รายการอ้างอิง

ภาษาไทย

ชอุ่ม พลอยมีค่า. 2523. การจัดการผลิต. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร: บริษัทซีเอ็ด ยูเคชั่น.

พิชิต สุขเจริญพงษ์. 2533. การบริหารการผลิต. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร: บริษัทซีเอ็ด ยูเคชั่น.

พิภพ เล้าประจง. 2529. ระบบควบคุมการผลิตเชิงวิศวกรรม. กรุงเทพมหานคร: สมาคมส่งเสริมเทคโนโลยี่(ไทย-ญี่ปุ่น).

ศิริจันทร์ ทองประเสริฐ. 2532. การจำลองแบบปัญหา. พิมพ์ครั้งที่ 2. กรุงเทพมหานคร: จุฬาลงกรณ์มหาวิทยาลัย.

_____. 2534. การวางแผนและควบคุมการผลิตด้วยคอมพิวเตอร์. วิศวกรรมสาร ฉบับเทคโนโลยี 2: 68-72.

สุรศักดิ์ นานานุกูล. 2517. การบริหารงานผลิต. พิมพ์ครั้งที่ 2. กรุงเทพมหานคร: บริษัทสำนักพิมพ์ ไทยวัฒนาพาณิชย์.

ภาษาอังกฤษ

Adam, E.E. and Ebert, R.J. 1989. Production and Operation Management. 4th ed. NJ: Prentice Hall.

Ben-arieh, D. and Moodie, C.L. 1987. Knowledge Based Routine and Sequencing for Discrete Part Production. Journal of Manufacturing System 6: 287-297.

Browne, J. 1988. Producction Activity Control - A key of Production Control. International of Production Research 26: 415-427.

Copas, C. and Browne, J. 1990. A Rules-based Scheduling System for Flow Type Assembly. International of Production

Research 28: 981-1005.

Day, J.E. and Hottenstein, M.P. 1970. Review of Sequencing Research. Naval Research and Logistics Quarterly 17: 11-39.

Ebner, M.L. and Vollmann, T.E. 1988. Manufacturing Systems for the 1990's. Benjamin/Cummings Publishing Company. 317-336.

Fogarty, R.W., Hoffmann, T.S. and Stonebraker, P.W. 1989. Production and Operation Management. South-western Publishing.

Godin, V.B. 1978. Introduction Scheduling : Historical Servey and State of the Art. AIIE Transaction 10: 33-37.

Hadavi, K., Shahraray, M.S. and Voigt, K. 1990. REDS - A Dynamic Planning, Scheduling and Control System for Manufacturing. Journal of Manufacturing Systems 9: 332-244.

Huang, Philip, Y. 1984. A Comparative Study of Priority Dispatching Rules in a Hybrid Assembly/Jobshop. International Journal of Production Research 22: 375-387.

Kang, B. and Markland, R.E. 1989. Solving the No-intermidiate Storage Flowshop Scheduling Problem. International Journal of Production Management 9: 48-59.

Laio, C. and Huang, R. 1991. An Algorithm for Minimizing the Range of Lateness on Single Machine. Journal of Operation Research Society 42: 183-186.

McComas, M.G. and Law, A.M. 1992. How to Select Simulation Software for Manufacturing Applications. Industrial Engineering No.7: 29-35.

Melnyk, S.A. 1988. Production Control : Issues and Challenges. Benjamin/Cummings Publishing Company. 199-232.

Nawaz, M., Enscore Jr, E.E. and Ham, I. 1983. A Heuristic Algorithm for the M-machine, N-job Flowshop Sequencing Problem. OMEGA. The International of Management Science 11: 91-95.

Pegden, D.C. 1985. Introduction to SIMAN with version 3.0 Enhanements. PA: Systems Modeling Corporation.

Philipoom, P.R. and FRY, T.D. 1990. The Robustness of Selected Job-shop Dispatching Rules with Respect to Load Balance and Work-flow Structure. Journal of Operation Research Society 41: 897-906.

Rodammer, F. and White, K.P. 1988. A Recent Survey of Production Scheduling. IEEE Transaction on Systems, Man and Cybernatics 18: 841-851.

Smith, S.B. 1989. Computer-based Production and Inventory Control. NJ: Prentice Hall.

Smith, R.L. and Platt, H. 1987. Benefit of Animation in the Simulation of a Machine and Assembly line. Simulation 48: 28-30.

Systems Modeling Corporation. 1988. CINEMA/EGA System for Simulation and Animation. PA: System Modeling Corporation.

ภาคผนวก ก

รายละเอียดเกี่ยวกับกรณีศึกษา

กรณีศึกษานี้เป็นงานวิจัยของ Nawaz et al.(1983) เรื่อง "A Heuristic Algorithm for the M-machine, N-job Sequencing Problem" โดยมีวัตถุประสงค์เพื่อต้องการเปรียบเทียบการจัดลำดับงานที่มีเวลาทั้งหมดที่งานอยู่ในระบบ ที่มากกว่าให้ทำก่อนงานที่มีเวลาทั้งหมดที่งานอยู่ในระบบที่น้อยกว่า กับการจัดลำดับงานโดยวิธีอื่นๆ อีก 15 วิธี

จากกรณีศึกษาสามารถสรุปพอสังเขปได้คือปัญหาการจัดลำดับงาน คือการกำหนดลำดับการผลิตให้กับงานต่างๆ ที่จะต้องผ่านจากเครื่องจักรหนึ่งไปยังอีกเครื่องจักรหนึ่ง ซึ่งเสนอการแก้ปัญหาแนว Heuristic มีชื่อว่า New Curtailed-enumeration โดยมีขั้นตอนการนำเสนอดังนี้

1. การกำหนดลักษณะปัญหา

   - มี N งาน M เครื่องจักรแต่ละงานจะต้องผ่านเครื่องจักรตามลำดับ
   - ให้ $T_{i,j}$ เป็น เวลางานที่ i ทำอยู่บนเครื่องจักร j

     โดยที่ $i = 1,2,3,\ldots,N$

     $j = 1,2,3,\ldots,M$

2. สมมุติฐาน

   - งานที่มีเวลาทั้งหมดที่งานอยู่ในระบบที่มากกว่าจะมีลำดับความสำคัญมากกว่างานที่มีเวลาทั้งหมดที่งานอยู่ในระบบที่น้อยกว่า

3. หลักการ

   - เลือก 2 งานที่มีเวลาทั้งหมดที่งานอยู่ในระบบมากที่สุดจาก N งาน โดยใช้ Exhaustive Search เลือกการจัดลำดับส่วนย่อยที่ดีที่สุด ความสัมพันธ์ของตำแหน่งงานทั้ง 2 ที่ถูกเลือกนี้จะกำหนดตายตัวในขั้นตอนวิธี

- เลือกงานที่มีเวลาทั้งหมดที่งานอยู่ในระบบอันดับ 3 จาก N งานใช้ Exhaustive Search เลือกการจัดลำดับส่วนย่อยที่ดีที่สุด ความสัมพันธ์ของตำแหน่งงานทั้ง 3 ที่ถูกเลือกนี้จะกำหนดตายตัวในขั้นตอนวิธี ในขั้นตอนนี้จะทำซ้ำๆ จนกระทั่งได้การจัดลำดับที่สมบูรณ์

4. วิธีการ

- ขั้นตอนที่ 1 หา $T_i = \sum_{i=1}^{m} t_{i,j}$

- ขั้นตอนที่ 2 เรียงลำดับงานตาม T_i จากมากไปหาน้อย

- ขั้นตอนที่ 3 เลือก 2 งานแรกจากการเรียงลำดับในขั้นตอนที่ 2 เลือกการลำดับงาน 2 งานที่ดีที่สุดโดยคำนวณค่า Makespan ของ 2 ลำดับที่เป็นไปได้ กำหนดให้ i = 3

- ขั้นตอนที่ 4 เลือกงานที่ i ในลำดับการเรียงในขั้นตอนที่ 2 และหาการลำดับที่ดีที่สุด โดยการวางงานนั้นในตำแหน่งที่ 1 ถึง i ที่ได้จากขั้นตอนที่แล้ว โดยไม่มีการสลับตำแหน่งของงานที่ได้กำหนดไว้แล้ว

- ขั้นตอนที่ 5 ถ้า i = N หยุด

กรณีอื่น กำหนดให้ i = i + 1

5. การแสดงทางตัวเลข

ตัวอย่างปัญหา

งาน	เครื่องจักร					T_i
	1	2	3	4	5	
1	5	9	8	10	1	33
2	9	3	10	1	8	31
3	9	4	5	8	6	32
4	4	8	8	7	2	29

- ขั้นตอนที่ 1 T_1 = 33

  T_2 = 31

  T_3 = 32

  T_4 = 29

- ขั้นตอนที่ 2 จัดลำดับงาน คือ 1--3--2--4

- ขั้นตอนที่ 3 แสดงค่า Makespan ของลำดับที่เป็นได้

งาน	เครื่องจักร				
	1	2	3	4	5
1	5/5	9/14	8/22	10/32	1/33
3	9/14	4/18	5/27	8/40	6/46
3	9/9	4/13	5/18	8/26	6/32
1	5/14	9/23	8/31	10/41	1/42

เลือก การจัดลำดับ 3--1

กำหนด i = 3

- ขั้นตอนที่ 4 ลำดับที่เป็นไปได้คือ

ลำดับงาน 3--1--2 ให้ค่า Makespan = 50

ลำดับงาน 3--2--1 ให้ค่า Makespan = 51

ลำดับงาน 2--3--1 ให้ค่า Makespan = 51

เลือก การจัดลำดับงาน 3--1--2

- ขั้นตอนที่ 5

กำหนด i = 4

- ขั้นตอนที่ 4 ลำดับที่เป็นไปได้คือ

  ลำดับงาน 3--1--2--4 ให้ค่า Makespan = 58

  ลำดับงาน 3--1--4--2 ให้ค่า Makespan = 58

  ลำดับงาน 3--4--1--2 ให้ค่า Makespan = 57

  ลำดับงาน 4--3--1--2 ให้ค่า Makespan = 54

  เลือก การจัดลำดับงาน 4--3--1--2 ซึ่งมี Makespan ต่ำสุด

- ขั้นตอนที่ 5

  กำหนด i = 5 หยุด

6. การสรุปผล

Nawaz et al. ได้สรุปเมื่อเปรียบเทียบกับวิธี Heuristic ต่างๆเช่น CDS เป็นต้น วิธี New Curtailed-enumeration เป็นวิธีที่มีประสิทธิภาพที่ดี และให้ผลลัพธ์ที่ใกล้เคียงกับผลลัพธ์ที่ดีที่สุดด้วย

ภาคผนวก ข

ขั้นตอนวิธีกฎลำดับความสำคัญ

<u>การกำหนดค่าตัวแปรต่างๆ</u>

1. มีงาน N งาน
2. มีเครื่องจักร M เครื่อง
3. JOB(n,m)

   คือ งาน n ที่ เครื่องจักร m
4. DD(n)

   คือ กำหนดวันส่งงานของงาน n
5. ARRIVAL-TIME-JOB(n,m)

   คือ เวลาที่สามารถเริ่มงาน n ที่เครื่องจักร m
6. PROCESS-TIME-JOB(n,m)

   คือ เวลาที่ใช้ในการปฏิบัติของงาน n ที่เครื่องจักร m

   คูณด้วยปริมาณงานของงานนั้น
7. FINISH-TIME-JOB(n,m)

   คือ เวลาที่สิ้นสุดของงาน n ที่เครื่องจักร m
8. S(i,j)

   คือ เลขที่งานที่ได้เรียงลำดับตามกฎแล้ว
9. ARRIVAL-TIME-S(i,j)

   คือ เวลาที่สามารถเริ่มงานที่เรียงลำดับแล้วของงาน i

   ที่เครื่องจักร j
10. FINISH-TIME-S(i,j)

    คือ เวลาที่สิ้นสุดของงานที่เรียงลำดับแล้วของงาน i ที่เครื่องจักร j

ขั้นตอนวิธีกฎลำดับความสำคัญ

1. กฎเข้าก่อนบริการก่อน

ขั้นตอนที่ 1 เริ่มพิจารณาการจัดลำดับของเครื่องจักรที่ 1

ให้ m = 1

ขั้นตอนที่ 2 หาเวลาที่สามารถเริ่มงาน n ที่เครื่องจักรที่ m

ให้ n = 1

ขั้นตอนที่ 2.1

ถ้า m = 1 ไปทำขั้นตอนที่ 2.2

ถ้า m <> 1

ให้ ARRIVAL-TIME-JOB(n,m) = FINISH-TIME-JOB(n,m-1)

ขั้นตอนที่ 2.2

ให้ n = n + 1

ขั้นตอนที่ 2.3

ถ้า n <= N ไปทำขั้นตอนที่ 2.1

ถ้า n > N ไปทำขั้นตอนที่ 3

ขั้นตอนที่ 3 เรียงลำดับงานตามเวลาที่งานสามารถเริ่มงานได้ ก่อน-หลัง ที่เครื่องจักรหนึ่งๆจากน้อยไปหามาก

ขั้นตอนที่ 4 กำหนดเลขที่งานที่เรียงลำดับแล้วให้กับ S(i,j)

ให้ j = m

ให้ S(1,j) = เลขที่งานที่มี ARRIVAL-TIME-JOB(n,m) น้อยที่สุด

. .

. .

ให้ S(n,j) = เลขที่งานที่มี ARRIVAL-TIME-JOB(n,m) มากที่สุด

ขั้นตอนที่ 5 หาเวลาที่สิ้นสุดของงานใดๆที่เครื่องจักรหนึ่งๆ

ให้ i = 1

ขั้นตอนที่ 5.1 หา ARRIVAL-TIME-S(i,j)

ถ้า i = 1 และ j = 1

ให้ ARRIVAL-TIME-S(1,1) = ARRIVAL-TIME-JOB(S(1,1),1)

ไปทำขั้นตอนที่ 5.2

ถ้า i = 1

ให้ ARRIVAL-TIME-S(1,j) = FINISH-TIME-JOB(S(1,j),j-1)

ไปทำขั้นตอนที่ 5.2

ถ้า j = 1

ให้ ARRIVAL-TIME-S(i,1) =

MAX[FINISH-TIME-S(i-1,1),

ARRIVAL-TIME-JOB(S(i,1),1)]

ไปทำขั้นตอนที่ 5.2

ถ้า i <> 1 และ j <> 1

ให้ ARRIVAL-TIME-S(i,j) =

MAX[FINISH-TIME-JOB(S(i,j),m-1),

FINISH-TIME-S(i-1,j)]

ขั้นตอนที่ 5.2 หา FINISH-TIME-S(i,j) และ FINISH-TIME-JOB(n,m)

ให้ FINISH-TIME-S(i,j) = ARRIVAL-TIME-S(i,j) +

PROCESS-TIME-JOB(S(i,j),m)

ให้ FINISH-TIME-JOB(S(i,j),m) = FINISH-TIME-S(i,j)

ขั้นตอนที่ 5.3

ให้ i = i + 1

ขั้นตอนที่ 5.4

ถ้า i <= N ไปทำขั้นตอนที่ 5.1

ถ้า i > N ไปทำขั้นตอนที่ 6

ขั้นตอนที่ 6 เริ่มพิจารณาการจัดลำดับของเครื่องจักรถัดไป

ให้ m = m + 1

ขั้นตอนที่ 7

ถ้า m <= M ไปทำขั้นตอนที่ 4

ถ้า m > M สิ้นสุดการทำงาน

2. <u>กฎเวลาปฏิบัติงานสั้นที่สุด</u>

ขั้นตอนที่ 1 เริ่มพิจารณาการจัดลำดับของเครื่องจักรที่ 1

ให้ m = 1

ขั้นตอนที่ 2 เรียงลำดับงานตามเวลาที่จะใช้ในการปฏิบัติงานใดๆที่เครื่องจักรหนึ่งๆ จากน้อยไปหามาก

ขั้นตอนที่ 3 กำหนดเลขที่งานที่เรียงลำดับแล้วให้กับ S(i,j)

ให้ j = m

ให้ S(1,j) = เลขที่งานที่มี PROCESS-TIME-JOB(n,m) น้อยที่สุด

. .

. .

. .

ให้ S(n,j) = เลขที่งานที่มี PROCESS-TIME-JOB(n,m) มากที่สุด

ขั้นตอนที่ 4 หาเวลาที่สิ้นสุดของงานใดๆที่เครื่องจักรหนึ่งๆ

ให้ i = 1

ขั้นตอนที่ 4.1 หา ARRIVAL-TIME-S(i,j)

ถ้า i = 1 และ j = 1

ให้ ARRIVAL-TIME-S(1,1) = ARRIVAL-TIME-JOB(S(1,1),1)

ไปทำขั้นตอนที่ 4.2

ถ้า i = 1

ให้ ARRIVAL-TIME-S(1,j) = FINISH-TIME-JOB(S(1,j),j-1)

ไปทำขั้นตอนที่ 4.2

ถ้า j = 1

ให้ ARRIVAL-TIME-S(i,1) =

MAX[FINISH-TIME-S(i-1,1),

ARRIVAL-TIME-JOB(S(i,1),1)]

ไปทำขั้นตอนที่ 4.2

ถ้า i <> 1 และ j <> 1 และ

FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า >=

FINISH-TIME-S ของเครื่องจักรปัจจุบัน

ให้ ARRIVAL-TIME-S(i,j) =

FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า

ถ้า i <> 1 และ j <> 1 และ

FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า <

FINISH-TIME-S ของเครื่องจักรปัจจุบัน

ให้นำงานที่เหลืออยู่เฉพาะงานที่ FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า มีค่าน้อยกว่า FINISH-TIME-S ของเครื่องจักรปัจจุบันลำดับงานก่อนหน้า มาเรียงลำดับงานใหม่ตามเวลาที่ใช้ในการปฏิบัติงานจากน้อยไปหามาก และ

ให้ ARRIVAL-TIME-S(i,j) =

FINISH-TIME-S ของเครื่องจักรปัจจุบันลำดับงานก่อนหน้า

ขั้นตอนที่ 4.2 หา FINISH-TIME-S(i,j) และ FINISH-TIME-JOB(n,m)

ให้ FINISH-TIME-S(i,j) = ARRIVAL-TIME-S(i,j) +

PROCESS-TIME-JOB(S(i,j),m)

ให้ FINISH-TIME-JOB(S(i,j),m) = FINISH-TIME-S(i,j)

ขั้นตอนที่ 4.3

ให้ i = i + 1

ขั้นตอนที่ 4.4

ถ้า i <= N ไปทำขั้นตอนที่ 4.1

ถ้า i > N ไปทำขั้นตอนที่ 5

ขั้นตอนที่ 5 เริ่มพิจารณาการจัดลำดับของเครื่องจักรถัดไป

ให้ m = m + 1

ขั้นตอนที่ 6

ถ้า m <= M ไป ทำขั้นตอนที่ 3

ถ้า m > M สิ้นสุดการทำงาน

3. <u>กฎวันกำหนดส่งงานเร็วที่สุด</u>

ขั้นตอนที่ 1 เรียงลำดับงานตามกำหนดวันส่งงานใดๆ จากน้อยไปหามาก

ขั้นตอนที่ 2 กำหนดเลขที่งานที่เรียงลำดับแล้วให้กับ S(i,j)

ให้ j = 1

ขั้นตอนที่ 2.1

ให้ S(1,j) = เลขที่งานที่มี กำหนดวันส่งงานน้อยที่สุด

. .

. .

. .

ให้ S(n,j) = เลขที่งานที่มี กำหนดวันส่งงานมากที่สุด

ขั้นตอนที่ 2.2

ให้ j = j + 1

ขั้นตอนที่ 2.3

ถ้า j <= M ไปทำขั้นตอนที่ 2.1

ถ้า j > M ไปทำขั้นตอนที่ 3

ขั้นตอนที่ 3 หาเวลาที่สิ้นสุดของงานใดๆที่เครื่องจักรหนึ่งๆ

ให้ j = 1

ขั้นตอนที่ 3.1

ให้ i = 1

ขั้นตอนที่ 3.2 หา ARRIVAL-TIME-S(i,j)

ถ้า i = 1 และ j = 1

ให้ ARRIVAL-TIME-S(1,1) = ARRIVAL-TIME-JOB(S(1,1),1)

ไปทำขั้นตอนที่ 3.3

ถ้า i = 1

ให้ ARRIVAL-TIME-S(1,j) = FINISH-TIME-JOB(S(1,j),j-1)

ไปทำขั้นตอนที่ 3.3

ถ้า j = 1

ให้ ARRIVAL-TIME-S(i,1) =

MAX[FINISH-TIME-S(i-1,1),

ARRIVAL-TIME-JOB(S(i,1),1)]

ไปทำขั้นตอนที่ 3.3

ถ้า i <> 1 และ j <> 1

ให้ ARRIVAL-TIME-S(i,j) =

MAX[FINISH-TIME-JOB(S(i,j),m-1),

FINISH-TIME-S(i-1,j)]

ขั้นตอนที่ 3.3 หา FINISH-TIME-S(i,j) และ

FINISH-TIME-JOB(S(i,j),j)

ให้ FINISH-TIME-S(i,j) = ARRIVAL-TIME-S(i,j) +

PROCESS-TIME-JOB(S(i,j),m)

ให้ FINISH-TIME-JOB(S(i,j),j) = FINISH-TIME-S(i,j)

ขั้นตอนที่ 3.4

ให้ i = i + 1

ขั้นตอนที่ 3.5

ถ้า i <= N ไปทำขั้นตอนที่ 3.2

ถ้า i > N ไปทำขั้นตอนที่ 3.6

ขั้นตอนที่ 3.6

ให้ j = j + 1

ขั้นตอนที่ 3.7

ถ้า j <= M ไปทำขั้นตอนที่ 3.1

ถ้า j > M สิ้นสุดการทำงาน

4. <u>กฎแบบสุ่ม</u>

ขั้นตอนที่ 1 เริ่มพิจารณาการจัดลำดับของเครื่องจักรที่ 1

ให้ m = 1

ขั้นตอนที่ 2 หาเวลาที่จะใช้ในการปฏิบัติงานใดๆ ที่เครื่องจักรหนึ่ง

ให้ n = 1

ขั้นตอนที่ 2.1

ให้ค่าสุ่มกับงานแต่ละงานโดยใช้ฟังก์ชั่น Random ซึ่งกำหนดค่า Seed ที่ได้จากการคำนวณ

ขั้นตอนที่ 2.2

ให้ n = n + 1

ขั้นตอนที่ 2.3

ถ้า n <= N ไปทำขั้นตอนที่ 2.1

ถ้า n > N ไปทำขั้นตอนที่ 3

ขั้นตอนที่ 3 เรียงลำดับงานตามค่าสุ่มของงานใดๆที่เครื่องจักรหนึ่งๆ
จากน้อยไปหามาก

ขั้นตอนที่ 4 กำหนดเลขที่งานที่เรียงลำดับแล้วให้กับ S(i,j)

ให้ j = m

ให้ S(1,j) = เลขที่งานที่มี ค่าสุ่มน้อยที่สุด

. .

. .

ให้ S(n,j) = เลขที่งานที่มี ค่าสุ่มมากที่สุด

ขั้นตอนที่ 5 หาเวลาที่สิ้นสุดของงานใดๆที่เครื่องจักรหนึ่งๆ

ให้ i = 1

ขั้นตอนที่ 5.1 หา ARRIVAL-TIME-S(i,j)

ถ้า i = 1 และ j = 1

ให้ ARRIVAL-TIME-S(1,1) = ARRIVAL-TIME-JOB(S(1,1),1)

ไปทำขั้นตอนที่ 5.2

ถ้า i = 1

ให้ ARRIVAL-TIME-S(1,j) = FINISH-TIME-JOB(S(1,j),j-1)

ไปทำขั้นตอนที่ 5.2

ถ้า j = 1

ให้ ARRIVAL-TIME-S(i,1) =

MAX[FINISH-TIME-S(i-1,1),

ARRIVAL-TIME-JOB(S(i,1),1)]

ไปทำขั้นตอนที่ 5.2

ถ้า i <> 1 และ j <> 1

ให้ ARRIVAL-TIME-S(i,j) =

MAX[FINISH-TIME-JOB(S(i,j),m-1),

FINISH-TIME-S(i-1,j)]

ขั้นตอนที่ 5.2 หา FINISH-TIME-S(i,j) และ FINISH-TIME-JOB(n,m)

ให้ FINISH-TIME-S(i,j) = ARRIVAL-TIME-S(i,j) +
PROCESS-TIME-JOB(S(i,j),m)

ให้ FINISH-TIME-JOB(S(i,j),m) = FINISH-TIME-S(i,j)

ขั้นตอนที่ 5.3

ให้ i = i + 1

ขั้นตอนที่ 5.4

ถ้า i <= N ไปทำขั้นตอนที่ 5.1

ถ้า i > N ไปทำขั้นตอนที่ 6

ขั้นตอนที่ 6 เริ่มพิจารณาการจัดลำดับของเครื่องจักรถัดไป

ให้ m = m + 1

ขั้นตอนที่ 7

ถ้า m <= M ไป ทำขั้นตอนที่ 2

ถ้า m > M สิ้นสุดการทำงาน

5. <u>กฎเข้าหลังบริการก่อน</u>

ขั้นตอนที่ 1 เริ่มพิจารณาการจัดลำดับของเครื่องจักรที่ 1

ให้ m = 1

ขั้นตอนที่ 2 หาเวลาที่สามารถเริ่มงาน n ที่เครื่องจักรที่ m

ให้ n = 1

ขั้นตอนที่ 2.1

ถ้า m = 1 ไปทำขั้นตอนที่ 2.2

ถ้า m <> 1

ให้ ARRIVAL-TIME-JOB(n,m) = FINISH-TIME-JOB(n,m-1)

ขั้นตอนที่ 2.2

ให้ n = n + 1

ขั้นตอนที่ 2.3

ถ้า n <= N ไปทำขั้นตอนที่ 2.1

ถ้า n > N ไปทำขั้นตอนที่ 3

ขั้นตอนที่ 3 เรียงลำดับงานตามเวลาที่งานสามารถเริ่มงานได้ ก่อน-หลัง ที่เครื่องจักรหนึ่งๆจากมากไปหาน้อย

ขั้นตอนที่ 4 กำหนดเลขที่งานที่เรียงลำดับแล้วให้กับ S(i,j)

ให้ j = m

ให้ S(1,j) = เลขที่งานที่มี ARRIVAL-TIME-JOB(n,m) มากที่สุด

. .

. .

. .

ให้ S(n,j) = เลขที่งานที่มี ARRIVAL-TIME-JOB(n,m) น้อยที่สุด

ขั้นตอนที่ 5 หาเวลาที่สิ้นสุดของงานใดๆที่เครื่องจักรหนึ่งๆ

ให้ i = 1

ขั้นตอนที่ 5.1 หา ARRIVAL-TIME-S(i,j)

ถ้า i = 1 และ j = 1

ให้ ARRIVAL-TIME-S(1,1) = ARRIVAL-TIME-JOB(S(1,1),1)

ไปทำขั้นตอนที่ 5.2

ถ้า i = 1

ให้ ARRIVAL-TIME-S(1,j) = FINISH-TIME-JOB(S(1,j),j-1)

ไปทำขั้นตอนที่ 5.2

ถ้า j = 1

ให้ ARRIVAL-TIME-S(i,1) =

MAX[FINISH-TIME-S(i-1,1),

ARRIVAL-TIME-JOB(S(i,1),1)]

ไปทำขั้นตอนที่ 5.2

ถ้า i <> 1 และ j <> 1 และ

FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า <

FINISH-TIME-S ของเครื่องจักรปัจจุบัน

ให้นำงานที่เหลืออยู่เฉพาะงานที่ FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า มีค่าน้อยกว่า FINISH-TIME-S ของเครื่องจักรปัจจุบันลำดับงานก่อนหน้ามาเรียงลำดับงานใหม่ตามเวลาที่งานสามารถเริ่มงานได้จากมากไปหาน้อย และ

ให้ ARRIVAL-TIME-S(i,j) =

FINISH-TIME-S ของเครื่องจักรปัจจุบันลำดับงานก่อนหน้า

ขั้นตอนที่ 5.2 หา FINISH-TIME-S(i,j) และ FINISH-TIME-JOB(n,m)

ให้ FINISH-TIME-S(i,j) = ARRIVAL-TIME-S(i,j) +

PROCESS-TIME-JOB(S(i,j),m)

ให้ FINISH-TIME-JOB(S(i,j),m) = FINISH-TIME-S(i,j)

ขั้นตอนที่ 5.3

ให้ i = i + 1

ขั้นตอนที่ 5.4

ถ้า i <= N ไปทำขั้นตอนที่ 5.1

ถ้า i > N ไปทำขั้นตอนที่ 6

ขั้นตอนที่ 6 เริ่มพิจารณาการจัดลำดับของเครื่องจักรถัดไป

ให้ m = m + 1

ขั้นตอนที่ 7

ถ้า m <= M ไปทำขั้นตอนที่ 4

ถ้า m > M สิ้นสุดการทำงาน

6. <u>กฎอัตราส่วนวิกฤติ</u>

ขั้นตอนที่ 1 เริ่มพิจารณาการจัดลำดับของเครื่องจักรที่ 1

ให้ m = 1

ขั้นตอนที่ 2 เรียงลำดับงานตามสัดส่วนระหว่างเวลาที่เหลือก่อนถึงกำหนดส่งงานต่อเวลาที่ต้องใช้ในการผลิตที่เหลือของงานนั้นๆจากน้อยไปหามาก

ขั้นตอนที่ 3 กำหนดเลขที่งานที่เรียงลำดับแล้วให้กับ S(i,j)

ให้ j = m

ให้ S(1,j) = เลขที่งานที่มีอัตราส่วนวิกฤติน้อยที่สุด

. .

. .

. .

ให้ S(n,j) = เลขที่งานที่มีอัตราส่วนวิกฤติมากที่สุด

ขั้นตอนที่ 4 หาเวลาที่สิ้นสุดของงานใดๆที่เครื่องจักรหนึ่งๆ

ให้ i = 1

ขั้นตอนที่ 4.1 หา ARRIVAL-TIME-S(i,j)

ถ้า i = 1 และ j = 1

ให้ ARRIVAL-TIME-S(1,1) = ARRIVAL-TIME-JOB(S(1,1),1)

ไปทำขั้นตอนที่ 4.2

ถ้า i = 1

ให้ ARRIVAL-TIME-S(1,j) = FINISH-TIME-JOB(S(1,j),j-1)

ไปทำขั้นตอนที่ 4.2

ถ้า j = 1

ให้ ARRIVAL-TIME-S(i,1) =

MAX[FINISH-TIME-S(i-1,1),

ARRIVAL-TIME-JOB(S(i,1),1)]

ไปทำขั้นตอนที่ 4.2

ถ้า i <> 1 และ j <> 1 และ

FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า >=

FINISH-TIME-S ของเครื่องจักรปัจจุบัน

ให้ ARRIVAL-TIME-S(i,j) =

FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า

ถ้า i <> 1 และ j <> 1 และ

FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า <

FINISH-TIME-S ของเครื่องจักรปัจจุบัน

ให้นำงานที่เหลืออยู่เฉพาะงานที่ FINISH-TIME-JOB ของเครื่องจักรก่อนหน้า มีค่าน้อยกว่า FINISH-TIME-S ของเครื่องจักรปัจจุบันลำดับงานก่อนหน้ามาเรียงลำดับงานใหม่ตามอัตราส่วนวิกฤติของงานจากน้อยไปหามาก และ

ให้ ARRIVAL-TIME-S(i,j) =

FINISH-TIME-S ของเครื่องจักรปัจจุบันลำดับงานก่อนหน้า

ขั้นตอนที่ 4.2 หา FINISH-TIME-S(i,j) และ FINISH-TIME-JOB(n,m)

ให้ FINISH-TIME-S(i,j) = ARRIVAL-TIME-S(i,j) +

PROCESS-TIME-JOB(S(i,j),m)

ให้ FINISH-TIME-JOB(S(i,j),m) = FINISH-TIME-S(i,j)

ขั้นตอนที่ 4.3

ให้ i = i + 1

ขั้นตอนที่ 4.4

ถ้า i <= N ไปทำขั้นตอนที่ 4.1

ถ้า i > N ไปทำขั้นตอนที่ 5

ขั้นตอนที่ 5 เริ่มพิจารณาการจัดลำดับของเครื่องจักรถัดไป

ให้ m = m + 1

ขั้นตอนที่ 6

ถ้า m <= M ไป ทำขั้นตอนที่ 3

ถ้า m > M สิ้นสุดการทำงาน

ภาคผนวก ค

องค์ประกอบของระบบ

ในการพัฒนาโปรแกรมคอมพิวเตอร์ จะมีส่วนประกอบของระบบงาน ดังนี้

1. <u>ส่วนของฮาร์ดแวร์</u>

1.1 เครื่องคอมพิวเตอร์ไอบีเอ็มพีซี/เอที หรือเครื่องที่ใช้แทนได้ โดยมีหน่วยความจำอย่างน้อย 640 กิโลไบต์

1.2 แผงวงจรแสดงภาพอีจีเอ/วีจีเอ หรือชนิดที่ใช้แทนได้ โดยมีหน่วยความจำอย่างน้อย 256 กิโลไบต์

1.3 จานบันทึก ซึ่งมีเนื้อที่ว่างอย่างน้อย 3 เมกะไบต์

1.4 ไมโครซอฟท์เมาส์

1.5 เครื่องพิมพ์ (จะมีหรือไม่มีก็ได้)

1.6 Math-coprocessor (จะมีหรือไม่มีก็ได้)

1.7 จอภาพอีจีเอ/วีจีเอ

2. <u>ส่วนของซอฟท์แวร์</u>

2.1 เอ็มเอส-ดอส 3.0 หรือสูงกว่า

2.2 CINEMA Animation Design Package

2.3 SIMAN Simulation Language (Animation Version)

2.4 SIMAN Simulation Language (Standard Version)

2.5 ไมโครซอฟท์ฟอร์แทรน 4.0 (จะมีหรือไม่มีก็ได้)

2.6 ไมโครซอฟท์ซี 4.0 (จะมีหรือไม่มีก็ได้)

ประวัติผู้เขียน

นายยอดชาย ฐิตวรรณโณเนตร์ เกิดวันที่ 17 มีนาคม 2504 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาวิทยาศาสตร์บัณฑิต (สาขาคณิตศาสตร์) จากคณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าวิทยาเขตธนบุรี ในปีการศึกษา 2525 และเข้าศึกษาต่อในระดับปริญญามหาบัณฑิต ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2532 ปัจจุบันทำงานที่บริษัทชินวัตรคอมพิวเตอร์ แอนด์ คอมมูนิเคชั่นส์ (มหาชน) จำกัด