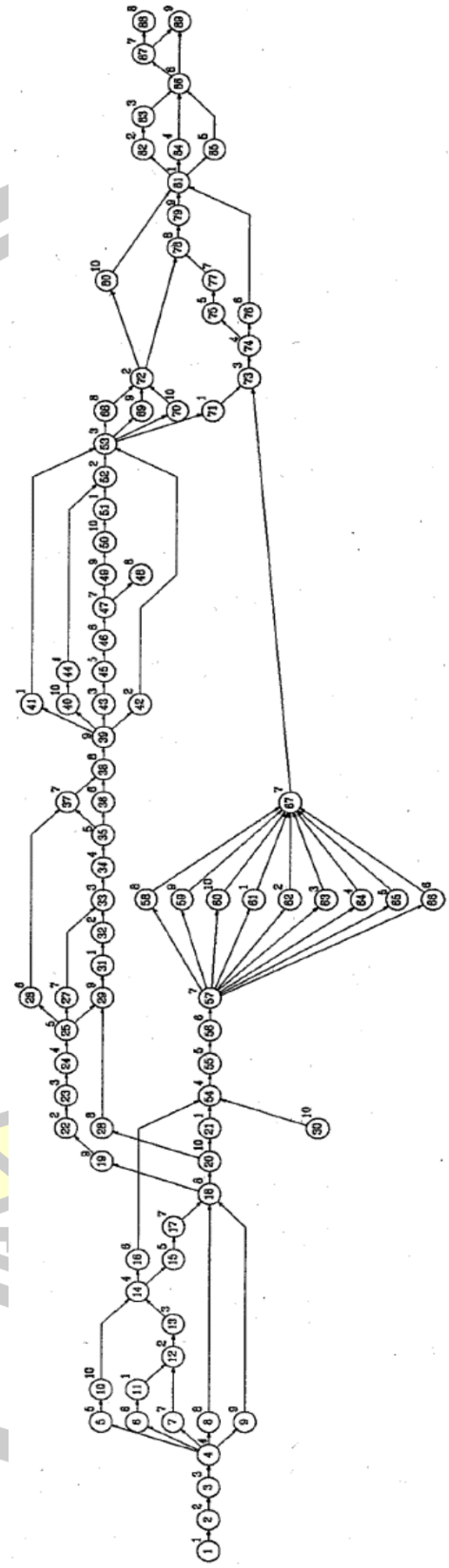
ภาพประกอบ 31 ชุดปัญหาของ Hahm ที่มีจำนวน 53 งาน



ภาพประกอบ 32 ชุดปัญหาของ Warnecke ที่มีจำนวน 58 งาน



ภาพประกอบ 33 ชุดปัญหาของ Tonge ที่มีจำนวน 70 งาน



ภาพประกอบ 34 ชุดปัญหาของ Lutz2 ที่มีจำนวน 89 งาน

Mukherjee: 94 tasks, assembly of an electronic consumer good in an Indian manufactury, Mukherjee and Basu (1964)

Task times:

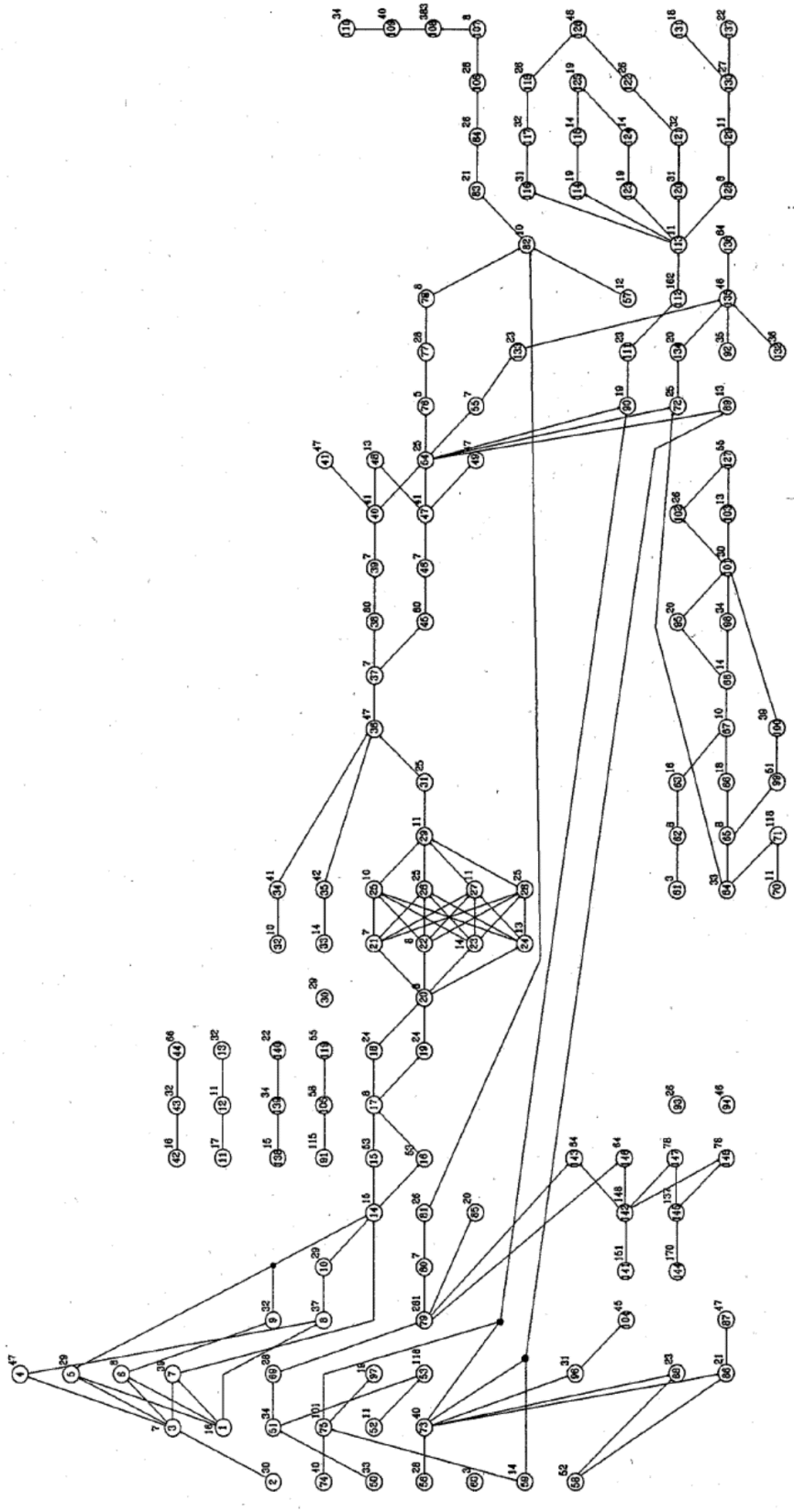
j	t_j	t_{j+1}	t_{j+2}	t_{j+3}	t_{j+4}	t_{j+5}	t_{j+6}	t_{j+7}	t_{j+8}	t_{j+9}
1	158	42	17	24	71	66	50	55	30	28
11	113	114	57	65	15	11	10	19	8	17
21	57	40	17	11	17	27	17	18	16	18
31	50	18	13	21	63	8	63	9	20	72
41	30	69	27	9	41	38	76	31	11	29
51	48	8	171	48	43	18	94	9	19	24
61	8	34	62	12	8	47	27	18	52	65
71	72	51	58	53	76	93	86	76	9	14
81	55	32	22	35	63	59	149	40	24	123
91	25	115	103	54						

Precedence constraints: pairs i, j for task i preceding task j

1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9
1,10	2,11	2,17	3,12	4,11	4,16	5,11	5,13
5,18	5,16	6,11	6,15	6,13	6,19	7,42	8,29
9,11	10,19	10,37	10,48	10,50	11,14	12,20	12,61
12,52	13,82	14,25	14,23	14,30	14,31	14,24	14,32
14,22	14,21	14,26	14,28	14,27	14,29	15,81	16,75
17,82	18,75	19,75	20,78	21,71	21,60	21,53	21,39
21,40	21,41	22,38	22,72	22,49	22,58	23,34	23,35
24,37	24,44	25,57	25,63	25,33	25,73	25,60	25,65
25,56	26,41	26,43	27,62	27,58	27,54	27,55	27,56
28,33	28,35	28,40	28,43	28,49	28,51	28,52	28,54
29,75	30,59	30,42	30,36	31,74	32,36	32,47	32,44
32,60	32,45	32,69	32,46	32,48	32,50	33,78	34,78
35,78	36,82	37,61	37,73	37,64	37,66	37,67	37,68
37,69	37,70	37,71	37,72	37,62	38,82	39,82	40,78
41,78	42,74	43,78	44,75	45,75	46,75	46,76	47,75
48,75	49,78	50,75	51,78	52,78	53,76	53,77	54,78
55,77	56,77	57,76	57,77	58,77	59,82	60,77	61,77
62,77	63,77	64,74	65,74	66,75	66,76	67,74	68,82
69,77	69,75	70,77	70,76	71,76	72,76	73,77	74,82
75,82	76,82	77,78	78,79	78,80	79,81	80,81	81,82
82,83	83,84	83,85	84,86	85,86	86,87	87,88	88,89
88,90	89,92	90,91	90,93	92,94			

พหุน ปณ ศาสตร ชีว

ภาพประกอบ 35 ชุดปัญหาของ Mukherjee ที่มีจำนวน 94 งาน



ภาพประกอบ 36 ชุดปัญหาของ Bartholdi ที่มีจำนวน 148 งาน

Scholl: 297 tasks, assembly of an engine

Task times:

j	t_j	t_{j+1}	t_{j+2}	t_{j+3}	t_{j+4}	t_{j+5}	t_{j+6}	t_{j+7}	t_{j+8}	t_{j+9}
1	270	270	130	148	190	293	348	182	490	212
11	248	248	248	248	248	268	268	268	288	248
21	268	60	268	240	240	171	490	182	170	306
31	108	248	190	240	339	288	248	455	268	270
41	180	121	270	440	249	194	162	130	388	90
51	212	246	188	270	160	79	466	240	137	184
61	110	275	149	280	119	184	140	150	190	150
71	150	284	192	347	232	140	608	80	40	130
81	110	350	140	240	240	90	54	294	203	150
91	270	155	190	78	140	241	430	90	110	9
101	430	130	289	110	160	442	159	250	190	184
111	690	72	190	190	90	889	170	155	190	130
121	390	301	54	227	142	184	741	868	230	121
131	320	126	440	127	134	150	140	110	320	250
141	232	188	250	377	90	140	90	90	70	90
151	110	150	101	377	118	290	209	150	150	79
161	150	91	59	218	351	873	130	68	126	120
171	227	198	132	121	150	100	38	70	355	284
181	122	75	160	140	520	99	182	80	514	96
191	50	272	226	194	164	96	107	108	167	98
201	82	482	72	50	130	230	50	240	190	190
211	240	74	139	339	260	132	550	420	152	12
221	90	5	128	100	120	100	320	835	740	223
231	100	390	140	304	120	403	21	246	160	1019
241	34	120	68	910	302	778	101	1310	20	278
251	81	290	100	372	72	28	90	250	144	303
261	220	58	224	211	99	44	120	70	421	231
271	214	196	280	398	72	280	356	193	140	130
281	300	456	7	170	252	210	308	308	121	52
291	426	104	1386	527	968	1047	538			

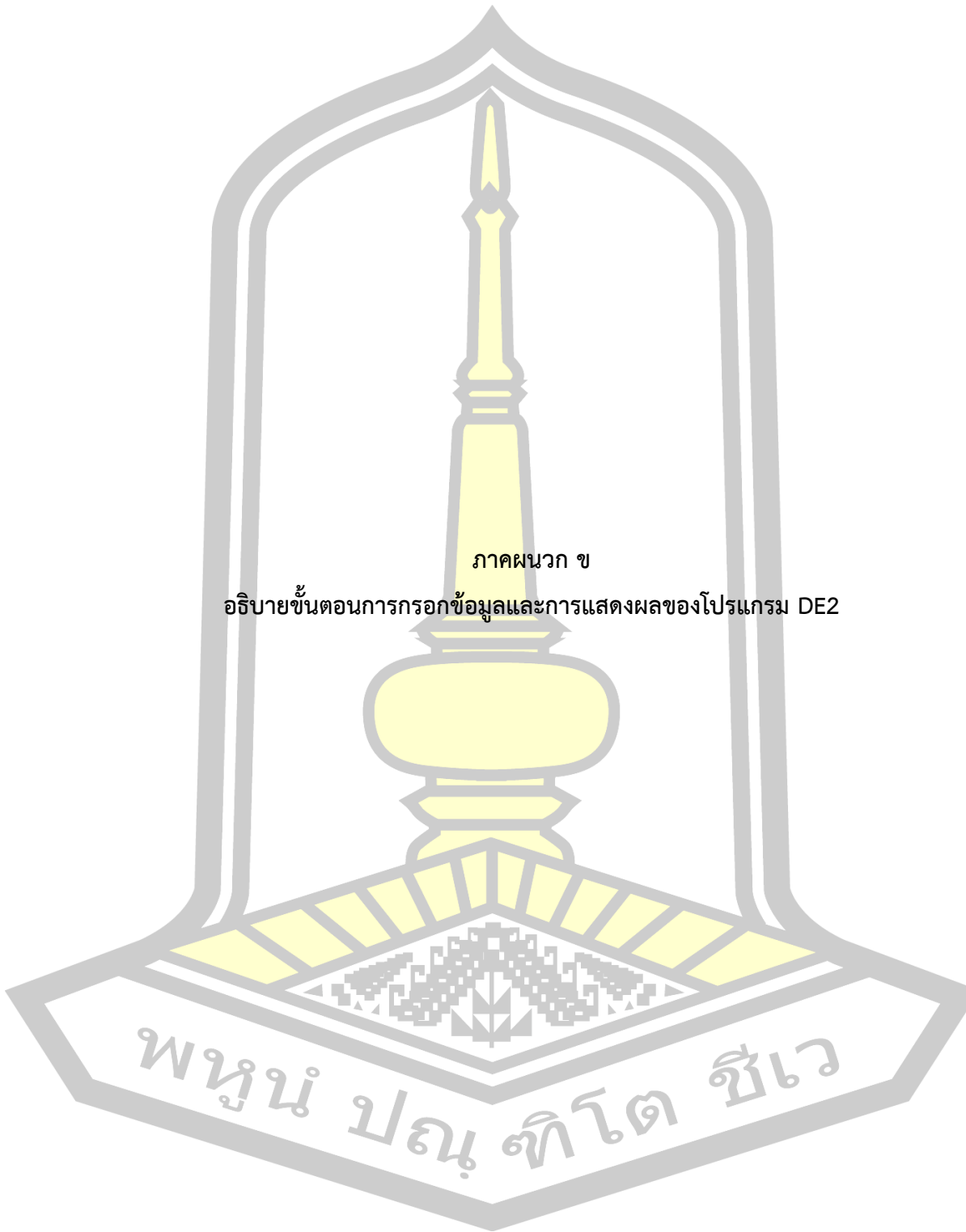
ภาพประกอบ 37 ชุดปัญหาของ Scholl ที่มีจำนวน 297 งาน

พหูน ปณฺ ฑิโต ชีเว

Precedence constraints: pairs i, j for task i preceding task j

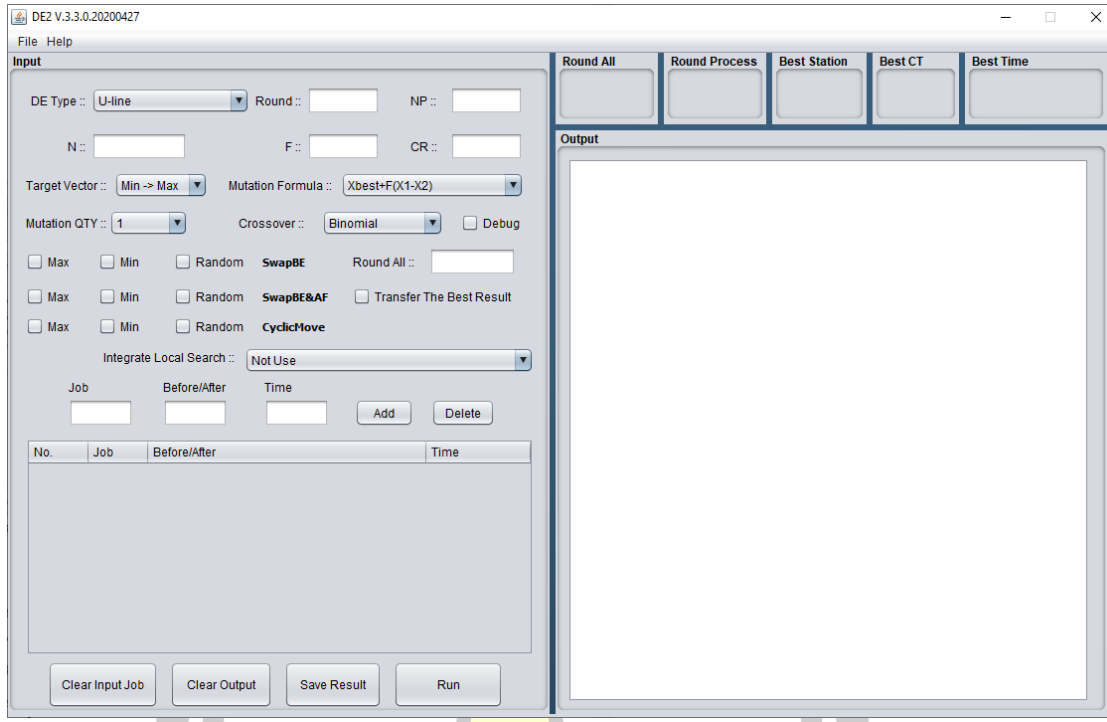
1,2	2,3	3,4	4,5	4,22	4,26	4,27	4,40
4,48	4,56	4,83	4,86	4,94	4,105	4,109	4,111
4,134	4,221	4,247	4,259	5,6	6,7	6,8	6,9
6,10	7,11	7,12	7,13	7,14	7,15	7,20	8,11
8,12	8,13	8,14	8,15	8,20	9,11	9,12	9,13
9,14	9,15	9,20	10,11	10,12	10,13	10,14	10,15
10,20	11,16	12,17	13,18	14,19	15,21	16,23	17,23
18,23	19,23	20,23	21,23	22,24	22,25	23,28	24,29
25,29	26,30	27,31	28,32	28,37	29,33	29,44	29,121
30,34	30,297	31,34	31,82	31,172	31,179	32,36	33,38
34,35	35,42	36,39	37,39	38,41	39,43	40,44	40,84
40,97	41,45	42,46	43,47	44,49	45,50	46,51	46,138
47,52	48,52	49,53	50,54	51,55	51,81	52,57	53,58
54,58	54,296	55,59	56,60	56,61	57,62	57,63	57,71
57,76	58,64	59,64	59,99	59,100	60,68	61,65	62,66
63,67	64,72	65,69	66,69	67,70	68,73	69,74	70,75
71,77	72,78	73,84	73,97	74,84	74,97	75,84	75,97
76,84	76,97	77,84	77,97	78,79	78,80	78,125	78,192
79,85	80,85	81,87	82,88	82,89	83,90	84,91	85,92
86,93	87,99	87,100	88,99	88,100	89,99	89,100	90,95
91,96	92,98	93,98	94,101	95,101	96,101	97,101	98,102
99,103	100,104	101,106	102,107	103,108	104,108	105,110	106,112
107,113	108,114	108,115	108,292	109,119	109,120	110,119	110,120
110,162	111,116	112,117	113,118	114,119	115,120	116,122	117,123
117,124	117,257	118,126	119,127	120,127	120,150	121,128	122,129
123,130	123,145	123,146	123,147	123,148	123,149	124,130	125,130
126,130	127,130	127,157	128,130	129,130	129,141	130,131	130,144
131,132	131,133	132,135	133,135	133,170	134,136	135,137	136,139
137,140	138,140	138,191	139,142	139,253	140,143	140,200	141,151
142,152	143,153	143,169	144,154	145,155	146,156	147,158	148,159
149,160	150,161	151,163	152,164	153,165	154,166	155,166	156,166
157,166	158,166	159,166	160,166	161,166	162,167	163,166	164,167
165,168	165,176	166,170	167,171	168,173	169,174	170,174	171,174
172,175	173,177	174,178	174,287	174,288	175,180	176,181	177,181
178,181	179,181	180,181	180,252	181,182	181,183	181,184	181,185
181,186	181,187	181,188	181,189	181,196	181,197	181,295	182,190
183,193	184,194	185,195	186,195	187,195	188,195	189,195	190,195
191,200	192,201	193,198	194,199	195,199	195,203	195,205	195,227
195,229	196,202	197,202	198,202	199,202	200,202	201,202	202,204
202,251	203,206	203,208	204,207	204,250	205,207	206,209	207,210
207,212	208,210	209,210	209,211	210,213	211,213	212,214	213,214
214,215	214,234	215,216	216,217	217,218	218,219	219,220	220,222
221,223	222,224	223,225	224,226	225,227	226,228	227,230	228,231
229,235	229,236	230,232	230,271	230,289	231,233	232,236	233,237
234,238	234,256	235,237	236,239	237,240	238,240	238,285	239,240
239,279	240,241	240,243	241,242	242,244	243,245	243,246	244,245
244,246	244,255	245,248	246,248	247,278	248,249	249,254	249,284
250,256	251,256	252,258	253,260	254,261	255,261	255,262	256,263
257,264	258,265	259,266	260,267	261,268	261,269	262,269	263,270
264,271	265,272	266,271	267,273	268,274	269,274	270,274	271,274
272,275	273,276	274,277	274,278	274,282	275,280	276,281	277,283
278,283	279,286	280,290	281,291	282,293	283,293	284,294	285,294
286,294	287,293	288,293	289,294	290,293	291,293	292,293	

ภาพประกอบ 38 ชุดปัญหาของ Scholl ที่มีจำนวน 297 งาน (ต่อ)



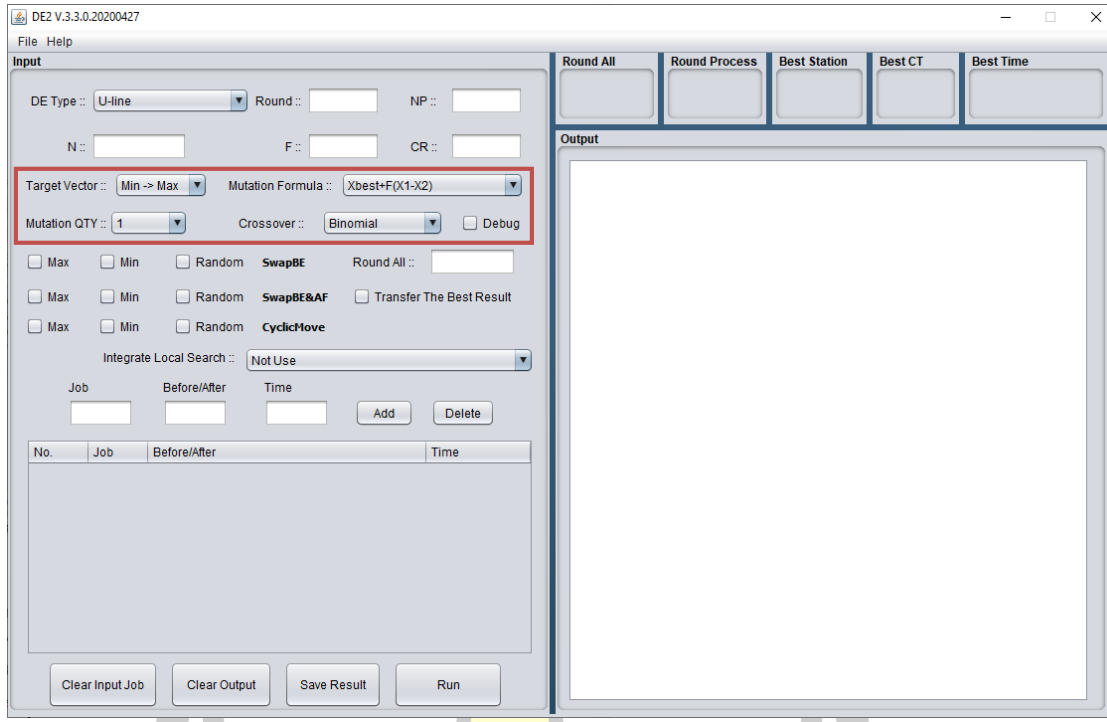
ภาคผนวก ข

อธิบายขั้นตอนการกรอกข้อมูลและการแสดงผลของโปรแกรม DE2



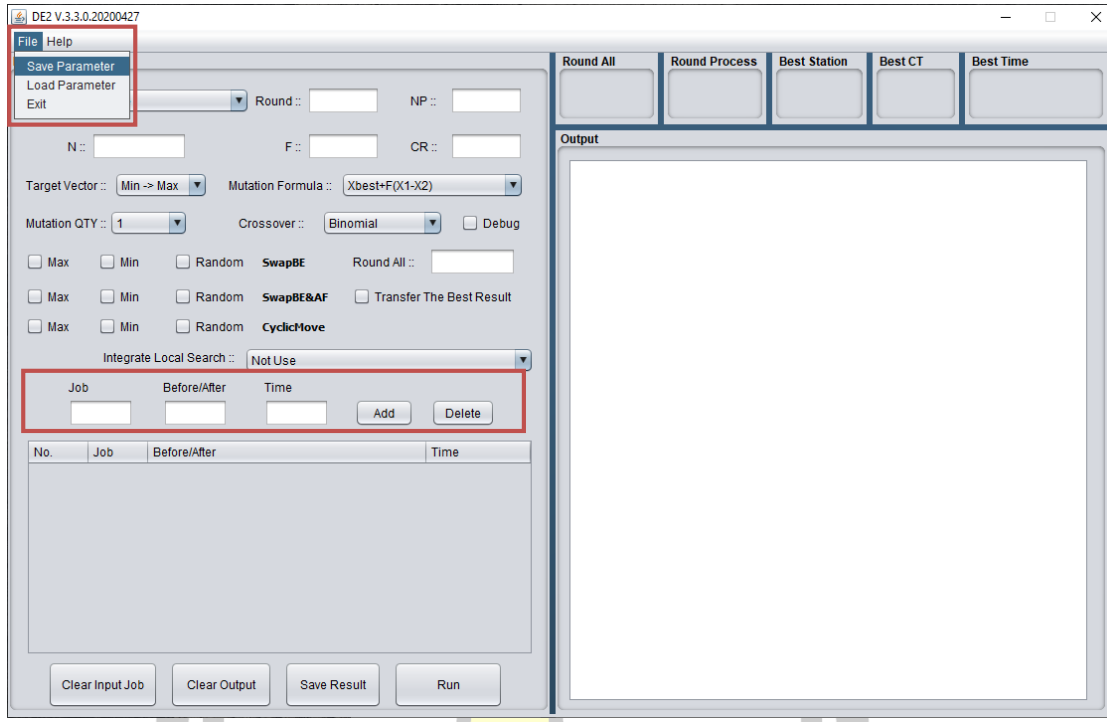
ภาพประกอบ 39 ขั้นตอนการกรอกพารามิเตอร์ที่เกี่ยวข้องกับการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่าง

จากภาพประกอบ 39 แสดงขั้นตอนการกรอกพารามิเตอร์ 5 ช่อง ที่เกี่ยวข้องกับการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่าง 1) ช่อง Round >>> เพื่อกรอกจำนวนรอบการรัน 2) ช่อง NP >>> เพื่อกรอกจำนวนขั้นงาน 3) ช่อง N >>> เพื่อกรอกรอบจำนวนสถานีงาน 4) ช่อง F >>> เพื่อกรอกแฟคเตอร์ในการขยายผลต่าง (Scaling factor) 5) ช่อง CR >>> เพื่อกรอกอัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate)



ภาพประกอบ 40 ขั้นตอนการกรอกสูตรการคำนวณวิธีวัฒนาการโดยใช้ผลต่าง

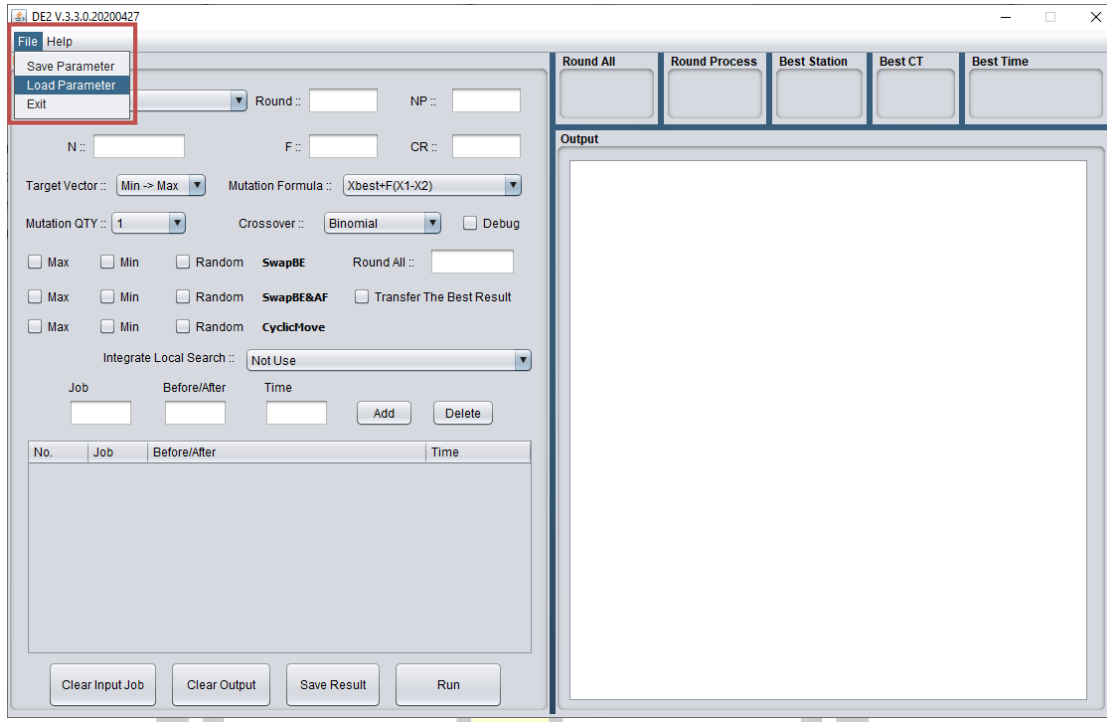
จากภาพประกอบ 40 แสดงขั้นตอนการกรอกสูตรการคำนวณวิธีวัฒนาการโดยใช้ผลต่าง โดยทำการกดตัวเลือกจากซ้ายไปขวาบนลงล่าง ช่อง Target Vector >>> เลือกจากค่าที่ Random จากน้อยไปมาก (Min-->Max) หรือ มากไปน้อย (Max-->Min) ช่อง Mutation Formula >>> ทำการเลือกวิธีแบบต่าง ๆ ในส่วนขั้นตอน Mutation โดยมีสูตรให้เลือกแต่ละวิธี, ช่อง Mutation QTY ทำการเลือกตามจำนวนรอบกด (1 รอบ และ NP) ช่อง Crossover ทำการเลือกวิธีแบบต่าง ๆ ในส่วนขั้นตอน Crossover โดยจะมีสูตรให้เลือกแต่ละวิธี



ภาพประกอบ 41 ขั้นตอนการกรอกข้อมูลของชุดปัญหาการจัดสมดุลสายการประกอบ

จากภาพประกอบ 41 แสดงขั้นตอนการกรอกมูลในแต่ละชุดปัญหาโดยทำการกรอกข้อมูลจากซ้ายไปขวาบนลงล่าง ช่อง Job >>> กรอกงาน (task) ช่อง Before/After >>> กรอกลำดับงานก่อนหลังของงาน Time >>> กรอกเวลาของงาน หลังจากเสร็จในหนึ่งงานแล้วให้กดปุ่ม Add เพื่อบันทึกเข้าไปยังโปรแกรม โดยจะทำการกรอกแบบนี้ไปจนครบทุกงานในชุดปัญหาแล้วเมื่อครบจะทำการ Save Parameter ไว้เพื่อใช้ในการทดสอบต่อไป

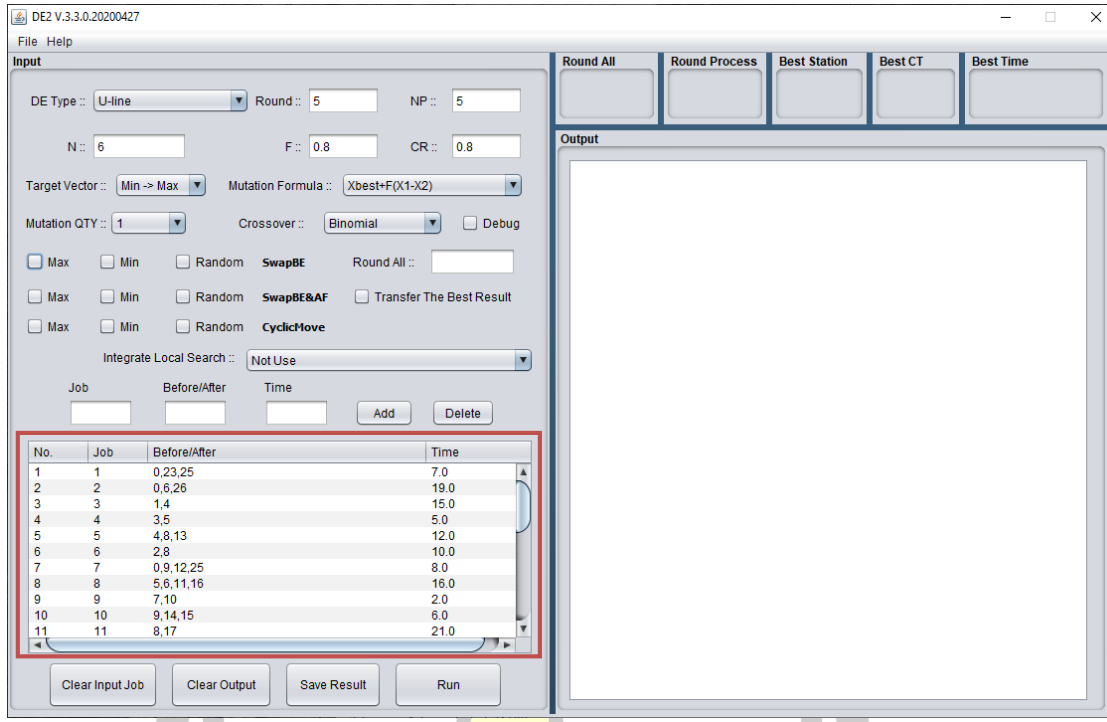
พหุน ปณ ฑิโต ชีเว



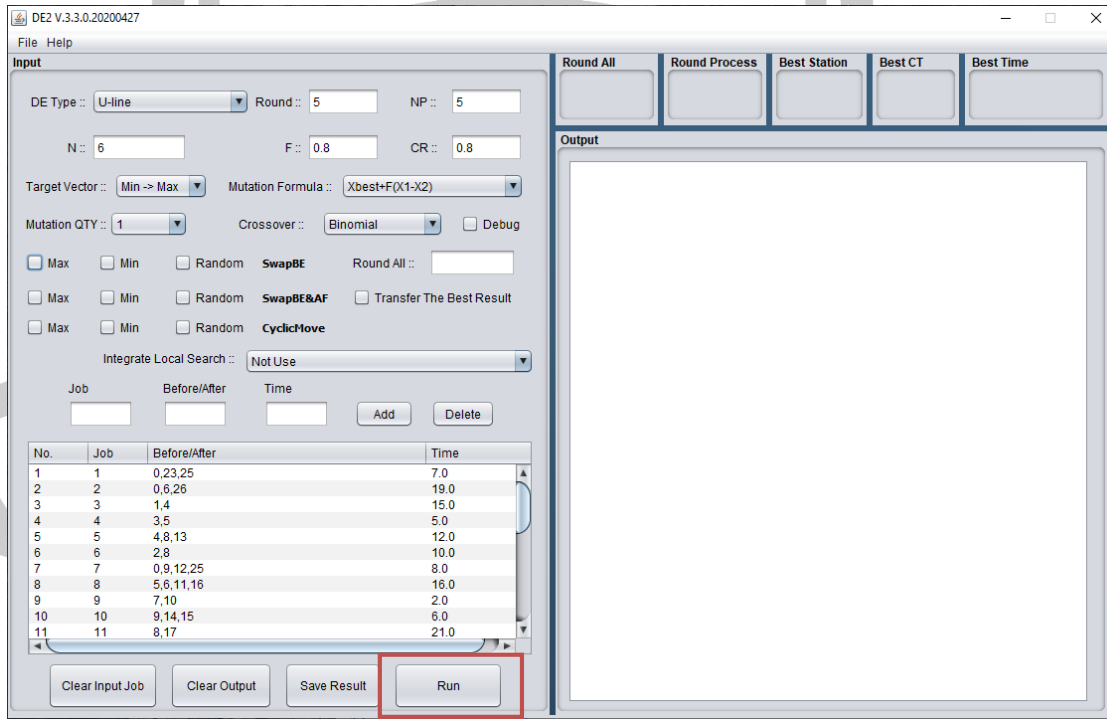
ภาพประกอบ 42 ขั้นตอนการโหลดข้อมูลของชุดปัญหาที่ทำการบันทึกไว้

หลังจากทำการบันทึกข้อในแต่ละปัญหาจนครับทุกชุดปัญหาแล้วนั้น ในการทดสอบปัญหาจะเริ่มจากเข้าไปที่ File >>> Load Parameter ในชุดปัญหาที่เราจะทำการทดสอบ ข้อมูลในชุดปัญหาจะแสดงได้ดัง ภาพประกอบ 43 ที่บอกจำนวนงาน (Job), ลำดับงานก่อนหลัง (Before/After) และเวลา (Time) ของในแต่ละงาน จากนั้นทำการทดสอบโดยทำการกดปุ่ม Run ดังแสดงในภาพประกอบ 44

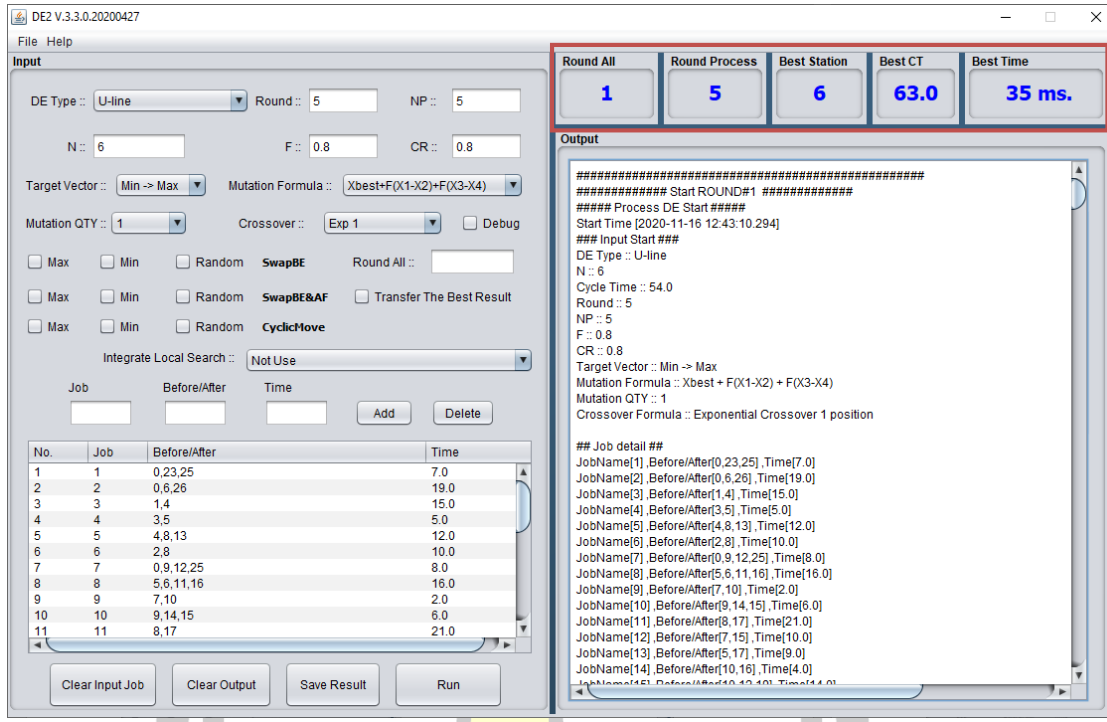
พหุน ปณ ฑิโต ชีเว



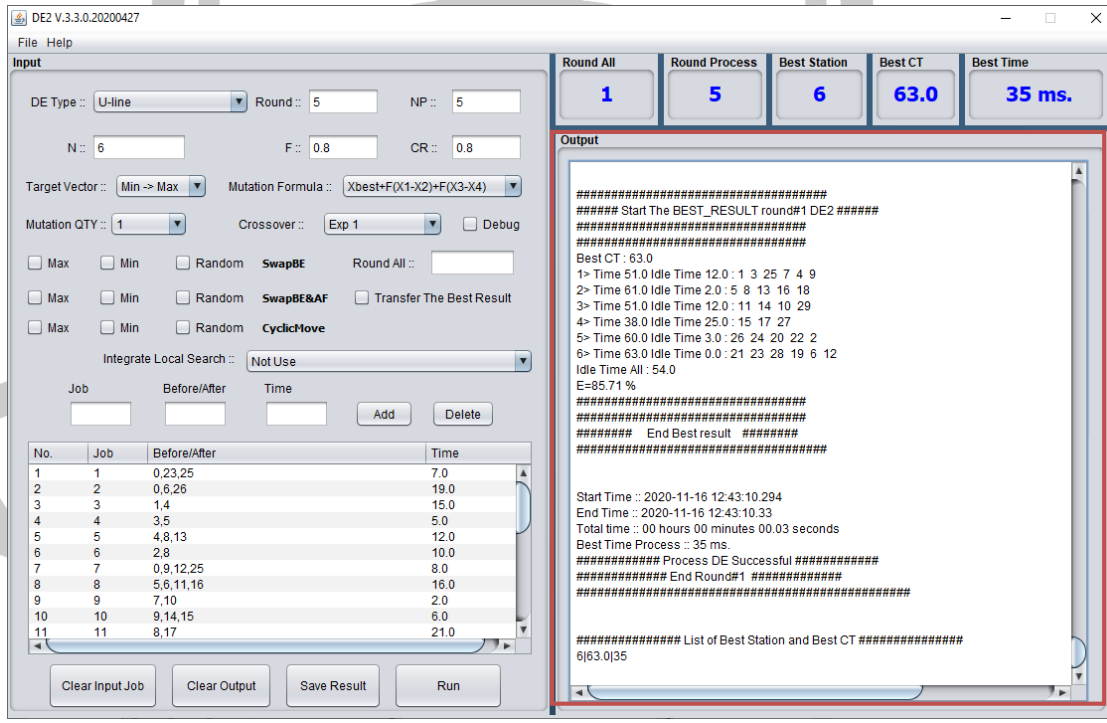
ภาพประกอบ 43 แสดงข้อมูลของชุดปัญหาที่จะทำการทดสอบ



ภาพประกอบ 44 ขั้นตอนของการรันผลของโปรแกรม

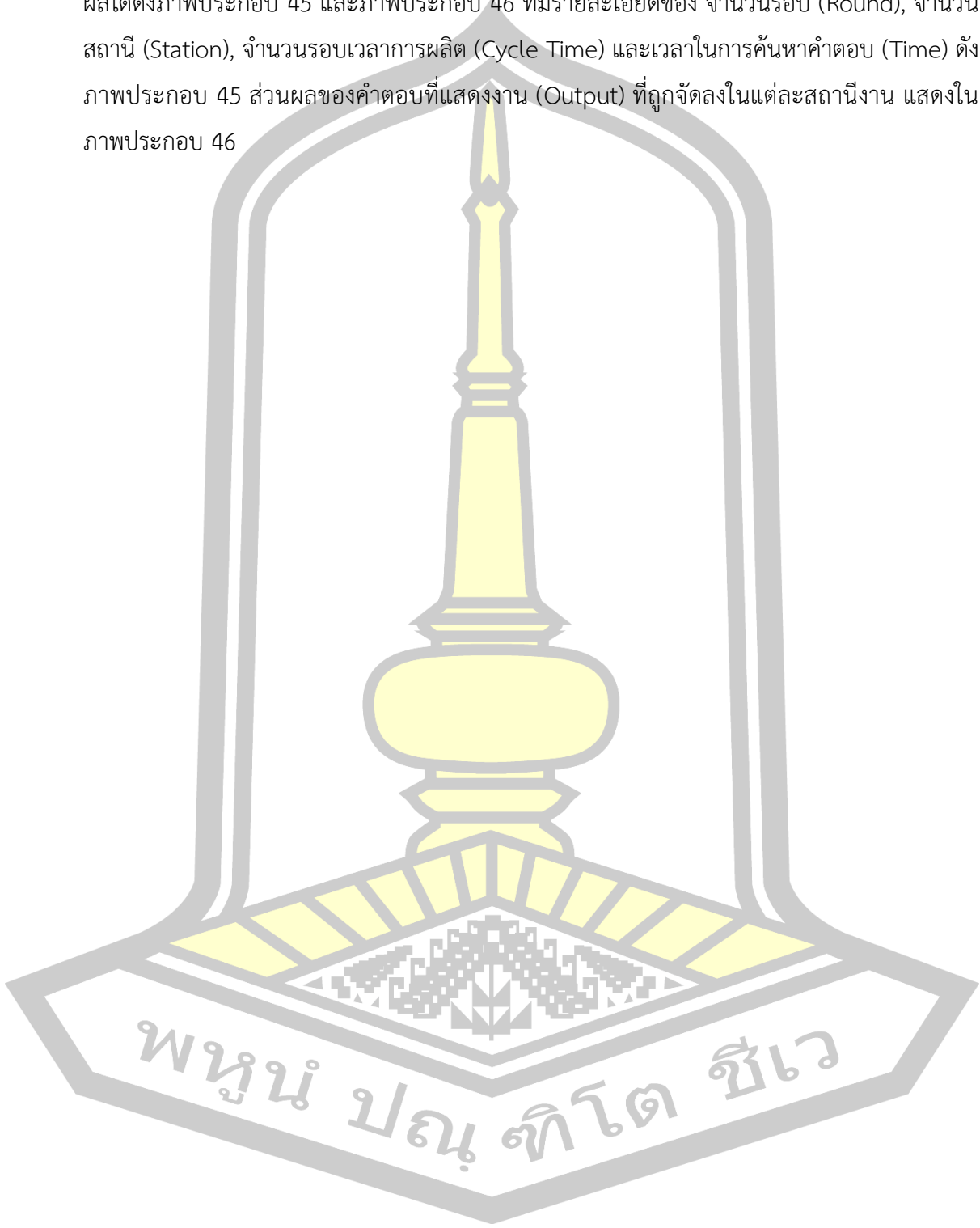


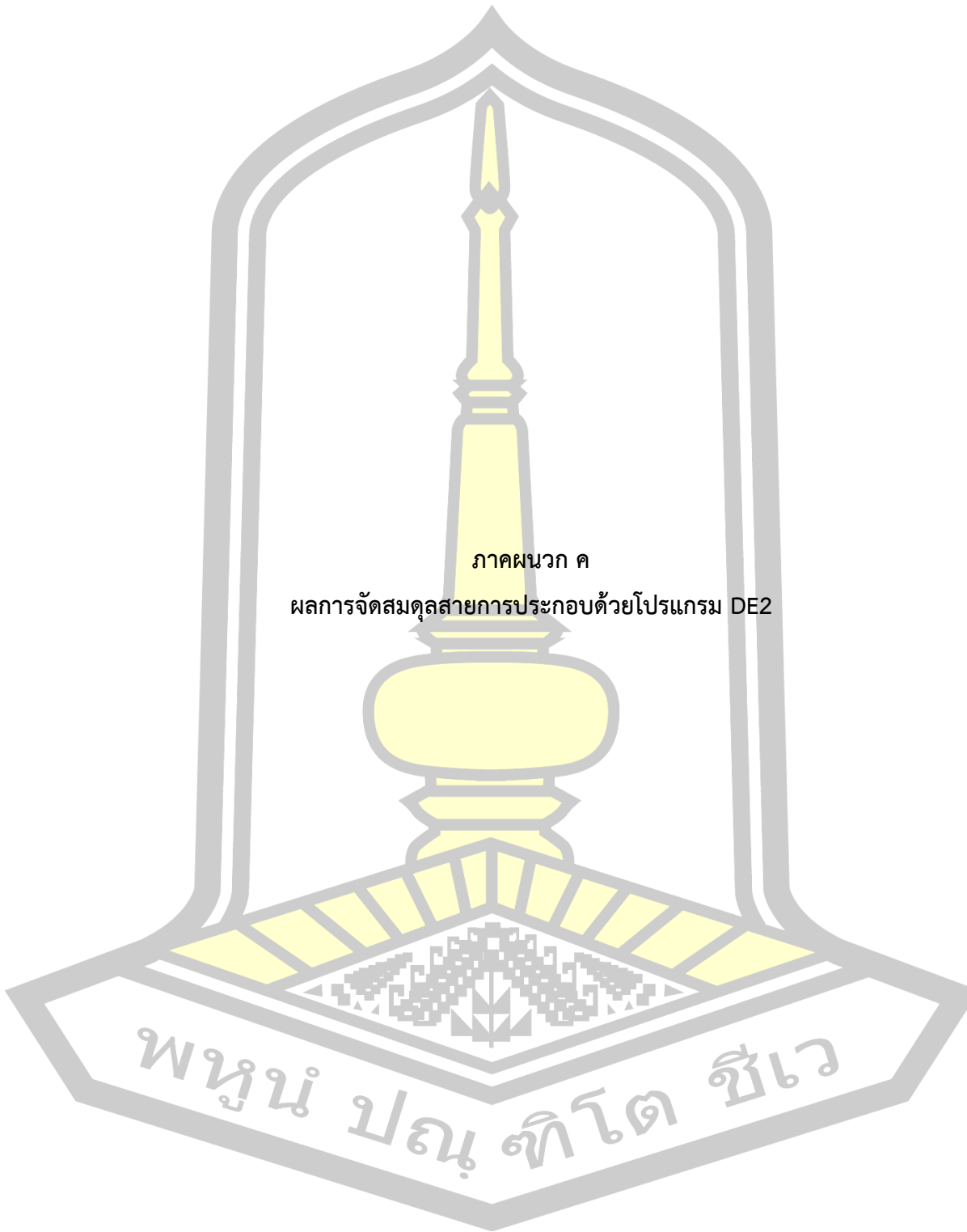
ภาพประกอบ 45 แสดงผลการคำนวณ



ภาพประกอบ 46 แสดงผลการจัดสมดุลสายการประกอบแบบ

หลังจากทำการทดสอบการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 สามารถแสดงผลได้ดังภาพประกอบ 45 และภาพประกอบ 46 ที่มีรายละเอียดของ จำนวนรอบ (Round), จำนวนสถานี (Station), จำนวนรอบเวลาการผลิต (Cycle Time) และเวลาในการค้นหาคำตอบ (Time) ดังภาพประกอบ 45 ส่วนผลของคำตอบที่แสดงงาน (Output) ที่ถูกจัดลงในแต่ละสถานีงาน แสดงในภาพประกอบ 46





ภาคผนวก ค

ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง

Problem	*m*	*c*	Result
Mitchell-21	4	27	1> 1 19 2 18 20 13 15
			2> 10 11 12 16 17 9
			3> 8 21 3 14
			4> 7 4 6 5
	5	21	1> 1 19 3 20 14
			2> 7 17
			3> 8 6 9 18
			4> 2 12 15 10 16 13 11
			5> 4 21 5
	6	18	1> 1 3 4
			2> 5 20 19 2
			3> 14 6 18 13
			4> 15 10 16 12 8
			5> 11 7 21
			6> 17 9
Rosenberg-25	4	32	1> 1 10 3 22 25 2 6 19
			2> 9 4 15 18 8 23 12
			3> 13 11 17 14 7
			4> 5 16 24 21 20
	5	25	1> 1 3 25 18 2
			2> 23 16 17 10 9
			3> 15 6 14 13 12 8
			4> 20 19 5 11 7
			5> 21 4 24 22
	6	21	1> 1 25 24 2 10
			2> 18 20 17
			3> 16 14 13 21 11
			4> 19 8 7 6
			5> 4 9 15 12 22
			6> 5 23 3

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Rosenberg-25		18	1> 1 10 3 25
			2> 23 4 6 20 2
	7		3> 7 13 14 19
			4> 11 21 16 22
			5> 17 15
			6> 5 12 8
			7> 24 9 18
	8	16	1> 1 22 19 15
			2> 21 3
			3> 17 12 10
			4> 4 14 7
			5> 11 13 18 20
			6> 24 25 2
			7> 5 6 16
			8> 8 23 9
Heskiaoff-28	6	171	1> 1 21 28 26 4 27 5
			2> 2 19 8 23
			3> 25 15 3 16 22 18
			4> 13 7 12 6
			5> 10 17 24
			6> 9 20 14 11
	7	147	1> 1 5 23 28
			2> 18 7 27 16 25 15
			3> 14 20 6
			4> 13 3 4
			5> 11 22 10 19 21
			6> 9 17 12
			7> 2 8 24 26

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Heskiaoff-28	8	129	1> 8 24 4 2 22 23 26 2> 1 19 27 21 3> 20 3 6 4> 9 7 25 18 5> 28 6> 10 14 7> 13 12 15 11 16 8> 5 17
	9	116	1> 1 23 5 26 3 27 2> 8 2 3> 28 9 4 22 4> 10 20 5> 6 7 18 12 24 21 6> 13 7> 16 17 8> 25 9> 19 11 15 14
	10	108	1> 1 3 23 5 2> 28 24 16 4 3> 18 2 6 22 15 4> 14 7 21 11 5> 10 19 27 26 6> 13 7> 20 9 8> 17 9> 25 10> 8 12
Buxey-29	5	65	1> 1 3 2 6 25 2> 8 11 26 29 4 3> 16 17 13 24 27 14 28 4> 23 10 18 7 22 5> 12 21 9 15 19 20 5

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Buxey-29	6	54	1> 1 7 9 29 28 27
			2> 23 2 6
			3> 20 17 8 10 26
			4> 16 13 18 5 4 14
			5> 11 24 22 12
			6> 21 19 3 25 15
	7	47	1> 1 2 6 26 7
			2> 8 5 3 9
			3> 11 12 13 10
			4> 14 15 17 27 4
			5> 20 16 19 25
			6> 18 21 22 29
			7> 28 24 23
	8	41	1> 1 3 25 4
			2> 5 29 7
			3> 28 27 24 12
			4> 23 8
			5> 6 22 11
			6> 21 20 16 18
			7> 13 14 17 26 9 19
			8> 10 2 15
	9	38	1> 1 29 28 7
			2> 23 9 22
			3> 10 24 14 21 19
			4> 18 16 8
			5> 5 20
			6> 4 15 17
			7> 11 25
			8> 13 12 2
			9> 3 6 27 26

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Buxey-29	10	33	1> 1 25 7 9
			2> 29 10 28
			3> 24 3 14
			4> 23 16
			5> 18 8
			6> 6 5 22 21
			7> 13 17 19
			8> 11 27
			9> 26 20 4 12
			10> 15 2
Sawyer-30	5	65	1> 1 28 27 2 29 5 30
			2> 26 23 12 10 22 13 9
			3> 14 25 20 8 16
			4> 24 21 19 3
			5> 18 6 11 17 4 15 7
	6	54	1> 1 28 3 4 16
			2> 27 23 26 30 5
			3> 25 20 29 9 6
			4> 21 14 22 24
			5> 13 12 11 19 17
			6> 18 2 7 15 8 10
	7	47	1> 1 5 4 2 30 11 10
			2> 6 3 24 29
			3> 17 25 19 26
			4> 9 18 7 12 28
			5> 13 20 27
			6> 14 23 15 16
			7> 21 22 8

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Sawyer-30	8	41	1> 1 3 10 28 30 2> 19 2 11 5 3> 16 20 29 4> 21 22 6 5> 7 23 18 6> 8 14 4 9 24 7> 26 25 12 8> 27 13 15 17
Lutz1-32	6	2424	1> 1 2 5 23 6 8 31 2> 3 22 25 26 7 32 3> 20 17 30 27 24 19 4> 29 28 4 14 5> 16 13 15 12 21 6> 11 9 10 18
	7	2092	1> 1 4 32 2> 15 9 2 17 31 3> 6 24 7 10 14 22 4> 20 3 25 30 19 5> 26 23 13 6> 16 5 8 29 7> 21 27 12 28 18 11
	8	1816	1> 1 3 2 5 6 2> 4 8 3> 32 24 9 7 22 4> 23 31 30 20 5> 29 25 11 10 6> 19 26 28 7> 17 16 14 27 8> 12 13 15 21 18

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Lutz1-32	9	1624	1> 1 32 3 31 2 2> 30 24 9 21 3> 29 28 20 10 4> 16 22 6 7 5> 27 14 19 18 6> 12 15 13 7> 17 5 11 8> 8 25 26 9> 23 4
	10	1474	1> 1 5 32 2 2> 4 3> 9 15 17 4> 24 23 11 5> 22 31 21 14 6> 19 30 8 25 7> 18 20 16 8> 26 12 10 9> 7 29 6 3 28 10> 27 13
Gunther-35	5	97	1> 1 34 27 35 33 2 5 17 29 26 2> 3 20 6 32 28 21 7 3> 22 24 11 8 18 4 31 30 14 4> 15 19 16 10 5> 9 23 12 13 25
	6	81	1> 1 29 12 17 34 35 18 7 2 27 14 2> 6 3 24 4 26 5 8 3> 23 11 10 22 4> 28 20 21 25 32 5> 31 13 16 30 15 6> 33 19 9

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Gunther-35	7	69	1> 1 35 12 18 5 2> 10 29 19 34 14 6 3> 15 20 8 27 26 21 4> 28 16 5> 25 13 11 32 22 30 7 6> 31 23 24 4 2 7> 3 33 9 17
	8	61	1> 1 12 17 2> 18 5 35 29 34 7 33 27 3> 24 6 19 8 4> 11 9 23 5> 13 32 22 4 2 14 6> 21 3 31 30 10 26 25 7> 16 20 8> 28 15
Kilbridge&Wester-45	6	92	1> 1 11 39 2 4 8 14 29 43 37 2> 32 30 6 25 10 31 26 38 3> 36 28 33 41 34 19 20 40 44 4> 22 16 35 27 23 18 42 45 5> 15 24 17 7 3 12 13 6>: 21 5 9
	7	79	1> 1 2 8 44 11 39 12 42 45 2> 10 14 29 37 13 15 17 3> 24 18 27 33 28 34 4> 6 16 36 19 22 26 38 5> 40 20 23 4 31 25 6> 21 35 7 32 7> 9 5 3 41 30 43

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Kilbridge&Wester-45	8	69	1> 1 3 7 14 45 39 44 2> 31 9 30 42 8 17 3> 26 27 38 10 11 12 43 40 4> 28 34 36 33 22 32 5> 19 21 18 29 6> 20 16 24 6 7> 23 25 35 2 8> 41 5 15 13 37 4
	9	62	1> 1 2 12 44 4 42 37 2> 8 6 11 41 3>: 40 38 36 14 31 32 29 30 4> 28 17 25 5> 22 9 27 5 26 6> 33 35 13 19 23 7> 20 24 16 43 18 8> 21 45 9> 3 7 10 15 34 39
	10	56	1> 1 2 12 44 4 42 37 2> 8 6 11 41 3> 40 38 36 14 31 32 29 30 4> 28 17 25 5> 22 9 27 5 26 6> 33 35 13 19 23 7> 24 16 43 18 8> 21 45 9> 3 7 10 15 34 39
Hahm-53	5	2820	1> 8 3 36 5 9 10 2 4 35 30 37 23 20 6 29 32 11 41 2> 38 39 28 31 34 12 1 16 24 27 40 7 14 3> 13 15 17 21 22 18 19 4> 25 42 33 26 51 5> 52 47 49 48 50 43 45 46 44 53

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Hahm-53	6	2416	1> 1 5 2 3 4 7 10 23 6 9 20 2> 37 38 24 35 17 11 12 13 33 40 39 14 30 31 32 29 3> 41 21 53 19 34 16 22 28 18 25 4> 27 26 15 36 5> 52 45 48 8 6> 50 46 49 51 47 44 43 42
	7	2010	1> 8 6 3 2 7 4 5 20 2> 36 9 37 40 35 32 31 28 23 38 1 24 39 3> 33 16 30 41 29 21 13 10 12 34 53 4> 14 22 25 18 27 19 26 5> 51 49 50 48 44 45 17 47 11 46 6> 43 52 15 7> 42
	8	1775	1> 1 7 3 36 4 35 6 2 34 2> 53 37 24 29 32 51 30 49 3> 45 8 16 44 28 4> 13 18 33 38 52 12 47 5> 15 41 39 21 6> 19 14 22 25 20 9 17 48 7> 42 8> 46 43 10 23 27 40 11 5 31 26 50
	9	1775	1> 1 2 7 36 35 9 32 5 2> 53 10 33 34 28 11 23 3> 27 51 8 49 47 45 50 4> 42 5> 41 46 52 26 40 6> 25 44 38 39 22 12 18 30 37 43 24 14 6 7> 19 16 17 4 20 21 29 13 48 31 8> 15 3

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Warnecke-58	10	155	1> 44 3 8 58 21
			2> 1 45 10 46 20 47
			3> 25 55 48 7 32 9
			4> 38 11 36 14
			5> 15 13 2 31 34
			6> 29 37 39 5 26
			7> 16 41 23 57 24 50 49
			8> 19 56 4 30 6 27
			9> 53 52 40 12 51 54 42 28
			10> 18 33 43 22 35 17
	11	143	1> 11 58 10 9 6 28 50
			2> 46 14 1 47 51 13 52
			3> 38 31 36 4> 2 37 34 17 40
			5> 55 25 4 54 20
			6> 21 3 8 16 41
			7> 42 45 19 26 53 57
			8> 22 27 7 29 49
			9> 56 5 12 15 48
			10> 32 43 24 23
			11> 35 30 18 39 44 33
	12	131	1> 44 8 5 25 28
			2> 6 11 38 9 21
			3> 20 3 17 15 4> 14 16 32 1
			5> 36 37 58 22
			6> 34 39 2 41 26
			7> 42 43 27 23 13 49
			8> 4 57 19 40 55 45
			9> 48 54 50 24 35 52 12
			10> 46 51 29 10
			11> 53 33 7 12> 31 30 18 47 56

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Warnecke-58	13	120	1> 1 8 5 6 7
			2> 36 23 19 18
			3> 37 43 39 34 28
			4> 45 42 30 33 40
			5> 27 44 58 10
			6> 50 41 21 24 16 57
			7> 56 20 25 26 54 9
			8> 17 53 13 52
			9> 15 14 11 46
			10> 51 29 3 49 48
			11> 55 2 12
			12> 22 32 4
			13> 31 35 47 38
	14	111	1> 7 8 58 9 12
			2> 2 1 49 17 50
			3> 5 13 46 15 47
			4> 51 55 48 3 54
			5> 43 28 52 57
			6> 39 35 37 32
			7> 56 25 53
			8> 22 21 26 41 45
			9> 29 11 14
			10> 4 20 44
			11> 33 27 42 16
			12> 30 34 6 10
			13> 23 19 36 40
			14> 31 18 24 38

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Warnecke-58	15	104	1> 3 7 4 28 2> 10 9 11 8 3> 21 44 20 5 4> 22 25 27 5> 1 30 36 13 29 6> 34 12 15 7> 31 24 37 16 8> 41 45 46 26 9> 2 47 50 48 49 10> 58 19 32 42 40 11> 39 14 12> 33 51 52 13> 6 56 55 18 14> 54 17 53 15> 35 38 23 43 57
	16	97	1> 5 16 7 8 2> 10 6 58 3> 44 45 46 4> 3 50 49 47 24 5> 51 32 43 6> 9 53 1 52 7> 22 35 38 42 37 18 8> 39 48 55 9> 41 33 11 10> 12 25 20 11> 13 15 17 19 4 12> 36 28 40 13> 56 27 2 14> 30 23 54 21 15> 14 26 16> 31 29 57 34

ตาราง 17 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดกลาง (ต่อ)

Problem	*m*	*c*	Result
Warnecke-58	17	94	1> 3 2 8 2> 32 7 28 3> 33 45 4> 1 6 46 35 5> 49 55 50 54 23 6> 43 24 47 7> 51 52 25 39 8> 48 38 36 57 9> 17 34 20 10> 14 30 13 37 11> 44 27 15 12> 41 11 4 13> 53 12 14> 56 22 42 15> 21 10 19 18 16> 26 5 58 9 17> 29 16 40 31

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่

Problem	*m*	*c*	Result
Tonge-70	10	359	1> 1 70 60 59 41 69 50 58 66 56 67 9 24 65 2> 57 61 64 30 63 62 49 35 3> 26 44 68 43 48 53 51 27 52 4> 45 46 54 15 31 5> 3 23 14 32 2 6> 22 8 16 17 12 7> 18 7 11 10 36 8> 37 25 39 5 28 21 34 42 9> 40 20 33 55 29 10> 47 13 6 38 4 19
	11	320	1> 1 66 2 60 55 68 56 70 2> 16 3 15 4 50 65 3> 18 19 49 24 22 7 30 4> 48 31 67 59 43 61 57 64 69 20 41 5> 9 63 32 62 58 42 6> 40 6 23 47 39 35 7> 38 13 29 12 37 28 53 51 8> 52 11 34 46 26 45 9> 10 44 25 27 10> 8 54 21 11> 5 14 36 17 33
	12	293	1> 66 50 47 41 2 42 60 2> 3 35 32 28 69 51 67 31 3> 26 25 52 55 53 4> 46 6 61 1 68 30 62 5> 45 63 34 48 6> 33 49 65 40 29 64 24 70 39 38 7> 37 54 56 9 59 8> 58 57 10 23 19 21 36 9> 11 14 12 22 10> 18 13 11> 17 20 8 44 12> 7 15 4 16 43 5 27

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Tonge-70	13	275	1> 1 66 56 70 15 41 65 50
			2> 61 35 32 44 5 6 30
			3> 28 48 69 34 31
			4> 25 23 36 37
			5> 21 14 38 22 45 51
			6> 39 13 67 2 49 40
			7> 54 3 9 19
			8> 18 10 20
			9> 11 33 4 64
			10> 63 7 26 27
			11> 12 42 53 62 16 57 60
			12> 43 59 58 8 55 29 52
			13> 46 17 24 68 47
	14	255	1> 1 2 55 67 5 65 3 69 30
			2> 61 31 4 15 7 66
			3> 23 16 50 22
			4> 21 18 17
			5> 43 13 49 19 41
			6> 12 42 57 6 68 58 40
			7> 11 39 10 47 38
			8> 46 37 54 70 59
			9> 52 53 36 51 8
			10> 25 20 45 24
			11> 44 27 26 48 64 60
			12> 63 62 29 35
			13> 34 14 56
			14> 33 28 9 32

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Tonge-70	15	237	1> 1 2 5 56 60 50 2> 24 49 6 35 55 3> 51 8 29 34 7 4> 36 32 37 44 15 38 5> 16 39 45 62 46 6> 40 25 4 59 7> 58 23 31 3 57 61 67 8> 22 14 64 69 9> 68 26 43 27 42 52 10> 54 48 21 11> 9 13 19 30 12> 10 33 53 65 13> 11 28 66 12 70 14> 63 17 41 15> 18 47 20
	16	232	1> 1 70 41 3 2 42 50 40 39 2> 43 49 24 15 38 48 37 3> 4 25 5 30 4> 31 28 61 64 55 7 5> 8 47 6> 54 46 7> 52 51 67 45 68 12 8> 11 44 9 69 9> 63 66 53 10> 62 14 32 11> 27 13 60 12> 59 58 10 56 36 29 13> 57 19 18 35 14> 17 34 20 21 15> 33 16 16> 6 65 22 26 23

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Lutz2-89	13	38	1> 1 89 48 2 30 47
			2> 87 46 49 45 3 4 54
			3> 16 8 14 6 55 7 21 11
			4> 43 15 56 57 66 9 12
			5> 61 58 65 17 60 67
			6> 59 20 10 63 5
			7> 28 50 64 39 13 73 41
			8> 53 68 29 51 72 69 25 71
			9> 27 31 78 77 26 52 75 42
			10> 32 70 33 79 74 81 34 35
			11> 76 84 85 36 82 40 24
			12> 23 83 22 19 18 80 62
			13> 38 86 88 37 44
	14	35	1> 1 30 88 48 54 21 2
			2> 3 20 47 55 16 4
			3> 8 9 28 14 46
			4> 29 45 13 31 25 49 43
			5> 24 32 26 39 33 50 51
			6> 38 52 41 36 44 34 53 35 71
			7> 73 70 67 69 23 42
			8> 58 59 61 62 63 60 72
			9> 78 77 75 79 66
			10> 81 76 7 64 10 37
			11> 85 84 82 40 18 86
			12> 83 19 6 68 56 22
			13> 11 89 65 87 80 12
			14> 27 74 15 17 57 5

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Lutz2-89	15	33	1> 1 48 30 88 2 54
			2> 3 21 87 55 47 46 4
			3> 9 49 50 45
			4> 56 20 43 51 52 28 53
			5> 29 31 68 7 32 5 41
			6> 33 10 14 89 34 42 71
			7> 35 37 36 38 26
			8> 18 70 17 6 72
			9> 19 80 13 44 22 11 12
			10> 81 85 23 40 84 39
			11> 76 74 79 82 75 24 73
			12> 83 77 67 63 86 27
			13> 59 62 65 64 60 61
			14> 66 78 57 8
			15> 25 69 16 58 15
	16	31	1> 1 89 48 47 2 3
			2> 4 46 6 49 45 11
			3> 12 7 43 13 50 5
			4> 39 51 10 14 52 15
			5> 41 17 44 53 16 40
			6> 70 18 69 42 71
			7> 73 67 19 20 22
			8> 74 63 60 64 66 61 23
			9> 76 28 21 75 29 31
			10> 77 25 78 32 65 24
			11> 27 88 58 57
			12> 56 9 30 62 33
			13> 55 34 79 68 54
			14> 81 35 85 38 59 72
			15> 84 86 36 83 82 8
			16> 26 37 80 87

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Lutz2-89	17	29	1> 1 88 2 30 3 54 2> 4 21 16 8 9 3> 55 56 20 7 4> 57 66 58 28 5> 61 29 59 63 67 6> 25 31 89 18 64 12 7> 19 32 11 33 17 87 8> 86 15 85 83 82 22 14 81 9> 84 76 10 23 5 10> 79 60 26 74 11> 37 35 75 77 73 62 12> 38 39 43 78 71 13> 40 53 44 70 42 14> 69 52 72 41 36 45 24 15> 51 50 49 47 16> 80 46 68 34 17> 6 27 48 65 13
	18	27	1> 1 88 89 2 87 2> 3 86 4 8 6 3> 7 9 84 81 5 11 4> 12 82 10 80 13 5> 72 14 70 78 53 6> 15 77 16 68 41 7> 75 74 39 73 52 54 8> 38 55 40 44 9> 37 35 26 56 51 21 71 10> 25 24 23 20 43 22 11> 57 61 36 27 64 42 12> 62 59 65 58 33 13> 60 48 19 14> 47 46 32 79 63, 15> 31 69 17 30 16> 18 34 67 28, 17> 50 76 66 45 18> 29 49 83 85

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Lutz2-89	19	26	1> 1 89 30 54 2
			2> 3 55 88 21 4 5
			3> 7 56 9 12
			4> 18 17 8 11
			5> 6 20 19
			6> 28 48 10
			7> 29 47 57 62 61
			8> 64 46 31 49 32 63
			9> 59 50 25 22
			10> 27 51 60 23 45
			11> 26 87 15 16 52
			12> 44 40 37 53 42
			13> 35 71 38 86 83 33
			14> 85 36 82 39 24
			15> 68 41 65 66 81 34
			16> 80 72 76 13 43
			17> 74 75 78 79
			18> 70 69 14 73
			19> 58 77 84 67
	20	25	1> 1 89 2 87 3, 2> 4 9 86 6
			3> 18 17 15 85 4> 83 8 48 11 14 81
			5> 10 76 79 6> 78 74 75 47
			7> 77 80 72 46 8> 69 68 53 42 82 71
			9> 52 51 41 49 7 44
			10> 50 30 54 21
			11> 55 19 39 12
			12> 22 38 56 57 61
			13> 60 62 67 36
			14> 58 59 43 23
			15> 37 16 24 63 84 16> 25 26 88 45
			17> 29 20 13 73 18> 31 32 33 40 27
			19> 34 64 65 35 5 20> 70 28 66

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Lutz2-89	21	24	1> 1 30 2 89 2> 87 3 86 4 83 3> 8 48 7 4> 18 82 88 12 13 81 5> 79 11 76 17 6> 15 20 9 7> 74 14 85 6 21 54 8> 16 55 73 10 9> 71 75 77 80 10> 53 70 69 41 11> 39 68 42 72 43 12> 45 52 5 46 44 51 13> 67 47 50 14> 59 66 61 65 62 15> 64 60 49 16> 38 63 57 84 17> 36 37 58 18> 35 26 25 27 19> 34 33 32 29 24 20> 31 23 19 40 21> 22 28 78 56
Mukherjee-94	14	307	1> 1 7 5 3 2> 93 8 29 4 94 18 10 3> 92 75 48 19 32 82 16 44 4>: 69 83 36 84 60 14 13 25 30 5> 22 38 23 21 24 73 59 26 27 50 33 6> 55 40 35 71 31 7> 57 78 37 81 61 8> 68 53 76 28 9> 52 12 39 11 2 10> 17 20 66 90 9 43 89 34 11> 88 45 80 42 63 91 46 58 12> 87 77 62 41 13> 47 74 70 6 79 49 56 14> 64 67 65 15 72 51 86 54 85

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Mukherjee-94	15	283	1> 1 6 3 10 13 2> 15 50 7 91 94 90 2 3> 17 82 5 12 4> 74 83 8 40 52 18 20 64 28 5> 77 29 14 32 44 30 43 69 6> 27 62 92 55 24 59 61 23 22 7> 34 38 35 31 76 58 49 8> 89 54 57 51 88 36 25 9> 63 73 93 72 33 10> 75 46 87 67 84 11> 66 45 78 9 65 12> 85 81 86 47 80 13> 11 79 4 26 56 60 21 48 16 14> 41 39 71 53 15> 42 37 70 68 19
	16	265	1> 1 90 2> 93 9 91 5 8 3> 16 94 7 2 3 4 42 4> 29 14 82 30 59 81 74 5> 83 80 78 33 17 67 10 21 19 6> 79 24 73 44 31 23 25 27 50 48 65 7> 75 77 60 62 58 26 61 64 8> 43 63 22 41 92 39 9> 69 38 12 54 46 10> 70 52 40 37 11> 11 35 55 72 20 28 12> 51 13 57 13> 34 68 76 32 66 56 89 18 49 14> 47 88 45 87 15> 53 6 84 16> 86 15 85 71 36

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Mukherjee-94	17	251	1> 1 9 7 2> 6 91 11 4 15 3> 14 31 22 26 32 58 30 27 4> 21 41 39 90 36 44 5> 53 78 6> 76 33 40 35 7> 62 48 5 42 43 79 8> 54 56 72 60 16 69 45 9> 37 68 23 66 8 46 38 10> 81 55 82 2 70 61 11> 12 83 84 17 52 20 80 18 19 12> 86 47 71 29 25 13> 85 63 93 28 14> 51 74 94 77 15> 67 65 92 89 88 64 34 16> 87 75 24 49 17> 10 50 73 13 59 3 57
	18	236	1> 6 91 10 3 12 52 2> 94 1 28 4 49 50 3> 90 20 22 14 4> 21 60 27 24 39 23 30 16 58 43 75 5> 51 34 55 42 46 6> 74 67 65 44 64 40 2 69 7> 11 36 53 8> 66 33 78 9> 77 25 73 61 81 72 10> 79 80 93 15 32 76 11> 63 48 47 35 45 12> 62 57 19 38 7 13> 59 13 29 31 8 56 41 14> 9 26 37 92 82 15> 71 83 70 18 16> 5 54 17 17> 84 68 86 18> 89 88 85 87

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Mukherjee-94	19	224	1> 82 2 17 6 2> 13 4 74 3 91 15 3> 81 73 19 65 67 64 25 80 37 4> 77 78 33 79 61 5> 66 28 52 40 54 27 6> 49 11 58 22 71 38 7> 21 56 55 39 41 8> 42 70 35 16 43 9> 90 57 68 10> 63 8 69 9 11> 32 44 29 45 14 62 36 50 12> 26 30 7 94 31 13> 59 23 92 24 12 14> 72 75 46 15> 89 84 60 10 51 83 16> 88 87 47 17> 86 85 20 34 93 18> 53 48 5 19> 18 76 1
	20	221	1> 69 10 3 19 2 37 32 2> 76 4 91 45 16 17 66 3> 24 46 44 60 14 29 5 4> 72 71 58 21 30 5> 22 39 70 49 18 41 33 48 6> 38 77 25 9 63 7> 1 27 68 55 8> 31 26 61 43 9> 73 13 7 56 74 10> 93 54 28 64 94 11> 90 59 52 92 12> 89 88 87 50 36 13> 82 83 57 67 14> 78 23 42 84 85 15> 86 79 65 8 51 16> 34 12 75 17> 11 40 20 18> 47 81 80 19> 6 53 20> 15 35 62

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Mukherjee-94	21	210	1> 8 9 11 2> 94 91 6 16 19 48 3> 5 29 10 4 50 47 4> 32 44 13 45 18 21 36 5> 77 60 25 28 6> 41 12 33 61 3 63 7> 57 53 8> 20 49 22 52 58 51 43 9> 38 27 73 10> 62 55 69 70 11> 93 56 2 82 74 12> 1 81 83 64 13> 17 65 84 67 26 80 14> 86 85 37 24 15> 66 78 54 15 79 16> 92 14 30 17> 23 89 88 59 31 39 34 18> 87 68 7 19> 40 35 71 20> 72 42 76 21> 90 46 75
	22	201	1> 1 9 2 2> 4 91 93 8 10 3> 48 94 16 90 17 4> 29 45 32 3 11 5> 12 46 36 6> 13 6 69 7> 57 60 44 7 25 8> 77 63 21 56 19 9> 33 28 40 52 51 10> 41 78 39 49 43 11> 42 92 79 70 12> 35 81 14 13> 71 89 14> 24 23 31 27 55 58 15 5 15> 88 87 20 16> 47 18 34 17> 62 22 74 30 59 18> 38 75 54 65 26 80 82 19> 64 67 73 50 66 20> 83 84 76 86 21> 53 37 22> 85 72 68 61

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Mukherjee-94	23	193	1> 1 16 5 2> 93 75 3> 82 18 46 4 32 4> 3 50 76 14 44 5> 77 72 24 30 60 22 6> 47 45 38 59 7> 31 27 83 25 61 58 49 73 8> 54 33 84 28 40 36 9> 11 74 65 10> 64 42 55 10 11> 37 13 56 2 12> 68 8 90 19 17 13> 94 57 29 6 14> 81 12 15> 80 66 67 43 63 16> 15 9 20 92 17> 53 62 18> 89 88 51 48 78 19> 71 20> 35 26 70 21> 21 79 41 86 22> 87 69 91 23> 85 39 23 7 34 52
	24	186	1> 1 3 8 2> 29 10 7 4 19 3> 93 9 91 37 4> 6 2 72 15 66 17 5> 13 61 70 14 44 6> 71 24 30 7> 82 83 25 23 36 27 55 31 64 8> 59 74 60 33 57 34 9> 54 65 28 40 22 38 10> 49 67 58 43 69 11> 78 16 79 76 12> 52 53 13> 62 81 39 14> 21 90 32 80 15> 84 47 5 45 16> 46 86 17> 11 48 85 18> 26 18 92 19> 94 35 51 20> 89 41 21> 12 56 22> 20 75 63 73 23> 88 50 68 87 24> 77 42

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Bartholdi-148	7	805	1> 1 48 33 147 13 35 52 148 94 127 137 132 44 144 41 103 42 57 110 60 2> 101 43 59 142 56 64 88 11 130 58 53 98 146 40 47 50 36 138 37 3> 86 73 96 65 39 46 51 69 45 31 129 139 38 54 90 141 29 111 27 49 126 24 99 4> 25 26 23 66 118 21 76 20 128 75 55 112 8 119 12 34 131 67 95 22 7 68 136 5 2 70 32 5> 133 77 14 15 113 3 123 19 63 116 114 16 120 78 79 89 109 122 140 18 30 93 61 6> 9 62 115 124 121 80 108 71 87 135 100 17 72 125 82 6 7> 81 107 106 83 105 84 28 134 145 102 104 143 85 97 4 74 91 92 10 117
	8	705	1> 1 44 32 148 127 97 42 6 64 65 47 43 71 110 142 103 57 2> 109 46 108 82 45 49 102 2 85 92 131 60 61 3> 79 94 141 9 91 30 40 52 4> 14 80 39 105 138 143 13 11 54 75 101 147 50 137 139 95 38 68 62 70 78 3 5> 12 55 76 63 15 17 77 4 19 51 100 20 72 87 107 132 58 53 8 74 98 10 37 6> 21 26 29 27 106 134 86 36 25 67 90 99 125 23 119 22 7 48 126 93 145 18 41 66 7> 84 122 35 111 118 130 140 124 73 34 88 112 83 69 133 135 5 104 24 33 89 8> 121 113 117 129 114 96 123 116 128 120 146 56 115 16 59 144 81 136 31 28

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Bartholdi-148	9	626	1> 1 41 97 146 148 57 40 94 110 70 52 141 87 86 137 32 61 2> 79 62 89 104 44 140 69 126 11 51 33 125 132 3> 144 35 12 53 58 88 13 56 82 118 39 73 48 74 80 4> 117 78 124 145 92 142 85 116 127 139 130 83 122 109 5> 103 101 108 77 76 102 121 123 47 46 107 129 138 6> 106 95 128 100 71 50 45 7 2 30 36 54 75 42 115 63 7> 99 14 31 3 16 55 4 65 114 17 66 136 120 135 5 67 119 43 84 59 81 6 37 8> 29 28 26 9 23 24 19 27 133 21 18 105 143 22 72 98 38 134 64 96 93 131 68 9> 25 20 34 60 49 113 147 91 8 90 10 15 112 111
	10	564	1> 1 70 64 126 50 87 74 30 72 118 148 91 5 8 60 2> 14 142 42 75 110 4 127 43 6 32 86 134 88 122 3 3> 102 90 101 95 100 111 138 34 121 117 59 89 9 125 135 13 99 103 119 48 4> 120 112 124 144 58 71 131 5> 113 123 128 116 41 85 132 7 10 141 94 44 136 6> 16 36 52 57 97 130 147 92 2 35 61 115 68 33 137 51 140 139 31 82 7> 37 29 28 27 23 22 38 25 69 17 26 18 81 79 21 8> 24 80 20 19 39 146 133 83 45 47 145 78 109 11 104 62 67 12 9> 96 84 108 107 54 55 77 76 46 106 66 10> 63 98 73 53 49 56 105 93 40 129 143 15 65 114

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Bartholdi-148	11	513	1> 1 110 140 143 79 48 52 104 6 97 2> 91 148 3 9 141 4 139 96 89 60 3> 105 147 62 145 30 40 94 13 63 109 137 11 4> 130 132 50 73 125 127 102 67 14 138 5 144 39 85 80 5> 103 86 135 133 38 142 15 47 55 136 32 6> 134 17 8 53 126 59 58 146 12 129 33 66 131 56 10 54 7> 18 34 19 118 20 24 28 65 128 122 121 21 36 41 119 114 74 117 42 27 46 76 8> 113 45 25 43 26 29 31 123 116 99 70 120 100 35 51 101 72 9> 98 95 112 124 88 75 87 93 2 7 37 22 10> 111 108 23 44 57 115 11> 107 64 82 49 92 68 71 77 69 106 16 90 81 61 78 83 84
	12	470	1> 1 60 42 58 86 119 127 93 102 125 50 137 91 52 2> 115 51 105 85 124 2 8 33 43 61 64 11 148 62 40 131 3 103 3> 72 123 4 79 65 66 63 47 46 39 80 4> 67 45 38 144 71 89 5> 73 96 68 41 141 110 143 35 95 6 6> 109 140 10 74 48 98 138 146 92 59 135 12 32 9 44 7> 133 55 108 14 107 56 37 8> 16 145 94 87 36 106 136 57 113 81 9> 17 18 84 82 31 120 112 15 30 126 54 78 114 10> 83 29 118 111 25 27 21 22 23 77 26 75 13 69 90 99 34 70 11> 100 117 24 121 76 122 142 128 88 104 147 19 12> 129 116 139 7 53 134 130 49 97 132 5 20 28 101

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Bartholdi-148	13	434	1> 1 44 127 103 60 33 88 119 70 58 41 50 110 32 2> 144 64 13 148 42 59 137 40 131 61 39 3> 54 12 62 47 72 134 91 7 142 3 4> 135 143 30 38 43 132 89 90 85 49 73 46 5> 79 96 37 75 57 6> 82 105 111 35 11 92 14 48 78 136 34 36 133 8 7> 83 16 9 84 4 141 125 31 130 52 140 8> 124 115 106 139 10 17 63 86 109 51 77 45 93 29 65 28 76 138 9> 99 67 66 114 107 19 27 23 25 24 18 100 26 113 21 126 87 81 56 10> 122 128 108 123 11> 118 121 120 116 53 68 2 98 15 69 55 6 12> 117 112 145 147 95 13> 146 74 104 22 71 102 97 94 20 80 129 101 5
	14	403	1> 1 50 126 140 138 127 89 30 147 62 70 132 139 52 57 2> 122 63 48 125 42 142 7 137 11 121 58 3> 14 86 130 56 120 92 12 3 131 44 87 146 5 97 4> 75 53 33 119 8 118 90 124 5> 117 111 116 123 141 10 49 74 32 67 6> 112 66 35 36 68 110 41 6 61 7> 34 37 45 38 145 4 39 8> 148 43 102 95 105 109 15 136 101 9> 17 100 93 19 99 91 115 18 94 13 65 129 10> 108 114 11> 107 128 2 51 135 144 9 31 64 113 46 12> 29 69 27 134 24 26 79 80 20 13> 23 25 22 21 81 82 143 28 47 72 104 16 40 98 14> 78 83 77 84 96 73 103 106 133 88 54 60 59 85 55 76 71

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	21	3460	1> 1 2 3 294 297 293 4 22 48 2> 25 52 24 134 26 83 30 94 111 259 247 221 105 110 162 266 136 278 283 56 3> 139 116 61 119 253 122 65 40 60 86 167 5 34 90 29 4> 68 260 31 69 93 171 82 35 57 296 5> 267 42 98 74 172 102 76 62 174 89 46 138 273 71 33 88 129 63 100 223 6> 59 66 67 178 288 141 99 181 104 38 103 186 185 276 191 200 188 187 7> 189 108 183 115 180 184 195 151 182 205 207 193 281 204 196 210 202 87 197 8> 227 209 213 230 229 251 130 199 127 236 6 256 9> 123 157 147 211 232 234 214 289 146 239 117 144 9 14 145 163 10> 20 279 286 155 23 19 124 166 238 7 263 156 11> 285 11 28 257 10 12 149 13 215 240 170 158 18 203 12> 216 243 37 16 160 206 241 17 217 242 218 246 208 13> 248 39 249 245 284 43 64 194 235 44 190 14> 254 261 255 47 49 269 268 51 101 142 274 287 8 262 15> 277 55 53 106 81 15 109 27 179 225 271 41 84 175 237 16> 45 75 91 21 112 58 96 291 264 114 32 154 140 219 17> 220 36 54 143 169 222 153 70 252 224 282 295 50 73 148 212 18> 226 165 159 176 258 168 228 77 131 270 121 19> 231 233 173 265 133 135 128 132 177 97 198 107 250 292 72 20> 78 113 272 80 118 192 85 137 275 92 125 79 95 152 244 120 280 201 126 150 164 290 161

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	23	3133	1> 1 294 2 293 297 4 2> 296 295 3 44 40 247 109 3> 259 26 86 134 56 223 221 105 120 84 127 157 166 94 91 4> 119 74 61 130 22 73 75 96 25 136 111 27 5> 131 170 123 133 148 146 117 149 159 116 147 132 160 257 6> 31 101 264 124 172 34 122 77 145 83 7> 175 180 181 183 271 155 196 30 185 150 139 278 129 188 182 230 8> 106 266 187 197 186 253 90 112 161 252 184 189 274 194 154 9> 227 289 258 142 95 265 190 195 110 156 5 282 229 152 10> 268 199 277 261 272 269 205 236 162 275 11> 164 255 235 214 262 207 237 280 167 240 239 210 6 241 212 12> 204 250 243 256 7 171 246 213 244 48 279 13> 263 234 9 12 13 245 17 11 52 18 8 215 14> 20 15 16 23 10 21 202 286 216 93 158 203 208 15> 206 28 98 37 35 287 57 242 49 174 225 62 16> 217 102 32 36 107 39 178 53 63 198 251 58 71 43 290 141 17> 64 47 42 59 72 54 82 218 97 209 89 18> 99 211 192 46 151 87 219 100 103 78 80 163 85 88 260 65 108 19> 125 138 79 220 191 200 51 104 55 115 92 121 81 140 267 69 66 29 60 292 68 20> 228 273 224 114 33 201 276 283 143 128 226 135 67 21> 153 38 281 165 137 168 291 41 248 231 22> 45 173 176 70 50 177 179 232 288 169 193 23> 249 233 254 284 14 24 238 270 222 144 76 113 285 118 126 19

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	25	2925	1> 1 2 293 294 4 2> 3 27 105 106 295 83 101 56 26 221 3> 48 247 109 134 31 94 119 259 86 127 30 90 179 4> 82 77 112 52 271 57 76 93 181 117 71 62 5> 185 184 124 197 274 63 180 84 66 187 268 186 182 6> 73 277 190 67 282 69 266 261 195 91 183 229 7> 227 255 203 202 205 70 96 40 98 172 8> 44 206 262 207 49 210 230 240 235 188 199 198 237 9> 209 213 246 243 116 239 241 289 279 10> 242 166 214 248 157 11> 234 249 284 215 217 216 254 170 174 193 158 12> 256 154 287 244 263 97 34 238 257 178 194 13> 285 218 219 35 236 208 89 204 278 220 100 14> 42 104 250 288 99 224 87 108 283 245 296 15> 46 138 114 103 292 115 140 95 102 110 251 211 226 107 228 212 22 123 16> 143 169 148 191 145 155 297 113 146 153 165 118 168 149 147 17> 176 126 156 24 160 173 51 88 270 121 252 29 18> 55 6 258 33 177 128 81 38 5 61 19> 265 139 7 15 65 21 9 41 136 159 264 10 20> 23 45 13 28 14 20 11 18 32 260 253 162 21> 167 269 19 50 267 36 171 54 273 142 12 200 60 22> 39 17 152 276 43 58 164 47 68 281 291 23> 64 72 144 74 130 175 231 272 37 75 59 8 24> 78 129 79 189 275 85 125 131 233 192 280 80 133 201 25> 92 132 290 135 16 25 53 122 141 232 286 120 223 196 150 137 161 151 225 163 222

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	27	2684	1> 1 295 296 2 2> 3 294 293 4 109 26 3> 86 259 247 48 27 134 40 97 136 5 105 83 266 94 4> 44 271 278 31 34 49 82 29 101 283 5> 90 53 6 33 10 15 111 119 30 221 6> 223 9 116 20 58 122 11 7> 16 129 7 130 35 52 23 88 110 12 8> 141 162 151 167 123 171 99 28 32 100 17 148 36 146 14 225 9> 145 104 149 160 155 166 117 163 124 147 37 158 156 10> 39 108 57 170 106 154 19 159 87 120 11> 63 43 150 71 114 161 179 103 64 47 59 67 115 42 12> 77 181 46 189 197 187 202 184 70 182 13> 51 138 81 196 75 190 204 199 55 180 140 198 194 230 195 251 14> 143 207 203 232 206 210 200 252 208 229 256 15> 236 209 212 239 234 235 240 211 241 237 16> 243 242 263 245 289 205 214 246 258 213 153 17> 215 216 270 248 265 249 165 176 168 18> 173 272 275 254 177 217 244 172 112 19> 218 261 255 269 268 144 185 274 56 20> 60 61 219 262 68 65 277 69 73 74 282 93 220 191 21> 222 224 226 297 227 62 238 175 8 183 38 22> 66 41 228 285 193 72 174 250 186 231 23> 45 233 78 80 79 85 125 288 50 287 92 178 139 89 98 24> 192 102 142 152 201 107 253 260 121 127 22 25> 128 24 267 273 164 276 25 113 279 26> 118 286 126 281 131 54 280 257 284 292 76 95 13 188 27> 132 290 291 133 135 264 18 21 84 169 157 91 137 96

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	29	2519	1> 1 293 2 4 2> 296 295 3 27 105
			3> 30 26 56 31 61 259 34 247 60 48 179 221 65
			4> 22 278 223 266 271 52 172 230 232 69 289 40
			86 175 5> 74 84 91 185 73 75 181 227 134 76
			6> 186 197 188 195 196 136 189 44 229 94 202
			7> 139 77 142 235 29 199 237 205 207
			8> 152 204 198 212 250 294 121 97 5 203 6
			9> 208 256 234 206 238 7 18 187 13
			10> 12 14 15 285 9 19 17 210 251 182
			11> 23 28 213 37 35 214 93 164 25 8
			12> 39 32 167 36 43 190 215 180 128 42
			13> 47 216 217 218 240 98 263
			14> 219 220 102 130 283 274 183 68 241 282
			15> 123 129 144 148 166 268 147 154 277 141 149
			16> 224 133 170 158 270 226 261 145 131 151 228
			17> 132 140 255 262 135 163 137 269 231 146
			18> 246 138 155 117 159 160 174 193 243 253 11
			46 184 242 239 257
			19> 279 191 156 143 169 244 245 194 200 178 124
			20> 260 51 81 16 153 165 267 233 273
			21> 55 57 87 276 100 288 168 281 264 59 24 291
			22> 173 177 64 72 103 20 33 99 176 225 88
			23> 62 66 63 38 78 108 192 292 80 125 114 79 67
			24> 201 115 120 70 150 89 85 119 127 71 286 10
			41 25> 92 248 161 157 297 82 252
			26> 45 101 96 83 104 209
			27> 90 95 258 265 272 275 249 284 236 171 109
			50 107 254 28> 280 111 54 211 113 58 21 110 118
			29> 106 290 162 112 126 222 287 49 116 53 122

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	31	2358	1> 1 295 2 297 3 2> 296 294 4 83 3> 26 48 221 111 86 109 119 247 110 56 22 259 266 120 94 4> 61 150 24 65 293 69 25 5> 161 166 127 170 271 40 6> 174 158 274 44 130 97 49 29 123 134 7> 121 268 282 149 131 144 147 53 58 148 178 8> 146 101 5 93 277 27 105 60 261 9> 6 269 283 106 31 82 89 88 34 100 10> 172 179 104 99 10 35 112 59 133 132 84 11> 13 20 108 42 18 135 103 14 46 15 87 255 12> 19 51 137 21 138 55 12 17 81 11 73 76 13> 16 77 91 23 140 191 54 181 262 8 14> 186 182 185 187 200 184 143 180 190 188 195 153 199 115 15> 165 203 169 168 227 176 229 206 288 160 16> 235 278 28 30 183 33 7 175 74 154 237 17> 37 38 32 240 41 239 241 18> 238 243 285 245 234 193 36 256 156 159 196 242 19> 246 248 249 263 20> 284 244 90 68 162 208 223 230 270 98 21> 232 225 95 102 289 210 207 213 254 279 157 114 117 22> 107 113 118 204 257 286 212 214 126 167 205 52 252 197 23> 57 215 71 258 250 264 216 171 63 62 24> 67 217 265 218 70 219 220 66 189 222 173 25> 272 275 280 224 177 96 116 64 209 45 26> 50 290 122 226 72 211 228 287 292 27> 78 79 231 85 192 92 125 201 80 75 128 28> 233 136 202 9 194 198 124 129 39 145 29> 139 251 141 253 151 260 155 43 163 236 30> 142 47 152 267 164 31> 273 276 281 291

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	33	2144	1> 3 295 2 294 2> 4 296 297 3> 1 109 40 84 77 27 134 83 221 105 259 4> 94 5 119 86 93 48 44 136 127 56 5> 271 98 232 230 274 278 61 289 6> 277 57 139 60 68 266 268 52 142 65 22 7> 253 260 111 236 261 255 267 269 8> 273 152 164 240 239 247 276 110 9> 246 281 238 243 49 262 10> 293 248 249 11> 284 254 282 279 241 12> 286 227 223 234 245 157 24 53 13> 195 58 225 199 229 235 166 25 14> 54 202 251 256 154 204 26 237 31 15> 34 263 82 172 179 175 207 30 170 162 16> 74 174 287 158 250 270 88 35 17> 288 181 182 197 100 184 42 185 104 116 89 18> 189 186 108 292 122 183 178 19> 114 193 180 129 210 209 62 196 285 115 20> 66 213 211 214 73 63 203 242 99 6 75 21> 67 87 208 206 252 103 215 9 76 70 22> 258 216 217 12 14 13 15 8 23> 20 21 18 101 7 17 71 24> 218 219 11 19 23 10 106 188 25> 220 16 28 121 32 112 117 257 37 69 29 26> 264 283 124 36 123 38 149 148 159 33 146 145 224 265 27> 226 272 147 160 155 144 156 41 141 130 90 212 28> 275 280 131 132 45 50 133 128 95 290 151 29> 135 163 64 228 120 30> 150 231 233 161 39 194 72 102 167 190 59 46 137 31> 107 140 143 171 97 169 138 200 43 153 51 32> 191 55 78 47 192 81 165 79 125 113 244 198 33> 118 176 201 80 126 168 85 91 291 187 205 92 173 96 177 222

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	35	2001	1> 1 294 2 297 3 4 27 2> 86 97 56 93 134 40 109 48 61 119 65 3> 31 30 179 82 25 26 105 181 4> 34 196 127 69 88 84 180 182 73 5> 91 96 77 104 59 22 183 186 195 6> 199 203 64 157 72 189 205 7> 229 208 293 207 8> 210 6 60 120 184 5 74 204 9> 9 150 8 250 10 11 68 161 108 221 256 10> 274 114 15 263 270 213 259 209 235 11> 214 266 271 100 282 237 278 89 99 240 12> 246 277 230 103 243 13> 241 242 232 244 87 268 14> 255 261 238 269 223 24 35 136 225 78 15> 192 139 125 248 253 80 85 21 190 16> 260 92 111 188 197 249 17> 284 52 12 101 57 79 292 262 18> 106 112 17 144 71 62 142 130 19> 156 123 145 63 152 164 66 148 14 76 146 289 20> 149 67 155 167 19 70 116 160 23 98 94 20 21> 28 171 32 37 102 122 22> 107 236 39 174 288 75 187 252 42 23> 178 287 43 47 258 46 117 245 147 212 24> 158 51 257 124 264 265 81 154 215 129 115 25> 55 272 141 216 217 44 36 26> 218 151 275 219 280 220 163 49 283 201 83 113 159 27> 118 290 126 90 224 140 194 227 234 138 28> 191 200 226 95 53 228 231 295 206 29> 58 54 233 143 153 296 30> 169 165 202 211 31> 198 251 7 121 29 110 193 285 13 32> 18 162 38 33 176 168 128 33> 133 41 173 45 177 50 131 172 267 239 34> 273 132 279 286 166 175 276 135 170 247 35> 137 281 291 222 254 16 185

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	37	1980	1> 1 294 297 2 2> 3 4 221 56 247 278 109 26 48 30 94 52 86 283 3> 83 259 61 134 119 274 268 65 282 261 4> 136 110 269 296 266 120 5> 162 167 27 69 295 150 6> 31 161 105 171 93 40 98 277 271 90 255 7> 84 102 82 91 73 88 95 75 262 22 8> 100 89 59 104 44 97 24 107 99 223 9> 225 113 227 195 103 29 87 118 199 230 205 10> 203 232 229 235 237 289 33 207 208 11> 240 238 38 41 241 204 12> 243 45 242 50 234 210 236 239 285 213 13> 244 54 209 211 279 206 14> 286 121 77 57 250 256 15> 128 263 62 66 71 34 16> 130 270 144 123 129 139 179 108 17> 141 292 115 145 181 253 148 182 117 149 187 159 142 155 184 18> 190 180 183 193 152 197 196 185 188 5 160 19> 198 6 7 164 13 15 20 23 20> 28 37 18 11 16 21 39 10 21> 9 43 32 14 74 252 212 22> 19 47 258 265 166 260 23> 170 272 267 158 273 189 101 275 186 24> 276 281 106 280 58 151 172 8 112 25> 290 291 163 175 60 96 126 36 12 63 26> 68 17 67 246 131 114 146 27> 70 248 156 132 147 249 28> 254 135 284 133 35 214 76 29> 42 137 46 138 51 191 140 200 215 154 257 30> 55 81 143 216 169 264 49 245 124 31> 217 218 53 111 174 32> 219 288 220 222 224 116 287 178 153 33> 226 228 122 165 168 25 34> 176 173 293 64 177 35> 72 78 80 79 125 192 85 202 194 231 36> 201 92 251 233 127 37> 157

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	39	1880	1> 1 297 295 2> 294 296 2 3> 3 293 4> 4 221 86 40 83 94 111 5 223 5> 26 6 9 7 22 10 13 6> 18 30 34 31 44 134 49 7> 136 25 172 27 93 225 259 20 56 8> 60 266 271 227 195 230 203 199 8 105 247 9> 205 206 68 232 229 207 110 10> 212 236 274 120 202 251 239 235 256 11> 282 198 263 127 270 150 237 12> 277 283 157 234 238 209 161 214 268 13> 261 269 215 255 285 97 216 162 14> 262 167 24 240 84 73 15> 77 241 74 76 243 75 171 48 204 16> 246 242 52 245 174 250 17> 248 249 284 254 18> 244 178 14 101 175 19> 19 180 181 287 57 11 196 20> 197 182 189 252 62 16 186 187 21> 258 185 265 66 71 23 288 190 22> 272 69 28 37 275 39 280 208 91 188 23> 43 32 96 47 290 116 24> 122 129 36 141 151 217 90 25> 218 95 139 119 29 278 109 61 279 26> 65 286 253 219 220 142 33 260 121 163 289 27> 152 222 224 267 38 226 228 231 28> 41 233 273 45 50 128 29> 276 54 281 164 82 106 30> 58 88 64 112 100 89 59 184 35 98 31> 42 102 107 72 99 104 87 117 103 124 194 32> 46 257 108 115 114 138 51 113 78 140 292 123 33> 148 264 149 160 159 79 81 80 118 55 191 200 147 85 145 155 34> 126 158 92 143 125 146 192 291 130 35> 156 131 201 169 153 144 132 12 210 36> 213 17 135 137 133 165 170 176 53 37> 168 173 154 15 166 183 38> 177 21 193 63 179 211 39> 67 70

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	41	1774	1> 1 295 294 2> 296 2 3> 3 4 5 22 83 26 259 297 109 4> 134 27 136 90 31 56 25 24 266 247 5> 61 65 6 105 139 29 94 40 33 6> 9 10 14 19 23 28 221 7> 37 223 11 111 120 39 8> 16 127 157 130 131 253 9> 133 132 135 166 69 10> 74 158 271 15 32 21 12 123 11> 274 146 277 172 60 147 179 283 12> 181 189 196 188 185 195 34 13> 35 182 180 197 229 199 203 14> 202 227 205 282 30 145 15> 198 155 82 116 235 278 16> 88 89 237 100 236 7 97 207 17> 240 104 108 212 43 204 18> 114 238 243 115 241 214 47 215 186 225 59 19> 52 285 242 57 62 66 216 190 20> 71 77 64 63 84 20 292 21> 72 91 122 183 252 76 129 149 22> 160 96 141 151 208 245 142 68 73 23> 163 152 248 164 24> 249 167 254 258 8 42 121 137 48 25> 128 261 265 284 140 200 78 26> 192 80 79 143 171 174 272 85 92 125 27> 287 178 169 153 138 269 165 191 13 28> 18 293 99 29> 103 87 260 75 217 250 168 30> 256 218 98 184 117 17 36 156 268 31> 263 270 219 194 220 257 124 102 44 255 222 32> 107 264 224 49 226 113 262 118 193 210 33> 228 53 209 213 231 230 150 34> 289 58 54 161 211 233 232 46 35> 51 101 170 119 239 67 288 86 173 36> 55 106 177 81 93 112 70 279 126 251 95 275 37> 280 290 286 144 176 110 267 38 175 38> 41 273 162 187 148 201 154 234 39> 45 276 50 159 281 206 40> 291 244 41> 246

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	43	1697	1> 1 295 2 2> 294 3 4 109 297 3> 247 26 134 56 61 105 48 83 120 278 110 150 22 4> 65 24 127 283 25 161 69 5> 111 274 5 157 94 221 6> 116 122 129 223 225 7> 74 27 6 119 29 60 8> 31 34 121 179 172 33 10 9> 15 21 128 68 181 10> 180 189 187 185 182 188 11> 190 73 252 8 84 186 175 12 76 12> 258 265 23 20 35 14 195 203 13> 19 199 208 42 229 205 14> 236 210 202 207 91 196 272 15> 96 213 251 214 38 44 16> 234 215 41 45 275 9 197 256 17> 280 50 290 263 54 49 227 209 18> 58 64 211 53 72 86 238 162 19> 78 79 59 167 93 99 192 100 85 125 171 104 20> 87 89 108 292 98 201 80 285 174 239 115 279 21> 286 240 288 198 204 22> 243 241 245 246 287 130 123 23> 148 149 144 145 117 124 147 131 159 160 24> 154 257 155 132 158 77 146 212 25> 264 156 244 40 26> 255 166 141 90 259 102 268 27> 107 95 266 261 269 113 262 151 118 163 184 28> 194 248 92 29> 249 293 75 30> 284 254 206 103 82 52 31> 57 62 63 66 282 71 32> 67 70 97 7 17 11 178 33> 16 170 13 46 28 114 230 270 34> 18 138 289 51 140 191 32 200 143 35> 169 55 81 37 232 39 36 153 36> 165 168 176 43 30 216 250 235 37> 173 177 237 47 217 271 277 126 38> 218 88 135 136 242 101 39> 219 139 220 142 106 112 137 253 222 183 40> 193 260 267 152 224 164 226 133 41> 273 276 228 281 42> 291 231 29643> 233

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	45	1616	1> 6 2 5 297 4 83 259 2> 86 3 134 247 9 56 136
			7 139 3> 12 266 15 278 23 17 60 40
			4> 142 68 14 61 84 11 22 16 5> 77 96 24 91 65
			6> 20 74 101 75 29 106 7> 97 93 25 76 8> 112 296
			271 69 9> 30 274 98 33 48 26 253 38
			10> 282 260 102 21 94 107 11> 41 34 35 31 42 277
			172 12> 185 181 175 179 46
			13> 199 184 188 186 194 189 187 268
			14> 269 195 203 227 206 229 261
			15> 209 255 143 210 208 196 138 140
			16> 165 236 153 207 204 212 182
			17> 27 168 214 173 234 176 235 191 190
			18> 177 237 252 211 169 180 113 197 256 263
			19> 174 240 178 258 265 250 8
			20> 272 275 270 280 13 126 118 18
			21> 290 130 246 243 22> 241 242 245 131 244 123
			133 23> 147 158 156 145 170 146 149 148 166
			24> 160 159 129 135 248 25> 141 151 121 137
			26> 128 163 249 82 27> 254 238 88 132 262 285
			28> 230 100 104 108 221 29> 115 120 292 114 150
			64 59 232 239 127 30> 110 223 119 157 72
			31> 154 161 289 28 162 279 225 286
			32> 103 167 287 99 117 152 155 257 164
			33> 87 171 264 37 32 144 205 34> 10 215 183 284
			73 78 192 125 35> 79 85 193 44 39 124 92
			36> 43 47 198 49 200 216 109 267 273
			37> 201 19 288 105 45 276
			38> 57 53 50 217 52 54 39> 281 111 291 63
			40> 58 62 66 71 116 80 41> 67 70 218 122 202
			42> 219 293 220 90 43> 1 251 283 36 89 51 213
			224 81 44> 226 55 95 295 222 45> 228 231 294 233

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	47	1544	1> 1 297 2 2> 3 4 40 26 109 84 259 76 94
			3> 101 247 134 48 86 22 97 56 266
			4> 136 139 77 221 52 223 5> 225 227 27 278 74
			283 6> 31 44 30 253 82 75
			7> 172 88 29 121 274 8> 100 277 268 59 73 91 175
			119 61 9> 104 261 180 108 269 181 188
			10> 184 115 252 197 96 255 114 196 65 194
			11> 199 262 69 183 189 193 89
			12> 25 296 182 99 13> 190 103 87 49 106 230
			14> 232 289 112 236 53 198 271 15> 24 34 117 202
			33 142 16> 251 38 123 41 35 257 152 147 149 256
			17> 145 45 146 42 263 155 264 160 156
			18> 166 46 50 154 19> 54 170 234 148 51 58 258
			204 20> 159 265 207 250 55 64 164 239 120
			21> 272 279 72 167 275 171 210 212 150 213
			22> 209 161 211 214 286 292 5 81
			23> 6 128 195 138 186 24> 7 10 8 203 208 9
			25> 206 205 12 14 11 140 191 200
			26> 16 23 28 143 37 15 78
			27> 192 21 169 153 165 168 79 80 125
			28> 85 173 92 39 17 177 176 201 32
			29> 36 294 111 30> 116 20 280 43
			31> 47 122 290 229 215 32> 129 141 235 151 216
			163 185 83 33> 237 90 217 105 158 238 270
			34> 95 218 219 220 285 93 124 130
			35> 144 222 224 226 127 60
			36> 68 157 228 131 37> 133 135 231 132 13 260
			110 38> 137 162 233 267 273 18 276 187
			39> 281 291 179 19 98 40> 102 107 113 295
			41> 118 293 42> 126 240 174 43> 241 287 243 242
			288 245 178 44> 246 57 45> 248 63 249
			46> 62 66 284 71 67 254 47> 70 244 282

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	49	1479	1> 1 297 2 3 2> 295 4 259 109 3> 247 48 26 134 111 86 94 266 4> 116 136 122 56 22 5> 271 61 5 24 120 119 30 150 6> 161 166 158 230 7> 170 34 274 277 6 283 8> 282 227 9 8 9> 11 16 23 14 19 289 10> 293 221 11> 232 15 28 110 12 195 12> 162 21 203 236 206 208 205 13> 239 207 279 210 209 32 17 278 14> 36 211 39 43 212 286 167 15> 47 213 214 29 13 171 60 16> 44 18 234 215 216 256 204 17> 217 238 202 251 31 18> 218 285 198 179 172 181 19> 187 175 183 188 196 193 182 82 190 20> 88 186 240 268 21> 100 59 243 104 99 103 108 241 245 68 22> 114 292 246 242 115 93 23> 98 248 249 87 24> 254 284 102 64 185 25> 72 7 49 197 263 223 26> 78 79 125 225 270 53 80 189 27> 130 58 144 123 147 146 129 73 28> 149 97 145 160 54 156 148 159 29> 155 84 91 141 151 180 219 30> 163 75 96 220 222 224 25 27 31> 226 76 40 89 117 20 33 65 32> 124 257 69 38 264 192 33> 41 77 201 121 10 34> 45 128 50 37 35> 228 74 252 36> 231 296 139 37> 253 233 237 101 154 261 142 38> 269 260 152 106 112 255 39> 164 235 267 262 294 184 105 174 40> 288 194 287 273 178 52 41> 276 281 291 57 42> 71 63 67 62 127 43> 70 66 157 229 83 44> 90 244 35 45> 42 95 46 131 107 250 199 46> 138 140 133 51 132 113 200 47> 135 191 143 153 137 169 55 165 81 48> 168 176 118 85 173 258 265 272 49> 275 92 126 280 177 290

ตาราง 18 ผลการจัดสมดุลสายการประกอบด้วยโปรแกรม DE2 ของปัญหาขนาดใหญ่ (ต่อ)

Problem	*m*	*c*	Result
Scholl-297	50	1421	1> 1 297 2 83 4 259 2> 221 94 101 3 106 97 24 3> 22 29 33 223 5 111 44 4> 121 56 86 112 117 5> 124 6 40 84 7 6> 15 77 73 130 7> 8 16 257 264 11 129 123 141 8> 147 145 128 149 160 12 76 155 75 9> 17 131 14 10> 38 133 20 247 170 11> 19 132 294 278 134 136 12> 174 266 91 287 139 146 13> 41 296 156 14> 45 23 109 142 26 148 15> 119 120 152 127 25 10 178 181 16> 28 150 110 185 182 188 17> 186 37 162 195 196 227 203 18> 167 32 164 197 39 184 194 199 19> 171 208 210 213 202 183 193 20> 209 207 212 36 30 251 205 21> 204 206 180 230 187 252 232 22> 27 236 258 256 234 23> 263 21 96 61 65 24> 270 31 172 157 239 240 50 271 25> 179 175 279 286 238 82 26> 241 34 253 35 27> 242 243 89 260 151 190 88 28> 42 103 163 267 100 104 273 69 99 29> 108 59 57 276 115 292 48 52 30> 64 72 78 192 62 31> 125 66 246 63 201 80 67 85 32> 114 71 248 79 33> 9 249 87 34> 284 74 92 254 159 261 35> 98 102 107 113 285 269 70 36> 118 126 268 255 116 250 262 37> 293 274 282 38> 277 122 54 189 39> 166 40> 244 265 41> 154 281 49 42> 272 291 58 93 275 225 43> 295 280 53 289 161 13 18 44> 144 90 95 45> 215 46 214 288 198 46> 138 217 216 51 191 140 143 153 47> 81 165 200 218 290 43 48> 283 219 168 55 176 220 229 158 169 47 49> 173 224 177 228 226 235 50> 222 237 231 60 135 233 68 211 245 137 105

ประวัติผู้เขียน

ชื่อ นายกฤต จันทรสมัย

วันเกิด วันที่ 21 เดือนธันวาคม พ.ศ. 2529

สถานที่เกิด โรงพยาบาลสรรพสิทธิประสงค์ อำเภอเมือง จังหวัดอุบลราชธานี

สถานที่อยู่ปัจจุบัน บ้านเลขที่ 8 ซอยชวาลานอก 1 ถนนชวาลานอก อำเภอเมือง จังหวัดอุบลราชธานี รหัสไปรษณีย์ 34000

ประวัติการศึกษา พ.ศ. 2548 มัธยมศึกษาตอนปลาย โรงเรียนเบ็ญจะมะมหาราช อำเภอเมือง จังหวัดอุบลราชธานี

พ.ศ. 2551 ปริญญาวิศวกรรมศาสตรบัณฑิต (วศ.บ.)
สาขาวิชาวิศวกรรมอุตสาหการ มหาวิทยาลัยอุบลราชธานี

พ.ศ. 2553 ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วศ.ม.)
สาขาวิชาวิศวกรรมอุตสาหการ มหาวิทยาลัยอุบลราชธานี

พ.ศ. 2564 ปริญญาปรัชญาดุษฎีบัณฑิต (ปร.ด.)
สาขาวิชาวิศวกรรมเครื่องกล มหาวิทยาลัยมหาสารคาม

พหูน ปณ ฑิโต ชีเว