บทที่ 6
การประยุกต์ใช้เจนเนติกอัลกอริทึม ในการออกแบบผังโรงงานที่แผนกมีขนาดไม่เท่ากัน

เนื้อหาของบทนี้กล่าวถึงการประยุกต์ใช้เจนเนติกอัลกอริทึมในการออกแบบผังโรงงานที่แผนกมีขนาดไม่เท่ากัน ซึ่งจะกล่าวถึงขั้นตอนต่างๆของเจนเนติกอัลกอริทึม เช่น การรีโปรดักชั่น การครอสโอเวอร์ การมิวเตชั่น อย่างละเอียด รวมถึงมีการยกตัวอย่างในการประยุกต์ใช้ในการแก้ปัญหาผังโรงงานที่ใช้ในการวิจัยอย่างละเอียด

6.1 ข้อจำกัดของการออกแบบ

ในการวิจัยนี้ ได้ใช้เจนเนติกอัลกอริทึมในการออกแบบผังโรงงานที่แผนกมีขนาดไม่เท่ากัน โดยปัญหาที่นำมาวิจัยนั้น มีลักษณะต่างๆดังนี้

ทำการศึกษาเฉพาะปัญหาการจัดผังโรงงานแบบกระบวนการผลิต แบ่งเป็น 2 ประเภท

1. การออกแบบผังโรงงานแบบวัตถุประสงค์เดียว (พิจารณาข้อมูลเชิงปริมาณ)
2. การออกแบบผังโรงงานแบบหลายวัตถุประสงค์ (พิจารณาทั้งข้อมูลเชิงปริมาณและข้อมูลเชิงคุณภาพ)

<u>ข้อกำหนดของผังโรงงาน</u>

- โรงงาน มีพื้นที่ใหญ่ที่สุด ขนาด(A) 40,000 ตารางหน่วย โดยกว้าง (W) 200 หน่วย และ ยาว(L) 200 หน่วย
- มีจำนวนแผนกมากที่สุด 20 (i) แผนก และจำนวนแผนกน้อยที่สุด 2 แผนกแต่ละแผนกมีขนาด N_i ตารางเมตร (i คือลำดับของแผนก)

<u>ข้อกำหนดทั่วไป</u> :

- พื้นที่รวมของแผนกทั้งหมดต้องไม่มากกว่าพื้นที่ของโรงงาน
- พื้นที่ของแต่ละแผนกที่สร้างขึ้นนั้นต้องเท่ากับความต้องการพื้นที่สำหรับแผนกนั้นๆ
- ไม่มีแผนกใดๆ ในโรงงานใช้พื้นที่เดียวกัน หรือสร้างขึ้นซ้อนทับกัน
- พื้นที่ของแต่ละแผนกต้องติดต่อกันเป็นพื้นที่เดียว
- ไม่มีพื้นที่ส่วนหนึ่งส่วนใดของแผนกใดๆอยู่นอกพื้นที่โรงงาน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

- รูปร่างของแผนกที่สร้างขึ้นนั้นจะมีพื้นฐานมาจากการต่อกันของรูปสี่เหลี่ยมมุมฉาก

<u>ข้อกำหนดด้านรูปร่างเพิ่มเติม</u> : (เพื่อความเหมาะสมในการนำไปใช้งานจริง)

- ในแต่ละแผนกจะมี ส่วนพื้นที่ที่เล็กที่สุดของแผนกนั้นๆ ซึ่งเปรียบเสมือนเครื่องจักรที่จำเป็นในแผนก โดยพื้นที่ที่เล็กที่สุดของแผนกที่ *i* จะกำหนดเป็นรูปสี่เหลี่ยมมุมฉาก กว้าง w_i หน่วย และยาว l_i หน่วย
- แผนกที่สร้างขึ้นนั้นต้องสามารถบรรจุพื้นที่ที่เล็กที่สุดของแผนกนั้นๆได้
- แต่ละแผนกมีการกำหนดอัตราส่วนด้านกว้างต่อด้านยาวของแผนก กำหนดไว้เพื่อป้องกันการสร้างผังโรงงานที่มีรูปร่างเป็นรูปสี่เหลี่ยมผืนผ้าที่ยาวและแคบจนไม่เหมาะสมกับการใช้งานจริง
- ในกรณีของออกแบบโรงงานแบบวัตถุประสงค์เดียว จะพิจารณาถึงข้อมูลเชิงปริมาณ และสามารถกำหนดให้ข้อมูลเชิงคุณภาพเป็นข้อจำกัดเพิ่มเติมได้ โดยพิจารณาความใกล้ชิดระหว่างแผนก 2 กรณีคือ
    1. แผนกที่จำเป็นต้องตั้งอยู่ใกล้กัน ให้จัดตั้งอยู่ติดกัน (A)
    2. แผนกที่ห้ามอยู่ใกล้กัน ให้จัดตั้งอยู่ห่างกัน (X)

<u>ตัวอย่าง 6.1</u> โรงงานแบบวัตถุประสงค์เดียวมีขนาด กว้าง 15 หน่วย ยาว 20 หน่วย รวมพื้นที่ 300 ตารางหน่วย มีทั้งหมด 4 แผนก โดยมีรายละเอียดดังนี้

แผนกที่	พื้นที่ทั้งหมด	พื้นที่บังคับ(กว้าง*ยาว)	อัตราส่วนด้านกว้างต่อด้านยาว
1	100	10*10	4
2	100	5*15	4
3	50	5*5	4
4	50	3*7	4

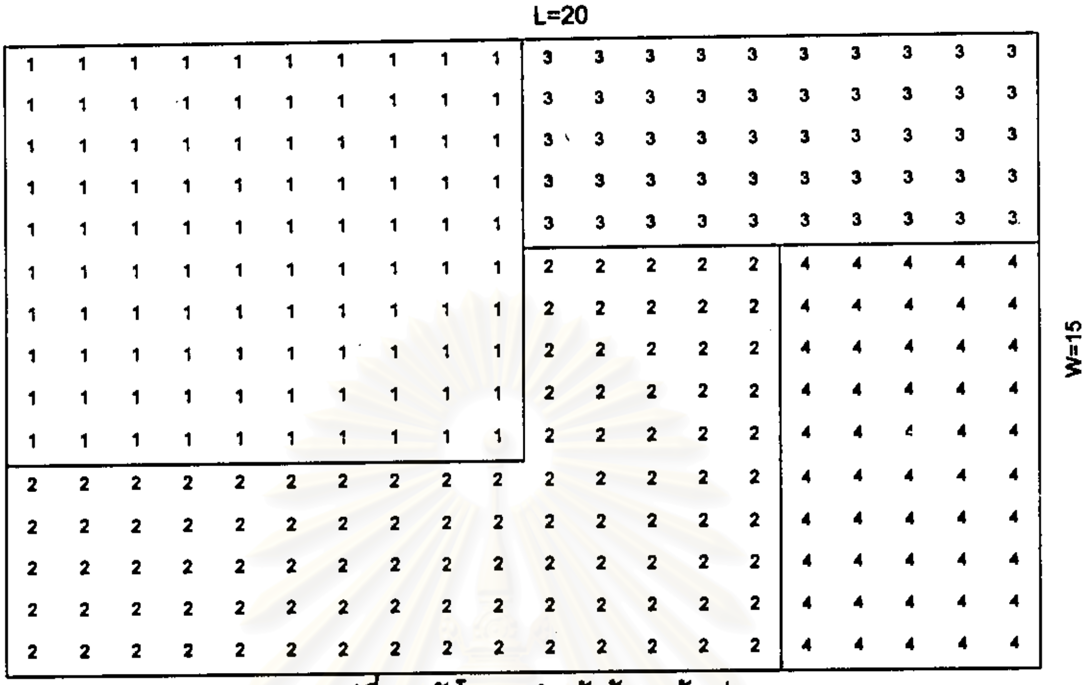
<u>ข้อจำกัดเชิงคุณภาพเพิ่มเติม</u>

แผนกที่ 1 ควรอยู่ใกล้กับแผนกที่ 3 (A)

แผนกที่ 2 ควรอยู่ใกล้กับแผนกที่ 3 (A)

แผนกที่ 1 ห้ามอยู่ใกล้แผนกที่ 4 (X)

สามารถออกแบบผังโรงงานได้ดังรูปที่6.1 โดยมีสตริงคำตอบคือ Department-Order:1-2-3-4 Band Width:10-5-5



รูปที่ 6.1 ผังโรงงานสำหรับปัญหาตัวอย่าง 6.1

6.2 แนวทางในการประยุกต์ใช้เจนเนติกอัลกอริทึม

เห็นได้ว่าสตริงของคำตอบนั้นจะประกอบด้วย 2 ส่วนคือ

1. **ลำดับการเรียงของแผนก(Department-Order)** คือลำดับของแผนกต่างๆในโรงงานที่เรียงจากจุดเริ่มต้นไปยังจุดสุดท้ายของเส้นทางของแถบ(Band)
2. **ขนาดความกว้างของแถบ(Band Width)** คือขนาดของแถบในแต่ละช่วงของโรงงาน ซึ่งแต่ละแถบนั้นอาจมีความกว้างแตกต่างกันได้ โดยผลรวมของความกว้างแถบจะต้องมีค่าเท่ากับความกว้างของโรงงาน

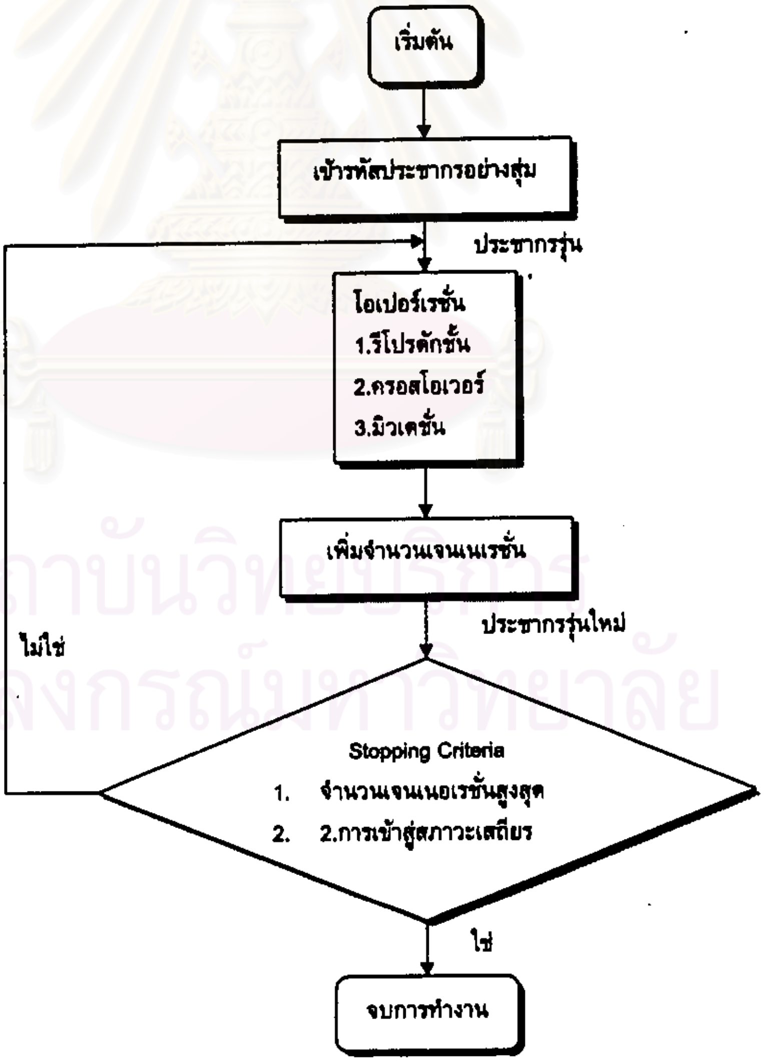
โดยในขั้นแรกจะสร้างสตริงลำดับการเรียงของแผนก(Department-Order) ขึ้นมาก่อน แล้วทำการสร้างสตริงขนาดความกว้างของแถบ(Band Width) ให้มีความสัมพันธ์กับสตริงของ Department-Order และสอดคล้องกับข้อกำจัดต่างๆของปัญหา โดยในการวิจัยนี้จะทำการสร้างสตริงขนาดความกว้างของแถบขึ้น 5 ครั้ง หากยังไม่สามารถหาสตริงขนาดความกว้างของแถบที่เหมาะสมได้ จะทำการสร้างสตริงลำดับการเรียงของแผนกใหม่ แล้วสร้างสตริงขนาดความกว้างของแถบอีกครั้ง

6.3 ขั้นตอนและวิธีการของเจนเนติกอัลกอริทึม

สิ่งที่สำคัญของ GAs คือการเข้ารหัสสตริงให้ตรงกับรูปแบบปัญหาก่อน ขั้นตอนแรกเริ่มจากทำการสุ่มประชากรเริ่มต้นซึ่งอยู่ในรูปแบบสตริงจากนั้นจึงนำประชากรนั้นผ่านโอเปอร์เรชั่นของการ Reproduction Crossover และMutationตามลำดับ เพื่อให้ได้ประชากรใหม่ โดยที่ในแต่ละขั้นตอนมีการเก็บคำตอบที่ดีที่สุดไว้ จากนั้นประชากรคำตอบก็จะผ่านกระบวนการอิลิทิส (Elitist) จะเก็บสตริงคำตอบที่ดีที่สุดของเจนเนอเรชั่นนั้นไว้ ถ้าในเจนเนอเรชั่นต่อไปได้คำตอบที่ไม่ดีกว่าคำตอบของเจนเนเรชั่นปัจจุบัน ก็จะแทนที่คำตอบที่ไม่ดีนั้นด้วยคำตอบที่ดีของเจนเนอเรชั่นปัจจุบัน ขั้นตอนสุดท้ายเป็นการตรวจสอบสภาวะการหยุดทำงาน (Stopping Criteria) โดยตรวจสอบจาก 2 กรณีคือ

1. จำนวนเจนเนอเรชั่น ถ้าจำนวนเจนเนอเรชั่น มากกว่าหรือเท่ากับจำนวนเจนเนอเรชั่นสูงสุดให้หยุดทำงาน แต่ถ้าไม่ใช่ให้ทำการคำนวณใหม่อีกต่อไป

2. การเข้าสู่สภาวะเสถียร(steady state)ของคำตอบ หากคำตอบเข้าสู่สภาวะเสถียรฺให้หยุดการทำงาน



รูปที่ 6.2 หลักการของGAs

6.3.1 การรีโปรดักชั่น (Reproduction)

กระบวนการรีโปรดักชั่น คือกระบวนการในการคัดเลือกสตริงไปยังเจนเนอเรชั่นถัดไป โดยที่สตริงที่มีค่าความเหมาะสมสูงหรือสตริงที่ให้ค่าใช้จ่ายต่ำกว่ามีโอกาสที่จะถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปได้มากกว่าสตริงที่ให้ค่าใช้จ่ายสูงกว่า ซึ่งกระบวนการนี้เลียนแบบกระบวนการคัดเลือกทางธรรมชาติ ตามทฤษฎี Survival of Fittest ของ ชาร์ล ดาร์วิน

ดังนั้นในกระบวนการรีดักชั่นจะต้องสร้างฟิตเนสฟังก์ชั่นเพื่อเป็นการกำหนดว่าสตริงคำตอบของการจัดผังโรงงานที่สร้างขึ้นควรจะถูกคัดเลือกหรือไม่ โดยยกตัวอย่างการจัดผังโรงงานเชิงปริมาณโดยพิจารณาถึงระยะทางระหว่างแผนก สตริงคำตอบที่มี Cost ที่น้อยกว่าควรมีโอกาสที่จะถูกคัดเลือกได้มากกว่าสตริงคำตอบที่มี Cost ที่มากกว่า จากฟิตเนสฟังก์ชั่นทำให้สามารถกำหนดค่าความน่าจะเป็นที่สตริงใดๆจะถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปได้

สำหรับวิธีการคัดเลือกสตริงคำตอบไปยังเจนเนอเรชั่นถัดไป คือวิธีการของวงล้อรูลเล็ต (Roulette Wheel) (Goldberg, 1989) ซึ่งจะสุ่มจำนวนครั้งที่สตริงใดๆ จะถูกคัดเลือกจริงตามความน่าจะเป็นที่สร้างขึ้นจากฟิตเนสฟังก์ชั่น

ตารางที่ 6.1 กระบวนการรีโปรดักชั่นของปัญหาการจัดผังโรงงานด้วยข้อมูลเชิงปริมาณ

สตริงเริ่มต้น (1)	$f(x) = \sum\sum V(r_{ij})d_{ij}$ (2)	$\sum f(x) - f(x_i)$ (3)	$F_i = (\sum f(x) - f(x_i)) / \sum f(x)$ (4)	ค่าเฉลี่ยของจำนวนครั้งที่ถูกคัดเลือก(5)	จำนวนครั้งที่ถูกคัดเลือกจริง(6)	สตริงใหม่ (7)	f(x) (8)
[6 0 3 9 2 7 4 1 5 8]	101.912	372.134	0.262	1.048	2	[6 0 3 9 2 7 4 1 5 8]	101.912
[6 5 8 4 1 0 9 7 3 2]	123.762	350.265	0.246	.984	1	[6 0 3 9 2 7 4 1 5 8]	101.912
[7 5 8 4 2 3 6 0 1 9]	124.176	349.871	0.246	.984	1	[6 5 8 4 1 0 9 7 3 2]	123.782
[6 5 7 4 0 2 9 8 3 1]	124.176	349.871	0.246	.984	0	[7 5 8 4 2 3 6 0 1 9]	124.176
ผลรวม $\sum f(x)$	474.046	1422.141	1.00	4	4		451.783
ค่าเฉลี่ย	118.512	355.535	.25	1	1		112.945

หมายเหตุ ค่าต่างๆ ในตาราง 6.1 เป็นเพียงตัวอย่างเพื่อความเข้าใจเรื่องวิธีการรีโปรดักชั่น ซึ่งไม่เกี่ยวข้องกับปัญหาที่จะใช้ในการวิจัย

จากตารางที่ 6.1 แสดงถึงรีโปรดักชั่นของปัญหาการจัดผังโรงงานด้วยข้อมูลเชิงปริมาณแบบวงล้อรูลเล็ท โดยกำหนดให้มีจำนวนประชากรหรือประชากรเริ่มต้นเป็น 4 สตริง

ให้คอลัมน์แรกเป็นสตริงที่สุ่มขึ้นซึ่งสามารถคำนวณ Cost ได้ตามฟิตเนสฟังก์ชั่นสมการที่ 4.1 ดังในคอลัมน์ที่ 2 สตริงที่มี Cost ที่น้อยที่สุดควรมีโอกาสที่จะอยู่รอดได้มากที่สุดจึงสร้างคอลัมน์ที่ 3 และ 4 เพื่อใช้ในการคำนวณขนาดของวงล้อรูลเล็ท ความน่าจะเป็นที่สตริงจะถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไป เนื่องจากฟิตเนสฟังก์ชั่นเป็นการหา Cost ที่ต่ำที่สุด ดังนั้นสตริงที่มี Cost ต่ำที่สุดควรจะมีโอกาสที่จะอยู่รอดต่อไปได้มากกว่าสตริงที่มี Cost สูงกว่า ดังจะเห็นได้ว่าสตริงตัวแรกมีความน่าจะเป็นในการอยู่รอด 26.2% ส่วนสตริงตัวสุดท้ายมีความน่าจะเป็นในการอยู่รอด 24.6% เราสามารถคำนวณหาจำนวนครั้งของสตริงที่จะถูกเลือกได้ดังคอลัมน์ที่ 5

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

คำนวณได้จาก $F_i / \overline{F_i}$ และคอลัมน์ที่ 6 เป็นจำนวนครั้งของโอกาสที่สตริงถูกคัดเลือกจริง จะเห็นได้ว่าสตริงตัวแรกถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปสองครั้ง สตริงตัวที่สองและสามถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปหนึ่งครั้ง และสตริงตัวสุดท้ายไม่ได้ถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปเลย เมื่อพิจารณาสตริงที่ผ่านการคัดเลือกแล้วในคอลัมน์สุดท้ายจะเห็นได้ว่า ผลรวมและค่าเฉลี่ยลดลง จากเดิมผลรวมของ Cost เป็น 474.0465 ลดลงเหลือ 451.783 และค่าเฉลี่ยลดลงจาก 118.512 เป็น 112.945

กระบวนการคัดเลือกประชากรใหม่จะใช้หลักการของวงล้อรูลเล็ท ที่มีขนาดความกว้างของสล็อท (Slot) แปรผันตามค่าความเหมาะสม การสร้างวงล้อรูลเล็ทมีวิธีการดังต่อไปนี้

- กำหนดให้สตริงแต่ละตัวเป็น str_i ซึ่งสามารถคำนวณค่าความเหมาะสมของสตริงแต่ละตัว $eval(str_i)$ หรือ *Cost* ของสตริงเป็น c_i แต่ละตัว โดยที่ $i = 1...pop$ (โดย *pop* คือจำนวนสตริงในแต่ละรุ่นประชากร)
- หาค่าความเหมาะสมโดยรวมของประชากร เนื่องจากสมการวัตถุประสงค์เป็นการหาค่าที่น้อยที่สุด ดังนั้นค่าที่น้อยที่สุดในประชากรจึงควรมีความน่าจะเป็นของการคัดเลือก (p_i) มากที่สุด ขนาดของช่องสล็อทในวงล้อรูลเล็ทสำหรับ Cost ที่มีค่าน้อยที่สุดควรมีขนาดกว้างมากที่สุด

$$CostSum = \sum_{i=1}^{pop} eval(str_i)$$

คำนวณหาค่า $CostSum - c_i$ และผลรวมของ $CostSum - c_i$ เป็น $Sum_of_(CostSum - c_i)$

- คำนวณหาค่าความน่าจะเป็นการคัดเลือก p_i ของสตริงแต่ละตัว

$$p_i = (CostSum - c_i) / Sum_of_(CostSum - c_i)$$

- คำนวณค่าความน่าจะเป็นสะสม (Cumulative Probability) ของสตริงแต่ละตัว (q_i) โดยที่ $i = 1...pop$

$$q_i = \sum_{j=1}^{i} p_i$$

กระบวนการคัดเลือกจะหมุนวงล้อรูลเล็ทเท่ากับจำนวน *pop* ครั้ง ในแต่ละครั้งจะได้สตริงสำหรับประชากรรุ่นต่อไป ดังมีรายละเอียดดังต่อไปนี้

- สร้างค่าสุ่ม (Random Number) *r* ที่มีค่าอยู่ในช่วง [0...1]

สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

- ถ้า $r < q_i$ เลือกสตริงตัวแรก ถ้าไม่ใช่ เลือกสตริงตัวที่ i โดยที่ i อยู่ระหว่าง ($2 \leq i \leq pop$) หรือ ($q_{i-1} \leq r \leq q_i$) ในบางครั้งสตริงที่ถูกคัดเลือกแล้วบางตัวบางตัวอาจถูกเลือกได้มากกว่า 1 ครั้ง

หมายเหตุ ใช้หลักการเดียวกันนี้ในการออกแบบผังโรงงานโดยใช้ข้อมูลเชิงคุณภาพ

6.3.2 กระบวนการครอสโอเวอร์(Crossover)

การครอสโอเวอร์คือกระบวนการจับคู่สตริง แล้วแลกเปลี่ยนส่วนซึ่งกันและกันเพื่อให้เกิดสตริงใหม่ขึ้น

เนื่องจากปัญหาของการจัดผังโรงงานไม่ได้ใช้การเข้ารหัสแบบเลขฐานสอง การครอสโอเวอร์แบบธรรมดา หรือการครอสโอเวอร์แบบตำแหน่งเดียวไม่สามารถนำมาใช้ได้ เนื่องจากการครอสโอเวอร์แบบตำแหน่งเดียวอาจทำให้สตริงมีลักษณะที่ไม่ถูกต้อง กล่าวคืออาจทำให้ยีนบางตำแหน่งเกิดค่าซ้ำกัน ซึ่งทำให้ไม่สามารถคำนวณหาระยะทางของแผนกในผังโรงงานนั้นๆได้ Goldberg(1989) ได้เสนอวิธีการครอสโอเวอร์แบบสองจุด เพื่อแก้ปัญหาของการครอสโอเวอร์แบบธรรมดา การครอสโอเวอร์ที่นำเสนอมี 3 วิธีคือ การครอสโอเวอร์แบบ PMX (Partially Match Crossover) การครอสโอเวอร์แบบ OX (Order Crossover) และการครอสโอเวอร์แบบ CX (Cycle Crossover) ดังมีรายละเอียดดังต่อไปนี้

6.3.2.1 การครอสโอเวอร์แบบ PMX วิธีการของ PMX อย่างแรกคือการเลือกคู่สตริงพ่อแม่ขึ้นมาอย่างสุ่ม จากนั้นทำการเลือกตำแหน่งที่จะทำการครอสโอเวอร์อย่างสุ่มที่อยู่ในช่วง [1, l - 1] โดยที่ l คือความยาวของสตริง และยีนตัวแรกคือยีนหมายเลข 1 และยีนตัวสุดท้ายคือยีนหมายเลข l ขอบเขตของการครอสโอเวอร์อยู่ในช่วงเครื่องหมาย " | "

$$p_1 = [\,9\ 8\ 4 \mid 5\ 6\ 7 \mid 1\ 3\ 2\ 0\,]$$

$$p_2 = [\,8\ 7\ 1 \mid 2\ 3\ 0 \mid 9\ 5\ 4\ 6\,]$$

ในขั้นตอนต่อไปจะทำการสลับค่าระหว่างสตริงที่อยู่ในช่วง " | " นั่นคือตำแหน่งสุ่มอยู่ในช่วง [4, 6] ของโครโมโซมลูกหลานทั้งสอง โดยที่ค่าที่อยู่นอกเครื่องหมาย " | " และเป็นค่าที่ซ้ำกันกับค่าที่อยู่ในช่วงที่กำหนดให้เป็น x

จะเห็นได้ว่าตำแหน่งที่สองของสตริง p_2 มีค่าเป็น 7 ซึ่งซ้ำกับค่าที่อยู่ในช่วง " | " ของสตริง p_1 ก่อนทำการสลับ จึงเปลี่ยนค่าที่ซ้ำให้เป็น x ก่อนและตำแหน่งอื่นๆที่มีค่าซ้ำกันก็จะเปลี่ยนเป็น # ด้วยเช่นกัน

$$O_1 = [\,9\ 8\ 4 \mid 2\ 3\ 0 \mid 1\ \#\ \#\ \#\,]$$

$$O_2 = [\,8\ x\ 1 \mid 5\ 6\ 7 \mid 9\ 5\ 4\ \#\,]$$

แล้วทำการแทนค่า (Map) ดังต่อไปนี้ 2 เป็น 5 และ 5 เป็น 2; 3 เป็น 6 และ 6 เป็น 3; 0 เป็น 7 และ 7 เป็น 0 การแทนค่าเหล่านี้ได้มาจากค่าที่อยู่ในช่วง " | " โดยพิจารณาตำแหน่งของสตริงที่ตรงกัน

สตริงที่ได้เมื่อทำการแทนที่แล้วคือ

$$O_1 = [9\ 8\ 4\ 2\ 3\ 0\ 1\ 6\ 5\ 7]$$

$$O_2 = [8\ 0\ 1\ 5\ 6\ 7\ 9\ 2\ 4\ 3]$$

6.3.2.2 การครอสโอเวอร์แบบ OX เช่นเดียวกับวิธีการของ PMX อย่างแรกคือการเลือกคู่สตริงพ่อแม่ขึ้นมาอย่างสุ่ม จากนั้นทำการเลือกตำแหน่งที่จะทำการครอสโอเวอร์อย่างสุ่ม ขอบเขตของการครอสโอเวอร์อยู่ในช่วงเครื่องหมาย " | "

$$P_1 = [8\ 9\ 4 \mid 5\ 6\ 7 \mid 1\ 3\ 2\ 0]$$

$$P_2 = [8\ 7\ 1 \mid 2\ 3\ 0 \mid 9\ 5\ 4\ 6]$$

สุ่มตำแหน่งที่จะทำการครอสโอเวอร์ได้อยู่ในช่วง [4, 6] จากนั้นเลือกคู่แม็ปที่อยู่ในช่วงการครอสโอเวอร์ถ้าค่าที่ตรงกันให้ทำเครื่องหมาย # เครื่องหมาย # หมายความว่าปล่อยตำแหน่งนั้นให้ว่าง สตริงลูกหลานจะเป็น

$$O_1 = [8\ 9\ 4 \mid 5\ 6\ 7 \mid 1\ \#\ \#\ \#]$$

$$O_2 = [8\ \#\ 1 \mid 2\ 3\ 0 \mid 9\ \#\ 4\ \#]$$

จากนั้นจะทำการเลื่อนตำแหน่ง โดยยึดถือตำแหน่งหลังช่วง " | " เป็นจุดอ้างอิง (สำหรับสตริง O_1 เป็น 1 และ สำหรับ O_2 เป็น 9) ได้สตริงลูกหลานดังต่อไปนี้

$$O_1 = [5\ 6\ 7 \mid \#\ \#\ \# \mid 1\ 8\ 9\ 4]$$

$$O_2 = [2\ 3\ 0 \mid \#\ \#\ \# \mid 9\ 4\ 8\ 1]$$

ขั้นตอนต่อไปจะเป็นการสลับตำแหน่งที่อยู่ในช่วงของการครอสโอเวอร์ของสตริงทั้งสองที่เหลือ สตริงที่ได้เมื่อทำการครอสโอเวอร์แล้วคือ

$$O_1 = [5\ 6\ 7 \mid 2\ 3\ 0 \mid 1\ 8\ 9\ 4]$$

$$O_2 = [2\ 3\ 0 \mid 5\ 6\ 7 \mid 9\ 4\ 8\ 1]$$

6.3.2.3 การครอสโอเวอร์แบบ CX ขั้นตอนแรกเลือกคู่สตริงพ่อแม่ขึ้นมาอย่างสุ่ม พิจารณาดังต่อไปนี้

$$P_1 = [2\ 0\ 3\ 4\ 8\ 5\ 9\ 1\ 7\ 6]$$

$$P_2 = [5\ 9\ 2\ 1\ 0\ 6\ 7\ 3\ 8\ 4]$$

พิจารณาที่ตำแหน่งแรกของสตริง P_1 และ P_2 คงเดิมไว้

O_1 = [2 - - - - - - - - -]

O_2 = [5 - - - - - - - - -]

ค่าตำแหน่งแรกของสตริง O_2เป็น 5 นำมาพิจารณาค่าที่เป็น 5 (ตำแหน่งที่ 6) ของสตริง O_1จะคงเดิมไว้ ส่วนค่าที่เป็น 6 (ตำแหน่งที่ 6) ของสตริง O_2จะคงเดิมไว้เช่นกัน จะได้เป็น

O_1 = [2 - - - - 5 - - - -]

O_2 = [5 - - - - 6 - - - -]

จากค่าที่เป็น 6 ของสตริง O_2 นำไปพิจารณาต่อในสตริง O_1ซึ่งจะคงเดิมไว้ ได้สตริงเป็น

O_1 = [2 - - - - 5 - - - 6]

O_2 = [5 - - - - 6 - - - 4]

ด้วยกระบวนการเดิมจะได้สตริงเป็น

O_1 = [2 - - 4 - 5 - - - 6]

O_2 = [5 - - 1 - 6 - - - 4]

↓

O_1 = [2 - - 4 - 5 - 1 - 6]

O_2 = [5 - - 1 - 6 - 3 - 4]

↓

O_1 = [2 - 3 4 - 5 - 1 - 6]

O_2 = [5 - 2 1 - 6 - 3 - 4]

กระบวนการในการเลือกตัวที่จะคงเดิมไว้จะสิ้นสุดต่อเมื่อ ค่าในตำแหน่งใดตำแหน่งหนึ่งของ O_2 มีค่าตรงกับค่าในตำแหน่งแรกของ สตริง O_1จากนั้นจะทำการสลับค่าที่ยังคงเหลืออยู่ทั้งสองสตริงแบบ ตำแหน่งต่อตำแหน่ง ได้สตริงลูกหลานเป็น

O_1 = [2 9 3 4 0 5 7 1 8 6]

O_2 = [5 0 2 1 8 6 9 3 7 4]

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

นอกเหนือจากวิธีการครอสโอเวอร์ของ Goldberg ทั้ง 3 วิธีแล้ว ในการวิจัยนี้ได้นำวิธีการครอสโอเวอร์ที่ดัดแปลงจากวิธีการของ Goldberg อีก 2 วิธีคือ

6.3.2.4 การPBX (Position-Base Crossover) พัฒนาโดย Syswerda โดยดัดแปลงมาจากวิธีการครอสโอเวอร์แบบ OX โดยเริ่มจากการเลือกสตริงพ่อแม่

P_1 = [2 0 3 4 8 5 9 1 7 6]

P_2 = [5 9 2 1 0 6 7 3 8 4]

เลือกตำแหน่งจากสตริงพ่อแม่ P_1 อย่างสุ่ม แล้วนำค่าในตำแหน่งที่เลือกของสตริงพ่อแม่ P_1 ไปใส่ในตำแหน่งเดียวกันของสตริงลูก O_1 ในที่นี้เลือกตำแหน่งที่ 4 และ 7

$$P_1 = [\,2\;0\;3\;\underline{4}\;8\;5\;\underline{9}\;1\;7\;6\,]$$

$$P_2 = [\,5\;9\;2\;1\;0\;6\;7\;3\;8\;4\,]$$

$$O_1 = [\,\#\;\#\;\#\;4\;\#\;\#\;9\;\#\;\#\;\#\;\#\,]$$

ตัดค่าที่อยู่ในตำแหน่งที่เลือกของสตริงพ่อแม่ P_1 ออกจากสตริงพ่อแม่ P_2 ดังนั้นค่าที่ตัดออกคือ 4และ9

$$P_2 = [\,5\;\#\;2\;1\;0\;6\;7\;3\;8\;\#\,]$$

นำค่าที่เหลืออยู่ในสตริงพ่อแม่ P_2 มาใส่ในสตริงลูก O_1 ตามลำดับ

$$P_2 = [\,5\;\#\;2\;1\;0\;6\;7\;3\;8\;\#\,]$$

$$\downarrow$$

$$O_1 = [\,5\;2\;1\;\underline{4}\;0\;6\;\underline{9}\;7\;3\;8\,]$$

6.3.2.5 การOBX (Order-Base Crossover) พัฒนาโดย Syswerda เช่นกัน มีความใกล้เคียงกับPBX แต่แตกต่างกันเล็กน้อย โดยมีวิธีการดังนี้

เลือกตำแหน่งอย่างสุ่มจากสตริงพ่อแม่ P_1 ในที่นี้เลือกตำแหน่งที่ 4 และ 7

$$P_1 = [\,2\;0\;3\;\underline{4}\;8\;5\;\underline{9}\;1\;7\;6\,]$$

$$P_2 = [\,5\;9\;2\;1\;0\;6\;7\;3\;8\;4\,]$$

นำค่าจากสตริงพ่อแม่ P_1 ไปใส่สตริงลูก O_1 ที่ตำแหน่งเดียวกันกับตำแหน่งของสตริงพ่อแม่ P_2 ที่มีค่าตรงกับค่าที่ถูกเลือกในสตริงพ่อแม่ P_1 โดยเรียงตามลำดับก่อนหลัง

$$P_1 = [\,2\;0\;3\;\underline{4}\;8\;5\;\underline{9}\;1\;7\;6\,]$$

$$O_1 = [\,\#\;4\;\#\;\#\;\#\;\#\;\#\;\#\;\#\;9\,]$$

$$P_2 = [\,5\;\underline{9}\;2\;1\;0\;6\;7\;3\;8\;\underline{4}\,]$$

นำค่าในตำแหน่งที่เหลือของสตริงพ่อแม่ P_2 ไปใส่ในตำแหน่งที่ว่างอยู่ของสตริงลูกตามลำดับ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

$$O_1 = [5\ \underline{4}\ 2\ 1\ 0\ 6\ 7\ 3\ 8\ \underline{9}]$$

$$\uparrow\quad \uparrow\uparrow\uparrow\uparrow\uparrow\uparrow\uparrow$$

$$P_2 = [5\ \underline{9}\ 2\ 1\ 0\ 6\ 7\ 3\ 8\ \underline{4}]$$

ในขั้นสุดท้ายจะทำการคืนสตริงเข้าสู่เมทติ้งพูล

ในการครอสโอเวอร์ จะต้องมีการกำหนดพารามิเตอร์เพื่อหาจำนวนของสตริงจากประชากรทั้งหมดที่จะเกิดการครอสโอเวอร์ ค่าพารามิเตอร์นี้คือค่าความน่าจะเป็นของการครอสโอเวอร์ (Probability of Crossover, p_c) ซึ่งทำให้ได้จำนวนประชากรที่เกิดการครอสโอเวอร์คือ p_c*pop ขั้นตอนการสุ่มประชากรที่เกิดการครอสโอเวอร์ตามค่าความน่าจะเป็นของการครอสโอเวอร์ มีดังต่อไปนี้

- สร้างค่าสุ่ม (Random Number) r ที่มีค่าอยู่ในช่วง [0...1]
- ถ้า $r < p_c$ เลือกสตริงนั้นเพื่อทำการครอสโอเวอร์ตัวแรก
- จับคู่กับสตริงอีกหนึ่งตัวเพื่อทำการครอสโอเวอร์ ตามวิธีการของการครอสโอเวอร์แต่ละแบบ

6.3.3 การมิวเตชั่น(Mutation)

กระบวนการมิวเตชั่น เป็นการเปลี่ยนค่าในบางตำแหน่งของสตริงเพื่อให้เกิดคำตอบใหม่โดยเลือกสตริงที่จะทำการมิวเตชั่นอย่างสุ่ม และนำมาผ่านการมิวเตชั่น ซึ่งในการวิจัยนี้ได้ใช้วิธีการมิวเตชั่นทั้งหมด 3 แบบคือ

6.3.3.1 การมิวเตชั่นแบบรีซิโพคอล เอ็กเชน(Reciprical Exchange Mutation) เริ่มจากเลือก 2 ตำแหน่งในสตริงอย่างสุ่ม แล้วทำการสลับที่กันระหว่างค่าในตำแหน่งดังกล่าว ในตัวอย่างได้เลือกตำแหน่งที่ 4 และ 6

$$m = [1\ 2\ 3\ \underline{4}\ 5\ \underline{6}\ 7\ 8\ 9]$$

$$\times$$

$$m = [1\ 2\ 3\ \underline{6}\ 5\ \underline{4}\ 7\ 8\ 9]$$

6.3.3.2 การมิวเตชั่นแบบอินเสอร์ทชั่น (Insertion Mutation) โดยทำการเลือกตำแหน่ง 1 ตำแหน่งหรือเป็นช่วงจากสตริงที่จะมิวเตชั่น ในตัวอย่างได้เลือกเป็นช่วงคือระหว่างตำแหน่งที่ 4 และ 6

$$m = [1\ 2\ 3\ \underline{4}\ 5\ \underline{6}\ 7\ 8\ 9]$$

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

แบ่งสตริงออกเป็น 2 ส่วน

m_1 = [1 2 3 5 7 8 9]

m_2 = [4 5 6]

สุ่มตำแหน่งที่จะแทรก m_2ใน m_1ในที่นี้สุ่มได้ตำแหน่งที่ 6ของ m_1

m_1 = [1 2 3 5 7 8 9]

m_1 = [1 2 3 5 7 8 4 5 6 9]

6.3.3.3 การมิวเตชั่นแบบแรนดอมซีเควนซ์(Random Seqeunce) เริ่มจากการสุ่มในตำแหน่งของสตริง *m* ทำการตัดสตริงที่อยู่หลังตำแหน่งที่สุ่มได้ทิ้ง แล้วสร้างสตริงที่เหลือโดยวิธีการสุ่ม ในตัวอย่างตำแหน่งที่สุ่มได้คือ 4

m = [1 2 3 4 5 6 7 8 9]

↓

m = [1 2 3 4 # # # # #]

↓

m = [1 2 3 4 6 5 9 8 7]

เช่นเดียวกับการครอสโอเวอร์ การมิวเตชั่นจะมีค่าความน่าจะเป็นในการมิวเตชั่น (Probability of Mutation, Pm)เพื่อกำหนดจำนวนสตริงที่จะต้องผ่านการมิวเตชั่น สรุปขั้นตอนของการทำมิวเตชั่นมีดังต่อไปนี้

- สร้างค่าสุ่ม (Float) หรือ *r* อยู่ในช่วง [0....1] สำหรับสตริงทุกสตริง
- สตริงใดที่ $r < p_m$ เลือกสตริงนั้น เพื่อทำการมิวเตชั่น

หลังจากที่ผ่านกระบวนการคัดเลือก ครอสโอเวอร์ และมิวเตชั่นแล้ว ประชากรรุ่นใหม่ก็พร้อมที่จะนำไปสู่กระบวนการคัดเลือกต่อไป กระบวนการคัดเลือกของรุ่นประชากรรุ่นใหม่ใช้การกระบวนการของวงล้อรูเล็ททีมีขนาดของช่องเป็นสัดส่วนกับค่าความเหมาะสม กระบวนการทั้งหมดจะเกิดขึ้นซ้ำๆกันซึ่งเป็นไปตามกระบวนการของ GAs

6.4 การประยุกต์เจนเนติกอัลกอริทึมกับการออกแบบผังโรงงานแบบวัตถุประสงค์เดียว

ต่อไปจะเป็นการแสดงรายละเอียดของขั้นตอนต่างๆของGAs ที่ประยุกต์ใช้ในการออกแบบผังโรงงานที่แผนกมีขนาดไม่เท่ากัน โดยยกตัวอย่างวิธีการรีโปรดักชั่นแบบวงล้อรูลเล็ต วิธีการครอสโอเวอร์แบบPMX และวิธีการมิวเตชั่นแบบReciprocal-Exchange

ตัวอย่าง 6.2 ผังโรงงานแบบวัตถุประสงค์เดียว ขนาด กว้าง 4 หน่วย * ยาว 6 หน่วย พื้นที่ 24 ตารางหน่วย มีทั้งหมด 6 แผนก

ตารางที่ 6.2 รายละเอียดของปัญหาตัวอย่าง 6.2

แผนกที่	พื้นที่ทั้งหมด	พื้นที่สำหรับวางเครื่องจักร(กว้าง *ยาว)	อัตราส่วนด้านกว้างต่อด้านยาว
1	3	1*3	5
2	9	2*3	5
3	3	1*3	5
4	1	1*1	5
5	2	1*2	5
6	6	2*2	5

หมายเหตุ

- พื้นที่สำหรับวางเครื่องจักร คือ พื้นที่ที่อยู่ภายในแผนก ไว้สำหรับวางเครื่องจักรที่จำเป็นในแผนกนั้นๆ
- อัตราส่วนด้านกว้างต่อด้านยาวของแผนก กำหนดไว้เพื่อป้องกันการสร้างผังโรงงานที่มีรูปร่างเป็นรูปสี่เหลี่ยมผืนผ้าที่ยาวและแคบ จนไม่เหมาะสมกับการใช้งานจริง

ตารางที่ 6.3 แผนภูมิการไหลระหว่างแผนก(เที่ยว) ของปัญหาตัวอย่าง 6.2

จาก-ไปยัง	1	2	3	4	5	6
1	0	63	605	551	116	136
2	63	0	635	941	50	191
3	104	71	0	569	136	55
4	65	193	622	0	77	90
5	162	174	607	591	0	179
6	156	13	667	661	175	0

ค่าใช้จ่ายของการขนถ่ายเท่ากับ 1 หน่วย/เที่ยว และคิดระยะทางระหว่างจุดเซ็นทรอยด์ของแต่ละแผนกแบบเรคติลิเนียร์ (ระยะทางรวมเท่ากับผลรวมของระยะทางตามแกน X และระยะทางตามแกน Y)

6.4.1 หลักการของการสร้างรหัสสตริง ในการสร้างสตริงคำตอบมีหลักการดังนี้

- สตริงจะประกอบด้วย 2 ส่วน คือ การเรียงลำดับของแผนกตามแถบ (Department Order) และ ความกว้างของแถบ(Band Width) ซึ่งความกว้างของแถบนั้นจะขึ้นกับการเรียงลำดับของแผนก
- ทำการสร้างรหัสของ Department Order ก่อน แล้วทำการสร้าง Band Width เพื่อให้เหมาะสม(เป็นไปตามข้อกำหนดต่าง ๆของปัญหา)กับ Department Order ที่สร้างขึ้นก่อนหน้า
- ในขั้นตอนของครอสโอเวอร์ และมิวเตชั่น นั้นจะมีการกระทำเฉพาะที่ Department Order ให้มีลักษณะเปลี่ยนไปก่อน แล้วจึงสร้าง Band Width ขึ้นมาเพื่อให้เหมาะสมกับ Department Order ที่ผ่านการครอสโอเวอร์ และมิวเตชั่นมาแล้ว กล่าวคือจะไม่ได้ทำการครอสโอเวอร์ และมิวเตชั่นที่ Band Width

6.4.2 การสร้างประชากรเริ่มต้น (Initial population) กำหนดให้ประชากรเริ่มต้นเท่ากับ 6 ตัว ทำการสร้างประชากรอย่างสุ่มได้ดังนี้

ตารางที่ 6.4 ประชากรเริ่มต้นของปัญหาตัวอย่าง6.2

สตริง	Order	Band	Cost
Str1	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381
Str2	[5 2 3 6 1 4]	[1 1 1 2 1 0]	33235
Str3	[1 2 4 3 5 6]	[1 1 1 1 2 0]	26951
Str4	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365
Str6	[1 5 3 4 2 6]	[1 1 1 3 0 0]	26844

Maximum Cost : 34381
Average Cost : 30257.17
Minimum Cost : 26844
Total : 161543
Std : 3437.96

Best String order: [1 5 3 4 2 6] band: [1 1 1 3 0 0] Cost: 26844

6.4.3 การรีโปรดักชั่นแบบวงล้อรูลเล็ต

การทำรีโปรดักชั่นใช้วิธีการของวงล้อรูลเล็ทโดยที่ Cost ของสตริงแต่ละตัวเป็นอัตราส่วนกับขนาดของช่องในวงล้อรูลเล็ท สตริงที่มีค่าความเหมาะสมสูงจะถูกคัดเลือกสำหรับทำการรีโปรดักชั่น เนื่องจากปัญหานี้เป็นการหา Cost ที่น้อยที่สุด จึงหาขนาดของช่องในวงล้อรู

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

เล็ทโดยการนำ Cost ของสตริง c_i แต่ละตัวมาลบออกจากผลรวมของ Cost ทั้งหมดในประชากร ทำให้ได้คำตอบเป็น *CostSum-* c_i จากนั้นจึงหาผลรวมของ *CostSum-* c_i ได้เป็น *Sum_of_(CostSum-*c_i*)* ขนาดของสล็อทในวงล้อรูเล็ทโดยการนำ *CostSum-* c_i ของสตริงแต่ละตัวหารด้วย *Sum_of_(CostSum-*c_i*)* โดยที่ขนาดของวงล้อรูเล็ทโดยรวมเป็น 100 เปอร์เซ็นต์

***ตารางที่ 6.5** การคำนวณfitness functionของปัญหาตัวอย่าง 6.2*

สตริง	Cost(C_i)	*CostSum-* c_i	*Fitness = (CostSum-* c_i*)/ Sum_of_(CostSum-*c_i*)*	*ความน่าจะเป็นสะสม*
Str1	34381	147162	0.1621	0.1621
Str2	33235	148308	0.1634	0.3255
Str3	26951	154592	0.1703	0.4958
Str4	27767	153776	0.1894	0.6852
Str5	32365	149178	0.1643	0.8295
Str6	26844	154699	0.1704	1.0000
total	181543	907715	1.0000	

จากนั้นจะทำการหมุนวงล้อรูเล็ท 6 ครั้ง การหมุนแต่ละครั้งหมายถึงการคัดเลือกสตริงเข้าไปสู่ประชากรใหม่ สมมุติค่าสุ่มจำนวน 6 ค่า ที่อยู่ในช่วง [0....1]

ตารางที่ 6.6 ค่าสุ่มจากวงล้อรูเล็ต

ครั้งที่	ค่าสุ่มจากวงล้อรูเล็ต	สตริงที่ผ่านการรีโปรดักชั่น
1	0.652588	Str4 Cost: 27767
2	0.551275	Str4 Cost: 27767
3	0.280907	Str2 Cost: 33235
4	0.002097	Str1 Cost: 34381
5	0.810821	Str5 Cost: 32365
6	0.958813	**Str6 Cost: 26844**

Maximum Cost : 34381

Average Cost : 30393.17

Minimum Cost : 26844

Total : 182359

Std : 3294.148

Best String order: [1 5 3 4 2 6] band: [1 1 1 3 0 0] Cost: 26844

เห็นได้ว่าหลังผ่านการรีโปรดักชั่นแล้ว สตริงที่ดีที่สุดคือ Str6 order: [1 5 3 4 2 6] band: [1 1 1 3 0 0] ได้ถ่ายทอดมายังลูกหลาน 1 สตริง จากนี้นำสตริงทั้งหมดเข้าสู่การครอสโอเวอร์แบบPMX

6.4.4 การครอสโอเวอร์แบบ PMX

กระบวนการต่อไปเป็นการครอสโอเวอร์แบบ PMX ค่าความน่าจะเป็นของการครอสโอเวอร์ p_c = 0.7 อาจคาดได้ว่า 70% ของประชากรที่มีการครอสโอเวอร์ (โดยเฉลี่ย) นั่นคือ 0.7 * 6 = 4.2 กระบวนการของการครอสโอเวอร์เริ่มแรกคือทำการสุ่มค่า r อยู่ในช่วง [0....1] ได้ค่าดังต่อไปนี้

ตารางที่ 6.7 สตริงที่ผ่านเข้าสู่การครอสโอเวอร์

สตริง	Order	Band	Cost
Str1	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767
Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767
Str3	[5 2 3 6 1 4]	[1 1 1 2 1 0]	33235
Str4	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365
Str6	[1 5 3 4 2 6]	[1 1 1 3 0 0]	26844

ตารางที่ 6.8 สุ่มค่าเพื่อเลือกสตริงไปทำการครอสโอเวอร์

ครั้งที่	ค่าสุ่ม	น้อยกว่า 0.7(Pc)
1	0.4877	✓
2	0.5807	✓
3	0.6055	✓
4	0.8059	
5	0.9354	
6	0.3515	✓

หมายเหตุ หากว่าสุ่มเลือกสตริงพ่อแม่ได้ไม่ครบคู่ ให้ทำการสุ่มต่อไปว่าจะทำการสุ่มหาสตริงเพิ่มหรือตัดสตริงที่สุ่มได้แล้วออกไป 1 สตริง

ตารางที่ 6.9 สตริงที่ผ่านการลือกเข้าสู่การครอสโอเวอร์

คู่ที่	สตริง	Order	Band
1	Str1	[2 3 4 5 1 6]	[3 1 2 0 0 0]
	Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]
2	Str3	[5 2 3 6 1 4]	[1 1 1 2 1 0]
	Str6	[1 5 3 4 2 6]	[1 1 1 3 0 0]

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

นำสตริงorder ของคู่ที่ 1เข้าทำการครอสโอเวอร์

P_1=[2 3 4 5 1 6]

P_2=[2 3 4 5 1 6]

สุ่มตำแหน่งที่จะทำการครอสโอเวอร์ได้ ตำแหน่งที่ 2 และ 5

P_1=[2 3 | 4 5 1 | 6]

P_1=[2 3 | 4 5 1 | 6]

ทำการสลับสตริงที่อยู่ภายในเครื่องหมาย | | ได้สตริงลูกหลาน

O_1=[2 3 4 5 1 6]

O_2=[2 3 4 5 1 6]

นำสตริงลูกหลานทั้ง 2 ไปสุ่มหาBand Width ที่เหมาะสม และ ซึ่งได้ผลดังนี้

ตารางที่ 6.10 สตริงคู่ที่ 1 หลังผ่านการครอสโอเวอร์

สตริง	Order	Band	Cost
O_1	[2 3 4 5 1 6]	[2 1 1 2 0 0]	23503
O_2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767

หมายเหตุ หลังการครอสโอเวอร์ครั้งนี้ สตริงลูกหลานทั้งสอง มีdepartment order เหมือนเดิม แต่มีBand Widthเปลี่ยนไป จึงทำให้ค่า Costเปลี่ยนไป

นำสตริงorder ของคู่ที่ 2เข้าทำการครอสโอเวอร์

P_1=[5 2 3 6 1 4]

P_2=[1 5 3 4 2 6]

สุ่มตำแหน่งที่จะทำการครอสโอเวอร์ได้ ตำแหน่งที่ 3 และ 5

P_1=[5 2 3 | 6 1 | 4]

P_2=[1 5 3 | 4 2 | 6]

ทำการสลับสตริงที่อยู่ภายในเครื่องหมาย | | ได้สตริงลูกหลาน

O_1=[5 1 3 4 2 6]

O_2=[2 5 3 6 1 4]

นำสตริงลูกหลานทั้ง 2 ไปสุ่มหาBand Width ที่เหมาะสม และ ซึ่งได้ผลดังนี้

ตารางที่ 6.11 สตริงคู่ที่ 2 หลังผ่านการครอสโอเวอร์

สตริง	Order	Band	Cost
O_1	[5 1 3 4 2 6]	ไม่สามารถหาได้	ไม่สามารถหาได้
O_2	[2 5 3 6 1 4]	[2 1 2 1 0 0]	28889

เนื่องจาก O_1 ของสดริงพ่อแม่คู่ที่2 ไม่สามารถหาค่าBand Width ที่เหมาะสมได้ จึงตัด O_1 ทิ้งและเลือกสดริงพ่อแม่ P_1 ,P_2ที่มีค่าCost น้อยกว่ามาเป็น O_1 ดังนั้นจึงเลือก *Str6* มาเป็น O_1แทน

ตารางที่ 6.12 นำสตริงลูกหลานที่ได้จากการครอสโอเวอร์แทนที่สตริงพ่อแม่ ในประชากรเดิม

สตริง	Order	Band	Cost
Str1	[2 3 4 5 1 6]	[2 1 1 2 0 0]	23503
Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767
Str3	[1 5 3 4 2 6]	[1 1 1 3 0 0]	26844
Str4	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365
Str6	[2 5 3 6 1 4]	[2 1 2 1 0 0]	28889

Maximum Cost : 34381

Average Cost : 28956.17

Minimum Cost : 23503

Total : 173749

Std : 3915.759

Best String order: [2 3 4 5 1 6] band: [2 1 2 0 0 0] Cost: 23503

หลังผ่านการครอสโอเวอร์แล้ว ได้มีการสร้างสดริงที่มีคุณสมบัติดีที่สุดคือ Str1 Order: [2 3 4 5 1 6] Band: [2 1 1 2 0 0] ขึ้นมาใหม่ 1 สดริง จากนี้นำสดริงทั้งหมดเข้าสู่มิวเตชั่นแบบรีซิโพรคอล เอ็กเชน(Reciprocal Exchange Mutation)

6.4.5 มิวเตชั่นแบบรีซิโพรคอล เอ็กเชน(Reciprocal Exchange Mutation)

กระบวนการต่อไปเป็นมิวเตชั่นแบบรีซิโพรคอล เอ็กเชน กำหนดความน่าจะเป็นของมิวเตชั่นp_m = 0.1 อาจคาดได้ว่า 10% ของประชากรที่มีการมิวเตชั่น(โดยเฉลี่ย) นั่นคือ 0.1 * 6 = 0.6 กระบวนการของการครอสโอเวอร์เริ่มแรกคือทำการสุ่มค่า *r* อยู่ในช่วง [0....1] ได้ค่าดังตารางที่6.14

ตารางที่ 6.13 สตริงที่ผ่านเข้าสู่การมิวเตชั่น

สตริง	Order	Band	Cost
Str1	[2 3 4 5 1 6]	[2 1 1 2 0 0]	23503
Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767
Str3	[1 5 3 4 2 6]	[1 1 1 3 0 0]	26844
Str4	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365
Str6	[2 5 3 6 1 4]	[2 1 2 1 0 0]	28889

ตารางที่ 6.14 ค่าสุ่มเพื่อเลือกสตริงไปมิวเตชั่น

ครั้งที่	ค่าสุ่ม	น้อยกว่า 0.1(Pm)
1	0.9798	
2	0.4193	
3	0.0902	✓
4	0.8291	
5	0.2713	
6	0.1318	

สตริงที่จะทำการมิวเตชั่น คือ Str3 ซึ่งจะทำการมิวเตชั่นส่วนOrder

Str3 = [1 5 3 4 2 6]

เลือกตำแหน่งที่จะทำการสลับที่ 2 ตำแหน่งคือ 4 และ 6

Str3 = [1 5 3 <u>4</u> 2 <u>6</u>]

Str3 = [1 5 3 <u>6</u> 2 <u>4</u>]

จากนั้นสร้าง Band Width จาก orderของ Str3

ตารางที่ 6.15 สตริงที่ได้จากการมิวเตชั่น

สตริง	Order	Band	Cost
Str3	[1 5 3 6 2 4]	[1 1 2 1 1 0]	31456

นำStr3 ที่ได้จากการมิวเตชั่นรวมเข้ากับประชากรเริ่มต้น ดังตารางที่ 6.18

สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 6.16 สตริงทั้งหมดหลังการมิวเตชั่น

สตริง	Order	Band	Cost
Str1	[2 3 4 5 1 6]	[2 1 1 2 0 0]	23503
Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767
Str3	**[1 5 3 6 2 4]**	**[1 1 2 1 1 0]**	**31456**
Str4	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365
Str6	[2 5 3 6 1 4]	[2 1 2 1 0 0]	28889

Maximum Cost : 34381

Average Cost : 29728.83

Minimum Cost : 23503

Total : 178361

Std : 3870.148

Best String order: [2 3 4 5 1 6] band: [2 1 2 0 0 0] Cost: 23503

พบว่าหลังผ่านการมิวเตชั่นแล้ว สตริงที่ดีที่สุดคือ Str1 Order: [2 3 4 5 1 6] Band: [2 1 2 0 0 0] Cost: 23503 ยังคงถ่ายทอดมายังรุ่นลูกหลาน

หมายเหตุ การอิลิทิส คือ การเก็บสตริงที่ดีไว้เสมอ กล่าวคือเมื่อระหว่างกระบวนการรีโปรดักชั่น ครอสโอเวอร์ มิวเตชั่น อาจจะทำให้สตริงที่ดีถูกทำลายไปได้ ดังนั้นเมื่อเสร็จสิ้นแต่ละขั้นตอนของGAs ให้ทำการเปรียบเทียบระหว่างสตริงที่ดีที่สุดของระหว่างประชากรก่อนและหลังผ่านขั้นตอนดังกล่าว เพื่อหาว่าสตริงใดมีคุณสมบัติดีกว่ากัน หากว่าสตริงที่ดีที่สุดของประชากรก่อนการเข้าสู่กระบวนการของGAs นั้นมีคุณสมบัติที่ดีกว่า ให้ทำการนำตริงนั้นมาแทนที่สตริงที่แย่ที่สุดของประชากรหลังผ่านกระบวนการGAs แล้ว

6.5 การประยุกต์เจนเนติกอัลกอริทึมกับการออกแบบผังโรงงานแบบหลายวัตถุประสงค์

การประยุกต์เจนเนติกอัลกอริทึมกับการออกแบบผังโรงงานแบบหลายวัตถุประสงค์ มีลักษณะคล้ายคลึงกับการประยุกต์เจนเนติกอัลกอริทึมกับการออกแบบผังโรงงานแบบวัตถุประสงค์เดียว แต่มีข้อแตกต่างที่ฟังก์ชั่นวัตถุประสงค์ โดยที่

- ผังโรงงานแบบวัตถุประสงค์เดียว ฟังก์ชั่นวัตถุประสงค์ คือ ค่าใช้จ่าย(Cost)
- ผังโรงงานแบบหลายวัตถุประสงค์ ฟังก์ชั่นวัตถุประสงค์ คือ ค่าfitnessรวม ซึ่ง

*ค่าfitnessรวม = (W1*Cost-fitness) + (W2*TCR-fitness)*

ค่าTCR พิจารณาระยะทางระหว่างแผนก

W1 คือ น้ำหนักที่ให้ความสำคัญกับค่าใช้จ่ายของผังโรงงาน (Cost) มีค่าระหว่าง[0,1]

W2 คือ น้ำหนักที่ให้ความสำคัญกับความสัมพันธ์ระหว่างแผนก (TCR) มีค่าระหว่าง [0,1]

และ *W1+W2*=1

ต่อไปเป็นตัวอย่างการการประยุกต์เจนเนติกอัลกอริทึมกับการออกแบบผังโรงงานแบบหลายวัตถุประสงค์ วิธีการรีโปรดักชั่นแบบวงล้อรูลเล็ต วิธีการครอสโอเวอร์แบบPMX และวิธีการมิวเตชั่นแบบReciprocal-Exchange

ตัวอย่าง 6.3 ผังโรงงานแบบหลายวัตถุประสงค์ ขนาด กว้าง 4 หน่วย * ยาว 6 หน่วย พื้นที่ 24 ตารางหน่วย มีทั้งหมด 6 แผนก W1=0.25 W2=0.75

ตารางที่ 6.17 รายละเอียดของปัญหาตัวอย่าง 6.3

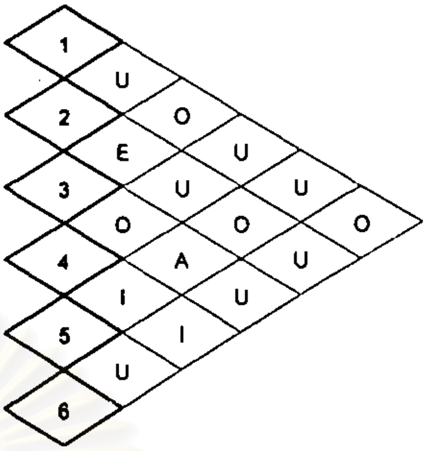
แผนกที่	พื้นที่ทั้งหมด	พื้นที่สำหรับวางเครื่องจักร(กว้าง *ยาว)	อัตราส่วนด้านกว้างต่อด้านยาว
1	3	1*3	5
2	9	2*3	5
3	3	1*3	5
4	1	1*1	5
5	2	1*2	5
6	6	2*2	5

ตารางที่ 6.18 แผนภูมิการไหลระหว่างแผนก(*เที่ยว*) ของปัญหาตัวอย่าง 6.3

จาก-ไปยัง	1	2	3	4	5	6
1	0	63	605	551	116	136
2	63	0	635	941	50	191
3	104	71	0	569	136	55
4	65	193	622	0	77	90
5	162	174	607	591	0	179
6	156	13	667	661	175	0

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย
สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

A: 5
E: 4
I: 3
O: 2
U: 1
X: 0



รูปที่ 6.3 แผนภูมิความสัมพันธ์ระหว่างแผนก ตัวอย่าง 6.3

ค่าใช้จ่ายของการขนถ่ายเท่ากับ 1 หน่วย/เที่ยว และคิดระยะทางระหว่างจุดเซ็นทรอยด์ของแต่ละแผนกแบบเรคติลิเนียร์ (ระยะทางรวมเท่ากับผลรวมของระยะทางตามแกน X และระยะทางตามแกน Y)

6.5.1 การสร้างประชากรเริ่มต้น (Initial population)

กำหนดให้ประชากรเริ่มต้นเท่ากับ 6 ตัว ทำการสร้างประชากรอย่างสุ่มได้ดังนี้

ตารางที่ 6.19 ประชากรเริ่มต้นของปัญหาตัวอย่าง6.3

สตริง	Order	Band	Cost	TCR
Str1	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381	214.00
Str2	[5 2 3 6 1 4]	[1 1 1 2 1 0]	33235	239.78
Str3	[1 2 4 3 5 6]	[1 1 1 1 2 0]	26951	194.00
Str4	**[2 3 4 5 1 6]**	**[3 1 2 0 0 0]**	**27767**	**178.00**
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365	266.67
Str6	[1 5 3 4 2 6]	[1 1 1 3 0 0]	26844	192.00

6.5.2 การรีโปรดักชั่นแบบวงล้อรูลเล็ต

ทำการคำนวณค่า fitnessรวมดังตารางที่6.20 จากนั้นจะทำการหมุนวงล้อรูเล็ท 6 ครั้ง การหมุนแต่ละครั้งหมายถึงการคัดเลือกสตริงเข้าไปสู่ประชากรใหม่ สมมุติค่าสุ่มจำนวน 6 ค่า ที่อยู่ในช่วง [0....1] ดังตารางที่ 6.21

ตารางที่ 6.20 การคำนวณfitness functionของปัญหาตัวอย่าง 6.3

สตริง	Cost	Cost-fitness	TCR	TCR-fitness	Total fitness	ความน่าจะเป็นสะสม
Str1	34381	0.1621	214.00	0.1667	0.1656	0.1656
Str2	33235	0.1634	239.78	0.1627	0.1629	0.3265
Str3	26951	0.1703	194.00	0.1898	0.1699	0.4984
Str4	**27767**	**0.1694**	**178.00**	**0.1723**	**0.1716**	**0.6700**
Str5	32365	0.1643	266.67	0.1585	0.1600	0.8300
Str6	26844	0.1704	192.00	0.1701	0.1702	1.0000
รวม	161543	1.0000	1284.45	1.0000	1.0000	

ตารางที่ 6.21 ค่าสุ่มจากวงล้อรูเล็ต

ครั้งที่	ค่าสุ่มจากวงล้อรูเล็ต	สตริงที่ผ่านการรีโปรดักชั่น
1	0.652588	Str4 fitness:0.1716
2	0.551275	Str4 fitness:0.1716
3	0.280907	Str2 fitness:0.1629
4	0.002097	Str1 fitness:0.1656
5	0.810621	Str5 fitness:0.1600
6	0.958613	Str6 fitness:0.1702

Best String Order:[2 3 4 5 1 6] Band:[3 1 2 0 0 0] Cost: 27767 TCR: 178.00 total fitness: 0.1716

เห็นได้ว่าหลังผ่านการรีโปรดักชั่นแล้ว สตริงที่ดีที่สุดคือ Str4 Order:[2 3 4 5 1 6] Band:[3 1 2 0 0 0] ได้ถ่ายทอดมายังลูกหลาน 2 สตริง จากนี้นำสตริงทั้งหมดเข้าสู่การครอสโอเวอร์แบบPMX

6.5.3 การครอสโอเวอร์แบบ PMX

กระบวนการต่อไปเป็นการครอสโอเวอร์แบบ PMX ค่าความน่าจะเป็นของการครอสโอเวอร์ p_c = 0.7 อาจคาดได้ว่า 70% ของประชากรที่มีการครอสโอเวอร์ (โดยเฉลี่ย) นั่นคือ 0.7 * 6 = 4.2 กระบวนการของการครอสโอเวอร์เริ่มแรกคือทำการสุ่มค่า *r* อยู่ในช่วง [0....1] ได้ค่าดังตารางที่ 6.23

ตารางที่ 6.22 สตริงที่ผ่านเข้าสู่การครอสโอเวอร์

สตริง	Order	Band	Cost	TCR	Total fitness
Str1	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767	178.00	0.1716
Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767	176.00	0.1716
Str3	[5 2 3 6 1 4]	[1 1 1 2 1 0]	33235	239.78	0.1629
Str4	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381	214.00	0.1656
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365	266.67	0.1600
Str6	[1 5 3 4 2 6]	[1 1 1 3 0 0]	26844	192.00	0.1702

ตารางที่ 6.23 สุ่มค่าเพื่อเลือกสตริงไปทำการครอสโอเวอร์

ครั้งที่	ค่าสุ่ม	น้อยกว่า 0.7(Pc)
1	0.4677	✓
2	0.5807	✓
3	0.6055	✓
4	0.8059	
5	0.9354	
6	0.3515	✓

หมายเหตุ หากว่าสุ่มเลือกสตริงพ่อแม่ได้ไม่ครบคู่ ให้ทำการสุ่มต่อไปว่าจะทำการสุ่มหาสตริงเพิ่มหรือตัดสตริงที่สุ่มได้แล้วออกไป 1 สตริง

ตารางที่ 6.24 สตริงที่ผ่านการลือกเข้าสู่การครอสโอเวอร์

คู่ที่	สตริง	Order	Band
1	Str1	[2 3 4 5 1 6]	[3 1 2 0 0 0]
	Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]
2	Str3	[5 2 3 6 1 4]	[1 1 1 2 1 0]
	Str6	[1 5 3 4 2 6]	[1 1 1 3 0 0]

ในสตริงคู่ที่ 1 และ 2 ได้สุ่มตำแหน่งครอสโอเวอร์ เหมือนการครอสโอเวอร์ในตัวอย่าง 6.2 ซึ่งๆได้ผลดังนี้

ตารางที่ 6.25 สตริงคู่ที่ 1 หลังผ่านการครอสโอเวอร์

สตริง	Order	Band	Cost	TCR
O_1	[2 3 4 5 1 6]	[2 1 1 2 0 0]	23503	147.33
O_2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767	178

หมายเหตุ หลังการครอสโอเวอร์ครั้งนี้ สตริงลูกหลานทั้งสอง มีDepartment Order เหมือนเดิม แต่มีBand Widthเปลี่ยนไป จึงทำให้ค่า Cost และ TCR เปลี่ยนไป

ตารางที่ 6.26 สตริงคู่ที่ 2 หลังผ่านการครอสโอเวอร์

สตริง	Order	Band	Cost	TCR
O_1	[5 1 3 4 2 6]	ไม่สามารถหาได้	ไม่สามารถหาได้	ไม่สามารถหาได้
O_2	[2 5 3 6 1 4]	[2 1 2 1 0 0]	28889	188.667

เนื่องจาก O_1 ของสตริงพ่อแม่คู่ที่2 ไม่สามารถหาค่าBand Width ที่เหมาะสมได้ จึงตัด O_1 ทิ้งและเลือกสตริงพ่อแม่ P_1 ,P_2ที่มีค่าCost น้อยกว่ามาเป็น O_1 ดังนั้นจึงเลือก Str6 มาเป็น O_1 แทน

ตารางที่ 6.27 นำสตริงลูกหลานที่ได้จากการครอสโอเวอร์แทนที่สตริงพ่อแม่ ในประชากรเดิม

สตริง	Order	Band	Cost	TCR	Total fitness
Str1	**[2 3 4 5 1 6]**	**[2 1 1 2 0 0]**	**23503**	**147.33**	**0.1746**
Str2	**[2 3 4 5 1 6]**	**[3 1 2 0 0 0]**	**27767**	**178.00**	**0.1695**
Str3	**[1 5 3 4 2 6]**	**[1 1 1 3 0 0]**	**26844**	**192.00**	**0.1680**
Str4	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34361	214.00	0.1630
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365	266.67	0.1570
Str6	**[2 5 3 8 1 4]**	**[2 1 2 1 0 0]**	**28889**	**188.667**	**0.1678**

Best String Order:[2 3 4 5 1 6] Band:[2 1 1 2 0 0] Cost::23503 TCR:147.33 total fitness: 0.1746

หลังผ่านการครอสโอเวอร์แล้ว ได้มีการสร้างสตริงที่มีคุณสมบัติดีที่สุดคือ Str1 Order: [2 3 4 5 1 6] Band: [2 1 1 2 0 0] ขึ้นมาใหม่ 1 สตริง จากนี้นำสตริงทั้งหมดเข้าสู่มิวเตชั่นแบบรีซิโพรคอล เอ็กเชน(Reciprocal Exchange Mutation)

6.5.4 มิวเตชั่นแบบรีซิโพรคอล เอ็กเชน(Reciprocal Exchange Mutation)

กระบวนการต่อไปเป็นมิวเตชั่นแบบรีซิโพรคอล เอ็กเชน กำหนดความน่าจะเป็นของมิวเตชั่นp_m = 0.1 อาจคาดได้ว่า 10% ของประชากรที่มีการมิวเตชั่น(โดยเฉลี่ย) นั่นคือ 0.1 * 6 = 0.6 กระบวนการของการครอสโอเวอร์เริ่มแรกคือทำการสุ่มค่า r อยู่ในช่วง [0....1] ได้ค่าดังตารางที่ 6.28

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 6.28 ค่าสุ่มเพื่อเลือกสตริงไปมิวเตชั่น

ครั้งที่	ค่าสุ่ม	น้อยกว่า 0.1(Pm)
1	0.9798	
2	0.4193	
3	0.0902	✓
4	0.8291	
5	0.2713	
6	0.1318	

สตริงที่จะทำการมิวเตชั่น คือ Str3 ซึ่งจะทำการมิวเตชั่นส่วนOrder

Str3 = [1 5 3 4 2 6]

เลือกตำแหน่งที่จะทำการสลับที่ 2 ตำแหน่งคือ 4 และ 6

Str3 = [1 5 3 4 2 6]

Str3 = [1 5 3 6 2 4]

จากนั้นสร้าง Band Width จาก orderของ Str3

ตารางที่ 6.29 สตริงที่ได้จากการมิวเตชั่น

สตริง	Order	Band	Cost	TCR
Str3	[1 5 3 6 2 4]	[1 1 2 1 1 0]	31456	214.00

ตารางที่ 6.30 นำสตริงลูกหลานที่ได้จากการครอสโอเวอร์แทนที่สตริงพ่อแม่ ในประชากรเดิม

สตริง	Order	Band	Cost	TCR	Total fitness
Str1	[2 3 4 5 1 6]	[2 1 1 2 0 0]	23503	147.33	0.1751
Str2	[2 3 4 5 1 6]	[3 1 2 0 0 0]	27767	178.00	0.1701
Str3	[1 5 3 6 2 4]	[1 1 2 1 1 0]	31456	214.00	0.1646
Str4	[4 5 1 6 2 3]	[1 2 3 0 0 0]	34381	214.00	0.1638
Str5	[4 5 2 1 3 6]	[1 1 1 1 2 0]	32365	266.87	0.1579
Str6	[2 5 3 6 1 4]	[2 1 2 1 0 0]	28889	188.667	0.1685

Best String Order:[2 3 4 5 1 6] Band:[2 1 1 2 0 0] Cost::23503 TCR:147.33 total fitness: 0.1751

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

6.6 สรุป

GAs เป็นการค้นหาคำตอบที่มีประสิทธิภาพสูง แต่การนำ GAs ไปใช้ในการแก้ปัญหาต่าง ๆนั้นจำเป็นต้องมีกระบวนการต่าง ๆในรายละเอียดปลีกย่อยเพื่อที่จะทำให้ GAs สามารถทำงานได้อย่างที่ต้องการ

การรีโปรดักชั่น(การคัดเลือก) แบบวงล้อรูลเล็ทสามารถทำได้โดย

- สร้างค่าสุ่ม (Random Number) r ที่มีค่าอยู่ในช่วง [0...1]
- ถ้า $r < q_i$ เลือกสตริงตัวแรก ถ้าไม่ใช่ เลือกสตริงตัวที่ i โดยที่ i อยู่ระหว่าง ($2 \leq i \leq pop$) หรือ ($q_{i-1} \leq r \leq q_i$) ในบางครั้งสตริงที่ถูกคัดเลือกแล้วบางตัวบางตัวอาจถูกเลือกได้มากกว่า 1 ครั้ง

การครอสโอเวอร์ ที่ใช้ในงานวิจัยมี 5 วิธี ได้แก่ การครอสโอเวอร์แบบ PMX (Partially Match Crossover) การครอสโอเวอร์แบบ OX (Order Crossover) การครอสโอเวอร์แบบ CX (Cycle Crossover) การครอสโอเวอร์แบบPBX(Position-Base) และวิธีการครอสโอเวอร์แบบ PBX(Order-Base) โดยสามารถทำได้ดังต่อไปนี้

- สร้างค่าสุ่ม (Random Number) r ที่มีค่าอยู่ในช่วง [0...1]
- ถ้า $r < p_c$ เลือกสตริงนั้นเพื่อทำการครอสโอเวอร์ตัวแรก
- จับคู่กับสตริงอีกหนึ่งตัวเพื่อทำการครอสโอเวอร์ ตามขั้นตอนของการครอสโอเวอร์แต่ละแบบ

การมิวเตชั่น ที่ใช้ในงานวิจัยมี 3 วิธี ได้แก่ การมิวเตชั่นแบบ Insertion การมิวเตชั่นแบบ Reciprocal Exchange และการมิวเตชั่นแบบ Random Sequence โดยสามารถทำได้ดังต่อไปนี้

- สร้างค่าสุ่ม (Float) หรือ r อยู่ในช่วง [0....1] สำหรับสตริงทุกสตริง
- สตริงใดที่ $r < p_m$ เลือกสตริงนั้น เพื่อทำการมิวเตชั่นแบบต่าง ๆ

การประยุกต์เจนเนติกอัลกอริทึมกับการออกแบบผังโรงงานแบบวัตถุประสงค์เดียวและการประยุกต์เจนเนติกอัลกอริทึมกับการออกแบบผังโรงงานแบบหลายวัตถุประสงค์ มีลักษณะคล้ายคลึงกับ แต่มีข้อแตกต่างที่ฟังก์ชั่นวัตถุประสงค์ คือ

- ผังโรงงานแบบวัตถุประสงค์เดียว ฟังก์ชั่นวัตถุประสงค์ คือ ค่าใช้จ่าย(Cost)
- ผังโรงงานแบบหลายวัตถุประสงค์ ฟังก์ชั่นวัตถุประสงค์ คือ ค่าfitnessรวม