

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 การจัดลำดับงาน (Scheduling)

ความหมายของการจัดลำดับงานนี้สามารถพิจารณาได้ 2 แนวมุม คือ การจัดลำดับงานเป็นกระบวนการที่ช่วยในการตัดสินใจ หรือกล่าวอีกนัยหนึ่งว่าเป็นกระบวนการสร้างตารางการทำงานของทรัพยากรขึ้นมา และการจัดลำดับงานเป็นทฤษฎีทางคณิตศาสตร์ ซึ่งรวบรวมหลักการแบบจำลอง เทคนิค และข้อสรุปเชิงตรรกะ ที่ทำให้มนุษย์เข้าใจหน้าที่ของการจัดลำดับงานได้ดีขึ้น ในความหมายที่ 2 นี้เอง สิ่งที่ได้เรียนรู้เกี่ยวกับการจัดลำดับงาน สามารถนำไปใช้กับทฤษฎีอื่น ๆ และส่งผลให้มีคุณค่าเชิงความคิดมากขึ้น

ในการศึกษาเรื่องการจัดลำดับงานนี้ได้มีการวิจัยทฤษฎีการจัดลำดับงานขึ้นโดยส่วนมากมีพื้นฐานมาจากการพิสูจน์เชิงคณิตศาสตร์ โดยหลักแล้วจะเกี่ยวข้องกับแบบจำลองเชิงคณิตศาสตร์ ที่นำมาใช้ในการจัดลำดับงานเป็นการพัฒนาแบบจำลองและโดยเพื่อความสัมพันธ์ระหว่างทฤษฎีและการปฏิบัติจริงในการแก้ปัญหาการจัดลำดับงานมีเงื่อนไข 2 ชนิดคือจำนวนทรัพยากรมีจำนวนจำกัดและข้อจำกัดในลำดับการทำงาน การหาคำตอบของการจัดลำดับงานโดยมากมี 2 คำถามคือต้องใช้ทรัพยากรตัวไหนในการจัดลำดับงานและงานแต่ละงานถูกทำขึ้นเมื่อไร

ในส่วนของโจทย์ของปัญหาการจัดลำดับงานจะประกอบด้วยค่าที่เข้ามาเกี่ยวข้องดังนี้

1. Processing time ใช้สัญลักษณ์ P_j คือเวลาที่ใช้ในการทำงานที่ j
2. Release time ใช้สัญลักษณ์ R_j คือเวลาในการปล่อยงานที่ j
3. Due date ใช้สัญลักษณ์ D_j คือเวลาซึ่งงาน j ต้องถูกส่งมอบ
4. Completion time ใช้สัญลักษณ์ C_j คือเวลาที่งาน j สิ้นสุด
5. Flow time ใช้สัญลักษณ์ F_j คือช่วงเวลาที่งาน j อยู่ในระบบ
6. Lateness ใช้สัญลักษณ์ L_j คือนำ $C_j - D_j$ และ Tardiness ใช้สัญลักษณ์ T_j

ตัววัดประสิทธิภาพการจัดลำดับงานของโจทย์ของปัญหาการจัดลำดับงานจะประกอบด้วยค่าที่เข้ามาเกี่ยวข้องดังนี้

1. Mean Flow time
2. Mean Tardiness
3. Maximum Flow time

4. Maximum Tardiness

5. Makespan ($C_{\max}$)

ในการอธิบายลักษณะของปัญหาการจัดลำดับงานเชิงทฤษฎีได้มีการสร้างสัญลักษณ์เพื่ออธิบายลักษณะของปัญหาการจัดลำดับงานโดยแบ่งสัญลักษณ์ออกเป็น 3 ช่วงได้แก่ α , β และ γ

สัญลักษณ์ α ใช้อธิบายสภาพแวดล้อมของเครื่องจักร สัญลักษณ์แทนในช่วงนี้มีดังต่อไปนี้
 สัญลักษณ์ 1 แทนการจัดลำดับงานที่มีเครื่องจักรเครื่องเดียว
 สัญลักษณ์ P แทนการจัดลำดับงานที่มีเครื่องจักร m เครื่อง ทำงานขนานกัน
 สัญลักษณ์ Q แทนการจัดลำดับงานที่มีเครื่องจักร m เครื่อง ทำงานขนานกันโดยเครื่องจักรมีความเร็วในการทำงานแตกต่างกัน
 สัญลักษณ์ R แทนการจัดลำดับงานที่มีเครื่องจักร m เครื่อง ทำงานขนานกันที่ไม่มีความสัมพันธ์กันเลย

สัญลักษณ์ β จะอธิบายถึงลักษณะของงาน สัญลักษณ์ที่ใช้แทนในช่วงนี้มีดังต่อไปนี้
 สัญลักษณ์ r_i แทนงานแต่ละงานจะมีเวลาในการปล่อยงาน (Release Date)
 สัญลักษณ์ d_i แทนงานแต่ละงานจะมีเวลาส่งมอบงาน (Due date)
 สัญลักษณ์ pmtn (preemption) แทนงานที่สามารถหยุดระหว่างทำได้

สัญลักษณ์ γ อธิบายฟังก์ชันเป้าหมายหรือ Objective Function สัญลักษณ์ที่ใช้แทนในช่วงนี้มีดังต่อไปนี้

สัญลักษณ์ $C_{\max}$ หมายถึง ปัญหาการจัดลำดับงานนี้มีเป้าหมายเพื่อหาค่า Makespan ต่ำที่สุด
 สัญลักษณ์ $L_{\max}$ หมายถึง ปัญหาการจัดลำดับงานนี้มีเป้าหมายเพื่อหาค่า Lateness ต่ำที่สุด
 สัญลักษณ์ $T_{\max}$ หมายถึง ปัญหาการจัดลำดับงานนี้มีเป้าหมายเพื่อหาค่า Tardiness ต่ำที่สุด
 สัญลักษณ์ $\sum C_i$ หมายถึง ปัญหาการจัดลำดับงานนี้มีเป้าหมายเพื่อทำให้ผลรวมของเวลาเสร็จสิ้นของทุกงานมีค่าต่ำสุดหรือน้อยที่สุด

โดยทั่วไปปัญหาการจัดลำดับงานถูกแบ่งเป็น 2 ลักษณะคือ ปัญหาที่ข้อมูลของโจทย์ปัญหาเป็นค่าที่แน่นอนและไม่เปลี่ยนแปลง (Deterministic Scheduled Problem) และปัญหาที่ข้อมูล

ของโจทย์ปัญหาเป็นค่าที่ไม่แน่นอนและขึ้นอยู่กับความน่าจะเป็น (Stochastic Scheduled Problem)

2.2 การจัดลำดับงานแบบ On-line

โดยทั่วไปแล้วปัญหาที่ต้องการการตัดสินใจในเชิงการวิจัยการดำเนินงานมักจะมีสมมติฐานว่าผู้ที่ทำการตัดสินใจจะทราบข้อมูลต่าง ๆ ของโจทย์ปัญหาโดยข้อมูลชุดนั้นเป็นค่าที่แน่นอน อย่างไรก็ตามยังมีปัญหาที่ต้องการตัดสินใจบางปัญหามีความไม่แน่นอนสูง ดังนั้นการแก้ปัญหาโดยใช้สมมติฐานขั้นต้นจึงไม่ใช่วิธีการที่เหมาะสมนัก ด้วยเหตุนี้เองจึงมีนักวิจัยกลุ่มหนึ่งได้พัฒนาวิธีการแก้ปัญหาการจัดตารางงาน ที่ต้องการตัดสินใจได้เป็น 2 รูปแบบ

ปัญหาแบบ Off-line ปัญหาประเภทนี้มีสมมติฐานว่าข้อมูลของโจทย์ปัญหาผู้แก้ปัญหาจะทราบข้อมูลทั้งหมดตั้งแต่มเริ่มต้น

ปัญหาแบบ On-line ปัญหาประเภทนี้ข้อมูลของปัญหาจะถูกแสดงออกมาให้ผู้แก้ปัญหาได้เห็นเมื่อเวลาผ่านไป

ในงานวิจัยนี้ผู้วิจัยจะศึกษาการแก้ปัญหาการจัดลำดับงานในกรณีที่มีข้อต้องห้ามแบบ On-line โดยทั่วไปแล้วการแก้ไขปัญหการจัดลำดับงานแบบ On-line สามารถแบ่งได้เป็น 2 ประเภท ดังนี้

2.2.1 การจัดลำดับงานแบบหนึ่งต่อหนึ่ง (One by one)

การจัดลำดับงานแบบหนึ่งต่อหนึ่ง เป็นปัญหาการจัดลำดับงานที่มีสมมติฐานว่าผู้จัดลำดับงานจะทราบข้อมูลของงานได้ทีละหนึ่งชิ้นเท่านั้น โดยหลังจากทราบข้อมูลของงานแล้วผู้จัดลำดับงานต้องตัดสินใจว่าจะกำหนดให้ทรัพยากรตัวไหนทำงานนั้นและเริ่มทำงานเมื่อเวลาเท่าไร

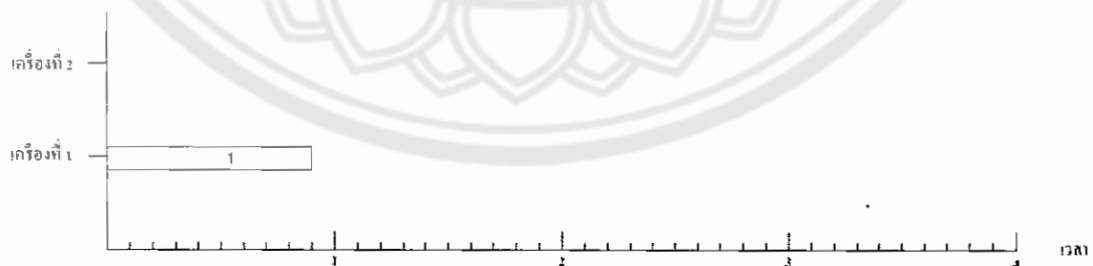
ตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักรสองเครื่อง โดยผู้จัดลำดับงานจะเห็นงานที่เข้ามาที่ละงาน โดยงานที่เข้ามาและระยะเวลาที่ทำงานแสดงไว้ที่ตารางที่ 2 และมีสัญลักษณ์ของปัญหาของการจัดลำดับงานคือ $P_2 \mid on-line \mid C_{max}$

ตารางที่ 2.1 ข้อมูลตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักรสองเครื่อง

ลำดับงาน	เวลาในการทำงาน
1	0.9
2	0.5
3	0.7
4	0.6
5	0.7
6	0.4
7	0.2
8	0.2
9	0.3
10	0.6

ลำดับงาน	เวลาในการทำงาน
1	0.9

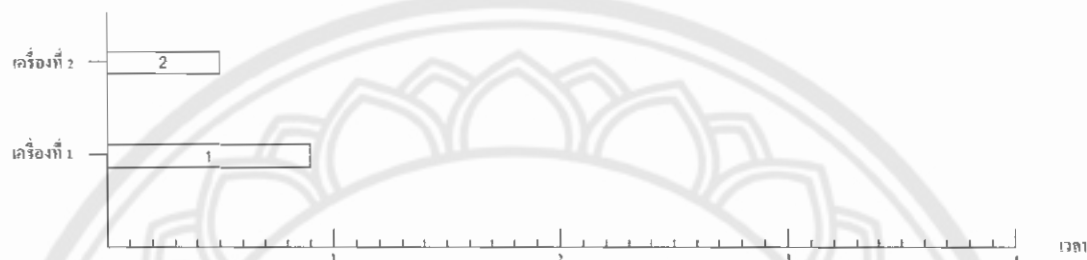
สมมติให้ผู้จัดลำดับงานนำงานที่ 1 วางไว้ ณ เครื่องที่ 1 เริ่มทำเวลาที่ 0 และทำงานที่ 1 เสร็จสิ้น ณ เวลาที่ 0.9



รูปที่ 2.1 การจัดลำดับงานที่ 1 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
2	0.5

สมมติให้ผู้จัดลำดับงานนำงานที่ 2 วางไว้ ณ เครื่องที่ 2 เริ่มทำเวลาที่ 0 และทำงานที่ 2 เสร็จสิ้น ณ เวลาที่ 0.5



รูปที่ 2.2 การจัดลำดับงานที่ 2 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
3	0.7

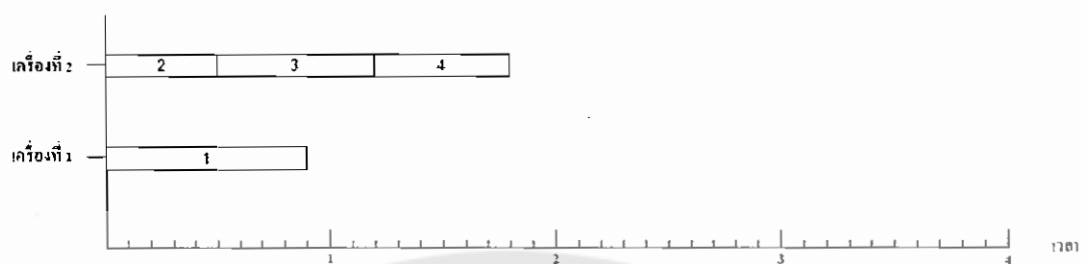
สมมติให้ผู้จัดลำดับงานนำงานที่ 3 วางไว้ ณ เครื่องที่ 2 เริ่มทำเวลาที่ 0.5 และทำงานที่ 3 เสร็จสิ้น ณ เวลาที่ 1.2



รูปที่ 2.3 การจัดลำดับงานที่ 3 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
4	0.6

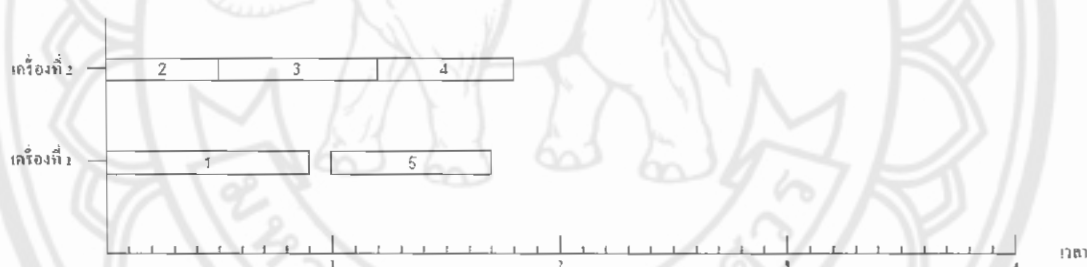
สมมติให้ผู้จัดลำดับงานนำงานที่ 4 วางไว้ ณ เครื่องที่ 2 เริ่มทำเวลาที่ 1.2 และทำงานที่ 4 เสร็จสิ้น ณ เวลาที่ 1.8



รูปที่ 2.4 การจัดลำดับงานที่ 4 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
5	0.7

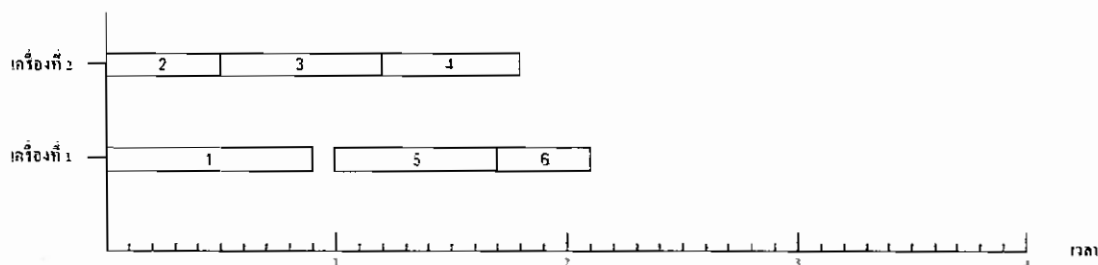
สมมติให้ผู้จัดลำดับงานนำงานที่ 5 วางไว้ ณ เครื่องที่ 1 เริ่มทำเวลาที่ 1.0 และทำงานที่ 5 เสร็จสิ้น ณ เวลาที่ 1.7



รูปที่ 2.5 การจัดลำดับงานที่ 5 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
6	0.4

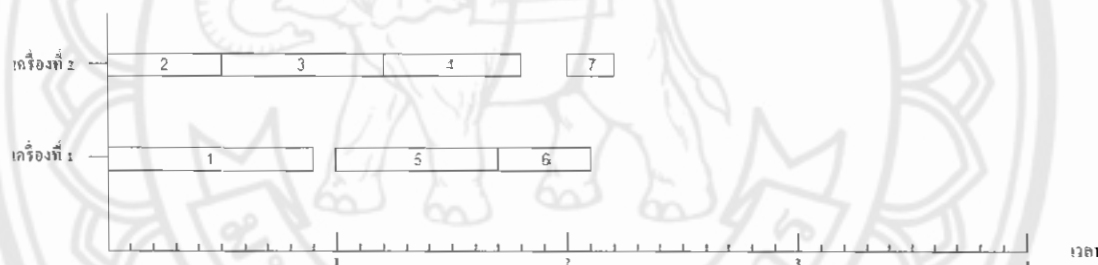
สมมติให้ผู้จัดลำดับงานนำงานที่ 6 วางไว้ ณ เครื่องที่ 1 เริ่มทำเวลาที่ 1.7 และทำงานที่ 6 เสร็จสิ้น ณ เวลาที่ 2.1



รูปที่ 2.6 การจัดลำดับงานที่ 6 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
7	0.2

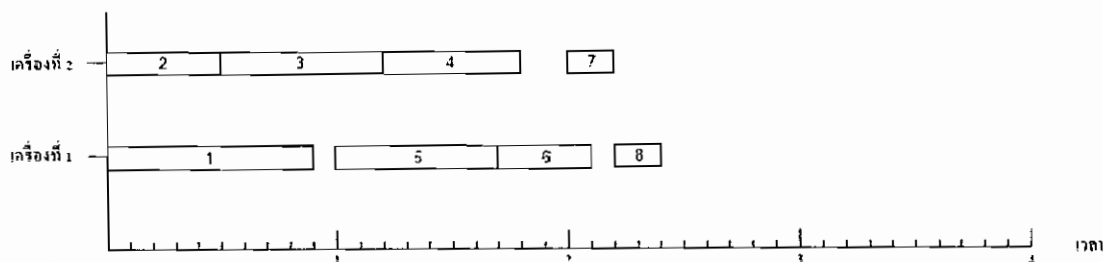
สมมติให้ผู้จัดลำดับงานนำงานที่ 7 วางไว้ ณ เครื่องที่ 2 เริ่มทำเวลาที่ 2.0 และทำงานที่ 7 เสร็จสิ้น ณ เวลาที่ 2.2



รูปที่ 2.7 การจัดลำดับงานที่ 7 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
8	0.2

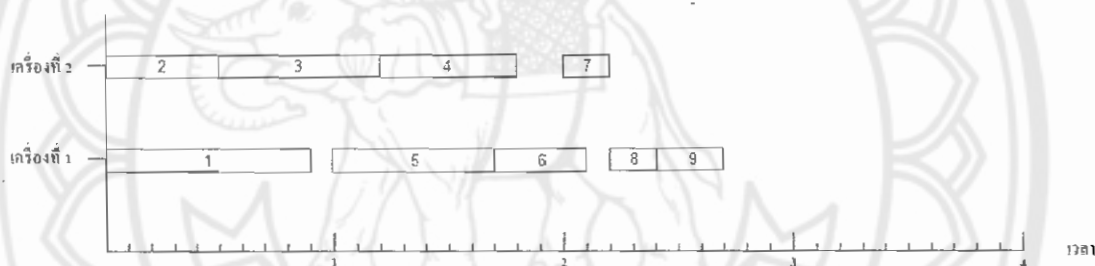
สมมติให้ผู้จัดลำดับงานนำงานที่ 8 วางไว้ ณ เครื่องที่ 1 เริ่มทำเวลาที่ 2.2 และทำงานที่ 8 เสร็จสิ้น ณ เวลาที่ 2.4



รูปที่ 2.8 การจัดลำดับงานที่ 8 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
9	0.3

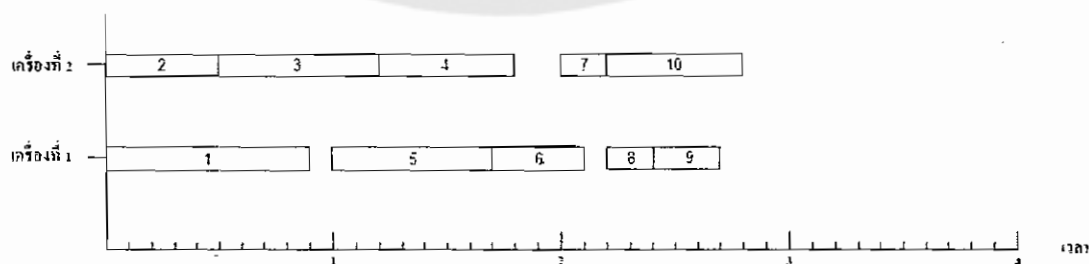
สมมติให้ผู้จัดลำดับงานนำงานที่ 9 วางไว้ ณ เครื่องที่ 1 เริ่มทำเวลาที่ 2.4 และทำงานที่ 9 เสร็จสิ้น ณ เวลาที่ 2.7



รูปที่ 2.9 การจัดลำดับงานที่ 9 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

ลำดับงาน	เวลาในการทำงาน
10	0.6

สมมติให้ผู้จัดลำดับงานนำงานที่ 10 วางไว้ ณ เครื่องที่ 2 เริ่มทำเวลาที่ 2.2 และทำงานที่ 10 เสร็จสิ้น ณ เวลาที่ 2.8



รูปที่ 2.10 การจัดลำดับงานที่ 10 ของตัวอย่างการจัดลำดับงานแบบหนึ่งต่อหนึ่งให้กับเครื่องจักร 2 เครื่อง

สรุป สามารถทำงานทั้ง 10 งานได้เสร็จภายใน เวลา 2.8 หน่วยหรือมี $C_{\max} = 2.8$

2.2.2 การจัดลำดับงานแบบมีเวลาในการปล่อยงาน

การจัดลำดับงานแบบมีเวลาในการปล่อยงานเป็นปัญหาการจัดลำดับงานที่มีสมมติฐานว่าผู้จัดลำดับงานจะทราบถึงข้อมูลของงานหลังจากเมื่อเวลาปล่อยงานผ่านไปแล้วเท่านั้น ผู้จัดลำดับงานจึงจะสามารถกำหนดได้ว่า งานถูกทำโดยทรัพยากรตัวใดและเริ่มทำงานเวลาใด

ตัวอย่างการจัดลำดับงานแบบมีเวลาในการปล่อยงานให้กับเครื่องจักรสองเครื่อง โดยผู้จัดลำดับงานจะเห็นงานที่เข้ามาที่ละงาน โดยงานที่เข้ามา เวลาที่ปล่อยงานและระยะเวลาที่ทำงานแสดงไว้ที่ตารางที่ 3 และมีสัญลักษณ์ของการจัดลำดับงานคือ $P_2 \mid R_i, on-line \mid C_{\max}$

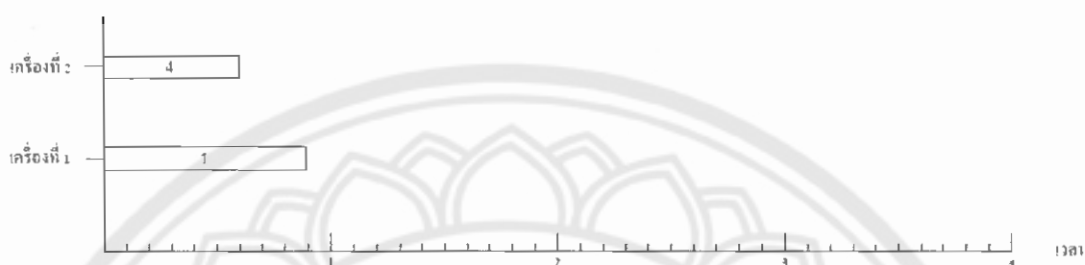
ตารางที่ 2.2 ข้อมูลตัวอย่างการจัดลำดับงานแบบมีเวลาในการปล่อยงานให้กับเครื่องจักรสองเครื่อง

ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
1	0.9	0
2	0.5	0.1
3	0.7	0.3
4	0.6	0
5	0.7	0.2
6	0.4	0.7
7	0.2	0.6
8	0.2	0.6
9	0.3	1.2
10	0.6	0.2

ที่เวลา 0.0 หน่วย

ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
1	0.9	0
4	0.6	0

สมมติให้ผู้จัดลำดับงานนำงานที่ 1 วางไว้ ณ เครื่องที่ 1 เริ่มทำเวลาที่ 0.0 และทำงานที่ 1 เสร็จสิ้น ณ เวลาที่ 0.9 และให้วางงานที่ 4 วางไว้ที่เครื่องที่ 2 เริ่มทำเวลาที่ 0.0 และทำงานที่ 4 เสร็จสิ้น ณ เวลาที่ 0.6



รูปที่ 2.11 การจัดลำดับงานที่ 1 และ 4 ของตัวอย่างการจัดลำดับงานแบบมีเวลาการปล่อยงานให้กับ เครื่องจักร 2 เครื่อง

ที่เวลา 0.1 หน่วย

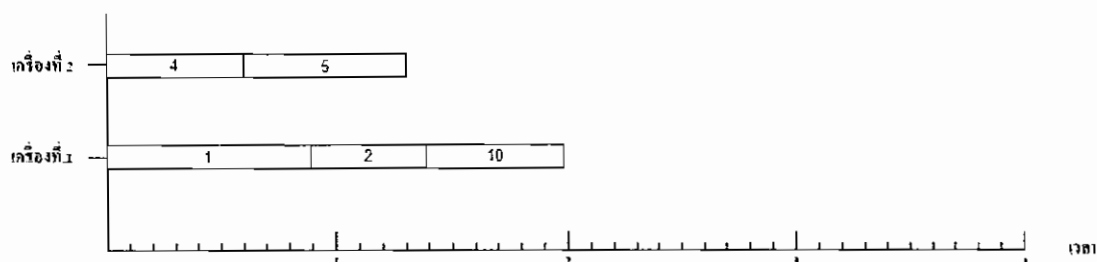
ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
2	0.5	0.1

สมมติให้ผู้จัดลำดับงานยังไม่ตัดสินใจนำงานที่ 2 ไปวาง

ที่เวลา 0.2 หน่วย

ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
2	0.5	0.1
5	0.7	0.2
10	0.6	0.2

สมมติให้ผู้จัดลำดับงานนำงานที่ 2 วางไว้ ณ เครื่องที่ 1 เริ่มทำเวลาที่ 0.9 และทำงานที่ 2 เสร็จสิ้น ณ เวลาที่ 1.4, ผู้จัดลำดับงานวางงานที่ 5 วางไว้ที่เครื่องที่ 2 เริ่มทำเวลาที่ 0.6 และทำงานที่ 5 เสร็จสิ้น ณ เวลาที่ 1.3 และผู้จัดลำดับงานให้วางงานที่ 10 วางไว้ที่เครื่องที่ 1 เริ่มทำเวลาที่ 1.4 และทำงานที่ 10 เสร็จสิ้น ณ เวลาที่ 2.0



รูปที่ 2.12 การจัดลำดับงานที่ 2, 5 และ 10 ของตัวอย่างการจัดลำดับงานแบบมีเวลาการปล่อยงานให้กับ เครื่องจักร 2 เครื่อง

ที่เวลา 0.3 หน่วย

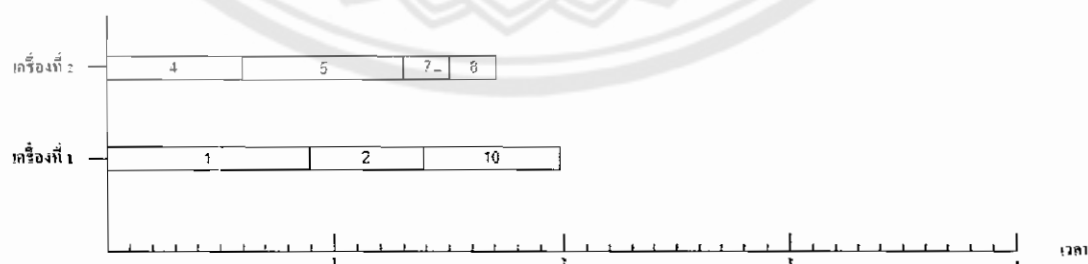
ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
3	0.7	0.3

สมมติให้ผู้จัดลำดับงานยังไม่ตัดสินใจงานที่ 3 ไปวาง

ที่เวลา 0.6 หน่วย

ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
3	0.7	0.3
7	0.2	0.6
8	0.2	0.6

สมมติให้ผู้จัดลำดับงานยังไม่ตัดสินใจงานที่ 3 ไปวาง ผู้จัดลำดับงานวางงานที่ 7 วางไว้ที่เครื่องที่ 2 เริ่มทำเวลาที่ 1.3 และทำงานที่ 7 เสร็จสิ้น ณ เวลาที่ 1.5 และผู้จัดลำดับงานให้วางงานที่ 8 วางไว้ที่เครื่องที่ 2 เริ่มทำเวลาที่ 1.5 และทำงานที่ 8 เสร็จสิ้น ณ เวลาที่ 1.7

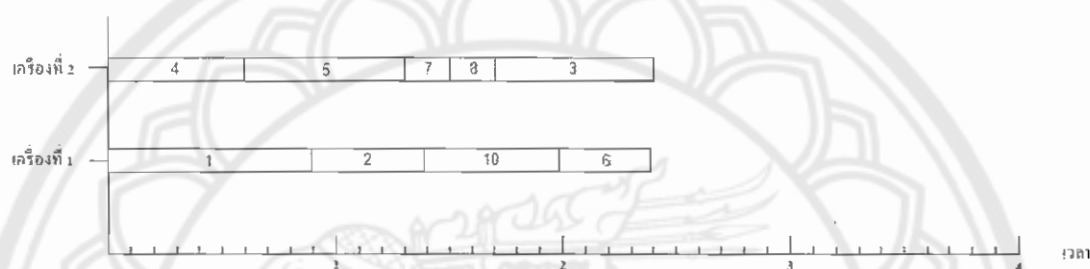


รูปที่ 2.13 การจัดลำดับงานที่ 7 และ 8 ของตัวอย่างการจัดลำดับงานแบบมีเวลาการปล่อยงานให้กับเครื่องจักร 2 เครื่อง

ที่เวลา 0.7 หน่วย

ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
3	0.7	0.3
6	0.4	0.7

สมมติให้ผู้จัดลำดับงานนำงานที่ 3 วางไว้ ณ เครื่องที่ 2 เริ่มทำเวลาที่ 1.5 และทำงานที่ 3 เสร็จสิ้น ณ เวลาที่ 2.2 และให้วางงานที่ 6 วางไว้ที่เครื่องที่ 1 เริ่มทำเวลาที่ 2.0 และทำงานที่ 6 เสร็จสิ้น ณ เวลาที่ 2.4

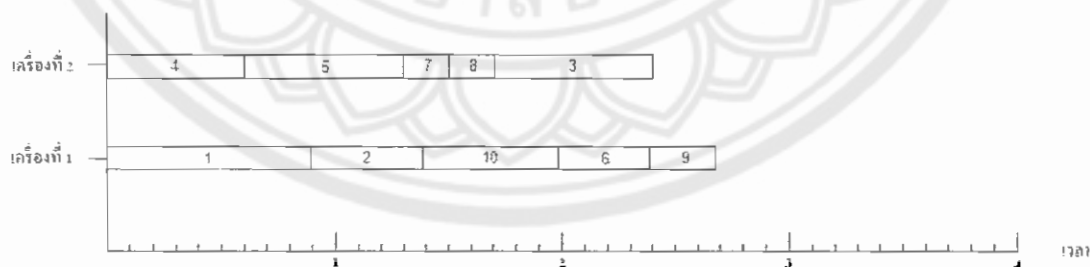


รูปที่ 2.14 การจัดลำดับงานที่ 3 และ 6 ของตัวอย่างการจัดลำดับงานแบบมีเวลาการปล่อยงานให้กับเครื่องจักร 2 เครื่อง

ที่เวลา 1.2 หน่วย

ลำดับงาน	เวลาในการทำงาน	เวลาที่ปล่อยงาน
9	0.3	1.2

สมมติให้ผู้จัดลำดับงานนำงานที่ 9 วางไว้ที่เครื่องที่ 1 เริ่มทำเวลาที่ 2.4 และทำงานที่ 9 เสร็จสิ้น ณ เวลาที่ 2.7



รูปที่ 2.15 การจัดลำดับงานที่ 9 ของตัวอย่างการจัดลำดับงานแบบมีเวลาการปล่อยงานให้กับ เครื่องจักร 2 เครื่อง

สรุป สามารถจัดลำดับงานทั้ง 10 งานในเวลา 2.7 หน่วยหรือมี $C_{\max} = 2.7$

2.3 ปัญหาการจัดลำดับงานแบบมีโซนต้องห้าม (Forbidden Zone)

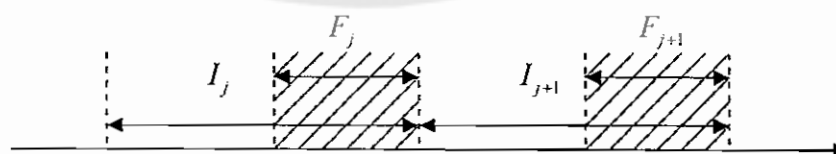
วิธีการแบบออนไลน์ได้มีความสำคัญมากขึ้นในสาขาวิชาการจัดลำดับงานในปัจจุบันปัญหาของการจัดลำดับงานจำนวนมากที่ผู้จัดลำดับงานมีข้อมูลของโจทย์ปัญหาเพียงบางส่วน แต่ต้องการผลลัพธ์ของคำตอบที่ดีและหาคำตอบนั้นได้ในเวลาอันรวดเร็วปัญหาเช่นนี้พบมากในอุตสาหกรรมการผลิตและเทคโนโลยีสารสนเทศ ในสาขาวิชาวิทยาการคอมพิวเตอร์ ก็มีปัญหามากมายที่มีงานที่จะต้องจัดลำดับให้ทำงานบน Server แต่ข้อมูลของปัญหานั้นมีไม่ครบถ้วน ส่วนในระบบการผลิตก็มีปัญหาซึ่งผู้จัดลำดับงานไม่มีข้อมูลที่ต้องเกี่ยวกับกระบวนการทำงานด้วยเหตุนี้เองปัญหาการจัดลำดับงานบางส่วนจึงถูกพิจารณาแก้ปัญหาในแบบ On-line มากขึ้น

ที่มาของโครงงานวิจัยชิ้นนี้มาจากการงานของ Abdekhodae และ Ernst ในการแก้ปัญหาห่วงโซ่อุปทานโดยในปัญหานี้กิจกรรมบางอย่าง (การจอดเรือ) ไม่สามารถเริ่มต้นทำกิจกรรมนั้นได้ภายในช่วงเวลาหนึ่ง ๆ ที่กำหนดไว้

ในปัญหานี้เราจะกำหนดให้มิงาน n งานที่จะต้องถูกจัดลำดับได้แก่ $J_1, \dots, J_n$ และให้ p_i เป็นเวลาในการทำงานของ J_i นอกจากนี้ยังมีสมมติฐานว่า $p_i \leq 1 \forall i$ เราจะแบ่งเวลาออกเป็นช่วง $I = \{I_1, I_2, \dots, I_s\}$ โดย $I_1 = [0, 2]$, $I_2 = [2, 4]$, $I_s = [2s-2, 2s]$ ในแต่ละ I_j จะมีช่วงเวลาโซนต้องห้าม $F_j \subset I_j, \forall j$ โดย $F_1 = (1, 2]$, $F_2 = (3, 4]$... $F_s = (2s-1, 2s]$ และจะเรียก $I_j \setminus F_j$ ว่าโซนอนุญาต

ช่วงโซนต้องห้ามจะเป็นช่วงเวลาซึ่งงานไม่สามารถเริ่มต้นได้ นอกจากนี้ถ้ามีงานงานหนึ่งถูกทำให้เสร็จสิ้นก่อนที่จะสิ้นสุดโซนต้องห้าม งานนี้จะถูกปล่อยออกไปเมื่อตอนเริ่มต้นของช่วงเวลาถัดไป

จุดประสงค์จัดทำปัญหานี้ คือ สร้างตารางการทำงานที่ทำให้มีจำนวนช่วงที่ใช้มีค่าต่ำสุด (ถ้าโซนต้องห้ามสุดท้ายมีงานถูกทำอยู่นั้น จุดประสงค์เช่นนี้จะเท่าเทียมกับการหาค่า makespan ต่ำสุด)



รูปที่ 2.16 แสดงช่วงเวลา I และโซนต้องห้าม F

Abdekhodae และ Ernst ได้แสดงให้เห็นว่าปัญหาการจัดลำดับงานนี้ในกรณีช่วงเวลา (Interval) ทุกช่วงมีขนาดเท่ากันและโซนต้องห้ามทุกโซนมีขนาดเท่ากัน เป็นปัญหา NP hard

ปัญหาการจัดลำดับงานที่มีโหนดต้องห้ามนี้คล้ายคลึงกับปัญหาการจัดของลงถัง (bin packing) ซึ่งหากพิจารณาจากแบบจำลองการโปรแกรมเชิงจำนวนเต็มของทั้งสองปัญหาจะเห็นความคล้ายคลึงชัดเจนยิ่งขึ้น เริ่มจากการพิจารณาปัญหา bin packing ก่อน สมมติว่าเรามีของอยู่ n ชิ้นที่มีขนาด $p_1, \dots, p_n$ โดยมีสมมติฐานว่า $p_i \leq 1 \forall i$ ถ้าเราให้ $x_{ij} = 1$ ถ้าของชิ้นที่ i ถูกใส่ลงในถังถึง j และ $x_{ij} = 0$ ในกรณีอื่นๆ นอกจากนั้นยังกำหนดตัวแปร $z_j = 1$ ถ้าถัง j ถูกใช้ และ $z_j = 0$ ในกรณีอื่นๆ ปัญหา bin packing จะสามารถเขียนในรูปของการโปรแกรมจำนวนเต็มได้ดังนี้

$$\min \sum_{j=1}^n z_j \quad (1)$$

$$\sum_{j=1}^n x_{ij} = 1, \quad \forall i \quad (2)$$

$$x_{ij} \leq z_j, \quad \forall i, j \quad (3)$$

$$\sum_{i=1}^n p_i x_{ij} \leq 1, \quad \forall j \quad (4)$$

$$x_{ij}, z_j \in \{0, 1\} \quad (5)$$

สำหรับปัญหาการจัดลำดับงานที่มีโหนดต้องห้าม กำหนดให้ $x_{ij} = 1$ ถ้างาน i เริ่มและเสร็จสิ้นในช่วงอนุญาตและ $x_{ij} = 0$ ในกรณีอื่นๆ และให้ $z_j = 1$ ถ้าช่วง j ถูกใช้งาน และ $z_j = 0$ ในกรณีอื่นๆ โดยกำหนดตัวแปรเพิ่มเติมที่เกี่ยวข้องกับโหนดต้องห้ามโดยให้ $y_{ij} = 1$ ถ้างาน i เริ่มต้นในช่วง j และเสร็จสิ้นในโหนดต้องห้าม และ $y_{ij} = 0$ ในกรณีอื่นๆ เราจะสามารถเขียนการจัดโหนดต้องห้ามได้ดังนี้

$$\min \sum_{j=1}^n z_j \quad (6)$$

$$\sum_{j=1}^n (x_{ij} + y_{ij}) = 1, \quad \forall i \quad (7)$$

$$x_{ij} + y_{ij} \leq z_j, \quad \forall i, j \quad (8)$$

$$\sum_{i=1}^n p_i x_{ij} \leq 1, \quad \forall j \quad (9)$$

$$\sum_{i=1}^n y_{ij} \leq 1, \quad \forall j \quad (10)$$

$$x_{ij}, y_{ij}, z_j \in \{0, 1\} \quad (11)$$

สำหรับปัญหาการจัดลำดับงานในโซนต้องห้ามแบบ On-line นี้เราจะพิจารณาในเฉพาะรูปแบบ One by One หมายความว่างานทุกงานจะมาพร้อมกันที่เวลา 0 แต่ผู้จัดลำดับงานจะมองเห็นงานเหล่านี้ได้ทีละหนึ่งงาน เมื่อผู้จัดลำดับงานได้กำหนดการทำงานของงานที่เห็นแล้วก็จะสามารถมองเห็นและได้รับข้อมูลของงานถัดไป

2.4 ศึกษาการจัดลำดับงานแบบ On-Line ในปัญหาที่มีโซนต้องห้าม

จากงานวิจัยของ K. Khammuang และคณะ ในการพัฒนาวิธีการจัดลำดับงานแบบ On-Line สำหรับปัญหาที่มีโซนต้องห้าม (Forbidden Zone) โดยได้เสนอไว้ 4 วิธี ดังต่อไปนี้

2.4.1 List scheduling

List scheduling เป็นวิธีการจัดเรียงงานอย่างง่ายโดยจะจัดเรียงงานตามตารางงาน กล่าวคือเมื่อมีงานเข้ามาก็นำงานนั้นเข้าไปใส่ใน Allowed Zone จนเต็มแล้วงานถัดมาเข้ามาใส่ใน Forbidden Zone ที่อยู่ติดกันเมื่อนำงานมาใส่ใน Forbidden Zone แล้วจะปิดช่วงเวลานั้นแล้วจะทำการเปิดช่วงเวลาใหม่ดังตัวอย่างต่อไปนี้

ตามโจทย์ปัญหามีสัญลักษณ์ปัญหาการจัดลำดับงานคือ $1|On-line|C_{max}$

ตารางที่ 2.3 ข้อมูลระยะเวลาที่ใช้ในการปฏิบัติงานในแต่ละงานของตัวอย่างการจัดลำดับงานในปัญหา On-line โดยใช้วิธีการจัดลำดับงานแบบ List scheduling

งานที่	ระยะเวลาที่ใช้ปฏิบัติงาน(หน่วย)
1	0.7
2	0.9
3	0.3
4	0.4
5	0.5
6	0.4
7	0.8

งานที่ 1

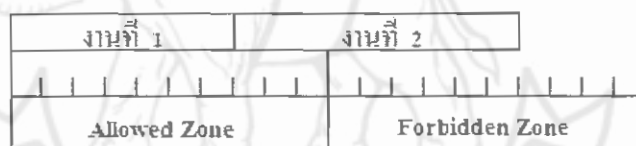
ผู้จัดลำดับงานได้นำงานที่มาวางที่ช่วง Allowed Zone ที่ 1 โดยวางที่จุดเริ่มต้นและผู้จัดทำ
ทำงานที่หนึ่งเสร็จสิ้น ณ เวลา 0.7



รูปที่ 2.17 จัดลำดับงานของงานที่ 1 ของปัญหาการจัดลำดับงานแบบ List scheduling

งานที่ 2

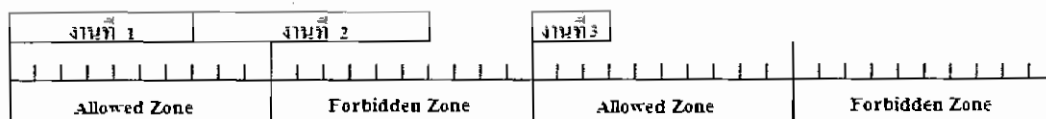
ผู้จัดลำดับงานได้นำงานที่มาวางที่ช่วง Allowed Zone ที่ 1 โดยวางไว้ ณ เวลาที่ 0.7 และ
ผู้จัดทำทำงานที่ 2 เสร็จสิ้น ณ เวลา 1.6



รูปที่ 2.18 จัดลำดับงานของงานที่ 2 ของปัญหาการจัดลำดับงานแบบ List scheduling

งานที่ 3

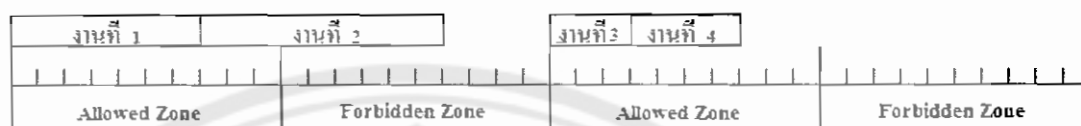
จะต้องทำการเปิดช่วงเวลาใหม่เนื่องจากไม่สามารถเข้าไปทำงานยัง Forbidden Zone จึง
ต้องนำงานที่ 3 มาใส่ไว้ ณ จุดเริ่มต้นของ Allowed Zone ที่ 2 ผู้จัดทำทำงานที่ 3 เสร็จสิ้น ณ เวลาที่
2.3



รูปที่ 2.19 จัดลำดับงานของงานที่ 3 ของปัญหาการจัดลำดับงานแบบ List scheduling

งานที่ 4

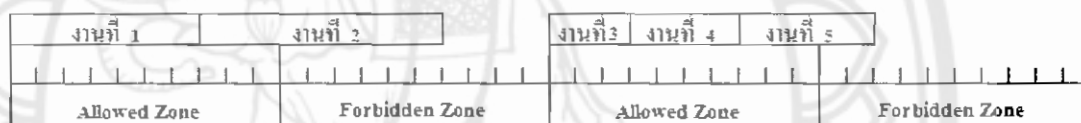
ผู้จัดลำดับงานได้นำงานที่ 4 มาวางในช่วง Allowed Zone ที่ 2 โดยวาง ณ เวลาที่ 2.3 และ
ผู้จัดทำทำงานที่ 4 เสร็จสิ้น ณ เวลาที่ 2.7



รูปที่ 2.20 จัดลำดับงานของงานที่ 4 ของปัญหาการจัดลำดับงานแบบ List scheduling

งานที่ 5

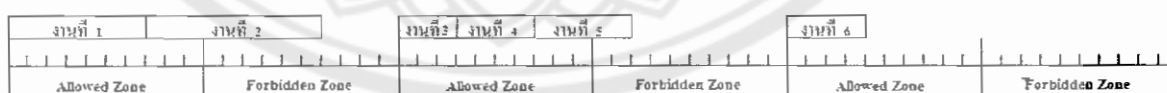
ผู้จัดลำดับงานได้นำงานที่ 5 มาวางในช่วง Allowed Zone ที่ 2 โดยวาง ณ เวลาที่ 2.7 และ
ผู้จัดทำทำงานที่ 5 เสร็จสิ้น ณ เวลาที่ 3.2



รูปที่ 2.21 จัดลำดับงานของงานที่ 5 ของปัญหาการจัดลำดับงานแบบ List scheduling

งานที่ 6

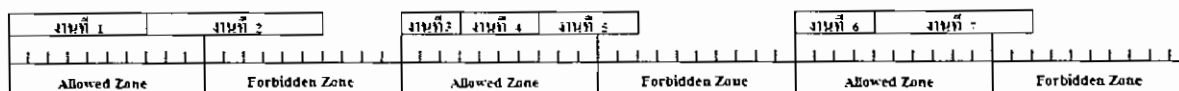
จะต้องทำการเปิดช่วงเวลาใหม่เนื่องจากไม่สามารถเข้าไปทำงานยัง Forbidden Zone จึง
ต้องนำงานที่ 6 มาใส่ไว้ ณ จุดเริ่มต้นของ Allowed Zone ที่ 3 และผู้จัดทำทำงานที่ 6 เสร็จสิ้น ณ
เวลาที่ 4.4



รูปที่ 2.22 จัดลำดับงานของงานที่ 6 ของปัญหาการจัดลำดับงานแบบ List scheduling

งานที่ 7

ผู้จัดลำดับงานได้นำงานที่ 7 มาวางในช่วง Allowed Zone ที่ 3 โดยวาง ณ เวลาที่ 4.4 และ
ผู้จัดทำทำงานที่ 7 เสร็จสิ้น ณ เวลาที่ 5.2



รูปที่ 2.23 จัดลำดับงานของงานที่ 7 ของปัญหาการจัดลำดับงานแบบ List scheduling

ในตัวอย่างการจัดลำดับงานแบบ List scheduling จะสามารถจัดลำดับงานทั้ง 7 งานได้เสร็จสิ้นภายใน 3 ช่วงเวลาหรือมีค่า Makespan เท่ากับ 5.2

2.4.2 Half Next Fit algorithm (HNF)

ถ้ามีงานเข้ามางานนั้นจะสามารถนำมาใน Allowed zone ได้แต่ถ้างานนั้นมีขนาดใหญ่กว่า หรือเท่ากับ $1/2$ ของช่วงเวลา Allowed zone จะนำงานนั้นมาใส่ใน Forbidden zone แล้วจึงปิดช่วง Allowed zone และ Forbidden zone นั้น ถ้างานต่อไปมีขนาดเล็กกว่า $1/2$ ของช่วงเวลาให้เปิด Allowed zone ขึ้นมาใหม่และนำงานนั้นมาใส่ในช่วงนี้ถึงแม้ Allowed zone ของช่วงที่แล้วจะมีที่ว่างเหลือก็ตามดังตัวอย่างที่จะแสดงต่อไปนี้

ตามโจทย์ปัญหามีสัญลักษณ์ปัญหาการจัดลำดับงานคือ $1|On-line|C_{max}$

ตารางที่ 2.4 ข้อมูลระยะเวลาที่ใช้ในการปฏิบัติงานในแต่ละงานของตัวอย่างการจัดลำดับงานในปัญหา On-line โดยใช้วิธีการจัดลำดับงานแบบ Half Next Fit Algorithm

งานที่	ระยะเวลาที่ใช้ปฏิบัติงาน(หน่วย)
1	0.7
2	0.9
3	0.3
4	0.4
5	0.5
6	0.4
7	0.8

งานที่ 1

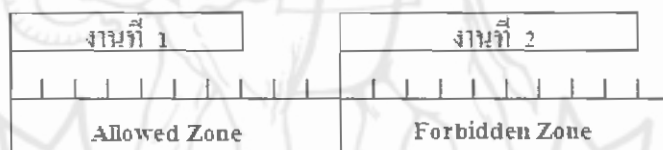
ผู้จัดลำดับงานได้นำงานที่มาวางในช่วง Allowed Zone ที่ 1 โดยวางที่จุดเริ่มต้นและผู้จัดทำ
ทำงานที่หนึ่งเสร็จสิ้น ณ เวลา 0.7



รูปที่ 2.24 แสดงจัดลำดับงานของงานที่ 1 ของปัญหาการจัดลำดับงานแบบ Half Next Fit Algorithm

งานที่ 2

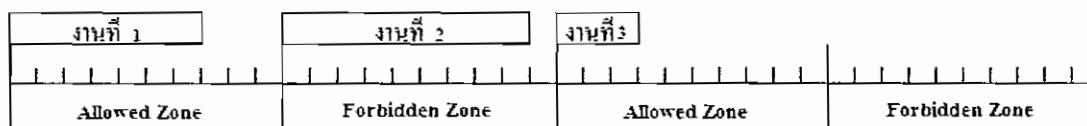
งานที่ 2 มีขนาดใหญ่กว่า $\frac{1}{2}$ ของ Allowed zone ตามหลักเกณฑ์จึงต้องนำมาวางไว้ใน
Forbidden zone ที่ 1 โดยเริ่ม ณ เวลา 1.0 และผู้จัดลำดับงานทำงานที่ 2 ได้สิ้นสุด ณ เวลาที่ 1.9



รูปที่ 2.25 จัดลำดับงานของงานที่ 2 ของปัญหาการจัดลำดับงานแบบ Half Next Fit Algorithm

งานที่ 3

งานที่ 3 ถึงแม้จะมีขนาดที่สามารถจะใส่ใน Allowed zone ที่ 1 ได้แต่ตามหลักเกณฑ์เมื่อนำ
งานไปไว้ใน Forbidden zone แล้ว จะต้องปิดช่วงเวลานั้นจึงทำให้ต้องนำงานที่ 3 มาใส่ไว้ใน
Allowed zone ที่ 2 โดยผู้จัดลำดับงานได้วางงานที่ 3 โดยเริ่ม ณ เวลา 2.0 และผู้จัดลำดับงาน
ทำงานที่ 3 ได้สิ้นสุด ณ เวลาที่ 2.3



รูปที่ 2.26 จัดลำดับงานของงานที่ 3 ของปัญหาการจัดลำดับงานแบบ Half Next Fit Algorithm

งานที่ 4

งานที่ 4 มีขนาดเล็กกว่า $\frac{1}{2}$ ของ Allowed zone สามารถนำมาวางต่อจากงานที่ 3 ได้ โดยเริ่ม ณ เวลา 2.3 และผู้จัดลำดับงานทำงานที่ 4 ได้สิ้นสุด ณ เวลาที่ 2.7

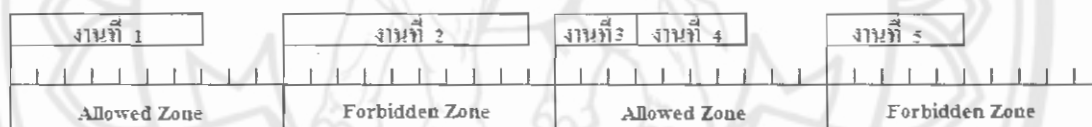


รูปที่ 2.27 จัดลำดับงานของงานที่ 4 ของปัญหาการจัดลำดับงานแบบ Half Next Fit

Algorithm

งานที่ 5

งานที่ 5 มีขนาดเท่ากับ $\frac{1}{2}$ ของ Allowed zone จึงไม่สามารถอยู่ใน Allowed zone ได้จึงต้องนำไปวางไว้ใน Forbidden zone 3 โดยเริ่ม ณ เวลา 3.0 และผู้จัดลำดับงานทำงานที่ 5 ได้สิ้นสุด ณ เวลาที่ 2.5

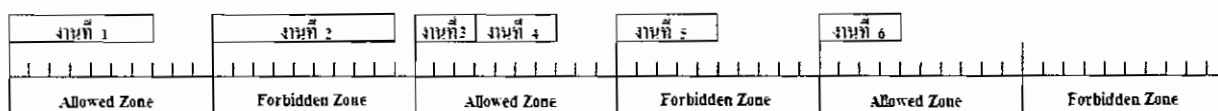


รูปที่ 2.28 จัดลำดับงานของงานที่ 5 ของปัญหาการจัดลำดับงานแบบ Half Next Fit

Algorithm

งานที่ 6

งานที่ 6 มีขนาดเล็กกว่า $\frac{1}{2}$ ของ Allowed zone แต่เราใช้ Forbidden zone ไปแล้วจึงต้องเปิดช่วงเวลาใหม่ขึ้นมาแล้วนำงานที่ 6 มาวางไว้ใน Allowed zone ที่ 3 โดยเริ่ม ณ เวลา 4.0 และผู้จัดลำดับงานทำงานที่ 6 ได้สิ้นสุด ณ เวลาที่ 4.4

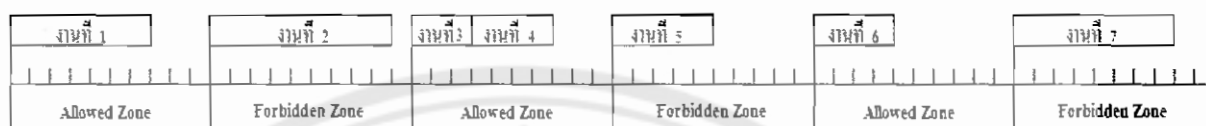


รูปที่ 2.29 จัดลำดับงานของงานที่ 6 ของปัญหาการจัดลำดับงานแบบ Half Next Fit

Algorithm

งานที่ 7

งานที่ 7 มีขนาดใหญ่กว่า $\frac{1}{2}$ ของ Allowed zone ตามหลักเกณฑ์จึงต้องนำมาวางไว้ที่ Forbidden zone ที่ 3 โดยเริ่ม ณ เวลา 5.0 และผู้จัดลำดับงานทำงานที่ 7 ได้สิ้นสุด ณ เวลาที่ 5.8



รูปที่ 2.30 จัดลำดับงานของงานที่ 7 ของปัญหาการจัดลำดับงานแบบ Half Next Fit

Algorithm

ในตัวอย่างการจัดลำดับงานแบบ Half Next Fit algorithm จะสามารถจัดลำดับงานทั้ง 7 งานได้เสร็จสิ้นภายใน 3 ช่วงเวลาหรือมีค่า Makespan เท่ากับ 5.8

2.4.3 $\frac{1}{4}$ Next Fit algorithm (1/4NF)

ถ้ามีงานเข้ามางานนั้นจะสามารถนำมาใน Allowed zone ได้แต่ถ้างานนั้นมีขนาดใหญ่กว่า หรือเท่ากับ $\frac{1}{4}$ ของช่วงเวลา Allowed zone จะนำงานนั้นมาใส่ใน Forbidden zone แล้วจึงปิดช่วง Allowed zone และ Forbidden zone นั้น ถ้างานต่อไปมีขนาดเล็กกว่า $\frac{1}{4}$ ของช่วงเวลาให้เปิด Allowed zone ขึ้นมาใหม่และนำงานนั้นมาใส่ในช่วงนี้ถึงแม้ Allowed zone ของช่วงที่แล้วจะมีที่ว่างเหลือก็ตาม

2.4.4 $\frac{3}{4}$ Next Fit algorithm (3/4NF)

ถ้ามีงานเข้ามางานนั้นจะสามารถนำมาใน Allowed zone ได้แต่ถ้างานนั้นมีขนาดใหญ่กว่า หรือเท่ากับ $\frac{3}{4}$ ของช่วงเวลา Allowed zone จะนำงานนั้นมาใส่ใน Forbidden zone แล้วจึงปิดช่วง Allowed zone และ Forbidden zone นั้น ถ้างานต่อไปมีขนาดเล็กกว่า $\frac{3}{4}$ ของช่วงเวลาให้เปิด Allowed zone ขึ้นมาใหม่และนำงานนั้นมาใส่ในช่วงนี้ถึงแม้ Allowed zone ของช่วงที่แล้วจะมีที่ว่างเหลือก็ตาม

2.4.5 Half First Fit algorithm (HFF)

ถ้ามีงานเข้ามาและงานนั้นมีขนาดเล็กกว่า $\frac{1}{2}$ ช่วงเวลาของ Allowed zone ให้นำงานนั้นใส่ใน Allowed zone และถ้างานนั้นมีขนาดใหญ่กว่า $\frac{1}{2}$ ของช่วงเวลาจะนำงานนั้นมาใส่ใน Forbidden zone นั้น ถ้างานต่อไปมีขนาดเล็กกว่าช่วงเวลา Allowed zone ในช่วงแรก



สำนักหอสมุด

HD

69

154

ข458ค

3551

เหลืออยู่ให้นำกลับไปได้ใน Allowed zone ของช่วงแรกได้ โดยวิธีนี้จะไม่มีการปิดช่วงเวลาเหมือน
Half Next Fit algorithm โดยจะแสดงในตัวอย่างดังนี้

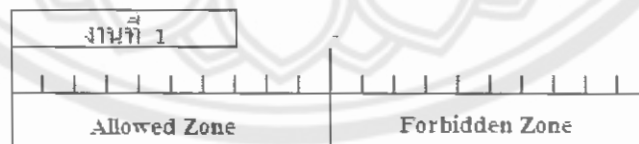
ตามโจทย์ปัญหามีสัญลักษณ์ปัญหาการจัดลำดับงานคือ $1|On-line|C_{max}$

ตารางที่ 2.5 ข้อมูลระยะเวลาที่ใช้ในการปฏิบัติงานในแต่ละงานของตัวอย่างการจัดลำดับ
งานในปัญหาOn-line โดยใช้วิธีการจัดลำดับงานแบบ Half First Fit algorithm

งานที่	ระยะเวลาที่ใช้ปฏิบัติงาน(หน่วย)
1	0.7
2	0.9
3	0.3
4	0.4
5	0.5
6	0.4
7	0.8

งานที่ 1

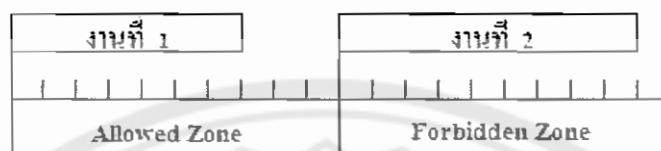
ผู้จัดลำดับงานได้นำงานที่มาวางที่ช่วง Allowed Zone ที่ 1 โดยวางที่จุดเริ่มต้นและผู้จัดทำ
ทำงานที่หนึ่งเสร็จสิ้น ณ เวลา 0.7



รูปที่ 2.31 จัดลำดับงานของงานที่ 1 ของปัญหาการจัดลำดับงานแบบ Half First Fit
algorithm

งานที่ 2

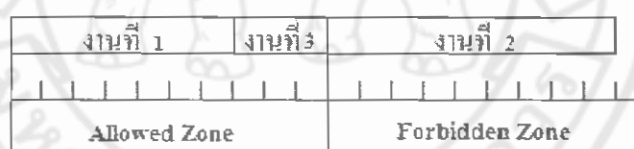
งานที่ 2 มีขนาดใหญ่กว่า $\frac{1}{2}$ ของ Allowed zone ตามหลักเกณฑ์จึงต้องนำมาวางไว้ใน Forbidden zone ที่ 1 โดยเริ่ม ณ เวลา 1.0 และผู้จัดลำดับงานทำงานที่ 2 ได้สิ้นสุด ณ เวลาที่ 1.9



รูปที่ 2.32 จัดลำดับงานของงานที่ 2 ของปัญหาการจัดลำดับงานแบบ Half First Fit algorithm

งานที่ 3

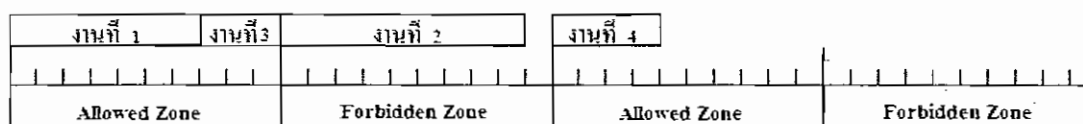
งานที่ 3 มีขนาดเท่ากับช่วงเวลาของ Allowed zone ที่ 1 ที่เหลืออยู่จึงสามารถนำงานที่ 3 วางไว้ใน Allowed zone ที่ 1 ได้ โดยเริ่ม ณ เวลา 0.7 และผู้จัดลำดับงานทำงานที่ 3 ได้สิ้นสุด ณ เวลาที่ 1.0



รูปที่ 2.33 จัดลำดับงานของงานที่ 3 ของปัญหาการจัดลำดับงานแบบ Half First Fit algorithm

งานที่ 4

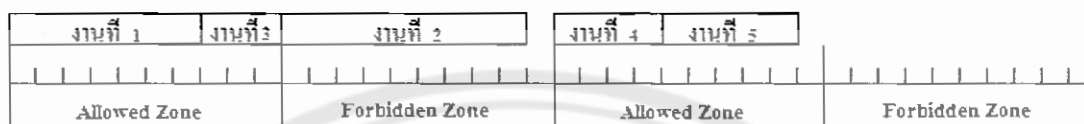
งานที่ 4 มีขนาดเล็กกว่า $\frac{1}{2}$ ของ Allowed zone แต่เนื่องจากช่วงเวลาก่อนนั้นได้ใช้ Allowed zone จนหมดแล้วจึงต้องเปิดช่วงเวลาคืนมาใหม่จึงสามารถนำงานที่ 4 มาวางใน Allowed zone ที่ 2 โดยเริ่ม ณ เวลา 2.0 และผู้จัดลำดับงานทำงานที่ 4 ได้สิ้นสุด ณ เวลาที่ 2.4



รูปที่ 2.34 จัดลำดับงานของงานที่ 4 ของปัญหาการจัดลำดับงานแบบ Half First Fit algorithm

งานที่ 5

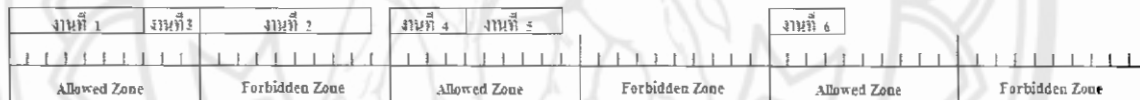
งานที่ 5 สามารถอยู่ใน Allowed zone ได้จึงต้องนำไปวางไว้ที่ Allowed zone ที่ 2 ต่อจากงานที่ 4 โดยเริ่ม ณ เวลา 2.4 และผู้จัดลำดับงานทำงานที่ 5 ได้สิ้นสุด ณ เวลาที่ 2.9



รูปที่ 2.35 จัดลำดับงานของงานที่ 5 ของปัญหาการจัดลำดับงานแบบ Half First Fit algorithm

งานที่ 6

งานที่ 6 มีขนาดเล็กกว่า $\frac{1}{2}$ ของ Allowed zone ไม่สามารถอยู่ใน Forbidden zone จึงต้องเปิดช่วงเวลาขึ้นมาใหม่และนำงานที่ 6 วางไว้ที่ Allowed zone ที่ 3 โดยเริ่ม ณ เวลา 4.0 และผู้จัดลำดับงานทำงานที่ 6 ได้สิ้นสุด ณ เวลาที่ 4.6



รูปที่ 2.36 จัดลำดับงานของงานที่ 6 ของปัญหาการจัดลำดับงานแบบ Half First Fit algorithm

งานที่ 7

งานที่ 7 มีขนาดใหญ่กว่า $\frac{1}{2}$ ของ Allowed zone สามารถอยู่ใน Forbidden zone จึงนำงานที่ 7 มาวางไว้ที่ Forbidden zone ที่ 2 โดยเริ่ม ณ เวลา 4.0 และผู้จัดลำดับงานทำงานที่ 7 ได้สิ้นสุด ณ เวลาที่ 4.8



รูปที่ 2.37 จัดลำดับงานของงานที่ 7 ของปัญหาการจัดลำดับงานแบบ Half First Fit algorithm

ในตัวอย่างการจัดลำดับงานแบบ Half First Fit algorithm จะสามารถจัดลำดับงานทั้ง 7 งานได้เสร็จสิ้นภายใน 3 ช่วงเวลาหรือมีค่า Makespan เท่ากับ 4.4

2.4.6 $\frac{1}{4}$ First Fit algorithm (1/4FF)

ถ้ามีงานเข้ามาและงานนั้นมีขนาดเล็กกว่า $\frac{1}{4}$ ช่วงเวลาของ Allowed zone ให้นำงานนั้นใส่ใน Allowed zone และถ้างานนั้นมีขนาดใหญ่กว่า $\frac{1}{4}$ ของช่วงเวลาจะนำงานนั้นมาใส่ใน Forbidden zone นั้น ถ้างานต่อไปมีขนาดเล็กกว่าช่วงเวลา Allowed zone ในช่วงแรกที่เหลืออยู่ให้นำกลับไปใส่ใน Allowed zone ของช่วงแรกได้

2.4.7 $\frac{3}{4}$ First Fit algorithm (3/4FF)

ถ้ามีงานเข้ามาและงานนั้นมีขนาดเล็กกว่า $\frac{3}{4}$ ช่วงเวลาของ Allowed zone ให้นำงานนั้นใส่ใน Allowed zone และถ้างานนั้นมีขนาดใหญ่กว่า $\frac{3}{4}$ ของช่วงเวลาจะนำงานนั้นมาใส่ใน Forbidden zone นั้น ถ้างานต่อไปมีขนาดเล็กกว่าช่วงเวลา Allowed zone ในช่วงแรกที่เหลืออยู่ให้นำกลับไปใส่ใน Allowed zone ของช่วงแรกได้

2.4.8 Half Harmonic M algorithm (HHM)

Half Harmonic M algorithm เป็นการจัดลำดับงานโดยจะแบ่งช่วงเวลาในแต่ละช่วงเวลาให้บรรจุนงานที่มีเวลาการทำงานต่างกันไปโดยกำหนดให้ M เป็นจำนวนช่วงเวลาย่อยโดย $I_j = (1/(j+1), 1/j]$ ($1 \leq j < M$) และ $I_M = (0, 1/M]$ โดยจะแสดงไว้ในตัวอย่างดังนี้

ตารางที่ 2.6 ข้อมูลระยะเวลาที่ใช้ในการปฏิบัติงานในแต่ละงานของตัวอย่างการจัดลำดับงานในปัญหา On-line โดยใช้วิธีการจัดลำดับงานแบบ Half Harmonic M algorithm

งานที่	ระยะเวลาที่ใช้ปฏิบัติงาน(หน่วย)
1	0.7
2	0.9
3	0.3
4	0.4
5	0.5

งานที่	ระยะเวลาที่ใช้ปฏิบัติงาน(หน่วย)
6	0.4
7	0.8

โดยกำหนดให้ $M = 3$ จะได้

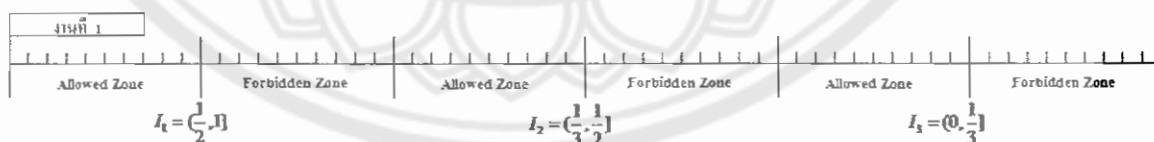
$$I_M = (0, (1/M)] = I_3 = (0, \frac{1}{3}]$$

$$I_j = (1/(j+1), (1/j)] = I_2 = (\frac{1}{3}, \frac{1}{2}]$$

$$I_j = (1/(j+1), (1/j)] = I_1 = (\frac{1}{2}, 1]$$

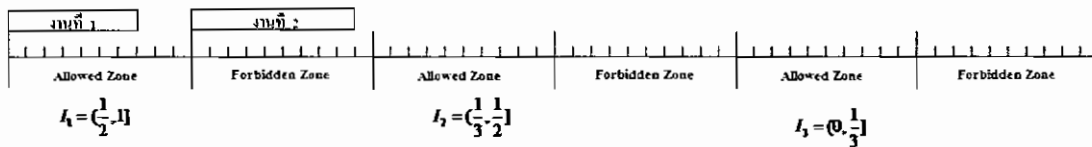
เพราะฉะนั้นตัวอย่างนี้จะมี 3 ช่วงเวลา คือช่วงเวลา I_1 จะสามารถบรรจุงานที่มีระยะเวลาในการทำงานตั้งแต่ $\frac{1}{2}$ หน่วยเวลาของเวลา Allowed zone ถึง 1 หน่วยเวลาของ Allowed zone, ช่วงเวลา I_2 จะสามารถบรรจุงานที่มีระยะเวลาในการทำงานตั้งแต่ $\frac{1}{3}$ หน่วยเวลาของเวลา Allowed zone ถึง $\frac{1}{2}$ หน่วยเวลาของเวลา Allowed zone และช่วงเวลา I_3 จะสามารถบรรจุงานที่มีระยะเวลาในการทำงานตั้งแต่ 0 หน่วยเวลาของเวลา Allowed zone ถึง $\frac{1}{3}$ หน่วยเวลาของเวลา Allowed zone

งานที่ 1 มีระยะเวลาทำงานในช่วงเวลา I_1 จึงนำงานที่ 1 ใส่ไว้ใน Allowed zone ที่ 1 ของช่วงเวลา I_1



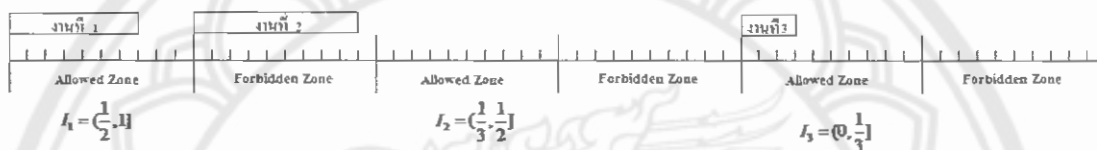
รูปที่ 2.38 จัดลำดับงานของงานที่ 1 ของปัญหาการจัดลำดับงานแบบ Half Harmonic M algorithm

งานที่ 2 มีระยะเวลาทำงานในช่วงเวลา I_1 จึงนำงานที่ 2 ใส่ไว้ใน Forbidden zone ที่ 1 ของช่วงเวลา I_1



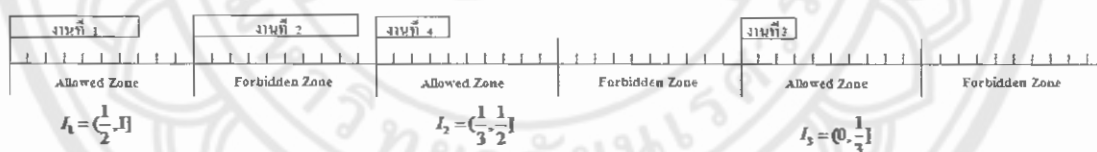
รูปที่ 2.39 จัดลำดับงานของงานที่ 2 ของปัญหาการจัดลำดับงานแบบ Half Harmonic M algorithm

งานที่ 3 มีระยะเวลาทำงานในช่วงเวลา I_3 จึงนำงานที่ 3 ใส่ไว้ใน Allowed zone ที่ 1 ของช่วงเวลา I_3



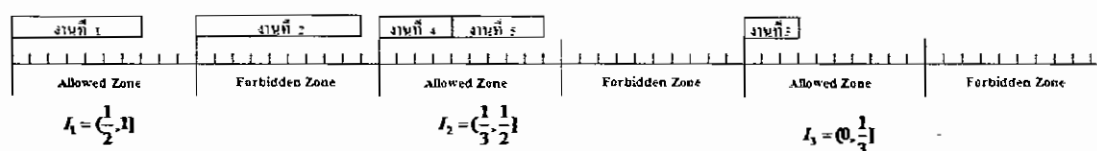
รูปที่ 2.40 จัดลำดับงานของงานที่ 3 ของปัญหาการจัดลำดับงานแบบ Half Harmonic M algorithm

งานที่ 4 มีระยะเวลาทำงานในช่วงเวลา I_2 จึงนำงานที่ 4 ใส่ไว้ใน Allowed zone ที่ 1 ของช่วงเวลา I_2



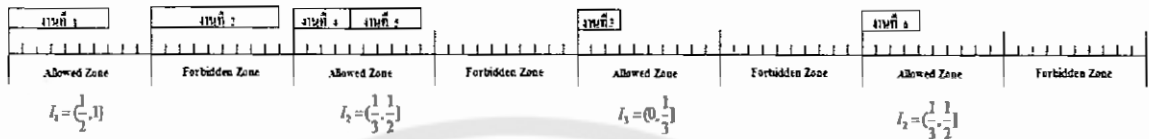
รูปที่ 2.41 แสดงจัดลำดับงานของงานที่ 4 ของปัญหาการจัดลำดับงานแบบ Half Harmonic M algorithm

งานที่ 5 มีระยะเวลาทำงานในช่วงเวลา I_2 จึงนำงานที่ 5 ใส่ไว้ใน Allowed zone ที่ 1 ของช่วงเวลา I_2 โดยวางต่อจากงานที่ 4



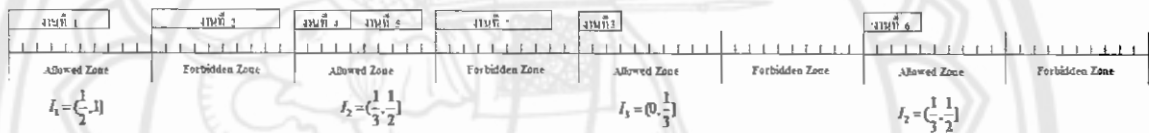
รูปที่ 2.42 จัดลำดับงานของงานที่ 5 ของปัญหาการจัดลำดับงานแบบ Half Harmonic M algorithm

งานที่ 6 มีระยะเวลาทำงานในช่วงเวลา I_2 แต่ช่วงเวลา I_2 นั้นเต็มแล้วจึงเปิดช่วงเวลา I_2 ขึ้นมาใหม่แล้วจึงนำงานที่ 6 ใส่ไว้ใน Allowed zone ของช่วงเวลา I_2 ที่ 2



รูปที่ 2.43 จัดลำดับงานของงานที่ 6 ของปัญหาการจัดลำดับงานแบบ Half Harmonic M algorithm

งานที่ 7 มีระยะเวลาทำงานในช่วงเวลา I_1 แต่เนื่องจากช่วงเวลาแรกของช่วงเวลา I_1 ใช้หมดแล้ว จึงนำงานที่ 7 ใส่ไว้ใน Forbidden zone ที่ 1 ของช่วงเวลา I_2



รูปที่ 2.44 จัดลำดับงานของงานที่ 7 ของปัญหาการจัดลำดับงานแบบ Half Harmonic M algorithm

ในตัวอย่างการจัดลำดับงานแบบ Half Harmonic M algorithm จะสามารถจัดลำดับงานทั้ง 7 งานได้เสร็จสิ้นภายใน 4 ช่วงเวลา

2.4.9 $\frac{1}{4}$ Harmonic M algorithm

$\frac{1}{4}$ Harmonic M algorithm เป็นการจัดลำดับงานโดยจะแบ่งช่วงเวลาในแต่ละช่วงเวลาให้บรรจุนงานที่มีเวลาการทำงานต่างกันไปโดยกำหนดให้ M เป็นจำนวนช่วงเวลาย่อยโดย $I_j = (1/(j+1), (1/j)]$ ($1 \leq j < M$) และ $I_M = (0, (1/M)]$ ส่วนงานที่สามารถเข้ามาทำได้ใน Forbidden zone จะมีขนาดตั้งแต่ $\frac{1}{4}$ ของ Allowed zone

2.4.10 $\frac{3}{4}$ Harmonic M algorithm

$\frac{3}{4}$ Harmonic M algorithm เป็นการจัดลำดับงานโดยจะแบ่งช่วงเวลาในแต่ละช่วงเวลาให้บรรจุนงานที่มีเวลาการทำงานต่างกันไปโดยกำหนดให้ M เป็นจำนวนช่วงเวลาย่อยโดย $I_j = (1/(j+1), (1/j)]$ ($1 \leq j < M$) และ $I_M = (0, (1/M)]$ ส่วนงานที่สามารถเข้ามาทำได้ใน Forbidden zone จะมีขนาดตั้งแต่ $\frac{3}{4}$ ของ Allowed zone

2.5 การวัดประสิทธิภาพของอัลกอริทึมแบบ On-Line

เป็นการยากที่จะสามารถวัดประสิทธิภาพของวิธีการแก้ปัญหาแบบ On-line ได้ แต่ก็ยังมีนักวิจัยกลุ่มหนึ่งได้กำหนดตัววัดประสิทธิภาพของวิธีการแบบ On-line ในกรณีที่ย่ำแย่ที่สุดซึ่งเรียกว่า อัตราส่วนสมรรถนะการแข่งขัน Competitive ratio โดยสำหรับวิธีการแก้ปัญหาแบบ On-line A ใด ๆ และโจทย์ปัญหา I ใด ๆ เราจะให้ $A(I)$ เป็นค่าของฟังก์ชันเป้าหมายที่ได้จากการใช้วิธีการแก้ปัญหาแบบ On-line A กับโจทย์ปัญหา I และให้ $OPT(I)$ เป็นค่าของฟังก์ชันเป้าหมายที่ให้คำตอบที่ดีที่สุดสำหรับโจทย์ปัญหา I ถ้า $A(I)/OPT(I) \leq c$ สำหรับโจทย์ปัญหา I ทุก ๆ โจทย์ปัญหา เราจะเรียกว่า อัตราส่วนสมรรถนะการแข่งขันของ A มีค่าเท่ากับ c

ตารางที่ 2.7 แสดงค่าของ อัตราส่วนสมรรถนะการแข่งขัน (Competitive ratio) ของการจัดลำดับงานแบบ List scheduling, Half Next Fit algorithm, Half First Fit algorithm และ Haft Harmonic M algorithm

วิธีการ	ค่า c
List scheduling	2
Half Next Fit algorithm	3
Half First Fit algorithm	$2.5 \leq C \leq 2.7$
Haft Harmonic M algorithm	1.923

2.6 งานวิจัยที่เกี่ยวข้องกับทำเรื่อง

จากงานวิจัยของ K. Khammuang และคณะ (2007) ได้ทำการในการพัฒนาวิธีการจัดลำดับงานแบบ On-line สำหรับปัญหาที่มีโซนต้องห้าม(Forbidden Zone)โดยได้เสนอไว้ 4 วิธีคือ List scheduling, Half Next Fit algorithm, Half First Fit algorithm และ Haft Harmonic M algorithm โดยทั้ง 4 วิธีนี้สามารถวัดความสามารถของการจัดลำดับงานได้โดยใช้อัตราส่วนสมรรถนะการแข่งขัน (Competitive ratio) ค่าของ Competitive ratio จะให้ค่าที่ดีที่สุดค่านั้นจะต้องมีค่าน้อยที่สุดซึ่งตารางที่ 2 ได้แสดงไว้ เห็นได้ว่าวิธีแบบ Haft Harmonic M algorithm ให้ค่า Competitive ratio ที่น้อยที่สุดและเป็นค่าที่ดีที่สุดคือ 1.923 และวิธี Half Next Fit algorithm ให้ค่า Competitive ratio ที่มากที่สุดคือ 3

2.7 การจำลองสถานการณ์ (Simulation)

เนื่องจากอุตสาหกรรมปัจจุบันเป็นอุตสาหกรรมที่มีความซับซ้อน และมีตัวแปรที่มีความไม่แน่นอนเกี่ยวข้องอยู่ด้วยเสมอ ตัวอย่างเช่น อุตสาหกรรมการขนส่ง อุตสาหกรรมการผลิต อุตสาหกรรมการให้บริการ ซึ่งอุตสาหกรรมเหล่านี้มีความไม่แน่นอนของการเข้ามาของลูกค้า ความไม่แน่นอนของเวลาในการให้บริการ ความไม่แน่นอนของตารางการทำงาน เกี่ยวข้องอยู่ด้วย จึงเป็นการยากในการบริหารงานและจัดสรรทรัพยากรต่าง ๆ เพื่อให้บรรลุวัตถุประสงค์ของอุตสาหกรรมนั้น ๆ ได้ ดังนั้นการสร้างแบบจำลองด้วยโปรแกรมคอมพิวเตอร์จึงเป็นทางเลือกหนึ่งในการศึกษาพฤติกรรมของระบบอีกทั้งการสร้างแบบจำลองด้วยโปรแกรมคอมพิวเตอร์ยังสะดวกในการเปลี่ยนแปลงตัวแบบจำลอง เพื่อสร้างแนวทางในการปรับปรุงระบบให้มีประสิทธิภาพ โดยปราศจากการรบกวนงานในระบบจริง

การจำลองสถานการณ์ Simulation คือ การเลียนแบบกระบวนการหรือระบบที่มีอยู่แล้วในชีวิตจริง โดยศึกษาว่าระบบเมื่อมีเวลาเปลี่ยนแปลงไปสมรรถภาพและประสิทธิภาพของระบบเปลี่ยนแปลงไปอย่างไร

ในการศึกษาว่าระบบเปลี่ยนไปเมื่อเวลาเปลี่ยนไป สามารถศึกษาได้โดยการพัฒนาระบบหรือสร้างระบบจำลอง Simulation ขึ้นมา โดยที่การจำลองมักมีสมมติฐานว่าระบบทำงานอย่างไร เช่น สมมติฐานทางคณิตศาสตร์ ตรรกศาสตร์

เมื่อมีการสร้างระบบจำลองหรือพัฒนาแบบจำลอง แล้วยืนยันแล้วว่าแบบจำลองเหมือนของจริง จะสามารถนำแบบจำลองมาตรวจสอบและประเมินประสิทธิภาพของระบบ

การแก้ปัญหาเพื่อศึกษาระบบเกี่ยวกับสมรรถนะระบบเมื่อเวลาเปลี่ยนไป โดยนำแบบจำลอง Simulation ที่แปลผลเสร็จแล้วในข้อมูลที่เป็น Output เปรียบเทียบกับประสิทธิภาพของระบบจริง โดยผลที่ได้จากการทำ Simulation นำมาประเมินตัววัดประสิทธิภาพของระบบ

ในปัญหาการจัดลำดับงานการจำลองสถานการณ์เป็นอีกวิธีหนึ่งที่ช่วยให้ผู้ปฏิบัติงานสามารถมองเห็นแนวโน้มของสถานการณ์ที่กำลังจะเกิดขึ้นในภายภาคหน้า ทำให้ผู้ปฏิบัติงานสามารถเลือกใช้วิธีการหรือเลือกใช้เครื่องในการจัดลำดับงานให้เกิดผลที่ดีที่สุด

2.8 Visual Basic for Application (VBA)

Visual Basic for Application หรือ VBA เป็นชุดคำสั่งแม่แบบวิซวลเบสิกในโปรแกรมชุดไมโครซอฟท์ออฟฟิศ (Microsoft Office) ของบริษัทไมโครซอฟท์ที่ได้รับความนิยมใช้กันมากในปัจจุบัน ซึ่งประกอบไปด้วยโปรแกรมเวิร์ด เพาเวอร์พอยต์ เอ็กเซล เอาต์ลुक และแอกเซส ในงานวิจัยนี้ผู้วิจัยเลือกใช้โปรแกรมไมโครซอฟท์เอ็กเซล (Microsoft Excel) ซึ่งเป็นโปรแกรมประเภท

ตารางคำนวณหรือสเปรดชีต (Spreadsheet) ซึ่งเหมาะกับผู้ใช้ที่ต้องการเขียนโปรแกรมสร้างระบบงานแบบอัตโนมัติ ลดขั้นตอนการทำงานที่ซับซ้อน แก้ปัญหางานที่ซับซ้อนต่าง ๆ โดยการเขียนโปรแกรมด้วย VBA บนเอ็กเซลโดยตรง

Visual Basic for Application เป็นเครื่องมือขึ้นหนึ่งที่สามารถนำมาเขียนโปรแกรมในการจัดลำดับงาน สำหรับงานวิจัยนี้ได้นำ VBA เข้ามาใช้ในการจำลองสถานการณ์ของงานที่เข้ามา ซึ่งการเข้ามาของงานแต่ละงานนั้น ผู้วิจัยจะทำการเขียนโปรแกรมให้การเข้ามาของงานแต่ละงานมีการแจกแจงแบบต่าง ๆ เช่น การแจกแจงแบบปกติ การแจกแจงแบบเอ็กโพเนนเชียล เป็นต้น

