การจัดสมดุลสายการประกอบแบบตัวยูโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่าง

พูนธนะ ศรีสระคู

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาปรัชญาดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยอุบลราชธานี
ปีการศึกษา 2561

ลิขสิทธิ์เป็นของมหาวิทยาลัยอุบลราชธานี

U-SHAPED ASSEMBLY LINE BALANCING BY USING DIFFERENTIAL EVOLUTION ALGORITHM

POONTHANA SRESRACOO

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
MAJOR IN INDUSTRIAL ENGINEERING
FACULTY OF ENGINEERING
UBON RATCHATHANI UNIVERSITY
ACADEMIC YEAR 2018

COPYRIGHT OF UBON RATCHATHANI UNIVERSITY

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยความกรุณาและความช่วยเหลือเป็นอย่างดีจาก รองศาสตราจารย์ ดร.นุชสรา เกรียงกรกฎ และ รองศาสตราจารย์ ดร.ปรีชา เกรียงกรกฎ อาจารย์ที่ปรึกษาร่วมวิทยานิพนธ์ทั้งสองท่าน ผู้ซึ่งกรุณาให้ความรู้ คำแนะนำข้อคิดเห็น ให้คำปรึกษา เป็นอย่างดีตลอดมา จึงขอกราบขอบพระคุณท่านเป็นอย่างสูง นอกจากนี้ขอขอบพระคุณอาจารย์ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยอุบลราชธานี ทุกท่านที่ได้ให้ความรู้ และให้โอกาสเข้ารับการศึกษา ณ มหาวิทยาลัยแห่งนี้

ขอขอบพระคุณท่านคณะกรรมการสอบวิทยานิพนธ์ทุกท่านที่สละเวลาอันมีค่าของท่านมาเป็นคณะกรรมการสอบวิทยานิพนธ์ ได้แก่ ผู้ช่วยศาสตราจารย์ ดร.สมบัติ สินธุเชาวน์ ผู้ช่วยศาสตราจารย์ ดร.นลิน เพียรทอง และรองศาสตราจารย์ ดร.เกียรติศักดิ์ ศรีประทีป ที่ช่วยกรุณาตรวจสอบวิทยานิพนธ์และให้คำแนะนำและขอขอบพระคุณมหาวิทยาลัยราชภัฏบุรีรัมย์ที่สนับสนุนทุนการศึกษาต่อในระดับดุษฎีบัณฑิตรวมทั้งขอขอบพระคุณเจ้าหน้าที่ของทางคณะวิศวกรรมศาสตร์ มหาวิทยาลัยอุบลราชธานี ทุกท่านที่คอยให้กำลังใจและให้คำแนะนำในการทำเอกสารวิทยานิพนธ์ในครั้งนี้

สุดท้ายนี้ ผู้วิจัยขอกราบขอบพระคุณครอบครัวผู้เบื้องหลังคอยสนับสนุนมาโดยตลอด ตลอดจนทุกคนที่คอยให้กำลังใจแก่ผู้วิจัยเสมอมา และนอกจากนี้ขอขอบคุณบุคคลที่เกี่ยวข้องทุกท่านที่ไม่ได้กล่าวนามที่คอยช่วยเหลือและเป็นกำลังใจจนประสบความสำเร็จ ผู้วิจัยขอขอบพระคุณทุกท่านไว้ ณ โอกาสนี้ด้วย

พูนธนะ ศรีสระคู
ผู้วิจัย

บทคัดย่อ

เรื่อง	:	การจัดสมดุลสายการประกอบแบบตัวยูโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่าง
ผู้วิจัย	:	พูนธนะ ศรีสระคู
ชื่อปริญญา	:	ปรัชญาดุษฎีบัณฑิต
สาขาวิชา	:	วิศวกรรมอุตสาหการ
อาจารย์ที่ปรึกษา	:	รองศาสตราจารย์ ดร.นุชสรา เกรียงกรกฎ
อาจารย์ที่ปรึกษาร่วม	:	รองศาสตราจารย์ ดร.ปรีชา เกรียงกรกฎ
คำสำคัญ	:	การจัดสมดุลสายการประกอบแบบตัวยู, วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน, วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ, ค่าคำตอบที่ดีที่สุด

งานวิจัยนี้มีวัตถุประสงค์ คือ เพื่อพัฒนาวิธีการเมตาฮิวริสติก โดยใช้วิธีการวิวัฒนาการโดยใช้ผลต่าง สำหรับการแก้ปัญหาและหาคำตอบที่ดีที่สุดในการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 ที่มีการผลิตสินค้าชนิดเดียว และทราบค่ารอบเวลาการผลิตที่แน่นอน โดยมีเป้าหมายเพื่อหาจำนวนสถานีงานที่น้อยสุด จากนั้นทำการเปรียบเทียบผลกับงานวิจัยอื่น ๆ ที่ผ่านมา

สำหรับการจัดสมดุลสายการประกอบแบบตัวยู ได้ทำการทดลองในปัญหาตัวอย่าง 2 กลุ่ม คือ 1) ปัญหาตัวยูที่มีขนาดกลาง (21-45 ขั้นงาน) ที่เป็นชุดโจทย์ปัญหาในการทดลองโดยวิธีระบบมดแบบแม็กมินไม่มีโลคอลเสิร์ช ของ นุชสรา เกรียงกรกฎ (2550) ทั้งหมด 25 instances จากผลการทดลองพบว่า วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ ให้ค่าคำตอบที่ดีกว่า เมื่อเปรียบเทียบกับวิธีเมตาฮิวริสติกแบบอื่น ๆ คือสามารถหาคำตอบที่ดีที่สุด จำนวน 25 คำตอบ จากทั้งหมด 25 instances (เวลาเฉลี่ยที่ใช้ในการคำนวณอยู่ระหว่าง 0.09 -1.56 วินาที/instance) ในขณะที่วิธีระบบมดแบบแม็กมินไม่มีโลคอลเสิร์ช สามารถหาคำตอบที่ดีที่สุดได้เพียง 23 คำตอบ จากทั้งหมด 25 instances และ 2) ปัญหาขนาดใหญ่ (75-297 ขั้นงาน 62 instances) จึงทำการเปรียบเทียบวิธีระบบมดแบบแม็กมินไม่มีโลคอลเสิร์ช จากผลการทดลองพบว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบให้ค่าคำตอบที่ดีกว่า และวิธี DE/best/2 ร่วมกับ Exponential Crossover 1 ให้ค่าคำตอบที่ดีที่สุดและใช้เวลาในการค้นหาคำตอบน้อยที่สุด เมื่อเปรียบเทียบกับวิธีเมตาฮิวริสติกแบบอื่น ๆ คือสามารถหาคำตอบที่ดีที่สุด จำนวน 36 คำตอบ จากทั้งหมด 62 instances (เวลาเฉลี่ยที่ใช้ในการคำนวณอยู่ระหว่าง 4.88 - 5.65 วินาที/instance) ในขณะที่วิธีระบบมดแบบแม็กมินไม่มีโลคอลเสิร์ช สามารถหาคำตอบที่ดีที่สุดได้เพียง 35 คำตอบ จากทั้งหมด 62 instances

ดังนั้น สรุปได้ว่า วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบให้ค่าคำตอบที่ดีกว่า และเป็นวิธีที่มีประสิทธิภาพเมื่อเปรียบเทียบกับวิธีเมตาฮิวริสติกอื่น ๆ ที่สามารถใช้ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูได้

ABSTRACT

TITLE : U-SHAPED ASSEMBLY LINE BALANCING BY USING DIFFERENTRIAL EVOLUTION ALGORITHM
AUTHOR : POONTANA SRESRACOO
DEGREE : DOCTOR OF PHILOSOPHY
MAJOR : INDUSTRIAL ENGINEERING
ADVISOR : ASSOC. PROF. NUCHSARA KRIENGKORAKOT, Ph.D.
CO-ADVISOR : ASSOC. PROF. PREECHA KRIENGKORAKOT, Ph.D.
KEYWORDS : U-SHAPED ASSEMBLY LINE BALANCING, BASIC DIFFERENTIAL EVOLUTION ALGORITHM, IMPROVED DIFFERENTIAL EVOLUTION ALGORITHM, OPTIMAL SOLUTION

This research aimed to develop a metaheuristic approach for solving U-shaped Assembly Balancing Problem (UALBP) Type 1 (given c, min m). The proposed algorithm was developed by applying basic differential evolution algorithm and improvement of differential evolution algorithm. The development of algorithms had been evaluated and would be compared to other metaheuristics from literature review.

For UALBP, the Differential Evolution Algorithm (DE) methods were tested with 2 data sets, 1) in medium scale problem (21-45 tasks, 25 instances, data set of Nuchsara et.al MMAS (no local search) method, Nuchsara K. (2007)) whose experimental results showed that the basic differential evolution algorithm and improved differential evolution algorithm were able to identify better solutions than other metaheuristic; the results indicated that optimal was 25 from 25 instances (average cal. time of each method were between 0.09 – 1.56 sec./instance) whereas MMAS (no local search) method reached optimal only 23 out of 23 instances and 2) in large scale problem (21-45 tasks, 62 instances) where the MMAS (no local search) method was compared with priority rule metaheuristic; the result showed that the basic differential evolution algorithm and improved differential evolution algorithm indicated good solution. The DE/best/2 and Exponential Crossover 1 algorithm gave the optimal solution and shorter time than other methods and was able to identify better solutions. The results revealed optimal solutions were 36 from 62 instances (average cal. time of each method was between 4.88 - 5.65 sec./instance) whereas MMAS (no local search) method reached optimal only for 35 from 62 instances.

The results showed that the proposed DE performed more effectively than other metaheuristics in UALBP.

สารบัญ

	หน้า
กิตติกรรมประกาศ	**ก**
บทคัดย่อภาษาไทย	**ข**
บทคัดย่อภาษาอังกฤษ	**ค**
สารบัญ	**ง**
สารบัญตาราง	**ฉ**
สารบัญภาพ	**ฌ**
คำอธิบายสัญลักษณ์	**ญ**
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการวิจัย	2
1.3 สมมติฐานของการวิจัย	2
1.4 ขอบเขตของการวิจัย	2
1.5 ขั้นตอนการดำเนินการวิจัย	3
1.6 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย	3
บทที่ 2 ทฤษฎี และงานวิจัยที่เกี่ยวข้อง	
2.1 แนวคิดพื้นฐาน	4
2.2 หลักการและทฤษฎีที่เกี่ยวข้อง	7
2.3 งานวิจัยที่เกี่ยวข้อง	22
บทที่ 3 วิธีวิวัฒนาการโดยใช้ผลต่างสำหรับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู	
3.1 การจัดสมดุลสายการประกอบแบบตัวยู	31
3.2 การแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง	31
3.3 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน	34
3.4 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ	38
บทที่ 4 วิเคราะห์ผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู	
4.1 การพัฒนาโปรแกรมการจัดสมดุลสายการประกอบแบบตัวยู	54
4.2 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน	54

สารบัญ (ต่อ)

หน้า

4.3 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 1 Position 56
4.4 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/ rand-to-best/1 ร่วมกับ Binomial Crossover 57
4.5 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/ rand-to-best/1 ร่วมกับ Binomial Crossover 59
4.6 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 Position 61
4.7 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 2 Position 62
4.8 ผลการเปรียบเทียบวิธีวิวัฒนาการโดยใช้ผลต่างเปรียบเทียบกับวิธีเมตาฮิวริสติกวิธีอื่น ๆ 66
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ
5.1 **สรุปผลการวิจัย** 74
5.2 ข้อเสนอแนะและแนวทางวิจัยในอนาคต 77
เอกสารอ้างอิง **78**
ภาคผนวก
ก Pseudo Code และแผนผังขั้นตอนการทำงานของโปรแกรมการจัดสมดุลสายการประกอบแบบตัวยู 85
ข ชุดโจทย์ปัญหาตัวอย่าง (Data set) และข้อมูลที่เกี่ยวข้อง 92
ค อธิบายขั้นตอนการกรอกข้อมูลและการแสดงผลของโปรแกรมกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู 109
ง ผลการเปรียบเทียบวิธีการที่พัฒนากับวิธีการอื่น ๆ ในปัญหาการจัดสมดุลสายการประกอบแบบตัวยู 116
จ จำนวนสถานีงานของแต่ละปัญหา ปัญหาขนาดกลางและปัญหาขนาดใหญ่ 123
ประวัติผู้วิจัย **167**

สารบัญตาราง

ตารางที่		หน้า
2.1	สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู	25
3.1	ผลการสร้างคำตอบเริ่มต้นของปัญหา โดยกำหนด NP=5 เวกเตอร์	35
3.2	ผลการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี DE/rand/1	36
3.3	ผลการ Binomial Crossover เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี DE/rand/1	37
3.4	ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.3	37
3.5	ผลการ Exponential Crossover 1 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี “DE/rand/1”	40
3.6	ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.5	40
3.7	ผลการ Exponential Crossover 2 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี “DE/rand/1”	42
3.8	ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.7	42
3.9	ผลการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี “DE/rand-to-best/1”	43
3.10	ผลการ Binomial Crossover เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี “DE/rand-to-best/1”	44
3.11	ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.10	45
3.12	ผลการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี “DE/best/2”	46
3.13	ผลการ Exponential Crossover 1 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี “DE/best/2”	47
3.14	ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.13	47
3.15	ผลการ Exponential Crossover 2 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี “DE/best/2”	49
3.16	ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.15	49
3.17	ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/rand/1” ร่วมกับ Binomial Crossover	50
3.18	ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 1 Position	50
3.19	ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 2 Position	51
3.20	ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/ rand-to-best/1” ร่วมกับ Binomial Crossover	51
3.21	ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 1 Position	51
3.22	ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 2 Position	52

สารบัญตาราง (ต่อ)

ตารางที่		หน้า
3.23	ผลการจัดสถานีงานการสร้างคำตอบเริ่มต้น (Initial population) เริ่มต้นด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง	52
3.24	สรุปใช้วิธีวิวัฒนาการโดยใช้ผลต่างกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยูแต่ละวิธี	53
4.1	ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน DE/rand/1 ร่วมกับ Binomial Crossover กับปัญหา Sawyer และ Arcus 1	55
4.2	ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1	56
4.3	ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 2 Position กับปัญหา Sawyer และ Arcus 1	58
4.4	ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand-to-best/1 ร่วมกับ Binomial Crossover กับปัญหา Sawyer และ Arcus 1	60
4.5	ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1	61
4.6	ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 2 Position กับปัญหา Sawyer และ Arcus 1	63
4.7	การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบมด	66
4.8	การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search)	68
4.9	การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search)	70
4.10	การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search)	71
5.1	การสรุปผลข้อมูลจากตารางที่ ง.2 และตารางที่ ง.3 Data Set ของ วิธี Integer Programming solution, result from Urban (1998) ในตัวอย่างปัญหา UALBP-1 ที่มีขนาดกลาง	75

สารบัญตาราง (ต่อ)

ตารางที่ **หน้า**

5.2 การสรุปผลข้อมูลจาก ตารางที่ ง.4 ตารางที่ ง.5 และตารางที่ ง.6 Data Set ของวิธี Optimal solution (1991) ในตัวอย่างปัญหา UALBP-1 ที่มีขนาดใหญ่ 76

สารบัญภาพ

ภาพที่		หน้า
2.1	สายการประกอบที่แบ่งกระบวนการผลิตตามจำนวนของชนิดของสินค้า	5
2.2	ระบบสายการประกอบเครื่องปรับอากาศที่เป็นแบบสินค้าชนิดเดียวหรือสินค้าหลายชนิด	6
2.3	ระบบสายการประกอบรถยนต์แบบสินค้าแบบผสม	6
2.4	ตัวอย่างแผนภาพลำดับขั้นตอนงาน	8
2.5	ประเภทของปัญหาการจัดสมดุลสายการประกอบ	9
2.6	ตัวอย่างการจัดงานที่อยู่ในระดับเดียวกัน	11
2.7	แผนผังลำดับขั้นตอนงาน	12
2.8	ผลของการจัดสมดุลงานแบบตัวยูในปัญหาของ Scholl and Klein (1999) ตัวอย่างการจัดสมดุลงานและการคำนวณประสิทธิภาพของสายการผลิต	13
2.9	แผนผังลำดับขั้นตอนงาน Jaeschke problem	13
2.10	ตัวอย่างการการคำนวณประสิทธิภาพของสายการผลิต Jaeschke problem	14
2.11	การจัดสายการผลิตแบบตัวยู	15
2.12	การจัดสมดุลสายการประกอบแบบตัวยู ของการประกอบ Honda Automobile	15
2.13	ขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง	17
2.14	ลำดับขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง	17
2.15	การค้นหา Mutant Vector ของฟังก์ชั่น 2 ตัวแปร	19
3.1	กระบวนการวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน	33
3.2	แผนผังลำดับขั้นตอนงาน Bowman problem	34
3.3	กระบวนการวิธีวิวัฒนาการโดยใช้ผลต่างปรับปรุงคำตอบ	38
4.1	การเปรียบเทียบวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบที่ใช้เวลาในการค้นหาคำตอบที่ดีที่สุดกับปัญหา Sawyer (ที่มี 30 ขั้นงาน 8 Instances)	64
4.2	การเปรียบเทียบวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบที่ใช้เวลาในการค้นหาคำตอบที่ดีที่สุดกับปัญหา Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances)	65

คำอธิบายสัญลักษณ์

สัญลักษณ์	ความหมาย
m	จำนวนสถานีงาน
c	รอบเวลาการผลิต
V,F	เซตการแบ่งกลุ่ม
n	จำนวนงานย่อย
U_i^p	เป็นเซตของงานที่จะต้องทำให้เสร็จก่อนงาน i
U_i^s	เป็นเซตของงานที่จะต้องทำหลังจากที่ทำงาน i เสร็จ
T	เซตของเวลางานใด ๆ
t_j	เวลาที่ใช้ในแต่ละงานย่อย
St_j	เวลาที่ใช้ในแต่ละสถานีงาน
P	เซตเงื่อนไขของลำดับงานก่อน-หลัง
E	ประสิทธิภาพสายการผลิต
D	การสูญเสียความสมดุล
I	เวลาสูญเปล่า
SALBP	ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรง
GALBP	ปัญหาการจัดสมดุลสายการประกอบแบบทั่วไป
UALBP	ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู
DE	วิธีวิวัฒนาการโดยใช้ผลต่าง
F	แฟคเตอร์ในการขยายผลต่าง
CR	อัตราการแลกเปลี่ยนค่าในพิกัด
NP	จำนวนขั้นตอนงาน

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในสภาพปัจจุบันสภาวะของธุรกิจอุตสาหกรรมทั่วโลก มีการตื่นตัวเพิ่มมากขึ้นมีเทคโนโลยีนำสมัยเข้ามาพัฒนาระบบการผลิตและการบริหารจัดการเพื่อที่จะให้บริษัทของตนเองอยู่รอดและแข็งแกร่งเมื่อเปรียบเทียบกับคู่แข่งรายอื่น ๆ ทำให้บริษัทของตนเองต้องมีปรับตัวเพื่อตอบสนองความต้องการของลูกค้าที่มีมาอย่างต่อเนื่องและหลากหลาย ถ้าบริษัทไหนมีการตอบสนองต่อลูกค้าได้ดีและต่อเนื่องก็จะคงอยู่ได้นานและมั่นคงมากยิ่งขึ้น ดังนั้นการบริการจัดการที่ดีจึงมีความสำคัญต่อบริษัท บริษัทจึงได้มีการพัฒนาปรับปรุงผลิตภัณฑ์อย่างต่อเนื่อง โดยการลดต้นทุนและเพิ่มคุณภาพของสินค้า โดยเฉพาะในระบบการผลิตมีปัญหาการจัดสมดุลสายการผลิต (Assembly line Balancing) โดยมีการเปลี่ยนแปลงรูปแบบการจัดสายการผลิต (Lay out) เป็นรูปแบบตัวยู (U-shaped production lines or U-lines) ร่วมด้วย เพื่อที่จะให้การจัดสมดุลสายการประกอบมีประสิทธิภาพซึ่งเมื่อเปรียบเทียบกับสายการผลิตแบบเดิม คือแบบเส้นตรง (Straight lines) พบว่าจะมีข้อดีมากกว่าทั้งในเรื่องของการจัดสมดุลสายการผลิต การปรับปรุงในเรื่องของการมองเห็นการทำงานของพนักงาน และการติดต่อสื่อสารจะทำได้ดีกว่า จำนวนสถานีงานที่น้อยกว่า มีความยืดหยุ่นสูง มีรอบการเดินทางในการทำงานที่สั้นกว่า และง่ายต่อการขนถ่ายวัสดุ เป็นต้น สำหรับงานวิจัยที่เกี่ยวข้องกับการจัดสมดุลสายการประกอบแบบเส้นตรงที่มีผลิตภัณฑ์รุ่นเดียว (Single model) และผลิตภัณฑ์แบบผสม (Mixed model) นั้นได้มีการตีพิมพ์ครั้งแรกเมื่อปี ค.ศ. 1955 ในขณะที่งานวิจัยการจัดสมดุลสายการประกอบแบบตัวยูเพิ่งจะมีการศึกษากันอย่างจริงจัง ตั้งแต่ปี ค.ศ. 1994 เป็นต้นมา ดังนั้น งานวิจัยทางด้านนี้ยังมีอีกหลายด้านที่น่าสนใจในการศึกษาวิจัยต่อไปในอนาคตได้ (Chen, 2003)

จากการทบทวนงานวิจัยที่เกี่ยวข้อง กับปัญหาสายการประกอบ ซึ่งจัดเป็นปัญหาทาง Combinatorial Optimization แบบ NP-hard (McMullen and Frazier, 1998) ที่มีความยุ่งยากซับซ้อนในการแก้ปัญหา ซึ่งถ้าใช้วิธีทางคณิตศาสตร์แบบ Exact methods เพื่อหาคำตอบที่เหมาะสมที่สุด (Optimal solution) ต้องใช้เวลาคำนวณค่อนข้างมาก และสิ้นเปลืองค่าใช้จ่าย โดยเฉพาะปัญหาขนาดใหญ่ที่มีตัวแปรและข้อจำกัดมากขึ้น ดังนั้นวิธีการฮิวริสติกหลาย ๆ วิธี ที่ให้ค่าผลลัพธ์ที่ดี (Good solution) จึงพัฒนาขึ้น (Suresh และ Sahu, 1994) ซึ่งเป็นระยะเวลา 10 ปีที่ผ่านมาวิธีการฮิวริสติกที่พัฒนาขึ้นที่เรียกว่าเมตาฮิวริสติก (Metaheuristic methods) ก็ยังเป็นวิธีที่นักวิจัยให้ความสนใจเพื่อนำไปใช้ในการหาผลลัพธ์ทั้งปัญหาในการจัดสมดุลสายการประกอบแบบทั่วไป (General Assembly Line Balancing Problem: GALBP) และแบบง่าย (Simple Assembly Line Balancing Problem: SALBP) (Scholl and Becker, 2006)

จากการทบทวนงานวิจัยที่เกี่ยวกับปัญหาการจัดสมดุลสายการประกอบ ซึ่งจัดว่าเป็นปัญหาคลาสสิคที่มีนักวิจัยให้ความสนใจทำวิจัยทางด้านนี้มาเป็นเวลานาน คือตั้งแต่ ค.ศ. 1955 เป็นต้นมา โดยนักวิจัยได้มีการพัฒนาวิธีการทั้งวิธีการทางคณิตศาสตร์ (Exact methods) วิธีฮิวริสติกแบบต่าง ๆ

รวมทั้งวิธีเมตาฮิวริสติกแบบต่าง ๆ ซึ่งพบว่า การพัฒนาวิธีการเมตาฮิวริสติกเพื่อนำมาใช้แก้ปัญหาทางด้านนี้ยังไม่มีมากนัก ดังนั้น จึงเป็นเรื่องที่น่าสนใจที่จะทำการพัฒนาวิธีการทางเมตาฮิวริสติกมาประยุกต์ใช้สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบเพื่อเพิ่มประสิทธิภาพในการหาคำตอบของปัญหานี้

ดังนั้น ในการทำวิทยานิพนธ์นี้ จะเป็นการศึกษาการจัดสมดุลสายการประกอบแบบตัวยู ที่มีการผลิตสินค้าชนิดเดียว ที่ทราบค่ารอบการผลิต (Cycle time: c) แน่นอน โดยมีเป้าหมายเพื่อหาจำนวนสถานีงานที่น้อยที่สุด (m) ที่น้อยที่สุด ซึ่งปัญหาในลักษณะนี้เป็นปัญหาการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 U-line Assembly Balancing Problem Type I: (UALBP-1) ที่มีการพัฒนาวิธีการเมตาฮิวริสติกสมัยใหม่ คือการค้นหาคำตอบด้วยวิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithm: DE) เพื่อให้ได้คำตอบที่ดี (Good solutions) หรือ คำตอบที่เหมาะสมที่สุด

1.2 วัตถุประสงค์ของการวิจัย

เพื่อพัฒนาวิธีการเมตาฮิวริสติก โดยใช้วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithm: DE) สำหรับการแก้ปัญหาและหาคำตอบที่ดีที่สุดในการจัดสมดุลสายการประกอบ แบบตัวยู U-line Assembly Balancing Problem Type I: (UALBP-1)

1.3 สมมติฐานของการวิจัย

1.3.1 สายการผลิตที่พิจารณาจะเป็นสายการผลิตแบบต่อเนื่อง สินค้าที่ผลิตเป็นสินค้าชนิดเดียวกันที่มีหลายขั้นงาน

1.3.2 รอบเวลาการผลิต (Cycle time) คงที่ โดยที่อัตราการปล่อยชิ้นงาน จะมีค่าเท่ากับรอบเวลาการผลิต

1.3.3 เวลาการทำงานจะทราบค่าเวลาแน่นอน

1.3.4 ไม่มีข้อจำกัดของการทำงาน นอกจากเงื่อนไขของลำดับงานก่อน-หลัง

1.3.5 ไม่พิจารณาการจัดสถานีงานแบบขนานกัน (Parallel station)

1.4 ขอบเขตของการวิจัย

ปัญหาที่ศึกษาจะเป็นปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ที่เป็นปัญหาประเภทที่ 1 คือ กรณีทราบค่ารอบเวลาการผลิต (c) และเป้าหมายเพื่อคำนวณหาจำนวนสถานีงานที่น้อยที่สุด โดยพัฒนาหลักการเมตาฮิวริสติก คือ วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithm: DE) เพื่อช่วยหาคำตอบที่ดีที่สุด หรือคำตอบที่ใกล้เคียงจากนั้นทำการเปรียบเทียบผลกับงานวิจัยอื่น ๆ ที่ผ่านมา โดยผู้วิจัยได้ศึกษากับชุดปัญหาขนาดกลาง (25-45 ขั้นงาน) และชุดปัญหาขนาดใหญ่ (75-297 ขั้นงาน) โดยทำการพัฒนาโปรแกรมเพื่อความเหมาะสมในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างด้วยโปรแกรมภาษา Java

1.5 ขั้นตอนการดำเนินการวิจัย

โดยมีขั้นตอนการดำเนินงานดังนี้

1.5.1 ศึกษางานวิจัยและค้นคว้าข้อมูลที่เกี่ยวข้อง

1.5.2 ศึกษาหลักการ และทฤษฎีที่เกี่ยวข้อง

1.5.3 พัฒนาหลักการเพื่อแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู โดยพัฒนาวิธีการวิวัฒนาการโดยใช้ (Differential Evolution Algorithm: DE) ที่ทำการทดลองเพื่อหาค่าพารามิเตอร์ต่าง ๆ ที่เกี่ยวข้อง เพื่อให้ได้ค่าที่เหมาะสม และพิจารณาคุณภาพของคำตอบที่ได้

1.5.4 วิเคราะห์และทำการเปรียบเทียบผลการวิจัย

1.5.5 สรุปและเสนอแนะ

1.5.6 จัดทำรูปเล่มวิทยานิพนธ์ และนำเสนองานวิจัย

1.6 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย

1.6.1 เป็นการประยุกต์ใช้วิธีการวิวัฒนาการโดยใช้ (Differential Evolution Algorithm: DE) สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ในปัญหาที่มีขนาดกลาง และปัญหาที่มีขนาดใหญ่

1.6.2 เป็นการบูรณาการวิธีเมตาฮิวริสติก คือ วิธีการวิวัฒนาการโดยใช้ (Differential Evolution Algorithm: DE) ในการพัฒนาหาคำตอบแบบใหม่

1.6.3 เป็นแนวทางในการประยุกต์วิธีการวิวัฒนาการโดยใช้ (Differential Evolution Algorithm: DE) สำหรับแก้ปัญหาการจัดสมดุลสายการประกอบ และปัญหาทางการผลิตในลักษณะอื่น ๆ ได้

บทที่ 2
ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะเป็นการอธิบายถึงแนวคิดพื้นฐานของระบบการผลิต หลักการและทฤษฎีที่เกี่ยวข้องกับการจัดสมดุลสายการผลิตหรือสายการประกอบ รวมถึงงานวิจัยต่าง ๆ ที่เกี่ยวข้องซึ่งจะนำเสนอในรายละเอียดต่อไป

2.1 แนวคิดพื้นฐาน

2.1.1 ระบบของการผลิต (Production system) สามารถแบ่งตามลักษณะของการผลิตได้เป็น 2 ประเภท คือ (นุชสรา เกรียงกรกฎ, 2550)

2.1.1.1 การผลิตแบบไม่ต่อเนื่อง (Intermittent production) หรือการผลิตแบบสั่งทำ (Job shop) หมายถึงกระบวนการผลิตสินค้าเป็นชุดหรือ เป็นครั้งคราวตามใบสั่งการผลิต ซึ่งสินค้าที่ผลิตอาจมีหลายชนิด ที่มีจำนวนการผลิตที่ไม่เท่ากันและไม่แน่นอน โดยขั้นตอนการทำงานของแต่ละผลิตภัณฑ์จะแตกต่างกันออกไป

2.1.1.2 การผลิตแบบต่อเนื่อง (Continuous production) เป็นการผลิตสินค้าที่มีจำนวนมาก มีกระบวนการผลิตที่ต่อเนื่องและค่อนข้างแน่นอน เครื่องจักรที่ใช้ส่วนมากจะเป็นเครื่องจักรขนาดใหญ่หรือชนิดพิเศษ เพื่อผลิตผลิตภัณฑ์เฉพาะอย่าง การผลิตจะเริ่มจากการป้อนวัตถุดิบเข้าไปในสายการผลิต และตำแหน่งของการทำงานต่าง ๆ จะถูกกำหนดตามลำดับขั้นตอนเป็นสายการผลิตที่แบ่งเป็นจุดการทำงานหรือหลายๆสถานีต่อเนื่องกันตามลำดับ จนเป็นผลิตภัณฑ์สำเร็จรูป ในการเคลื่อนย้ายชิ้นส่วนมักใช้สายพานลำเลียง หรืออื่น ๆ ซึ่งการผลิตแบบต่อเนื่องนี้ ยังสามารถแบ่งกระบวนการผลิตตามชนิดของระบบการผลิต และตามจำนวนชนิดของสินค้า ได้ดังนี้

1) การแบ่งกระบวนการผลิตตามชนิดของระบบการผลิต มี 2 แบบ คือ

1.1) สายการผลิตแบบส่งถ่าย (Transfer Line)

จะอาศัยการทำงานของเครื่องอัตโนมัติ เพื่อส่งถ่ายวัสดุและชิ้นงาน แรงงานคนที่ใช้ส่วนใหญ่จะเป็นการควบคุมการทำงานของเครื่องจักร ไม่ว่าแผนการผลิตจะมีการเปลี่ยนแปลงอย่างไร สายการผลิตนี้จะไม่มีการเปลี่ยนแปลง กำลังการผลิตจะคงที่และผลผลิตขึ้นกับชั่วโมงการทำงานของเครื่องจักร ตัวอย่างสายการผลิตแบบนี้ เช่น การผลิตเยื่อกระดาษ การผลิตน้ำอัดลม การกลั่นน้ำมัน ฯลฯ

1.2) สายการผลิตแบบงานประกอบ (Assembly Line)

สายการผลิตแบบงานประกอบ ส่วนใหญ่จะใช้แรงงานคนเป็นหลักในการประกอบชิ้นส่วนต่าง ๆ เข้าด้วยกัน โดยอาจใช้คนเข้าประจำตามสถานีงานต่าง ๆ หรือเครื่องจักรที่มีความยืดหยุ่นในการเปลี่ยนแปลงได้พอสมควร ซึ่งการผลิตแบบนี้จะมีการเปลี่ยนแปลงสายการผลิตในสถานีงานต่าง ๆ เพื่อสอดคล้องกับแผนการผลิต เช่น โรงงานประกอบตู้เย็น โทรทัศน์ เครื่องใช้ไฟฟ้า ฯลฯ

2) แบ่งกระบวนการผลิตตามจำนวนชนิดของสินค้า

2.1) สายการผลิตแบบสินค้าชนิดเดียว (Single Model Line)

จะเป็นสายการผลิต ที่จะผลิตสินค้าชนิดใดชนิดหนึ่งโดยเฉพาะอุปสงค์ของสินค้าชนิดนี้ต้องมีมากพอที่จะให้สายการผลิตสามารถผลิตสินค้าชนิดเดียวได้ตลอดเวลา

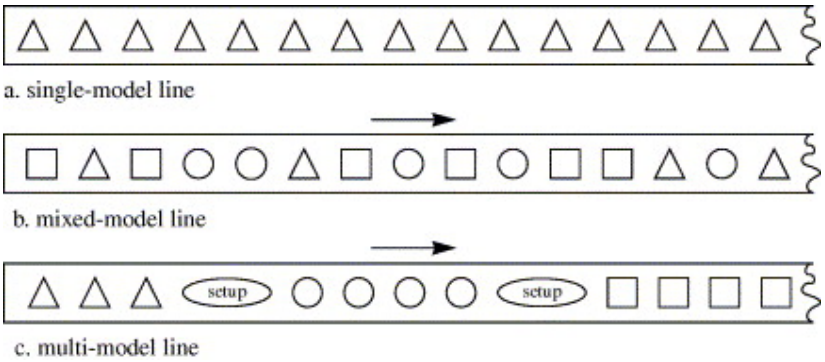
2.2) สายการผลิตแบบสินค้าหลายชนิด (Multi-Model Line)

เป็นสายการผลิตที่ผลิตสินค้าตั้งแต่ 2 ชนิดขึ้นไป โดยสินค้าแต่ละชนิดจะมีกระบวนการผลิตที่ใกล้เคียงกัน จะทำการผลิตสินค้าทีละชนิด สินค้าจะมาเป็นชุด และช่วงจะเปลี่ยนรุ่นผลิต อาจจะต้องมีการปรับสายการผลิตใหม่

2.3) สายการผลิตแบบสินค้าผสม (Mixed-Model Line)

จะผลิตสินค้าตั้งแต่ 2 ชนิดขึ้นไป เช่นเดียวกับสายการผลิตแบบสินค้าหลายชนิด แต่จะต่างกันตรงที่ สายการผลิตแบบผสมจะผลิตสินค้าต่างชนิดพร้อม ๆ กันในสายการผลิต โดยระหว่างการผลิตจะไม่มีการปรับสายการผลิต

รูปภาพแสดงสายการประกอบที่แบ่งกระบวนการผลิตตามชนิดของสินค้า ระบบสายการประกอบโทรทัศน์ที่เป็นแบบสินค้าชนิดเดียวหรือสินค้าหลายชนิด และระบบสายการประกอบรถยนต์แบบสินค้าผสม ดังในภาพที่ 2.1, 2.2 และ 2.3 ตามลำดับ



ภาพที่ 2.1 สายการประกอบที่แบ่งกระบวนการผลิตตามจำนวนของชนิดของสินค้า
ที่มา: Scholl and Becker (2006)

ภาพที่ 2.2 ระบบสายการประกอบเครื่องปรับอากาศที่เป็นแบบสินค้าชนิดเดียวหรือสินค้าหลายชนิด
ที่มา: Prasit, K. (2015: Website)

ภาพที่ 2.3 ระบบสายการประกอบรถยนต์แบบสินค้าแบบผสม
ที่มา: Suneeya, U. (2014: Website)

2.2 หลักการและทฤษฎีที่เกี่ยวข้อง

2.2.1 การจัดสมดุลสายการประกอบ

การจัดสมดุลสายการประกอบ (Assembly Line Balancing) หมายถึง การพยายามที่จะจัดสถานีงานต่าง ๆ ให้มีอัตราการทำงาน หรือเวลาที่ใช้ในการผลิตแต่ละชิ้นเท่า ๆ กัน ถ้าหากเวลาที่ใช้ในการผลิตไม่เท่ากันแล้ว อัตราการผลิตสินค้านั้น ๆ จะถูกกำหนดโดยเวลาการทำงานของสถานีงานที่ใช้เวลามากที่สุด ซึ่งเวลาที่ใช้ในสถานีงานที่เป็นตัวกำหนดอัตราการผลิตของสินค้านี้ เรียกว่า รอบเวลาการผลิต ซึ่งหมายถึง เวลาระหว่างที่สินค้าเสร็จออกมาแต่ละชิ้น ซึ่งจะเท่ากับเวลาของสถานีงานที่ช้าที่สุด ฉะนั้น จะเห็นว่าเกิดการรอคอยขึ้นในสถานีงานที่ใช้เวลาน้อยกว่า ทำให้เกิดการสูญเสียอัตราการผลิต หรือการว่างงานเกิดขึ้น หรือมีของค้างปริมาณมากรอที่จะผ่านสถานีงานที่ช้านั้น ซึ่งเราจะต้องพยายามทำให้เกิดขึ้นน้อยที่สุด (นุชสรา เกรียงกรกฎ, 2550)

ในการจัดสมดุลสายการประกอบ โดยทั่วไปจะเริ่มต้นด้วยการกำหนดรอบเวลาการผลิตตามลำดับขั้นงานต่าง ๆ และเวลามาตรฐานของการทำงานแต่ละขั้นตอนของงานนั้น ๆ จากนั้นจะพยายามรวมขั้นงานเข้าด้วยกันให้เป็นสถานีงาน โดยจัดให้มีเวลาว่างทั้งหมดน้อยที่สุด ในกรณีที่จำนวนสถานีงานมีมากหรือน้อยไป ก็อาจจัดใหม่โดยให้มีรอบเวลาผลิตมากขึ้น หรือน้อยลง

2.2.1.1 ระบบสายการประกอบ มีส่วนประกอบต่าง ๆ ดังนี้

1) สายการประกอบจะมีหลายสถานีงาน: k = 1, ... , m

2) เวลาระหว่างที่สินค้าเสร็จออกมาแต่ละชิ้น จะเรียกว่า รอบเวลาการผลิต (cycle time: c)

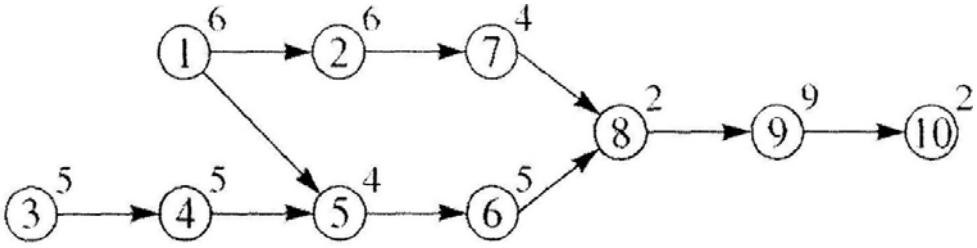
3) โดยในกลุ่มงานใด ๆ จะประกอบด้วยหลายงานย่อย ๆ (V = {1, ... , n})

4) เวลาที่ใช้ในแต่ละงานย่อย จะเรียกเป็นเวลางานย่อยหรือเวลาขั้นงาน (Task time: t_j)

5) เวลาที่ใช้ในแต่ละสถานีงาน คือ เวลาสถานีงาน (Workstation time)

2.2.1.2 แผนภาพลำดับขั้นตอนงาน (Precedence diagram)

แผนภาพลำดับขั้นตอนงาน จะแสดงขั้นตอนของงานย่อยต่าง ๆ โดยจะใช้จุดเชื่อม (Node) เป็นสัญลักษณ์ในการเชื่อมต่อระหว่างส่วนของงาน และมีลูกศรเป็นตัวกำหนดทิศทางการดำเนินงาน โดยการดำเนินงานจะเริ่มจากด้านซ้ายสุดของผังผ่านกระบวนการประกอบต่าง ๆ จนเป็นผลิตภัณฑ์ที่อยู่ทางด้านขวามือ ดังในภาพที่ 2.4 จะแสดงแผนภาพลำดับขั้นตอนงานหนึ่งที่ประกอบด้วย 10 งานย่อย ที่มีเวลาของงานย่อย หรือเวลาขั้นงาน ตั้งแต่ 2.9



ภาพที่ 2.4 ตัวอย่างแผนภาพลำดับขั้นตอนงาน

2.2.1.3 ประสิทธิภาพของสายการผลิต หรือ ประสิทธิภาพของความสมดุล (Line-Efficiency or Efficiency of Balance: E)

ประสิทธิภาพของสายการผลิต จะเป็นดัชนีที่แสดงให้เห็นถึงความสามารถ ของการจัดงานลงในสถานีงานเพื่อให้เกิดเวลาสูญเปล่าน้อยที่สุด สามารถคำนวณได้จากสูตร

$$E = (\sum t / mc) * (100) \qquad (2.1)$$

โดยค่า E เป็นค่าเปอร์เซนต์

$\sum t$ = เวลาขั้นงานรวม

m = จำนวนสถานีงาน

c = รอบเวลาผลิต

2.2.1.4 การสูญเสียความสมดุล (Balance Delay: D)

เป็นเครื่องชี้ถึงประสิทธิภาพของสายการประกอบ หรืองานผลิตที่มีความไม่สมบูรณ์เกิดขึ้น โดยพิจารณาที่เวลาสูญเปล่า (Idle time) ของการจัดงานลงสถานีงาน จะแทนสัญลักษณ์ด้วย D ที่มีสูตรการหาค่า D คือ

$$D = ((mc - \sum t) / mc) * 100 \qquad (2.2)$$

หรือ

$$D = 100 - E \qquad (2.3)$$

โดยค่า D เป็นค่าเปอร์เซนต์

m = จำนวนสถานีงาน

c = รอบเวลาผลิต

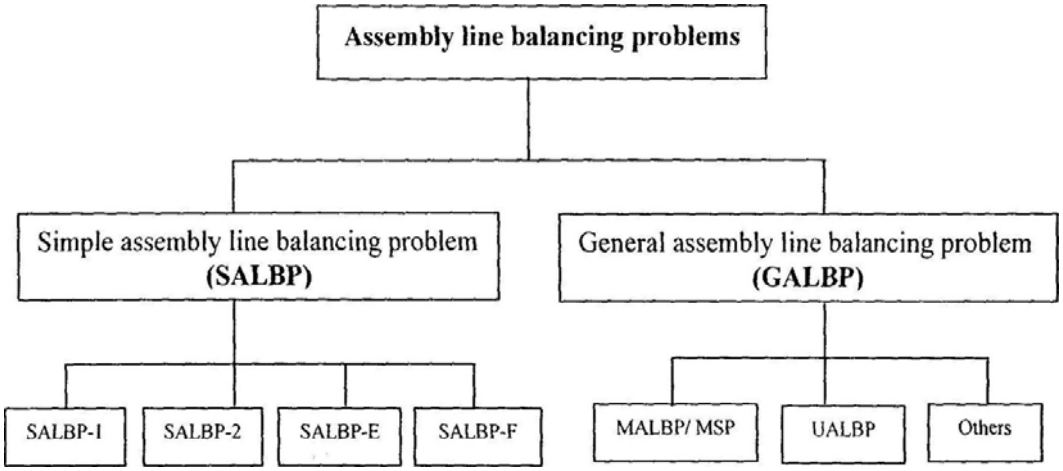
$\sum t$ = เวลาขั้นงานรวม

2.2.1.5 เวลาสูญเปล่า (Idle Time: I)

$$I = mc - \sum t \qquad (2.4)$$

2.2.2 ประเภทของปัญหาการจัดสมดุลสายการประกอบ

Scholl and Becker (2006) ได้จัดแบ่งประเภทของปัญหาการจัดสมดุลสายการประกอบ (Classification of Assembly Line Balancing Problem) ได้ดังภาพที่ 2.5



ภาพที่ 2.5 ประเภทของปัญหาการจัดสมดุลสายการประกอบ
ที่มา: Nuchsara and Nalin (2007)

2.2.2.1 ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรง (Simple Assembly -Line Balancing Problem: SALBP)

ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงที่มีการผลิตสินค้าชนิดเดียว ซึ่งปัญหานี้สามารถแบ่งออกได้อีก 4 ชนิดด้วยกัน คือ ปัญหา SALBP-1, SALBP-2, SALBP-E และปัญหา SALBP-F โดยรายละเอียดของปัญหานี้จะอธิบายในลำดับต่อไป

2.2.2.2 ปัญหาการจัดสมดุลสายการประกอบแบบทั่วไป (General Assembly Line Balancing Problem: GALBP)

ปัญหาการจัดสมดุลสายการประกอบแบบทั่วไปนี้เป็นปัญหาที่มีความยุ่งยากซับซ้อนขึ้น ที่สามารถแบ่งปัญหาออกได้เป็น 3 ชนิด ได้แก่ ปัญหาแบบ MALBP/MSP ปัญหา UALBP และปัญหาอื่น ๆ

1) ปัญหา MALBP/MSP (Mixed model Assembly Line Balancing Problem and Mixed model Sequencing Problem)

ปัญหา MALBP จะเป็นปัญหาการจัดสมดุลสายการประกอบที่มีการผลิตสินค้าแบบผสมที่มีหลายรุ่น โดยการจัดงานลงสถานีงานจะทำการพิจารณาเวลางาน (Task time) ที่ต่างกัน สำหรับสินค้าแต่ละรุ่น เพื่อหาจำนวนสถานีงาน หรือรอบเวลาการผลิตที่เหมาะสมที่สุด

สำหรับปัญหา MSP เป็นการจัดลำดับการผลิตให้สินค้าทุกรุ่นที่จะทำการผลิตให้เกิดความเหมาะสมที่สุด เพื่อลดเหตุการณ์ต่าง ๆ ที่ไม่มีประสิทธิภาพในสายการผลิตให้เกิดขึ้นน้อยที่สุด เช่น การหยุดชะงักของสายการผลิต และการหยุดสายการผลิต เพื่อซ่อม ฯลฯ ซึ่ง ปัญหา MSP

จะสัมพันธ์กันกับปัญหา MALBP คือ ถ้าสามารถจัดสมดุลงานของปัญหา MALBP ให้ ได้ผลลัพธ์ที่ดี ก็จะทำให้การจัดลำดับการผลิตในปัญหา MSP ได้ผลลัพธ์ที่ดีเช่นกัน

2) ปัญหา UALBP (U-line Assembly Line Balancing Problem)

เป็นปัญหาการจัดสมดุลสายการประกอบแบบตัวยูที่มีการผลิตสินค้าชนิดเดียว ผลจากการจัดสมดุลงาน สถานีงานต่าง ๆ จะถูกจัดเป็นรูปตัวยูที่สามารถจัดงานเริ่มต้นและงานสุดท้ายของแผนภูมิลำดับงานก่อน-หลัง ลงในสถานีงานเดียวกันได้ นั่นคือ ปัญหาการสมดุลงานแบบตัวยูจะมีความซับซ้อนมากกว่าปัญหาแบบเส้นตรง โดยปัญหา UALBP สามารถแบ่งเป็นปัญหาย่อย ๆ ได้อีก 3 ชนิด คือ ปัญหา UALBP-1, UALBP-2 และ UALBP-E ซึ่งรายละเอียดจะอธิบายในลำดับต่อไป

3) ปัญหาอื่น ๆ

ในปัญหา GALBP จะเป็นปัญหาที่มีขอบเขตที่กว้างมาก ในปัญหาแบบอื่นก็อาจจะพิจารณาตามเงื่อนไขอื่น ๆ ที่เกี่ยวข้องเพิ่มเติม เช่น การเลือกเครื่องมือและอุปกรณ์ (Equipment selection) การผลิตในแบบอื่น ๆ (Processing alternatives) และข้อจำกัดของการกำหนดงาน (Assignment restrictions) ฯลฯ ซึ่งรายละเอียดสามารถดูได้จาก Scholl and Becker (2006)

จากขอบเขตของการศึกษาวิจัยในครั้งนี้ จะพิจารณาเฉพาะปัญหาการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 ซึ่งรายละเอียดของปัญหาดังกล่าว มีดังต่อไปนี้

2.2.3 ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู

ในปัญหาการจัดสมดุลสายการประกอบแบบตัวยู (U-line Assembly Line Balancing Problem: UALBP) สามารถแบ่งปัญหาออกได้เป็น 3 ประเภท ซึ่งคล้ายกับปัญหาการจัดสมดุลสายการประกอบแบบง่าย (Miltenburg and Wijngaard, 1994; Scholl and Klein, 1999) ดังนี้

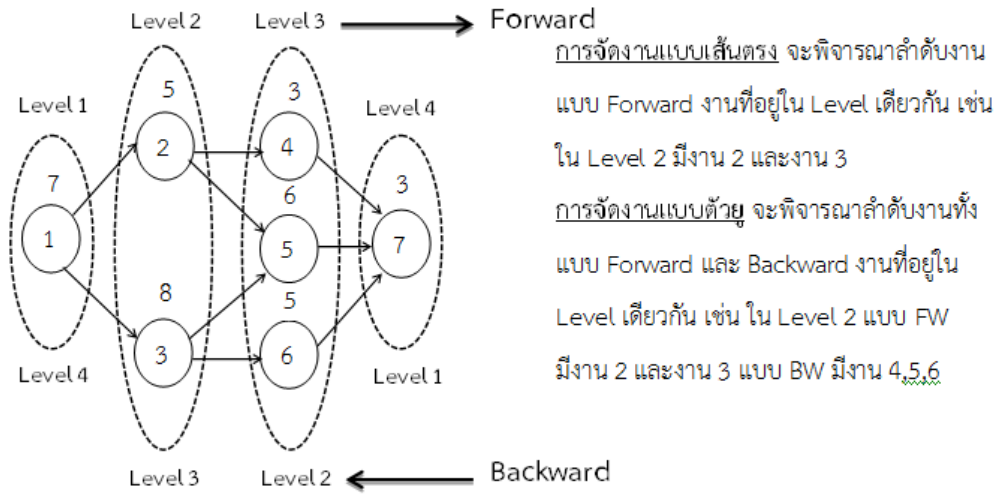
2.2.3.1 UALBP-1: กรณีทราบค่ารอบเวลาการผลิต (c) และหาจำนวนสถานีงานที่น้อยที่สุด

2.2.3.2 UALBP-2: กรณีทราบค่าจำนวนสถานีงาน (m) และหารอบเวลาการผลิตที่น้อยที่สุด

2.2.3.3 UALBP-E: กรณีค่า c และ m เป็นตัวแปร จากนั้นหาค่าประสิทธิภาพของสายการผลิตสูงสุด (E)

โดยรูปแบบของปัญหาแบบตัวยู (UALBP) จะต่างจากปัญหาแบบง่าย (SALBP) เฉพาะเงื่อนไขของลำดับขั้นตอนงานก่อน-หลัง (Precedence constraints) ในปัญหาแบบง่าย SALBP ทุกงานที่อยู่ก่อนหน้า (Predecessors) งาน j จะถูกจัดลงสถานีงานที่ 1, ... , k ก่อน และแต่ละงานจะถูกจัดลงได้เพียง 1 สถานี และสำหรับปัญหาแบบตัวยู ทุกงานที่อยู่ก่อนหน้า (Predecessors) และ/หรือทุกงานที่อยู่ตามหลัง (Successors) งาน j จะถูกจัดลงสถานีงานที่ 1, ... , k ได้เพียง 1 สถานีเท่านั้น โดยที่เวลารวมของสถานีงานนั้น ๆ จะต้องมีค่าไม่มากกว่ารอบเวลาการผลิต นั่นคือในการจัดสมดุลงานในปัญหาแบบตัวยู สามารถพิจารณางานที่จะจัดลงสถานีงานได้ทั้ง 2 ทิศทาง คือ เลือกงานจากข้างหน้าไปข้างหลัง หรือจากซ้ายไปขวา (Forward) ของแผนผังลำดับขั้นงาน (Precedence diagram) และจากข้างหลังไปข้างหน้า หรือจากขวาไปซ้าย (Backward) ในขณะที่ปัญหาแบบเส้นตรงจะพิจารณาเลือกงานทิศทางเดียวเช่น เลือกงานจากข้างหน้าไปข้างหลัง หรือจากซ้ายไปขวา (Forward) เป็นต้น

ซึ่งจากหลายตัวอย่าง พบว่าปัญหาแบบตัวยู จะให้ค่าประสิทธิภาพของสายการผลิต (Line efficiency) ที่ดีกว่า เนื่องจาก แต่ละสถานีงานมีภาระงานที่เท่า ๆ กัน หรือใกล้เคียงกันมากกว่า (Scholl, 1999)



ภาพที่ 2.6 ตัวอย่างการจัดงานที่อยู่ในระดับเดียวกัน

จากปัญหาการจัดสมดุลสายการประกอบจะมุ่งเน้นศึกษาปัญหาประเภทที่ 1 คือ กรณีทราบค่ารอบเวลาการผลิต และหาจำนวนสถานีงานที่น้อยที่สุด ดังนั้น รูปแบบทางคณิตศาสตร์ของปัญหาการสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 จะมีรายละเอียดดังนี้

2.2.3.4 รูปแบบทางคณิตศาสตร์ของปัญหา UALBP-1 (Mathematical Formulations of UALBP-1)

Miltenburg and Wijngaard (1994) เป็นผู้กำหนดรูปแบบทางคณิตศาสตร์ของปัญหาการสมดุลสายการประกอบแบบตัวยู ที่พิจารณารูปแบบทางคณิตศาสตร์มาจากปัญหาการสมดุลสายการประกอบแบบเส้นตรงที่ถูกกำหนดขึ้นโดย Gutjahr และ Nemhauser ในปี ค.ศ.1964 (Gutjahr and Nemhauser, 1964)

โดยกำหนดให้เซตของงานใด ๆ คือ F ซึ่ง F = {i| i =1,2,...,n} และ P เป็นเซตของเงื่อนไขของลำดับงานก่อน-หลัง (Precedence constraints) ซึ่ง P = {(x,y) | โดยที่งาน x ต้องทำเสร็จก่อนงาน y } และ T เป็นเซตของเวลางานใด ๆ ที่ T = { t_i | i =1, 2, ..., n} กำหนดให้รอบเวลาการผลิตเป็น c และ m เป็นจำนวนสถานีงาน ทั้งนี้เพื่อหาเซตย่อย ๆ ของ F, {S1, S2, ..., Sn) ที่ S_k = {i| โดยที่งาน i ถูกจัดลงสถานีงานที่ k} และให้เป็นไปตามเงื่อนไขต่าง ๆ ดังต่อไปนี้

$$\bigcup_{k=1}^{m} S_k = F \tag{2.5}$$

$$S_k \bigcap_{k^1 j} S_j = \varnothing \tag{2.6}$$

$$\sum_{i=S_k} t_i \le c \text{ , } k = 1,2,....n \tag{2.7}$$

For each task *y*,

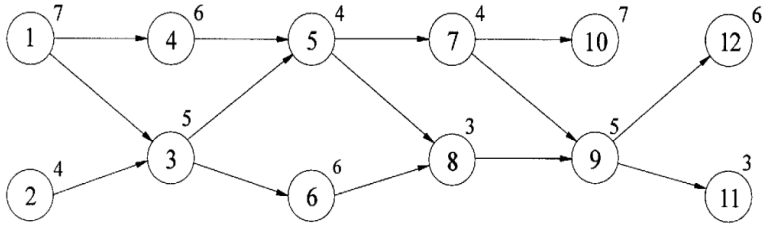
If $(x, y) \in P, x \in S_k, y \in S_j$, then $k \le j$, for all x; or (2.8)

If $(y, z) \in P, y \in S_j, z \in S_i$, then $i \le j$, for all z.

$$\left[mc - \sum_{k=1}^{m} \sum_{i=S_k} t_i \right] \text{is minimized} \tag{2.9}$$

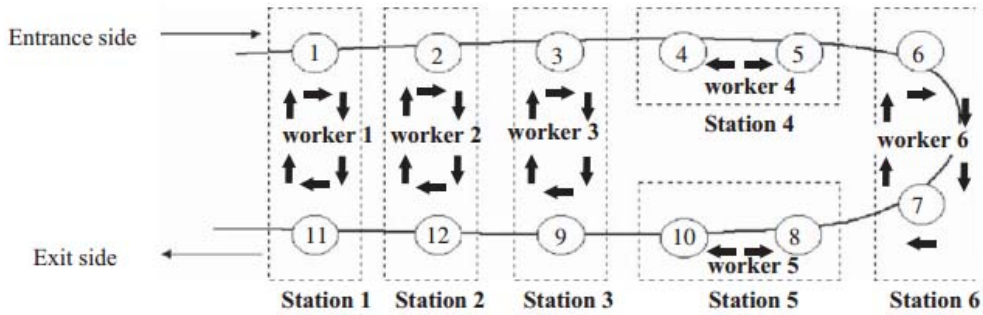
โดยเงื่อนไขที่ (2.5) เป็นการแสดงให้เห็นว่างานทุกงานจะถูกจัดลงในสถานีงาน และจากเงื่อนไขที่ (2.6) คืองานทุกงานจะถูกกำหนดลงสถานีงานเพียงครั้งเดียว เงื่อนไขที่ (2.7) เพื่อให้มั่นใจว่าเวลารวมของงานในแต่ละสถานีงาน จะต้องไม่เกินรอบเวลาการผลิต (Cycle time constraints) และเงื่อนไขที่ (2.8) เป็นเงื่อนไขของลำดับงานก่อน-หลัง ซึ่งจะต้องไม่ผิดกฎของการจัดสมดุลงานแบบตัวยู สำหรับฟังก์ชันเป้าหมาย เพื่อหาจำนวนสถานีงานที่น้อยที่สุด ดังในเงื่อนไขที่ (2.9)

รูปภาพแสดงตัวอย่างแผนผังลำดับขั้นตอนงาน และแบบตัวยู ดังแสดงในภาพที่ 2.7, 2.8, 2.9 และภาพที่ 2.10 เป็นการแสดงการคำนวณหาประสิทธิภาพของสายการผลิตตามลำดับ



ภาพที่ 2.7 แผนผังลำดับขั้นตอนงาน

ที่มา: Scholl and Klein (1999)



สมมติ cycle time = 10

U-line m = 6 สถานีงาน

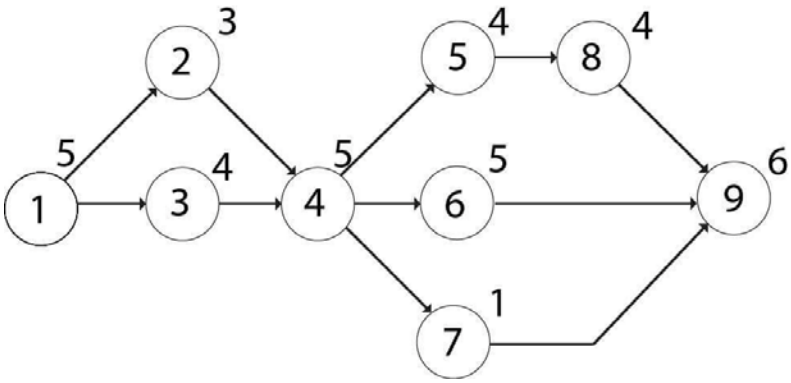
$$E = \frac{\sum t}{m \times c} \times 100$$

E = 60/6(10)x100

= 100%

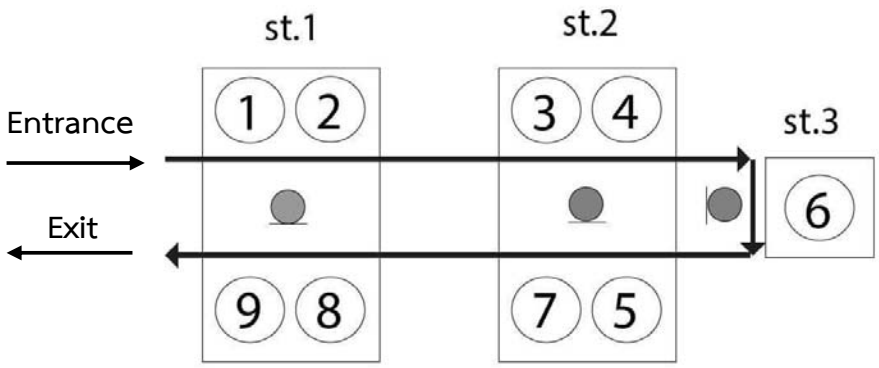
Balance delay = 100-100 = 0%

ภาพที่ 2.8 ผลของการจัดสมดุลงานแบบตัวยูในปัญหาของ Scholl and Klein (1999) ตัวอย่างการจัดสมดุลงานและการคำนวณประสิทธิภาพของสายการผลิต



ภาพที่ 2.9 แผนผังลำดับขั้นตอนงาน Jaeschke problem

ที่มา: Jaeschke (1964)



สมมติ total tasks time = 37 , assume c =18

U-line m = 3 สถานีงาน

E = 37/3 (18)x100

= 68.52%

Balance delay = 100-68.52 = 31.48%

ภาพที่ 2.10 ตัวอย่างการการคำนวณประสิทธิภาพของสายการผลิต Jaeschke problem
ที่มา: Jaeschke (1964)

2.2.3.5 ข้อดีของการจัดสายการผลิตแบบ U-line (Monden,1993)

1) ทำให้พนักงานมีทักษะงานที่หลากหลาย (Multi skill, cross-trained) และสามารถทำงานในสถานีงานที่แตกต่างกันได้ อีกทั้งสามารถตอบสนองต่อการเปลี่ยนแปลงของสภาพแวดล้อมการทำงานที่ไม่ปกติ เช่น กรณีเครื่องจักรเกิดเหตุขัดข้อง หรือ พนักงานขาดงานได้อย่างรวดเร็ว

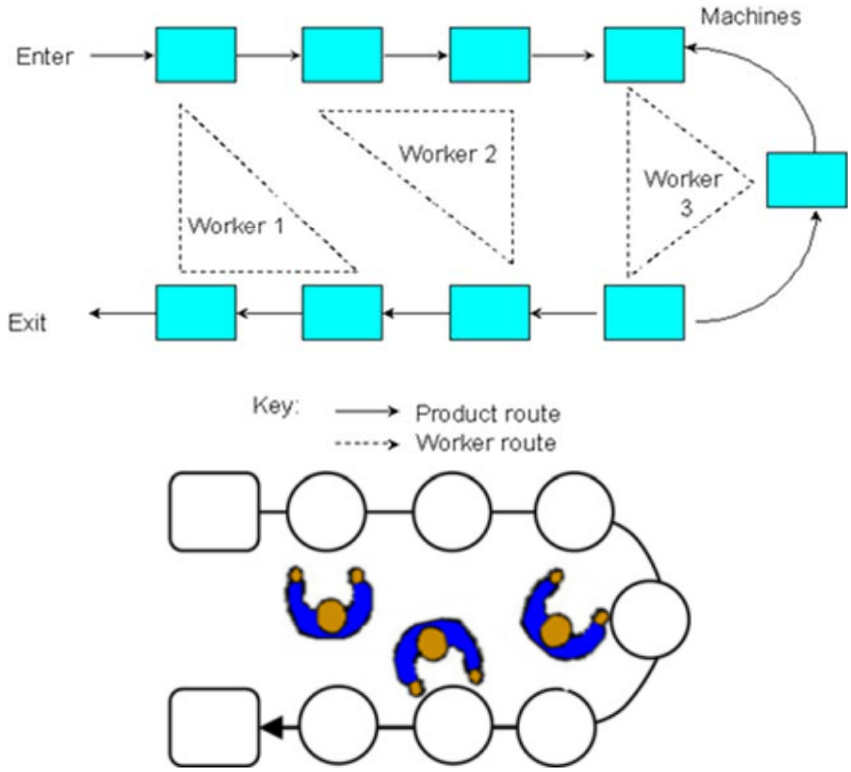
2) สามารถจัดงานที่จุดเริ่มต้นและจุดสุดท้ายของแผนผังลำดับขั้นงานให้อยู่ในสถานีงานเดียวกันได้

3) ง่ายต่อการปรับเปลี่ยนรอบเวลาผลิต เนื่องจากพนักงานมีศักยภาพสูงในการจัดสมดุลสายการผลิตใหม่ (Re-balancing) สำหรับรอบเวลาการผลิตใหม่

4) พนักงานมีทักษะสูง ในการเปลี่ยนแปลงกระบวนการผลิต

5) ยืดหยุ่นต่อการเพิ่ม/ย้ายพนักงาน สำหรับสายการผลิตที่มีการจัดสมดุลใหม่

6) ที่สำคัญที่สุด คือ การจัดสมดุลงานแบบตัวยู จะให้จำนวนสถานีงานที่น้อยกว่าหรือเท่ากับ แบบเส้นตรง และการจัดสมดุลงานแบบตัวยูจะซับซ้อนกว่า เนื่องจากมีพื้นที่ ของคำตอบ (Larger search space) ที่มากกว่านั่นเองลักษณะรูปแบบโรงงานอุตสาหกรรมได้มีการจัดสายการประกอบแบบตัวยูดังรูปภาพในภาพที่ 2.11 และ 2.12



ภาพที่ 2.11 การจัดสายการผลิตแบบตัวยู
ที่มา: Thailandindustry New (2017: Website)

ภาพที่ 2.12 การจัดสมดุลสายการประกอบแบบตัวยู ของการประกอบ Honda Automobile
ที่มา: Thailand Honda Automobile (2016: Website)

จากการศึกษางานวิจัยที่เกี่ยวข้องผู้วิจัยพบว่าวิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) เป็นวิธีที่ใหม่สำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ยังไม่มีนักวิจัยคนไหนทำมาก่อนและวิธีการวิวัฒนาการโดยใช้ผลต่างเป็นวิธีที่ช่วยสามารถค้นหาคำตอบใกล้เคียงกับคำตอบที่ดีที่สุดหรือให้ค่าคำตอบที่เหมาะสมที่สุด สำหรับปัญหาขนาดใหญ่และมีความซับซ้อนของปัญหา โดยสามารถหาคำตอบภายในพื้นที่คำตอบที่เป็นไปได้ โดยใช้ระยะเวลาที่สั้นในการหาคำตอบเมื่อเปรียบกับวิธีเมตาฮิวริสติกวิธีอื่น ๆ ผู้วิจัยจึงได้เลือกวิธีการวิวัฒนาการโดยใช้ผลต่างในการแก้ปัญหาในงานวิจัยในครั้งนี้

2.2.4 การหาค่าที่เหมาะสมที่สุดโดยใช้วิธีการวิวัฒนาการโดยใช้ผลต่าง

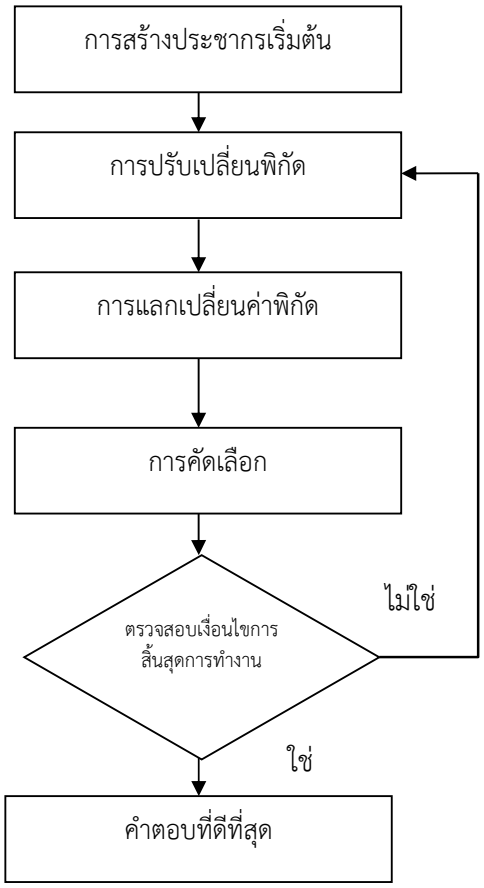
วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithm: DE) ถูกนำเสนอโดย Storn andd Price (1997) โดยวิธีการนี้พัฒนามาเพื่อแก้ปัญหาที่มีความยาก ซับซ้อน ซึ่งไม่สามารถแก้ปัญหาด้วยวิธีการแม่นตรงได้หรือถ้าแก้ได้ก็อาจจะต้องใช้เวลานานในการแก้ปัญหา โดยใช้ประชากรในแต่ละรุ่นในการค้นหาคำตอบและมีการพัฒนาคำตอบจากรุ่นสู่รุ่น ในกระบวนการวิวัฒนาการโดยใช้ผลต่างจะใช้เลขจำนวนจริงในการแทนคำตอบในแต่ละรุ่นของสมาชิกของประชากรในแต่รุ่น โดยวิธีการกลยุทธ์หรือขั้นตอนการวิวัฒนาการเริ่มต้นตั้งแต่ การสร้างคำตอบเริ่มต้น ดำเนินการวิธีการกลายพันธุ์โดยการเพิ่มหรือปรับเปลี่ยนเลขจำนวนจริงสุ่มเข้าไปในแต่ละส่วนประกอบของเวคเตอร์หรือโครโมโซม ดำเนินการแลกเปลี่ยนส่วนประกอบกับเวคเตอร์ และการคัดเลือกเวคเตอร์หรือโครโมโซมซึ่งหากเวคเตอร์หรือโครโมโซมที่ผ่านกระบวนการ กลายพันธ์และการแลกเปลี่ยนส่วนประกอบดีกว่าคำตอบของเวคเตอร์หรือโครโมโซมต้นแบบ จะทำการแทนที่เวคเตอร์หรือโครโมโซมต้นแบบด้วยโครโมโซมใหม่ หากคำตอบไม่ดีขึ้น โครโมโซมหรือเวคเตอร์ต้นแบบจะยังคงถูกใช้เป็นโครโมโซมหรือเวคเตอร์ต้นแบบของรอบถัดไป

2.2.4.1 กระบวนการขั้นตอนวิธีวิวัฒนาการโดยใช้ผลต่าง

การกลายพันธ์ในกระบวนการวิวัฒนาการโดยใช้ผลต่างจะถูกเรียกว่าการปรับเปลี่ยนค่าในพิกัดสามารถดำเนินการได้ด้วยการหาผลต่างของเวคเตอร์ที่สุ่มมาสองเวคเตอร์ของประชากรรุ่นหนึ่ง ๆ โดยนำผลต่างนี้อาจจะนำไปคูณกับปัจจัยขยายผลต่าง (scaling factor: F) แล้วจึงนำไปบวกกับเวคเตอร์ที่สุ่มมาอีกเวคเตอร์หนึ่ง ประชากรรุ่นที่กำลังจะถูกปรับเปลี่ยนค่าในพิกัดเรียกว่าเวคเตอร์เป้าหมาย (target vector) และประชากรที่ถูกปรับเปลี่ยนแล้วจะเรียกว่า มิวแทนต์เวคเตอร์ เมื่อประชากรถูกปรับเปลี่ยนจนครบจำนวนประชากรที่ตั้งไว้ล่วงหน้า (Number of Population: NP) จะทำการแลกเปลี่ยนค่าในพิกัด ระหว่างเวคเตอร์เป้าหมายและมิวแทนต์เวคเตอร์โดยใช้ความน่าจะเป็นในการแลกเปลี่ยนค่าในพิกัดซึ่ง เป็นพารามิเตอร์ที่ควบคุมเรียกว่าอัตราการปรับเปลี่ยน (crossover rate: CR) เวคเตอร์ที่ผ่านการ แลกเปลี่ยนค่าในพิกัดแล้วจะถูกเรียกว่าไทรอัลเวคเตอร์ (trial vector) จากนั้นนำไทรอัลเวคเตอร์ไปหาค่าฟิตเนสฟังก์ชั่นหรือฟังก์ชั่นเป้าหมาย จากนั้นทำการเปรียบเทียบค่าฟิตเนสฟังก์ชั่นของไทรอัล เวคเตอร์กับเวคเตอร์เป้าหมาย เวคเตอร์ใดดีกว่าจะถูกใช้เป็นเวคเตอร์เป้าหมายในรอบถัดไป โดยได้แสดงขั้นตอนวิธีแสดงอัลกอริทึมของวิธีวิวัฒนาการโดยใช้ผลต่างและลำดับขั้นตอนของวิธีวิวัฒนาการโดยใช้ผลต่างในภาพที่ 2.13 และ 2.14

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)

ภาพที่ 2.13 ขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง
ที่มา: กาญจนา เศรษฐนันท์ (2558)



ภาพที่ 2.14 ลำดับขั้นตอนการทำงานของวิธีวิวัฒนาการโดยใช้ผลต่าง
ที่มา: กาญจนา เศรษฐนันท์ (2558)

1) การสร้างประชากรเริ่มต้น (Initial population)

ขั้นตอนการสร้างประชากรของคำตอบเริ่มต้นสามารถทำได้โดยการสุ่มค่าประชากรหรือเวกเตอร์ตั้งต้น (Initial population) โดยกำหนดขอบเขตของตัวแปรการตัดสินใจเพื่อให้แน่ใจว่าขอบเขตของตัวแปรการตัดสินใจจะคลอบคลุมถึงคำตอบที่ดีที่สุด และกำหนดให้โอกาสที่จะถูกเลือกของคำตอบที่มีค่าสม่ำเสมอ (Uniform probability distribution) ที่มีมิติเท่ากับ D ซึ่งอาจหมายถึงพิกัดหรือตำแหน่ง และจำนวนคำตอบที่เป็นไปได้เริ่มต้นเท่ากับ NP โดยแต่ละคำตอบหรือแต่ละเวกเตอร์เป้าหมายเรียกว่า เวกเตอร์การตัดสินใจ (Decision vector) ซึ่งจะนำมาคำนวณเพื่อหาค่าคำตอบของสมการเป้าหมายของเวกเตอร์เป้าหมาย (Cost value) หรือค่าฟิตเนส (Fitness value)

2) การปรับเปลี่ยนค่าในพิกัด (Mutation)

ขั้นตอนการปรับเปลี่ยนค่าในพิกัดเวกเตอร์เพื่อให้ได้คำตอบใหม่ของการหาผลลัพธ์ด้วยวิธี DE ประกอบไปด้วย กลยุทธ์ในการหาผลลัพธ์ การกำหนดค่าตัวแปรที่ทราบค่า และมีสัญลักษณ์ที่เกี่ยวข้องดังนี้

$X_{r1,j,G}$ = เวกเตอร์เป้าหมาย (Target vector)

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

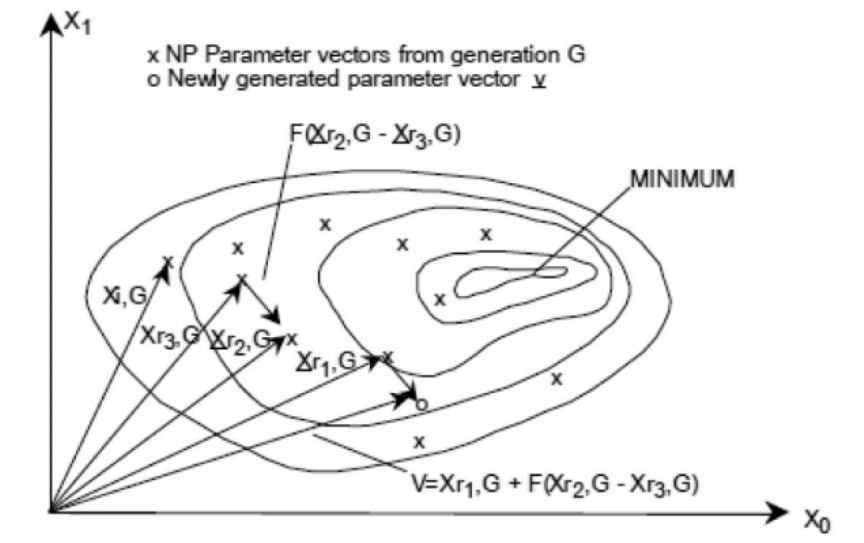
$X_{r1,j,G}$, $X_{r2,j,G}$, $X_{r3,j,G}$ = เวกเตอร์สุ่ม (Random vector) คือเวกเตอร์ที่ทำการสุ่มเลือกมา 3 เวกเตอร์

F = แฟกเตอร์ในการขยายผลต่าง (Scaling factor) คือจำนวนจริงที่มีค่าคงที่และมีค่าระหว่าง 0 ถึง 2

ในการปรับเปลี่ยนค่าพิกัดนั้นสามารถทำได้โดยทำการสุ่มเวกเตอร์มา 3 เวกเตอร์ (ตัวแทนผลลัพธ์) $X_{r1,j,G}$, $X_{r2,j,G}$ และ $X_{r3,j,G}$ มาดำเนินการ โดยเวกเตอร์ที่ถูกปรับเปลี่ยนค่าพิกัดไปแล้วจะถูกเรียกว่า “มิวแทนเวกเตอร์” โดยที่ค่ามิวแทนเวกเตอร์ ($V_{i,j,G+1}$) จะมีค่าเท่ากับ $X_{r1,j,G}$ รวมกับผลต่างของเวกเตอร์ $X_{r2,j,G}$ และ $X_{r3,j,G}$ คูณด้วย F โดยสามารถอธิบายการปรับค่าในพิกัดของเวกเตอร์ได้ดังสมการที่ 2.10

$$V_{i,j,G+1} = X_{r1,j,G} + F(X_{r2,j,G} - X_{r3,j,G}) \qquad (2.10)$$

ซึ่งลักษณะการค้นหา Mutant Vector ของฟังก์ชั่น 2 ตัวแปร สามารถแสดงดังภาพที่ 2.15



ภาพที่ 2.15 การค้นหา Mutant Vector ของฟังก์ชั่น 2 ตัวแปร
ที่มา: Price and Storn (1997)

3) การแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (Crossover หรือ Recombination)

การแลกเปลี่ยนค่าในพิกัดเวกเตอร์เป็นการเพิ่มความหลากหลายในการหาคำตอบของปัญหาโดยขั้นตอนนี้จะเป็นการสร้างไทรอัลเวกเตอร์ (Trial vector) ($U_{ji,G+1}$) โดยสมการที่ใช้ในการแลกเปลี่ยนค่าเพื่อสร้างไทรอัลเวกเตอร์แสดงในสมการที่ (2.11) ซึ่งจะใช้ในการสุ่มตัวเลขที่อยู่ระหว่าง 0-1 โดยมีขนาดตำแหน่งเท่ากับ D และจำนวนประชากรเท่ากับ NP หากตัวเลขที่สุ่มนี้ของทุกเวกเตอร์และทุกตำแหน่ง มีค่าน้อยกว่าหรือเท่ากับค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (CR) ค่าในตำแหน่งพิกัดของเวกเตอร์นั้นจะมีค่าเท่ากับค่าของมิวแทนต์เวกเตอร์ในตำแหน่งและพิกัดเดียวกัน และในกรณีที่เป็นอย่างอื่นค่าในตำแหน่งพิกัดไทรอัลเวกเตอร์ในตำแหน่งนั้นจะมีค่าเท่ากับเวกเตอร์เป้าหมาย (Target vector)

$$U_{i,j,G+1} = \begin{cases} V_{i,j,G+1} & \text{if (randb j [0,1)} \leq \text{CR) or j = rnbr (j)} \\ X_{ji,G+1} & \text{if (randb (j) > CR) or j} \neq \text{rnbr (j)} \end{cases} \quad (2.11)$$

เมื่อ $U_{i,j,G+1}$ = ไทรอัลเวกเตอร์

$V_{i,j,G+1}$ = มิวแทนเวกเตอร์

$X_{i,j,G+1}$ = เวกเตอร์เป้าหมาย

rnbr (j) = ดัชนีจากการสุ่มจากการสุ่มเลือกมีค่าเป็นเลขจำนวนเต็มที่มีค่าตั้งแต่ [1, D] โดยที่ D คือขนาดของเวกเตอร์

rndb (j) = การสุ่มตัวเลขจำนวนจริงที่มีค่าระหว่าง 0-1 ครั้งที่ j: j = 1, 2, ..., D

CR = อัตราแลกเปลี่ยนตำแหน่งพิกัดที่มีค่าเป็นเลขจำนวนจริงระหว่าง 0-1

4) การคัดเลือก (Selection)

ค่าการเลือกเป็นขั้นตอนในการเลือกเวกเตอร์เป้าหมายในรอบถัดไปหรือการเลือกประชากรสำหรับการดำเนินในรุ่นต่อไป โดยเวกเตอร์ที่ให้คำตอบที่ดีกว่าจะถูกเลือกโดยใช้วิธีการเปรียบเทียบค่าคำตอบของสมการเป้าหมาย (Objective value) หรือค่าฟิตเนส (Fitness value) ของไทรอัลเวกเตอร์กับเวกเตอร์เป้าหมาย ซึ่งถ้าหากค่าคำตอบของสมการเป้าหมาย หรือค่าฟิตเนสของตัวใดมีค่าที่ดีกว่าก็จะเลือกเวกเตอร์นั้นเพื่อเป็นประชากรในรุ่นถัดไป โดยมีสมการดังแสดงในสมการที่ 2.12

$$X_{i,j,G+1} = \begin{cases} U_{i,j,G} & \text{if } f(U_{i,j,G}) \leq f(X_{i,j,G}) \\ X_{i,j,G} & \text{otherwise} \end{cases} \tag{2.12}$$

$X_{r1,j,G+1}$ = เวกเตอร์เป้าหมาย (Target vector)

$U_{i,j,G}$ = ไทรอัลเวกเตอร์ (Trial vector)

2.2.4.2 ตัวแปรที่ทราบค่าในวิธีวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithm Parameters)

ตัวแปรที่ทราบค่าในการแก้ปัญหาด้วยวิธีวิวัฒนาการโดยใช้ผลต่างนั้น จะมีตัวแปรที่ทราบค่าหลักที่เกี่ยวข้อง คือแฟกเตอร์ในการขยายผลต่าง (Scaling factor) และค่าอัตราแลกเปลี่ยนตำแหน่งพิกัด (CR) โดยมีรายละเอียดของแต่ละตัวแปรดังนี้ (กาญจนา เศรษฐนันท์, 2558)

แฟกเตอร์ในการขยายผลต่าง (Scaling factor) ค่าแฟกเตอร์ในการขยายผลต่างเป็นค่าที่จะทำให้เวกเตอร์แต่ละตัวเปลี่ยนแปลงไปอย่างช้าหรือรวดเร็ว โดยถ้าค่าแฟกเตอร์ในการขยายผลต่างมีค่าสูงจะทำให้มีความรวดเร็วในการหาคำตอบในขอบเขตที่กว้าง แต่อาจทำให้ลดประสิทธิภาพของการค้นหาคำตอบอย่างละเอียดในตำแหน่งใกล้เคียงกับเวกเตอร์เดิม ในทางตรงกันข้ามหากแฟกเตอร์ในการขยายผลต่างมีค่าน้อยจะทำให้การหาคำตอบช้า แต่จะสามารถเพิ่มความละเอียดในการหาคำตอบในพื้นที่ใกล้เคียงได้ดี ซึ่งค่าแฟกเตอร์ในการขยายผลต่างที่นิยมใช้กันจะมีค่าอยู่ในช่วง [0, 2]

2.2.4.3 การปรับปรุงวิธีวิวัฒนาการโดยใช้ผลต่างเพื่อเพิ่มประสิทธิภาพในการหาคำตอบ (Variants Differential Evolution)

เพื่อให้วิธีวิวัฒนาการโดยใช้ผลต่างได้ค่าคำตอบที่ดีที่สุดจึงมีการปรับปรุงวิธีวิวัฒนาการโดยใช้ผลต่างด้วยการปรับปรุงประสิทธิภาพในการหาคำตอบซึ่งจะเน้นการปรับเปลี่ยนวิธีการพัฒนาคำตอบของเวกเตอร์ทั้งในส่วนของการปรับเปลี่ยนและการแลกเปลี่ยนพิกัดเวกเตอร์

1) การปรับปรุงสมการปรับเปลี่ยนพิกัด

การปรับเปลี่ยนพิกัดปกติแล้วจะใช้สมการที่ 2.10 ทั้งนี้ได้มีผู้พัฒนาสมการในการปรับเปลี่ยนเพื่อเพิ่มประสิทธิภาพและสร้างทางเลือกการหาคำตอบมากยิ่งขึ้น Qin and et al

(2009) นำเสนอสมการที่ใช้ในการปรับเปลี่ยนค่าในพิกัดจำนวน 5 สมการ ซึ่งเรียกว่า DE/rand/1 สมการที่ 2.10 เป็นสมการพื้นฐานของขั้นตอนการปรับเปลี่ยนพิกัด ส่วนวิธีของ DE/best/1, DE/rand-to-best/1, DE/best/2, DE/rand/2 เป็นวิธีการปรับปรุงการปรับเปลี่ยนพิกัดโดยสมการใหม่นี้แสดงไว้ในสมการที่ 2.13, 2.14, 2.15, 2.16 ตามลำดับ

$$V_{i,G+1} = X_{best,G} + F(X_{r1,G} - X_{r2,G}) \tag{2.13}$$

$$V_{i,G+1} = X_{i,G} + F(X_{best,G} - X_{i,G}) + F(X_{r1,G} - X_{r2,G}) \tag{2.14}$$

$$V_{i,G+1} = X_{best,G} + F(X_{r1,G} - X_{r2,G}) + F(X_{r3,G} - X_{r4,G}) \tag{2.15}$$

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) + F(X_{r4,G} - X_{r5,G}) \tag{2.16}$$

2) การปรับปรุงสมการการแลกเปลี่ยนค่าในพิกัด

การแลกเปลี่ยนค่าในพิกัดนั้นปกติแล้วจะใช้สมการที่ 2.11 ซึ่งเป็นการแลกเปลี่ยนค่าในพิกัดแบบไบโนเมียล เป็นสมการพื้นฐานของขั้นตอนการแลกเปลี่ยนพิกัด ซึ่งต่อมาได้มีการพัฒนาสมการการแลกเปลี่ยนค่าในพิกัดแบบเอกซ์โพเนนเชียลเกิดขึ้น Rapeepan, P. (2015) นำเสนอการแลกเปลี่ยนค่าในพิกัดแบบเอ็กโพเนนเชียลแบบ 1 จุด และพิกัดแบบเอ็กโพเนนเชียลแบบ 2 จุด ซึ่งเป็นการแลกเปลี่ยนค่าในพิกัดแบบเอ็กโพเนนเชียลแบบ 1 จุด และพิกัดแบบเอ็กโพเนนเชียลแบบ 2 จุด สามารถแสดงได้ดังสมการที่ 2.17 และ 2.18

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{เมื่อ n มีค่าตั้งแต่ 1 ถึง R และ R คือตำแหน่งแรกที่พบ (rand}\mathbf{b}\text{ (j) < CR)} \\ X_{ji,G} & \text{ในกรณีอื่น} \end{cases} \tag{2.17}$$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{เมื่อ n มีค่าตั้งแต่ 1 ถึง R และ R คือตำแหน่งแรกที่พบ (rand}\mathbf{b}\text{ (j) > CR และตำแหน่ง} \\ R_2 & \text{เมื่อ N มีค่าตั้งแต่ } R_1 \text{ คือตำแหน่งแรกที่พบ (rand}\mathbf{b}\text{ (j) > CR) ที่อยู่ถัดจากตำแหน่ง } R_2 \\ X_{ji,G} & \text{ในกรณีอื่น} \end{cases} \tag{2.18}$$

จากการศึกษาทฤษฎีวิธีวิวัฒนาการโดยใช้ผลต่างผู้วิจัยได้ทำการเลือกวิธีวิวัฒนาการโดยใช้ผลต่างขั้นตอนการปรับปรุงสมการการปรับเปลี่ยนพิกัด 3 แบบ ได้แก่ DE/rand/1, DE/rand-to-best/1 และ DE/best/2 ร่วมกับวิธีวิวัฒนาการโดยใช้ผลต่างขั้นตอนปรับปรุงสมการการแลกเปลี่ยนค่าในพิกัด 3 แบบ ได้แก่ การแลกเปลี่ยนค่าในพิกัดแบบไบโนเมียล (Binomial) การแลกเปลี่ยน

ค่าในพิกัดแบบเอ็กโพเนนเชียลแบบ 1 จุด และ การแลกเปลี่ยนค่าในพิกัดแบบเอ็กโพเนนเชียลแบบ 2 จุด ซึ่งจะได้วิธีการปรับปรุงคำตอบโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบใหม่ 5 วิธีเพื่อพัฒนาคำตอบให้ดีกว่าเดิมเมื่อเปรียบเทียบกับวิธีวิวัฒนาการโดยใช้ผลต่างวิธีแบบพื้นฐานที่ใช้วิธีวิวัฒนาการโดยใช้ผลต่างขั้นตอนการแลกเปลี่ยนค่าในพิกัด DE/rand/1 ร่วมกับขั้นตอนการแลกเปลี่ยนค่าในพิกัดแบบไบโนเมียล (Binomial) โดยจะอธิบายทั้ง 5 วิธีปรับปรุงคำตอบแบบใหม่ไว้ในบทที่ 3 ในลำดับถัดไป

2.3 งานวิจัยที่เกี่ยวข้อง

ในช่วงครึ่งศตวรรษที่ผ่านมาได้มีผู้ศึกษางานวิจัยทางด้านการจัดสายการประกอบแบบตัวยูไว้เป็นจำนวนมาก ทั้งภายในประเทศและต่างประเทศ ซึ่งมีรายละเอียดดังนี้

2.3.1 งานวิจัยที่เกี่ยวกับการจัดสมดุลสายการประกอบแบบตัวยูกับวิธีเมตาฮิวริสติกแบบอื่น ๆ

งานวิจัยในปัญหาการสมดุลสายการประกอบแบบตัวยู ได้เริ่มมีการวิจัยเป็นครั้งแรกเมื่อ ค.ศ. 1994 โดย Miltenburg, J. And Wijngaard, J. (1994) ซึ่งได้ทำวิจัยเรื่อง The U-line Line Balancing Problem ที่เป็นการแนะนำรูปแบบทางคณิตศาสตร์ของปัญหาโดยใช้กำหนดการพลวัต (Dynamic programming) สำหรับปัญหา U-line ที่ผลิตผลิตภัณฑ์ชนิดเดียวเพื่อลดจำนวนสถานีงานให้น้อยที่สุด และได้พัฒนาวิธีการ RPWT (Ranked Positional Weight Technique) เพื่อแก้ปัญหาของการสมดุลสายการประกอบแบบ U-line ในปัญหาขนาดใหญ่ (111 ขั้นงาน) ที่เป็นโจทย์ปัญหาจากงานวิจัยอื่น ๆ ที่อ้างอิง ซึ่งจะมีความซับซ้อนมากกว่าปัญหาแบบเดิม เนื่องจากงานสามารถถูกจัดกลุ่มจากข้างหน้าไปข้างหลัง (Forward), และจากข้างหลังไปข้างหน้า (Backward) หรือทั้งสองทิศทางพร้อม ๆ กันตามแผนผังลำดับงานก่อนหลัง และจากนั้น ในปีถัดมา Miltenburg, J. and Sparling, D. (1995) ได้พัฒนาวิธีการทางคณิตศาสตร์ (Exact methods) 3 วิธีการ เพื่อแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูโดยวิธีแรกใช้กำหนดการพลวัต (Dynamic programming formulation: DP) และอีก 2 วิธี คือ วิธี Breadth-1 และ Depth-first branch และ bound (B&B) ผลการคำนวณแสดงให้เห็นว่าวิธี B&B มีประสิทธิภาพที่ดีกว่าวิธี DP-based algorithm และวิธี Breadth-first จะใช้เวลาคำนวณน้อยกว่าวิธี Depth-first แต่วิธี Depth-first จะเจอคำตอบ Optimal เร็วกว่า จากนั้น Sparling, D. and Miltenburg, J. (1998) ได้ทำวิจัยเรื่อง The mixed-model U-line balancing problem ได้พัฒนาวิธีฮิวริสติกเพื่อแก้ปัญหาการจัดสมดุลแบบตัวยูที่ผลิตสินค้าแบบผสม (Mixed- model) ที่มีการรวมผังแสดงลำดับงาน (Precedence graph) ของแต่ละรุ่นผลิตภัณฑ์เข้าด้วยกันที่มี 25 งานย่อย ซึ่งในปีเดียวกัน Uban, T. L. (1998) ได้นำเสนอตัวแบบกำหนดการแบบเลขจำนวนเต็ม (Integer Programming: IP) เพื่อหาค่าสมดุลที่เหมาะสุดสำหรับปัญหาการสมดุลสายการประกอบแบบ U-line ซึ่งตัวแบบนี้สามารถแก้ปัญหาขนาดใหญ่ได้ดีกว่าวิธีการเดิม ต่อจากนั้น Miltenburg, J. (1998) ได้นำเสนอเกี่ยวกับรูปแบบทางคณิตศาสตร์ของปัญหา โดยใช้กำหนดการพลวัต (Dynamic programming) สำหรับปัญหาที่มีหลาย U-lines และมีจำนวนงานไม่เกิน 22 ขั้นงาน วัตถุประสงค์เพื่อจัดงานลงสถานีงานในลักษณะต่าง ๆ ให้มีจำนวนสถานีงานและลดเวลาสูญเปล่าให้น้อยสุดจากหลักการที่เสนอแนะก็เป็นวิธีที่มีประสิทธิภาพในการจัดสมดุลการประกอบ และการสมดุลสายการประกอบใหม่ (Re-balancing) ของปัญหาแบบตัวยูได้ และต่อมา Sparling, D. (1998) ได้พัฒนาวิธีการฮิวริสติก

สำหรับปัญหาการจัดสมดุลแบบตัวยูที่มี 9 U-lines ที่แต่ละ U-line จะมี 18 งานย่อย ที่มีเวลารอบการผลิตเป็นค่าเดียวกันโดยเวลาในระหว่างงานและเวลาในระหว่าง U-lines จะถูกนำมาพิจารณาด้วย และในปีถัดมา Carnahan, B. J. (1999) ทำวิจัยเรื่อง Incorporating Ergonomics Criteria into Assembly Line Balancing ได้พัฒนาวิธีฮิวริสติก 3 วิธี (1. Multiple Ranking heuristic 2. A Combinatorial GA 3. a Problem-Space GA) โดยนำเอาเกณฑ์การออกแบบ ทางการยศาสตร์มาพิจารณาประกอบสำหรับแก้ปัญหา ALB ทั้งแบบ I-Shaped และ U-Shaped line เพื่อลดรอบเวลาการผลิต และความสูญเสียของแรงที่ใช้ในการจับ (Grip Strength) เนื่องจากความเมื่อยล้าในระหว่างสถานีงาน ให้มีค่าน้อยสุด ผลที่ได้จากงานวิจัยทำให้โรงงานอุตสาหกรรมได้รับประโยชน์ ทั้งในด้านการผลิตและทางการยศาสตร์ ต่อจากนั้น Scholl, A. and Klein, R. (1999) ทำวิจัยเรื่อง ULINO-Optimally balancing U-shaped JIT assembly line โดยใช้วิธีการ Branch และ Bound (B&B) ที่พัฒนามาจากวิธี SALOME ที่ใช้ในการแก้ปัญหาแบบเส้นตรง มาใช้สำหรับแก้ปัญหาแบบตัวยู ที่มี 297 ขั้นงาน ซึ่งวิธีการใหม่ คือ ULINO (U-line optimizer) ที่มีพื้นฐานมาจากวิธี depth-first branch และ bound และ dominance rules เพื่อลดจำนวนสถานีงาน หรือ รอบเวลาผลิต หรือ ทั้งสองอย่างให้น้อยสุด และใน ค.ศ. 2003 Chen, S. (2003) ได้พัฒนาวิธีการเพื่อหาผลลัพธ์ที่ดีที่สุดของการจัดสมดุลสายการประกอบแบบตัวยู ที่มีสถานีงานคู่ขนาน โดยใช้วิธี Multiple lower bounding และวิธีฮิวริสติกแบบใหม่ เพื่อหา Upper bounding ผลของวิธีการที่พัฒนาขึ้นทั้ง 2 วิธีแสดงให้เห็นถึงการปรับปรุงที่ดีขึ้นกว่าวิธีการเดิม จากนั้นในปีต่อมา Martinez, U. and Duff, W.S. (2004) ได้ประยุกต์วิธีการเจเนติกอัลกอริทึม (GA) ที่ใช้ในปัญหาการสมดุลสายการประกอบแบบเส้นตรงเพื่อนำมาแก้ปัญหาการสมดุลงานแบบตัวยู ประเภทที่ 1 จากนั้นทำการเปรียบเทียบกับค่าคำตอบที่ดีที่สุดของงานวิจัยที่ผ่านมา ผลจากการวิจัยแสดงให้เห็นว่า วิธีการ GA สามารถหาผลลัพธ์ที่ดีที่สุด หรือคำตอบที่ใกล้เคียงได้ในรอบแรกๆของการทดลอง จากนั้น Gokcen, and et al. (2005) ได้ทำการวิจัยเรื่อง A shortest route formulation of simple U-type assembly line balancing problem ที่ใช้ทฤษฎี เส้นทางที่สั้นที่สุด (Shortest route formulation) ซึ่งนำเสนอวิธีการนี้โดย Gutjahr and Nemhauser (1964) ที่ใช้ในการแก้ปัญหาการสมดุลงานแบบเส้นตรง ที่ได้พัฒนารูปแบบของหลักการดังกล่าวมาใช้ในการแก้ปัญหาการจัดสมดุลงานแบบตัวยู นอกจากนี้ ยังมีตัวอย่างของการคำนวณด้วยวิธีการนี้ ในบทความงานวิจัย และในปีต่อมา Gokcen and Agpak (2006) ได้พัฒนารูปแบบของสมการเป้าหมายสำหรับปัญหาการจัดสมดุลงานแบบตัวยู ที่มีพื้นฐานมาจากรูปแบบกำหนดการเชิงจำนวนเต็มที่พัฒนาโดย Urban (1998) สำหรับปัญหารูปตัวยู และรูปแบบสมการเป้าหมายของในปัญหาแบบเส้นตรง เป็นหลักการที่มีความยืดหยุ่นที่ช่วยในการตัดสินใจในปัญหาการสมดุลงานแบบตัวยู กรณีที่มีหลายเงื่อนไข และในปีเดียวกัน Chiang, W.C. and Urban, T. L. (2006) ได้ทำวิจัยเรื่อง The Stochastic U-line Balancing Problem: A Heuristic Procedure ที่ได้อธิบายถึงเหตุผลที่สายการผลิตแบบตัวยูเป็นที่นิยม เนื่องจากระบบการผลิตแบบ JIT ได้เข้ามามีบทบาท ดังนั้นสายการประกอบแบบ U-shaped ก็ได้มาแทนการจัดสายการประกอบแบบเดิม คือแบบเส้นตรง (Straight assembly lines) จุดที่คำนึงถึงในปัญหาของการจัดสมดุลสายการประกอบแบบ U-shaped คือ ความไม่แน่นอนของเวลางาน (Task time) เนื่องจากพนักงานหรือปัจจัยอื่น ๆ ที่เวลาเป็นแบบไม่แน่นอนโดยมีการพัฒนาวิธีการฮิวริสติก 2 ส่วน คือ การหาผลลัพธ์เบื้องต้นและการปรับปรุงคำตอบ ผลจากการคำนวณแสดง

ให้เห็นถึงประสิทธิภาพของวิธีการที่นำมาใช้ ซึ่งอีกงานวิจัยหนึ่งโดย Erel and et al (2001) ได้พัฒนาวิธีการ Simulated Annealing (SA) มาใช้ในการแก้ปัญหา UALBP ซึ่งกำลังเป็นที่นิยมในปัจจุบัน เป้าหมายเพื่อลดจำนวนสถานีงานลงให้น้อยที่สุด (Type-I) โดยประสิทธิภาพของหลักการนี้จะถูกวัดจากการหาคำตอบในปัญหาที่มีขนาดใหญ่ และเปรียบเทียบผล กับ วิธี ULINO (U-Line optimizer: เป็น B&B based heuristic procedure) ของ Scholl and Klein (1999) นอกจากนี้ นักวิจัยยังได้เสนอแนะทิศทางงานวิจัยที่จะทำต่อไปดังนี้ 1) สามารถใช้วิธีการ SA ในโจทย์ปัญหาที่ยากขึ้น เช่น Mixed/Multi model line, Stochastic task time และ U-line ในลักษณะอื่น ๆ 2) นำเอาหลักการ Metaheuristic วิธีอื่น ๆ เพื่อแก้ปัญหาทาง UALBP 3) พัฒนาวิธีการ Exact method เพื่อแก้ปัญหา U-line ที่เป็นปัญหาขนาดใหญ่ 4) ใช้หลักการ Metaheuristic วิธีอื่น ๆ เพื่อแก้ปัญหาแบบ Type-II (min. c, given m) หลังงานนั้นได้มีผู้วิจัยได้ศึกษาวิธีการแก้ปัญหาในสายการประกอบแบบตัวยูโดยพัฒนาวิธีเมตาฮิวริส GA ส่วนงานวิจัยของ Rea, K. H. and et al. (2008) ได้นำเสนออัลกอริทึมทางพันธุกรรมแบบ (GA) เพื่อแก้ปัญหาสมดุลสายการผลิตแบบตัวยู (UALBP) โดยนำหลักการผลิตแบบ Just-in-time (JIT) มาใช้งานแก้ปัญหาร่วมด้วยในสายการผลิตแบบตัวยู ซึ่งได้ผลลัพธ์ที่ที่ดีกว่าวิธีเดิมที่เคยแก้ปัญหาสายยู จึงได้นำวิธีการแก้ปัญหาของอัลกอริทึม (GA) มาแก้ปัญหาในแบบตัวยู ผลที่ได้การตรวจสอบพบว่ามีประสิทธิภาพมากขึ้นเกณฑ์ประสิทธิภาพที่พิจารณาคือจำนวนเวิร์กสเตชันของสายการผลิต และความแปรผันของปริมาณงาน ผลการทดลองแสดงให้เห็นว่าวิธีที่นำเสนอมีประสิทธิภาพรวมกลุ่มเวิร์กสเตชันได้ดีขึ้นและปรับปรุงรูปแบบของภาระงานดีขึ้น และมีผู้วิจัยหลายท่านได้นำวิธีเมตาฮิวริสติกมาใช้ในการแก้ปัญหาแบบตัวยูโดยผู้วิจัยได้เสนอวิธีอาณานิคมมดในการหาคำตอบ Nuchasara K. and et al (2007) ได้นำเสนอวิธีฮิวริสติก 4 คือ วิธีการของ Kilbridge and Wester (K&W), วิธีการใช้เกณฑ์น้ำหนักเป็นตัวกำหนดตำแหน่ง (Ranked Positional Weight: RPW) วิธีการใช้เกณฑ์เวลามากที่สุด (Maximum Task Time: Max. T.) และวิธีการที่ใช้เกณฑ์จำนวนงานย่อยที่ตามมาทั้งหมดมากที่สุด (Total Maximum Number of Following Tasks: Max. N.F.) โดยทำการเปรียบเทียบกับวิธีเมตาฮิวริสติกวิธีอาณานิคมมดในการหาคำตอบ พบว่าวิธีอาณานิคมมดสำหรับปัญหาการจัดสมดุลสายการประกอบผลคำตอบที่กว่าวิธีฮิวริสติก 4 วิธี จากนั้นผู้วิจัยท่านอื่นได้ศึกษาการแก้ปัญหาสายการประกอบแบบตัวยูโดยวิธีอื่น ๆ ปารเมศ ชุติมา และคณะ (2552) ได้วิจัยและนำเสนออัลกอริทึมในการแก้ปัญหาการจัดสมดุลสายการประกอบลักษณะตัวยู ที่มีการผลิตผลิตภัณฑ์ผสมที่มีหลายวัตถุประสงค์ในระบบผลิตแบบทันเวลาพอดี ภายใต้ปัญหาประเภทที่ 1 โดยจะพิจารณาฟังก์ชันวัตถุประสงค์ทั้งหมด 3 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 จากนั้นผู้วิจัยท่านอื่นได้ศึกษาการแก้ปัญหาสายการประกอบแบบตัวยูโดยวิธีอื่น ๆ Nuchsara and et al (2007a) ได้วิจัยและพัฒนาวิธีฮิวริสติกและวิธีเมตาฮิวริสติก โดยวิธี

ระบบมดแม๊กมิน วิธี Max. task time และ Min. task time ร่วมกับวิธีโลคอลเสิร์ช ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงและตัวยู สำหรับการจัดสมดุลสายการ ประกอบแบบเส้นตรง ได้ทำการทดลองในชุดปัญหาตัวอย่างที่เป็นปัญหาขนาดใหญ่มี 45-111 งาน และชุดปัญหาตัวอย่างของ Lapierre โดยวิธีทาบูเสิร์ช ซึ่งเป็นปัญหาขนาดใหญ่มี 297 งาน ส่วนในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ได้ทำการทดลองในชุดปัญหาตัวอย่าง ในการทดลองโดยวิธี Max. RPW กับปัญหาขนาดกลางที่มี 21-45 งาน และชุดปัญหาตัวอย่าง ข้อมูลบางส่วนที่เป็นปัญหาขนาดใหญ่มี 75-297 งาน จากการวิจัยสรุปผลได้ว่าวิธีระบบมดแม๊กมิน ที่พัฒนาขึ้นเป็นวิธีที่มีประสิทธิภาพดีที่สุดเมื่อเปรียบเทียบกับวิธีฮิวริสติก และวิธี Max. RPW ของ Miltenburg and Wijngaard (1994) โดยจากการทดลองวิธี Min. task time เป็นวิธีที่ให้ค่าคำตอบที่แย่ที่สุด เมื่อเปรียบเทียบกับวิธีการอื่น ๆ และเมื่อมีการเปลี่ยนแปลงของค่ารอบเวลาการผลิตที่เพิ่มขึ้นผล ปรากฏว่า ไม่มีผลต่อความสามารถของการหาคำตอบของวิธีระบบมดแบบแม๊กมิน ดังนั้นสรุปได้ว่า วิธีการที่พัฒนาขึ้นเป็นวิธีที่มีประสิทธิภาพมากวิธีหนึ่ง จากนั้นผู้วิจัยท่านอื่นได้ศึกษาการแก้ปัญหาสายการประกอบแบบตัวยูโดยวิธีอื่น ๆ อมรพงศ์ สงวนสินธุ์ (2557) งานวิจัยนี้ได้วิจัยการจัดสมดุลสายงานการประกอบรูปทรงตัวยู ด้วยวิธีเจเนติกอัลกอริทึม (Genetic Algorithm: GA) ร่วมกับระบบมดแม็ก-มิน (Max-Min Ant System: MMAS) วิธีการที่นำเสนอนี้ GA ทำหน้าที่กำหนดจำนวนขั้นงานที่เหมาะสมให้กับแต่ละสถานีงาน ในขณะที่ MMAS ทำหน้าที่ปรับละเอียดลำดับขั้นงานที่ทำให้ค่าความแปรปรวนของภาระงานของสายงานการประกอบมีค่าน้อยที่สุด โดยสอดคล้องกับเงื่อนไขความสัมพันธ์ก่อน-หลังของขั้นงาน ค่าความแปรปรวนของภาระงาน ถูกกำหนดให้เป็นฟังก์ชันเป้าหมายของกระบวนการค้นหาด้วยวิธีเจเนติกอัลกอริทึม วิธีการที่นำเสนอนี้ถูกทดสอบกับปัญหา UALB จำนวน 24 ปัญหาที่ได้จากการรวบรวมของ Scholl จากผลการทดสอบพบว่า วิธีการที่นำเสนอนี้สามารถลดค่าความแปรปรวนของภาระงานในทุกปัญหา เมื่อเปรียบเทียบกับวิธี COMSOAL และวิธีระบบมดแม็ก-มิน ร่วมกับเทคนิคแบบสุ่ม มีค่าค่าความแปรปรวนของภาระงานมีค่าเฉลี่ยเท่ากับ 81.95% และ 73.16% ตามลำดับ

ซึ่งจากการทบทวนงานวิจัยสายการประกอบแบบตัวยู ที่ผ่านมาสามารถสรุปได้ดังในตารางที่ 2.1

ตารางที่ 2.1 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู

ผู้วิจัย	วิธีการแก้ปัญหา	รูปแบบของปัญหา	จำนวนงานในโจทย์ปัญหา	เป้าหมาย
Miltenburg and Wijngaard (1994)	DP formulation RPWT-based heuristic	Single model	up to 11 tasks up to 111 tasks	m
Miltenburg and Sparling (1995)	DP-based exact alg. Depth-first and breadth-first B&B	Single model	up to 40 tasks	m

ตารางที่ 2.1 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู (ต่อ)

ผู้วิจัย	วิธีการแก้ปัญหา	รูปแบบของปัญหา	จำนวนงานในโจทย์ปัญหา	เป้าหมาย
Sparling and Miltenburg, (1998)	heuristic	Mixed model	up to 25 tasks	m
Urban (1998)	IP formulation	Single model	up to 45 tasks	m
Miltenburg (1998)	DP-based exact alg.	U-line facility with several individual U-lines	Individual U-line with up to 22 tasks	m and idle time in a single st.
Sparling (1998)	heuristic	U-line facility with several individual U-lines	up to 9 U-line with up to 18 tasks in each U-line	m
Scholl and Klein (1999)	B&B based heuristic	Single model	Up to 297 tasks	m
Erel and et al. (2001)	SA	Single model	Up to 297 tasks	m
Martinez and Duff (2004)	Heuristic rules and GA	Single model	Up to 21 tasks	m
Gokcen and et al. (2005)	Shortest route formulation (network programming)	Single model	Up to 7 tasks	m
Gokcen and Agpak (2006)	IP formulation and GP	Single model	Up to 30 tasks	m, c and no. tasks/st
Chaing and Urban (2006)	Hybrid heuristic	Single model stochastic	Up to 111 tasks	m
Hwang Katayama and Gen (2007)	GA	Single model	Up to 111 tasks	m
Nuchsara K. and et al. (2007b)	ACO	Single model	Up to 45 tasks Up to 297 tasks	m

ตารางที่ 2.1 สรุปงานวิจัยที่เกี่ยวข้องกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู (ต่อ)

ผู้วิจัย	วิธีการแก้ปัญหา	รูปแบบของปัญหา	จำนวนงานในโจทย์ปัญหา	เป้าหมาย
Sabuncuoglu and et al. (2009)	ACO	Single model	Up to 75 tasks	m
Zhang and et al (2009)	ACO	Single model	Up to 45 tasks	m
Rabbani and et al. (2012)	GA	Mixed model	Up to 75 tasks	m
**ผู้วิจัย	DE	Single model	Up to 45 tasks Up to 297 tasks	m

หมายเหตุ: ** เป็นงานที่ผู้วิจัยกำลังดำเนินการ

2.3.2 งานวิจัยที่เกี่ยวกับการจัดสมดุลสายการประกอบใช้วิธีวิวัฒนาการโดยใช้ผลต่าง

งานวิจัยที่มีการจัดสมดุลสายการประกอบโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างกับปัญหาการจัดสมดุลสายการประกอบมีผู้วิจัยทำการศึกษา Andreas, C. N. (2005) ได้ศึกษาและนำเสนอวิธีการวิวัฒนาการโดยใช้ผลต่างในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรง (SALB) จากตัวอย่างปัญหาที่มาจาก http://www.assembly-line-balancing.de ที่มีขั้นงาน 7-111 ขั้นงาน ซึ่งมีวัตถุประสงค์เพื่อที่จะหาจำนวนสถานีงานน้อยที่สุด โดยที่เวลาของแต่ละสถานีจะต้องไม่เกินรอบเวลาการผลิต (Cycle time) และเป็นไปตามเงื่อนไขลำดับงานก่อนหลัง ซึ่งมีขั้นตอนการสุ่มตัวเลขค่าจริง 0 ถึง 1 เพื่อสร้าง เวกเตอร์เริ่มต้น ตัวอย่างเช่น 0.23, 0.82, 0.03, 0.47, 0.62 และทำการ Mutation โดยนำหมายเลขตำแหน่งของแต่ละงานหารด้วยจำนวนงานจะได้ว่า งานตำแหน่งที่ 1 มีค่าที่ได้จากการ Mutation เท่ากับ 1/5 เท่ากับ 0.2 ดังนั้นจากตัวอย่างที่มี 5 งาน จะทำการ Mutation ได้ 0.2, 0.4, 0.6, 0.8, 1.0 ตามลำดับจากนั้นทำการ Crossover โดยการนำตัวเลขที่ได้จากการสุ่มโครโมโซมเริ่มต้นมาพิจารณาว่าอยู่ในช่วงใดของค่าที่ได้จากการ Mutation เรียกว่าค่า Sub-Rage (SR) หากอยู่ในช่วงใดให้ใช้ค่า ตำแหน่งงานในช่วงนั้นที่มีค่ามากกว่าหรือเท่ากับในช่วงนั้นมาใช้เป็นคำตอบใหม่เพื่อใช้ในประชากรรุ่นถัดไป เช่น 0.23 อยู่ในช่วง 0.2 ถึง 0.4 (SR = 0.2 < 0.23 < 0.4) ดังนั้นคำตอบที่ได้จากการ Crossover คือ งานในตำแหน่งที่มีค่าจากการกลายพันธ์ 0.4 นั้นคืองานตำแหน่งที่ 2 ดังนั้นคำตอบที่ได้ คืองานที่ 2, _, _, _, _ และ 0.82 อยู่ในช่วง 0.8 ถึง 0.1 คำตอบที่ได้คือ 2, 5, _, _, _ เมื่อพิจารณาจนครบทุกงานจะได้คำตอบเป็น 2, 5, 1, 3, 4 เป็นต้น ผลจากการวิจัยพบว่าวิธีที่นำเสนอให้คำตอบที่เหมาะสมและใช้เวลาน้อยในการหา ซึ่งจากการศึกษาพบว่าขั้นตอนดังกล่าวยังมีจุดที่ไม่สอดคล้องในการทำงานเมื่อใช้กับปัญหาที่มีหลายงานหรือตัวเลขสุ่มเริ่มต้นมีค่าที่ใกล้เคียงกันมาก ๆ เช่น 0.3, 0.23, 0.24, 0.26, 0.8 เมื่อทำการ Crossover จะพบว่า 0.3, 0.23, 0.24, 0.26 จะอยู่ในช่วง 0.2 ถึง 0.4 ทั้งหมด และช่วงที่ 0 ถึง 0.2 จะไม่มีตัวเลขที่สุ่มเริ่มต้นอยู่ในช่วงนั้นเลย ซึ่งจะทำให้คำตอบเริ่มต้นเป็นคำตอบที่ไม่เป็นจริงแต่จะมีการซ่อมแซมคำตอบในภายหลัง หลังจากนั้นผู้วิจัยได้

ทำการศึกษา กนกกาญจน์ จิรศิริเลิศ และระพีพันธ์ ปิตาคะโส (2556) การประยุกต์ใช้วิธีการวิวัฒนาการโดยใช้ผลต่างในการจัดสมดุลสายงานการประกอบแบบเส้นตรงประเภทที่ 1 กรณีศึกษาอุตสาหกรรมผลิตเสื้อผ้าสำเร็จรูป โดยงานวิจัยนี้มีวัตถุประสงค์ในการหาจำนวนสถานีงานที่น้อยที่สุดสำหรับปัญหาการจัดสมดุลสายงานการประกอบเย็บของเสื้อแขนสั้นรุ่น 518729 ของโรงงานกรณีศึกษาโดยศึกษาและประยุกต์ใช้วิธีการวิวัฒนาการโดยใช้ผลต่างสำหรับแก้ปัญหา โดยมีการพิจารณาประเภทเครื่องจักรเป็นเงื่อนไขประกอบในการจัดสมดุลสายงานการประกอบเย็บ และทำการเปรียบเทียบผล กับสายการงานการประกอบเย็บของเสื้อแขนสั้นรุ่น 518729 ปัจจุบันของโรงงานกรณีศึกษา ซึ่งวิธีการวิวัฒนาการโดยใช้ผลต่าง จะทำหน้าที่ในการสร้างเวคเตอร์เริ่มต้นด้วยการสุ่มจากจำนวนจริงและปรับปรุงคำตอบด้วยวิธีการปรับเปลี่ยนค่าในพิกัด วิธีการแลกเปลี่ยนพิกัด และการคัดเลือก จากผลการทดลองประยุกต์ใช้สูตรในขั้นตอนการปรับเปลี่ยนคำในพิกัดทั้ง 3 สูตร พบว่าวิธีการวิวัฒนาการโดยใช้ผลต่างโดยการประยุกต์ใช้สูตรที่ 1 สามารถลดจำนวนสถานีงานได้จากสายงานการประกอบเย็บ ปัจจุบันจาก 23 สถานี เป็น 17 สถานี มีประสิทธิภาพของสายงานการประกอบเย็บเสื้อแขนสั้นรุ่น 518729 เพิ่มขึ้นเท่ากับ 56 เปอร์เซ็นต์ และใช้เวลาประมวลผลได้อย่างรวดเร็วและมีนักวิจัยได้ทำการวิจัยการจัดสมดุลสายการประกอบโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างกับปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงคือ ภาณุภัณฑ์ ภาระเวช (2557) โดยได้ใช้วิธีการดิฟเฟอร์เรนเชียลอิโวลูชั่น (Differential Evolution: DE) เพื่อเพิ่มประสิทธิภาพสายการประกอบ แบบเส้นตรง ซึ่งมีวัตถุประสงค์เพื่อหาจำนวนสถานีงานน้อยที่สุด (m) จัดเป็นปัญหาการจัดสมดุลสายการประกอบแบบเส้นตรงประเภทที่ 1 (Simple Assembly Line Balancing Problem Type 1: SALBP-1) 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 ในการจัดสมดุลสายการประกอบแบบเส้นตรงประเภทที่ 1 ที่มีงาน 7-111 งาน จำนวน 64 ปัญหา จากการทดลองพบว่าการนำวิธี “DE/Best/1” มาใช้ร่วมกับวิธี Exponential Crossover 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 และ “DE/Rand-to-Best/1” ร่วมกับ Exponential Crossover ที่สามารถให้คำตอบที่ดีที่สุดในเวลาที่รวดเร็วที่สุด จำนวน 27, 34, 31, 45 และ 54 ปัญหาตามลำดับ นอกจากนี้ได้นำวิธีวิวัฒนาการโดยใช้ผลต่างมาคิดวิธีใหม่เพื่อหาคำตอบผลลัพธ์ที่มีประสิทธิภาพมากยิ่งขึ้นจึงได้ทำการวิจัย Rapeepan, P. (2015) โดยงานวิจัยได้ใช้อัลกอริธึมวิธีวิวัฒนาการของโดยใช้ผลต่าง (DE) ในการแก้ปัญหาสมดุลสายการประกอบแบบที่ 1 ปัญหาที่นำเสนอประกอบด้วย 4 ขั้นตอนหลัก ได้แก่ (1) การเริ่มต้น (2) การผ่าเหล่า (Mutation) (3) การข้ามสายพันธุ์ (Crossover) (4) กระบวนการคัดเลือก โดยงานวิจัยได้มีการคิดการผสมผสานกันระหว่างกระบวนการการผ่าเหล่า (Mutation) และ การข้ามสายพันธุ์ (Crossover) จึงทำให้ได้วิธีการใหม่เกิดขึ้น ซึ่งผลใน

การให้การคำนวณจากการทดสอบเปรียบเทียบกับชุดปัญหามาตรฐานแสดงให้เห็นว่าอัลกอริธึม DE ที่คิดขึ้นใหม่สามารถแก้ปัญหา SALPB-1 ได้ดีมาก

2.3.3 งานวิจัยที่เกี่ยวกับวิธีวิวัฒนาการโดยใช้ผลต่างในการแก้ปัญหางานวิจัยแบบอื่น ๆ

สำหรับวิธีวิวัฒนาการโดยใช้ผลต่างในปัญหางานวิจัยแบบอื่นเพื่อที่จะหาคำตอบโดยมีดังนี้ Qin, A. K. and et al. (2009) ได้ใช้วิธีการวิวัฒนาการโดยใช้ผลต่างโดยมีวิธีการ Mutation ที่นำพิกัดของเวกเตอร์ที่ดีที่สุดในประชากรแต่ละรุ่นเข้ามาเกี่ยวข้องและใช้วิธี Crossover หรือ Recombination ที่มีการแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ โดยใช้การเปรียบเทียบค่า Crossove Rate และการสุ่มพิกัด ซึ่งพบว่าเป็นวิธีที่มีประสิทธิภาพการหาคำตอบ และในปี ค.ศ.2011 ได้มีผู้วิจัยทำการศึกษา Dexuan Z. and et al (2011) พัฒนาวิธีการวิวัฒนาการโดยใช้ผลต่างสำหรับการแก้ปัญหาการมอบหมายงาน โดยได้ทำการปรับปรุง 2 ส่วนที่สำคัญของพารามิเตอร์ในขั้นตอนของวิธีการวิวัฒนาการโดยใช้ผลต่าง คือ Weighting Factor (F) และค่าของ Crossover Rate (CR) โดยใช้วิธีที่เรียกว่า Improve Differential Evolution (IDE) โดยให้ค่าของ F สามารถปรับค่าได้และค่าของ CR มีการเปลี่ยนค่าเป็นลักษณะเป็นขั้นๆ โดยได้เอาตัวอย่างของปัญหามาเปรียบเทียบคำตอบกับ วิธีการวิวัฒนาการโดยใช้ผลต่างสองวิธีได้แก่ ODE และ JADE ผลปรากฏว่า IDE ที่พัฒนาขึ้นให้ คำตอบดีกว่าทั้งสองวิธีไม่ว่าจะเป็นลักษณะการลดต้นทุนและประสิทธิภาพที่เพิ่มขึ้นในระบบ และในปีถัดมาได้มีผู้วิจัย Xianhui Zeng et al. (2012) ได้พัฒนาวิธี PUDDE หรือ วิธีพาเรโตวิวัฒนาการแบบไม่ต่อเนื่อง ในการจัดการกับปัญหาการจัดสรรผู้ปฏิบัติงานที่เรียกว่าปัญหา Operator Allocation Problems (OAP) เพื่อการจัดสรรงานที่เหมาะสมกับการควบคุมความสมดุลของสายการประกอบเมื่อมีการสร้างฟังก์ชั่นเป้าหมายหลายวัตถุประสงค์และเงื่อนไขและไม่สามารถตัดสินใจได้ในวัตถุประสงค์เดียว มีขั้นตอนคือใช้ วิธี DES Model ในการจำลองสถานการณ์ทั่วไปและใช้ PUDDE ในการแก้ปัญหา OAP โดยการปรับปรุงผู้ปฏิบัติงานมีสองแบบคือ ลดจำนวนหรือเพิ่มจำนวน ผลการทดลองสรุปได้ว่าวิธีการ PUDDE สามารถหาคำตอบได้อย่างมีประสิทธิภาพแต่อย่างไรก็ตามวิธีการนี้เหมาะกับการจัดสรรลดจำนวนผู้ปฏิบัติงาน หากเปรียบเทียบกับวิธี DE แบบเดิมแล้ว วิธีการ PUDDE จะมีประสิทธิภาพที่ดีกว่าในกรณีการหาวัตถุประสงค์ที่มีการผสมผสานของสายงานการประกอบในปัญหาเดียวกันหลัง และในปีถัดมาได้มีผู้วิจัย เอกชัย คุปตาวิน (2555) ทำการวิจัยโดยวัตถุประสงค์เพื่อเปรียบเทียบผลของอัลกอริทึ่มในการแก้ปัญหาการจัดสรรวัตถุดิบ ในการขนส่งมันสำปะหลังแบบหลายระดับชั้น ระหว่าง การสร้างตัวแบบทางคณิตศาสตร์แล้วแก้ปัญหาด้วยโปรแกรมสำเร็จรูป Lingo V.11 และการประยุกต์ใช้วิธีการหาค่าที่เหมาะสมแบบวิวัฒนาการคำตอบ (Differential Evolution: DE) โดยขั้นตอนการวิจัยมีสองขั้นตอน คือ 1) การเปรียบเทียบผลคำตอบในด้านเศรษฐศาสตร์ และ 2) ระยะเวลาในการคำนวณหาคำตอบที่น้อยที่สุด ในการทดสอบอัลกอริทึ่มที่นำเสนอ จะทำการทดสอบกับปัญหาที่ผู้วิจัยทำการจำลองสถานการณ์ขึ้น เพื่อใช้ในการวิจัยทั้งสิ้น 5 กรณีศึกษา จากผลการทดลองพบว่าวิธีการหาคำตอบของ DE แย่กว่าคำตอบที่ได้จากโปรแกรมสำเร็จรูป Lingo V.11 เฉลี่ย 0.281 % แต่ใช้ระยะเวลาในการหาคำตอบที่เร็วกว่าเฉลี่ย 85.4 นาที เมื่อเทียบกับผลที่ได้จากโปรแกรมสำเร็จรูป Lingo V.11 และในปีเดียวกันได้มีผู้วิจัย ทองพูน ทองดี (2555) ทำการวิจัยโดยมีวัตถุประสงค์ คือการเลือกโรงงานที่มีศักยภาพที่สามารถจะเปิดเป็นโรงงานเอทานอลได้โดยโรงงานที่จะเปิดจะเลือกใช้ชานอ้อยจากโรงงานน้ำตาล และกากมันสำปะหลังจากโรงงานมันสำปะหลังอุตสาหกรรม มาเป็นวัตถุดิบของกระบวนการผลิต

ภายใต้วัตถุประสงค์หลักด้านเศรษฐศาสตร์ สิ่งแวดล้อม และความปลอดภัย โดยได้กำหนดขอบเขตในพื้นที่ภาคตะวันออกเฉียงเหนือของประเทศไทยเป็นกรณีศึกษา ซึ่งงานวิจัยนี้เป็นรูปแบบการขนส่งแบบหลายลำดับขั้น หลายสถานที่และหลายวัตถุประสงค์กระบวนการของการแก้ปัญหาเริ่มจากกำหนดแบบจำลองทางคณิตศาสตร์เพื่อเป็นตัวแทนของกรณีศึกษา และจากนั้นทำการแก้ปัญหาด้วยโปรแกรมสำเร็จรูป Lingo V.11 แต่ไม่สามารถหาคำตอบที่ดีที่สุดได้ หาคำตอบได้แค่ที่ใกล้เคียงเท่านั้นโดยที่ใช้เวลากว่า 5 ชั่วโมงในการคำนวณ ดังนั้นในงานวิจัยนี้จึงได้พัฒนาอัลกอริทึมที่มีการแก้ไขวิวัฒนาการ (MODDE) เพื่อแก้ปัญหากรณีศึกษานี้ จากผลการคำนวณพบว่า MODDE ได้คำตอบที่ให้ต้นทุนสูงกว่าประมาณ 5% แต่ใช้เวลาเร็วกว่า Lingo 99%

ซึ่งผู้วิจัยได้ทำการสรุปจากการทบทวนวรรณกรรมและงานวิจัยที่เกี่ยวข้องจะพบว่าวิธีเชิงพันธุกรรมเป็นวิธีที่มีผู้วิจัยได้นิยมนำมาใช้ในการแก้ปัญหาในการจัดสมดุลสายการประกอบ ซึ่งถือได้ว่าเป็นวิธีที่มีประสิทธิภาพมากวิธีหนึ่ง และเมื่อศึกษางานวิจัยที่เกี่ยวกับวิธีเชิงพันธุกรรม ก็พบว่ามีวิธีที่สามารถให้คำตอบได้เทียบเท่ากับวิธีเชิงพันธุกรรมและสามารถหาคำตอบที่เป็นไปได้และใกล้เคียงคำตอบที่ดีที่สุด ในเวลาที่สั้น คือ วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) เป็นวิธีที่สามารถที่จะค้นหาคำตอบที่ดีที่สุด สำหรับปัญหาขนาดใหญ่ที่มีความซับซ้อนของปัญหา ภายในพื้นที่คำตอบที่เป็นไปได้โดยใช้ระยะเวลาที่สั้นในการหาคำตอบเมื่อเปรียบกับวิธีเมตาฮิวริสติกวิธีอื่น ๆ จากงานวิจัยของ Dervis and Selcuk (2004) ได้อธิบายว่า วิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution Algorithms: DEA) เป็นหนึ่งในวิธีการ (Evolutionary Algorithms :EA) ที่มีขั้นตอนวิธีการวิวัฒนาการที่เป็นเทคนิคใหม่ในการเพิ่มประสิทธิภาพ ความสามารถในการจัดการปัญหาประเภท Non-Differentiable, Non-Linear และ Multimodal Objective Functions โดยเฉพาะในปัญหาขนาดใหญ่ที่ซับซ้อน ก็พบว่าความเร็วในหาคำตอบของวิธีการวิวัฒนาการโดยใช้ผลต่างมีประสิทธิภาพที่ดีกว่า ดังนั้นวิธีการวิวัฒนาการโดยใช้ผลต่าง จึงเป็นวิธีการที่สามารถนำมาใช้ในการแก้ปัญหาและช่วยเพิ่มประสิทธิภาพในการหาคำตอบสำหรับปัญหาการจัดสมดุลสายการประกอบได้ ดังนั้นผู้วิจัยจึงได้เลือกวิธีวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) สำหรับการแก้ปัญหาสายการประกอบแบบตัวยู (UALBP-1) ในงานวิจัยครั้งนี้

บทที่ 3
วิธีวิวัฒนาการโดยใช้ผลต่างสำหรับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู

จากที่ผู้วิจัยได้ทำการศึกษางานวิจัยที่เกี่ยวข้องพบว่าวิธีการวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) เป็นวิธีที่สามารถค้นหาคำตอบที่ดีที่สุด สำหรับปัญหาขนาดใหญ่ที่มีความซับซ้อนของปัญหา ภายในพื้นที่คำตอบที่เป็นไปได้ โดยใช้ระยะเวลาที่สั้นในการหาคำตอบเมื่อเปรียบกับวิธีเมตาฮิวริสติกวิธีอื่น ๆ โดยขั้นตอนวิธีการวิวัฒนาการโดยใช้ผลต่างเป็นเทคนิคใหม่ในการเพิ่มประสิทธิภาพและความสามารถในการจัดการสำหรับปัญหาขนาดใหญ่ที่ซับซ้อนในการหาคำตอบและใช้เวลาไม่นาน ดังนั้นวิธีการวิวัฒนาการโดยใช้ผลต่าง จะเป็นวิธีการที่นำมาใช้แก้ปัญหาและช่วยเพิ่มประสิทธิภาพทางในการหาคำตอบสำหรับปัญหาการจัดสมดุลสายการประกอบได้และอีกทั้งวิธีการนี้ยังไม่เคยมีนักวิจัยนำมาใช้ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู (UALBP-1) จึงถือได้ว่าน่าจะเป็นวิธีที่ใหม่ที่ช่วยในการค้นหาคำตอบสำหรับการแก้ปัญหาสายการประกอบแบบตัวยู (UALBP-1) ดังนั้นผู้วิจัยจึงได้เลือกวิธีวิวัฒนาการโดยใช้ผลต่าง (Differential Evolution: DE) สำหรับการแก้ปัญหาสายการประกอบแบบตัวยู (UALBP-1) ในงานวิจัยครั้งนี้

ในบทนี้จะเป็นการอธิบายถึงวิธีวิวัฒนาการโดยใช้ผลต่าง สำหรับใช้ในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 การพัฒนาโปรแกรมสำหรับวิธีการต่าง ๆ ที่เสนอแนะ เพื่อแก้ปัญหาสำหรับการจัดสมดุลสายการประกอบแบบตัวยู

3.1 การจัดสมดุลสายการประกอบแบบตัวยู

รูปแบบของปัญหาการจัดสมดุลสายการประกอบแบบตัวยู (UALBP) โดยปัญหาการจัดสมดุลงานแบบตัวยูทุกงานที่อยู่ก่อนหน้า (Predecessors) และ/หรือ ทุกงานที่อยู่ตามหลัง (Successors) งาน j จะสามารถถูกจัดลงสถานีงานที่ 1, ..., k ได้เพียง 1 สถานีเท่านั้น โดยที่เวลารวมของสถานีงานนั้น ๆ จะต้องมีค่าไม่มากกว่ารอบเวลาการผลิต นั่นคือในการจัดสมดุลสายการประกอบแบบตัวยู สามารถพิจารณางานที่จะจัดลงสถานีงานได้ทั้ง 2 ทิศทาง คือเลือกงานจากข้างหน้าไปข้างหลัง หรือจากซ้ายไปขวา (Forward) ของแผนผังลำดับขั้นงาน (Precedence diagram) และจากข้างหลังไปข้างหน้า หรือจากขวาไปซ้าย (Backward) ได้พร้อม ๆ กันเป็นไปตามเงื่อนไขของจากจัดเข้าสถานีงานตามรูปแบบการจัดสมดุลสายการประกอบแบบตัวยู (UALBP)

3.2 การแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง

การแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยูกระบวนการของวิธีการวิวัฒนาการโดยใช้ผลต่างมีขั้นตอนหลักโดยทั่วไปดังนี้

ขั้นตอนที่ 1 สร้างประชากรเริ่มต้น

ขั้นตอนที่ 2 ผลิตประชากรรุ่นใหม่ด้วยวิธีการดังสมการดังขั้นตอนที่ 2.1-2.3 และทำการวนซ้ำ

ดัชนี (Indics)

F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor)

CR = อัตราการแลกเปลี่ยนค่าในพิกัด (crossover rate)

ขั้นตอนที่ 2.1 สร้างมิวแทนต์เวกเตอร์ โดยการปรับเปลี่ยนค่าในพิกัดของเวกเตอร์ (mutation) โดยสมการนี้เป็นแบบ DE/rand/1 ด้วยสมการที่ 3.1

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

เมื่อ i คือ ลำดับแต่ละเวคเตอร์ของ NP, j คือ พิกัดของแต่ละเวกเตอร์ (vector dimension) G คือ รอบในการวนซ้ำ เช่น รอบที่ 1, 2 หรือ 3 เป็นต้น และ r_1, r_2, และ r_3 คือเวกเตอร์ที่ทำการสุ่มเลือกมา 3 เวกเตอร์ F คือ แฟคเตอร์ในการขยายผลต่าง (Scaling factor) $V_{i,j,G+1}$ คือ มิวแทนเวกเตอร์สำหรับเวกเตอร์ที่ i พิกัดของเวกเตอร์ j ในการวนซ้ำรอบที่ G+1 และ $X_{r1,j,G}$ คือ เวกเตอร์เป้าหมายที่ r_1 ซึ่งเป็นเวกเตอร์ที่สุมเป็นอันดับแรก พิกัดของเวกเตอร์ j ในรอบการวนซ้ำที่ G ส่วน $X_{r2,j,G}$ และ $X_{r3,j,G}$ เป็นเวกเตอร์เป้าหมายที่สุ่มเป็นลำดับที่ 2 และ 3 ตามลำดับ ดังนั้นค่ามิวแทนต์เวกเตอร์ของเวกเตอร์ที่ i พิกัดของเวกเตอร์ j ในการวนซ้ำรอบที่ G+1 มีค่าเท่ากับ ค่าในพิกัดของเวคเตอร์เป้าหมายที่ r_1 พิกัดของเวกเตอร์ j ในการวนซ้ำรอบที่ G รวมกับ ผลต่างของค่าในพิกัดเวกเตอร์เป้าหมายที่ r_2 และ r_3 คูณกับค่าแฟคเตอร์การขยายผลต่าง F

ขั้นตอนที่ 2.2 สร้างไทรอัลเวกเตอร์ โดยวิธีการแลกเปลี่ยนค่าในพิกัดของเวกเตอร์ (recombination) โดยสมการนี้เป็นแบบไบโนเมียล (Binomial Crossover) ดังสมการที่ 3.2

$$U_{i,j,G+1} = \begin{cases} V_{i,j,G+1} & \text{if } (rand_{ij}\ [0,1) \le CR) \text{ or } j = Irand \\ X_{i,j,G+1} & \text{if } (rand_{ij} > CR) \text{ or } j \ne Irand \end{cases} \qquad (3.2)$$

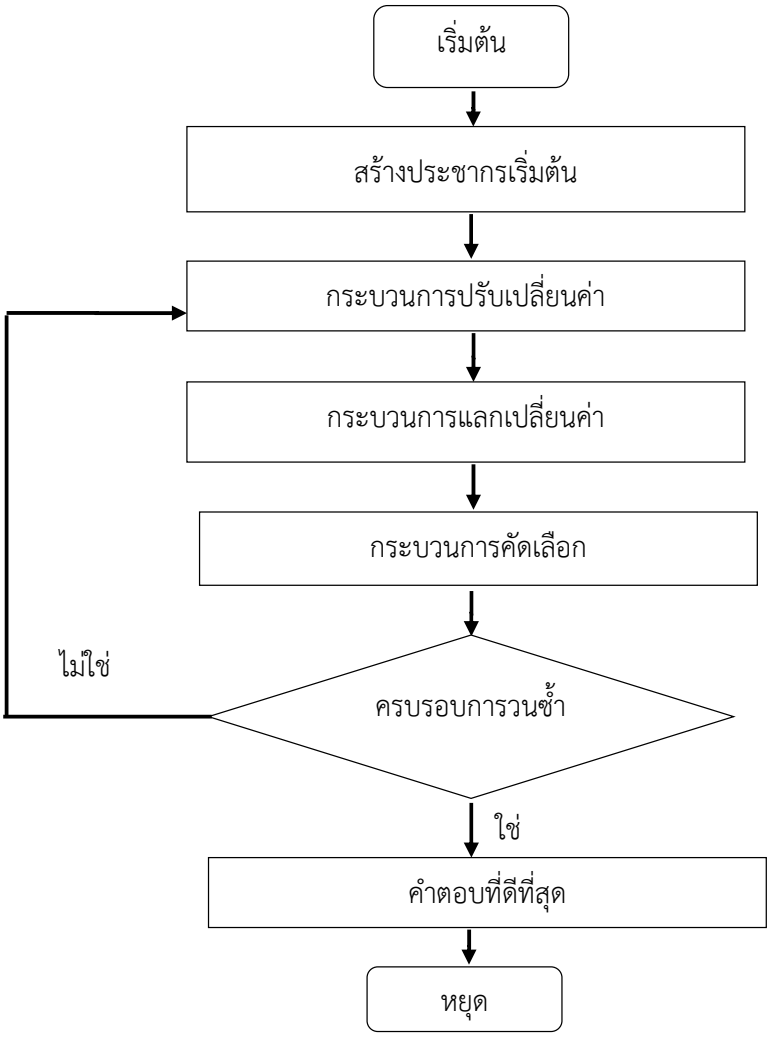
เมื่อ $U_{i,j,G+1}$ คือ ไทรอัลเวคเตอร์ของเวคเตอร์ที่ i พิกัดของเวกเตอร์ j ในรอบการวนซ้ำที่ G+1 $rand_{ij}$ คือ เลขสุ่มจำนวนจริงที่อยู่ในช่วง [0, 1) ของเวกเตอร์ที่ i พิกัดเวคเตอร์ที่ j ค่า CR คือ อัตราการแลกเปลี่ยนค่าในพิกัด (crossover rate) j คือ ตำแหน่งของพิกัดของเวคเตอร์ที่ i และ Irand คือ เลขจำนวนเต็มสุ่มของเวคเตอร์ที่ i ที่มีค่าตั้งแต่ [1, n] เมื่อ n คือ ขนาดของเวคเตอร์ ดังนั้นค่า $U_{i,j,G+1}$ ไทรอัลเวคเตอร์ที่ i พิกัดของเวกเตอร์ j ในการวนรอบซ้ำที่ G จะมีค่าเท่ากับ $V_{i,j,G+1}$ มิวแทนต์เวคเตอร์ หากเลขสุ่มของเวคเตอร์ที่ i พิกัดของเวกเตอร์ j มีค่าน้อยกว่าหรือเท่ากับ CR หรือเมื่อเลขสุ่ม Irand เท่ากับตำแหน่งพิกัด j และจะมีค่าเท่ากับ $X_{i,j,G+1}$ เวคเตอร์เป้าหมาย หากเลขสุ่มของเวคเตอร์ที่ i พิกัดของเวกเตอร์ j มีค่ามากกว่า CR หรือเมื่อเลขสุ่ม Irand ไม่เท่ากับตำแหน่งพิกัด j

ขั้นตอนที่ 2.3 เลือกเวคเตอร์เป้าหมายในรอบถัดไป โดยการคัดเลือกเวคเตอร์ (selection) ด้วยสมการที่ 3.3

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

เมื่อ $X_{ij,G+1}$ คือเวคเตอร์เป้าหมายที่ i พิกัดที่ j ในรอบที่ G+1 จะมีค่าเท่ากับ เวคเตอร์ที่มีค่าฟิตเนสฟังชั่นดีกว่าระหว่างค่าของเวคเตอร์เป้าหมายและค่าของไทรอัลเวคเตอร์ในรอบที่ G

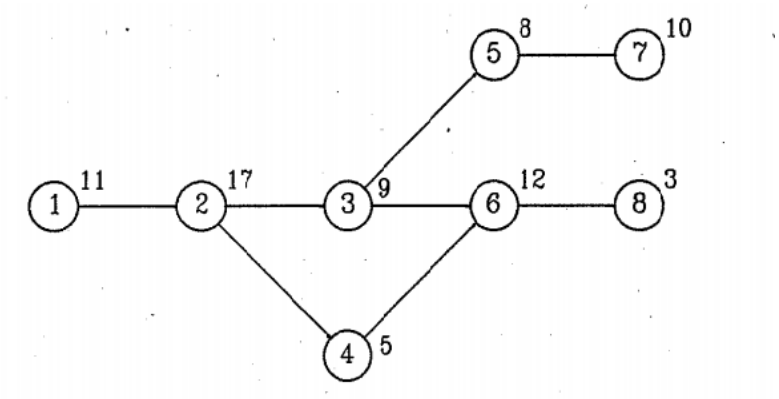
ขั้นตอนที่ 3 หยุดการวนซ้ำ ที่สามารถอธิบายการออกแบบกระบวนการประยุกต์ใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานได้ดังภาพที่ 3.1



ภาพที่ 3.1 กระบวนการวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน

จากกระบวนการวิธีวิวัฒนาการโดยใช้ผลต่าง ดังภาพที่ 3.1 นั้น เริ่มต้นจากการออกแบบเวคเตอร์เริ่มต้น และนำเวคเตอร์ที่ออกแบบไปผ่านกระบวนการปรับเปลี่ยนค่าในพิกัดของเวกเตอร์กระบวนการ

แลกเปลี่ยนค่าในพิกัดของเวคเตอร์ และกระบวนการคัดเลือกเวคเตอร์ ซึ่งสามารถอธิบายลำดับขั้นตอนกับปัญหาตัวอย่าง Bowman problem ในภาพที่ 3.2



ภาพที่ 3.2 แผนผังลำดับขั้นตอนงาน Bowman problem
ที่มา: Bowman (1960)

3.3 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน

จากชุดปัญหาของ Bowman problem ในภาพที่ 3.2 ได้แสดงแผนภาพลำดับงานก่อน-หลังของปัญหาการจัดสมดุลสายการประกอบอย่างง่ายที่มี 8 งานตัวเลขในวงกลมแสดงชื่อของแต่ละงานและตัวเลขบนวงกลมแสดงเวลาของงานนั้น ๆ ซึ่งก่อนที่จะทำการจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณปัญหานี้กำหนดให้ R=1 รอบ, NP=5, F=0.8, CR=0.8 Ronkkonen, Kukkonen and Price (2005) และ CT (Cycle Time) = 20 นาที โดยได้แสดงวิธีการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน แบบ “DE/rand/1” ร่วมกับ Binomial Crossover ได้ดังนี้

3.3.1 แสดงการคำนวณ แบบ “DE/rand/1” ร่วมกับ Binomial Crossover

3.3.1.1 การสร้างคำตอบเริ่มต้น (Initial population) ขั้นตอนการสุ่มจำนวนจริง 0-1 ให้กับแต่ละงาน เป็นการสร้างค่า Target Vector หรือคำตอบเริ่มต้นเพื่อเป็นค่าตัดสินใจในการจัดงานลงสถานีงานและเป็นค่าเริ่มต้นของเวกเตอร์ที่ใช้ในการ Mutation และ Crossover ต่อไป จะได้ดังตารางที่ 3.1 โดยตารางที่ 3.1 จะแสดงการคำนวณการสร้างคำตอบเริ่มต้น (Initial population) โดยทำการสุ่มจำนวนจริง 0-1 ให้กับแต่ละงาน เพื่อเป็นการสร้างค่า Target Vector ให้กับแต่ละงานเพื่อเป็นการสร้างคำตอบเริ่มต้นและเป็นค่าตัดสินใจในการจัดงานลงสถานีงาน โดยการจัดสถานีงานเป็นไปตามเงื่อนไขโดยเรียงลำดับการสุ่มจากเลขน้อยไปมากในการจัดลงสถานีงาน จากตารางกำหนดให้ NP = 5 เวกเตอร์ ซึ่งแสดงให้เห็นจากปัญหาตัวอย่างภาพปัญหาที่ 3.2 จะได้ตารางที่ 3.1 ดังนี้

ตารางที่ 3.1 ผลการสร้างคำตอบเริ่มต้นของปัญหา โดยกำหนด NP=5 เวกเตอร์

Vector	Station								
Vector 1	Station	1		2		3	4		5
	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	**0.3**	**0.57**	**0.44**	**0.61**	**0.72**	**0.53**	**0.68**	**0.92**
Vector 2	Station	1		2		3	4	5	
	Work	7	5	1	8	6	3	4	2
	Time	10	8	11	3	12	12	5	11
	Target Vector	**0.57**	**0.32**	**0.74**	**0.92**	**0.21**	**0.44**	**0.69**	**0.82**
Vector 3	Station	1		2		3		4	5
	Work	1	8	7	5	6	4	3	2
	Time	11	3	10	8	12	5	9	17
	Target Vector	**0.51**	**0.96**	**0.88**	**0.67**	**0.84**	**0.62**	**0.41**	**0.92**
Vector 4	Station	1		2		3	4		5
	Work	1	8	7	5	2	3	4	6
	Time	11	3	10	8	17	9	5	12
	Target Vector	**0.13**	**0.26**	**0.58**	**0.81**	**0.64**	**0.42**	**0.21**	**0.69**
Vector 5	Station	1			2		3		4
	Work	8	6	4	7	5	3	1	2
	Time	3	12	5	11	8	9	11	17
	Target Vector	**0.84**	**0.41**	**0.59**	**0.91**	**0.76**	**0.32**	**0.98**	**0.48**

3.3.1.2 Mutation ขั้นตอนการการปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากรเดิม โดยจุดประสงค์เพื่อการผ่าเหล่า ทำการคำนวณหา Mutant Vector ($V_{i,j,G+1}$) ดังสมการที่ 3.4 และสามารถแสดงตัวอย่างการ Mutation ดังตารางที่ 3.2 ดังนี้

$$V_{i,j,G+1} = X_{r1,j,G} + F(X_{r2,j,G} - X_{r3,j,G}) \tag{3.4}$$

เมื่อ

$V_{i,j,G+1}$ = มิวแทนเวกเตอร์ (Mutant Vector)

$X_{r1,j,G}$, $X_{r2,j,G}$, $X_{r3,j,G}$ = สุ่มเวคเตอร์จากรอบที่ G

F = Scaling factor เป็นเลขสุ่มจำนวนจริงตั้งแต่ 0-2

ตารางที่ 3.2 ผลการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี DE/rand/1

Vector		Position							
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
Vector 2	Work	7	5	1	8	6	3	4	2
	Time	10	8	11	3	12	12	5	11
	Target Vector	0.57	0.32	0.74	0.92	0.21	0.44	0.69	0.82
Vector 3	Work	1	8	7	5	6	4	3	2
	Time	11	3	10	8	12	5	9	17
	Target Vector	0.51	0.96	0.88	0.67	0.84	0.62	0.41	0.92
Mutant Vector 1		**0.35**	**0.06**	**0.33**	**0.81**	**0.22**	**0.39**	**0.90**	**0.84**

จากตารางที่ 3.2 แสดงผลของการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี “DE/rand/1” ซึ่งสุ่ม $X_{r1,j,G}$, $X_{r2,j,G}$ และ $X_{r3,j,G}$ ได้เวกเตอร์ที่ 1, 2 และ 3 ตามลำดับและกำหนดให้ค่า F = 0.8 เช่น พิกัด (Position) ที่ 6 $X_{r1,j,G}$ = 0.53, $X_{r2,j,G}$ = 0.44 และ $X_{r3,j,G}$ = 0.62 แทนค่าในสูตร $V_{i,G+1}= X_{r1,j,G} +F(X_{r2,j,G} - X_{r3,j,G})$ จะได้ $V_{i,G+1}$= 0.53+0.8(0.44-0.62) = 0.39 ดังนั้น Mutant Vector ที่ 1 พิกัดที่ 6 จึงเท่ากับ 0.39 หลังจากนั้นทำการคำนวณเหมือนกันทุกพิกัดจนครบ 8 พิกัด

3.3.1.3 Crossover หรือ Recombination ขั้นตอนวิธีการแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ ซึ่งจะได้เวกเตอร์ใหม่ของคำตอบที่มีทั้งดีกว่าและแย่กว่าอย่างหลากหลาย โดยมีการสร้าง Trial Vector ($U_{i,j,G+1}$) ดังสมการที่ 3.5 และทำการเปรียบเทียบพิจารณาในการแลกเปลี่ยนพิกัดดังสมการที่ 3.6 ซึ่งสามารถแสดงตัวอย่างดังตารางที่ 3.3 และ 3.4 ดังนี้

$$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, \dots , U_{Di,G+1}) \tag{3.5}$$

$$U_{i,j,G+1} = \begin{cases} V_{i,j,G+1} \text{ if (rand j [0, 1)} \le \text{CR)} \\ X_{i,j,G+1} \text{ if (rand (j)} > \text{CR)} \end{cases} \tag{3.6}$$

เมื่อ $V_{i,G+1}$ = Mutant Vector

$X_{i,j,G+1}$ = Target Vector

CR = Crossover Constant จำนวนจริงมีค่า 0 ถึง 1

rand j [0,1) = สุ่มค่าจริง 0 – 1 ทุกพิกัด, j = 1, 2, 3 ,..., G (G = จำนวนพิกัด)

ตารางที่ 3.3 ผลการ Binomial Crossover เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี DE/rand/1

Vector	Position								
	Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
	Mutant Vector	0.35	0.06	0.33	0.81	0.22	0.39	0.90	0.84
	Rand (j)	0.40	0.40	0.06	0.96	0.47	0.40	0.94	0.33
	Trial Vector	**0.35**	**0.06**	**0.33**	**0.61**	**0.22**	**0.39**	**0.68**	**0.84**

จากตารางที่ 3.3 แสดงผลของการ Binomial Crossover เวกเตอร์ที่ 1 ซึ่งกำหนดให้ค่า CR = 0.8 ดังนั้น Trial Vector ของพิกัดที่ 4 และ 7 จะใช้ค่าของ Target Vector ส่วนตำแหน่งอื่น ๆ ที่เหลือจะใช้ค่าของ Mutant Vector

ตารางที่ 3.4 ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.3

Vector	Station								
Vector 1	Station	1		2		3	4		5
	Work	8	6	1	3	2	4	5	7
	Time	3	12	11	9	17	5	8	10
	Trial Vector	0.06	0.33	0.35	0.39	0.22	0.61	0.68	0.84

จากตารางที่ 3.4 เมื่อนำค่า Trial Vector ที่ได้มาจัดสมดุลสายการประกอบแบบตัวยูตามเงื่อนไขลำดับงานก่อนหลังและเวลารวมแต่ละสถานีงานต้องไม่เกินรอบเวลาผลิตโดยพิจารณา Trial Vector ที่มีค่าต่ำก่อนลงสถานีงานทำให้ได้คำตอบ 5 สถานี

3.3.1.4 Selection ขั้นตอนการคัดเลือกประชากรในรุ่นต่อไป (G+1) ซึ่งจะคัดเลือกเอาแต่เฉพาะ คำตอบที่ดีกว่า โดยการเปรียบเทียบผลของ Target Vector กับ Trial Vector ในกรณีที่จำนวนสถานีงานที่ได้ของ Trial Vector ต่ำกว่าหรือเท่ากับ Target Vector ดังนั้นจะเลือก Trial Vector เป็นประชากรในรุ่นต่อไป ดังสมการที่ 3.7

$$X_{i,j,G+1} = \begin{cases} U_{i,j,G} \text{ if } (U_{i,j,G}) < f(X_{i,j,G}) \\ X_{i,j,G} \text{ Otherwise} \end{cases} \quad (3.7)$$

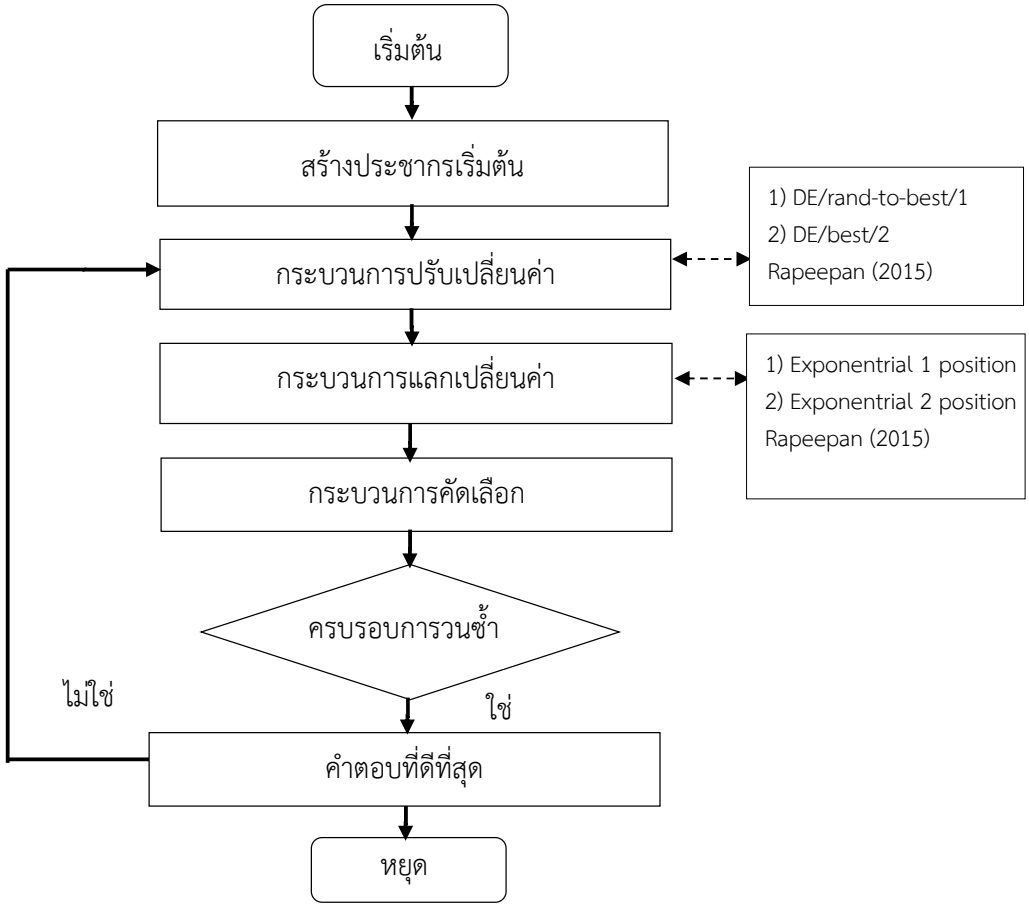
เมื่อ $U_{i,j,G}$ = Trial Vector

$X_{i,j,G+1}$ = Traget Vector ในรุ่นถัดไป, i = 1,2,...n

สรุปได้ว่าจากการขั้นตอนของคัดเลือกประชากรในรุ่นต่อไป Trial Vector ของสถานีที่จัดได้ 5 สถานีงานซึ่งยังมีค่ามากกว่า Target Vector ที่จัดได้ 4 สถานีงานจึงทำให้เลือก Target Vector เป็นคำตอบที่ดีที่สุด และทำการวนรอบซ้ำใหม่ในรุ่นถัดไป

3.4 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ

ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ สามารถอธิบายการออกแบบกระบวนการประยุกต์ใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบได้ดังภาพที่ 3.3



ภาพที่ 3.3 กระบวนการวิธีวิวัฒนาการโดยใช้ผลต่างปรับปรุงคำตอบ

จากชุดปัญหาของ Bowman problem ในภาพที่ 3.2 ได้แสดงแผนภาพลำดับงานก่อน-หลังของปัญหาการจัดสมดุลสายการประกอบอย่างง่ายที่มี 8 งานตัวเลขในวงกลมแสดงชื่อของแต่ละงานและตัวเลขบนวงกลมแสดงเวลาของงานนั้น ๆ ซึ่งก่อนที่จะทำการจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณปัญหานี้กำหนดให้ R=1 รอบ NP=5 F=0.8 CR=0.8 Ronkkonen Kukkonen and Price (2005) และ CT (Cycle Time) = 20 นาที โดยได้แสดงวิธีการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ

การพัฒนาประยุกต์ใช้วิธีการวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ โดยการนำวิธีการวิวัฒนาการโดยใช้ผลต่างขั้นตอนกระบวนการปรับเปลี่ยนค่า Mutation มาใช้ร่วมกับขั้นตอน ได้แก่วิธี DE/rand/1, DE/rand-to-best/1 และ DE/best/2 กระบวนการแลกเปลี่ยนค่า Crossover หรือ Recombination ได้แก่วิธี Binomial Crossover, Exponential Crossover 1 position และ Exponential Crossover 2 position ได้วิธีการใหม่แล้วนำวิธีการใหม่ที่เกิดขึ้นวิวัฒนาการโดยผลต่างในการปรับปรุงคำตอบออกมาได้ 5 วิธี ดังนี้ Rapeepan, (2015)

(1) “DE/rand/1” ร่วมกับ Exponential Crossover 1 Position
(2) “DE/rand/1” ร่วมกับ Exponential Crossover 2 Position
(3) “DE/ rand-to-best/1” ร่วมกับ Binomial Crossover
(4) “DE/Best/2” ร่วมกับ Exponential Crossover 1 Position
(5) “DE/Best/2” ร่วมกับ Exponential Crossover 2 Position

โดยได้แสดงวิธีการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบทั้ง 5 วิธีดังนี้ แสดงการคำนวณจากชุดปัญหาเดิมของ Bowman problem ในภาพที่ 3.2

3.4.1 การคำนวณ แบบ “DE/rand/1” ร่วมกับ Exponential 1 Position

3.4.1.1 การสร้างคำตอบเริ่มต้น (Initial population) ขั้นตอนการสุ่มจำนวนจริง 0-1 ให้กับแต่ละงาน เป็นการสร้างค่า Target Vector หรือคำตอบเริ่มต้นเพื่อเป็นค่าตัดสินใจในการจัดงานลงสถานีงานและเป็นค่าเริ่มต้นของเวกเตอร์ที่ใช้ในการ Mutation และ Crossover ต่อไป ซึ่งแสดงให้เห็นจากปัญหาตัวอย่างภาพที่ 3.2 และตารางที่ 3.1

3.4.1.2 Mutation ขั้นตอนการการปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากรเดิม เรียกว่า Scaling Factor: F หรือเรียกว่า Mutation Factor: F เพื่อจุดประสงค์ของการผ่าเหล่า โดยคำนวณหา Mutant Vector ($V_{i,G+1}$) ดังสมการที่ 3.4 และสามารถแสดงตัวอย่างการ Mutation ดังตารางที่ 3.2

3.4.1.3 Crossover หรือ Recombination ขั้นตอนวิธีการแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ ซึ่งจะได้เวกเตอร์ใหม่ของคำตอบที่มีทั้งดีกว่าและแย่กว่าอย่างหลากหลาย โดยมีการสร้าง Trial Vector ($U_{ji,G+1}$) ดังสมการที่ 3.8 และทำการเปรียบเทียบพิจารณาในการแลกเปลี่ยนพิกัดดังสมการที่ 3.9 ซึ่งสามารถแสดงตัวอย่างดังตารางที่ 3.5 และ 3.6 ดังนี้

$$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, \ldots , U_{Di,G+1}) \quad (3.8)$$

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

เมื่อ

$V_{ji,G+1}$ = Mutant Vector,

$X_{ji,G+1}$ = Target Vector

CR = Crossover Constant จำนวนจริงมีค่า 0 ถึง 1

rand(j) = การสุ่มตำแหน่งพิกัดในเวกเตอร์, j = 1,2,3,...,G (G=จำนวนพิกัด) และ

r1 = พิกัดที่ถูกสุ่มตำแหน่งที่ 1

ตารางที่ 3.5 ผลการ Exponential Crossover 1 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี "DE/rand/1"

Vector		Position							
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
	Mutant Vector	0.35	0.06	0.33	0.81	0.22	0.39	0.90	0.84
	Rand (j)	-	2	-	-	-	-	-	-
	Trial Vector	**0.35**	**0.06**	**0.44**	**0.61**	**0.72**	**0.53**	**0.68**	**0.92**

จากตารางที่ 3.5 แสดงผลของ Exponential Crossover 1 Position เวกเตอร์ที่ 1 ซึ่งสุ่มพิกัดได้ r1 = 2 ดังนั้น Trial Vector ของพิกัดที่ 1 - 2 จะใช้ค่าของ Mutant Vector ส่วนตำแหน่งอื่น ๆ ที่เหลือจะใช้ค่าของ Target Vector

ตารางที่ 3.6 ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.5

Vector		Station							
Vector 1	Station	**1**		**2**		**3**	**4**		**5**
	Work	8	1	6	4	2	3	5	7
	Time	3	11	12	5	17	9	8	10
	Trial Vector	0.06	0.35	0.44	0.61	0.72	0.53	0.68	0.92

จากตารางที่ 3.6 เมื่อนำค่า Trial Vector ที่ได้มาจัดสมดุลสายการประกอบแบบตัวยูตามเงื่อนไขลำดับงานก่อนหลังและเวลารวมแต่ละสถานีงานต้องไม่เกินรอบเวลาผลิตโดยพิจารณา Trial Vector ที่มีค่าต่ำก่อนทำให้ได้คำตอบ 5 สถานี

3.4.1.4 Selection ขั้นตอนการคัดเลือกประชากรในรุ่นต่อไป (G+1) ซึ่งจะคัดเลือกเอาแต่เฉพาะ คำตอบที่ดีกว่า โดยการเปรียบเทียบผลของ Target Vector กับ Trial Vector ในกรณีที่จำนวนสถานีงานที่ได้ของ Trial Vector ต่ำกว่าหรือเท่ากับ Target Vector ดังนั้นจะเลือก Trial Vector เป็นประชากรในรุ่นต่อไป ดังสมการที่ 3.7 สรุปได้ว่าจากการขั้นตอนของคัดเลือกประชากรในรุ่นต่อไป Trial Vector ของสถานีที่จัดได้ 5 สถานีงานซึ่งยังมีค่ามากกว่า Target Vector ที่จัดได้ 4 สถานีงานจึงทำให้เลือก Target Vector เป็นคำตอบที่ดีที่สุด และทำการวนรอบซ้ำใหม่ในรุ่นถัดไป

3.4.2 การคำนวณ แบบ “DE/rand/1” ร่วมกับ Exponential 2 Position

3.4.2.1 การสร้างคำตอบเริ่มต้น (Initial population) ขั้นตอนการสุ่มจำนวนจริง 0-1 ให้กับแต่ละงาน เป็นการสร้างค่า Target Vector หรือคำตอบเริ่มต้นเพื่อเป็นค่าตัดสินใจในการจัดงานลงสถานีงานและเป็นค่าเริ่มต้นของเวกเตอร์ที่ใช้ในการ Mutation และ Crossover ต่อไป ซึ่งแสดงให้เห็นจากปัญหาตัวอย่างภาพที่ 3.2 และตารางที่ 3.1

3.4.2.2 Mutation ขั้นตอนการการปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากรเดิม เรียกว่า Scaling Factor: F หรือเรียกว่า Mutation Factor: F เพื่อจุดประสงค์ของการผ่าเหล่า โดยคำนวณหา Mutant Vector ($V_{i,G+1}$) ดังสมการที่ 3.4 และสามารถแสดงตัวอย่างการ Mutation ดังตารางที่ 3.2 ดังนี้

3.4.2.3 Crossover หรือ Recombination ขั้นตอนวิธีการแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ ซึ่งจะได้เวกเตอร์ใหม่ของคำตอบที่มีทั้งดีกว่าและแย่กว่าอย่างหลากหลาย โดยมีการสร้าง Trial Vector ($U_{ji,G+1}$) ดังสมการที่ 3.10 และทำการเปรียบเทียบพิจารณาในการแลกเปลี่ยนพิกัดดังสมการที่ 3.11 ซึ่งสามารถแสดงตัวอย่างดังตารางที่ 3.7 และ 3.8 ดังนี้

$$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, \ldots , U_{Di,G+1}) \tag{3.10}$$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} \text{ if } (\text{rand } (j) \leq r1 \text{ , } (j) > r2 \\ X_{ji,G+1} \quad \text{otherwise} \end{cases} \tag{3.11}$$

เมื่อ

$V_{ji,G+1}$ = Mutant Vector,

$X_{ji,G+1}$ = Target Vector

CR = Crossover Constant จำนวนจริงมีค่า 0 ถึง 1

Rand (j) = การสุ่มตำแหน่งพิกัดในเวกเตอร์, j = 1, 2, 3, ..., G (G=จำนวนพิกัด) และ r1, r2 = พิกัดที่ถูกสุ่มตำแหน่งที่ 1 และ 2

ตารางที่ 3.7 ผลการ Exponential Crossover 2 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี "DE/rand/1"

Vector	Position								
	Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
	Mutant Vector	0.35	0.06	0.33	0.81	0.22	0.39	0.90	0.84
	Rand (j)	-	2	-	-	5	-	-	-
	Trial Vector	**0.35**	**0.06**	**0.44**	**0.61**	**0.72**	**0.39**	**0.90**	**0.84**

จากตารางที่ 3.7 แสดงผลของ Exponential Crossover 2 Position เวกเตอร์ที่ 1 ซึ่งสุ่มพิกัดได้ r1 = 2 และ r2 = 5 ดังนั้น Trial Vector ของพิกัดที่ 1 - 2 และ 6 - 8 จะใช้ค่าของ Mutant Vector ส่วนตำแหน่งอื่น ๆ ที่เหลือจะใช้ค่าของ Target Vector

ตารางที่ 3.8 ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.7

Vector	Station								
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	1	6	4	2	3	7	5
	Time	3	11	12	5	17	9	10	8
	Trial Vector	0.06	0.35	0.44	0.61	0.72	0.39	0.84	0.90

จากตารางที่ 3.8 เมื่อนำค่า Trial Vector ที่ได้มาจัดสมดุลสายการประกอบแบบตัวยูตามเงื่อนไขลำดับงานก่อนหลังและเวลารวมแต่ละสถานีงานต้องไม่เกินรอบเวลาผลิตโดยพิจารณา Trial Vector ที่มีค่าต่ำก่อนทำให้ได้คำตอบ 5 สถานี

3.4.2.4 Selection ขั้นตอนการคัดเลือกประชากรในรุ่นต่อไป (G+1) ซึ่งจะคัดเลือกเอาแต่เฉพาะ คำตอบที่ดีกว่า โดยการเปรียบเทียบผลของ Target Vector กับ Trial Vector ในกรณีที่จำนวนสถานีงานที่ได้ของ Trial Vector ต่ำกว่าหรือเท่ากับ Target Vector ดังนั้นจะเลือก Trial Vector เป็นประชากรในรุ่นต่อไป ดังสมการที่ 3.7 สรุปได้ว่าจากการขั้นตอนของคัดเลือกประชากรในรุ่นต่อไป Trial Vector ของสถานีที่จัดได้ 5 สถานีงานซึ่งยังมีค่ามากกว่า Target Vector ที่จัดได้ 4 สถานีงานจึงทำให้เลือก Target Vector เป็นคำตอบที่ดีที่สุด และทำการวนรอบซ้ำใหม่ในรุ่นถัดไป

3.4.3 การคำนวณ แบบ "DE/rand-to-best/1 "ร่วมกับ Binomial Crossover

3.4.3.1 การสร้างคำตอบเริ่มต้น (Initial population) ขั้นตอนการสุ่มจำนวนจริง 0-1 ให้กับแต่ละงาน เป็นการสร้างค่า Target Vector หรือคำตอบเริ่มต้นเพื่อเป็นค่าตัดสินใจในการจัดงาน

ลงสถานีงานและเป็นค่าเริ่มต้นของเวกเตอร์ที่ใช้ในการ Mutation และ Crossover หรือ Recombination ต่อไป ซึ่งแสดงให้เห็นจากปัญหาตัวอย่างภาพที่ 3.2 และตารางที่ 3.1

3.4.3.2 Mutation ขั้นตอนการการปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากรเดิม เรียกว่า Scaling Factor: F หรือเรียกว่า Mutation Factor: F เพื่อจุดประสงค์ของการผ่าเหล่า โดยคำนวณหา Mutant Vector ($V_{i,G+1}$) ดังสมการที่ 3.12 และสามารถแสดงตัวอย่างการ Mutation ดังตารางที่ 3.9 ดังนี้

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

เมื่อ

F = Scaling factor เป็นจำนวนจริงตั้งแต่ 0-2

$X_{i,G}$ = เวคเตอร์เป้าหมายในรอบที่ G

$X_{best,G}$ = เวคเตอร์ที่ดีที่สุดในรอบที่ G

$X_{r1,G}$, $X_{r2,G}$ = สุ่มเวคเตอร์จากรอบที่ G

ตารางที่ 3.9 ผลการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี “DE/rand-to-best/1”

Vector	Position								
	Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
Vector 2	Work	7	5	1	8	6	3	4	2
	Time	10	8	11	3	12	12	5	11
	Target Vector	0.57	0.32	0.74	0.92	0.21	0.44	0.69	0.82
Vector 3	Work	1	8	7	5	6	4	3	2
	Time	11	3	10	8	12	5	9	17
	Target Vector	0.51	0.96	0.88	0.67	0.84	0.62	0.41	0.92
Vector 5 (best)	Work	8	6	4	7	5	3	1	2
	Time	3	12	5	11	8	9	11	17
	Target Vector	0.84	0.41	0.59	0.91	0.76	0.32	0.98	0.48
Mutant Vector 1		**0.78**	**-0.07**	**0.45**	**1.05**	**0.25**	**0.22**	**1.14**	**0.49**

จากตารางที่ 3.9 แสดงผลของการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี “DE/rand-to-best/1” ซึ่งสุ่ม $X_{r1,G}$, $X_{r2,G}$ ได้เวกเตอร์ที่ 2 และ 3 ตามลำดับและกำหนดให้ค่า F = 0.8 เช่น พิกัด (Position) ที่ 5 $X_{r1,G}$ = 0.21, $X_{r2,G}$ = 0.84, $X_{i,G}$ = 0.72 และ $X_{best,G}$ = 0.76 แทนค่าในสมการ

$V_{i,G+1} = X_{i,G} + F(X_{best,G} - X_{i,G}) + F(X_{r1,G} - X_{r2,G})$ จะได้ $V_{i,G+1}$ = 0.72+0.8(0.76-0.72)+0.8(0.21-0.84) = 0.25 ดังนั้น Mutant Vector ที่ 1 พิกัดที่ 5 จึงเท่ากับ 0.25 หลังจากนั้นทำการคำนวณเหมือนกันทุกพิกัดจนครบ 8 พิกัด

3.4.3.3 Crossover หรือ Recombination ขั้นตอนวิธีการแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ ซึ่งจะได้เวกเตอร์ใหม่ของคำตอบที่มีทั้งดีกว่าและแย่กว่าอย่างหลากหลาย โดยมีการสร้าง Trial Vector ($U_{ji,G+1}$) ดังสมการที่ 3.13 และทำการเปรียบเทียบพิจารณาในการแลกเปลี่ยนพิกัดดังสมการที่ 3.14 ซึ่งสามารถแสดงตัวอย่างดังตารางที่ 3.10 และ 3.11 ดังนี้

$$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, \ldots, U_{Di,G+1}) \tag{3.13}$$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} \text{ if (rand j [0 , 1)} \le CR) \\ X_{ji,G+1} \text{ if (rand (j)} > CR) \end{cases} \tag{3.14}$$

เมื่อ

$V_{ji,G+1}$ = Mutant Vector,

$X_{ji,G+1}$ = Target Vector

CR = Crossover Constant จำนวนจริงมีค่า 0 ถึง 1

rand j [0, 1) = สุ่มค่าจริง 0 – 1 ทุกพิกัด, j = 1, 2, 3, ..., G (G = จำนวนพิกัด)

ตารางที่ 3.10 ผลการ Binomial Crossover เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี “DE/rand-to-best/1”

Vector	Position								
	Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
	Mutant Vector	0.78	-0.07	0.45	1.05	0.25	0.22	1.14	0.49
	Rand (j)	0.40	0.40	0.06	0.96	0.47	0.40	0.94	0.33
	Trial Vector	**0.78**	**-0.07**	**0.45**	**0.61**	**0.25**	**0.22**	**0.68**	**0.49**

จากตารางที่ 3.10 แสดงผลของการ Binomial Crossover เวกเตอร์ที่ 1 ซึ่งกำหนดให้ค่า CR = 0.8 ดังนั้น Trial Vector ของพิกัดที่ 4 และ 7 จะใช้ค่าของ Target Vector ส่วนตำแหน่งอื่น ๆ ที่เหลือจะใช้ค่าของ Mutant Vector

ตารางที่ 3.11 ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.10

Vector	Station								
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	6	4	1	2	3	7	5
	Time	3	12	5	11	17	9	10	8
	Trial Vector	-0.07	0.45	0.61	0.78	0.25	0.22	0.49	0.68

จากตารางที่ 3.11 เมื่อนำค่า Trial Vector ที่ได้มาจัดสมดุลสายการประกอบแบบตัวยูตามเงื่อนไขลำดับงานก่อนหลังและเวลารวมแต่ละสถานีงานต้องไม่เกินรอบเวลาผลิตโดยพิจารณา Trial Vector ที่มีค่าต่ำก่อนทำให้ได้คำตอบ 5 สถานี

3.4.3.4 Selection ขั้นตอนการคัดเลือกประชากรในรุ่นต่อไป (G+1) ซึ่งจะคัดเลือกเอาแต่เฉพาะ คำตอบที่ดีกว่า โดยการเปรียบเทียบผลของ Target Vector กับ Trial Vector ในกรณีที่จำนวนสถานีงานที่ได้ของ Trial Vector ต่ำกว่าหรือเท่ากับ Target Vector ดังนั้นจะเลือก Trial Vector เป็นประชากรในรุ่นต่อไป ดังสมการที่ 3.7 สรุปได้ว่าจากการขั้นตอนของคัดเลือกประชากรในรุ่นต่อไป Trial Vector ของสถานีที่จัดได้ 5 สถานีงานซึ่งยังมีค่ามากกว่า Target Vector ที่จัดได้ 4 สถานีงานจึงทำให้เลือก Target Vector เป็นคำตอบที่ดีที่สุด และทำการวนรอบซ้ำใหม่ในรุ่นถัดไป

3.4.4 การคำนวณ แบบ “DE/best/2” ร่วมกับ Exponential 1 Position

3.4.4.1 การสร้างคำตอบเริ่มต้น (Initial population) ขั้นตอนการสุ่มจำนวนจริง 0-1 ให้กับแต่ละงาน เป็นการสร้างค่า Target Vector หรือคำตอบเริ่มต้นเพื่อเป็นค่าตัดสินใจในการจัดงานลงสถานีงานและเป็นค่าเริ่มต้นของเวกเตอร์ที่ใช้ในการ Mutation และ Crossover ต่อไป ซึ่งแสดงให้เห็นจากปัญหาตัวอย่างภาพที่ 3.2 และตารางที่ 3.1

3.4.4.2 Mutation ขั้นตอนการการปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากรเดิม เรียกว่า Scaling Factor: F หรือ เรียกว่า Mutation Factor: F เพื่อจุดประสงค์ของการผ่าเหล่า โดยคำนวณหา Mutant Vector ($V_{i,G+1}$) ดังสมการที่ 3.15 และสามารถแสดงตัวอย่างการ Mutation ดังตารางที่ 3.12

$$V_{i,G+1} = X_{best,G} + F(X_{r1,G} - X_{r2,G}) + F(X_{r3,G} - X_{r4,G}) \quad (3.15)$$

เมื่อ

F = Scaling Factor เป็นจำนวนจริงตั้งแต่ 0-2

$X_{best,G}$ = เวคเตอร์ที่ดีที่สุดในรอบที่ G

$X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$, $X_{r4,G}$ = สุ่มเวคเตอร์จากรอบที่ G

ตารางที่ 3.12 ผลการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี “DE/best/2”

Vector	Position								
	Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 2	Work	7	5	1	8	6	3	4	2
	Time	10	8	11	3	12	12	5	11
	Target Vector	0.57	0.32	0.74	0.92	0.21	0.44	0.69	0.82
Vector 3	Work	1	8	7	5	6	4	3	2
	Time	11	3	10	8	12	5	9	17
	Target Vector	0.51	0.96	0.88	0.67	0.84	0.62	0.41	0.92
Vector 4	Work	1	8	7	5	2	3	4	6
	Time	11	3	10	8	17	9	5	12
	Target Vector	0.13	0.26	0.58	0.81	0.64	0.42	0.21	0.69
Vector 5 (best)	Work	8	6	4	7	5	3	1	2
	Time	3	12	5	11	8	9	11	17
	Target Vector	0.84	0.41	0.59	0.91	0.76	0.32	0.98	0.48
Mutant Vector1		**0.32**	**-0.22**	**0.47**	**1.03**	**0.16**	**0.26**	**0.59**	**0.57**

จากตารางที่ 3.12 แสดงผลของการ Mutation เวกเตอร์ที่ 1 โดยใช้วิธี “DE/best/2” ซึ่งสุ่ม $X_{r2,G}$, $X_{r3,G}$, $X_{r4,G}$, $X_{r5,G}$ ได้เวกเตอร์ที่ 2, 3, 4 และ 5 ตามลำดับและกำหนดให้ค่า F = 0.8 เช่นพิกัด (Position) ที่ 5 $X_{r1,G}$ = 0.21, $X_{r2,G}$ = 0.84, $X_{r3,G}$ = 0.64 และ $X_{r4,G}$ = 0.76 ดังนั้น Mutant Vector ที่ 1 พิกัดที่ 5 โดย $X_{best,G}$ = 0.76 ดังนั้นแทนค่าสมการตามสูตร $V_{i,G+1}$= $X_{best,G}$ +F($X_{r1,G}$ – $X_{r2,G}$) + F($X_{r3,G}$ – $X_{r4,G}$) ได้เท่ากับ $V_{i,G+1}$ = 0.76+0.8(0.21-0.84)+0.8(0.64-0.76) จึงได้เท่ากับ 0.16

3.4.4.3 Crossover หรือ Recombination ขั้นตอนวิธีการแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ ซึ่งจะได้เวกเตอร์ใหม่ของคำตอบที่มีทั้งดีกว่าและแย่กว่าอย่างหลากหลาย โดยมีการสร้าง Trial Vector ($U_{ji,G+1}$) ดังสมการที่ 3.16 และทำการเปรียบเทียบพิจารณาในการแลกเปลี่ยนพิกัดดังสมการที่ 3.17 ซึ่งสามารถแสดงตัวอย่างดังตารางที่ 3.13 และ 3.14 ดังนี้

$$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, \ldots , U_{Di,G+1}) \tag{3.16}$$

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} & \text{if (rand (j)} \le \text{r} \\ X_{ji,G+1} & \text{otherwise} \end{cases} \tag{3.17}$$

เมื่อ

$V_{ji,G+1}$ = Mutant Vector

$X_{ji,G+1}$ = Target Vector

CR = Crossover Constant จำนวนจริงมีค่า 0 ถึง 1

rand(j) = การสุ่มตำแหน่งพิกัดในเวกเตอร์, j = 1, 2, 3, ..., G (G=จำนวนพิกัด) และ r1 = พิกัดที่ถูกสุ่มตำแหน่งที่ 1

ตารางที่ 3.13 ผลการ Exponential Crossover 1 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี "DE/best/2"

Vector	Position								
	Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
	Mutant Vector	0.32	-0.22	0.47	1.03	0.16	0.26	0.59	0.57
	rand(j)	-	-	3	-	-	-	-	-
	Trial Vector	**0.32**	**-0.22**	**0.47**	**0.61**	**0.72**	**0.53**	**0.68**	**0.92**

จากตารางที่ 3.13 แสดงผลของ Exponential Crossover 1 Position เวกเตอร์ที่ 1 ซึ่งสุ่มพิกัดได้ r = 3 ดังนั้น Trial Vector ของพิกัดที่ 1 - 3 จะใช้ค่าของ Mutant Vector ส่วนตำแหน่งอื่น ๆ ที่เหลือจะใช้ค่าของ Target Vector

ตารางที่ 3.14 ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.13

Vector	Station							
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	1	6	4	2	3	5	7
	Time	3	11	12	5	17	9	8	10
	Trial Vector	-0.22	0.32	0.47	0.61	0.72	0.53	0.68	0.92

จากตารางที่ 3.14 เมื่อนำค่า Trial Vector ที่ได้มาจัดสมดุลสายการประกอบแบบตัวยูตามเงื่อนไขลำดับงานก่อนหลังและเวลารวมแต่ละสถานีงานต้องไม่เกินรอบเวลาผลิตโดยพิจารณา Trial Vector ที่มีค่าต่ำก่อนทำให้ได้คำตอบ 5 สถานี

3.4.4.4 Selection ขั้นตอนการคัดเลือกประชากรในรุ่นต่อไป (G+1) ซึ่งจะคัดเลือกเอาแต่เฉพาะ คำตอบที่ดีกว่า โดยการเปรียบเทียบผลของ Target Vector กับ Trial Vector ในกรณีที่จำนวนสถานีงานที่ได้ของ Trial Vector ต่ำกว่าหรือเท่ากับ Target Vector ดังนั้นจะเลือก Trial

Vector เป็นประชากรในรุ่นต่อไป ดังสมการที่ 3.7 สรุปได้ว่าจากการขั้นตอนของคัดเลือกประชากรในรุ่นต่อไป Trial Vector ของสถานีที่จัดได้ 5 สถานีงานซึ่งยังมีค่ามากกว่า Target Vector ที่จัดได้ 4 สถานีงานจึงทำให้เลือก Target Vector เป็นคำตอบที่ดีที่สุด และทำการวนรอบซ้ำใหม่ในรุ่นถัดไป

3.4.5 การคำนวณ แบบ "DE/best/2" ร่วมกับ Exponential 2 Position

3.4.5.1 การสร้างคำตอบเริ่มต้น (Initial population) ขั้นตอนการสุ่มจำนวนจริง 0-1 ให้กับแต่ละงาน เป็นการสร้างค่า Target Vector หรือคำตอบเริ่มต้นเพื่อเป็นค่าตัดสินใจในการจัดงานลงสถานีงานและเป็นค่าเริ่มต้นของเวกเตอร์ที่ใช้ในการ Mutation และ Crossover หรือ Recombination ต่อไป ซึ่งแสดงให้เห็นจากปัญหาตัวอย่างภาพที่ 3.2 และตารางที่ 3.1

3.4.5.2 Mutation ขั้นตอนการการปรับเปลี่ยนพิกัดให้ได้คำตอบใหม่ที่แปลกแตกต่างไปจากกลุ่มจำนวนประชากรเดิม เพื่อจุดประสงค์ของการผ่าเหล่า โดยคำนวณหา Mutant Vector ($V_{i,G+1}$) ดังสมการที่ 3.15 และสามารถแสดงตัวอย่างการ Mutation ดังตารางที่ 3.12

3.4.5.3 Crossover หรือ Recombination ขั้นตอนวิธีการแลกเปลี่ยนพิกัดระหว่างเวกเตอร์ ซึ่งจะได้เวกเตอร์ใหม่ของคำตอบที่มีทั้งดีกว่าและแย่กว่าอย่างหลากหลาย โดยมีการสร้าง Trial Vector ($U_{i,j,G+1}$) ดังสมการที่ 3.18 และทำการเปรียบเทียบพิจารณาในการแลกเปลี่ยนพิกัด ดังสมการที่ 3.19 ซึ่งสามารถแสดงตัวอย่างดังตารางที่ 3.15 และ 3.16 ดังนี้

$$U_{ji,G+1} = (U_{1i,G+1}, U_{2i,G+1}, \ldots , U_{Di,G+1}) \tag{3.18}$$

$$U_{i,j,G+1} = \begin{cases} V_{i,j,G+1} \text{ if } (\text{rand } (j) \le r1 \text{ , } (j) > r2 \\ X_{i,j,G+1} \qquad \text{otherwise} \end{cases} \tag{3.19}$$

เมื่อ

$V_{i,j,G+1}$ = Mutant Vector

$X_{i,j,G+1}$ = Target Vector

CR = Crossover Constant จำนวนจริงมีค่า 0 ถึง 1

Rand (j) = การสุ่มตำแหน่งพิกัดในเวกเตอร์ , j = 1, 2, 3, ..., G

(G=จำนวนพิกัด) และ r1, r2 = พิกัดที่ถูกสุ่มตำแหน่งที่ 1 และ 2

ตารางที่ 3.15 ผลการ Exponential Crossover 2 Position เวกเตอร์ที่ 1 ที่ใช้ร่วมกับวิธี "DE/best/2"

Vector	Position								
	Position	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Vector 1	Work	1	8	6	4	2	3	5	7
	Time	11	3	12	5	17	9	8	10
	Target Vector	0.3	0.57	0.44	0.61	0.72	0.53	0.68	0.92
	Mutant Vector	0.32	-0.22	0.47	1.03	0.16	0.26	0.59	0.57
	rand(j)	-	-	3	-	5	-	-	-
	Trial Vector	**0.32**	**-0.22**	**0.47**	**0.61**	**0.72**	**0.26**	**0.59**	**0.57**

จากตารางที่ 3.15 แสดงผลของ Exponential Crossover 2 Position เวกเตอร์ที่ 1 ซึ่งสุ่มพิกัดได้ r1 = 3 และ r2 = 5 ดังนั้น Trial Vector ของพิกัดที่ 1 - 3 และ 6 - 8 จะใช้ค่าของ Mutant Vector ส่วนตำแหน่งอื่น ๆ ที่เหลือจะใช้ค่าของ Target Vector

ตารางที่ 3.16 ผลการจัดสมดุลสายการประกอบโดยใช้ค่า Trial Vector จากตารางที่ 3.15

Vector	Station								
Vector 1	**Station**	**1**		**2**		**3**		**4**	**5**
	Work	8	1	6	4	7	5	3	2
	Time	3	10	12	5	10	8	9	17
	Trial Vector	-0.22	0.32	0.47	0.61	0.57	0.59	0.26	0.72

จากตารางที่ 3.16 เมื่อนำค่า Trial Vector ที่ได้มาจัดสมดุลสายการประกอบแบบตัวยูตามเงื่อนไขลำดับงานก่อนหลังและเวลารวมแต่ละสถานีงานต้องไม่เกินรอบเวลาผลิตโดยพิจารณา Trial Vector ที่มีค่าต่ำก่อนทำให้ได้คำตอบ 5 สถานี

3.4.5.4 Selection ขั้นตอนการคัดเลือกประชากรในรุ่นต่อไป (G+1) ซึ่งจะคัดเลือกเอาแต่เฉพาะ คำตอบที่ดีกว่า โดยการเปรียบเทียบผลของ Target Vector กับ Trial Vector ในกรณีที่จำนวนสถานีงานที่ได้ของ Trial Vector ต่ำกว่าหรือเท่ากับ Target Vector ดังนั้นจะเลือก Trial Vector เป็นประชากรในรุ่นต่อไป ดังสมการที่ 3.7 สรุปได้ว่าจากการขั้นตอนของคัดเลือกประชากรในรุ่นต่อไป Trial Vector ของสถานีที่จัดได้ 5 สถานีงานซึ่งยังมีค่ามากกว่า Target Vector ที่จัดได้ 4 สถานีงานจึงทำให้เลือก Target Vector เป็นคำตอบที่ดีที่สุด และทำการวนรอบซ้ำใหม่ในรุ่นถัดไป

ผู้วิจัยได้สรุปผลที่ได้จากการคำนวณปัญหาการจัดสมดุลสายการประกอบแบบตัวยูทั้ง 6 วิธี โดยวิธีพื้นฐาน 1 วิธี คือ แบบ "DE/rand/1" ร่วมกับ Binomial Crossover และวิธีปรับปรุงคำตอบ 5 วิธี คือ 1) "DE/rand/1" ร่วมกับ Exponential Crossover 1 Position 2) "DE/rand/1" ร่วมกับ Exponential Crossover 2 Position 3) "DE/ rand-to-best/1" ร่วมกับ Binomial Crossover

4) “DE/Best/2” ร่วมกับ Exponential Crossover 1 Position และ 5) “DE/Best/2” ร่วมกับ Exponential Crossover 2 Position สรุปได้ดังนี้

(1) วิธีพื้นฐาน 1 วิธี = แบบ “DE/rand/1” ร่วมกับ Binomial Crossover คำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างได้เท่ากับ 5 สถานีงาน ดังตารางที่ 3.17

ตารางที่ 3.17 ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/rand/1” ร่วมกับ Binomial Crossover

Vector	**Station**								
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	6	1	3	2	4	5	7
	Time	3	12	11	9	17	5	8	10
	Trial Vector	**0.06**	**0.33**	**0.35**	**0.39**	**0.22**	**0.61**	**0.68**	**0.84**

(2) วิธีปรับปรุงคำตอบวิธี 1 = แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 1 Position คำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างได้เท่ากับ 5 สถานีงาน ดังตารางที่ 3.18

ตารางที่ 3.18 ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 1 Position

Vector	**Station**								
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	1	6	4	2	3	5	7
	Time	3	11	12	5	17	9	8	10
	Trial Vector	**0.06**	**0.35**	**0.44**	**0.61**	**0.72**	**0.53**	**0.68**	**0.92**

(3) วิธีปรับปรุงคำตอบวิธี 2 = แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 2 Position คำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างได้เท่ากับ 5 สถานีงาน ดังตารางที่ 3.19

ตารางที่ 3.19 ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 2 Position

Vector	Station								
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	1	6	4	2	3	7	5
	Time	3	11	12	5	17	9	10	8
	Trial Vector	**0.06**	**0.35**	**0.44**	**0.61**	**0.72**	**0.39**	**0.84**	**0.90**

(4) วิธีปรับปรุงคำตอบวิธี 3 = แบบ “DE/ rand-to-best/1” ร่วมกับ Binomial Crossover คำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างได้เท่ากับ 5 สถานีงาน ดังตารางที่ 3.20

ตารางที่ 3.20 ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/ rand-to-best/1” ร่วมกับ Binomial Crossover

Vector	Station								
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	6	4	1	2	3	7	5
	Time	3	12	5	11	17	9	10	8
	Trial Vector	**-0.07**	**0.45**	**0.61**	**0.78**	**0.25**	**0.22**	**0.49**	**0.68**

(5) วิธีปรับปรุงคำตอบวิธี 4 = แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 1 Position คำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างได้เท่ากับ 5 สถานีงาน ดังตารางที่ 3.21

ตารางที่ 3.21 ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 1 Position

Vector	Station								
Vector 1	**Station**	**1**		**2**		**3**	**4**		**5**
	Work	8	1	6	4	2	3	5	7
	Time	3	11	12	5	17	9	8	10
	Trial Vector	**-0.22**	**0.32**	**0.47**	**0.61**	**0.72**	**0.53**	**0.68**	**0.92**

(6) วิธีปรับปรุงคำตอบวิธี 5 = แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 2 Position คำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างได้เท่ากับ 5 สถานีงาน ดังตารางที่ 3.22

ตารางที่ 3.22 ผลสรุปการจัดสถานีงานด้วยวิธี แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 2 Position

Vector	Station								
Vector 1	**Station**	**1**		**2**		**3**		**4**	**5**
	Work	8	1	6	4	7	5	3	2
	Time	3	10	12	5	10	8	9	17
	Trial Vector	**-0.22**	**0.32**	**0.47**	**0.61**	**0.57**	**0.59**	**0.26**	**0.72**

จากการสร้างคำตอบเริ่มต้น (Initial population) เริ่มต้นจากการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างจากตารางที่ 1 ข้างต้น จะเห็นได้ว่า Vector ที่ 5 จัดสถานีงานได้เท่ากับ 4 สถานีงาน ดังตารางที่ 3.23

ตารางที่ 3.23 ผลการจัดสถานีงานการสร้างคำตอบเริ่มต้น (Initial population) เริ่มต้นด้วยวิธีวิวัฒนาการโดยใช้ผลต่าง

Vector	Station								
Vector 5	**Station**	**1**			**2**		**3**		**4**
	Work	8	6	4	7	5	3	1	2
	Time	3	12	5	11	8	9	11	17
	Target Vector	**0.84**	**0.41**	**0.59**	**0.91**	**0.76**	**0.32**	**0.98**	**0.48**

และทำการสรุปเปรียบเทียบการคำนวณแต่ละวิธีโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ดังตารางที่ 3.24

ตารางที่ 3.24 สรุปใช้วิธีวิวัฒนาการโดยใช้ผลต่างกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยูแต่ละวิธี

วิธี / ปัญหา	DE^1	DE^2	DE^3	DE^4	DE^5	DE^6	DE-Target Vector
Bowman problem (1960)	5 Station	5 Station	5 Station	5 Station	5 Station	5 Station	4 Station

หมายเหตุ: DE^1 = แบบ “DE/rand/1” ร่วมกับ Binomial Crossover (พื้นฐาน)

DE^2 = แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 1 Position

DE^3 = แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 2 Position

DE^4 = แบบ “DE/ rand-to-best/1” ร่วมกับ Binomial Crossover

DE^5 = แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 1 Position

DE^6 = แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 2 Position

จากตารางที่ 3.24 สรุปวิธีการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยูกับปัญหาตัวอย่างจะเห็นได้ว่าเมื่อทำการจัดสถานีงานตามเงื่อนไขแบบตัวยูและไม่เกินรอบเวลาการผลิต DE-Target Vector ซึ่งเป็นการจัดด้วยวิธีวิวัฒนาการโดยใช้ผลต่างกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยูจะได้คำตอบที่ดีที่สุดที่ 4 สถานีงาน = {8, 6, 4}, {7, 5}, {3, 1}, {2} ได้ค่าคำตอบเท่ากับ Data set ของ Armin Scholl การจัดสมดุลสายการประกอบแบบตัวยูจากชุดปัญหาเดียวกันจากงานวิจัยของ Armin Scholl (2016: Website) แสดงให้เห็นว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยูให้ค่าตำตอบที่ดีที่สุด

บทที่ 4
วิเคราะห์ผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู

ในบทนี้จะเป็นการวิเคราะห์ผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับปัญหาการจัดสมดุลสายการประกอบแบบตัวยูแบบพื้นฐานและแบบปรับปรุงคำตอบและทำการเปรียบเทียบผลการทดลองกับวิธีเมตาฮิวริสติกแบบอื่น ๆ โดยมีวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับปัญหาการจัดสมดุลสายการประกอบแบบตัวยูแบบพื้นฐานและแบบปรับปรุงคำตอบประกอบไปด้วย 6 วิธีดังนี้

(1) แบบ “DE/rand/1” ร่วมกับ Binomial Crossover (พื้นฐาน)
(2) แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 1 Position
(3) แบบ “DE/rand/1” ร่วมกับ Exponential Crossover 2 Position
(4) แบบ “DE/ rand-to-best/1” ร่วมกับ Binomial Crossover
(5) แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 1 Position
(6) แบบ “DE/Best/2” ร่วมกับ Exponential Crossover 2 Position

4.1 การพัฒนาโปรแกรมการจัดสมดุลสายการประกอบแบบตัวยู

จากการแก้ปัญหาการจัดสมดุลสายการประกอบ โดยการประยุกต์โปรแกรมภาษา Java (รันบนเครื่องคอมพิวเตอร์รุ่น Core i3, 2.3 GHz, 2GB RAM บนระบบปฏิบัติการ Windows 7) ที่พัฒนาวิธีการเมตาฮิวริสติก คือ วิธีการจัดสมดุลสายการประกอบแบบตัวยูแบบพื้นฐานและแบบปรับปรุงคำตอบโดยมี Pseudo code และแผนผังขั้นตอนการทำงานการจัดสมดุลสายการประกอบแบบตัวยู ดังภาคผนวก ก

ซึ่งในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 จะเป็นการหาคำตอบของจำนวนสถานีงานที่น้อยที่สุด ของแต่ละชุดโจทย์ปัญหาตัวอย่าง ที่มีการกำหนดค่ารอบเวลาการผลิต และคำตอบที่ดีที่สุด ซึ่งได้ข้อมูลมาจาก http://www.assembly-line-balancing.de และจากงานวิจัยอื่น ๆ ที่ผ่านมา เพื่อใช้ในการอ้างอิงและเปรียบเทียบผล โดยตัวอย่างของชุดปัญหาและลำดับขั้นงานก่อน-หลัง ดังภาคผนวก ข

4.2 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน

การจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณกำหนดให้ R=30 รอบ NP=30 F=0.9 CR=0.8 Ronkkonen, Kukkonen and Price (2005) จึงทำการทดลองเพื่อหาค่าที่เหมาะสมดังกล่าว ดังนี้

ใช้ชุดโจทย์ปัญหาของ Sawyer (ที่มี 30 ขั้นงาน 8 Instances) และของ Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) รวมทั้งหมด 24 Instances

ตารางที่ 4.1 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน DE/rand/1 ร่วมกับ Binomial Crossover กับปัญหา Sawyer และ Arcus 1

Problem	c	m*	DE/rand/1 to Binomial Crossover	Cal Time (sec.)
Sawyer 30	25	14	14	1.87
	27	13	13	1.11
	30	11	11	1.23
	33	10	10	1.11
	36	10	10	1.34
	41	8	8	1.96
	54	6	6	1.61
	75	5	5	1.94
Arcus 1 83	3786	21	21	0.91
	3985	20	20	0.89
	4206	19	19	0.64
	4454	18	18	0.85
	4732	17	17	0.46
	5048	16	16	0.31
	5408	15	15	0.89
	5824	14	14	0.64
	5853	13	13	0.06
	6309	13	13	1.98
	6842	12	12	0.64
	6883	12	12	0.79
	7571	11	11	0.74
	8412	10	10	1.97
	8898	9	9	1.89
	10816	7	7	1.98
Total optimal solutions found from 24 instances problem			**24**	**1.16**

หมายเหตุ: m is the no. of station (m* is the optimal), c is the cycle time

จากตารางที่ 4.1 เป็นผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน DE/rand/1 ร่วมกับ Binomial Crossover กับปัญหา Sawyer และ Arcus 1 จากตารางผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน DE/rand/1 ร่วมกับ Binomial Crossover จากปัญหา Sawyer ได้ค่าคำตอบที่ดีที่สุด 24 คำตอบ จาก 24 Instances และจากตารางพบว่าค่าเฉลี่ยจากค้นหาคำตอบที่ดีที่สุดใช้เวลาเฉลี่ย 1.16 วินาที ซึ่งการค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุด คือ ปัญหา Arcus 1 ที่ cycle time เท่ากับ 5853 ใช้เวลา 0.06 วินาที และพบว่าการค้นหาคำตอบที่ดีที่สุดในการหาจำนวนสถานีงานเทียบเท่ากับ data set

4.3 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 1 Position

การจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณกำหนดให้ R=30 รอบ NP=30 F=0.9 CR=0.8 Ronkkonen, Kukkonen and Price (2005) จึงทำการทดลองเพื่อหาค่าที่เหมาะสมดังกล่าว ดังนี้

ใช้ชุดโจทย์ปัญหาของ Sawyer (ที่มี 30 ขั้นงาน 8 Instances) และของ Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) รวมทั้งหมด 24 Instances

ตารางที่ 4.2 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1

Problem	c	m*	DE/rand/1 to Exponential Crossover 1 Position	Cal Time (sec.)
Sawyer 30	25	14	14	0.67
	27	13	13	0.44
	30	11	11	0.83
	33	10	10	0.67
	36	10	10	0.55
	41	8	8	0.59
	54	6	6	0.78
	75	5	5	0.53
Arcus 1 83	3786	21	21	0.31
	3985	20	20	0.89
	4206	19	19	0.64
	4454	18	18	0.55

ตารางที่ 4.2 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1 (ต่อ)

Problem	c	m*	DE/rand/1 to Exponential Crossover 1 Position	Cal Time (sec.)
	4732	17	17	0.98
	5048	16	16	0.77
	5408	15	15	0.44
	5824	14	14	0.34
	5853	13	13	0.50
	6309	13	13	0.61
	6842	12	12	0.57
	6883	12	12	0.99
	7571	11	11	0.44
	8412	10	10	0.34
	8898	9	9	0.50
	10816	7	7	0.61
Total optimal solutions found from 24 instances problem			**24**	**0.61**

หมายเหตุ: m is the no. of station (m* is the optimal) c is the cycle time

จากตารางที่ 4.2 เป็นผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1 จากตารางผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 1 Position จากปัญหา Sawyer ได้ค่าคำตอบที่ดีที่สุด 24 คำตอบ จาก 24 Instances และจากตารางพบว่าค่าเฉลี่ยจากค้นหาคำตอบที่ดีที่สุดใช้เวลาเฉลี่ย 0.61 วินาที ซึ่งการค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุด คือ ปัญหา Arcus 1 ที่ cycle time เท่ากับ 3786 ใช้เวลา 0.31 วินาที และพบว่าการค้นหาคำตอบที่ดีที่สุดในการหาจำนวนสถานีงานเทียบเท่ากับ data set

4.4 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/ rand-to-best/1 ร่วมกับ Binomial Crossover

การจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณกำหนดให้ R=30 รอบ

NP=30 F=0.9 CR=0.8 Ronkkonen, Kukkonen and Price (2005) จึงทำการทดลองเพื่อหาค่าที่เหมาะสมดังกล่าว ดังนี้

ใช้ชุดโจทย์ปัญหาของ Sawyer (ที่มี 30 ขั้นงาน 8 Instances) และของ Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) รวมทั้งหมด 24 Instances

ตารางที่ 4.3 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 2 Position กับปัญหา Sawyer และ Arcus 1

Problem	c	m*	DE/rand/1 to Exponential Crossover 2 Position	Cal Time (sec.)
Sawyer 30	25	14	14	0.11
	27	13	13	0.13
	30	11	11	0.42
	33	10	10	0.67
	36	10	10	0.55
	41	8	8	0.59
	54	6	6	0.38
	75	5	5	0.07
Arcus 1 83	3786	21	21	0.54
	3985	20	20	0.17
	4206	19	19	0.14
	4454	18	18	0.31
	4732	17	17	0.89
	5048	16	16	0.64
	5408	15	15	0.06
	5824	14	14	1.98
	5853	13	13	0.24
	6309	13	13	0.55
	6842	12	12	0.54
	6883	12	12	0.47
	7571	11	11	0.98
	8412	10	10	1.98

ตารางที่ 4.3 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 2 Position กับปัญหา Sawyer และ Arcus 1 (ต่อ)

Problem	c	m*	DE/rand/1 to Exponential Crossover 2 Position	Cal Time (sec.)
	8898	9	9	0.24
	10816	7	7	0.55
Total optimal solutions found from 24 instances problem			**24**	**0.53**

หมายเหตุ: m is the no. of station (m* is the optimal), c is the cycle time

จากตารางที่ 4.3 เป็นผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 2 Position กับปัญหา Sawyer และ Arcus 1 จากตารางผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 2 Position จากปัญหา Sawyer ได้ค่าคำตอบที่ดีที่สุด 24 คำตอบ จาก 24 Instances และจากตารางพบว่าค่าเฉลี่ยจากค้นหาคำตอบที่ดีที่สุดใช้เวลาเฉลี่ย 0.53 วินาที ซึ่งการค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุด คือ ปัญหา Arcus 1 ที่ cycle time เท่ากับ 5408 ใช้เวลา 0.06 วินาที และจากการค้นหาคำตอบที่ดีที่สุดโดยวิธีวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand/1 ร่วมกับ Exponential Crossover 2 Position ค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานเทียบเท่ากับ date set

4.5 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/ rand-to-best/1 ร่วมกับ Binomial Crossover

การจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณกำหนดให้ R=30 รอบ NP=30 F=0.9 CR=0.8 Ronkkonen, Kukkonen and Price (2005) จึงทำการทดลองเพื่อหาค่าที่เหมาะสมดังกล่าว ดังนี้

ใช้ชุดโจทย์ปัญหาของ Sawyer (ที่มี 30 ขั้นงาน 8 Instances) และของ Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) รวมทั้งหมด 24 Instances

ตารางที่ 4.4 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/rand-to-best/1 ร่วมกับ Binomial Crossover กับปัญหา Sawyer และ Arcus 1

Problem	c	m*	DE/ rand-to-best/1 to Binomial Crossover	Cal Time (sec.)
Sawyer 30	25	14	14	1.87
	27	13	13	0.98
	30	11	11	0.77
	33	10	10	0.44
	36	10	10	0.34
	41	8	8	0.50
	54	6	6	0.61
	75	5	5	0.57
Arcus 1 83	3786	21	21	0.31
	3985	20	20	0.89
	4206	19	19	0.64
	4454	18	18	0.55
	4732	17	17	0.46
	5048	16	16	0.31
	5408	15	15	0.89
	5824	14	14	0.64
	5853	13	13	0.06
	6309	13	13	1.98
	6842	12	12	0.24
	6883	12	12	0.59
	7571	11	11	0.54
	8412	10	10	0.47
	8898	9	9	0.51
	10816	7	7	0.98
Total optimal solutions found from 24 instances problem			**24**	**0.67**

หมายเหตุ: m is the no. of station (m* is the optimal) c is the cycle time

จากตารางที่ 4.4 เป็นผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/ rand-to-best/1 ร่วมกับ Binomial

Crossover กับปัญหา Sawyer และ Arcus 1 กับปัญหา Sawyer ได้ค่าคำตอบที่ดีที่สุด 24 คำตอบ จาก 24 Instances และจากตารางพบว่าค่าเฉลี่ยจากค้นหาคำตอบที่ดีที่สุดใช้เวลาเฉลี่ย 0.67 วินาที ซึ่งการค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุด คือ ปัญหา Arcus 1 ที่ cycle time เท่ากับ 5853 ใช้เวลา 0.06 วินาที และการค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานเทียบเท่ากับ data set

4.6 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 Position

การจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณกำหนดให้ R=30 รอบ NP=30 F=0.9 CR=0.8 Ronkkonen, Kukkonen and Price (2005) จึงทำการทดลองเพื่อหาค่าที่เหมาะสมดังกล่าว ดังนี้

ใช้ชุดโจทย์ปัญหาของ Sawyer (ที่มี 30 ขั้นงาน 8 Instances) และของ Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) รวมทั้งหมด 24 Instances

ตารางที่ 4.5 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1

Problem	c	m*	DE/best/2 to Exponential Crossover 1 Position	Cal Time (sec.)
Sawyer 30	25	14	14	0.07
	27	13	13	0.12
	30	11	11	0.08
	33	10	10	0.18
	36	10	10	0.05
	41	8	8	0.04
	54	6	6	0.09
	75	5	5	0.06
Arcus 1 83	3786	21	21	0.30
	3985	20	20	0.06
	4206	19	19	0.07
	4454	18	18	0.10
	4732	17	17	0.28
	5048	16	16	0.08
	5408	15	15	0.12

ตารางที่ 4.5 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1 (ต่อ)

Problem	c	m*	DE/best/2 to Exponential Crossover 1 Position	Cal Time (sec.)
	5824	14	14	0.04
	5853	13	13	0.06
	6309	13	13	0.09
	6842	12	12	0.19
	6883	12	12	0.38
	7571	11	11	0.35
	8412	10	10	0.22
	8898	9	9	0.45
	10816	7	7	0.71
Total optimal solutions found from 24 instances problem			**24**	**0.17**

หมายเหตุ: m is the no. of station (m* is the optimal) c is the cycle time

จากตารางที่ 4.5 เป็นผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 Position กับปัญหา Sawyer และ Arcus 1 จากตารางผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 Position จากปัญหา Sawyer ได้ค่าคำตอบที่ดีที่สุด 24 คำตอบ จาก 24 Instances และจากตารางพบว่าค่าเฉลี่ยจากค้นหาคำตอบที่ดีที่สุดใช้เวลาเฉลี่ย 0.17 วินาที ซึ่งการค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุด คือ ปัญหา Arcus 1 ที่ cycle time เท่ากับ 5824 ใช้เวลา 0.04 วินาที และปัญหา Sawyer cycle time เท่ากับ 41 ใช้เวลา 0.04 วินาที และการค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานเทียบเท่ากับ data set

4.7 ขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 2 Position

การจัดสมดุลสายการประกอบต้องกำหนดค่าที่ใช้สำหรับการคำนวณ โดยตัวแปรดังนี้ R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) โดยการคำนวณกำหนดให้ R=30 รอบ NP=30 F=0.9 CR=0.8 Ronkkonen, Kukkonen and Price (2005) จึงทำการทดลองเพื่อหาค่าที่เหมาะสมดังกล่าว ดังนี้

ใช้ชุดโจทย์ปัญหาของ Sawyer (ที่มี 30 ขั้นงาน 8 Instances) และของ Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) รวมทั้งหมด 24 Instances

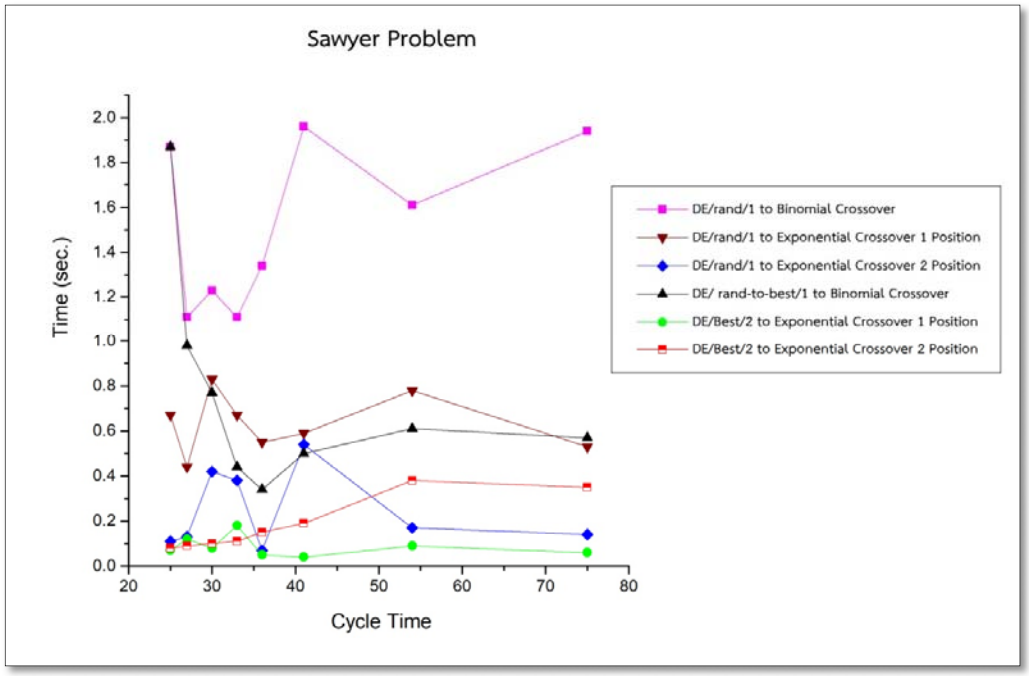
ตารางที่ 4.6 ผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 2 Position กับปัญหา Sawyer และ Arcus 1

Problem	c	m*	DE/best/2 to Exponential Crossover 2 Position	Cal Time (sec.)
Sawyer 30	25	14	14	0.08
	27	13	13	0.09
	30	11	11	0.10
	33	10	10	0.11
	36	10	10	0.15
	41	8	8	0.19
	54	6	6	0.38
	75	5	5	0.35
Arcus 1 83	3786	21	21	0.14
	3985	20	20	0.11
	4206	19	19	0.23
	4454	18	18	0.98
	4732	17	17	0.16
	5048	16	16	0.25
	5408	15	15	0.06
	5824	14	14	1.98
	5853	13	13	0.24
	6309	13	13	0.55
	6842	12	12	0.54
	6883	12	12	0.16
	7571	11	11	0.25
	8412	10	10	0.06
	8898	9	9	1.66
	10816	7	7	0.78
Total optimal solutions found from 24 instances problem			**24**	**0.40**

หมายเหตุ: m is the no. of station (m* is the optimal), c is the cycle time

จากตารางที่ 4.6 เป็นผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยูประเภทที่ 1 โดยใช้วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 2 Position กับปัญหา Sawyer และ Arcus 1 จากตารางผลการทดลองวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 2 Position จากปัญหา Sawyer ได้ค่าคำตอบที่ดีที่สุด 24 คำตอบ จาก 24 Instances และจากตารางพบว่าค่าเฉลี่ยจากค้นหาคำตอบที่ดีที่สุดใช้เวลาเฉลี่ย 0.40 วินาที ซึ่งการค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุด คือ ปัญหา Arcus 1 ที่ cycle time เท่ากับ 5408 และ 8412 ใช้เวลาเท่ากันคือ 0.06 วินาที และการค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานเทียบเท่ากับ data set

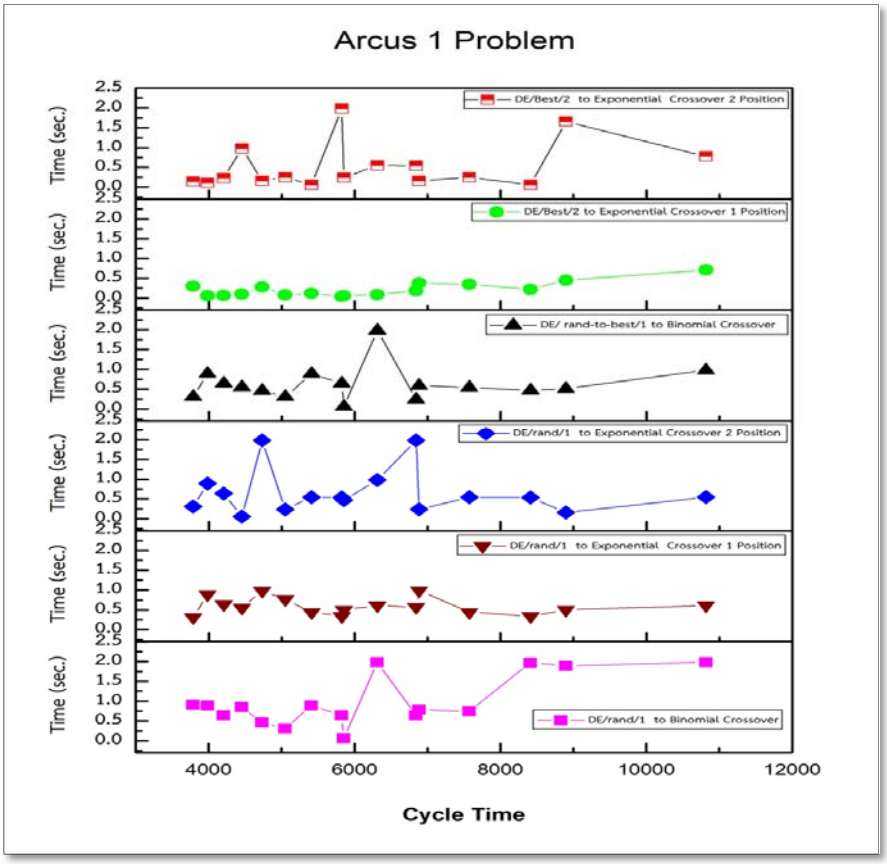
ดังนั้นทำการเปรียบเทียบวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบที่ใช้เวลาในการค้นหาคำตอบที่ดีที่สุดกับปัญหา Sawyer (ที่มี 30 ขั้นงาน 8 Instances) ดังภาพที่ 4.1



ภาพที่ 4.1 การเปรียบเทียบวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบที่ใช้เวลาในการค้นหาคำตอบที่ดีที่สุดกับปัญหา Sawyer (ที่มี 30 ขั้นงาน 8 Instances)

จากภาพที่ 4.1 พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับปัญหา Sawyer (ที่มี 30 ขั้นงาน 8 Instances) เป็นวิธีค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุดเมื่อเปรียบเทียบกับวิธีวิวัฒนาการโดยใช้ผลต่างแบบอื่น ๆ โดยใช้เวลาเฉลี่ยในการค้นหาคำตอบที่ดีที่สุดเท่ากับ 0.09 และวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน DE/rand/1 ร่วมกับ Binomial Crossover กับปัญหา Sawyer (ที่มี 30 ขั้นงาน 8 Instances)

ใช้เวลามากที่สุดเท่ากับ 1.52 วินาที และการค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานเทียบเท่ากับ data set และทำการเปรียบเทียบวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบที่ใช้เวลาในการค้นหาคำตอบที่ดีที่สุดกับปัญหา Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) ดังภาพที่ 4.2



ภาพที่ 4.2 การเปรียบเทียบวิธีของวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบที่ใช้เวลาในการค้นหาคำตอบที่ดีที่สุดกับปัญหา Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances)

จากภาพที่ 4.2 พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับปัญหา Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances) เป็นวิธีค้นหาคำตอบที่ดีที่สุดที่ใช้เวลาน้อยที่สุดเมื่อเปรียบเทียบกับวิธีวิวัฒนาการโดยใช้ผลต่างแบบอื่น ๆ โดยใช้เวลาเฉลี่ยในการค้นหาคำตอบที่ดีที่สุดเท่ากับ 0.22 และวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน DE/rand/1 ร่วมกับ Binomial Crossover กับปัญหา Arcus 1 (ที่มี 83 ขั้นงาน 16 Instances)

ใช้เวลามากที่สุดเท่ากับ 0.98 วินาที และการค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานเทียบเท่ากับ data set

4.8 ผลการเปรียบเทียบวิธีวิวัฒนาการโดยใช้ผลต่างเปรียบเทียบกับวิธีเมตาฮิวริสติกวิธีอื่น ๆ

จากผลการทดลองจะเห็นได้ว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 เป็นวิธีที่ค้นหาค่าคำตอบที่ดีที่สุดในการหาจำนวนสถานีงานและใช้เวลาน้อยที่สุดในการค้นหาคำตอบที่ดีที่สุด

ดังนั้นจึงเอาวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 มาทำการเปรียบเทียบกับวิธีเมตาฮิวริสติกวิธีอื่น ๆ เพื่อวัดประสิทธิภาพในการค้นหาคำตอบที่ดีที่สุด โดยผู้วิจัยได้ทำการเปรียบกับวิธีระบบมด ซึ่งได้แสดงในตารางที่ 4.6

ตารางที่ 4.7 การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบมด

Problems	Size	Cycle time	IP solution*	ACO**	DE/best/2 to Exponential 1 position***			
			m*	m	m	%	cal. Time (sec.)	E
Mitchell	21	14	8	8	8	0	0.05	93.75
		15	8	8	8	0	0.03	87.75
		21	5	5	5	0	0.01	100.00
Heskiaoff	28	138	8	8	8	0	0.09	92.75
		205	5	5	5	0	0.07	99.99
		216	5	5	5	0	0.13	94.81
		256	4	4	4	0	0.16	100.00
		324	4	4	4	0	0.09	79.01
		342	3	3	3	0	0.12	99.81
Sawyer	30	25	14	14	14	0	0.07	92.57
		27	13	13	13	0	0.12	92.31
		30	11	12	11	9.09	0.08	98.18
		36	10	10	10	0	0.05	90.00
		41	8	8	8	0	0.04	98.78
		54	6	7	6	16.67	0.09	100.00
		75	5	5	5	0	0.06	86.84

ตารางที่ 4.7 การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบมด (ต่อ)

Problems	Size	Cycle time	IP solution*	ACO**	DE/best/2 to Exponential 1 position ***			
			m*	m	m	%	cal. Time (sec.)	E
Kilbridge & Wester	45	57	10	10	10	0	0.11	96.84
		79	7	7	7	0	0.12	99.82
		92	6	6	6	0	0.14	100.00
		138	4	4	4	0	0.14	100.00
		184	3	3	3	0	0.15	100.00
Total optimal solution (or lower bound) found from 21 instances problem				**19**	**21/ inst.**	**1.23/ inst.**	**0.09/ inst.**	**95.39/ inst.**

หมายเหตุ: * IP solution = Integer Programming solution, result from Urban (1998: 740)
** ACO by Zhang et al. (2008)
*** DE1 = DE/best/2 to Exponential 1 position
m* is the optimal solution
กำหนดให้ค่า F = 0.9 Ronkkonen, Kukkonen and Price (2005) CR = 0.8 N = 30 Round = 30, m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance

จากตารางที่ 4.7 แสดงผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ในโจทย์ปัญหาขนาดกลาง (21-45 ขั้นงาน) ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับวิธีระบบมด (ไม่ได้ระบุรุ่นของเครื่องคอมพิวเตอร์ที่ใช้ในการทดลอง) และเปรียบเทียบกับค่าคำตอบที่เป็น lower bound หรือ คำตอบที่ดีที่สุดซึ่งเป็นผลจากงานวิจัยของ Urban (1998) (ที่ทดลองบนเครื่อง Pentium 90 MHz โดยใช้โปรแกรม CPLEX 3.0) โดยใช้โจทย์ปัญหาตัวอย่าง 4 ชุด (ชุดปัญหาของ Mitchell, Heskiaoff, Sawyer และ Kilbridge & Wester) ที่มี 21-45 ขั้นงาน และมี 21 Instances จากผลลัพธ์ในการหาค่าจำนวนสถานีงานที่น้อยที่สุด ดังแสดงในตารางพบว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถหาคำตอบที่ดีที่สุดได้ 21 คำตอบจาก 21 Instances ซึ่งสามารถหาค่าคำตอบที่ดีที่สุดได้ดีกว่าระบบมดที่หาค่าคำตอบที่ดีที่สุด 19 คำตอบจาก 21 Instances ซึ่งจากการเปรียบเทียบดังตารางที่ 4.7 เป็นชุดปัญหาขนาดกลาง (21-45 ขั้นงาน)

และผู้วิจัยได้ทำการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ในโจทย์ชุดปัญหาขนาดใหญ่ (75-297 ขั้นงาน) และทำการเปรียบเทียบชุดปัญหาขนาดใหญ่ (75-297 ขั้นงาน) ด้วยวิธีวิวัฒนาการ

โดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับวิธี MMAS (no local search) Nuchsara K. and et al (2007a, 2007b) ซึ่งได้แสดงในตารางที่ 4.8, 4.9 และ 4.10 ตามลำดับซึ่งเป็นตารางชุดเดียวกันที่เป็นปัญหาขนาดใหญ่ (75-297 ขั้นงาน)

ตารางที่ 4.8 การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search)

Problems	Size	Cycle time	Optimal solution	MMAS**				DE1***			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
Wee-mag	75	28	63	63	0	1.19	93.75	63	0	0.12	93.75
21 inst.		29	63	63	0	1.19	87.75	63	0	0.14	87.75
		30	62	62	0	1.23	100.00	62	0	0.07	100.00
		31	62	62	0	1.14	89.82	62	0	0.05	89.82
		32	61	61	0	1.16	88.89	61	0	0.10	88.89
		33	61	61	0	1.22	92.75	61	0	0.28	92.75
		34	61	61	0	1.22	99.99	61	0	0.08	99.99
		35	60	60	0	1.22	94.81	60	0	0.12	94.81
		36	60	60	0	1.25	100.00	60	0	0.04	100.00
		37	60	60	0	1.23	79.01	60	0	0.06	79.01
		38	60	60	0	1.19	99.81	60	0	0.22	99.81
		39	60	60	0	1.14	92.57	60	0	0.08	92.57
		40	60	60	0	1.25	92.31	60	0	0.19	92.31
		41	59	59	0	1.19	98.18	59	0	0.10	98.18
		42	55	55	0	1.16	98.18	55	0	0.11	98.18
		43	50	50	0	1.30	90.00	50	0	0.15	90.00
		45	38	38	0	4.31	98.78	38	0	0.19	98.78
		46	34	34	0	3.59	100.00	34	0	0.38	100.00
		52	31	31	0	2.59	86.84	31	0	0.35	86.84

ตารางที่ 4.8 การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search) (ต่อ)

Problems	Size	Cycle time	Optimal solution	MMAS**				DE1***			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
		54	31	31	0	1.17	96.84	31	0	0.22	96.84
		56	30	30	0	1.55	99.82	30	0	0.45	99.82
Total optimal solution found from 21 instances problem				**21/ inst**	**0**	**1.58/ inst**	**94.30/ inst**	**21/ inst**	**0**	**0.17/ inst**	**94.30/ inst**

หมายเหตุ: m* is the optimal solution

** MMAS (no local search) by Nuchsara and et al (2007a; 2007b)

*** DE1 = DE/best/2 to Exponential 1 position

กำหนดให้ค่า F = 0.9 CR = 0.8 N = 30, Round = 30 Ronkkonen, Kukkonen and Price (2005), m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance

จากตารางที่ 4.8 แสดงผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ในโจทย์ปัญหาขนาดใหญ่ (75-297 ขั้นงาน) ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับวิธีระบบ MMAS (no local search) (รันบนของเครื่องคอมพิวเตอร์รุ่น Pentium 4, 3.0 GHz 512 MB RAM บนระบบปฏิบัติการ Window XP) และเปรียบเทียบกับค่าคำตอบ Data set ซึ่งได้ข้อมูลมาจาก http://www.assembly-line-balancing.de โดยใช้โจทย์ปัญหาตัวอย่าง 3 ชุด โดยชุดปัญหาจากตารางที่ 4.8 ของ Wee-mag ที่มี 75 ขั้นงาน และมี 21 Instances ดังแสดงในตาราง พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถค้นหาคำตอบใช้เวลาที่น้อยกว่า วิธีระบบ MMAS (no local search)

ตารางที่ 4.9 การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search)

Problems	Size	Cycle time	Optimal solution[1]	MMAS**				DE1***			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
Arcus 1 16 inst.	83	3786	21	21	0	1.03	100.00	21	0	0.30	100.00
		3985	20	20	0	1.00	83.64	20	0	0.06	83.64
		4206	19	19	0	1.08	89.82	19	0	0.07	89.82
		4454	18	18	0	1.03	88.89	18	0	0.10	88.89
		4732	17	17	0	1.02	92.75	17	0	0.28	92.75
		5048	16	16	0	1.05	99.99	16	0	0.08	99.99
		5408	15	15	0	1.08	94.81	15	0	0.12	94.81
		5824	14	14	0	1.10	100.00	14	0	0.04	100.00
		5853	13	14	7.69	1.05	79.01	13	0	0.06	89.63
		6309	13	13	0	1.07	99.81	13	0	0.09	99.81
		6842	12	12	0	1.03	92.57	12	0	0.19	92.57
		6883	12	12	0	1.03	92.31	12	0	0.38	92.31
		7571	11	11	0	1.04	98.18	11	0	0.35	98.18
		8412	10	10	0	1.05	98.18	10	0	0.22	98.18
		8898	9	9	0	1.03	90.00	9	0	0.45	90.00
		10816	7	8	14.29	1.04	98.78	8	14.29	0.71	98.78
Total optimal solution found from 16 instances problem				**14**	**1.37/ inst**	**1.05/ inst**	**93.61/ inst**	**15**	**0.89/ inst**	**0.22/ inst**	**94.34/ inst**

หมายเหตุ: m* is the optimal solution

** MMAS (no local search) by Nuchsara and et al (2007a; 2007b)

*** DE1 = DE/best/2 to Exponential 1 position

กำหนดให้ค่า F = 0.9 CR = 0.8 N = 30, Round = 30 Ronkkonen, Kukkonen and Price (2005), m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance

จากตารางที่ 4.9 แสดงผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ในโจทย์ปัญหาขนาดใหญ่ (75-297 ขั้นงาน) ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential

Crossover 1 กับวิธีระบบ MMAS (no local search) และเปรียบเทียบกับค่าคำตอบ Data set ซึ่งได้ข้อมูลมาจาก http://www.assembly-line-balancing.de โดยใช้โจทย์ปัญหาตัวอย่าง 3 ชุด โดยชุดปัญหาจากตารางที่ 4.9 ชุดปัญหาของ Arcus 1 ที่มี 83 ขั้นงาน และมี 16 Instances จากผลลัพธ์ในการหาค่าจำนวนสถานีงานที่น้อยที่สุด ดังแสดงในตารางพบว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถหาคำตอบที่ดีที่สุดได้ 15 คำตอบจาก 16 Instances ซึ่งสามารถหาค่าคำตอบที่ดีที่สุดดีกว่า วิธีระบบ MMAS (no local search) ที่หาค่าคำตอบที่ดีที่สุด 14 คำตอบจาก 16 Instances และการค้นหาค้นหาคำตอบที่ดีที่สุดวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถค้นหาคำตอบใช้เวลาที่น้อยกว่า วิธีระบบ MMAS (no local search)

ตารางที่ 4.10 การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search)

Problems	Size	Cycle time	Optimal solution	MMAS**				DE1***			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
Scholl 25 inst.	297	1394	50	51	2.00	12.08	100.00	51	2.00	12.28	100.00
		1452	48	49	2.08	16.10	86.84	49	2.08	11.08	86.84
		1483	47	48	2.13	11.30	96.84	48	2.13	11.30	96.84
		1515	46	47	2.17	11.55	99.82	47	2.17	11.55	99.82
		1548	45	46	2.22	11.92	89.82	46	2.22	11.92	89.82
		1584	44	45	2.27	11.63	88.89	45	2.27	11.63	88.89
		1620	43	44	2.33	11.91	92.75	44	2.33	11.91	92.75
		1659	42	43	2.38	11.97	99.99	43	2.38	11.97	99.99
		1699	41	42	2.44	12.28	94.81	42	2.44	12.28	94.81
		1742	40	41	2.50	11.08	100.00	41	2.50	11.08	100.00
		1787	39	40	2.56	11.96	79.01	40	2.56	11.96	79.01
		1834	38	39	2.63	11.45	99.81	39	2.63	11.55	99.81
		1883	37	38	2.70	12.22	92.57	38	2.70	12.30	92.57
		1935	36	37	2.77	12.30	92.31	37	2.77	12.10	92.31
		1991	35	36	2.86	12.10	98.18	36	2.86	11.55	98.18
		2049	34	35	2.94	11.55	98.18	35	2.94	12.03	98.18
		2111	33	34	3.03	12.30	90.00	34	3.03	12.85	90.00

ตารางที่ 4.10 การเปรียบเทียบผลของวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับ วิธีระบบ MMAS (no local search) (ต่อ)

Problems	Size	Cycle time	Optimal solution	MMAS**				DE1***			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
		2177	32	33	3.13	12.10	98.78	33	3.13	11.55	98.78
		2247	31	32	3.23	11.55	100.00	32	3.23	11.92	100.00
		2322	30	31	3.33	12.03	86.84	31	3.33	11.63	86.84
		2402	29	30	3.45	12.85	96.84	30	3.45	11.91	96.84
		2488	28	29	3.57	12.84	99.82	29	3.57	11.97	99.82
		2580	27	28	3.70	12.81	92.57	28	3.7	12.28	92.57
		2680	26	27	3.85	11.77	92.31	27	3.85	11.08	92.31
		2787	25	26	4.00	12.63	98.18	26	4.00	11.99	98.18
Total optimal solution found from 25 instances problem				**0**	**2.81/ inst.**	**12.17/ inst.**	**94.61/ inst.**	**0**	**2.81/ inst.**	**11.8/ inst.**	**94.61/ inst.**

หมายเหตุ: m* is the optimal solution
** MMAS (no local search) by Nuchsara and et al (2007a; 2007b)
*** DE1 = DE/best/2 to Exponential 1 position
กำหนดให้ค่า F = 0.9 CR = 0.8 N = 30, Round = 30 Ronkkonen, Kukkonen and Price (2005), m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance

จากตารางที่ 4.10 แสดงผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ในโจทย์ปัญหาขนาดใหญ่ (75-297 ขั้นงาน) ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับวิธีระบบ MMAS (no local search) และเปรียบเทียบกับค่าคำตอบ Data set ซึ่งได้ข้อมูลมาจาก http://www.assembly-line-balancing.de โดยใช้โจทย์ปัญหาตัวอย่าง 3 ชุด โดยชุดปัญหาจากตารางที่ 4.10 ชุดปัญหาของ Scholl ที่มี 297 ขั้นงาน และมี 25 Instances ดังแสดงในตารางพบว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 การค้นหาค้นหาคำตอบที่ดีที่สุดวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถค้นหาคำตอบใช้เวลาที่น้อยกว่าวิธีระบบ MMAS (no local search)

สรุป จากตารางที่ 4.8, 4.9 และ 4.10 ตามลำดับ ทั้งหมดปัญหาขนาดใหญ่ (75-297 ขั้นงาน) แสดงผลการทดลองการจัดสมดุลสายการประกอบแบบตัวยู ในโจทย์ปัญหาขนาดใหญ่ (75-297 ขั้นงาน) ด้วยวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 กับวิธีระบบ MMAS (no local search และเปรียบเทียบกับค่าคำตอบ Data set ซึ่งได้ข้อมูลมาจาก http://www.assembly-

line-balancing.de สรุปในภาพรวมโดยใช้โจทย์ปัญหาตัวอย่าง 3 ชุด (ชุดปัญหาของ Wee-mag, Arcus 1 และ Scholl) ที่มี 75-297 ขั้นงาน และมี 62 Instances จากผลลัพธ์ในการหาค่าจำนวนสถานีงานที่น้อยที่สุด ดังแสดงในตารางพบว่าวิธีวิวัฒนาการโดย ใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถหาคำตอบที่ดีที่สุดได้ 36 คำตอบจาก 62 Instances ซึ่งสามารถหาค่าคำตอบที่ดีที่สุดได้ดีกว่า วิธีระบบ MMAS (no local search) ที่หาค่าคำตอบที่ดีที่สุด 35 คำตอบจาก 36 Instances และการค้นหาค้นหาคำตอบที่ดีที่สุดวิธีวิวัฒนาการโดยใช้ผลต่างแบบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สารถค้นหาคำตอบใช้เวลาที่น้อยกว่า วิธีระบบ MMAS (no local search) ซึ่งจากการเปรียบเทียบดังตารางที่ 4.7 เป็นชุดปัญหาขนาดกลาง (21-45 ขั้นงาน) และจากตารางที่ 4.8, 4.9 และ 4.10 ตามลำดับ เป็นชุดปัญหาขนาดกลาง (75-297 ขั้นงาน) และได้มีการเปรียบวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบกับวิธีเมตาฮิวริสติกแบบอื่น ๆ ในปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 ดังภาคผนวก ง ที่ตารางที่ ง.2 - ง.6 กับชุดปัญหาแบบตัวยู UALBP-1 ที่มีขนาดกลาง (21-45 ขั้นงาน) และ ชุดปัญหาแบบตัวยู UALBP-1 ที่มีขนาดใหญ่ (75-297 ขั้นงาน)

ดังนั้นจากผลการทดลองจะเห็นได้ว่าวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบกับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 สามารถค้นหาคำตอบที่ดีที่สุดได้ดีเมื่อเทียบกับชุดปัญหา dataset และมีประสิทธิภาพในการค้นหาคำตอบที่ดีที่สุดที่ดีกว่าวิธีเมตาฮิวริสติกแบบอื่น ๆ และใช้เวลาในการค้นหาคำตอบที่ดีที่สุดในระยะเวลาน้อยกว่าวิธีเมตาฮิวริสติกแบบอื่น ๆ

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

สำหรับการวิจัยในครั้งนี้ เป็นการพัฒนาวิธีเมตาฮิวริสติกสมัยใหม่ คือวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบ ในการแก้ปัญหาในการจัดสมดุลสายการประกอบแบบตัวยู การวิจัยสรุปผลได้ดังนี้ต่อไปนี้

5.1 สรุปผลการวิจัย

ผลการแก้ปัญหาการจัดสมดุลสายการประกอบตัวยูด้วยวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและวิธีวิวัฒนาการโดยใช้ผลต่างปรับปรุงคำตอบเมื่อเทียบกับวิธีอื่น ๆ

จากการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 ได้ทำการประยุกต์โปรแกรมภาษา Java (รันบนเครื่องคอมพิวเตอร์รุ่น Core i3, 2.3 GHz, 2GB RAM บนระบบปฏิบัติการ Windows 7) วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน และวิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ ซึ่งในการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 (UALBP-1) เป็นการหาค่าคำตอบของจำนวนสถานีงานที่น้อยที่สุด ของแต่ละชุดโจทย์ปัญหาตัวอย่าง ที่มีการกำหนดค่ารอบเวลาการผลิต และค่าคำตอบที่เหมาะสมที่สุด ซึ่งได้ข้อมูลมาจาก http://www.assembly-line-balancing.de และจากงานวิจัยที่ผ่านมาเพื่อใช้ในการอ้างอิงและเปรียบเทียบผล โดยชุดโจทย์ปัญหาตัวอย่างที่ใช้ในปัญหาแบบตัวยู ได้แก่ ปัญหาขนาดต่าง ๆ ดังนี้

5.1.1 ปัญหาแบบตัวยูที่มีขนาดกลาง (21-45 ขั้นงาน) ที่เป็น Data set ในการทดลองวิธี Integer Programming solution, result from Urban (1998) ทั้งหมด 25 Instances โดยรายละเอียดของผลการทดลองดังภาคผนวก ง ที่ตารางที่ ง.2 และ ตารางที่ ง.3 และตารางสรุปผลได้ดังตารางที่ 5.1 ดังนี้

ตารางที่ 5.1 การสรุปผลข้อมูลจากตารางที่ ง.2 และตารางที่ ง.3 Data Set ของ วิธี Integer Programming solution, result from Urban (1998) ในตัวอย่างปัญหา UALBP-1 ที่มีขนาดกลาง

วิธีการ	รายการ			
	No. of instances	จำนวนที่เจอ Optimal	ค่าเฉลี่ยของค่าความแตกต่าง (%)	เวลาเฉลี่ยที่ใช้ (Sec./instance)
Integer Programming solution*	25	25	0	N/A
DE/rand/1 to Binomial Crossover	25	25	0	1.36

ตารางที่ 5.1 การสรุปผลข้อมูลจากตารางที่ ง.2 และตารางที่ ง.3 Data Set ของ วิธี Integer Programming solution, result from Urban (1998) ในตัวอย่างปัญหา UALBP-1 ที่มีขนาดกลาง (ต่อ)

วิธีการ	รายการ			
	No. of instances	จำนวนที่เจอ Optimal	ค่าเฉลี่ยของค่าความแตกต่าง (%)	เวลาเฉลี่ยที่ใช้ (Sec./instance)
DE/rand/1 to Exponential Crossover 1 Position	25	25	0	0.49
DE/rand/1 to Exponential Crossover 2 Position	25	25	0	0.24
DE/ rand-to-best/1 to Binomial Crossover	25	25	0	0.98
DE/Best/2 to Exponential Crossover 1 Position	25	25	0	0.09
DE/Best/2 to Exponential Crossover 2 Position	25	25	0	0.14

หมายเหตุ: * Integer Programming solution, result from Urban (1998: 740) (in table1)

จากผลการทดลองในตัวอย่างปัญหาแบบตัวยูที่มีขนาดกลาง (21-45 ขั้นงาน) พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างเป็นวิธีที่ค้นหาคำตอบได้คำตอบที่ดีที่สุดของจำนวนสถานีงานเท่ากันทุกวิธี แต่วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถหาค่าคำตอบได้ดีที่สุดโดยใช้เวลาในการหาคำตอบน้อยที่สุด คือ ให้ค่าคำตอบที่ดีที่สุดใช้เวลาในการค้นหาค่าคำตอบที่ดีที่สุด (เวลาเฉลี่ยที่ใช้ในการคำนวณค้นหาคำตอบ 0.09 วินาที/ Instances) ตามด้วยวิธี DE/Best/2 to Exponential Crossover 2 Position DE/rand/1 to Exponential Crossover 2 Position, DE/rand/1 to Exponential Crossover 1 Position, DE/ rand-to-best/1 to Binomial Crossover และ DE/rand/1 to Binomial Crossover ตามลำดับ

ดังนั้นสรุปได้ว่า วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 เป็นวิธีการที่มีประสิทธิภาพในการค้นหาคำตอบที่ใช้เวลาน้อยที่สุดในการหาคำตอบที่ดีที่สุด และ วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานเป็นวิธีที่แย่ที่สุดโดยใช้เวลาเยอะที่สุดในการค้นหาคำตอบที่ดีที่สุดเมื่อเทียบวิธีอื่น ๆ ที่เป็นวิธีวิวัฒนาการโดยใช้ผลต่างด้วยกัน ส่วนการ

ค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานได้ค่าคำตอบเท่ากันทุกวิธีเมื่อเทียบวิธีอื่น ๆ ที่เป็นวิธีวิวัฒนาการโดยใช้ผลต่างด้วยกัน

5.1.2 ปัญหาแบบตัวยูที่มีขนาดใหญ่ (75-297 ขั้นงาน) ที่เป็น Data set ในการทดลองวิธี MMAS (no local search) by Nuchsara and et al (2007a; 2007b) ทั้งหมด 62 Instances โดยรายละเอียดของผลการทดลองดังภาคผนวก ง ที่ตารางที่ ง.4 ตารางที่ ง.5 และตารางที่ ง.6 ตารางสรุปผลได้ดังตารางที่ 5.2 ดังนี้

ตารางที่ 5.2 การสรุปผลข้อมูลจาก ตารางที่ ง.4 ตารางที่ ง.5 และตารางที่ ง.6 Data Set ของวิธี Optimal solution (1991) ในตัวอย่างปัญหา UALBP-1 ที่มีขนาดใหญ่

วิธีการ	รายการ			
	No. of instances	จำนวนที่เจอ Optimal	ค่าเฉลี่ยของค่าความแตกต่าง (%)	เวลาเฉลี่ยที่ใช้ (Sec./instance)
MMAS (no local search)*	62	35	0	5.72
DE/rand/1 to Binomial Crossover	62	36	0	5.65
DE/rand/1 to Exponential Crossover 1 Position	62	36	0	5.48
DE/rand/1 to Exponential Crossover 2 Position	62	36	0	5.34
DE/ rand-to-best/1 to Binomial Crossover	62	36	0	5.56
DE/Best/2 to Exponential Crossover 1 Position	62	36	0	4.88
DE/Best/2 to Exponential Crossover 2 Position	62	36	0	5.13

หมายเหตุ: * MMAS (no local search) research by Nuchsara and et al (2007a; 2007b)

จากผลการทดลองในตัวอย่างปัญหาแบบตัวยูที่มีขนาดใหญ่ (75-297 ขั้นงาน) พบว่าวิธีวิวัฒนาการโดยใช้ผลต่างเป็นวิธีที่ค้นหาคำตอบได้คำตอบที่ดีที่สุดของจำนวนสถานีงานเท่ากันทุกวิธี แต่วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 สามารถหาค่าคำตอบได้ดีที่สุดโดยใช้เวลาในการหาคำตอบน้อยที่สุด คือ ให้ค่าคำตอบที่ดีที่สุดใช้เวลาในการค้นหาค่าคำตอบที่ดีที่สุด (เวลาเฉลี่ยที่ใช้ในการคำนวณค้นหาคำตอบ 4.88 วินาที/Instances) ตามด้วยวิธี DE/Best/2 to Exponential Crossover 2 Position, DE/rand/1 to Exponential Crossover 2 Position, DE/rand/1 to Exponential Crossover 1 Position, DE/

rand-to-best/1 to Binomial Crossover และ DE/rand/1 to Binomial Crossover ตามลำดับ และ ผลลัพธ์ในการหาค่าจำนวนสถานีงานที่น้อยที่สุด วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานและแบบปรับปรุงคำตอบสามารถหาคำตอบที่ดีที่สุดได้ 36 คำตอบจาก 62 Instances ซึ่งสามารถหาค่าคำตอบที่ดีที่สุดได้ดีกว่า วิธีระบบ MMAS (no local search) ที่หาค่าคำตอบที่ดีที่สุด 35 คำตอบจาก 36 Instances

ดังนั้นสรุปได้ว่า วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ DE/Best/2 ร่วมกับ Exponential Crossover 1 เป็นวิธีการที่มีประสิทธิภาพในการค้นหาคำตอบที่ใช้เวลาน้อยที่สุดในการหาคำตอบที่ดีที่สุด และ วิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐานเป็นวิธีที่แย่ที่สุดโดยใช้เวลาเยอะที่สุดในการค้นหาคำตอบที่ดีที่สุดเมื่อเทียบวิธีอื่น ๆ ที่เป็นวิธีวิวัฒนาการโดยใช้ผลต่างด้วยกัน ส่วนการค้นหาคำตอบที่ดีที่สุดของจำนวนสถานีงานได้ค่าคำตอบเท่ากันทุกวิธีเมื่อเทียบวิธีอื่น ๆ ที่เป็นวิธีวิวัฒนาการโดยใช้ผลต่างด้วยกัน

จากผลการทดลองและวิเคราะห์ผลสรุปผลการเปรียบเทียบปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 1 วิธีวิวัฒนาการโดยใช้ผลต่างกับวิธีเมตาฮิวริสติกวิธีอื่น ๆ ปัญหาขนาดกลาง (21-45 ขั้นงาน) วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบและแบบพื้นฐานให้ค่าคำตอบที่ดีที่สุดในการหาจำนวนสถานีงานดีกว่าและใช้เวลาในการค้นหาคำตอบที่ดีที่สุดน้อยกว่า วิธีระบบมด และ วิธีระบบมด MMAS (no local search) ส่วนวิธี GA วิธีวิวัฒนาการโดยใช้ผลต่างปรับปรุงคำตอบและแบบพื้นฐานจะใช้เวลาในการค้นหาคำตอบที่ดีที่สุดน้อยกว่า วิธี GA แต่การค้นหาคำตอบที่ดีที่สุดจำนวนสถานีงานให้ค่าคำตอบที่ดีที่สุดเท่ากัน และ ปัญหาขนาดใหญ่ (75-297 ขั้นงาน) วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบและแบบพื้นฐานให้ค่าคำตอบที่ดีที่สุดในการหาจำนวนสถานีงานดีกว่าและใช้เวลาในการค้นหาคำตอบที่ดีที่สุดน้อยกว่า วิธีระบบมด MMAS (no local search)

5.2 ข้อเสนอแนะและแนวทางวิจัยในอนาคต

5.2.1 ควรมีการพัฒนาวิธีวิวัฒนาการโดยใช้ผลต่างในโจทย์ปัญหาที่ยากขึ้น เช่น Mixed/Multi model line, Stochastic task time และ U-line ในลักษณะอื่น ๆ

5.2.2 ควรมีการพัฒนาหลักการเมตาฮิวริสติกอื่น ๆ เพื่อแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ในปัญหาขนาดใหญ่

5.2.3 ควรมีการพัฒนาหลักการเมตาฮิวริสติกอื่น ๆ หรือวิธีการต่าง ๆ เพื่อแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู ประเภทที่ 2 (Type-2) กำหนดค่าจำนวนสถานีงาน (m) และหาค่ารอบเวลาผลิตที่น้อยที่สุด (Min.c, given m) และ ประเภทที่ E กำหนดค่า c และ m หาค่าประสิทธิภาพของสายการผลิตสูงสุด (E) (Min.E, given c, m)

เอกสารอ้างอิง

เอกสารอ้างอิง

กนกกาญจน์ จิรศิริเลิศ และระพีพันธ์ ปิตาคะโส. “การประยุกต์ใช้วิธีการวิวัฒนาการโดยใช้ผลต่างในการจัดสมดุลสายการประกอบเส้นตรงประเภทที่ 1: กรณีศึกษา อุตสาหกรรมผลิตเสื้อผ้าสำเร็จรูป”, **วารสารไทยการวิจัยดำเนินงาน**. 1(2): 39-50; กรกฎาคม-ธันวาคม, 2556.

กาญจนา เศรษฐนันท์. **เมตะฮิวริสติกและการประยุกต์ใช้ในงานอุตสาหกรรม**. ขอนแก่น: คลังนานาวิทยา, 2558.

ทองพูน ทองดี และระพีพันธ์ ปิตาคะโส. “การแก้ปัญหาการเลือกสถานที่ตั้งและการมอบหมายเส้นทางแบบหลายลำดับขั้น หลายต้นกำเนิดและหลายจุดประสงค์ด้วยกระบวนการโมดิฟายดิฟเฟอเรนเชียลอีโวลูชั่น”, **วารสารวิชาการ มข**. 18(4): 662-676; มกราคม –มิถุนายน, 2555.

นุชสรา เกรียงกรกฎ. **วิธีการเมตาฮิวริสติกสำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบ**. วิทยานิพนธ์ปริญญาปรัชญาดุษฎีบัณฑิต: มหาวิทยาลัยอุบลราชธานี, 2550.

ปารเมศ ชุติมา และคณะ. “การประยุกต์ใช้ขั้นตอนวิธีการบรรจวบสำหรับปัญหาการจัดสำหรับผลิตภัณฑ์ที่มีหลายวัตถุประสงค์บนสายการประกอบผลิตภัณฑ์ผสมแบบตัวยูในระบบผลิตแบบทันเวลาพอดี”, ใน **การประชุมทางวิชาการมหาวิทยาลัยเกษตรศาสตร์ครั้งที่ 47**. น.57-66. กรุงเทพฯ: มหาวิทยาลัยเกษตรศาสตร์, 2552.

ภาณุภัณฑ์ ภาระเวช และคณะ. “การแก้ปัญหาการจัดสมดุลสายการประกอบโดยใช้วิธีการดิฟเฟอร์เรนเชียลอิโวลูชั่น กรณีศึกษาโรงงานตัดเย็บเสื้อผ้าสำเร็จรูป”, **วารสารมหาวิทยาลัยนราธิวาสราชนครินทร์**. 6(2): 92-104; กรกฎาคม-ธันวาคม, 2557.

อมรพงษ์ สงวนสินธ์ และจักรวาล คุณะดิลก. “วิธีฮิวริสติกสำหรับลดค่าความแปรปรวนของภาระงานในการจัดสมดุลสายการประกอบรูปทรงตัวยู”, **วารสารไทยการวิจัยดำเนินงาน**. 2(2): 11-21; กรกฎาคม-ธันวาคม, 2557.

เอกชัย คุปตาวาทิน และระพีพันธ์ ปิตาคะโส. “การประยุกต์ใช้วิธีการหาค่าที่เหมาะสมแบบวิวัฒนาการคำตอบ สำหรับการจัดสรรวัตถุดิบ ในการขนส่งมันสำประหลังแบบหลายระดับชั้น”, **วารสารวิชาการ มข**. 18(3): 419-434; กรกฎาคม-ธันวาคม, 2555.

Andreas, C. N. “A Differential Evolution Algorithm For Simple Assembly Line Balancing”, In **Proceedings on 16th Triennial World Congress of. International Federation of Automatic Control IFAC 2005**. p.1462-1467. Czech Republic: Elsevier-IFAC Publications, 2005.

Bowman, E. H. “Assembly Line Balancing by Linear Programming”, **Operations Research**. 8(3): 385-389; July-December; 1960.

Chanahan, B. J. **Incorporating Ergonomics Criteria into Assemble Line Balancing**. Doctor’s Thesis: University of Pittsburgh, 1999.

เอกสารอ้างอิง (ต่อ)

Chen, S. **Just-In-Time U-Shaped Assembly Line Balancing**. Doctor's Thesis: Lehigh University, 2003.

Chiang, W. C. and Urban, T. L. "The Stochastic U-line Balancing Problem: A Heuristic Procedure", **European Journal of Operational Research**. 175(3): 1767-1781; January-June, 2006.

Dervis, K. and Selcuk, O. "A Simple and Global Optimization Algorithm for Engineering Problems: Differential Evolution Algorithm", **Turkish Journal of Electrical Engineering & Computer Sciences**. 12(1): 53-60; July-December, 2004.

Dexuan, Z. and et al. "An improved differential evolution algorithm for the task assignment problem", **Engineering Applications of Artificial Intelligence**. 24(1): 616-624; January-June, 2011.

Erel, E. and et al. "Balancing of U-type Assembly Systems Using Simulated Annealing", **International Journal of Production Research**. 39(13): 3003-3015; January-April, 2001.

Gokcen, H. and Agpak, K. "A shortest route formulation of simple U-type assembly line balancing problem", **Applied Mathematical Modelling**. 98(4): 373-380; October, 2006.

Gokcen, H. and et al. "A shortest route formulation of simple U-type assembly line balancing problem", **Applied Mathematical Modelling**. 22(4): 373-380; May-August, 2005.

Gutjahr, A. L. and Nemhauser, G. L. "An algorithm for the line balancing problem", **Management Science**. 11(2): 308-315; January-June, 1964.

Hwang, R., Katayama, H. and Gen, M. "U-shaped assemble line balancing problem withgenetic algorithm", **International Journal of Production Research**. 46(16): 4637 – 4649; January-June, 2008.

Jaeschke, G. "Eineallgemaine Methods Zur Losung Kombinatorisher Probleme", **Ablaujund Planung-forschung**. 5(1): 133-153; July-December, 1964.

Martinez, U. and Duff, W. S. "Heuristic approaches to solve the U-shaped line balancing problem augmented by Genetic Algorithms", **Proceeding of the 2004 System and Information Engineering Design Symposium**. 11(4): 246-259; January-June, 2004.

McMullen, P. R. and Frazier, G. V. "Using Simulated Annealing to Solve a Multiobjective Assembly Line Balancing Problem with parallel workstations", **International Journal of Production**. 36(10): 2717-2741; May-August, 1998.

เอกสารอ้างอิง (ต่อ)

Miltenburg, J. “Balancing U-lines in a multiple U-line facility”, **European Journal of Operational Research**. 15(3): 1-23; July-December, 1998.

Miltenburg, J. and Wijingaard, J. “The U-line Line Balancing Problem”, **Management Sciences**. 40(10): 1378-1388; January-June, 1994.

Miltenburg, J. and Sparling, D. **Optimal solution algorithms for the U-line balancing problem**. Master’s Thesis: University, Hamilton, 1995.

Monden, Y. **Toyota Production System**. Norcross, GA: Industrial Engineering and Management Press, Institute of Industrial Engineering, 1993.

Nuchsara, K. and Nalin, P. “The Assembly Line Balancing Problem: Review articles”, **KKU Engineering Journal**. 34(2): 133-140; May, 2007.

Nuchsara, K. and et al. “Balancing of U-shaped assembly line. Proceeding of Papers (CD ROM)”, In **The 2nd International Conference on Operations and Supply Chain management (OSCM2007)**. p.513-519. Bangkok: Novotel Siam Square Hotel, 2007a.

_____. “The Ant Algorithm for U-Shaped Assembly Line Balancing. Proceeding of Papers (CD ROM)”, In **The PSU-UNS International Conference on Engineering and Environment (ICEE2007)**. p.89-93. Phuket: Phuket Graceland Resort and Spa Hotel, 2007b.

Prasit, K. “Production of Daikin System”, **Daikinthai**. http://www.daikinthai.com. 6 April, 2015.

Price, K. and Storn, R. “A Simple and Efficient Heuristic for global Optimization over Continuous Spaces”, **Journal of Global Optimization**. 11(4): 341–359; December, 1997.

Qin, A. K. and et al. “Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization”, **IEEE Transactions on Evolutionary Computation**. 13(2): 398-417; January-June, 2009.

Rabbani, K. and et al. “A Heuristic Method for the Simple Assembly Line Balancing Problem”, **IEEE Fourth International Conference on Natural Computer**. 13(2): 402-416; May, 2012.

Rapeepan, P. “Differential evolution algorithm for simple assembly line balancing type 1 (SALBP-1)”, **Journal of industrial and Production Engineering**. 32(2): 104-114; October, 2015.

เอกสารอ้างอิง (ต่อ)

Rea, K. H. and et al. “U-shaped assembly line balancing problem with Genetic algorithm”, **International Journal of Production Research**. 46(16): 4637-4649; January-June, 2008.

Ronkkonen, J., Kukkonen, S. and Price, K. V. “Real-Parameter Optimization with Differential Evolution”, **IEEE Congress on Evolutionary Computation**. 1(1): 506-513; July-December, 2005.

Sabuncuoglu, L. and et al. “A Simple Assembly Line Balancing Problem”, **International Journal of Production Research**. 76(2): 117-131; March, 2009.

Scholl, A. and Becker, C. “State-of-the-art Exact and Heuristic Solution Procedures for Simple Assembly Line Balancing”, **European Journal of Operational Research**. 168(3): 666-693; May-August, 2006.

Scholl, A. and Klein, R. “ULINO: optimally balancing U-Shaped JIT assemble lines”, **International of Production Research**. 37(4): 721-736; July, 1999.

Scholl, A. **Balancing and Sequencing of Assembly**. Publisher: Springer, 1999.

Scholl, A. (1993). “Data of Assembly Line Balancing Problems”, **Data**. http://www.assembly-line-balancing.de/. 10 May, 2016.

Sparling, D. and Miltenburg, J. “The mixed-model U-line balancing problem”, **International Journal of Production Research**. 36(2): 485-501; November, 1998.

Suneeya, U. “ระบบสายการประกอบรถยนต์แบบสินค้าแบบผสม”, **Headlightmag**. http://www.headlightmag.com. 9 March, 2014.

Thailand Honda. “Assembly CAR”, **Automobile**. http://www.HondaAutomobile.com/. 10 May, 2016.

Thailandindustry.com. “Industry”, **Thailandindustry New**. http://www.thailandindustry.com/indust_newweb/index.php. 14 January, 2017.

Urban, T. L. “Optimal Balancing of U-Shaped Assembly Lines”, **Management Sciences**. 44(5): 738-741; July-December, 1998.

Zeng, X. and et al. “An operator allocation Optimization model for balancing control of the hybrid assembly lines using Pareto utility discrete differential evolution algorithm”, **Computers and Operations Research**. 39(1): 1145-1159; February, 2012.

เอกสารอ้างอิง (ต่อ)

Zhang, Z. and Sanney, K. “A novel ant colony optimization algorithm for U-shaped line balancing problem”, **IEEE Fourth International Conference on Natural Computer**. 1(1): 445-459; June, 2009.

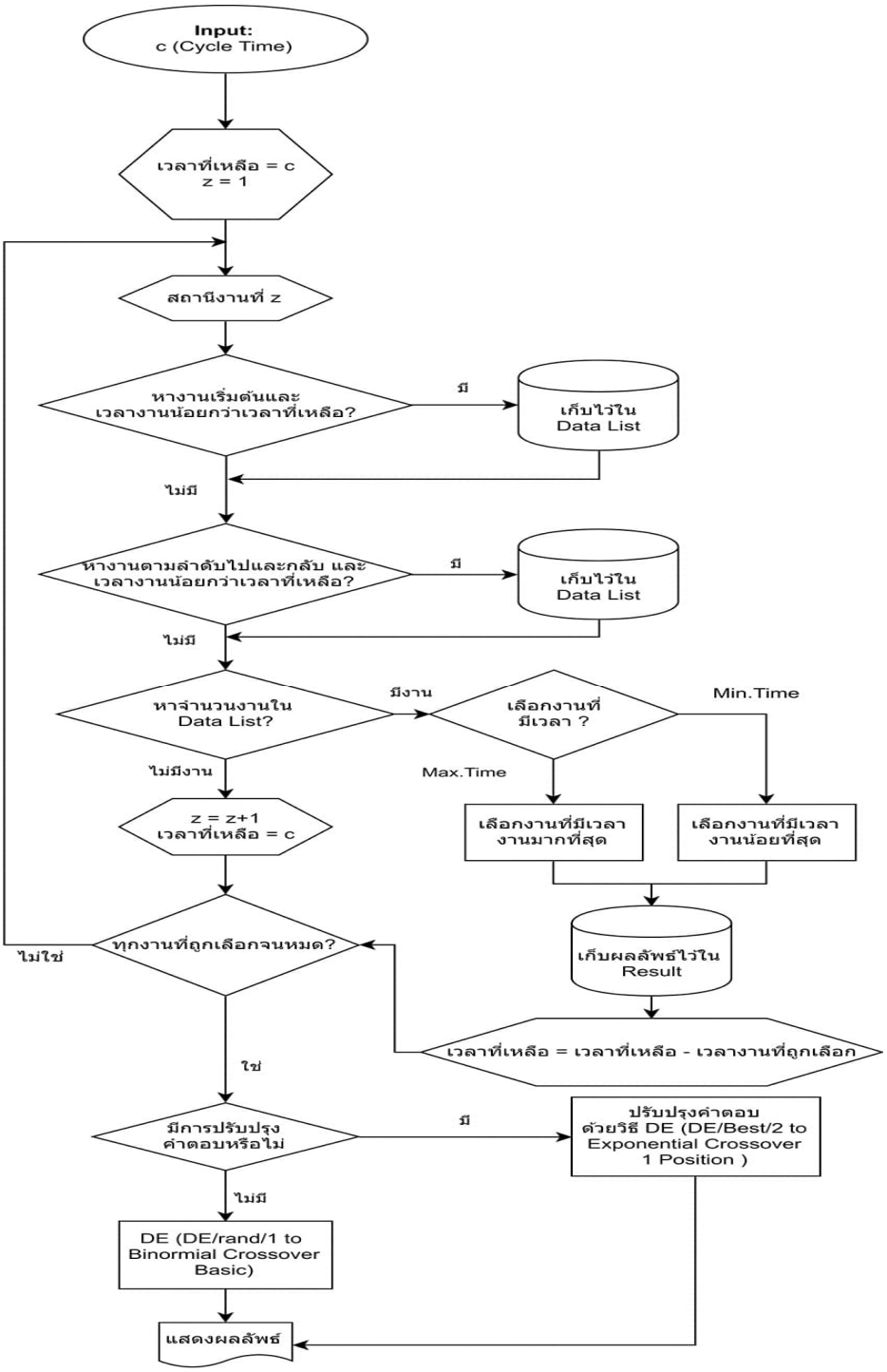
ภาคผนวก

ภาคผนวก ก
Pseudo Code และแผนผังขั้นตอนการทำงานของโปรแกรม
การจัดสมดุลสายการประกอบแบบตัวยู

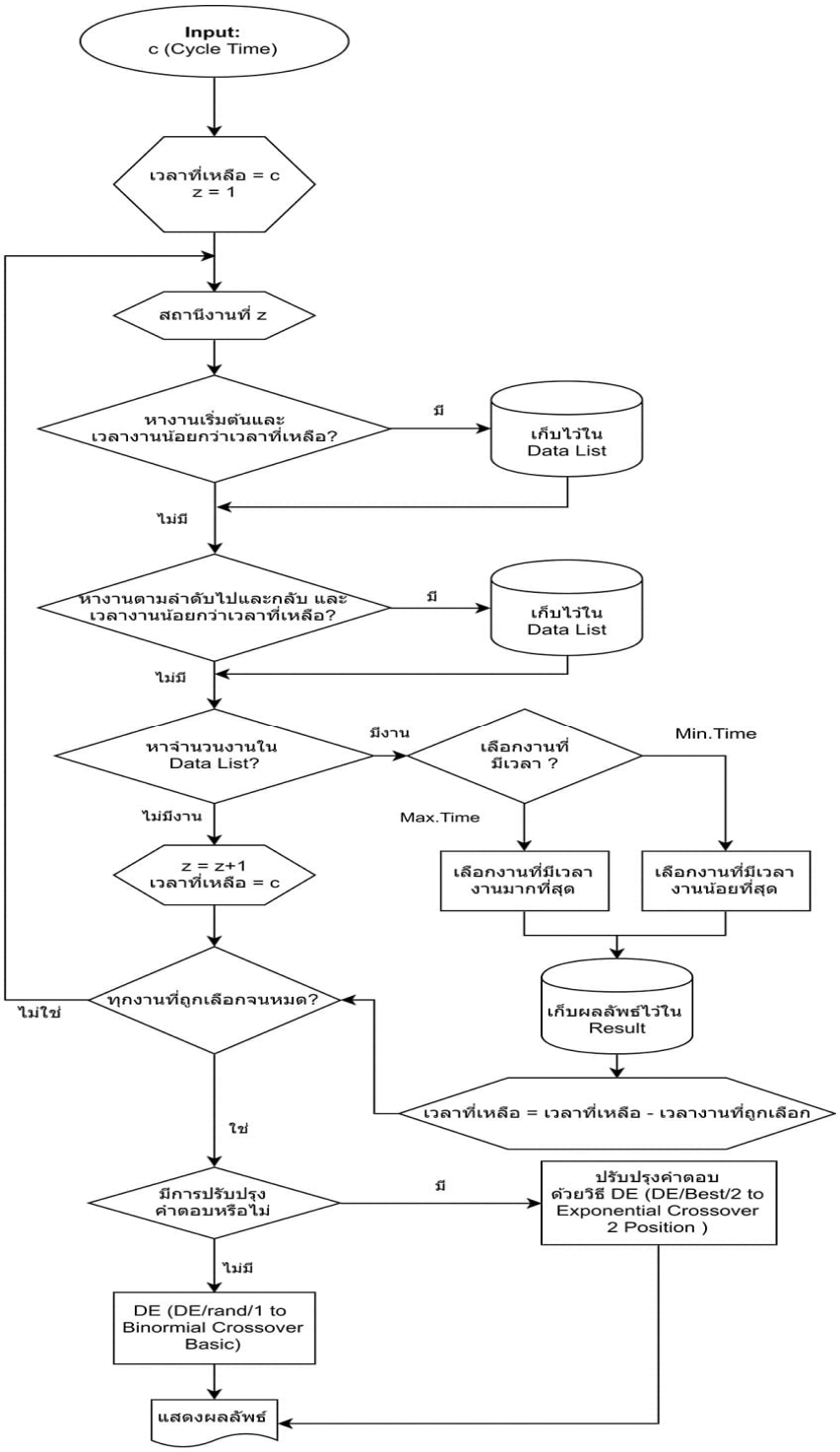
Pseudo Code ของวิธีวิวัฒนาการโดยใช้ผลต่างที่พัฒนาสำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบ

```
Setup initial DE parameter
Do while    from first iteration to final iteration
            Do while    from first DE to final DE
                        Setup initial parameters: cycle time, remaining time, station number
                Do while        from first task to final task
                        Find start/following task with task time is less than or equal to
                                Remaining time, and proper precedence to data list
                        Input  scaling factor, crossover rate and NP to data list
                                Select task randomly to list
                                Update remaining time/station number
                Enddo
            Enddo
            Select best solution from all DE in the iteration
Enddo
Show/select best solution from all DE in all iteration
```

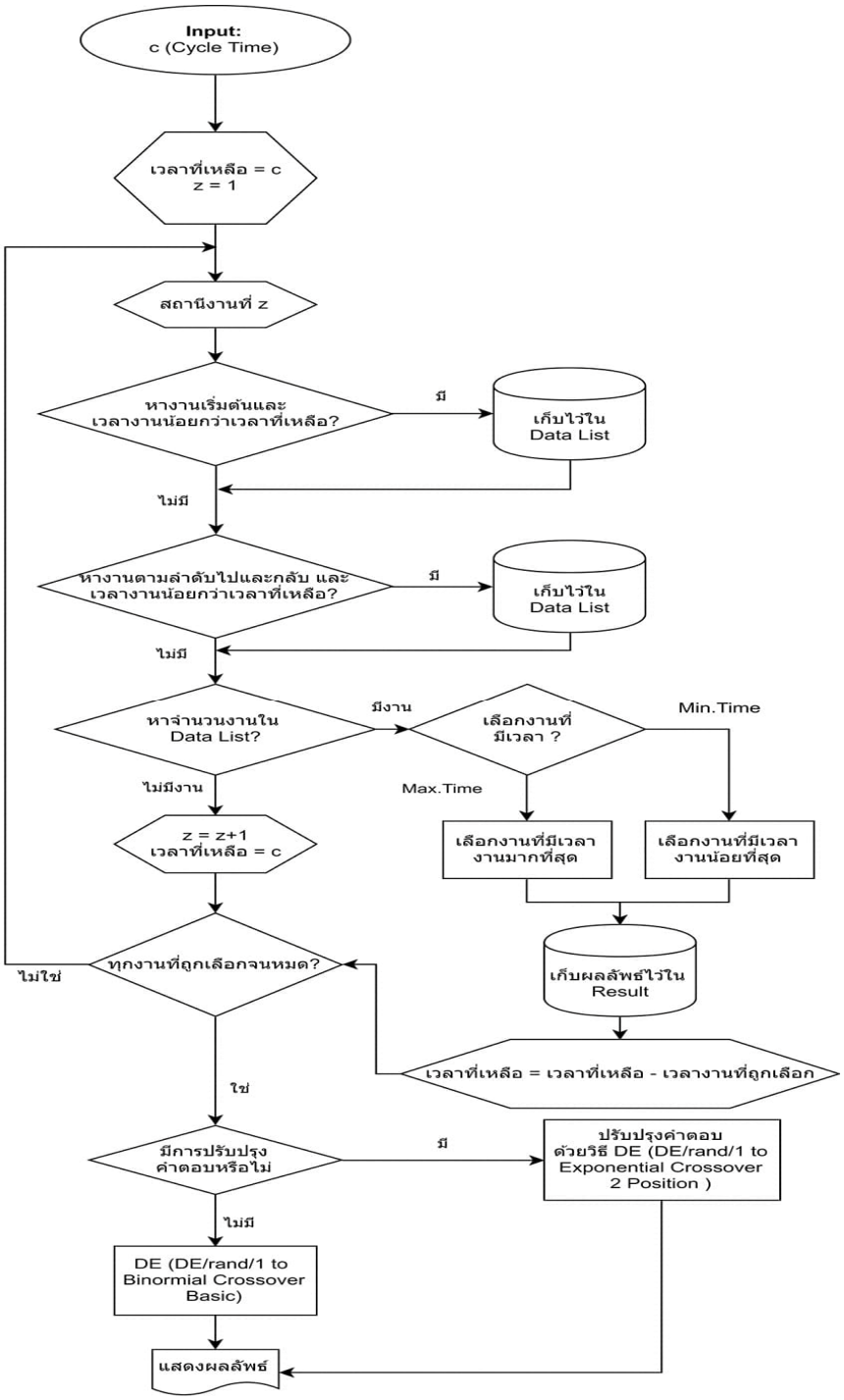
จาก Code ข้างต้น แสดง Pseudo Code ของวิธีวิวัฒนาการโดยใช้ผลต่างสำหรับการแก้ปัญหาการจัดสมดุลสายการประกอบแบบตัวยู โดยกำหนดค่าพารามิเตอร์ต่าง ๆ เช่น R = Round (จำนวนรอบการทดลอง) NP = จำนวนขั้นตอนงาน F = แฟคเตอร์ในการขยายผลต่าง (Scaling factor) และ CR = อัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate) ซึ่งเป็นค่าที่เหมาะสมที่ได้จากการทดลอง ในขั้นตอนของการสร้างคำตอบ จะทำการจัดงานลงสถานีงาน โดยวิธีวิวัฒนาการโดยใช้ผลต่างแบบพื้นฐาน และ วิธีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ ซึ่งจะกระทำขั้นตอนนี้ซ้ำไปจนครบตามจำนวนรอบที่ทำการทดลอง เมื่อเสร็จสิ้นการทำงาน ก็จะพิจารณาผลลัพธ์ที่ให้ค่าคำตอบที่ดีที่สุด มีจำนวนสถานีงานน้อยที่สุด ของการทดลองนั้น ๆ เพื่อนำผลลัพธ์ที่ได้ไปเปรียบเทียบกับคำตอบที่เหมาะสมที่สุด ในลำดับต่อไป



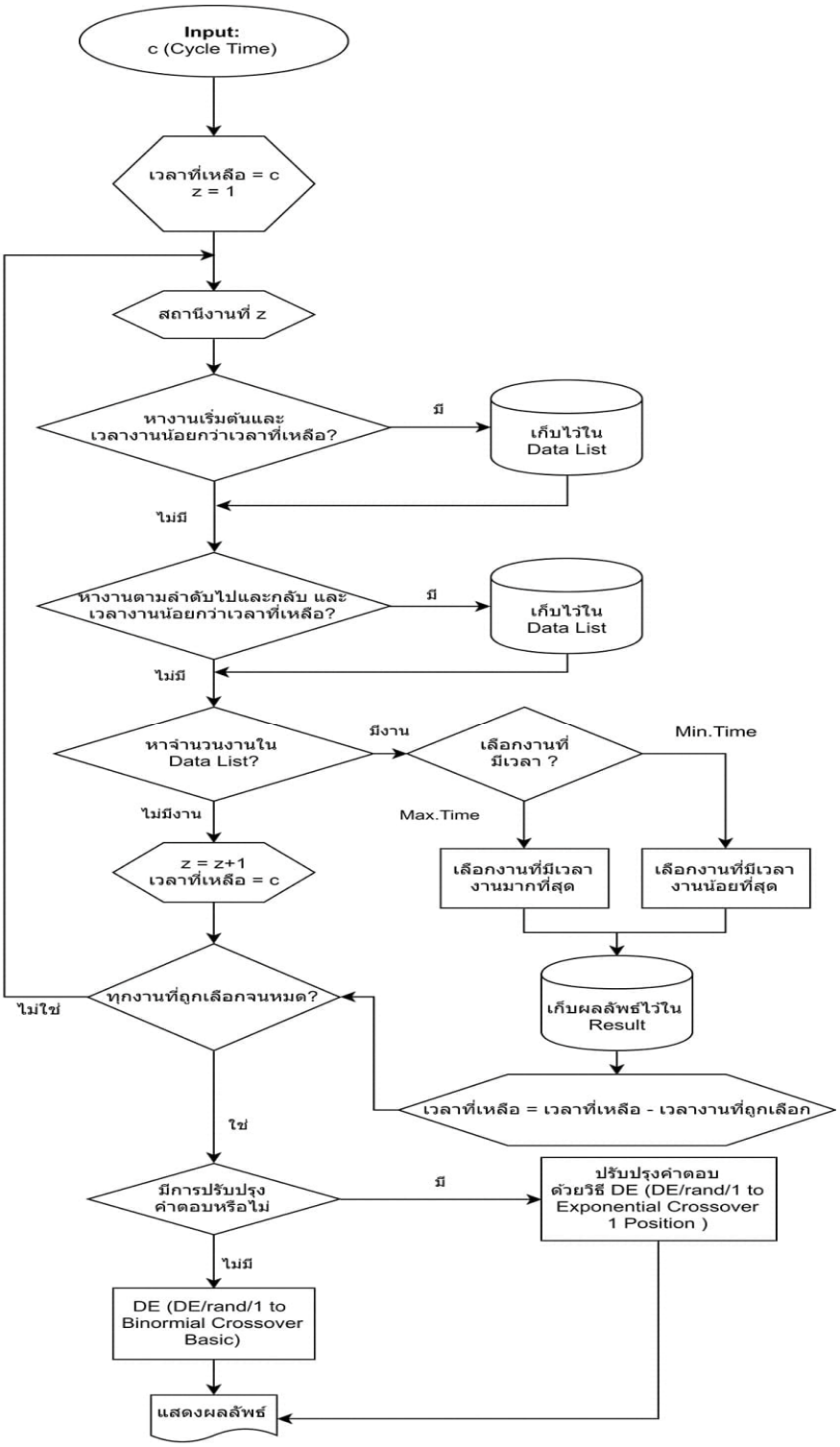
ภาพที่ ก.1 แผนผังขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูโดยวีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ (DE/Best/2 ร่วมกับ Exponential Crossover 1 Position)



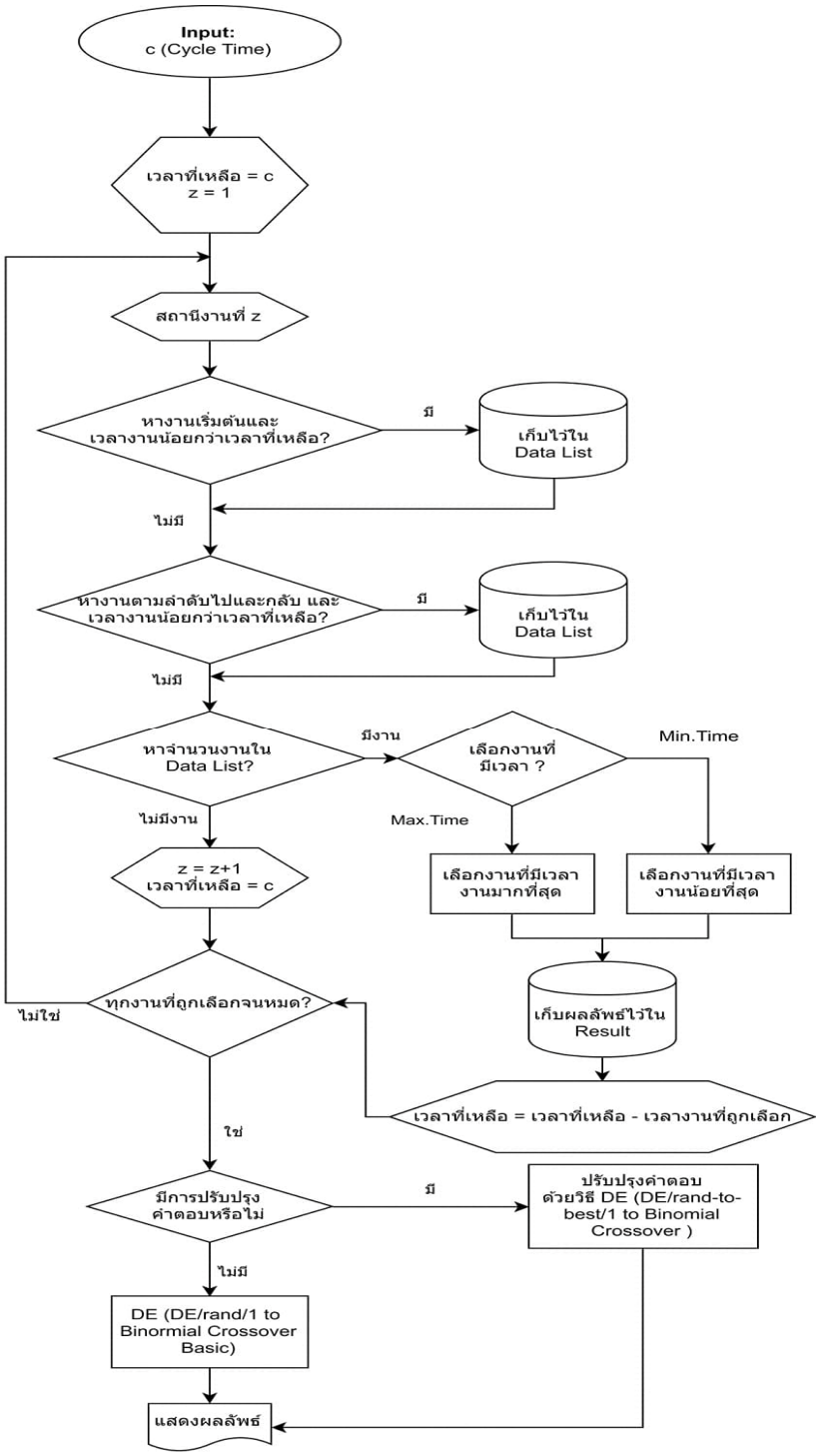
ภาพที่ ก.2 แผนผังขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูโดยวีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ (DE/Best/2 ร่วมกับ Exponential Crossover 2 Position)



ภาพที่ ก.3 แผนผังขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูโดยวีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ (DE/rand /1 ร่วมกับ Exponential Crossover 2 Position)



ภาพที่ ก.4 แผนผังขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูโดยวีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ (DE/rand /1 ร่วมกับ Exponential Crossover 1 Position)



ภาพที่ ก.5 แผนผังขั้นตอนการจัดสมดุลสายการประกอบแบบตัวยูโดยวีวิวัฒนาการโดยใช้ผลต่างแบบปรับปรุงคำตอบ (DE/rand-to-best /1 ร่วมกับ Binomial Crossover)

ภาคผนวก ข
ชุดโจทย์ปัญหาตัวอย่าง (Data set) และข้อมูลที่เกี่ยวข้อง

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
n	83	111	28	45	30	297	21	70
Task time	1673	1960	70	9	8	270	4	17
Direct pre	985	1715	59	9	7	270	3	66
End mark(-1, -1)	1836	735	33	10	19	130	9	54
	973	1715	6	10	10	148	5	52
	1700	490	1	17	2	190	9	6
	2881	1225	27	17	6	293	4	88
	2231	169	17	13	14	348	8	21
	1040	2252	62	13	10	182	7	128
	1793	1225	31	20	1	490	5	68
	1250	2319	53	20	4	212	1	70
	700	1715	21	10	14	248	3	85
	464	980	19	11	15	248	1	21
	500	735	108	6	5	248	5	134
	1133	2281	52	22	12	248	3	135
	577	2750	5	11	9	248	5	94
	483	77	8	19	10	268	3	90
	880	89	97	12	2	268	13	50
	667	51	8	3	10	268	5	143
	600	364	47	7	18	288	2	19
	233	405	67	4	16	248	3	54
	408	3060	17	55	21	268	7	50
	847	125	8	14	14	60	1,2	40
	767	3429	3	27	16	268	1,3	73
	850	43	21	29	7	240	2,21	12
	780	3430	107	26	17	240	3,4	152
	912	1960	3	6	9	171	4,5	42
	748	29	2	5	25	490	4,21	45
	1863	27	72	24	7	182	5,6	74
	714	15	1,3	4	14	170	5,7	26
	1004	121	1,4	5	2	306	6,8	11
	713	1715	1,5	7	1,4	108	7,8	31
	642	2127	1,8	4	1,5	248	7,14	50
	629	1470	1,19	15	2,11	190	8,9	102
	1234	4037	1,21	3	2,12	240	9,10	46
	1143	68	1,22	7	3,16	339	9,11	35
	1266	62	1,23	9	3,17	288	9,12	40
	792	42	1,24	4	4,7	248	9,13	2
	1251	364	1,26	7	5,6	455	10,15	1
	1310	4998	2,6	5	6,7	268	11,15	3
	663	1470	2,17	4	7,8	270	12,15	13
	494	2963	3,28	21	8,9	180	13,17	16
	1288	5689	4,28	12	9,26	121	13,18	25
	792	68	5,28	6	10,24	270	14,19	21
	578	18	6,7	5	12,13	440	15,16	43
	594	10	7,18	5	13,14	249	15,18	30

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
	578	81	8,9	1,3	14,15	194	16,17	83
	622	5200	9,10	1,7	14,20	162	17,20	89
	578	39	10,11	2,4	15,22	130	18,19	56
	564	67	10,12	2,8	16,20	388	-1,-1	59
	578	27	11,15	3,5				
	578	15	12,13	4,6	17,18	90		43
	578	121	13,14	5,9	18,19	212		11
	578	58	13,16	6,10	20,21	246		26
	578	1715	14,28	7,9	20,24	188		44
	578	125	15,28	7,14	21,22	270		121
	578	4010	16,28	8,10	22,23	160		38
	578	1470	17,28	8,14	23,27	79		68
	578	1470	18,28	9,41	24,25	466		22
	578	2303	19,28	10,41	25,26	240		7
	578	1960	20,28	11,13	26,27	137		16
	578	2205	21,28	12,13	27,28	184		32
	578	4018	22,28	12,37	27,29	110		25
	578	2744	23,28	13,14	29,30	275		27
	578	2999	24,28	13,15	-1,-1	149		156
	578	735	25,28	14,17		280		28
	578	735	26,28	14,25		119		15
	578	735	27,28	14,29		184		26
	578	735	-1,-1	14,30		140		18
	467	545		14,31		150		72
	887	3386		14,32		190		23
						150		27
	396	3234		15,16		150		1,2
	1296	2205		15,18		284		1,41
	1100	2206		15,23		192		1,69
	2543	490		15,24		347		1,70
	764	825		16,19		232		2,3
	357	3528		17,26		140		3,4
	701	3568		17,27		608		3,68
	1164	1200		18,19		80		4,6
	286	618		19,20		40		4,7
	2100	1470		19,33		130		5,6
	450	1715		20,21		110		5,24
	1300	735		21,22		350		5,30
	3691	1960		22,28		140		6,8
	1,2	2889		23,33		240		7,8
	2,3	618		24,33		240		8,12
	2,4	490		25,26		90		9,10
	2,5	735		26,38		54		10,11
	3,6	490		27,28		294		11,12
	4,6	921		27,33		203		12,13
	4,7	326		28,38		150		12,14

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
	5,8	5390		29,41		270		13,23
	6,9	243		30,41		155		14,23
	6,10	371		31,41		190		15,16
	7,11	58		32,41		78		16,17
	8,77	5059		33,34		140		16,18
	8,78	1225		33,35		241		17,19
	9,12	769		33,36		430		18,19
	10,13	768		34,38		90		19,20
	10,14	1670		35,40		110		19,22
	10,25	1670		36,38		9		19,57
	11,15	490		37,43		430		20,21
	12,16	202		38,40		130		21,23
	13,17	203		39,41		289		22,23
	13,18	202		40,41		110		23,25
	13,20	2744		41,42		160		23,31
	14,19	162		42,44		442		23,33
	15,20	324		42,45		159		24,25
	15,39	162		-1,-1		250		25,26
	16,77	121				190		25,27
	16,78	162				184		25,28
	17,21	91				690		25,29
	17,22	1,2				72		26,35
	17,28	2,3				190		27,35
	18,23	3,4				190		28,35
	19,24	4,5				90		29,35
	20,26	4,6				889		30,31
	21,27	4,7				170		31,32
	22,27	4,8				155		32,35
	23,74	4,9				190		33,34
	24,28	4,10				130		34,35
	24,29	5,39				390		35,36
	25,32	6,39				301		35,44
	26,74	7,83				54		35,48
	27,69	8,71				227		35,51
	28,32	9,32				142		35,53
	29,30	10,11				184		35,56
	30,31	10,12				741		35,60
	31,39	11,13				868		35,61
	32,33	11,14				230		35,62
	32,34	11,15				121		36,37
	32,35	11,16				320		37,38
	32,36	11,17				126		38,39
	33,37	11,18				440		39,40
	34,77	11,19				127		40,42
	34,78	11,20				134		41,42
	35,77	11,21				150		42,43

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
	35,78	12,13				140		43,50
	36,38	12,14				110		44,45
	36,39	12,15				320		45,46
	37,40	12,16				250		46,47
	38,41	12,17				232		47,50
	39,42	12,18				188		48,49
	39,43	12,19				250		49,50
	39,44	12,20				377		51,52
	39,75	12,21				90		52,54
	40,77	13,71				140		53,54
	40,78	14,22				90		54,55
	41,45	14,23				90		57,58
	42,77	14,24				70		58,59
	42,78	14,25				90		59,60
	43,77	15,26				110		61,65
	43,78	16,27				150		62,63
	44,46	17,28				101		63,64
	45,47	18,29				377		64,65
	46,48	19,30				118		64,66
	47,49	20,91				290		64,67
	48,50	21,111				209		-1,-1
	49,69	22,31				150		
	50,51	22,83				150		
	51,52	23,32				79		
	52,53	23,33				150		
	53,54	24,69				91		
	54,55	24,70				59		
	55,56	25,34				218		
	56,57	26,82				351		
	57,58	27,35				873		
	58,59	28,36				130		
	59,60	29,37				68		
	60,61	30,38				126		
	61,62	31,39				120		
	62,63	32,41				227		
	63,64	33,111				198		
	64,65	34,42				132		
	65,66	35,43				121		
	66,67	36,44				150		
	67,68	36,91				100		
	68,74	37,45				38		
	68,75	37,91				70		
	69,70	38,46				355		
	69,71	38,91				284		
	70,72	39,40				122		
	71,73	40,111				75		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
	72,73	41,69				160		
	73,74	41,70				140		
	73,75	42,47				520		
	74,76	43,48				99		
	75,76	43,49				182		
	76,77	44,50				80		
	76,78	44,51				514		
	77,79	46,52				96		
	78,79	47,54				50		
	79,80	47,55				272		
	79,81	47,56				226		
	80,82	47,57				194		
	81,83	47,58				164		
	82,83	47,59				96		
	-1,-1	47,60				107		
		48,53				108		
		49,91				167		
		50,111				98		
		51,111				82		
		52,111				482		
		53,111				72		
		54,69				50		
		54,70				130		
		55,61				230		
		55,62				50		
		55,63				240		
		56,63				190		
		56,64				190		
		57,65				240		
		57,91				74		
		58,66				139		
		58,91				339		
		59,67				260		
		59,91				132		
		60,68				550		
		60,91				420		
		61,69				152		
		61,70				12		
		62,71				90		
		63,111				5		
		64,72				128		
		65,111				100		
		66,111				120		
		67,111				100		
		68,111				320		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
		69,77				835		
		69,78				740		
		70,73				223		
		71,91				100		
		72,74				390		
		73,75				140		
		74,76				304		
		75,77				120		
		75,78				403		
		75,79				21		
		76,80				246		
		76,81				160		
		76,82				1019		
		77,83				34		
		78,84				120		
		79,85				68		
		80,86				910		
		80,91				302		
		81,87				778		
		81,91				101		
		82,111				1310		
		83,91				20		
		84,88				278		
		84,89				81		
		84,91				290		
		85,111				100		
		86,111				372		
		87,90				72		
		88,105				28		
		89,105				90		
		90,111				250		
		91,92				144		
		91,93				303		
		91,94				220		
		92,95				58		
		93,95				224		
		94,95				211		
		95,96				99		
		95,97				44		
		95,98				120		
		95,99				70		
		95,100				421		
		95,104				231		
		96,101				214		
		97,102				196		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
		98,103				280		
		99,111				398		
		100,111				72		
		101,105				280		
		102,106				356		
		102,107				193		
		103,107				140		
		103,108				130		
		104,111				300		
		105,111				456		
		106,109				7		
		107,111				170		
		108,110				252		
		109,111				210		
		110,111				308		
		-1,-1				308		
						121		
						52		
						426		
						104		
						1386		
						527		
						968		
						1047		
						538		
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						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		
						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		

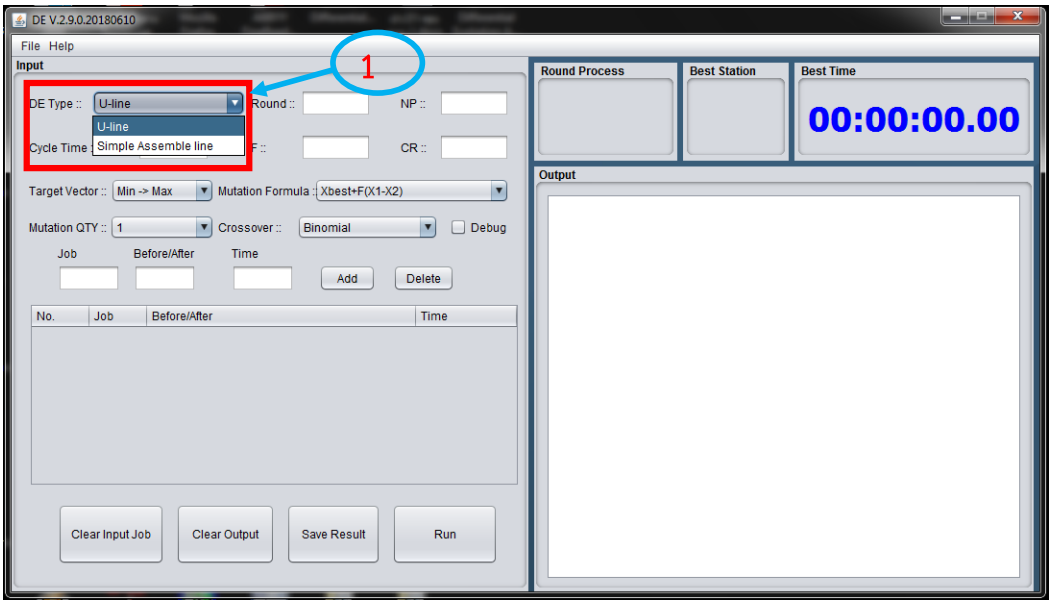
ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						221,223		
						222,224		
						223,225		
						224,226		
						225,227		
						226,228		
						227,230		
						228,231		
						229,235		
						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		
						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		

ตารางที่ ข.1 ข้อมูล Precedence Diagram จำนวนงานและเวลางานย่อยที่ใช้ในการทดลองของ 8 ชุดโจทย์ปัญหาที่มี 21-297 ขั้นงาน (ต่อ)

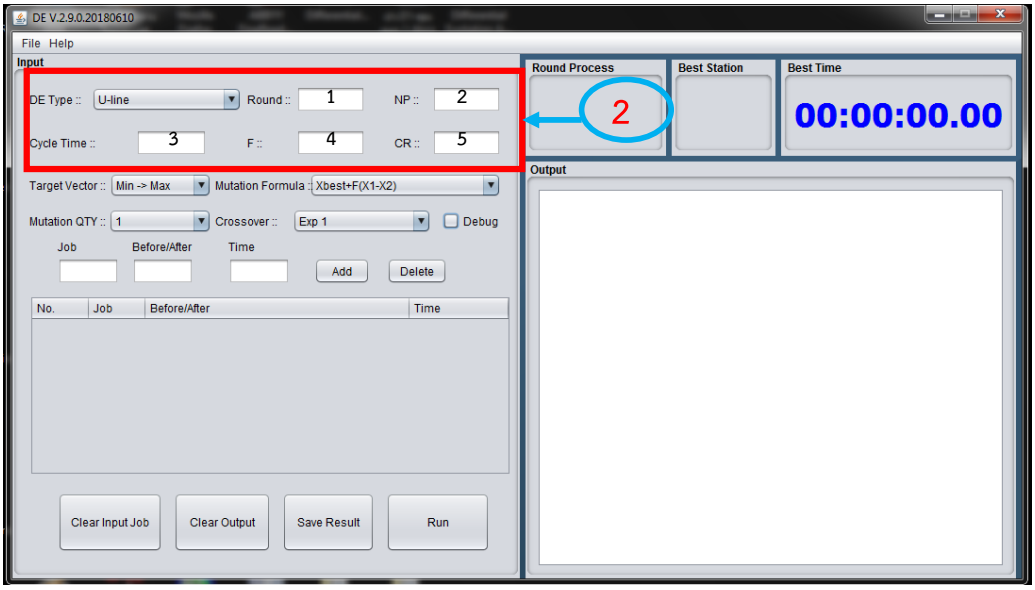
Item	ARC83	ARC111	HESKIA	KILBRID	SAWYER	SCHOLL	MITCHE	TONGE
						256,263		
						257,264		
						258,265		
						259,266		
						260,267		
						261,268		
						261,269		
						262,269		
						263,270		
						264,271		
						265,272		
						266,271		
						267,273		
						268,274		
						269,274		
						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		
						-1,-1		

ภาคผนวก ค
อธิบายขั้นตอนการกรอกข้อมูลและการแสดงผลของโปรแกรม
กับปัญหาการจัดสมดุลสายการประกอบแบบตัวยู



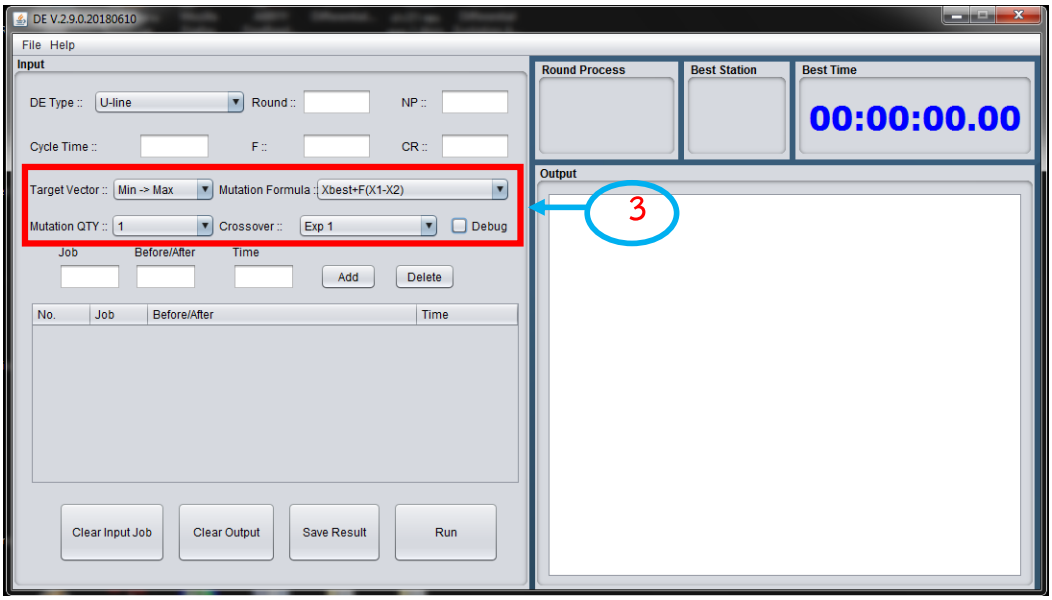
ภาพที่ ค.1 ขั้นตอนที่หนึ่งการเลือกรูปแบบของปัญหาการจัดสมดุลสายการประกอบแบบตัวยู

จากภาพที่ ค.1 ขั้นตอนที่หนึ่งเลือกรูปแบบของปัญหาการจัดสมดุลสายการประกอบแบบตัวยู โดยเลือกช่อง DE Type >>> U-line



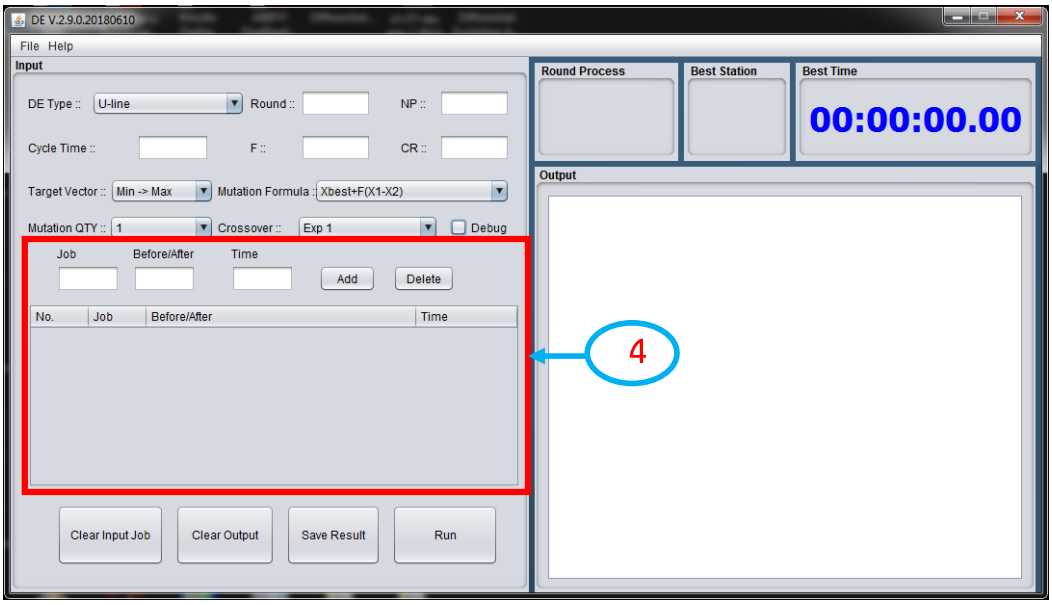
ภาพที่ ค.2 ขั้นตอนที่สองการกรอกพารามิเตอร์ 5 ช่อง ที่เกี่ยวข้องกับการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่าง

จากภาพที่ ค.2 ขั้นตอนที่สองการกรอกพารามิเตอร์ 5 ช่อง ที่เกี่ยวข้องกับการคำนวณโดยใช้วิธีวิวัฒนาการโดยใช้ผลต่าง 1) ช่อง Round >>> เพื่อกรอกจำนวนรอบการรัน 2) ช่อง NP >>> เพื่อกรอกจำนวนขั้นงาน 3) ช่อง Cycle time >>> เพื่อกรอกรอบเวลาการผลิตของปัญหา 4) ช่อง F >>> เพื่อกรอกแฟคเตอร์ในการขยายผลต่าง (Scaling factor) 5) ช่อง CR >>> เพื่อกรอกอัตราการแลกเปลี่ยนค่าในพิกัด (Crossover rate)

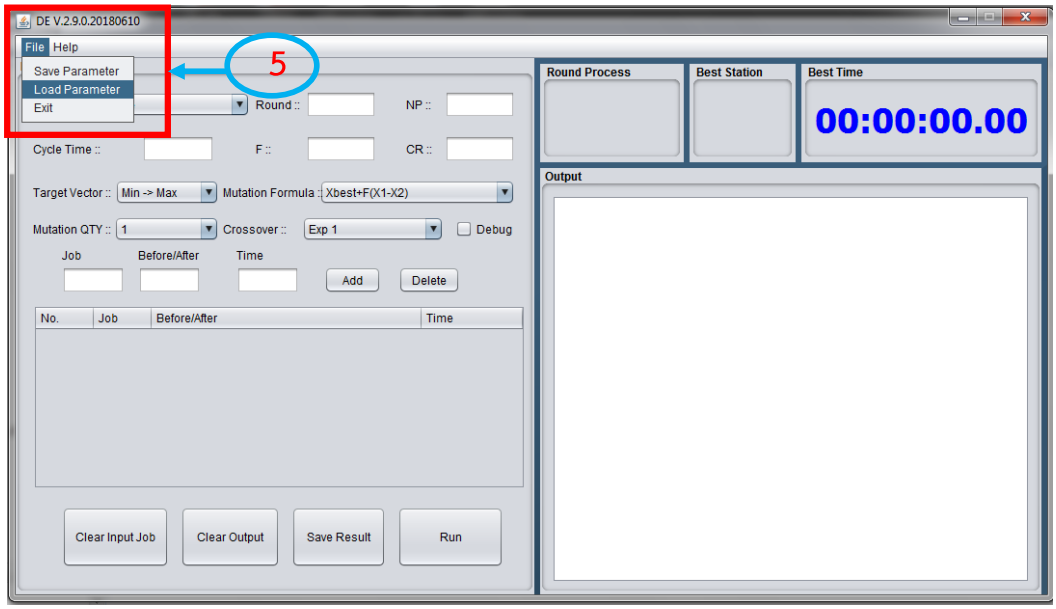


ภาพที่ ค.3 ขั้นตอนที่สามกรอกสูตรการคำนวณวิธีวัฒนาการโดยใช้ผลต่าง

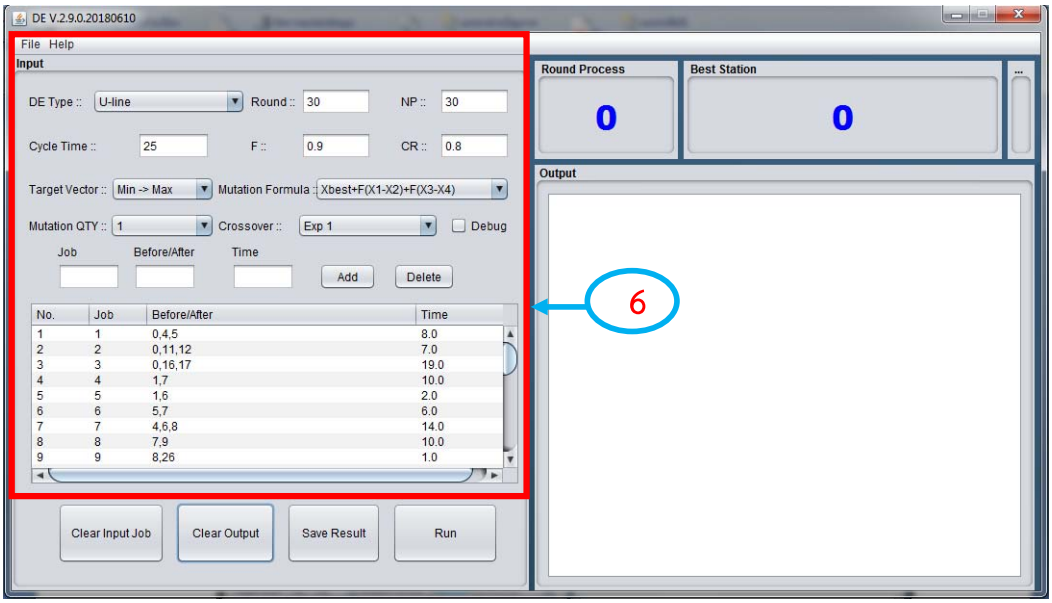
จากภาพที่ ค.3 ขั้นตอนที่สามกรอกสูตรการคำนวณวิธีวัฒนาการโดยใช้ผลต่างโดยทำการกรอง คลิ๊กตัวเลือกจากซ้ายไปขวาบนลงล่าง ช่อง Target Vector >>> เลือกจากค่าที่ Random จากน้อยไปมาก (Min-->Max) หรือ มากไปน้อย (Max-->Min) ช่อง Mutation Formula >>> ทำการเลือกวิธีแบบต่าง ๆ ในส่วนขั้นตอน Mutation โดยมีสูตรให้เลือกแต่ละวิธี, ช่อง Mutation QTY ทำการเลือกตามจำนวนรอบคลิ๊ก (1 รอบ และ NP) ช่อง Crossover ทำการเลือกวิธีแบบต่าง ๆ ในส่วนขั้นตอน Crossover โดยจะมีสูตรให้เลือกแต่ละวิธี



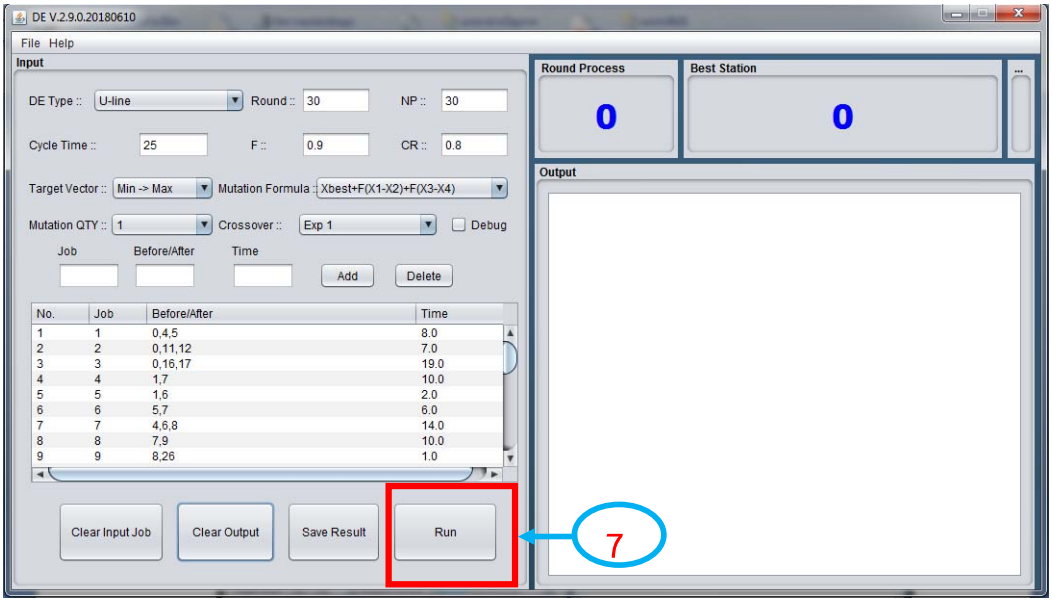
ภาพที่ ค.4 ขั้นตอนที่สี่กรอกข้อมูลปัญหาการจัดสมดุลสายการประกอบ



ภาพที่ ค.5 ขั้นตอนที่ห้าโหลดที่ทำการบันทึกไว้เรียกข้อมูลปัญหาที่ต้องการจะรันมาคำนวณ

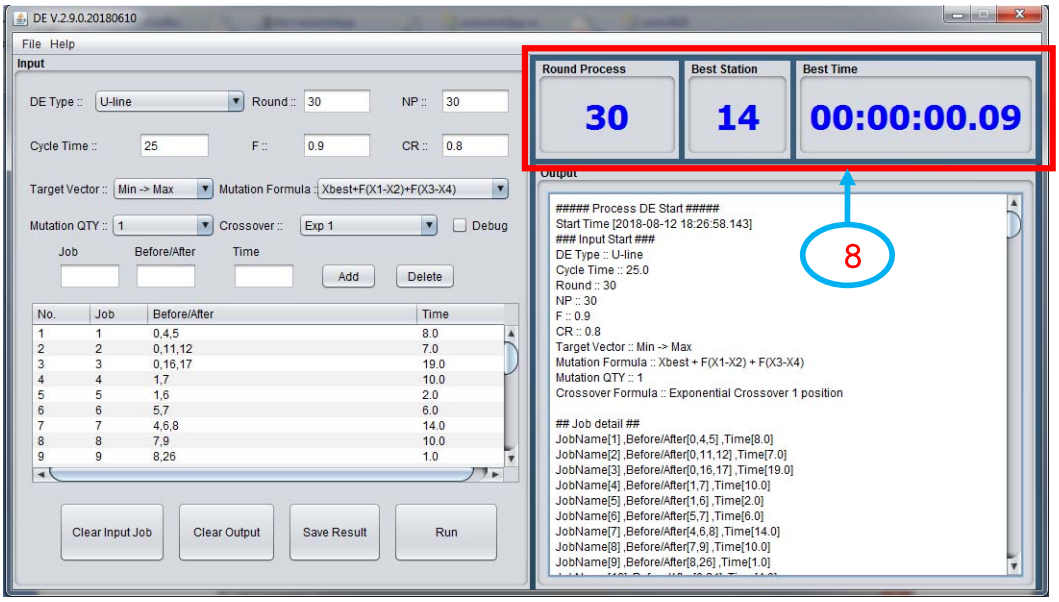


ภาพที่ ค.6 ขั้นตอนที่หกเรียกที่ทำการบันทึกไว้เรียกข้อมูล ทำการโหลดปัญหา Sawyer Cycle time 14 ทดสอบการจัดสมดุลสายการประกอบแบบตัวยูมาคำนวณ



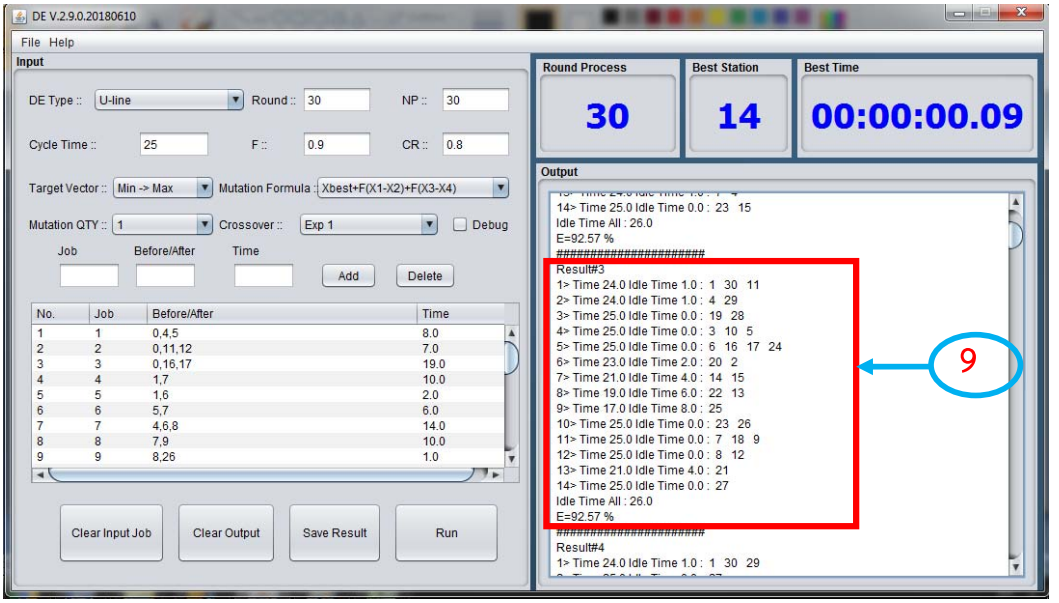
ภาพที่ ค.7 ขั้นตอนที่เจ็ดเมื่อกรอกข้อมูลครบ ทำการโหลดปัญหา Kilbrige & Wester Cycle time 79 ทดสอบและทำการรันผลของโปรแกรม

จากภาพที่ ค.7 ขั้นตอนที่เจ็ดเมื่อกรอกข้อมูลครบ ทำการโหลดปัญหา Kilbrige & Wester Cycle time 79 ทดสอบและทำการรันผลของโปรแกรมโดยคลิ๊ก Run



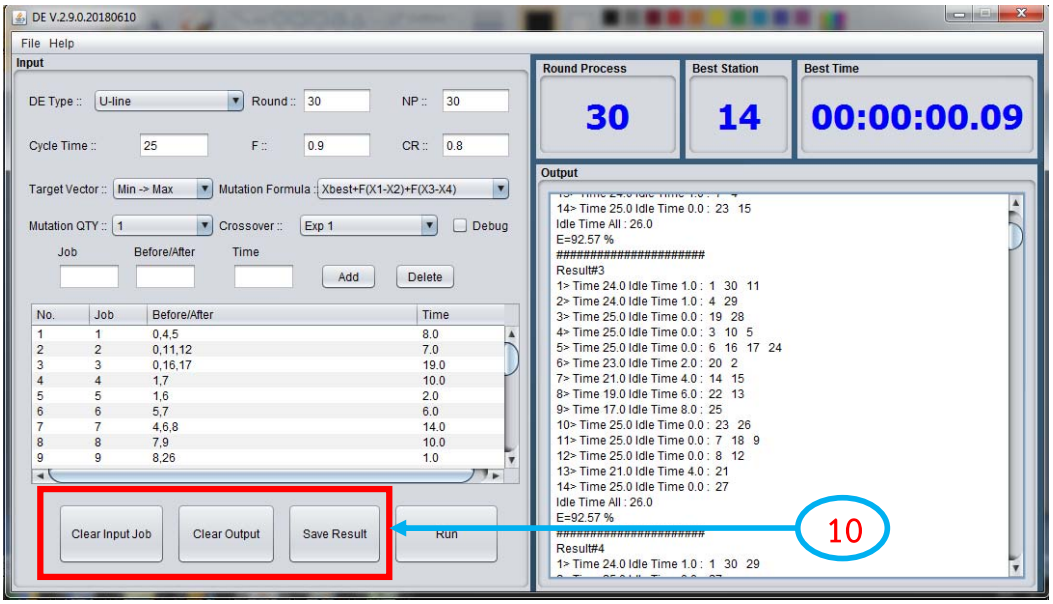
ภาพที่ ค.8 ขั้นตอนที่แปดแสดงผลการคำนวณ

จากภาพที่ ค.8 ได้ค่า Best Station = 14 สถานีงาน Best Time = 0.09 วินาที



ภาพที่ ค.9 ขั้นตอนที่เก้า แสดงผลการจัดสมดุลสายการประกอบแบบตัวยู

จากภาพที่ ค.9 แต่ละสถานีงานประกอบไปด้วยกี่งาน สถานีงานมีงานอะไรบ้าง และประสิทธิภาพของสายการประกอบแบบตัวยู และเวลาว่างงาน



ภาพที่ ค.10 ขั้นตอนที่สิบ แสดงหน้าต่าง การ Clear input job, Clear Output, Save Result

จากภาพที่ ค.10 คลิ๊ก Clear input job เพื่อที่จะนำข้อมูลนำเข้าออกจากหน้าต่าง คลิ๊ก Clear Output เพื่อที่จะทำการแสดงผลลัพธ์ออกจากหน้าต่าง คลิ๊ก Save Result เพื่อที่จะทำการบันทึกผลการทดลอง

ภาคผนวก ง
ผลการเปรียบเทียบวิธีการที่พัฒนากับวิธีการอื่น ๆ ในปัญหาการจัดสมดุลสายการประกอบแบบตัวยู

ตารางที่ ง.1 ผลของวิธีวิวัฒนาการโดยใช้ผลต่างที่นำเสนอเทียบกับ วิธี GA (Hwang et al., 2008) ในตัวอย่างปัญหาแบบตัวยู (UALBP-1) ที่มีขนาดกลาง (21-45 ขั้นงาน)

Problems	Size	Cycle time	IP[1] solution	GA[2]				DE1[3]			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
Mitchell	21	14	8	8	0	0.06	93.75	8	0	0.05	93.75
		15	8	8	0	0.14	87.75	8	0	0.03	87.75
		21	5	5	0	0.01	100.00	5	0	0.01	100.00
Heskiaoff	28	138	8	8	0	0.11	92.75	8	0	0.09	92.75
		205	5	5	0	0.01	99.99	5	0	0.07	99.99
		324	4	4	0	0.07	79.01	4	0	0.09	79.01
Sawyer	30	27	13	13	0	0.02	92.31	13	0	0.12	92.31
		33	10	10	0	0.01	98.18	10	0	0.02	98.18
		54	6	6	0	0.04	100.00	6	0	0.09	100.00
Kilbridge & Wester	45	79	7	7	0	0.09	99.82	7	0	0.06	99.82
		92	6	7	0	0.08	100.00	7	0	0.04	100.00
		184	3	3	0	0.12	100.00	3	0	0.05	100.00
Tonge	70	176	21	21	0	0.23	95.00	21	0	0.05	95.00
		364	10	10	0	0.56	96.40	10	0	0.03	96.40
		468	8	8	0	0.04	93.80	8	0	0.01	93.80
Arcus 1	83	5853	13	13	0	0.03	92.40	13	0	0.09	92.40
		6842	12	12	0	0.05	92.20	12	0	0.07	92.20
		8412	10	10	0	0.12	90.00	10	0	0.09	90.00
		10816	8	8	0	0.23	87.50	8	0	0.08	87.50
Arcus 2	111	5755	27	27	0	0.03	96.80	27	0	0.09	96.80
		10027	16	16	0	0.07	93.70	16	0	0.03	93.70
		10743	15	15	0	0.06	93.30	15	0	0.07	93.30
		17067	9	9	0	0.01	97.90	9	0	0.06	97.90
Total optimal solution (or lower bound) found from 23 instances problem				23	0 /inst.	0.06 /inst.	95.30 /inst.	23	0.94 /inst.	0.06 /inst.	95.30 /inst.

[1] Integer Programming solution, result from Urban (1998) (in table1, p.740)

[2] Genetic Algorithm , researched by Hwang et al (2008)

[3] DE1 = DE/best/2 to Exponential 1 position

กำหนดให้ค่า F = 0.9 Juni Ronkkonen,et al (2005), CR = 0.8, N = 30, Round = 30

*m is the number of station, % is the average relative deviation from the best know solution,

E = Efficiency of Balance"

ตารางที่ ง.2 ผลของวิธีวิวัฒนาการโดยใช้ผลต่างที่นำเสนอเทียบกับ MMAS no local search (นุชสรา เกรียงกรกฎ, 2550) ในตัวอย่างปัญหาแบบตัวยู (UALBP-1) ที่มีขนาดกลาง (21-45 ขั้นงาน)

Problems	Size	Cycle time	IP[1] solution	MMAS[2]				DE1[3]				DE2[4]				DE3[5]				DE4[6]				DE5[7]				DE6[8]			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
Mitchell	21	14	8	8	0	1.50	93.75	8	0	0.05	93.75	8	0	0.06	93.75	8	0	0.05	93.75	8	0	0.14	93.75	8	0	0.13	93.75	8	0	0.23	93.75
		15	8	8	0	1.63	87.75	8	0	0.03	87.75	8	0	0.04	87.75	8	0	0.03	87.75	8	0	0.16	87.75	8	0	0.17	87.75	8	0	0.27	87.75
		21	5	5	0	5.43	100.00	5	0	0.01	100.00	5	0	0.03	100.00	5	0	0.01	100.00	5	0	0.25	100.00	5	0	0.45	100.00	5	0	0.45	100.00
Heskiaoff	28	114	9	10	11.11	30.00	89.82	9	11.11	0.04	89.82	9	11.11	0.05	89.82	9	11.11	0.04	89.82	9	11.11	0.06	89.82	9	11.11	0.78	89.82	9	11.11	0.78	89.82
		128	8	9	12.50	30.00	88.89	8	12.50	0.08	88.89	8	12.50	0.10	88.89	8	12.50	0.23	88.89	8	12.50	1.98	88.89	8	12.50	0.98	88.89	8	12.50	0.98	88.89
		138	8	8	0	1.71	92.75	8	0	0.09	92.75	8	0	0.28	92.75	8	0	0.12	92.75	8	0	0.24	92.75	8	0	1.06	92.75	8	0	1.06	92.75
		205	5	5	0	1.68	99.99	5	0	0.07	99.99	5	0	0.08	99.99	5	0	0.05	99.99	5	0	0.43	99.99	5	0	0.98	99.99	5	0	0.98	99.99
		216	5	5	0	2.36	94.81	5	0	0.13	94.81	5	0	0.08	94.81	5	0	0.05	94.81	5	0	0.54	94.81	5	0	2.23	94.81	5	0	2.23	94.81
		256	4	4	0	5.52	100.00	4	0	0.16	100.00	4	0	0.04	100.00	4	0	0.39	100.00	4	0	0.47	100.00	4	0	2.06	100.00	4	0	2.06	100.00
		324	4	4	0	2.35	79.01	4	0	0.09	79.01	4	0	0.06	79.01	4	0	0.69	79.01	4	0	0.98	79.01	4	0	3.11	79.01	4	0	3.11	79.01
		342	3	3	0	2.45	99.81	3	0	0.12	99.81	3	0	0.03	99.81	3	0	0.06	99.81	3	0	0.78	99.81	3	0	2.68	99.81	3	0	2.68	99.81

[1] Integer Programming solution, result from Urban (1998) (in table1, p.740)

[2] MMAS (no local search) , researched by Nuchsara,et al (2007a,2007b)

[3] DE1 = DE/best/2 to Exponential 1 position

[4] DE2 = DE/best/2 to Exponential 2 position

[5] DE3 = DE/rand/1 to Exponential 2 position

[6] DE4 = DE/rand/1to Exponential 1 position

[7] DE5 = DE/rand-to-best/1 to Exponential Binomial

[8] DE6 = DE/rand/1 tol Binomial

กำหนดให้ค่า F = 0.9 Ronkkonen,et al (2005), CR = 0.8, N = 30, Round = 30

m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance"

ตารางที่ ง.3 ผลของวิธีวิวัฒนาการโดยใช้ผลต่างที่นำเสนอเทียบกับ MMAS no local search (นุชสรา เกรียงกรกฎ, 2550) ในตัวอย่างปัญหาแบบตัวยู (UALBP-1) ที่มีขนาดกลาง (21-45 ขั้นงาน)

Problems	Size	Cycle time	IP[1] solution	MMAS[2]				DE1[3]				DE2[4]				DE3[5]				DE4[6]				DE5[7]				DE6[8]			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
Sawyer	30	25	14	14	0	1.22	92.57	14	0	0.07	92.57	14	0	0.08	92.57	14	0	0.11	92.57	14	0	0.67	92.57	14	0	1.87	92.57	14	0	1.87	92.57
		27	13	13	0	0.96	92.31	13	0	0.12	92.31	13	0	0.09	92.31	13	0	0.13	92.31	13	0	0.44	92.31	13	0	0.98	92.31	13	0	1.11	92.31
		30	11	11	0	7.48	98.18	11	0	0.08	98.18	11	0	0.10	98.18	11	0	0.42	98.18	11	0	0.83	98.18	11	0	0.77	98.18	11	0	1.23	98.18
		33	10	10	0	20.15	98.18	10	0	0.18	98.18	10	0	0.11	98.18	10	0	0.38	98.18	10	0	0.67	98.18	10	0	0.44	98.18	10	0	1.11	98.18
		36	10	10	0	2.08	90.00	10	0	0.05	90.00	10	0	0.15	90.00	10	0	0.07	90.00	10	0	0.55	90.00	10	0	0.34	90.00	10	0	1.34	90.00
		41	8	8	0	14.67	98.78	8	0	0.04	98.78	8	0	0.19	98.78	8	0	0.54	98.78	8	0	0.59	98.78	8	0	0.50	98.78	8	0	1.96	98.78
		54	6	6	0	5.56	100.00	6	0	0.09	100.00	6	0	0.38	100.00	6	0	0.17	100.00	6	0	0.78	100.00	6	0	0.61	100.00	6	0	1.61	100.00
		75	5	5	0	2.97	86.84	5	0	0.06	86.84	5	0	0.35	86.84	5	0	0.14	86.84	5	0	0.53	86.84	5	0	0.57	86.84	5	0	1.94	86.84
Kilbridge & Wester	45	57	10	10	0	1.33	96.84	10	0	0.11	96.84	10	0	0.22	96.84	10	0	0.07	96.84	10	0	0.23	96.84	10	0	0.99	96.84	10	0	0.99	96.84
		79	7	8	14.28	30.00	99.82	7	0	0.12	99.82	7	0	0.45	99.82	7	0	0.33	99.82	7	0	0.19	99.82	7	0	0.46	99.82	7	0	0.46	99.82
		92	6	7	16.67	30.00	100.00	6	0	0.14	100.00	6	0	0.3	100.00	6	0	0.23	100.00	6	0	0.14	100.00	6	0	0.31	100.00	6	0	0.31	100.00
		110	6	6	0	1.48	83.64	6	0	0.10	83.64	6	0	0.06	83.64	6	0	0.44	83.64	6	0	0.11	83.64	6	0	0.89	83.64	6	0	1.93	83.64
		138	4	4	0	4.08	100.00	4	0	0.14	100.00	4	0	0.07	100.00	4	0	0.35	100.00	4	0	0.23	100.00	4	0	0.64	100.00	4	0	1.64	100.00
		184	3	3	0	6.73	100.00	3	0	0.15	100.00	3	0	0.09	100.00	3	0	0.93	100.00	3	0	0.22	100.00	3	0	0.55	100.00	3	0	1.55	100.00
Total optimal solution (or lower bound) found from 25 instances problem				23	2.18 /inst.	9.19 /inst.	95.51 /inst.	25	0.94 /inst.	0.10 /inst.	95.51 /inst.	25	0.94 /inst.	0.19 /inst.	95.51 /inst.	25	0.00 /inst.	0.31 /inst.	95.51 /inst.	25	0.00 /inst.	0.44 /inst.	95.51 /inst.	25	0.94 /inst.	0.71 /inst.	95.51 /inst.	25	0.94 /inst.	1.36 /inst.	95.51 /inst.

[1] Integer Programming solution, result from Urban (1998) (in table1, p.740)

[2] MMAS (no local search) , researched by Nuchsara,et al (2007a, 2007b)

[3] DE1 = DE/best/2 to Exponential 1 position

[4] DE2 = DE/best/2 to Exponential 2 position

[5] DE3 = DE/rand/1 to Exponential 2 position

[6] DE4 = DE/rand/1to Exponential 1 position

[7] DE5 = DE/rand-to-best/1 to Exponential Binomial

[8] DE6 = DE/rand/1 tol Binomial

กำหนดให้ค่า F = 0.9 Ronkkonen,et al (2005), CR = 0.8, N = 30, Round = 30, m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance"

ตารางที่ ง.4 ผลของวิธีวิวัฒนาการโดยใช้ผลต่างที่นำเสนอเทียบกับ MMAS no local search (นุชสรา เกรียงกรกฎ, 2550) ในตัวอย่างปัญหาแบบตัวยู (UALBP-1) ที่มีขนาดใหญ่ (75-297 ขั้นงาน)

Problems	Size	Cycle time	0ptimal solution[1]	MMAS[2]				DE1[3]				DE2[4]				DE3[5]				DE4[6]				DE5[7]				DE6[8]			
			*m**	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*
Wee-mag 21 inst.	75	**28**	**63**	63	0	1.19	93.75	63	0	0.12	93.75	63	0	0.14	93.75	63	0	0.23	93.75	63	0	0.98	93.75	63	0	0.78	93.75	63	0	0.78	93.75
		29	**63**	63	0	1.19	87.75	63	0	0.14	87.75	63	0	0.16	87.75	63	0	0.17	87.75	63	0	2.23	87.75	63	0	0.98	87.75	63	0	0.98	87.75
		30	**62**	62	0	1.23	100.00	62	0	0.07	100.00	62	0	0.25	100.00	62	0	0.45	100.00	62	0	2.06	100.00	62	0	1.06	100.00	62	0	1.06	100.00
		31	**62**	62	0	1.14	89.82	62	0	0.05	89.82	62	0	0.06	89.82	62	0	0.78	89.82	62	0	3.11	89.82	62	0	0.98	89.82	62	0	0.98	89.82
		32	**61**	61	0	1.16	88.89	61	0	0.10	88.89	61	0	1.98	88.89	61	0	0.98	88.89	61	0	2.68	88.89	61	0	2.23	88.89	61	0	2.23	88.89
		33	**61**	61	0	1.22	92.75	61	0	0.28	92.75	61	0	0.24	92.75	61	0	1.06	92.75	61	0	1.87	92.75	61	0	2.06	92.75	61	0	2.06	92.75
		34	**61**	61	0	1.22	99.99	61	0	0.08	99.99	61	0	0.43	99.99	61	0	0.98	99.99	61	0	0.98	99.99	61	0	3.11	99.99	61	0	3.11	99.99
		35	**60**	60	0	1.22	94.81	60	0	0.12	94.81	60	0	0.54	94.81	60	0	2.23	94.81	60	0	2.23	94.81	60	0	2.23	94.81	60	0	2.23	94.81
		36	**60**	60	0	1.25	100.00	60	0	0.04	100.00	60	0	0.47	100.00	60	0	2.06	100.00	60	0	2.06	100.00	60	0	2.06	100.00	60	0	2.06	100.00
		37	**60**	60	0	1.23	79.01	60	0	0.06	79.01	60	0	0.98	79.01	60	0	3.11	79.01	60	0	3.11	79.01	60	0	3.11	79.01	60	0	3.11	79.01
		38	**60**	60	0	1.19	99.81	60	0	0.22	99.81	60	0	0.16	99.81	60	0	2.68	99.81	60	0	2.68	99.81	60	0	2.68	99.81	60	0	2.68	99.81
		39	**60**	60	0	1.14	92.57	60	0	0.08	92.57	60	0	0.25	92.57	60	0	1.87	92.57	60	0	1.87	92.57	60	0	1.87	92.57	60	0	1.87	92.57
		40	**60**	60	0	1.25	92.31	60	0	0.19	92.31	60	0	0.06	92.31	60	0	0.98	92.31	60	0	0.98	92.31	60	0	0.98	92.31	60	0	0.98	92.31
		41	**59**	59	0	1.19	98.18	59	0	0.10	98.18	59	0	1.98	98.18	59	0	0.77	98.18	59	0	0.77	98.18	59	0	0.77	98.18	59	0	0.77	98.18
		42	**55**	55	0	1.16	98.18	55	0	0.11	98.18	55	0	0.24	98.18	55	0	0.44	98.18	55	0	0.44	98.18	55	0	0.44	98.18	55	0	0.44	98.18
		43	**50**	50	0	1.30	90.00	50	0	0.15	90.00	50	0	0.55	90.00	50	0	0.34	90.00	50	0	0.34	90.00	50	0	0.34	90.00	50	0	0.34	90.00
		45	**38**	38	0	4.31	98.78	38	0	0.19	98.78	38	0	0.54	98.78	38	0	0.50	98.78	38	0	0.50	98.78	38	0	0.50	98.78	38	0	0.50	98.78
		46	**34**	34	0	3.59	100.00	34	0	0.38	100.00	34	0	0.47	100.00	34	0	0.61	100.00	34	0	0.61	100.00	34	0	0.61	100.00	34	0	0.61	100.00
		52	**31**	31	0	2.59	86.84	31	0	0.35	86.84	31	0	0.98	86.84	31	0	0.57	86.84	31	0	0.57	86.84	31	0	0.57	86.84	31	0	0.87	86.84
		54	**31**	31	0	1.17	96.84	31	0	0.22	96.84	31	0	0.23	96.84	31	0	0.99	96.84	31	0	0.99	96.84	31	0	0.99	96.84	31	0	0.99	96.84
		56	**30**	30	0	1.55	99.82	30	0	0.45	99.82	30	0	0.19	99.82	30	0	0.46	99.82	30	0	0.46	99.82	30	0	0.46	99.82	30	0	0.46	99.82

[1] m* is the optimal solution

[2] MMAS (no local search), researched by Nuchsara,et al (2007a, 2007b)

[3] DE1 = DE/best/2 to Exponential 1 position

[4] DE2 = DE/best/2 to Exponential 2 position

[5] DE3 = DE/rand to Exponential 2 position

[6] DE4 = DE/rand to to Exponential 1 position

[7] DE5 = DE/rand-to-best to Exponential Binomial

[8] DE6 = DE/rand/1 to Binomial

กำหนดให้ค่า F = 0.9 Ronkkonen et al (2005), CR = 0.8, N = 30, Round = 30, m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance

ตารางที่ ง.5 ผลของวิธีวิวัฒนาการโดยใช้ผลต่างที่นำเสนอเทียบกับ MMAS no local search (นุชสรา เกรียงกรกฎ, 2550) ในตัวอย่างปัญหาแบบตัวยู (UALBP-1) ที่มีขนาดใหญ่ (75-297 ขั้นงาน)

Problems	Size	Cycle time	0ptimal solution[1]	MMAS[2]				DE1[3]				DE2[4]				DE3[5]				DE4[6]				DE5[7]				DE6[8]			
			m*	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E	m	%	cal. Time (sec.)	E
Arcus 1 16.inst.	83	3786	21	21	0	1.03	100.00	21	0	0.3	100.00	21	0	0.14	100.00	21	0	0.31	100.00	21	0	0.31	100.00	21	0	0.31	100.00	21	0	0.91	100.00
		3985	20	20	0	1.00	83.64	20	0	0.06	83.64	20	0	0.11	83.64	20	0	0.89	83.64	20	0	0.89	83.64	20	0	0.89	83.64	20	0	0.89	83.64
		4206	19	19	0	1.08	89.82	19	0	0.07	89.82	19	0	0.23	89.82	19	0	0.64	89.82	19	0	0.64	89.82	19	0	0.64	89.82	19	0	0.64	89.82
		4454	18	18	0	1.03	88.89	18	0	0.10	88.89	18	0	0.98	88.89	18	0	0.06	88.89	18	0	0.55	88.89	18	0	0.55	88.89	18	0	0.85	88.89
		4732	17	17	0	1.02	92.75	17	0	0.28	92.75	17	0	0.16	92.75	17	0	1.98	92.75	17	0	0.98	92.75	17	0	0.46	92.75	17	0	0.46	92.75
		5048	16	16	0	1.05	99.99	16	0	0.08	99.99	16	0	0.25	99.99	16	0	0.24	99.99	16	0	0.77	99.99	16	0	0.31	99.99	16	0	0.31	99.99
		5408	15	15	0	1.08	94.81	15	0	0.12	94.81	15	0	0.06	94.81	15	0	0.55	94.81	15	0	0.44	94.81	15	0	0.89	94.81	15	0	0.89	94.81
		5824	14	14	0	1.10	100.00	14	0	0.04	100.00	14	0	1.98	100.00	14	0	0.54	100.00	14	0	0.34	100.00	14	0	0.64	100.00	14	0	0.64	100.00
		5853	13	14	7.69	1.05	79.01	13	0	0.06	89.63	13	0	0.24	89.63	13	0	0.47	89.63	13	0	0.50	89.63	13	0	0.06	89.63	13	0	0.06	89.63
		6309	13	13	0	1.07	99.81	13	0	0.09	99.81	13	0	0.55	99.81	13	0	0.98	99.81	13	0	0.61	99.81	13	0	1.98	99.81	13	0	1.98	99.81
		6842	12	12	0	1.03	92.57	12	0	0.19	92.57	12	0	0.54	92.57	12	0	1.98	92.57	12	0	0.57	92.57	12	0	0.24	92.57	12	0	0.64	92.57
		6883	12	12	0	1.03	92.31	12	0	0.38	92.31	12	0	0.16	92.31	12	0	0.24	92.31	12	0	0.99	92.31	12	0	0.59	92.31	12	0	0.79	92.31
		7571	11	11	0	1.04	98.18	11	0	0.35	98.18	11	0	0.25	98.18	11	0	0.55	98.18	11	0	0.44	98.18	11	0	0.54	98.18	11	0	0.74	98.18
		8412	10	10	0	1.05	98.18	10	0	0.22	98.18	10	0	0.06	98.18	10	0	0.54	98.18	10	0	0.34	98.18	10	0	0.47	98.18	10	0	1.97	98.18
		8898	9	9	0	1.03	90.00	9	0	0.45	90.00	9	0	1.66	90.00	9	0	0.16	90.00	9	0	0.50	90.00	9	0	0.51	90.00	9	0	1.89	90.00
		10816	7	8	14.29	1.04	98.78	8	14.29	0.71	98.78	8	14.29	0.78	98.78	8	14.29	0.55	98.78	8	14.29	0.61	98.78	8	14.29	0.98	98.78	8	14.29	1.98	98.78

[1] m* is the optimal solution

[2] MMAS (no local search) , researched by Nuchsara et al (2007b,2007c)

[3] DE1 = DE/best/2 to Exponential 1 position

[4] DE2 = DE/best/2 to Exponential 2 position

[5] DE3 = DE/rand to Exponential 2 position

[6] DE4 = DE/rand to to Exponential 1 position

[7] DE5 = DE/rand-to-best to Exponential Binomial

[8] DE6 = DE/rand/1 to Binomial

กำหนดให้ค่า F = 0.9 Ronkkonen et al (2005), CR = 0.8, N = 30, Round = 30

m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance

ตารางที่ ง.6 ผลของวิธีวิวัฒนาการโดยใช้ผลต่างที่นำเสนอเทียบกับ MMAS no local search (นุชสรา เกรียงกรกฎ, 2550) ในตัวอย่างปัญหาแบบตัวยู (UALBP-1) ที่มีขนาดใหญ่ (75-297 ขั้นงาน)

Problems	Size	Cycle time	0ptimal solution[1]	MMAS[2]				DE1[3]				DE2[4]				DE3[5]				DE4[6]				DE5[7]				DE6[8]			
			*m**	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*	*m*	*%*	*cal. Time (sec.)*	*E*
Scholl 25 inst.	297	1394	50	51	2.00	12.08	100.00	51	2.00	12.28	100.00	51	2.00	12.30	100.00	51	2.00	11.97	100.00	51	2.00	11.92	100.00	51	2.00	12.22	100.00	51	2.00	12.22	100.00
		1452	48	49	2.08	16.10	86.84	49	2.08	11.08	86.84	49	2.08	12.10	86.84	49	2.08	12.28	86.84	49	2.08	11.63	86.84	49	2.08	12.30	86.84	49	2.08	12.30	86.84
		1483	47	48	2.13	11.30	96.84	48	2.13	11.30	96.84	48	2.13	11.55	96.84	48	2.13	11.08	96.84	48	2.13	11.91	96.84	48	2.13	12.10	96.84	48	2.13	12.10	96.84
		1515	46	47	2.17	11.55	99.82	47	2.17	11.55	99.82	47	2.17	12.03	99.82	47	2.17	11.96	99.82	47	2.17	11.97	99.82	47	2.17	11.45	99.82	47	2.17	11.45	99.82
		1548	45	46	2.22	11.92	89.82	46	2.22	11.92	89.82	46	2.22	12.85	89.82	46	2.22	11.45	89.82	46	2.22	12.28	89.82	46	2.22	12.22	89.82	46	2.22	12.22	89.82
		1584	44	45	2.27	11.63	88.89	45	2.27	11.63	88.89	45	2.27	11.55	88.89	45	2.27	12.22	88.89	45	2.27	11.97	88.89	45	2.27	12.30	88.89	45	2.27	12.30	88.89
		1620	43	44	2.33	11.91	92.75	44	2.33	11.91	92.75	44	2.33	11.92	92.75	44	2.33	12.30	92.75	44	2.33	12.28	92.75	44	2.33	12.10	92.75	44	2.33	12.10	92.75
		1659	42	43	2.38	11.97	99.99	43	2.38	11.97	99.99	43	2.38	11.63	99.99	43	2.38	12.10	99.99	43	2.38	11.08	99.99	43	2.38	12.55	99.99	43	2.38	12.55	99.99
		1699	41	42	2.44	12.28	94.81	42	2.44	12.28	94.81	42	2.44	11.91	94.81	42	2.44	11.45	94.81	42	2.44	11.96	94.81	42	2.44	12.30	94.81	42	2.44	12.30	94.81
		1742	40	41	2.50	11.08	100.00	41	2.50	11.08	100.00	41	2.50	11.97	100.00	41	2.50	12.22	100.00	41	2.50	11.45	100.00	41	2.50	11.97	100.00	41	2.50	11.97	100.00
		1787	39	40	2.56	11.96	79.01	40	2.56	11.96	79.01	40	2.56	12.28	79.01	40	2.56	12.30	79.01	40	2.56	12.22	79.01	40	2.56	12.28	79.01	40	2.56	12.28	79.01
		1834	38	39	2.63	11.45	99.81	39	2.63	11.55	99.81	39	2.63	11.97	99.81	39	2.63	12.10	99.81	39	2.63	12.30	99.81	39	2.63	12.08	99.81	39	2.63	12.08	99.81
		1883	37	38	2.70	12.22	92.57	38	2.70	12.30	92.57	38	2.70	12.28	92.57	38	2.70	11.55	92.57	38	2.70	12.10	92.57	38	2.70	11.96	92.57	38	2.70	11.96	92.57
		1935	36	37	2.77	12.30	92.31	37	2.77	12.10	92.31	37	2.77	11.08	92.31	37	2.77	12.30	92.31	37	2.77	12.10	92.31	37	2.77	11.45	92.31	37	2.77	11.45	92.31
		1991	35	36	2.86	12.10	98.18	36	2.86	11.55	98.18	36	2.86	11.96	98.18	36	2.86	11.97	98.18	36	2.86	11.96	98.18	36	2.86	13.97	98.18	36	2.86	13.97	98.18
		2049	34	35	2.94	11.55	98.18	35	2.94	12.03	98.18	35	2.94	11.45	98.18	35	2.94	12.28	98.18	35	2.94	11.55	98.18	35	2.94	12.28	98.18	35	2.94	12.28	98.18
		2111	33	34	3.03	12.30	90.00	34	3.03	12.85	90.00	34	3.03	12.22	90.00	34	3.03	11.08	90.00	34	3.03	12.30	90.00	34	3.03	11.08	90.00	34	3.03	11.08	90.00
		2177	32	33	3.13	12.10	98.78	33	3.13	11.55	98.78	33	3.13	12.30	98.78	33	3.13	11.96	98.78	33	3.13	12.10	98.78	33	3.13	11.96	98.78	33	3.13	11.96	98.78
		2247	31	32	3.23	11.55	100.00	32	3.23	11.92	100.00	32	3.23	12.10	100.00	32	3.23	11.45	100.00	32	3.23	11.55	100.00	32	3.23	13.45	100.00	32	3.23	13.45	100.00
		2322	30	31	3.33	12.03	86.84	31	3.33	11.63	86.84	31	3.33	12.10	86.84	31	3.33	12.22	86.84	31	3.33	12.03	86.84	31	3.33	12.22	86.84	31	3.33	12.22	86.84
		2402	29	30	3.45	12.85	96.84	30	3.45	11.91	96.84	30	3.45	11.55	96.84	30	3.45	12.30	96.84	30	3.45	12.85	96.84	30	3.45	12.30	96.84	30	3.45	12.30	96.84
		2488	28	29	3.57	12.84	99.82	29	3.57	11.97	99.82	29	3.57	12.30	99.82	29	3.57	12.10	99.82	29	3.57	11.91	99.82	29	3.57	11.55	99.82	29	3.57	11.55	99.82
		2580	27	28	3.70	12.81	92.57	28	3.7	12.28	92.57	28	3.7	12.10	92.57	28	3.7	11.97	92.57	28	3.7	11.97	92.57	28	3.7	13.92	92.57	28	3.7	13.92	92.57
		2680	26	27	3.85	11.77	92.31	27	3.85	11.08	92.31	27	3.85	11.55	92.31	27	3.85	12.28	92.31	27	3.85	12.28	92.31	27	3.85	11.63	92.31	27	3.85	11.63	92.31
		2787	25	26	4.00	12.63	98.18	26	4.00	11.99	98.18	26	4.00	12.03	98.18	26	4.00	11.08	98.18	26	4.00	11.08	98.18	26	4.00	11.91	98.18	26	4.00	11.91	98.18
Total optimal solution found from 62 instances problem				35	1.49 /inst.	5.72 /inst.	94.61 /inst.	36	2.81 /inst.	11.83 /inst.	94.61 /inst.	36	2.81 /inst.	11.96 /inst.	94.61 /inst.	36	2.81 /inst.	11.92 /inst.	94.61 /inst.	36	2.81 /inst.	11.95 /inst.	94.61 /inst.	36	2.81 /inst.	12.22 /inst.	94.61 /inst.	36	2.81 /inst.	12.22 /inst.	94.61 /inst.

[1] m* is the optimal solution

[2] MMAS (no local search) , researched by Nuchsara et al (2007b,2007c)

[3] DE1 = DE/best/2 to Exponential 1 position

[4] DE2 = DE/best/2 to Exponential 2 position

[5] DE3 = DE/rand to Exponential 2 position

[6] DE4 = DE/rand to to Exponential 1 position

[7] DE5 = DE/rand-to-best to Exponential Binomial

[8] DE6 = DE/rand/1 to Binomial

กำหนดให้ค่า F = 0.9 Ronkkonen et al (2005), CR = 0.8, N = 30, Round = 30

m is the number of station, % is the average relative deviation from the best know solution, E = Efficiency of Balance

ภาคผนวก จ
จำนวนสถานีงานของแต่ละปัญหา ปัญหาขนาดกลางและปัญหาขนาดใหญ่

1. Problem Mitchell Size 21

1.1 Problem Mitchell Size 21 (Cycle Time = 14)

Result

1> Time 14.0 Idle Time 0.0 : 1 21 20

2> Time 14.0 Idle Time 0.0 : 4 3

3> Time 14.0 Idle Time 0.0 : 19 5 2

4> Time 13.0 Idle Time 1.0 : 18 15 11

5> Time 14.0 Idle Time 0.0 : 16 12 13 9

6> Time 11.0 Idle Time 3.0 : 8 10 14

7> Time 12.0 Idle Time 2.0 : 7 6

8> Time 13.0 Idle Time 1.0 : 17

Idle Time All : 7.0 E=93.75 %

1.2 Problem Mitchell Size 21 (Cycle Time = 15)

Result

1> Time 14.0 Idle Time 1.0 : 1 21 20

2> Time 15.0 Idle Time 0.0 : 17 19

3> Time 14.0 Idle Time 1.0 : 18 16 15 12

4> Time 13.0 Idle Time 2.0 : 10 11 3

5> Time 13.0 Idle Time 2.0 : 14 9 13

6> Time 15.0 Idle Time 0.0 : 7 8

7> Time 13.0 Idle Time 2.0 : 6 5

8> Time 8.0 Idle Time 7.0 : 4 2

Idle Time All : 15.0 E=87.5 %

1.3 Problem Mitchell Size 21 (Cycle Time = 21)

Result

1> Time 21.0 Idle Time 0.0 : 1 2 19 21 18

2> Time 21.0 Idle Time 0.0 : 15 3 14 12 16

3> Time 21.0 Idle Time 0.0 : 11 7 10 5

4> Time 21.0 Idle Time 0.0 : 8 6 4 9

5> Time 21.0 Idle Time 0.0 : 17 13 20

Idle Time All : 0.0 E=100.0 %

2. Problem Heskiaoff Size 28

2.1 Problem Heskiaoff Size 28 (Cycle Time = 114)

Result

1> Time 110.0 Idle Time 4.0 : 1 21 4 5 26 23 22 27

2> Time 114.0 Idle Time 0.0 : 28 24 16 18 15

3> Time 114.0 Idle Time 0.0 : 8 14

4> Time 108.0 Idle Time 6.0 : 13
5> Time 101.0 Idle Time 13.0 : 9 10 7
6> Time 100.0 Idle Time 14.0 : 6 12 11 3 17
7> Time 107.0 Idle Time 7.0 : 25
8> Time 106.0 Idle Time 8.0 : 2 19
9> Time 67.0 Idle Time 47.0 : 20
Idle Time All : 99.0 E=84.82 %

2.2 Problem Heskiaoff Size 28 (Cycle Time = 128)
Result
1> Time 126.0 Idle Time 2.0 : 1 19 26 23 5 27
2> Time 123.0 Idle Time 5.0 : 20 3 21 4 2
3> Time 117.0 Idle Time 11.0 : 28 18 22 24 16
4> Time 74.0 Idle Time 54.0 : 14 15 7 10
5> Time 110.0 Idle Time 18.0 : 11 6 8
6> Time 128.0 Idle Time 0.0 : 9 17
7> Time 108.0 Idle Time 20.0 : 13
8> Time 126.0 Idle Time 2.0 : 12 25
Idle Time All : 112.0 E=88.89 %

2.3 Problem Heskiaoff Size 28 (Cycle Time = 138)
Result
1> Time 136.0 Idle Time 2.0 : 1 2 5 23 26
2> Time 137.0 Idle Time 1.0 : 6 28 21 27 18 15 4
3> Time 133.0 Idle Time 5.0 : 20 16 3 7 22
4> Time 118.0 Idle Time 20.0 : 17 11
5> Time 130.0 Idle Time 8.0 : 8 19 24
6> Time 136.0 Idle Time 2.0 : 9 10 14
7> Time 126.0 Idle Time 12.0 : 12 25
2.4 Problem Heskiaoff Size 28 (Cycle Time = 138) (ต่อ)
Result
8> Time 108.0 Idle Time 30.0 : 13
Idle Time All : 80.0 E=92.75 %

2.5 Problem Heskiaoff Size 28 (Cycle Time = 205)
Result
1> Time 205.0 Idle Time 0.0 : 1 3 22 28 24 5
2> Time 205.0 Idle Time 0.0 : 25 4 8 27 21 26 15 23
3> Time 205.0 Idle Time 0.0 : 14 19 2 18 9 16
4> Time 204.0 Idle Time 1.0 : 7 10 12 6 11 20

5> Time 205.0 Idle Time 0.0 : 13 17
Idle Time All : 1.0 E=99.9 %

2.6 Problem Heskiaoff Size 28 (Cycle Time = 216)
Result
1> Time 211.0 Idle Time 5.0 : 1 3 22 26 23 19 4 5 24 21 27
2> Time 193.0 Idle Time 23.0 : 25 2 6
3> Time 202.0 Idle Time 14.0 : 17 7 28 16 18
4> Time 205.0 Idle Time 11.0 : 14 15 13 12 11
5> Time 213.0 Idle Time 3.0 : 10 20 9 8
Idle Time All : 56.0 E=94.81 %

2.7 Problem Heskiaoff Size 28 (Cycle Time = 256)
Result
1> Time 256.0 Idle Time 0.0 : 1 28 21 15 2 3
2> Time 256.0 Idle Time 0.0 : 11 25 8 14 27 16 23 5
3> Time 256.0 Idle Time 0.0 : 19 13 9 26 10 22 4
4> Time 256.0 Idle Time 0.0 : 12 18 17 6 20 24 7
Idle Time All : 0.0 E=100.0 %

2.8 Problem Heskiaoff Size 28 (Cycle Time = 324)
Result
1> Time 322.0 Idle Time 2.0 : 1 5 19 3 22 28 24 2 4 27 26
2> Time 316.0 Idle Time 8.0 : 17 25 20 18 15 11 23 16
3> Time 247.0 Idle Time 77.0 : 7 10 6 14 8 12 21
4> Time 139.0 Idle Time 185.0 : 13 9
Idle Time All : 272.0 E=79.01 %

2.9 Problem Heskiaoff Size 28 (Cycle Time = 342)
Result
1> Time 341.0 Idle Time 1.0 : 1 26 28 24 19 16 22 17 23 4 5 15
2> Time 342.0 Idle Time 0.0 : 11 8 10 2 9 13 18
3> Time 341.0 Idle Time 1.0 : 12 7 6 14 21 20 25 27 3
Idle Time All : 2.0 E=99.81 %

3. **Problem Sawyer Size 30**
3.1 Problem Sawyer Size 30 (Cycle Time = 25)
Result
1> Time 25.0 Idle Time 0.0 : 1 5 6 30 2
2> Time 24.0 Idle Time 1.0 : 7 4

3> Time 25.0 Idle Time 0.0 : 29 8 9
4> Time 22.0 Idle Time 3.0 : 19 10
5> Time 24.0 Idle Time 1.0 : 18 11
6> Time 25.0 Idle Time 0.0 : 27
7> Time 25.0 Idle Time 0.0 : 23 26
8> Time 25.0 Idle Time 0.0 : 22 15 17
9> Time 24.0 Idle Time 1.0 : 14 13 24
10> Time 23.0 Idle Time 2.0 : 20 28
11> Time 25.0 Idle Time 0.0 : 16 12
12> Time 21.0 Idle Time 4.0 : 21
13> Time 17.0 Idle Time 8.0 : 25
14> Time 19.0 Idle Time 6.0 : 3
Idle Time All : 26.0 E=92.57 %

3.2 Problem Sawyer Size 30 (Cycle Time = 27)

Result

1> Time 27.0 Idle Time 0.0 : 1 2 4 30
2> Time 26.0 Idle Time 1.0 : 7 6 10 5
3> Time 27.0 Idle Time 0.0 : 12 13 24
4> Time 26.0 Idle Time 1.0 : 20 16
5> Time 26.0 Idle Time 1.0 : 29 14
6> Time 25.0 Idle Time 2.0 : 27
7> Time 26.0 Idle Time 1.0 : 23 26 9
8> Time 23.0 Idle Time 4.0 : 22 15
9> Time 27.0 Idle Time 0.0 : 25 8
10> Time 26.0 Idle Time 1.0 : 3 28
11> Time 26.0 Idle Time 1.0 : 17 18 11
12> Time 21.0 Idle Time 6.0 : 21
13> Time 18.0 Idle Time 9.0 : 19
Idle Time All : 27.0 E=92.31 %

3.3 Problem Sawyer Size 30 (Cycle Time = 30)

Result

1> Time 30.0 Idle Time 0.0 : 1 19 30 5
2> Time 30.0 Idle Time 0.0 : 6 2 4 28
3> Time 29.0 Idle Time 1.0 : 3 16
4> Time 30.0 Idle Time 0.0 : 11 20
5> Time 30.0 Idle Time 0.0 : 14 24 15 17
6> Time 27.0 Idle Time 3.0 : 25 18
7> Time 28.0 Idle Time 2.0 : 22 7
8> Time 30.0 Idle Time 0.0 : 23 8 10

9> Time 30.0 Idle Time 0.0 : 9 12 29
10> Time 30.0 Idle Time 0.0 : 21 26
11> Time 30.0 Idle Time 0.0 : 13 27
Idle Time All : 6.0 E=98.18 %

3.4 Problem Sawyer Size 30 (Cycle Time = 33)

Result

1> Time 33.0 Idle Time 0.0 : 1 11 2 30 5
2> Time 33.0 Idle Time 0.0 : 29 3
3> Time 33.0 Idle Time 0.0 : 6 12 16 17
4> Time 33.0 Idle Time 0.0 : 20 18 28
5> Time 33.0 Idle Time 0.0 : 19 4 13
6> Time 29.0 Idle Time 4.0 : 27 10
7> Time 33.0 Idle Time 0.0 : 23 26 9 24
8> Time 33.0 Idle Time 0.0 : 8 22 15
9> Time 33.0 Idle Time 0.0 : 21 14
10> Time 31.0 Idle Time 2.0 : 7 25
Idle Time All : 6.0 E=98.18 %

3.5 Problem Sawyer Size 30 (Cycle Time = 36)

Result

1> Time 35.0 Idle Time 1.0 : 1 30 10 2 28 24
2> Time 35.0 Idle Time 1.0 : 20 3
3> Time 35.0 Idle Time 1.0 : 21 14 5
4> Time 36.0 Idle Time 0.0 : 15 29 6 17 13
5> Time 34.0 Idle Time 2.0 : 7 8 18
6> Time 32.0 Idle Time 4.0 : 9 25 22
7> Time 35.0 Idle Time 1.0 : 23 26 4
8> Time 35.0 Idle Time 1.0 : 27 16
9> Time 32.0 Idle Time 4.0 : 19 11
10> Time 15.0 Idle Time 21.0 : 12
Idle Time All : 36.0 E=90.0 %

3.6 Problem Sawyer Size 30 (Cycle Time = 41)

Result

1> Time 41.0 Idle Time 0.0 : 1 5 19 30 28 10
2> Time 41.0 Idle Time 0.0 : 24 25 18 2
3> Time 40.0 Idle Time 1.0 : 26 9 29 20
4> Time 41.0 Idle Time 0.0 : 16 21 8
5> Time 40.0 Idle Time 1.0 : 7 22 4 17

3.6 Problem Sawyer Size 30 (Cycle Time = 41) (ต่อ)

Result

6> Time 41.0 Idle Time 0.0 : 23 3 6

7> Time 41.0 Idle Time 0.0 : 12 14 11

8> Time 39.0 Idle Time 2.0 : 13 15 27

Idle Time All : 4.0 E=98.78 %

3.7 Problem Sawyer Size 30 (Cycle Time = 54)

Result

1> Time 54.0 Idle Time 0.0 : 1 4 10 7 28 2 5 30

2> Time 54.0 Idle Time 0.0 : 29 6 27 26

3> Time 54.0 Idle Time 0.0 : 25 9 23 12 13

4> Time 54.0 Idle Time 0.0 : 22 8 14 19

5> Time 54.0 Idle Time 0.0 : 20 18 21 24

6> Time 54.0 Idle Time 0.0 : 16 17 15 3 11

Idle Time All : 0.0 E=100.0 %

3.8 Problem Sawyer Size 30 (Cycle Time = 75)

Result

1> Time 75.0 Idle Time 0.0 : 1 11 3 5 2 10 17 28 16 30

2> Time 75.0 Idle Time 0.0 : 27 29 18 6 12 13

3> Time 74.0 Idle Time 1.0 : 7 14 8 20 21 9

4> Time 73.0 Idle Time 2.0 : 15 23 22 19 24 26

5> Time 27.0 Idle Time 48.0 : 25 4

Idle Time All : 51.0 E=86.4 %

4. Problem Kilbridge&Wester Size 45

4.1 Problem Kilbridge&Wester Size 45 (Cycle Time = 57)

Result

1> Time 55.0 Idle Time 2.0 : 1 11 39 2 45 13 12

2> Time 55.0 Idle Time 2.0 : 37 41 4 9

3> Time 54.0 Idle Time 3.0 : 29 5 8 10

4> Time 57.0 Idle Time 0.0 : 30 6 7 42 43 32

5> Time 54.0 Idle Time 3.0 : 14 31 3 40 15

4.1 Problem Kilbridge&Wester Size 45 (Cycle Time = 57)

Result

6> Time 56.0 Idle Time 1.0 : 23 25 18

7> Time 56.0 Idle Time 1.0 : 16 19 38 20 17 35

8> Time 55.0 Idle Time 2.0 : 27 28 22 34 36

9> Time 55.0 Idle Time 2.0 : 24 26 33 44

10> Time 55.0 Idle Time 2.0 : 21
Idle Time All : 18.0 E=96.84 %

4.2 Problem Kilbridge&Wester Size 45 (Cycle Time = 79)
Result
1> Time 79.0 Idle Time 0.0 : 1 3 5 11 13 45 42 43 37
2> Time 79.0 Idle Time 0.0 : 15 24 41 18 12 32
3> Time 78.0 Idle Time 1.0 : 31 33 29 40 35 38 9 39 2
4> Time 79.0 Idle Time 0.0 : 8 23 4 14 19
5> Time 79.0 Idle Time 0.0 : 17 25 20 36 6 30 26
6> Time 79.0 Idle Time 0.0 : 21 22 44 27
7> Time 79.0 Idle Time 0.0 : 7 10 28 34 16
Idle Time All : 1.0 E=99.82 %

4.3 Problem Kilbridge&Wester Size 45 (Cycle Time = 92)
Result
1> Time 92.0 Idle Time 0.0 : 1 12 3 44 45 43 13 7 11 5
2> Time 92.0 Idle Time 0.0 : 14 30 42 8 31 41 29 32 40
3> Time 92.0 Idle Time 0.0 : 38 9 25 35 17 26 36 39
4> Time 92.0 Idle Time 0.0 : 28 33 24 2 37 15
5> Time 92.0 Idle Time 0.0 : 4 22 21 19 18 34
6> Time 92.0 Idle Time 0.0 : 20 23 6 16 10 27
Idle Time All : 0.0 E=100.0 %

4.4 Problem Kilbridge&Wester Size 45 (Cycle Time = 110)
Result
1> Time 108.0 Idle Time 2.0 : 1 44 43 39 2 11 42 3 8 7 12 45
2> Time 108.0 Idle Time 2.0 : 41 13 4 29 10 30 5 40 6 37
3> Time 107.0 Idle Time 3.0 : 9 31 14 35 17 25 38 26
4.4 Problem Kilbridge&Wester Size 45 (Cycle Time = 110) (ต่อ)
Result
4> Time 110.0 Idle Time 0.0 : 28 27 22 21 34 36
5> Time 71.0 Idle Time 39.0 : 20 19 18 33 23 15 32
6> Time 48.0 Idle Time 62.0 : 16 24
Idle Time All : 108.0 E=83.64 %

4.5 Problem Kilbridge&Wester Size 45 (Cycle Time = 138)
Result
1> Time 138.0 Idle Time 0.0 : 1 7 9 41 40 10 31 3 8 30 29 35 45
2> Time 138.0 Idle Time 0.0 : 38 2 42 28 34 5 43 44 33 6 39 36 27 32

3> Time 138.0 Idle Time 0.0 : 23 4 15 26 37 19 22 24 25 20
4> Time 138.0 Idle Time 0.0 : 18 21 16 13 17 11 14 12
Idle Time All : 0.0 E=100.0 %

4.6 Problem Kilbridge&Wester Size 45 (Cycle Time = 184)
Result
1> Time 184.0 Idle Time 0.0 : 1 44 11 13 45 42 43 2 39 3 14 29 32 37 25 30 12 31 7 15
2> Time 184.0 Idle Time 0.0 : 5 17 4 41 27 24 33 18 26 16 9 23
3> Time 184.0 Idle Time 0.0 : 10 40 19 36 34 6 20 21 28 35 8 22 38
Idle Time All : 0.0 E=100.0 %

5. Problem Arcus 1 83

5.1 Problem Arcus 1 83 (Cycle Time = 3786)
Result
1> Time 3631.0 Idle Time 155.0 : 1 2 4
2> Time 3691.0 Idle Time 95.0 : 83
3> Time 3716.0 Idle Time 70.0 : 5 81 79 82
4> Time 3659.0 Idle Time 127.0 : 8 78 43 40
5> Time 3748.0 Idle Time 38.0 : 37 35 16 33 77
6> Time 3767.0 Idle Time 19.0 : 80 39 76
7> Time 3785.0 Idle Time 1.0 : 36 31 75 44 12
8> Time 3412.0 Idle Time 374.0 : 46 68 48 50 73
9> Time 3426.0 Idle Time 360.0 : 51 72 52 71 67
10> Time 3526.0 Idle Time 260.0 : 66 69 65 27 15 64
11> Time 3634.0 Idle Time 152.0 : 20 49 22 26 63 13
12> Time 3664.0 Idle Time 122.0 : 62 10 3
13> Time 3589.0 Idle Time 197.0 : 25 7 61
14> Time 3765.0 Idle Time 21.0 : 60 59 17 70 32
15> Time 3706.0 Idle Time 80.0 : 58 28 18 53
16> Time 3527.0 Idle Time 259.0 : 57 56 54 9
17> Time 3527.0 Idle Time 259.0 : 55 74 21
18> Time 3335.0 Idle Time 451.0 : 23 30 24 29
19> Time 3561.0 Idle Time 225.0 : 19 38 41 45 47
20> Time 3581.0 Idle Time 205.0 : 6 11
21> Time 3655.0 Idle Time 131.0 : 34 42 14
Idle Time All : 3601.0 E=95.47 %

5.2 Problem Arcus 1 83 (Cycle Time = 3985)
Result

1> Time 3631.0 Idle Time 354.0 : 1 2 4
2> Time 3691.0 Idle Time 294.0 : 83
3> Time 3331.0 Idle Time 654.0 : 6 81
4> Time 3809.0 Idle Time 176.0 : 9 10 13 79
5> Time 3900.0 Idle Time 85.0 : 14 80 18
6> Time 3983.0 Idle Time 2.0 : 23 25 3 19
7> Time 3917.0 Idle Time 68.0 : 77 24 16 28
8> Time 3820.0 Idle Time 165.0 : 29 43 39 30
9> Time 3904.0 Idle Time 81.0 : 40 35 15 42 20
10> Time 3938.0 Idle Time 47.0 : 37 33 36 38
11> Time 3821.0 Idle Time 164.0 : 11 74 44
12> Time 3883.0 Idle Time 102.0 : 68 75 73 67 71 69
13> Time 3977.0 Idle Time 8.0 : 46 70 49 47 27 66
14> Time 3961.0 Idle Time 24.0 : 65 21 22 45 64 12 41
15> Time 3791.0 Idle Time 194.0 : 63 62 5 61 76
16> Time 3880.0 Idle Time 105.0 : 60 72 78 32
17> Time 3690.0 Idle Time 295.0 : 59 58 34 82
18> Time 3789.0 Idle Time 196.0 : 57 8 31 17 48
19> Time 3965.0 Idle Time 20.0 : 56 50 51 7
20> Time 3224.0 Idle Time 761.0 : 55 52 54 53 26
Idle Time All : 3795.0 E=95.24 %

5.3 Problem Arcus 1 83 (Cycle Time = 4206)

Result

1> Time 3631.0 Idle Time 575.0 : 1 2 4
2> Time 4141.0 Idle Time 65.0 : 83 81
3> Time 4007.0 Idle Time 199.0 : 7 11 79 15 20
4> Time 3821.0 Idle Time 385.0 : 80 77 40 76
5> Time 4037.0 Idle Time 169.0 : 43 35 37 39
6> Time 4023.0 Idle Time 183.0 : 31 30 36 8
7> Time 3955.0 Idle Time 251.0 : 29 38 13 26 44
8> Time 4206.0 Idle Time 0.0 : 46 10 14 18 48
9> Time 4088.0 Idle Time 118.0 : 19 23 50 17 25 16
10> Time 4178.0 Idle Time 28.0 : 51 21 22 28 12
11> Time 3841.0 Idle Time 365.0 : 27 9 82
12> Time 4079.0 Idle Time 127.0 : 3 75 33 24
13> Time 4019.0 Idle Time 187.0 : 73 68 72 69 52
14> Time 4159.0 Idle Time 47.0 : 53 71 70 67 49 54 66
15> Time 4006.0 Idle Time 200.0 : 55 65 47 64 56 63 41
16> Time 4028.0 Idle Time 178.0 : 45 62 57 5 58
17> Time 3740.0 Idle Time 466.0 : 59 61 60 78 32

18> Time 3831.0 Idle Time 375.0 : 74 42
19> Time 4115.0 Idle Time 91.0 : 6 34
Idle Time All : 4009.0 E=94.98 %

5.4 Problem Arcus 1 83 (Cycle Time = 4454)

Result

1> Time 4358.0 Idle Time 96.0 : 1 2 5
2> Time 4141.0 Idle Time 313.0 : 83 81
3> Time 3994.0 Idle Time 460.0 : 79 78 34 4 76
4> Time 4015.0 Idle Time 439.0 : 35 32 75 36
5> Time 4290.0 Idle Time 164.0 : 33 25 6
6> Time 4444.0 Idle Time 10.0 : 37 43 10 16 40 12
7> Time 3964.0 Idle Time 490.0 : 14 19 7
8> Time 4357.0 Idle Time 97.0 : 24 73 71 39 77
9> Time 4379.0 Idle Time 75.0 : 69 70 44 46 48 29 15
10> Time 4394.0 Idle Time 60.0 : 50 30 31 42 20 51
11> Time 4323.0 Idle Time 131.0 : 52 26 9 8
12> Time 4339.0 Idle Time 115.0 : 82 53 28 54
13> Time 4349.0 Idle Time 105.0 : 17 22 55 27 72
14> Time 4162.0 Idle Time 292.0 : 21 56 80 68 13
15> Time 4257.0 Idle Time 197.0 : 57 58 59 67 60 18 11
16> Time 4221.0 Idle Time 233.0 : 61 23 66 62 63 64 49
17> Time 4287.0 Idle Time 167.0 : 65 47 3 38
18> Time 3631.0 Idle Time 823.0 : 45 41 74
Idle Time All : 4267.0 E=94.68 %

5.5 Problem Arcus 1 83 (Cycle Time = 4732)

Result

1> Time 4494.0 Idle Time 238.0 : 1 2 3
2> Time 4664.0 Idle Time 68.0 : 4 83
3> Time 4116.0 Idle Time 616.0 : 82 80 79 81
4> Time 4397.0 Idle Time 335.0 : 77 34 16 76 32 25
5> Time 4445.0 Idle Time 287.0 : 10 35 42 75
6> Time 4693.0 Idle Time 39.0 : 73 14 72 78
7> Time 4599.0 Idle Time 133.0 : 19 71 69 33 36 40 68
8> Time 4427.0 Idle Time 305.0 : 38 67 27 37 49 41
9> Time 4666.0 Idle Time 66.0 : 45 66 21 47 39 65 44
10> Time 4369.0 Idle Time 363.0 : 64 46 63 17 31 62 12
11> Time 4667.0 Idle Time 65.0 : 61 30 29 9 60
12> Time 4240.0 Idle Time 492.0 : 59 58 24 22 70 13
13> Time 4440.0 Idle Time 292.0 : 18 20 23 8 15 48 57

5.5 Problem Arcus 1 83 (Cycle Time = 4732) (ต่อ)

Result

14> Time 4138.0 Idle Time 594.0 : 50 11 26 56 51 43

15> Time 4615.0 Idle Time 117.0 : 52 55 53 6

16> Time 4692.0 Idle Time 40.0 : 54 7 28

17> Time 4243.0 Idle Time 489.0 : 74 5

Idle Time All : 4539.0 E=94.36 %

5.6 Problem Arcus 1 83 (Cycle Time = 5048)

Result

1> Time 4358.0 Idle Time 690.0 : 1 2 5

2> Time 4731.0 Idle Time 317.0 : 8 83

3> Time 5030.0 Idle Time 18.0 : 81 77 82 78 43 76 79

4> Time 4960.0 Idle Time 88.0 : 35 80 34 16

5> Time 4829.0 Idle Time 219.0 : 42 32 40 12 25 37

6> Time 5029.0 Idle Time 19.0 : 36 38 28 33

7> Time 4881.0 Idle Time 167.0 : 41 17 13 22 24 39

8> Time 4666.0 Idle Time 382.0 : 20 44 46 21 18 26 29 48

9> Time 4613.0 Idle Time 435.0 : 23 50 51 52 19 27 75

10> Time 4685.0 Idle Time 363.0 : 53 73 54 14 72

11> Time 4757.0 Idle Time 291.0 : 55 71 56 57 58 69 30 68

12> Time 4606.0 Idle Time 442.0 : 49 67 47 59 66 4 31

13> Time 4615.0 Idle Time 433.0 : 65 64 6 60

14> Time 4852.0 Idle Time 196.0 : 63 7 62 61 70

15> Time 4493.0 Idle Time 555.0 : 11 74 10

16> Time 4800.0 Idle Time 248.0 : 9 45 3 15

Idle Time All : 4863.0 E=93.98 %

5.7 Problem Arcus 1 83 (Cycle Time = 5408)

Result

1> Time 5364.0 Idle Time 44.0 : 1 83

2> Time 4774.0 Idle Time 634.0 : 81 79 2 4 80

3> Time 5156.0 Idle Time 252.0 : 78 40 37 77 3

4> Time 5219.0 Idle Time 189.0 : 5 7 42

5> Time 5176.0 Idle Time 232.0 : 11 39 15 33 76 44 20 43

6> Time 5397.0 Idle Time 11.0 : 46 36 32 25 13 75 18

7> Time 5300.0 Idle Time 108.0 : 10 73 72 70 23

8> Time 5323.0 Idle Time 85.0 : 14 74 71 38

9> Time 5131.0 Idle Time 277.0 : 41 6 48 68 19

10> Time 5363.0 Idle Time 45.0 : 24 67 66 29 30 45 50 69

11> Time 5302.0 Idle Time 106.0 : 51 47 27 52 17 21 65 26

12> Time 5092.0 Idle Time 316.0 : 53 22 64 34 54 31 49
13> Time 5315.0 Idle Time 93.0 : 55 63 9 28 16
14> Time 4959.0 Idle Time 449.0 : 62 61 60 12 35 8 56
15> Time 3034.0 Idle Time 2374.0 : 59 58 57 82
Idle Time All : 5215.0 E=93.57 %

5.8 Problem Arcus 1 83 (Cycle Time = 5824)

Result

1> Time 5814.0 Idle Time 10.0 : 1 83 81
2> Time 5439.0 Idle Time 385.0 : 82 79 77 40 34 37 16
3> Time 5687.0 Idle Time 137.0 : 42 78 76 43 32 12 25
4> Time 5729.0 Idle Time 95.0 : 75 9 74 33
5> Time 5671.0 Idle Time 153.0 : 35 26 8 36 39
6> Time 5648.0 Idle Time 176.0 : 38 31 30 15 29 44 68 20
7> Time 5561.0 Idle Time 263.0 : 24 19 46 11 14 5
8> Time 5516.0 Idle Time 308.0 : 48 50 51 7 4 67
9> Time 5345.0 Idle Time 479.0 : 52 53 23 66 10 41 73
10> Time 5732.0 Idle Time 92.0 : 45 65 54 55 71 56 18 69 72
11> Time 5730.0 Idle Time 94.0 : 47 27 70 21 22 64 49 57 13
12> Time 4755.0 Idle Time 1069.0 : 58 59 17 60 61 63 2
13> Time 5295.0 Idle Time 529.0 : 62 3 6
14> Time 3983.0 Idle Time 1841.0 : 80 28
Idle Time All : 5631.0 E=93.09 %

5.9 Problem Arcus 1 83 (Cycle Time = 5853)

Result

1> Time 5814.0 Idle Time 39.0 : 1 83 81
2> Time 5360.0 Idle Time 493.0 : 2 4 3 79 82
3> Time 5495.0 Idle Time 358.0 : 80 7 78
4> Time 5593.0 Idle Time 260.0 : 40 42 37 76 75 73 33
5> Time 5538.0 Idle Time 315.0 : 74 23 8 43 71
6> Time 5814.0 Idle Time 39.0 : 32 28 39 72 16 10
7> Time 5746.0 Idle Time 107.0 : 12 44 31 30 36 35 68
8> Time 5753.0 Idle Time 100.0 : 46 29 70 26 48 34 24
9> Time 5742.0 Idle Time 111.0 : 19 50 20 67 17 51 38 15 69 77
10> Time 5415.0 Idle Time 438.0 : 13 41 22 21 52 27 66 49 11
11> Time 5815.0 Idle Time 38.0 : 65 47 64 63 62 6 57
12> Time 5363.0 Idle Time 490.0 : 61 25 53 45 5 14 56
13> Time 5350.0 Idle Time 503.0 : 60 54 55 59 58 9 18
Idle Time All : 4037.0 E=95.63 %

5.10 Problem Arcus 1 83 (Cycle Time = 6309)

Result

1> Time 6080.0 Idle Time 229.0 : 1 83 81 79

2> Time 6177.0 Idle Time 132.0 : 82 80 77 34 32

3> Time 6285.0 Idle Time 24.0 : 43 40 8 5 39 25

4> Time 6237.0 Idle Time 72.0 : 44 15 10 14 46 42 19 20

5> Time 6218.0 Idle Time 91.0 : 48 26 31 28 16 76 13 37

6> Time 5838.0 Idle Time 471.0 : 18 12 30 9 23 35

7> Time 5868.0 Idle Time 441.0 : 6 17 29 75 33

8> Time 6113.0 Idle Time 196.0 : 68 21 67 3 4 66 78

9> Time 6101.0 Idle Time 208.0 : 65 64 7 11 36 27

10> Time 6025.0 Idle Time 284.0 : 63 69 49 50 51 22 24 2 62

11> Time 5468.0 Idle Time 841.0 : 71 52 70 47 53 61 54 38

12> Time 5796.0 Idle Time 513.0 : 45 72 60 55 41 56 57 73

13> Time 3699.0 Idle Time 2610.0 : 59 58 74

Idle Time All : 6112.0 E=92.55 %

5.11 Problem Arcus 1 83 (Cycle Time = 6842)

Result

1> Time 6799.0 Idle Time 43.0 : 1 2 83 81

2> Time 6577.0 Idle Time 265.0 : 3 82 5 8 77

3> Time 6836.0 Idle Time 6.0 : 40 78 6 76 42 16

4> Time 6777.0 Idle Time 65.0 : 39 75 12 9 68 31 15 44

5> Time 6764.0 Idle Time 78.0 : 67 74 66 10 30 46 20

6> Time 6787.0 Idle Time 55.0 : 29 48 24 25 13 18 50 65 32 11

7> Time 6747.0 Idle Time 95.0 : 17 21 51 33 64 27 63 28 69

8> Time 6808.0 Idle Time 34.0 : 70 52 62 53 54 22 36 73 71

9> Time 6815.0 Idle Time 27.0 : 55 14 43 7 49 38 79

10> Time 6827.0 Idle Time 15.0 : 56 41 45 72 37 80 4

11> Time 6434.0 Idle Time 408.0 : 35 34 26 61 23 57 47 19

12> Time 1734.0 Idle Time 5108.0 : 58 59 60

Idle Time All : 6199.0 E=92.45 %

5.12 Problem Arcus 1 83 (Cycle Time = 6883)

Result

1> Time 6664.0 Idle Time 219.0 : 1 83 82

2> Time 3535.0 Idle Time 3348.0 : 2 81 80

3> Time 5815.0 Idle Time 1068.0 : 5 4 3 79 8

4> Time 6844.0 Idle Time 39.0 : 77 34 35 76 32 33 40 16 43

5> Time 6567.0 Idle Time 316.0 : 25 36 37 75 10 38 12

6> Time 6574.0 Idle Time 309.0 : 68 41 13 20 67 6 39
7> Time 6414.0 Idle Time 469.0 : 15 44 31 45 46 26 66 17 30
8> Time 6473.0 Idle Time 410.0 : 21 29 47 24 19 65 64 22 11 48
9> Time 6807.0 Idle Time 76.0 : 63 50 51 62 7 73 78
10> Time 6722.0 Idle Time 161.0 : 52 61 72 71 74 49 23
11> Time 6764.0 Idle Time 119.0 : 70 60 59 69 28 9 53
12> Time 6726.0 Idle Time 157.0 : 58 54 57 56 14 18 27 55 42
Idle Time All : 6691.0 E=91.9 %

5.13 Problem Arcus 1 83 (Cycle Time = 7571)

Result

1> Time 6799.0 Idle Time 772.0 : 1 2 83 81
2> Time 7294.0 Idle Time 277.0 : 82 79 77 40 80 35 76 75
3> Time 7437.0 Idle Time 134.0 : 16 73 68 34 78 3 67 12
4> Time 7522.0 Idle Time 49.0 : 9 66 6 65 72 71
5> Time 7382.0 Idle Time 189.0 : 70 64 63 4 74 69 43 49
6> Time 7285.0 Idle Time 286.0 : 27 62 7 39 44 23 18 21
7> Time 7419.0 Idle Time 152.0 : 36 61 15 46 5 38 47 22
8> Time 7259.0 Idle Time 312.0 : 41 20 45 42 48 11 26 60 8 32
9> Time 7489.0 Idle Time 82.0 : 59 50 33 25 28 58 13 10 31
10> Time 6893.0 Idle Time 678.0 : 30 24 29 51 14 57 52 53 17
11> Time 3126.0 Idle Time 4445.0 : 56 19 55 54 37
Idle Time All : 7376.0 E=91.14 %

5.14 Problem Arcus 1 83 (Cycle Time = 8412)

Result

1> Time 7322.0 Idle Time 1090.0 : 1 2 4 83
2> Time 8167.0 Idle Time 245.0 : 82 6 81 5 3
3> Time 8192.0 Idle Time 220.0 : 8 9 78 76 42 39 15 40
4> Time 8401.0 Idle Time 11.0 : 75 77 12 11 20 35 37 16 74 68
5> Time 8242.0 Idle Time 170.0 : 67 32 33 26 31 73 36 25 44 79 66
6> Time 8399.0 Idle Time 13.0 : 28 65 38 24 72 71 64 19 69 13
7> Time 8036.0 Idle Time 376.0 : 27 63 18 30 62 70 17 29 14 22
8> Time 8014.0 Idle Time 398.0 : 61 60 59 58 46 7 21 57 48 23 49
9> Time 8012.0 Idle Time 400.0 : 50 47 56 55 51 34 10 80 41
10> Time 3120.0 Idle Time 5292.0 : 52 45 53 54 43
Idle Time All : 8215.0 E=90.23 %

5.15 Problem Arcus 1 83 (Cycle Time = 8898)

Result

1> Time 7649.0 Idle Time 1249.0 : 1 2 83 82

2> Time 8867.0 Idle Time 31.0 : 4 3 7 80 11 81 15
3> Time 8838.0 Idle Time 60.0 : 39 79 77 16 43 76 5 75 31 42 12
4> Time 8752.0 Idle Time 146.0 : 8 34 74 68 26 73 23 67

5.15 Problem Arcus 1 83 (Cycle Time = 8898)

Result

5> Time 8698.0 Idle Time 200.0 : 71 18 72 66 30 9 44 20 70 36
6> Time 8812.0 Idle Time 86.0 : 13 10 69 38 65 17 64 22 46 28
7> Time 8387.0 Idle Time 511.0 : 27 48 21 63 62 61 41 29 24 50 40 32 25
8> Time 8386.0 Idle Time 512.0 : 60 37 33 45 19 51 59 6 52 53
9> Time 7516.0 Idle Time 1382.0 : 54 55 47 58 56 57 14 35 49 78
Idle Time All : 4177.0 E=94.78 %

5.16 Problem Arcus 1 83 (Cycle Time = 10816)

Result

1> Time 9022.0 Idle Time 1794.0 : 1 2 83 5 4
2> Time 10792.0 Idle Time 24.0 : 82 3 81 79 77 35 78 80 8 43
3> Time 10726.0 Idle Time 90.0 : 39 36 40 16 38 42 37 76 12 75 15 11 68 20
4> Time 10611.0 Idle Time 205.0 : 44 46 13 33 73 48 67 66 50 74 65 9
5> Time 10703.0 Idle Time 113.0 : 17 28 64 24 63 62 21 23 61 51 29 71 18 10
6> Time 10785.0 Idle Time 31.0 : 27 60 69 30 14 59 25 26 7 72 41 49
7> Time 10354.0 Idle Time 462.0 : 58 70 57 47 45 56 6 32 31 22 34
8> Time 2912.0 Idle Time 7904.0 : 55 54 53 19 52
Idle Time All : 10623.0 E=87.72 %

6. Problem Scholl 297

6.1 Problem Scholl 297 (Cycle Time = 1394)

Result

1> Time 1317.0 Idle Time 77.0 : 1 294 286 284 279
2> Time 1388.0 Idle Time 6.0 : 296 289 249 283 278
3> Time 1391.0 Idle Time 3.0 : 297 239 261 254 247
4> Time 1362.0 Idle Time 32.0 : 30 94 4 83 111
5> Time 1385.0 Idle Time 9.0 : 285 255 101 96 95 22 5
6> Time 1388.0 Idle Time 6.0 : 295 181 262 25
7> Time 1377.0 Idle Time 17.0 : 240 176 177 188 184
8> Time 1334.0 Idle Time 60.0 : 237 243 241 245 242 179 195 91
9> Time 1341.0 Idle Time 53.0 : 190 196 268 109 116
10> Time 1393.0 Idle Time 1.0 : 187 152 253 260 142 136 139
11> Time 1381.0 Idle Time 13.0 : 244 165 267

6.1 Problem Scholl 297 (Cycle Time = 1394) (ต่อ)

Result

12> Time 1354.0 Idle Time 40.0 : 168 183 153 180 194 84 31 203 134

13> Time 1361.0 Idle Time 33.0 : 178 193 77 199 76 71

14> Time 1328.0 Idle Time 66.0 : 174 208 73 74 206 172

15> Time 1370.0 Idle Time 24.0 : 238 171 202 68 69 182

16> Time 1392.0 Idle Time 2.0 : 207 256 250 259 234 204 201 192 66

17> Time 1393.0 Idle Time 1.0 : 266 214 119 229 78

18> Time 1366.0 Idle Time 28.0 : 130 127 131 110

19> Time 1358.0 Idle Time 36.0 : 135 128 129 132

20> Time 1371.0 Idle Time 23.0 : 123 141 124 144 121 162

21> Time 1392.0 Idle Time 2.0 : 215 145 158 147 157 163 159 148 151 126

22> Time 1385.0 Idle Time 9.0 : 155 166 65 62

23> Time 1387.0 Idle Time 7.0 : 246 160 156 161 150

24> Time 1389.0 Idle Time 5.0 : 248 56

25> Time 1386.0 Idle Time 8.0 : 293

26> Time 1387.0 Idle Time 7.0 : 205 269 2 282 61

27> Time 1373.0 Idle Time 21.0 : 197 230 117 167 213 120 90 75 290 79

28> Time 1384.0 Idle Time 10.0 : 189 257 252 29 114 280

29> Time 1387.0 Idle Time 7.0 : 287 258 265 100 104 108 103 275

30> Time 1391.0 Idle Time 3.0 : 210 51 81 87 89 112 133 99

31> Time 1327.0 Idle Time 67.0 : 138 46 271 55 42 40 173 169

32> Time 1377.0 Idle Time 17.0 : 140 191 277 85 125 35

33> Time 1342.0 Idle Time 52.0 : 200 88 97 185

34> Time 1394.0 Idle Time 0.0 : 105 115 26 273 175 6 143

35> Time 1377.0 Idle Time 17.0 : 236 212 276 9 3

36> Time 1388.0 Idle Time 6.0 : 263 251 211 44 233 92 198

37> Time 1335.0 Idle Time 59.0 : 53 270 209 49 98 11

38> Time 1346.0 Idle Time 48.0 : 58 164 48 27 16

39> Time 1373.0 Idle Time 21.0 : 64 52 23 17 10 186

40> Time 1334.0 Idle Time 60.0 : 47 19 15 82 272 221

41> Time 1389.0 Idle Time 5.0 : 33 43 154 72 18

42> Time 1350.0 Idle Time 44.0 : 39 36 170 291 13

43> Time 1347.0 Idle Time 47.0 : 93 60 122 227 20 292

44> Time 1350.0 Idle Time 44.0 : 34 216 24 12 288 28

45> Time 1374.0 Idle Time 20.0 : 137 59 149 106 102 118 281

46> Time 1334.0 Idle Time 60.0 : 38 107 113 8 7

47> Time 1333.0 Idle Time 61.0 : 146 41 32 21 223 225 45

48> Time 1376.0 Idle Time 18.0 : 54 50 217 57

49> Time 1347.0 Idle Time 47.0 : 264 218 37 63 70 219 220 222

50> Time 1326.0 Idle Time 68.0 : 235 274 67 231 86 80 14 224

51> Time 1325.0 Idle Time 69.0 : 232 226 228
Idle Time All : 1439.0 E=97.98 %

6.2 Problem Scholl 297 (Cycle Time = 1452)

Result

1> Time 1384.0 Idle Time 68.0 : 1 297 2 30
2> Time 1430.0 Idle Time 22.0 : 34 31 82 88 99 26 100 4
3> Time 1435.0 Idle Time 17.0 : 111 89 87 94 104 83 105
4> Time 1429.0 Idle Time 23.0 : 101 81 106 112 259 22 266 134
5> Time 1444.0 Idle Time 8.0 : 271 25 116 247
6> Time 1431.0 Idle Time 21.0 : 122 139 117 123 274 142
7> Time 1442.0 Idle Time 10.0 : 277 130 128 148 283
8> Time 1404.0 Idle Time 48.0 : 144 159 126 146 24 230 86
9> Time 1406.0 Idle Time 46.0 : 289 158 232 227 40 118
10> Time 1393.0 Idle Time 59.0 : 113 84 236 44 225
11> Time 1398.0 Idle Time 54.0 : 49 131 73 68 53 239
12> Time 1435.0 Idle Time 17.0 : 240 132 243 125 78
13> Time 1441.0 Idle Time 11.0 : 135 133 72 170 174 192 178
14> Time 1444.0 Idle Time 8.0 : 288 169 171 137 140 143 181 237
15> Time 1452.0 Idle Time 0.0 : 233 196 231 200 202 197 204 191 251 184 188
256
16> Time 1431.0 Idle Time 21.0 : 189 245 199 287 279
17> Time 1430.0 Idle Time 22.0 : 244 185
18> Time 1433.0 Idle Time 19.0 : 255 77 121 182 286 177 79
19> Time 1424.0 Idle Time 28.0 : 85 97 95 27 257 241
20> Time 1435.0 Idle Time 17.0 : 183 229 270 110 235
21> Time 1446.0 Idle Time 6.0 : 193 120 152 156 64 149 164 201
22> Time 1427.0 Idle Time 25.0 : 119 127 160 59 221 109
23> Time 1403.0 Idle Time 49.0 : 157 55 180 253 108 93 147 242
24> Time 1449.0 Idle Time 3.0 : 252 258 265 98 292 272 175 91
25> Time 1451.0 Idle Time 1.0 : 275 260 280 186 80 74 261 136
26> Time 1424.0 Idle Time 28.0 : 267 254 268 58 249 60 75 190 150
27> Time 1410.0 Idle Time 42.0 : 273 70 54 161 284 167 138 71
28> Time 1435.0 Idle Time 17.0 : 46 166 69 223 207
29> Time 1435.0 Idle Time 17.0 : 66 163 62 42 210 65 151 154
30> Time 1417.0 Idle Time 35.0 : 61 208 213 214 107 176 250 290
31> Time 1424.0 Idle Time 28.0 : 215 165 234 162 168 48 50 205
32> Time 1397.0 Idle Time 55.0 : 216 52 45 47 217 262
33> Time 1431.0 Idle Time 21.0 : 41 43 38 39 155 67
34> Time 1404.0 Idle Time 48.0 : 33 295 238
35> Time 1415.0 Idle Time 37.0 : 218 276 51 195 35

36> Time 1434.0 Idle Time 18.0 : 219 281 291 220 102 285 203 145
37> Time 1416.0 Idle Time 36.0 : 222 248 153
38> Time 1419.0 Idle Time 33.0 : 296 187 5
39> Time 1438.0 Idle Time 14.0 : 6 246 124 76
40> Time 1450.0 Idle Time 2.0 : 7 8 9 12 198 212
41> Time 1432.0 Idle Time 20.0 : 17 13 15 21 18 173
42> Time 1422.0 Idle Time 30.0 : 11 10 14 20 23 172
43> Time 1424.0 Idle Time 28.0 : 16 28 37 19 103 63
44> Time 1390.0 Idle Time 62.0 : 32 206 263 141 282
45> Time 1412.0 Idle Time 40.0 : 57 114 179 264 115 224
46> Time 1358.0 Idle Time 94.0 : 226 294 36 194 29 56
47> Time 1418.0 Idle Time 34.0 : 278 90 228 211
48> Time 1367.0 Idle Time 85.0 : 209 92 96 129 3 269
49> Time 1386.0 Idle Time 66.0 : 293
Idle Time All : 1493.0 E=97.9 %

6.3 Problem Scholl 297 (Cycle Time = 1483)

Result

1> Time 1407.0 Idle Time 76.0 : 1 294 2 3 286
2> Time 1461.0 Idle Time 22.0 : 235 229 195 279 237 203 190 198
3> Time 1479.0 Idle Time 4.0 : 186 187 208 189 181 197 233 182
4> Time 1483.0 Idle Time 0.0 : 210 196 178 184 231 179 202 207
5> Time 1464.0 Idle Time 19.0 : 200 174 191 251 295 31 177
6> Time 1483.0 Idle Time 0.0 : 256 172 175 171 170 34 287 188 173
7> Time 1471.0 Idle Time 12.0 : 263 30 166 168
8> Time 1473.0 Idle Time 10.0 : 163 159 165 270 26 156 167 162
9> Time 1480.0 Idle Time 3.0 : 148 146 110 120 164 193 4 22 206 123
10> Time 1450.0 Idle Time 33.0 : 117 152 24 109 5 56 94 115 130 112 149
11> Time 1435.0 Idle Time 48.0 : 127 131 60 119
12> Time 1472.0 Idle Time 11.0 : 114 106 101 40 147 204
13> Time 1441.0 Idle Time 42.0 : 96 44 84 97 86
14> Time 1448.0 Idle Time 35.0 : 77 71 111
15> Time 1473.0 Idle Time 10.0 : 128 6 29 125
16> Time 1451.0 Idle Time 32.0 : 78 192 79 144 73 142 10 145
17> Time 1454.0 Idle Time 29.0 : 194 154 236 93 183 48
18> Time 1478.0 Idle Time 5.0 : 52 98 102 107 274 92 68 161
19> Time 1469.0 Idle Time 14.0 : 277 283 271 269 230 37
20> Time 1478.0 Idle Time 5.0 : 227 264 225 261 254 268 289 266
21> Time 1451.0 Idle Time 32.0 : 57 28 25 201 153 205 129 249
22> Time 1467.0 Idle Time 16.0 : 143 63 32 141 160 67 39 247
23> Time 1460.0 Idle Time 23.0 : 248 70

24> Time 1461.0 Idle Time 22.0 : 246 43 259 213 80
25> Time 1469.0 Idle Time 14.0 : 211 257 47 133 169 72 134
26> Time 1476.0 Idle Time 7.0 : 293 221
27> Time 1472.0 Idle Time 11.0 : 290 292 136 284 253 151 121 150 20 262
28> Time 1482.0 Idle Time 1.0 : 214 83 132 157 155 199 33 278
29> Time 1465.0 Idle Time 18.0 : 90 234 38 288 15
30> Time 1447.0 Idle Time 36.0 : 21 41 116 138
31> Time 1419.0 Idle Time 64.0 : 46 42 82 126 140 12 255
32> Time 1440.0 Idle Time 43.0 : 118 88 99 17 89 8 176 87 212
33> Time 1428.0 Idle Time 55.0 : 103 100 81 104 244
34> Time 1456.0 Idle Time 27.0 : 232 122 124 7 209
35> Time 1476.0 Idle Time 7.0 : 51 291 139 49 280
36> Time 1447.0 Idle Time 36.0 : 281 276 76 53 74 275 242
37> Time 1461.0 Idle Time 22.0 : 69 65 66 273 241 14 238 105
38> Time 1478.0 Idle Time 5.0 : 19 267 297 13 180
39> Time 1404.0 Idle Time 79.0 : 252 18 27 36 243
40> Time 1477.0 Idle Time 6.0 : 258 228 75 55
41> Time 1469.0 Idle Time 14.0 : 265 226 59 224 222 215 285 11 23
42> Time 1483.0 Idle Time 0.0 : 220 16 219 216 113 45 272 158 135
43> Time 1430.0 Idle Time 53.0 : 217 50 54 64 58
44> Time 1439.0 Idle Time 44.0 : 240 218
45> Time 1376.0 Idle Time 107.0 : 239 282 91 9
46> Time 1461.0 Idle Time 22.0 : 260 108 137 250 61 95 85
47> Time 1262.0 Idle Time 221.0 : 62 223 35 185
48> Time 1349.0 Idle Time 134.0 : 245 296
Idle Time All : 1529.0 E=97.85 %

6.4 Problem Scholl 297 (Cycle Time = 1515)

Result

1> Time 1498.0 Idle Time 17.0 : 1 295 181 176 177
2> Time 1493.0 Idle Time 22.0 : 165 153 183 188 168 143 173 187 186 178
3> Time 1502.0 Idle Time 13.0 : 140 200 296 197
4> Time 1475.0 Idle Time 40.0 : 138 191 294 180 174 288 182
5> Time 1467.0 Idle Time 48.0 : 171 175 278 46 54 167 169 284 283
6> Time 1496.0 Idle Time 19.0 : 50 293 249
7> Time 1491.0 Idle Time 24.0 : 277 164 292 45 290 254 184
8> Time 1513.0 Idle Time 2.0 : 280 152 108 115 275 291 104 190 41 100
9> Time 1505.0 Idle Time 10.0 : 120 127 281 276 87
10> Time 1482.0 Idle Time 33.0 : 157 119 150 297 130 126 161
11> Time 1461.0 Idle Time 54.0 : 123 147 128 124 125 78
12> Time 1512.0 Idle Time 3.0 : 79 158 118 85 131 144 148 146

13> Time 1512.0 Idle Time 3.0 : 133 159 121 154 92
14> Time 1512.0 Idle Time 3.0 : 98 192 202 282 51
15> Time 1489.0 Idle Time 26.0 : 55 201 204 199 207 103 72 2 59
16> Time 1507.0 Idle Time 8.0 : 210 3 213 109 30 88 110 212
17> Time 1507.0 Idle Time 8.0 : 208 105 82 26 179 289 81
18> Time 1494.0 Idle Time 21.0 : 230 31 271 266 287 264 142 172
19> Time 1472.0 Idle Time 43.0 : 268 257 139 136 38 132 117 162
20> Time 1510.0 Idle Time 5.0 : 253 112 134 33 93 185 42 114
21> Time 1501.0 Idle Time 14.0 : 86 35 272 156 80 193 129
22> Time 1491.0 Idle Time 24.0 : 265 198 258 203 195 211 64 102 4
23> Time 1515.0 Idle Time 0.0 : 94 56 206 83 221 40 223 84 22 48 149
24> Time 1490.0 Idle Time 25.0 : 60 44 52 76 25 24
25> Time 1440.0 Idle Time 75.0 : 47 75 101 90 57
26> Time 1506.0 Idle Time 9.0 : 63 95 71 62 66 77
27> Time 1515.0 Idle Time 0.0 : 69 65 70 107 97 225 74
28> Time 1512.0 Idle Time 3.0 : 49 229 68 259 145
29> Time 1447.0 Idle Time 68.0 : 235 53 91 61 274 286 237 205
30> Time 1505.0 Idle Time 10.0 : 270 279 106 261 250 194
31> Time 1466.0 Idle Time 49.0 : 255 244 242 245 256 241
32> Time 1512.0 Idle Time 3.0 : 246 34 155 67 233 196
33> Time 1509.0 Idle Time 6.0 : 240 27
34> Time 1498.0 Idle Time 17.0 : 238 234 232 239 231 58 262
35> Time 1472.0 Idle Time 43.0 : 99 122 252 214 247 285 160
36> Time 1476.0 Idle Time 39.0 : 215 189 73 227 5
37> Time 1498.0 Idle Time 17.0 : 6 216 9 8 141 151 163
38> Time 1460.0 Idle Time 55.0 : 7 13 12 20 15 170
39> Time 1512.0 Idle Time 3.0 : 21 23 11 14 17 10
40> Time 1457.0 Idle Time 58.0 : 28 16 37 39 36 89
41> Time 1451.0 Idle Time 64.0 : 32 18 228 226
42> Time 1500.0 Idle Time 15.0 : 224 116 43 96
43> Time 1477.0 Idle Time 38.0 : 222 166 273 29 243 251
44> Time 1442.0 Idle Time 73.0 : 267 220 248
45> Time 1407.0 Idle Time 108.0 : 219 218 269 263 113
46> Time 1396.0 Idle Time 119.0 : 260 217 236 137
47> Time 1302.0 Idle Time 213.0 : 111 209 135 19

6.4 Problem Scholl 297 (Cycle Time = 1515) (ต่อ)

Result

Idle Time All : 1550.0 E=97.82 %

6.5 Problem Scholl 297 (Cycle Time = 1548)

Result

1> Time 1457.0 Idle Time 91.0 : 1 294 278 283 2 284 249

2> Time 1537.0 Idle Time 11.0 : 296 54 50 3

3> Time 1473.0 Idle Time 75.0 : 254 4 221 48 110 45 83 105

4> Time 1498.0 Idle Time 50.0 : 86 27 94 22 101 120 261

5> Time 1479.0 Idle Time 69.0 : 56 52 106 115 25 268 112 150 90

6> Time 1543.0 Idle Time 5.0 : 57 108 62 114 76 255 161

7> Time 1529.0 Idle Time 19.0 : 66 244 262 154

8> Time 1464.0 Idle Time 84.0 : 242 248 241

9> Time 1480.0 Idle Time 68.0 : 277 5 259 119 117 97

10> Time 1539.0 Idle Time 9.0 : 6 10 292 13 266 264 162 257 123 73

11> Time 1547.0 Idle Time 1.0 : 15 21 7 271 148 149 146 145 160

12> Time 1539.0 Idle Time 9.0 : 9 159 155 95 230 158 23

13> Time 1538.0 Idle Time 10.0 : 285 77 37 28 32

14> Time 1460.0 Idle Time 88.0 : 18 238 232 39 36

15> Time 1526.0 Idle Time 22.0 : 24 239 234 236 256 130 43

16> Time 1526.0 Idle Time 22.0 : 156 263 250 131 126 129

17> Time 1508.0 Idle Time 40.0 : 174 170 133 270 204 41 141 135

18> Time 1508.0 Idle Time 40.0 : 171 178 169 140 137 207 38 33

19> Time 1524.0 Idle Time 24.0 : 143 181 191 188 196 183 212 205 153 165 138

20> Time 1526.0 Idle Time 22.0 : 177 180 189 252 197 186 46

21> Time 1529.0 Idle Time 19.0 : 269 200 258 185 58

22> Time 1518.0 Idle Time 30.0 : 286 182 144 49 53 64

23> Time 1540.0 Idle Time 8.0 : 240 237 235 190 72

24> Time 1520.0 Idle Time 28.0 : 233 231 51 243 128 173

25> Time 1541.0 Idle Time 7.0 : 87 81 99 228 226 224 75

26> Time 1521.0 Idle Time 27.0 : 104 100 89 88 222 220 82 297

27> Time 1521.0 Idle Time 27.0 : 109 30 219 166

28> Time 1546.0 Idle Time 2.0 : 218 217 168 157 163 34

29> Time 1520.0 Idle Time 28.0 : 210 251 246 216 35

30> Time 1529.0 Idle Time 19.0 : 208 29 44 61 17 122

31> Time 1546.0 Idle Time 2.0 : 265 203 198 14 65 55 229

32> Time 1529.0 Idle Time 19.0 : 272 60 68 151 116

33> Time 1516.0 Idle Time 32.0 : 247 275 111 16 206 118

34> Time 1480.0 Idle Time 68.0 : 103 127 215 113

35> Time 1529.0 Idle Time 19.0 : 26 223 102 187 107 245 59 227

36> Time 1494.0 Idle Time 54.0 : 289 202 225 98 176 71 134 195 184

37> Time 1548.0 Idle Time 0.0 : 199 201 192 136 63 287 139 253

38> Time 1526.0 Idle Time 22.0 : 67 293

39> Time 1522.0 Idle Time 26.0 : 142 147 69 280 290 31 282 288

40> Time 1512.0 Idle Time 36.0 : 152 172 213 20 47 12 279 124
41> Time 1547.0 Idle Time 1.0 : 164 211 42 295
42> Time 1546.0 Idle Time 2.0 : 194 193 78 209 19 291 125
43> Time 1516.0 Idle Time 32.0 : 79 85 74 260 40 132 93
44> Time 1449.0 Idle Time 99.0 : 8 274 80 273 267 214
45> Time 1464.0 Idle Time 84.0 : 84 96 70 175 92 11 276
46> Time 1445.0 Idle Time 103.0 : 167 179 91 121 281
Idle Time All : 1553.0 E=97.82 %

6.6 Problem Scholl 297 (Cycle Time = 1584)

Result

1> Time 1508.0 Idle Time 76.0 : 1 2 295
2> Time 1521.0 Idle Time 63.0 : 181 187 177 173 294 185
3> Time 1549.0 Idle Time 35.0 : 179 186 195 184 284 286 279 249 289 205
4> Time 1582.0 Idle Time 2.0 : 248 207 212 188 168
5> Time 1577.0 Idle Time 7.0 : 204 245 246 254 182
6> Time 1583.0 Idle Time 1.0 : 202 201 165 261 214 251 256
7> Time 1535.0 Idle Time 49.0 : 215 234 263 268 285 270 194
8> Time 1584.0 Idle Time 0.0 : 192 296 278 203
9> Time 1578.0 Idle Time 6.0 : 247 4 40 48 198 134 208 193 136 94
10> Time 1545.0 Idle Time 39.0 : 44 27 86 83 238 216 283
11> Time 1580.0 Idle Time 4.0 : 240 277 90 237 241
12> Time 1577.0 Idle Time 7.0 : 233 95 231 244 242 199
13> Time 1575.0 Idle Time 9.0 : 228 217 235 178
14> Time 1536.0 Idle Time 48.0 : 174 288 218 287 171 219
15> Time 1580.0 Idle Time 4.0 : 170 133 220 166 3 222
16> Time 1582.0 Idle Time 2.0 : 160 155 163 131 151 297 93 176 243
17> Time 1565.0 Idle Time 19.0 : 149 130 98 128 102 123 125 148
18> Time 1577.0 Idle Time 7.0 : 121 144 154 126 107 145
19> Time 1541.0 Idle Time 43.0 : 118 293
20> Time 1584.0 Idle Time 0.0 : 292 282 290 291 281 159 190
21> Time 1576.0 Idle Time 8.0 : 280 276 30 84 167 54 239 22
22> Time 1544.0 Idle Time 40.0 : 162 50 77 236 275 273
23> Time 1580.0 Idle Time 4.0 : 272 164 74 45 69 34 76
24> Time 1582.0 Idle Time 2.0 : 41 265 258 65 25 75 267 91 255
25> Time 1574.0 Idle Time 10.0 : 252 38 96 61 70 262 33 78
26> Time 1555.0 Idle Time 29.0 : 260 161 58 72 35 226 213
27> Time 1507.0 Idle Time 77.0 : 42 53 64 274 109 80 79 105
28> Time 1576.0 Idle Time 8.0 : 119 59 85 127 55 31
29> Time 1572.0 Idle Time 12.0 : 99 103 100 172 104 46 269 67 153
30> Time 1537.0 Idle Time 47.0 : 63 175 143 138 140 250 183 114

31> Time 1562.0 Idle Time 22.0 : 191 111 108 120 49 87
32> Time 1555.0 Idle Time 29.0 : 115 81 116 57
33> Time 1578.0 Idle Time 6.0 : 139 62 232 189 56
34> Time 1538.0 Idle Time 46.0 : 60 68 150 101 110 71 82
35> Time 1544.0 Idle Time 40.0 : 113 197 271 51 196 180 92 142 200
36> Time 1561.0 Idle Time 23.0 : 264 5 157 259 141 266 147 122 146
37> Time 1513.0 Idle Time 71.0 : 158 6 257 8 12 10 20 221
38> Time 1562.0 Idle Time 22.0 : 23 9 21 19 13
39> Time 1522.0 Idle Time 62.0 : 16 18 28 37 39 36
40> Time 1507.0 Idle Time 77.0 : 43 15 7 32 230 29
41> Time 1573.0 Idle Time 11.0 : 47 106 124 227 129 73
42> Time 1574.0 Idle Time 10.0 : 112 225 135 152 137 211 210 223 206 117
43> Time 1550.0 Idle Time 34.0 : 229 209 26 52 89
44> Time 1582.0 Idle Time 2.0 : 88 11 156 66 24 224 169 253
45> Time 1072.0 Idle Time 512.0 : 97 17 14 132
Idle Time All : 1625.0 E=97.72 %

6.7 Problem Scholl 297 (Cycle Time = 1620)

Result

1> Time 1508.0 Idle Time 112.0 : 1 295 2
2> Time 1620.0 Idle Time 0.0 : 293 3 292
3> Time 1590.0 Idle Time 30.0 : 291 4 283 83 247 111 94
4> Time 1597.0 Idle Time 23.0 : 277 90 221 109 223 120 115 40 278
5> Time 1611.0 Idle Time 9.0 : 95 225 227 150 290 116
6> Time 1597.0 Idle Time 23.0 : 282 195 280 139 294
7> Time 1580.0 Idle Time 40.0 : 281 187 203 185 188 190 275 136 198
8> Time 1606.0 Idle Time 14.0 : 276 182 256 204 251 202 196 199 193
289
9> Time 1599.0 Idle Time 21.0 : 296 263 250 207
10> Time 1600.0 Idle Time 20.0 : 105 127 54 130 288
11> Time 1572.0 Idle Time 48.0 : 123 147 146 148 174 178 171 201 158
122 170 134
12> Time 1620.0 Idle Time 0.0 : 101 206 149 169 212 249 284 140 143
13> Time 1618.0 Idle Time 2.0 : 96 97 74 76 270 161 56
14> Time 1614.0 Idle Time 6.0 : 69 75 66 62 70 65 167 68 60
15> Time 1613.0 Idle Time 7.0 : 106 67 63 189 142 153 160
16> Time 1590.0 Idle Time 30.0 : 274 112 5 229 119
17> Time 1594.0 Idle Time 26.0 : 6 7 10 20 236 50

18> Time 1605.0 Idle Time 15.0 : 13 11 23 183 285 45 41
19> Time 1614.0 Idle Time 6.0 : 205 210 22 235 194 9 28 37
20> Time 1558.0 Idle Time 62.0 : 24 253 237 184 25 157 77
21> Time 1540.0 Idle Time 80.0 : 260 240 243 71
22> Time 1619.0 Idle Time 1.0 : 21 241 246 186 133
23> Time 1581.0 Idle Time 39.0 : 267 197 244 255 254
24> Time 1619.0 Idle Time 1.0 : 73 154 155 166 163
25> Time 1595.0 Idle Time 25.0 : 91 108 145 104 100 138 262 273 32 29
26> Time 1594.0 Idle Time 26.0 : 214 87 191 46 42 88 35 89
27> Time 1597.0 Idle Time 23.0 : 86 234 215 216 51 144 181 176
28> Time 1537.0 Idle Time 83.0 : 93 297 179 180 173 177
29> Time 1580.0 Idle Time 40.0 : 81 30 34 31 172 175 252 168 61
30> Time 1571.0 Idle Time 49.0 : 261 245 217 152 258 265
31> Time 1591.0 Idle Time 29.0 : 114 268 200 218 26 219 27
32> Time 1599.0 Idle Time 21.0 : 125 248 48 220 222
33> Time 1620.0 Idle Time 0.0 : 78 80 79 52 18 224 8 226 72 33
34> Time 1607.0 Idle Time 13.0 : 85 47 43 228 231
35> Time 1606.0 Idle Time 14.0 : 92 12 156 126 165 98 19
36> Time 1619.0 Idle Time 1.0 : 269 39 102 107 44 99 162
37> Time 1527.0 Idle Time 93.0 : 238 239 131 230 49 113
38> Time 1589.0 Idle Time 31.0 : 132 53 124 84 137 151 117 279 14
39> Time 1526.0 Idle Time 94.0 : 129 141 38 287 257 264
40> Time 1615.0 Idle Time 5.0 : 64 128 286 59 242
41> Time 1595.0 Idle Time 25.0 : 17 58 208 213 135 55 272 164
42> Time 1582.0 Idle Time 38.0 : 211 110 209 233 232 159 36
43> Time 1494.0 Idle Time 126.0 : 121 192 271 16 82
44> Time 1346.0 Idle Time 274.0 : 266 103 15 57 118 259
Idle Time All : 1625.0 E=97.72 %

6.8 Problem Scholl 297 (Cycle Time = 1659)

Result

1> Time 1587.0 Idle Time 72.0 : 1 296 2
2> Time 1620.0 Idle Time 39.0 : 3 293 292
3> Time 1594.0 Idle Time 65.0 : 108 115 114 170 174 287 290 283 282
4> Time 1595.0 Idle Time 64.0 : 294 133 288 131
5> Time 1635.0 Idle Time 24.0 : 137 135 128 130 125 129

6.8 Problem Scholl 297 (Cycle Time = 1659) (ต่อ)

Result

6> Time 1635.0 Idle Time 24.0 : 285 140 200 124 144 154 123

7> Time 1658.0 Idle Time 1.0 : 153 143 103 138 191 171 122 148 159 147

8> Time 1613.0 Idle Time 46.0 : 4 176 165 88 99 87 46 51 158

9> Time 1657.0 Idle Time 2.0 : 94 247 259 56 221 83 27 31 82 168 100

10> Time 1609.0 Idle Time 50.0 : 134 223 5 266 90 179 6 10 61

11> Time 1614.0 Idle Time 45.0 : 136 271 264 9 14 8 65

12> Time 1659.0 Idle Time 0.0 : 22 111 257 177 173 69 66 62

13> Time 1640.0 Idle Time 19.0 : 26 24 60 12 74 84 268 76

14> Time 1650.0 Idle Time 9.0 : 86 15 97 249 91 261 254

15> Time 1612.0 Idle Time 47.0 : 248 245

16> Time 1628.0 Idle Time 31.0 : 29 172 202 246

17> Time 1639.0 Idle Time 20.0 : 44 175 196 23 251 204 207 212 28 37

18> Time 1653.0 Idle Time 6.0 : 49 205 210 199 39 43 208

19> Time 1622.0 Idle Time 37.0 : 53 181 197 17 256 213 203 18 101

20> Time 1609.0 Idle Time 50.0 : 225 182 189 187 184 188 206 198 183

21> Time 1655.0 Idle Time 4.0 : 286 19 263 201 190 96 106 112

22> Time 1651.0 Idle Time 8.0 : 235 279 211 25 104 7 126 284 186

23> Time 1624.0 Idle Time 35.0 : 237 233 231 121 192 277 118 113

24> Time 1655.0 Idle Time 4.0 : 250 295 42 36

25> Time 1638.0 Idle Time 21.0 : 55 13 145 119 35 269 109

26> Time 1622.0 Idle Time 37.0 : 110 59 155 262 240 241 255

27> Time 1642.0 Idle Time 17.0 : 162 185 243 230 229

28> Time 1648.0 Idle Time 11.0 : 11 120 239 77 68 242 141

29> Time 1618.0 Idle Time 41.0 : 195 116 151 30 163 150

30> Time 1616.0 Idle Time 43.0 : 297 146 21 227 48 161 149

31> Time 1648.0 Idle Time 11.0 : 78 58 167 52 166 160

32> Time 1622.0 Idle Time 37.0 : 80 79 72 164 152 142 157 253 260

33> Time 1643.0 Idle Time 16.0 : 270 267 169 33 132 54 117 40 95

34> Time 1656.0 Idle Time 3.0 : 289 193 105 273 238 50 194 214

35> Time 1635.0 Idle Time 24.0 : 281 276 81 45 234 215 216

36> Time 1604.0 Idle Time 55.0 : 178 75 41 217 218 219

37> Time 1582.0 Idle Time 77.0 : 70 67 32 228 222 73 220

38> Time 1658.0 Idle Time 1.0 : 209 63 20 280 226 224 127

39> Time 1604.0 Idle Time 55.0 : 93 64 47 71 275 57 180

40> Time 1600.0 Idle Time 59.0 : 98 102 232 272 265 92 258 252

41> Time 1552.0 Idle Time 107.0 : 16 291 236 38

42> Time 1582.0 Idle Time 77.0 : 244 278 139 107

43> Time 1371.0 Idle Time 288.0 : 156 274 85 89 34

Idle Time All : 1682.0 E=97.64 %

6.9 Problem Scholl 297 (Cycle Time = 1699)

Result

1> Time 1630.0 Idle Time 69.0 : 1 295 2 181

2> Time 1664.0 Idle Time 35.0 : 182 184 3 183 294 189 177 188

3> Time 1672.0 Idle Time 27.0 : 195 293 178 290

4> Time 1657.0 Idle Time 42.0 : 291 283 174 284 187 171 194 4 203 221 249

5> Time 1697.0 Idle Time 2.0 : 248 83 134 170

6> Time 1699.0 Idle Time 0.0 : 245 40 44 133 94 243 247

7> Time 1681.0 Idle Time 18.0 : 254 90 240 95

8> Time 1695.0 Idle Time 4.0 : 237 238 241 234 256 263 250 239 29 206

9> Time 1682.0 Idle Time 17.0 : 37 279 25 242 236 121 251 22

10> Time 1682.0 Idle Time 17.0 : 232 28 111 84 280 204

11> Time 1683.0 Idle Time 16.0 : 23 76 18 77 91 207 56

12> Time 1672.0 Idle Time 27.0 : 61 57 32 71 63 52 65 60

13> Time 1680.0 Idle Time 19.0 : 36 229 49 73 275

14> Time 1627.0 Idle Time 72.0 : 116 288 97

15> Time 1635.0 Idle Time 64.0 : 139 281 297 74 48

16> Time 1677.0 Idle Time 22.0 : 253 260 166 17 163 212

17> Time 1694.0 Idle Time 5.0 : 155 151 156 161 141 285 62 53 160

18> Time 1667.0 Idle Time 32.0 : 145 149 150 58 64 66 59 99 276 123 173

19> Time 1697.0 Idle Time 2.0 : 148 87 100 117 124 168 257 128 289

20> Time 1695.0 Idle Time 4.0 : 104 244 193 154 112

21> Time 1638.0 Idle Time 61.0 : 144 106 255 262 269 21

22> Time 1677.0 Idle Time 22.0 : 136 15 88 274 259 230 261

23> Time 1695.0 Idle Time 4.0 : 8 266 9 7 6 14 147

24> Time 1682.0 Idle Time 17.0 : 158 296 270 159 292

25> Time 1685.0 Idle Time 14.0 : 202 179 131 33 223 67 268

26> Time 1662.0 Idle Time 37.0 : 201 200 225 70 120 47 227 278 286 197 86

27> Time 1699.0 Idle Time 0.0 : 93 115 98 92 85 43 110 192 198 176

28> Time 1698.0 Idle Time 1.0 : 78 79 80 162 101 19 169 167 96 125

29> Time 1670.0 Idle Time 29.0 : 164 152 129 208 105 31 72 273

30> Time 1663.0 Idle Time 36.0 : 34 127 130 267 122 233

31> Time 1659.0 Idle Time 40.0 : 231 126 228 69 82

32> Time 1695.0 Idle Time 4.0 : 226 142 39 246 186 10 191

33> Time 1680.0 Idle Time 19.0 : 287 282 185 146 205 132

34> Time 1667.0 Idle Time 32.0 : 89 16 272 11 165 119 264

35> Time 1662.0 Idle Time 37.0 : 114 153 265 27 258 30 102 196

36> Time 1643.0 Idle Time 56.0 : 107 252 54 271 224 118 38

37> Time 1662.0 Idle Time 37.0 : 113 222 220 41 26 199 35 143 172 68

38> Time 1642.0 Idle Time 57.0 : 219 42 45 175 218 217

39> Time 1669.0 Idle Time 30.0 : 46 216 51 215 24 157 109 75
40> Time 1654.0 Idle Time 45.0 : 180 235 5 12 108 277 190 138
41> Time 1579.0 Idle Time 120.0 : 13 135 140 50 214 81 20 55
42> Time 1188.0 Idle Time 511.0 : 213 211 137 209 103 210
Idle Time All : 1703.0 E=97.61 %

6.10 Problem Scholl 297 (Cycle Time = 1742)

Result

1> Time 1726.0 Idle Time 16.0 : 1 297 294 289 2
2> Time 1738.0 Idle Time 4.0 : 230 249 181 182 183 295 284
3> Time 1710.0 Idle Time 32.0 : 227 248 188
4> Time 1725.0 Idle Time 17.0 : 243 246 187 180 178 196 184 197
5> Time 1737.0 Idle Time 5.0 : 244 255 175 174 252 194
6> Time 1728.0 Idle Time 14.0 : 240 202 171
7> Time 1741.0 Idle Time 1.0 : 238 239 237 233 242 167 251 256 288 199 261
170
8> Time 1736.0 Idle Time 6.0 : 279 241 269 166 163 157
9> Time 1723.0 Idle Time 19.0 : 236 156 263 160 151 161 154 150
10> Time 1707.0 Idle Time 35.0 : 231 225 245 146 155 195 190 262 207 204 270
250
11> Time 1735.0 Idle Time 7.0 : 228 145 158 203 212 189
12> Time 1727.0 Idle Time 15.0 : 208 206 200 214 213 234 144
13> Time 1733.0 Idle Time 9.0 : 179 137 140 143 191 201 138 268 176 198 164
14> Time 1741.0 Idle Time 1.0 : 31 130 162 133 132 135 165 120 147 152
15> Time 1727.0 Idle Time 15.0 : 128 126 168 186 142 139
16> Time 1711.0 Idle Time 31.0 : 27 34 110 131 118 173 119
17> Time 1722.0 Idle Time 20.0 : 4 56 22 83 48 5 52 86 26 25 177 114
18> Time 1729.0 Idle Time 13.0 : 247 259 94 90 47 43 39 221 57
19> Time 1703.0 Idle Time 39.0 : 266 36 32 111 6 76
20> Time 1727.0 Idle Time 15.0 : 134 61 28 95 7 8 20 12 125
21> Time 1736.0 Idle Time 6.0 : 24 93 9 14 17 78 149 71
22> Time 1709.0 Idle Time 33.0 : 60 98 105 113 79 63 77 19
23> Time 1725.0 Idle Time 17.0 : 68 92 13 35 123 107 108 40 253
24> Time 1665.0 Idle Time 77.0 : 148 117 257 112 73 100 292 101 159 104 15
25> Time 1698.0 Idle Time 44.0 : 106 81 87 89 99 96 91 21
26> Time 1739.0 Idle Time 3.0 : 88 102 296 18
27> Time 1711.0 Idle Time 31.0 : 232 44 185 42 84
28> Time 1734.0 Idle Time 8.0 : 3 11 82 49 109 85 53
29> Time 1699.0 Idle Time 43.0 : 286 136 16 51 65 75 23 58
30> Time 1726.0 Idle Time 16.0 : 235 55 103 70 129 10 172 124 67
31> Time 1736.0 Idle Time 6.0 : 59 74 122 169 264 287 30

32> Time 1709.0 Idle Time 33.0 : 254 193 62 72 64 192
33> Time 1660.0 Idle Time 82.0 : 115 66 293
34> Time 1729.0 Idle Time 13.0 : 283 80 54 226 290 282 280 277 223 224
35> Time 1705.0 Idle Time 37.0 : 50 219 220 222 116 69 275 272 265
36> Time 1683.0 Idle Time 59.0 : 45 218 291 210 274
37> Time 1715.0 Idle Time 27.0 : 41 216 217 281 260 258
38> Time 1715.0 Idle Time 27.0 : 33 38 121 276 273 267
39> Time 1696.0 Idle Time 46.0 : 278 29 153 141 215 229
40> Time 1708.0 Idle Time 34.0 : 285 271 205 46 211 37 97
41> Time 931.0 Idle Time 811.0 : 127 209
Idle Time All : 1767.0 E=97.53 %

6.11 Problem Scholl 297 (Cycle Time = 1787)

Result

1> Time 1705.0 Idle Time 82.0 : 1 294 297 284 283 278
2> Time 1759.0 Idle Time 28.0 : 249 254 30 26 247 274 2 289
3> Time 1784.0 Idle Time 3.0 : 34 31 4 86 5 3 94 22 27 105 221
4> Time 1692.0 Idle Time 95.0 : 56 172 296 223 25
5> Time 1739.0 Idle Time 48.0 : 35 175 6 10 83 110 269
6> Time 1726.0 Idle Time 61.0 : 14 7 12 17 262 23 19
7> Time 1785.0 Idle Time 2.0 : 48 82 18 28 32 16 37 162
8> Time 1785.0 Idle Time 2.0 : 88 99 87 52 100 103 42 134 29 89 47
9> Time 1722.0 Idle Time 65.0 : 108 46 138 136 44 49 33
10> Time 1774.0 Idle Time 13.0 : 9 115 292 90 140 191 200 137 114 51
11> Time 1767.0 Idle Time 20.0 : 204 180 202 196 201 197 143 135 55 181
12> Time 1776.0 Idle Time 11.0 : 256 263 207 212 205 188 133 153 178 184 194
177 169 251
13> Time 1720.0 Idle Time 67.0 : 238 234 210 213 187 131 214
14> Time 1782.0 Idle Time 5.0 : 250 195 240 130 168 173
15> Time 1767.0 Idle Time 20.0 : 186 229 237 59 123 146 117 112 158 126
16> Time 1775.0 Idle Time 12.0 : 208 233 182 231 228 241 165
17> Time 1736.0 Idle Time 51.0 : 189 166 163 252
18> Time 1760.0 Idle Time 27.0 : 248 145 151 258
19> Time 1741.0 Idle Time 46.0 : 245 295 170 157 125
20> Time 1777.0 Idle Time 10.0 : 104 21 121 64 160 159 58 215
21> Time 1764.0 Idle Time 23.0 : 259 243 72 149 150 106 120 15 36
22> Time 1739.0 Idle Time 48.0 : 266 271 190 277 60 225 161 129 124 155
23> Time 1760.0 Idle Time 27.0 : 227 264 268 68 73 253 226 78 239 154
24> Time 1772.0 Idle Time 15.0 : 279 139 80 79 244 141
25> Time 1776.0 Idle Time 11.0 : 282 286 144 260 97
26> Time 1736.0 Idle Time 51.0 : 235 261 74 75 81 265 77

6.11 Problem Scholl 297 (Cycle Time = 1787) (ต่อ)

Result

27> Time 1754.0 Idle Time 33.0 : 53 43 71 76 216 67 70 272 20 95

28> Time 1781.0 Idle Time 6.0 : 255 63 257 280 275 224 101 217 142

29> Time 1739.0 Idle Time 48.0 : 122 69 290 222 220 218 91 11 96

30> Time 1772.0 Idle Time 15.0 : 203 66 293 167

31> Time 1717.0 Idle Time 70.0 : 206 198 62 291 288 164 219

32> Time 1786.0 Idle Time 1.0 : 193 281 236 24 152 171 84

33> Time 1754.0 Idle Time 33.0 : 230 54 232 276 40 147 270

34> Time 1768.0 Idle Time 19.0 : 285 93 50 132 13 57 267 174 118

35> Time 1748.0 Idle Time 39.0 : 179 98 211 45 41 92 102 113 107

36> Time 1786.0 Idle Time 1.0 : 209 38 85 127 183

37> Time 1767.0 Idle Time 20.0 : 242 116 61 176 39 273

38> Time 1765.0 Idle Time 22.0 : 128 246 65

39> Time 1757.0 Idle Time 30.0 : 109 199 119 185 111

40> Time 1142.0 Idle Time 645.0 : 192 148 156 8 287

Idle Time All : 1825.0 E=97.45 %

6.12 Problem Scholl 297 (Cycle Time = 1834)

Result

1> Time 1514.0 Idle Time 320.0 : 1 297 2 3 30

2> Time 1745.0 Idle Time 89.0 : 34 31 82 296

3> Time 1775.0 Idle Time 59.0 : 4 48 26 89 27 35 88

4> Time 1796.0 Idle Time 38.0 : 40 105 111 22 56 61 259 100 99 104 87

5> Time 1793.0 Idle Time 41.0 : 5 60 68 44 266 271 264 108 81

6> Time 1820.0 Idle Time 14.0 : 6 94 29 86 230 268 9 292 51 115

7> Time 1799.0 Idle Time 35.0 : 8 10 227 37 12 225 223 261 289

8> Time 1745.0 Idle Time 89.0 : 14 232 39 255 262 269 36

9> Time 1833.0 Idle Time 1.0 : 195 116 254 13 83 249

10> Time 1820.0 Idle Time 14.0 : 186 187 181 185 177 122 18 90 95

11> Time 1815.0 Idle Time 19.0 : 188 203 178 183 184 194 208 173 139 142 253
65

12> Time 1813.0 Idle Time 21.0 : 206 207 210 174 287 288 171 198 136 42

13> Time 1829.0 Idle Time 5.0 : 212 170 166 159 167 202

14> Time 1824.0 Idle Time 10.0 : 182 157 162 160 201 155 192 200 78 204 156
146 24

15> Time 1790.0 Idle Time 44.0 : 149 191 79 140 85 138 137 161 229

16> Time 1830.0 Idle Time 4.0 : 150 250 260 256 25 158 127

17> Time 1798.0 Idle Time 36.0 : 205 221 263 147 133 119 103 92 69

18> Time 1828.0 Idle Time 6.0 : 270 213 66 74 62 46 251 154

19> Time 1809.0 Idle Time 25.0 : 33 49 190 257 211 209 274 148 134

20> Time 1827.0 Idle Time 7.0 : 145 38 41 278 277 283 282 98
21> Time 1831.0 Idle Time 3.0 : 45 72 102 121 32 180 52
22> Time 1809.0 Idle Time 25.0 : 64 107 113 128 175 47
23> Time 1787.0 Idle Time 47.0 : 58 7 118 126 131 59 236
24> Time 1823.0 Idle Time 11.0 : 97 123 132 241 239 240
25> Time 1830.0 Idle Time 4.0 : 28 76 77 130 242 19 237 279 286
26> Time 1787.0 Idle Time 47.0 : 143 124 248
27> Time 1777.0 Idle Time 57.0 : 153 196 197 245 246 193 199
28> Time 1800.0 Idle Time 34.0 : 23 165 144 101 135 84
29> Time 1834.0 Idle Time 0.0 : 163 21 73 252 106 112 96 91
30> Time 1816.0 Idle Time 18.0 : 151 141 244 258 110 120
31> Time 1819.0 Idle Time 15.0 : 57 265 238 284 75 17 152 53
32> Time 1803.0 Idle Time 31.0 : 243 63 285 234 189 15 16
33> Time 1828.0 Idle Time 6.0 : 67 50 293 290 55
34> Time 1794.0 Idle Time 40.0 : 168 291 280 11 70 93 214 247 125
35> Time 1792.0 Idle Time 42.0 : 275 281 276 20 215 114 233 117 216
36> Time 1823.0 Idle Time 11.0 : 273 294 231 169 267 217 235
37> Time 1825.0 Idle Time 9.0 : 164 228 226 224 218 219
38> Time 1766.0 Idle Time 68.0 : 295 129 222 179 272 220
39> Time 1308.0 Idle Time 526.0 : 43 109 176 54 71 80 172
Idle Time All : 1871.0 E=97.38 %

6.13 Problem Scholl 297 (Cycle Time = 1883)

Result

1> Time 1844.0 Idle Time 39.0 : 1 296 294
2> Time 1799.0 Idle Time 84.0 : 54 2 289 284 295
3> Time 1878.0 Idle Time 5.0 : 174 178 50 249 230 196 197 179 181 180 232
4> Time 1882.0 Idle Time 1.0 : 288 169 202 297 170 204 271 266
5> Time 1873.0 Idle Time 10.0 : 207 200 191 30 177 166 163 155 151 26
6> Time 1849.0 Idle Time 34.0 : 212 34 257 264 156 4 134 83 221 90 223 247 22
7> Time 1878.0 Idle Time 5.0 : 286 140 146 159 268 5 86 27 157 56
8> Time 1864.0 Idle Time 19.0 : 279 137 40 111 61 189
9> Time 1838.0 Idle Time 45.0 : 240 241 237 44 195 239
10> Time 1850.0 Idle Time 33.0 : 233 231 121 29 242 229 33
11> Time 1878.0 Idle Time 5.0 : 41 228 38 183 37
12> Time 1879.0 Idle Time 4.0 : 244 255 238 160 193 39 94
13> Time 1846.0 Idle Time 37.0 : 246 245 149 198 253 101 262
14> Time 1879.0 Idle Time 4.0 : 128 105 269 97
15> Time 1877.0 Idle Time 6.0 : 130 129 124 144 110 120 77
16> Time 1868.0 Idle Time 15.0 : 167 162 164 125 122 143 93 150 126 187 145
17> Time 1858.0 Idle Time 25.0 : 152 250 192 78 79 154 36 153 115 28

18> Time 1877.0 Idle Time 6.0 : 243 256 214 205 72 64 32 48 173 31 80
19> Time 1831.0 Idle Time 52.0 : 254 213 215 201 52 168 172 57
20> Time 1871.0 Idle Time 12.0 : 45 211 123 117 47 69 74 66 62
21> Time 1878.0 Idle Time 5.0 : 293 209 206 112
22> Time 1880.0 Idle Time 3.0 : 290 292 283 282 277 147 59 203 43 25 190
23> Time 1883.0 Idle Time 0.0 : 248 99 100 55 88
24> Time 1861.0 Idle Time 22.0 : 234 89 104 103 87 81 49 236
25> Time 1880.0 Idle Time 3.0 : 171 227 226 225 53 127 60
26> Time 1882.0 Idle Time 1.0 : 216 138 224 85 3 133 176 175 109 71 95
27> Time 1851.0 Idle Time 32.0 : 46 42 222 217 165 6 259 278
28> Time 1882.0 Idle Time 1.0 : 220 219 218 7 9 12 10
29> Time 1812.0 Idle Time 71.0 : 8 20 11 141 252 184 68 73 280
30> Time 1837.0 Idle Time 46.0 : 96 16 275 82 119 142 261 287
31> Time 1868.0 Idle Time 15.0 : 91 136 272 274 114 194 158 139
32> Time 1832.0 Idle Time 51.0 : 263 260 270 161 265 98 131 199 15
33> Time 1847.0 Idle Time 36.0 : 210 21 267 208 132 235 51 75 35
34> Time 1881.0 Idle Time 2.0 : 251 13 23 273 70 58 84 24 135
35> Time 1871.0 Idle Time 12.0 : 17 19 182 276 18 67 281 285
36> Time 1880.0 Idle Time 3.0 : 92 106 118 76 186 116
37> Time 1841.0 Idle Time 42.0 : 148 188 113 107 258 65 102 14 63 291
38> Time 770.0 Idle Time 1113.0 : 185 108
Idle Time All : 1899.0 E=97.35 %

6.14 Problem Scholl 297 (Cycle Time = 1935)

Result

1> Time 1855.0 Idle Time 98.0 : 1 296 297
2> Time 1855.0 Idle Time 98.0 : 293 30 292 290 283
3> Time 1884.0 Idle Time 69.0 : 108 295 291 34
4> Time 1887.0 Idle Time 66.0 : 103 99 35 281 104 115 100 282 120 87
5> Time 1877.0 Idle Time 76.0 : 89 119 81 150 277 294 42 276
6> Time 1932.0 Idle Time 21.0 : 278 285 127 88 181 26 130 177
7> Time 1936.0 Idle Time 17.0 : 4 176 5 125 144 124 131 238 221 196
8> Time 1927.0 Idle Time 26.0 : 6 10 12 7 17 132 117 135 223
9> Time 1915.0 Idle Time 38.0 : 202 200 191 201 133 234 8 173 182 178
10> Time 1940.0 Idle Time 13.0 : 168 256 250 214 179 189 126 225 123
11> Time 1919.0 Idle Time 34.0 : 149 160 212 253 180 154 40 192 260 147
12> Time 1934.0 Idle Time 19.0 : 207 252 267 44 157 20 199 286 138 145
13> Time 1952.0 Idle Time 1.0 : 279 49 239 287 263 155 114 56 247 195 188
14> Time 1949.0 Idle Time 4.0 : 203 236 60 61 229 206 48 78
15> Time 1947.0 Idle Time 6.0 : 65 72 69 52 57 47 79 43 170 204
16> Time 1950.0 Idle Time 3.0 : 74 66 71 97 63 101 95 235

17> Time 1945.0 Idle Time 8.0 : 67 237 73 90 106 64 232 54 22
18> Time 1937.0 Idle Time 16.0 : 59 55 230 50 58 240 243
19> Time 1952.0 Idle Time 1.0 : 241 242 244 27 288 257
20> Time 1915.0 Idle Time 38.0 : 255 262 264 46 197 146 233 208 105 174 68 184 187
21> Time 1952.0 Idle Time 1.0 : 171 269 268 231 167 228 162 94
22> Time 1946.0 Idle Time 7.0 : 226 224 166 128 222
23> Time 1946.0 Idle Time 7.0 : 220 163 159 151 121 141 190 2 165 118 289
24> Time 1946.0 Idle Time 7.0 : 153 110 161 280 273 158 210 80 275 261 45 148
25> Time 1903.0 Idle Time 50.0 : 143 272 254 41 38 227 3
26> Time 1947.0 Idle Time 6.0 : 249 248 164 284 265 205
27> Time 1935.0 Idle Time 18.0 : 14 33 77 31 245 274 251
28> Time 1926.0 Idle Time 27.0 : 175 213 271 82 62 39 36 172 266

6.14 Problem Scholl 297 (Cycle Time = 1935) (ต่อ)

Result

29> Time 1945.0 Idle Time 8.0 : 139 13 19 51 169 134 259 15 183 112
30> Time 1884.0 Idle Time 69.0 : 18 193 136 21 246 194
31> Time 1895.0 Idle Time 58.0 : 152 186 113 137 76 53 70 29 142 25 24
32> Time 1947.0 Idle Time 6.0 : 37 107 28 185 140 23 129 86
33> Time 1919.0 Idle Time 34.0 : 102 93 16 98 122 9 198 109 219
34> Time 1937.0 Idle Time 16.0 : 218 116 32 84 83
35> Time 1883.0 Idle Time 70.0 : 91 217 216 96 111
36> Time 1914.0 Idle Time 39.0 : 270 211 85 92 258 11 215 156
37> Time 422.0 Idle Time 1531.0 : 209 75
Idle Time All : 2606.0 E=96.39 %

6.15 Problem Scholl 297 (Cycle Time = 2049)

Result

1> Time 1987.0 Idle Time 62.0 : 1 2 54 296 3
2> Time 2040.0 Idle Time 9.0 : 27 4 94 86 56 22 221 247 101 259 58 50
3> Time 1964.0 Idle Time 85.0 : 83 25 40 24 120 84 48 266 90 109 5
4> Time 2047.0 Idle Time 2.0 : 150 75 271 106 76 112 95 64 74 115
5> Time 2046.0 Idle Time 3.0 : 70 230 67 123 117 72 57 100 114 108 104
6> Time 2045.0 Idle Time 4.0 : 149 130 289 257 127 99 157 103 88
7> Time 2020.0 Idle Time 29.0 : 144 160 129 122 232 89 148 82
8> Time 2032.0 Idle Time 17.0 : 166 159 141 227 253 260 87
9> Time 1987.0 Idle Time 62.0 : 170 163 274 270 147 236 225 52 139
10> Time 2010.0 Idle Time 39.0 : 174 282 239 47 256 263 251 152 142 133
11> Time 1993.0 Idle Time 56.0 : 178 43 240 241 237 171 164 135
12> Time 2029.0 Idle Time 20.0 : 288 39 238 243 53 242 137 167 269 233
13> Time 1989.0 Idle Time 60.0 : 78 246 244 255 262 162

14> Time 2029.0 Idle Time 20.0 : 151 140 124 79 192 285 80 261 279 49
15> Time 2042.0 Idle Time 7.0 : 46 138 191 201 254 85 229 110 268
16> Time 2044.0 Idle Time 5.0 : 181 31 35 42 93 249 284 92 145 292 267 6 51
17> Time 2012.0 Idle Time 37.0 : 185 177 189 188 176 182 186 34 190 98 55
18> Time 2019.0 Idle Time 30.0 : 183 165 172 168 8 10 273 30 102 173
19> Time 2003.0 Idle Time 46.0 : 175 23 11 13 19 14 60 18 153
20> Time 2024.0 Idle Time 25.0 : 7 248 17 200
21> Time 2021.0 Idle Time 28.0 : 297 125 245 68 116
22> Time 2029.0 Idle Time 20.0 : 295 278 128
23> Time 2005.0 Idle Time 44.0 : 61 97 283 28 44 119 45 91 134
24> Time 2017.0 Idle Time 32.0 : 161 65 37 21 59 81 105 126 107 202
25> Time 1997.0 Idle Time 52.0 : 212 207 204 210 77 41 118 113 194 213 199
26> Time 2004.0 Idle Time 45.0 : 198 214 205 208 203 33 206 211 38
27> Time 2014.0 Idle Time 35.0 : 197 184 223 294 15 169 209 277 73
28> Time 2035.0 Idle Time 14.0 : 154 155 131 195 235 71 250 215 20
29> Time 2013.0 Idle Time 36.0 : 179 180 234 216 287 146 276 286
30> Time 1986.0 Idle Time 63.0 : 252 217 218 281 291
31> Time 2014.0 Idle Time 35.0 : 193 258 219 293
32> Time 1927.0 Idle Time 122.0 : 187 132 12 265 220 222 9 26 272 290 231 158
196
33> Time 2032.0 Idle Time 17.0 : 16 36 69 63 224 275 228 280
34> Time 2037.0 Idle Time 12.0 : 143 66 111 32 62 156 226
35> Time 1162.0 Idle Time 887.0 : 29 121 136 96 264
Idle Time All : 2060.0 E=97.13 %

6.16 Problem Scholl 297 (Cycle Time = 2111)

Result

1> Time 1855.0 Idle Time 256.0 : 1 296 297
2> Time 2071.0 Idle Time 40.0 : 294 295 54 30
3> Time 2057.0 Idle Time 54.0 : 284 286 285 235 237 249 181 180 176 58 50 53
34
4> Time 2084.0 Idle Time 27.0 : 233 238 231 178 197 182 186 177 188 189 179
196 195
5> Time 2080.0 Idle Time 31.0 : 234 228 202 200 199 194
6> Time 2101.0 Idle Time 10.0 : 254 214 256 201 140 138 229 203 31
7> Time 2107.0 Idle Time 4.0 : 27 263 205 236 232 206 208
8> Time 2094.0 Idle Time 17.0 : 247 279 4 48 105 134 86 111 94 83 90 137
9> Time 2101.0 Idle Time 10.0 : 101 165 109 168 132 135 49 212 136 5
10> Time 2080.0 Idle Time 31.0 : 97 76 96 153 119 114 130 74 131
11> Time 2100.0 Idle Time 11.0 : 73 84 126 125 123 69 128 129

6.16 Problem Scholl 297 (Cycle Time = 2111) (ต่อ)

Result

12> Time 2077.0 Idle Time 34.0 : 108 78 65 118 117 148 79 257 146 124 77 198

13> Time 2087.0 Idle Time 24.0 : 175 187 104 80 85 66 141 113 107 193 72

14> Time 2092.0 Idle Time 19.0 : 248 100 87 81 82 151 163 145

15> Time 2111.0 Idle Time 0.0 : 110 147 51 245 56 57 246

16> Time 2076.0 Idle Time 35.0 : 162 213 251 174 127 190 22 243 156 45 184

17> Time 2096.0 Idle Time 15.0 : 167 164 173 207 170 288 287 25 24 41 29

18> Time 2100.0 Idle Time 11.0 : 210 92 171 62 121 46 33 37 270

19> Time 2110.0 Idle Time 1.0 : 289 95 99 191 144 75 149 42 116

20> Time 2086.0 Idle Time 25.0 : 230 64 252 154 155 70 67 139 142

21> Time 2076.0 Idle Time 35.0 : 271 266 227 103 35 38 59 250

22> Time 2103.0 Idle Time 8.0 : 143 183 93 292 52 160 102 158 61 211 68 88

23> Time 2081.0 Idle Time 30.0 : 89 122 226 221 47 71 239 261 26 209 159 60

24> Time 2097.0 Idle Time 14.0 : 224 222 220 223 43 268 269 255 240

25> Time 2109.0 Idle Time 2.0 : 219 218 217 216 28 3 241 253 262 63 120 112

26> Time 2098.0 Idle Time 13.0 : 44 40 32 106 242 115 150 274

27> Time 2069.0 Idle Time 42.0 : 283 277 36 55 157 161 282 152 6

28> Time 2098.0 Idle Time 13.0 : 21 20 15 7 12 9 13

29> Time 2103.0 Idle Time 8.0 : 166 16 18 11 14 172

30> Time 2106.0 Idle Time 5.0 : 204 98 23 258 133 2 19 8 17

31> Time 2093.0 Idle Time 18.0 : 259 39 265 272 293

32> Time 2083.0 Idle Time 28.0 : 278 281 291 275 290 280 244

33> Time 2001.0 Idle Time 110.0 : 185 225 276 264 10 169 192 215

34> Time 973.0 Idle Time 1138.0 : 91 260 267 273

Idle Time All : 2119.0 E=97.05 %

6.17 Problem Scholl 297 (Cycle Time = 2177)

Result

1> Time 2014.0 Idle Time 163.0 : 1 297 294 30 285 289

2> Time 2157.0 Idle Time 20.0 : 26 34 31 179 4 3 230 238 234 259 256 22

3> Time 2149.0 Idle Time 28.0 : 24 232 27 83 247 105 29 33 109 94

4> Time 2146.0 Idle Time 31.0 : 110 240 214 44 120 241

5> Time 2172.0 Idle Time 5.0 : 237 233 25 263 38 121 40 49 266

6> Time 2138.0 Idle Time 39.0 : 270 119 90 128 84 213 95 41

7> Time 2156.0 Idle Time 21.0 : 130 125 210 215 216 45 114 150 239 227 181 78 231

8> Time 2177.0 Idle Time 0.0 : 123 131 207 188 225 177 180 187 186 108 132 161 72 149 178

9> Time 2169.0 Idle Time 8.0 : 204 173 174 217 192 147 228 160 79

10> Time 2171.0 Idle Time 6.0 : 218 219 220 175 117 50 115 205 53 2 223 80 196 279 222
11> Time 2152.0 Idle Time 25.0 : 257 224 112 58 264 106 101 74 242 253
12> Time 2159.0 Idle Time 18.0 : 268 69 293 260 170 145
13> Time 2147.0 Idle Time 30.0 : 65 61 283 261 254 267 255 155 249 291 290 197 195 85
14> Time 2123.0 Idle Time 54.0 : 269 262 280 190 203 198 127 212 148 146 278
15> Time 2165.0 Idle Time 12.0 : 156 275 193 159 226 202 200 287 182 191 168 199 56
16> Time 2140.0 Idle Time 37.0 : 194 165 138 46 153 51 185 158 251 171
17> Time 2120.0 Idle Time 57.0 : 143 81 167 55 87 59 164 152 135 250 183 92 60
18> Time 2137.0 Idle Time 40.0 : 142 100 88 77 37 111 176
19> Time 2139.0 Idle Time 38.0 : 116 71 28 32 68 91 140
20> Time 2167.0 Idle Time 10.0 : 122 57 36 139 39 282 243
21> Time 2164.0 Idle Time 13.0 : 245 62 136 43 63 246 208
22> Time 2158.0 Idle Time 19.0 : 47 248 133 52
23> Time 2156.0 Idle Time 21.0 : 23 144 162 288 277 157 124 221 129
24> Time 2176.0 Idle Time 1.0 : 19 20 141 8 12 17 14 151 6 163
25> Time 2164.0 Idle Time 13.0 : 11 10 7 15 9 18 16 201
26> Time 2149.0 Idle Time 28.0 : 13 82 64 274 48 271 236 169
27> Time 2158.0 Idle Time 19.0 : 96 137 103 295 97 98
28> Time 2164.0 Idle Time 13.0 : 102 93 252 99 5 76 272 281 284 104 86 134 42
29> Time 2148.0 Idle Time 29.0 : 258 107 265 276 244 286 211
30> Time 2153.0 Idle Time 24.0 : 113 209 118 235 229 172 273 184 67
31> Time 2165.0 Idle Time 12.0 : 70 296 189 54 126
32> Time 2055.0 Idle Time 122.0 : 89 206 292 166 21 154
33> Time 947.0 Idle Time 1230.0 : 35 66 73 75
Idle Time All : 2186.0 E=96.96 %

6.18 Problem Scholl 297 (Cycle Time = 2247)

Result

1> Time 2125.0 Idle Time 122.0 : 1 296 297 2
2> Time 2234.0 Idle Time 13.0 : 54 50 294 4 3 247 295
3> Time 2192.0 Idle Time 55.0 : 58 26 285 289 111 94 221 22 27
4> Time 2226.0 Idle Time 21.0 : 53 293 283 162 110 286 105
5> Time 2244.0 Idle Time 3.0 : 290 287 292 291 288 49 167 235 279 239 31
6> Time 2233.0 Idle Time 14.0 : 281 108 104 100 174 171 277 170 229
7> Time 2244.0 Idle Time 3.0 : 87 89 178 64 24 86 274 109 103 101
8> Time 2224.0 Idle Time 23.0 : 114 88 238 271 59 29 195 237 95 97 164
9> Time 2229.0 Idle Time 18.0 : 186 185 189 203 205 233 231 70 73 75 188

10> Time 2239.0 Idle Time 8.0 : 206 207 212 228 67 90 74 68 60 56
11> Time 2212.0 Idle Time 35.0 : 34 72 214 208 83 63 61 115 169 234 5
12> Time 2215.0 Idle Time 32.0 : 266 35 78 79 80 143 85 215 140 125 65 256 6
13> Time 2232.0 Idle Time 15.0 : 92 98 216 217 130 123 145 147 250 9 8
14> Time 2214.0 Idle Time 33.0 : 129 144 154 131 146 148 126 12 14
15> Time 2245.0 Idle Time 2.0 : 42 122 141 118 159 156 23 17 10 11
16> Time 2242.0 Idle Time 5.0 : 46 218 128 152 55 187 16
17> Time 2234.0 Idle Time 13.0 : 82 219 117 166 133 113 163
18> Time 2245.0 Idle Time 2.0 : 257 48 157 220 158 222 112 106 160 40 135 107
149 263 251 155
19> Time 2213.0 Idle Time 34.0 : 127 52 253 179 47 119 224 278 132
20> Time 2202.0 Idle Time 45.0 : 139 260 267 43 284 240
21> Time 2229.0 Idle Time 18.0 : 276 151 134 249 248 243 241 273
22> Time 2232.0 Idle Time 15.0 : 255 245 246 244 242 204
23> Time 2244.0 Idle Time 3.0 : 116 51 39 45 261 262 226 15
24> Time 2220.0 Idle Time 27.0 : 30 25 138 136 36 32 44 84 172
25> Time 2217.0 Idle Time 30.0 : 230 181 180 175 252 191 81 258 265 91 96 223
26> Time 2211.0 Idle Time 36.0 : 196 177 232 183 173 227 225 272 176 197 182
99 124 184
27> Time 2172.0 Idle Time 75.0 : 168 165 57 121 76 275 93 268 194 270
28> Time 2165.0 Idle Time 82.0 : 153 120 200 192 161 71 62 137 280 282 199
190
29> Time 2224.0 Idle Time 23.0 : 264 150 142 201 77 66 210 18 236
30> Time 2241.0 Idle Time 6.0 : 33 202 213 37 254 259 102 19 13
31> Time 2098.0 Idle Time 149.0 : 38 211 209 193 69 198 269 21
32> Time 958.0 Idle Time 1289.0 : 41 28 20 7
Idle Time All : 2249.0 E=96.87 %

6.19 Problem Scholl 297 (Cycle Time = 2322)

Result

1> Time 2259.0 Idle Time 63.0 : 1 294 2 286 278 3 297 289
2> Time 2304.0 Idle Time 18.0 : 285 284 283 247 4 249 48 293 221
3> Time 2320.0 Idle Time 2.0 : 27 105 248 287 290
4> Time 2261.0 Idle Time 61.0 : 235 134 237 22 246 245 291 174 30
5> Time 2300.0 Idle Time 22.0 : 229 195 188 190 178 244 34
6> Time 2319.0 Idle Time 3.0 : 186 24 181 196 202 197 183 177 184 200 281 255
199 262 233
7> Time 2308.0 Idle Time 14.0 : 238 94 191 240 261 292 268 269 231
8> Time 2297.0 Idle Time 25.0 : 236 234 40 109 101 96 214 170
9> Time 2306.0 Idle Time 16.0 : 83 84 97 212 95 106 44 280 91
10> Time 2321.0 Idle Time 1.0 : 256 74 241 75 49 73 171 86 65 70 67 69 66

11> Time 2301.0 Idle Time 21.0 : 120 53 250 187 167 93 35 98 58 61 63 62
12> Time 2300.0 Idle Time 22.0 : 127 115 130 129 126 124 150 203 259 42 64
13> Time 2299.0 Idle Time 23.0 : 128 131 121 207 208 118 141 266
14> Time 2318.0 Idle Time 4.0 : 77 125 132 182 210 213 198 113 206 230 92 173
15> Time 2320.0 Idle Time 2.0 : 279 205 193 71 180 211 54 85 79 76 90 168 29
112
16> Time 2314.0 Idle Time 8.0 : 239 175 135 172 242 209 201 50 102 46 263 161
33 37 123
17> Time 2318.0 Idle Time 4.0 : 192 119 270 51 154 146 149 155 145 158 147 81
39
18> Time 2277.0 Idle Time 45.0 : 114 100 87 72 156 160 228 32 36
19> Time 2289.0 Idle Time 33.0 : 104 162 80 189 148 253 144 89 226 164 277
20> Time 2318.0 Idle Time 4.0 : 232 273 267 260 159 82 139 68 224 152 222
21> Time 2301.0 Idle Time 21.0 : 185 108 31 26 227 25 60 142 220 169 28
22> Time 2310.0 Idle Time 12.0 : 204 225 103 7 20 19 18 23 78 143 153
23> Time 2315.0 Idle Time 7.0 : 6 12 10 14 11 17 21 8 176 15
24> Time 2316.0 Idle Time 6.0 : 166 136 52 110 59 88 165 251
25> Time 2289.0 Idle Time 33.0 : 271 276 163 99 111 107 38 117 219
26> Time 2320.0 Idle Time 2.0 : 264 257 218 275 116 41 157 45
27> Time 2291.0 Idle Time 31.0 : 217 272 265 258 57 252 133
28> Time 2292.0 Idle Time 30.0 : 215 216 194 223 56 151 179 5 13 137 282
29> Time 2296.0 Idle Time 26.0 : 254 274 295 140 288
30> Time 2286.0 Idle Time 36.0 : 243 296 138 122 9 43
31> Time 590.0 Idle Time 1732.0 : 16 47 55
Idle Time All : 2327.0 E=96.77 %

6.20 Problem Scholl 297 (Cycle Time = 2402)

Result

1> Time 2394.0 Idle Time 8.0 : 1 297 2 3 295 181 196
2> Time 2361.0 Idle Time 41.0 : 4 195 188 189 187 186 175 180 252 178 183 177
197 182
3> Time 2383.0 Idle Time 19.0 : 83 109 101 203 190 205 172 94 174 287 40 258
176
4> Time 2381.0 Idle Time 21.0 : 259 90 31 91 207 212 171 84 198 153 22 165
168 97
5> Time 2350.0 Idle Time 52.0 : 204 250 256 214 213 266 215 185 106 143
6> Time 2397.0 Idle Time 5.0 : 138 293 140 75 200 24 251
7> Time 2367.0 Idle Time 35.0 : 283 46 282 291 290 281 276 280 292 29 37
8> Time 2354.0 Idle Time 48.0 : 42 191 273 277 108 119 121 33 38 275
9> Time 2346.0 Idle Time 56.0 : 35 267 104 103 44 41 128
10> Time 2366.0 Idle Time 36.0 : 294 5 95 230 130 100 110 170 271 216 88 51

11> Time 2381.0 Idle Time 21.0 : 289 278 284 286 249 135 133 162 264 123 147 55 81 144
12> Time 2362.0 Idle Time 40.0 : 279 232 227 132 131 117 257 125 158 154 59
13> Time 2389.0 Idle Time 13.0 : 239 240 85 80 169 268 64 72 78
14> Time 2401.0 Idle Time 1.0 : 237 238 243 241 92 242 233 211 254 73 255 269 231 261
15> Time 2368.0 Idle Time 34.0 : 98 102 107 79 112 285 217 209 126 6 10 272
16> Time 2396.0 Idle Time 6.0 : 93 137 9 221 296 45 113
17> Time 2400.0 Idle Time 2.0 : 26 247 115 25 15 14 20 229 50 263
18> Time 2372.0 Idle Time 30.0 : 260 288 245 167 21 19 8 235 270 99 120
19> Time 2396.0 Idle Time 6.0 : 74 218 146 262 208 129 236 184 48 164 149
20> Time 2374.0 Idle Time 28.0 : 34 52 69 219 122 194 253 152 70 202 148 160
21> Time 2385.0 Idle Time 17.0 : 71 57 62 47 67 220 222 139 136 199 66 201 192
22> Time 2400.0 Idle Time 2.0 : 248 63 224 127 226
23> Time 2388.0 Idle Time 14.0 : 86 77 114 105 193 87 159 206 89 39 157
24> Time 2368.0 Idle Time 34.0 : 244 23 145 28 36 274 141
25> Time 2374.0 Idle Time 28.0 : 76 68 7 17 16 18 134 56 265 151 156 124
26> Time 2386.0 Idle Time 16.0 : 30 61 60 223 246 111 210
27> Time 2363.0 Idle Time 39.0 : 118 53 49 27 150 225 163 166
28> Time 2321.0 Idle Time 81.0 : 179 82 96 65 173 12 13 58 155 43
29> Time 2297.0 Idle Time 105.0 : 11 54 234 161 116 142 32
30> Time 835.0 Idle Time 1567.0 : 228
Idle Time All : 2405.0 E=96.66 %

6.21 Problem Scholl 297 (Cycle Time = 2488)

Result

1> Time 2480.0 Idle Time 8.0 : 1 297 293 290 292 280
2> Time 2421.0 Idle Time 67.0 : 283 277 282 108 291 287 275 30 34
3> Time 2483.0 Idle Time 5.0 : 114 281 115 174 26 171 169 170 178 181 100 35 4 221 143 104
4> Time 2485.0 Idle Time 3.0 : 197 187 188 179 88 177 183 176 22 42 56 61 140 200 182 48 52
5> Time 2460.0 Idle Time 28.0 : 186 190 82 25 24 133 184 65 180 252 47
6> Time 2467.0 Idle Time 21.0 : 166 156 195 29 33 44 69 175
7> Time 2440.0 Idle Time 48.0 : 160 155 146 158 229 203 163 49 37 131 132
8> Time 2452.0 Idle Time 36.0 : 130 128 149 236 206 151 232 28 94
9> Time 2457.0 Idle Time 31.0 : 125 144 230 78 129 80 227 72 225 101 289
10> Time 2485.0 Idle Time 3.0 : 271 266 154 79 85 92 192 95 96 106 91 191
11> Time 2464.0 Idle Time 24.0 : 264 124 257 201 202 185 46 90 23 84
12> Time 2488.0 Idle Time 0.0 : 268 117 261 121 51 15 21 20 87 77

13> Time 2486.0 Idle Time 2.0 : 255 147 64 27 83 81 7 13 14 12 76 112
14> Time 2477.0 Idle Time 11.0 : 262 244 242 58 54 193 122 75 235
15> Time 2478.0 Idle Time 10.0 : 246 243 241 237 50 196 116 70 245 204
16> Time 2488.0 Idle Time 0.0 : 240 233 250 207 239 173 231 139 67 63
17> Time 2488.0 Idle Time 0.0 : 256 136 66 11 10 19 168 248
18> Time 2439.0 Idle Time 49.0 : 263 126 161 208 109 62 8 17 32 249 210 18
19> Time 2443.0 Idle Time 45.0 : 150 270 118 189 5 142 198 59 214 73 213 55
20> Time 2480.0 Idle Time 8.0 : 238 113 107 194 89 138 152 123 16 215 211 99
272 253
21> Time 2482.0 Idle Time 6.0 : 102 164 260 157 165 148 199 265 71 209 285 45
212
22> Time 2459.0 Idle Time 29.0 : 127 105 111 273 267 36 41
23> Time 2412.0 Idle Time 76.0 : 274 110 6 9 296
24> Time 2456.0 Idle Time 32.0 : 276 278 86 145 228 295
25> Time 2473.0 Idle Time 15.0 : 103 2 134 93 234 141 259 226 224 222 220 219
218 223
26> Time 2447.0 Idle Time 41.0 : 269 120 98 217 119 294 153 167 288
27> Time 2468.0 Idle Time 20.0 : 3 286 159 74 137 247 216 97 43 251 53 162
172
28> Time 2270.0 Idle Time 218.0 : 205 31 279 258 39 68 284 60 40 135 57
29> Time 827.0 Idle Time 1661.0 : 38 254
Idle Time All : 2497.0 E=96.54 %

6.22 Problem Scholl 297 (Cycle Time = 2580)

Result

1> Time 2255.0 Idle Time 325.0 : 1 2 297 3 296
2> Time 2579.0 Idle Time 1.0 : 30 4 86 247 27 54 278 22 83 93 98 26 221 58
3> Time 2542.0 Idle Time 38.0 : 34 56 283 109 50 49 274 25 59 53 90 64 92
4> Time 2532.0 Idle Time 48.0 : 44 29 100 37 55 282 39 270 95 85 79 80
5> Time 2535.0 Idle Time 45.0 : 271 46 51 33 277 81 87 99 89 104 36 43 263
6> Time 2560.0 Idle Time 20.0 : 266 42 82 38 32 47 103 138 61 72 251 250 256
7> Time 2549.0 Idle Time 31.0 : 121 140 200 191 52 65 24 45 28 230 66 204 69
94
8> Time 2554.0 Idle Time 26.0 : 76 74 71 57 62 23 17 16 11 207 212
9> Time 2523.0 Idle Time 57.0 : 84 75 97 67 73 68 7 8 10 13 63
10> Time 2543.0 Idle Time 37.0 : 77 70 91 137 14 227 9 210 134
11> Time 2556.0 Idle Time 24.0 : 96 143 135 133 132 187 195 203 225 186 208
198 206 136
12> Time 2576.0 Idle Time 4.0 : 153 169 174 178 123 124 130 126 170 185 190
193 146 188 149 131

13> Time 2536.0 Idle Time 44.0 : 117 165 147 118 129 141 151 113 145 122 144 158 257
14> Time 2575.0 Idle Time 5.0 : 41 108 112 107 181 182 163 160 155 148 6 202 12 21 78
15> Time 2546.0 Idle Time 34.0 : 106 114 196 177 184 197 180 159 253 179 201 192 252
16> Time 2563.0 Idle Time 17.0 : 60 260 139 267 293 258
17> Time 2523.0 Idle Time 57.0 : 35 88 142 152 273 291 111 292 290
18> Time 2573.0 Idle Time 7.0 : 5 259 101 194 162 164 120 281 280 275 272 167 15 176
19> Time 2570.0 Idle Time 10.0 : 232 223 150 161 214 116 154 213 168
20> Time 2558.0 Idle Time 22.0 : 40 236 199 18 215 216 234 211 189
21> Time 2523.0 Idle Time 57.0 : 128 238 217 218 209 175 265
22> Time 2547.0 Idle Time 33.0 : 285 229 127 287 288 172
23> Time 2496.0 Idle Time 84.0 : 102 157 20 295 235 239 269 173 31
24> Time 2572.0 Idle Time 8.0 : 171 156 110 279 261 244 255 262 115 242 237 233 231
25> Time 2542.0 Idle Time 38.0 : 249 254 243 268 246 241 105 228 226 224 222
26> Time 2532.0 Idle Time 48.0 : 294 284 248 245 264 220
27> Time 2573.0 Idle Time 7.0 : 166 125 19 205 289 240
28> Time 1122.0 Idle Time 1458.0 : 183 119 276 286 48 219
Idle Time All : 2585.0 E=96.42 %

6.23 Problem Scholl 297 (Cycle Time = 2680)

Result

1> Time 2634.0 Idle Time 46.0 : 1 294 296 285 297
2> Time 2652.0 Idle Time 28.0 : 284 286 279 293 288 290 249 289 292 237 235
3> Time 2658.0 Idle Time 22.0 : 283 239 108 240 241 233 229 287
4> Time 2679.0 Idle Time 1.0 : 277 230 291 271 227 104 100 242 2 264 280 114 231
5> Time 2647.0 Idle Time 33.0 : 4 56 3 109 120 150 161 5 83 90 266 225 119 88 99 40 94 87 257
6> Time 2652.0 Idle Time 28.0 : 247 22 26 105 221 27 24 154 6 95 44 86
7> Time 2649.0 Idle Time 31.0 : 8 10 13 18 144 11 29 243 14 25 49
8> Time 2628.0 Idle Time 52.0 : 248 130 16 127 53
9> Time 2628.0 Idle Time 52.0 : 20 123 148 145 155 129 146 37 82 101 157 174 93 61
10> Time 2618.0 Idle Time 62.0 : 110 159 156 98 102 171 65 39 69 36 28 97 149
11> Time 2656.0 Idle Time 24.0 : 9 111 166 96 167 75
12> Time 2640.0 Idle Time 40.0 : 295 141 116 163 43 275 70
13> Time 2676.0 Idle Time 4.0 : 122 254 261 246 23 121 74
14> Time 2629.0 Idle Time 51.0 : 12 269 255 21 47 66 89 236 7 139

15> Time 2638.0 Idle Time 42.0 : 259 260 178 19 117 62 57 76 272 58 253 52
16> Time 2678.0 Idle Time 2.0 : 278 273 267 64 164 223 262 71 63 59 158 48
142 54 124
17> Time 2671.0 Idle Time 9.0 : 125 17 276 81 50 73 134 103 282 15 84 152 160
18> Time 2679.0 Idle Time 1.0 : 78 79 80 85 192 162 68 228 115 181 32 30 182
19> Time 2629.0 Idle Time 51.0 : 179 175 180 202 153 165 196 176 252 201 183
177 184
20> Time 2658.0 Idle Time 22.0 : 172 251 143 204 197 200 250 191 207 258 210
199 213 205 168 193 173 194
21> Time 2647.0 Idle Time 33.0 : 281 256 208 211 214 138 232 209 147 265 106
188 186
22> Time 2613.0 Idle Time 67.0 : 151 72 46 42 215 234 203 206 238 189 190 187
23> Time 2674.0 Idle Time 6.0 : 268 169 34 216 217 218 212 219 244
24> Time 2637.0 Idle Time 43.0 : 128 263 51 55 140 91 35 220 245
25> Time 2669.0 Idle Time 11.0 : 126 92 274 31 170 270 33 137 135 112 67 198
131 222 224 226 195
26> Time 2647.0 Idle Time 33.0 : 133 38 41 132 118 113 107 136 77 60
27> Time 769.0 Idle Time 1911.0 : 185 45
Idle Time All : 2705.0 E=96.26 %

6.24 Problem Scholl 297 (Cycle Time = 2787)

Result

1> Time 2782.0 Idle Time 5.0 : 1 2 296 297 294 3
2> Time 2765.0 Idle Time 22.0 : 289 285 30 286 295 4 105 86 34 54
3> Time 2777.0 Idle Time 10.0 : 230 26 279 235 237 181 184 188 233 194 48 5
27 110 239 195 31
4> Time 2769.0 Idle Time 18.0 : 232 180 182 186 190 189 197 199 205 150 56
109 183 94 50 120 221
5> Time 2774.0 Idle Time 13.0 : 252 202 200 204 207 161 210 60 58 101 95 22
90 119 178
6> Time 2777.0 Idle Time 10.0 : 191 177 74 97 75 84 77 76 106 258
7> Time 2729.0 Idle Time 58.0 : 138 140 250 64 59 53 173 71 70 25 24 46 115
137 175
8> Time 2780.0 Idle Time 7.0 : 271 125 166 78 256 79 72 80 55 100 104 67 108
227
9> Time 2754.0 Idle Time 33.0 : 264 257 117 274 277 130 159 270 85 148 103
155 156
10> Time 2764.0 Idle Time 23.0 : 123 126 124 146 269 129 283 149 118 141 261
147 89 87 122 292 255
11> Time 2767.0 Idle Time 20.0 : 266 128 282 151 99 29 116 170

12> Time 2783.0 Idle Time 4.0 : 145 160 213 134 51 96 91 133 33 127 196 163 265
13> Time 2780.0 Idle Time 7.0 : 214 211 82 136 234 93 247 244 272
14> Time 2776.0 Idle Time 11.0 : 240 68 98 102 246 162 131 172
15> Time 2753.0 Idle Time 34.0 : 243 215 241 208 242 248 249 167 132 164 171
16> Time 2779.0 Idle Time 8.0 : 92 216 268 187 114 45 157 44 6 192 231 81 144
17> Time 2767.0 Idle Time 20.0 : 61 113 69 217 253 38 41 9 201 218
18> Time 2742.0 Idle Time 45.0 : 65 66 260 14 20 11 15 19 12 23 18 112
19> Time 2765.0 Idle Time 22.0 : 143 62 37 154 236 267 40 121 176 28 152
20> Time 2781.0 Idle Time 6.0 : 153 165 273 185 49 17 63 32 36 142
21> Time 2757.0 Idle Time 30.0 : 276 281 57 21 284 209 7 35 223 39
22> Time 2780.0 Idle Time 7.0 : 291 293 287 206 263 135 203
23> Time 2787.0 Idle Time 0.0 : 193 198 288 238 158 111 228 219 275
24> Time 2737.0 Idle Time 50.0 : 254 10 16 220 226 42 52 212 174 169 225 290 224 280 43 251 73 83
25> Time 2749.0 Idle Time 38.0 : 222 229 47 278 259 88 107 8 13 139 245
26> Time 481.0 Idle Time 2306.0 : 179 262 168
Idle Time All : 2807.0 E=96.13 %

ประวัติผู้วิจัย

ชื่อ-สกุล	นายพูนธนะ ศรีสระคู
ประวัติการศึกษา	พ.ศ.2544 มัธยมศึกษาตอนต้น โรงเรียนร้อยเอ็ดวิทยาลัย
	พ.ศ.2547 มัธยมศึกษาตอนปลาย โรงเรียนร้อยเอ็ดวิทยาลัย
	พ.ศ.2551 วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยอุบลราชธานี
	พ.ศ.2553 วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่
ด้านการทำงาน	พ.ศ. 2555 – ปัจจุบัน อาจารย์ประจำ (พนักงานมหาวิทยาลัย) สาขาวิศวกรรมการจัดการอุตสาหกรรม คณะเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยราชภัฏบุรีรัมย์
ที่อยู่ที่ทำงาน	สาขาวิศวกรรมการจัดการอุตสาหกรรม คณะเทคโนโลยีอุตสาหกรรม มหาวิทยาลัยราชภัฏบุรีรัมย์ 439 ถนนจิระ ตำบลในเมือง อำเภอเมือง จังหวัดบุรีรัมย์ รหัสไปรษณีย์ 31000