การจัดตารางเวลาของกลุ่มงานให้กับเครื่องจักรแบบขนานเพื่อให้ได้เวลาเสร็จสิ้นที่ต่ำสุด

Scheduling of n Job-Groups through m Parallel Machines with Minimum Makespan

ศิขรินทร์ สุขโต

ภาควิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น อ.เมือง จ.ขอนแก่น 400002

โทร. 043-343117, 01-5550570 โทรสาร 043-343117

บทคัดย่อ

วิธีการโดยประมาณ ในการคำนวณหาเวลาที่น้อยที่สุดในการผลิตให้เสร็จสิ้นทั้งหมด สำหรับปัญหาการจัดตารางเวลาที่มีจำนวนกลุ่มของงาน n กลุ่ม โดยในแต่ละกลุ่มมีปริมาณความต้องการในการผลิตเป็น q_i และเวลาในการผลิตต่อหน่วยรวมเวลาการเตรียมงานเป็น p_i ให้กับเครื่องจักร m เครื่องที่ขนานกันและเป็นอิสระต่อกัน เรียกว่าวิธี IPCCG ที่ใช้วิธีโปรแกรมเชิงจำนวนเต็ม (Integer programming, IP) ในการสร้างรูปแบบเบื้องต้นของตารางเวลา จากผลลัพธ์ที่เป็นไปได้ภายใต้การกำหนดขอบเขตในการประมวลผลจากโปรแกรมสำเร็จรูป LINGO 6 ที่ 2×10^5 ขั้นตอน แล้วทำการปรับปรุงรูปแบบของตารางเวลาดังกล่าวให้อยู่ในขอบเขตล่างโดยวิธี AIPP (Adjusted IP Pattern) หลังจากนั้นทำการสร้างรูปแบบของตารางเวลาโดยวิธี CCG (Continuous Column Generation) ซึ่งประยุกต์มาจากวิธีการสร้างสดมภ์ วิธีการนี้จะประมวลผลด้วยวิธี IP AIPP และ CCG ตามลำดับซ้ำ ๆ กันจนกระทั่งสามารถสร้างรูปแบบของตารางเวลาอยู่ภายในขอบเขตที่กำหนดได้ พบว่าสมรรถภาพโดยเฉลี่ยของวิธีการ IPCCG นั้นดีกว่าวิธี LPTCCG [16] อย่างมีนัยสำคัญ

Abstract

An optimization based heuristic for minimizing makespan on scheduling n job-groups with q_i as the number of identical jobs and p_i as the processing time, included any setup time required, in group i through a set of m identical parallel machines, referred as IPCCG, is based on IP (Integer Programming) heuristic by LINGO 6.0 software and CG (Column Generation) which is well known in cutting stock problem. The first method, IP Heuristic, is used to construct the initial pattern, which is interrupted at 2×10^5 iterations and adjusted to a lower bound by AIPP (Adjusted IP Pattern); while the second method, CG, is modified to CCG (Continuous Column Generation) attempting to strengthen makespan bound and collectively generated patterns to minimize a number of machines required. IPCCG heuristic is repeatedly looped by IP Heuristic, AIPP and CCG to reach an upper bound of minimum makespan. The average performance of the proposed IPCCG heuristic is significantly better than that of LPTCCG heuristic [16].

1. คำนำ

การศึกษาปัญหาด้านการจัดตารางเวลา (Scheduling problem) นั้นมีมายาวนานเริ่มตั้งแต่ราวปี ค.ศ. 1968 เป็นต้นมา และเริ่มมีบทความทางวิชาการในด้านนี้อย่างมากมาย จนกระทั่งได้รับการบรรจุเป็นวิชาเรียนในการศึกษาทางด้านวิศวกรรมอุตสาหการ ซึ่งเป็นวิชาที่ว่าด้วยการจัดสรรทรัพยากรที่มีอยู่ให้มีคุณค่ามากที่สุด โดยกำหนดการทำงานหรือกิจกรรมให้เป็น "งาน" ที่มีเพียงกิจกรรมเดียวหรือหลายกิจกรรมในงานนั้น ๆ ก็ได้ ส่วน "เครื่องจักร" นั้นเป็นทรัพยากรที่นำมาจัดหรือบรรจุงานดังกล่าวลงไป ปัญหานี้นั้นมีบทบาทอย่างมากในอุตสาหกรรมการผลิตโดยทั่วไป การขนส่งสินค้า การกระจายสินค้า การติดต่อสื่อสาร การจัดการทางด้านสื่อ และการกีฬา โดยเฉพาะอย่างยิ่งการจัดตารางเวลาที่ดีหรือเหมาะสมในแต่ละการผลิตนั้น ย่อมส่งผลโดยตรงในเชิงเศรษฐศาสตร์เป็นอย่างมาก

ปัญหาการจัดตารางเวลาเครื่องจักรแบบขนานนั้น เป็นการจัดงานที่มีจำนวนและเวลาในการผลิตที่แตกต่างกัน ให้กับแต่ละเครื่องจักรที่สามารถทำงานที่กำหนดให้ได้ ภายใต้สมรรถภาพที่กำหนด ปัญหาการจัดตารางเวลาสำหรับเครื่องจักรตัวเดียวนั้นเป็นปัญหาย่อยที่มีบทบาทสำคัญในปัญหาการจัดตารางเวลาเครื่องจักรแบบขนาน ที่มีความซับซ้อนมากยิ่งขึ้น เครื่องจักรแบบขนานนั้นอาจจะทำงานได้อย่างอิสระต่อกัน โดยที่ทำงานแต่ละงานสำเร็จได้ภายในระยะเวลาที่เท่ากัน แต่ถ้าเครื่องจักรนั้นมีเทคโนโลยีหรืออายุการใช้งานที่แตกต่างกัน ก็จะทำให้เวลาในการทำงานมีระยะเวลาที่แตกต่างกันได้ [10] การจัดประเภทของปัญหาการจัดลำดับและตารางเวลาใช้สัญลักษณ์ 3 ลำดับอักษร $\alpha/\beta/\gamma$ ที่บ่งบอกถึง ระบบของการผลิต ลักษณะเฉพาะของงาน และเกณฑ์การวัดสมรรถภาพของตารางเวลา ตามลำดับ

Dyckhoff [3] ได้ทำการจัดกลุ่มประเภทของปัญหาการตัดและการบรรจุ (Cutting and packing problem) หรือ ปัญหา C & P โดยพบว่าโครงสร้างทางตรรกะของลักษณะปัญหาแบบนี้ มีความสัมพันธ์คู่ควบกันที่สามารถกำหนดได้เป็นรูปแบบมาตรฐานของปัญหาดังกล่าว โดยการใช้สัญลักษณ์ 4 ลำดับอักษร $\alpha/\beta/\gamma/\delta$ ที่บ่งบอกถึง จำนวนขนาดของมิติการกำหนดชนิด ลักษณะของวัสดุขนาดใหญ่ และลักษณะของวัสดุขนาดย่อย ตามลำดับ สำหรับกรณีการจัดตารางเวลาเครื่องจักรแบบขนานนั้นเป็นมิติของเวลาในการทำงาน ซึ่งสามารถกำหนดให้เป็นรูปแบบ

มาตรฐานของปัญหานี้ได้เป็น 1/V/1/M และมีความสัมพันธ์กับปัญหา C&P ดังแสดงไว้ในตารางที่ 1

ตารางที่ 1 ความสัมพันธ์ของปัญหาการจัดตารางเวลาของเครื่องจักรแบบขนาน ปัญหาการบรรจุกล่อง และปัญหาการตัดวัสดุ

รูปแบบของปัญหา	การจัดตารางเวลาเครื่องจักรแบบขนาน	การบรรจุกล่อง	การตัดวัสดุ
ประเภท	$P//C_{max}$	1/V/1/M	1/V/1/R
วัตถุขนาดใหญ่	เวลาของเครื่องจักร	กล่องบรรจุ	วัสดุมาตรฐาน
วัตถุขนาดเล็ก	เวลาของแต่ละงาน	กล่องย่อย	วัสดุย่อย
สมรรถภาพ	C_{max} (Min. makespan)	ปริมาณความจุ	ความยาววัสดุ

ปัญหาการตัดวัสดุ (Cutting stock problem) เป็นหนึ่งในปัญหาการวิจัยดำเนินงาน ที่ริเริ่มโดย Kantorovich [14] นักเศรษฐศาสตร์ชาวรัสเซียราวปี ค.ศ. 1939 ซึ่งเป็นช่วงเดียวกันที่การวิจัยดำเนินงานเริ่มมีบทบาทในวิศวกรรมอุตสาหการ ต่อมาในช่วงปี ค.ศ. 1961–1963 Gilmore และ Gomory [6-7] ได้ริเริ่มการหาวิธีการแก้ปัญหานี้ได้เป็นอย่างดี โดยใช้วิธีการสร้างรูปแบบการตัดในการแก้ปัญหาการตัดวัสดุมิติให้สูญเสียวัสดุต่ำที่สุด ที่สามารถใช้ทฤษฎีโปรแกรมเชิงเส้นตรงในการประมวลผล จึงทำให้ปัญหา และวิธีการนี้เป็นที่สนใจกันอย่างกว้างขวางในเวลาต่อมา ปัญหาลักษณะนี้นั้นมีอยู่มากมายในหลากหลายอุตสาหกรรม ดังนั้นผลลัพธ์ที่ดีย่อมมีผลโดยตรงต่อต้นทุนในการผลิต วิธีการนี้นั้นมีศักยภาพและง่ายในการประยุกต์ใช้กับปัญหาที่มีลักษณะที่คล้ายคลึงกัน ส่วนปัญหาที่คล้ายกันมากจนถือได้ว่าเป็นปัญหาคู่ควบกัน คือปัญหาการบรรจุ (Packing problem) ดังแสดงในตารางที่ 1 ซึ่งเริ่มมีการศึกษาราวปี ค.ศ. 1970 [8] สำหรับการกำหนดเกณฑ์ในการวัดสมรรถภาพของตารางเวลานั้นมีอยู่หลากหลายเกณฑ์ สำหรับบทความนี้ต้องการหาเวลาเสร็จสิ้นของทุกงานที่ต่ำที่สุด หรือเวลาเสร็จสิ้นที่ต่ำที่สุดของเวลาเสร็จสิ้นที่มากที่สุด (Makespan) ซึ่งกำหนดเป็น C_{max}

2. ลักษณะของปัญหาและวิธีการพื้นฐาน

ลักษณะของปัญหานี้พิจารณาจากปัญหาการจัดตารางเวลา ที่มีจำนวนกลุ่มของงาน n กลุ่ม โดยที่ $i = 1, 2, 3, \ldots, n$ ซึ่งแต่ละกลุ่มนั้นมีเวลาในการผลิตต่อหน่วยเป็น p_i และมีปริมาณความต้องการในการผลิตเป็น q_i นำมาจัดตารางเวลาในการผลิตบนเครื่องจักร m เครื่องที่ขนานกันและเป็นอิสระต่อกัน โดยที่แต่ละงานนั้นสามารถผลิตได้ในทุก ๆ เครื่องจักร มีความต้องการในการผลิตเพียงครั้งเดียวในแต่ละเครื่องจักร ซึ่งในแต่ละหน่วยของงานจะต้องทำการผลิตจนเสร็จสิ้นโดยไม่มีการหยุดพักในแต่ละเครื่องจักรที่ j โดยที่ $j = 1, 2, 3, \ldots, m$ กำหนดให้ S_j เป็นตารางเวลาของงานที่จัดให้กับเครื่องจักรที่ j นั่นคือ $S_j = \{s_{ij}\}$ ดังนั้นเวลาเสร็จสิ้นในการผลิตของแต่ละเครื่องจักรกำหนดได้เป็น $C(S_j) = \sum_{i=1}^{n} p_i s_{ij}$ ซึ่งจะทำให้เวลาในการผลิตให้เสร็จสิ้นที่มากที่สุด กำหนดให้เป็น C_{max} ของตารางเวลาเป็นดังต่อไปนี้

$$C_{max} = C_{max}(S_j) = \max_{1 \le j \le m} C(S_j) = \max_{1 \le j \le m} \{ \textstyle\sum_{i=1}^{n} p_i s_{ij} \} \qquad (1)$$

เมื่อต้องการคำนวณหาเวลาที่น้อยที่สุดในการผลิตให้เสร็จสิ้นที่มากที่สุด หรือ min C_{max} สำหรับในกรณีที่แต่ละกลุ่มของงานไม่มีเวลาเตรียมการผลิตและกำหนดเวลาส่งมอบนั้น เวลาที่น้อยที่สุดในการผลิตให้เสร็จสิ้นที่มากที่สุดเท่าที่เป็นไปได้นั้น กำหนดเป็น C^*_{max} ซึ่งเป็นขอบเขตล่างที่ต่ำที่สุด ดังนั้น

$$C^*_{max} = \left\lceil \textstyle\sum_{i=1}^{n} p_i q_i / m \right\rceil;\ \text{เลขจำนวนเต็ม} \qquad (2)$$

ลักษณะปัญหาเช่นนี้ เมื่อเปรียบเทียบกับการจัดประเภทของปัญหาการตัดและการบรรจุ [3] กำหนดปัญหาการตัดมิติเดียวเป็นประเภท 1/V/I/R ปัญหาการบรรจุกล่องมิติเดียวเป็นประเภท 1/V/I/M ปัญหาการบรรทุกเป็นประเภท 1/V/I/F หรือ 1/V/I/M ปัญหาการสมดุลสายงานการประกอบเป็นประเภท 1/V/I/M และปัญหาการจัดลำดับและตารางเวลาเป็นประเภท 1/V/I/M ซึ่งปัญหาทั้งหมดเป็นแบบมิติเดียว (1) ชนิดเลือกทั้งวัสดุขนาดใหญ่และวัสดุขนาดย่อย (V) วัสดุขนาดใหญ่หลายชิ้นที่มีขนาดเท่ากัน (I) ทั้งสิ้น แตกต่างกันที่อักษรลำดับที่ 4 ซึ่งเป็นลักษณะของวัสดุขนาดย่อยเท่านั้น ถ้าเป็น (F) แล้วนั้นจะมีจำนวนของวัสดุขนาดย่อยน้อย เช่น 10 ชิ้นในปัญหาการบรรทุก ถ้าเป็น (M) จะมีจำนวนของวัสดุขนาดย่อยมาก เช่นหลายร้อยชิ้นและมีขนาดที่แตกต่างกันมาก เช่นในปัญหาการบรรจุกล่อง ปัญหาการบรรทุก ปัญหาการสมดุลสายงานการประกอบ และปัญหาการจัดลำดับและตารางเวลา แต่ถ้าเป็น (R) จะต้องมีจำนวนของวัสดุขนาดย่อยมาก ๆ เช่นหลายพันชิ้นแต่มีขนาดที่ไม่แตกต่างกันมาก เช่น 50 ขนาด เป็นต้น แต่อย่างไรก็ตามลักษณะของวัสดุขนาดย่อยที่แตกต่างกันนั้น เพียงเพื่อต้องการจัดประเภทของปัญหาเท่านั้น แต่ถ้าในแต่ละปัญหามีวัตถุประสงค์เช่นเดียวกัน ก็สามารถใช้วิธีการเดียวกันในการแก้ปัญหานั้น ๆ ได้ Garey และ Johnson [5] ได้แสดงให้เห็นว่าปัญหาทั้งหมดดังกล่าวมาแล้วนั้น ล้วนแล้วแต่เป็นปัญหาการหาคำตอบที่ดีที่สุดในระดับ NP-complete หรือ NP-hard แทบทั้งสิ้น ซึ่งอาจจะมีคำตอบที่เป็นไปได้มากมาย จนไม่สามารถประมวลผลให้อยู่ภายใต้ขอบเขตของเวลาแบบโพลีโนเมียลได้ อีกทั้งคำตอบที่ดีที่สุดอาจจะมีจำนวนมากมายสำหรับในทางปฏิบัติแล้วนั้น จำเป็นต้องหาผลลัพธ์ที่เหมาะสมที่สุดภายใต้เวลาที่เหมาะสม ดังนั้นวิธีการโดยประมาณ (Heuristic) จึงถูกนำมาใช้ในการแก้ปัญหาลักษณะนี้ สำหรับปัญหา $P//C_{max}$ นี้ วิธี LPT (Longest processing time) ที่ริเริ่มโดย Graham [9] นั้นเป็นที่รู้จักกันดีในการจัดลำดับตารางเวลาที่มีเวลาในการประมวลผลเป็น O(n log n) และมีสมรรถภาพต่ำสุดที่ 4/3 – 1/3m ต่อมา Coffman และคณะ [1] ได้ทำการ

ประยุกต์แนวคิดของการบรรจุกล่อง เพื่อจัดตารางเวลาสำหรับงานที่มีหลายหน่วยผลิต ที่แสดงให้เห็นว่าปัญหาการบรรจุกล่องมีความสัมพันธ์คู่ควบกับปัญหาจัดตารางเวลางานที่มีหลายหน่วยผลิต เรียกวิธีการนี้ว่า MULTIFIT ซึ่งใช้หลักการของ FFD (First-fit decreasing weight) ในการจัดลำดับตารางเวลา ถึงแม้ว่า MULTIFIT จะมิได้มีสมรรถภาพที่ดีกว่า LPT โดยสิ้นเชิงก็ตาม แต่ก็มีสมรรถภาพต่ำสุดที่ดีกว่าวิธี LPT ที่ $13/11 + 2^{-b}$ โดยที่ b คือจำนวนรอบในการประมวลผล และใช้เวลาในการประมวลผลเป็น $O(n \log n + bn \log m)$ ต่อมา Lee และ Massey [15] ได้ทำการประยุกต์วิธีการดังกล่าวโดยรวมเอาทั้ง 2 วิธี คือ LPT และ MULTIFIT เข้าด้วยกันเรียกว่าวิธี COMBINE แต่เวลาในการประมวลผลก็ยังเป็น $O(n \log n + bn \log m)$ และมีสมรรถภาพต่ำสุดเป็น $13/11 + 2^{-b}$ โดยที่ b คือ จำนวนรอบในการประมวลผล ซึ่งเท่ากันกับ MULTIFIT ในช่วงหลังสหัสวรรษใหม่นี้ Gupta และ Ruiz-Torres [10] ได้เสนอวิธีการที่มีชื่อว่า LISTFIT ซึ่งได้ผสมผสานระหว่างวิธี MULTIFIT กับวิธีการจัดลำดับตารางเวลา (SPT หรือ LPT) วิธีการนี้จะจัดลำดับงาน n ครั้งซ้ำกัน 4 รอบ ดังนั้นจึงมีการจัดลำดับงานทั้งสิ้น 4n ครั้ง สำหรับเวลาในการประมวลผลนั้นเป็น $O(n^2 \log n + bn^2 \log m)$ ซึ่งมากกว่าวิธีที่ผ่านมา และมีสมรรถภาพต่ำสุดเป็น $13/11 + 2^{-b}$ เท่ากัน แต่จากการสังเกตผลลัพธ์ที่ได้นั้นมีสมรรถภาพโดยเฉลี่ยสูงกว่าทั้ง 3 วิธีที่ผ่านมา ต่อมาข้าพเจ้า Sukto และ Charnsethikul [16] ได้เสนอวิธี LPTCCG ซึ่งเป็นการผสมผสานระหว่างวิธี CCG (Continuous column generation) ที่ประยุกต์จากวิธีการสร้างสดมภ์ (Column generation) ของวิธีการที่มีประสิทธิภาพในปัญหาการตัดวัสดุ ร่วมกับวิธีการจัดลำดับตารางเวลา (LPT) โดยใช้วิธี LPT ในการสร้างรูปแบบเบื้องต้นของตารางเวลา แล้วจึงปรับปรุงรูปแบบของตารางเวลาดังกล่าวให้อยู่ในขอบเขตล่าง (2) หลังจากนั้นทำการสร้างรูปแบบของตารางเวลาด้วยวิธี CCG ให้อยู่ในขอบเขตล่าง ถ้าไม่อยู่ในขอบเขตดังกล่าวก็ทำการขยายขอบเขตล่างออก แล้วจึงดำเนินวิธีการนี้เช่นเดิมจนกระทั่งสามารถสร้างรูปแบบของตารางเวลา อยู่ภายในขอบเขตที่กำหนดได้ การประมวลผลจาก 25 กลุ่มตัวอย่างรวม 250 ตัวอย่าง พบว่าสามารถลด C_{max} ลงได้จากวิธี LPT ได้ในกรณีที่อัตราส่วนของ n/m มีค่าน้อยกว่า 2.5 อย่างมีนัยสำคัญ

2.1 โปรแกรมเชิงจำนวนเต็ม (Integer Programming)

จากวัตถุประสงค์หลักเพื่อต้องการหาเวลาที่น้อยที่สุด ในการผลิตให้เสร็จสิ้นมากที่สุด ซึ่งสามารถสร้างแบบจำลองทางคณิตศาสตร์ในรูปของโปรแกรมเชิงจำนวนเต็มได้ดังต่อไปนี้ โดยกำหนดให้

C_{max} = เวลาในการผลิตให้เสร็จสิ้นที่มากที่สุด

i = กลุ่มของงานจำนวน n กลุ่ม

j = เครื่องจักรที่ขนานกัน และเป็นอิสระต่อกัน m เครื่อง

n_i = กลุ่มงานที่ i

m_j = เครื่องจักรที่ j

p_i = เวลาในการผลิตต่อหน่วยสำหรับงานกลุ่มที่ i

q_i = ปริมาณความต้องการในการผลิตสำหรับงานกลุ่มที่ i

s_{ij} = จำนวนงานของกลุ่มงานที่ i บนเครื่องจักรที่ j

$$Z = \min C_{max} \tag{3}$$

ข้อจำกัด

$$\sum_{j=1}^{m} s_{ij} \geq q_i \quad ; \forall\ i = 1, 2, 3, \ldots, n \tag{4}$$

$$\sum_{i=1}^{n} p_i s_{ij} - C_{max} \leq 0 \quad ; \forall\ j = 1, 2, 3, \ldots, m \tag{5}$$

$$C_{max}, s_{ij} \geq 0 \text{ และ เลขจำนวนเต็ม} \tag{6}$$

จากแบบจำลองทางคณิตศาสตร์ข้างต้น ในสมการที่ (3) เป็นการกำหนดสมการเป้าหมาย เพื่อคำนวณหาเวลาที่น้อยที่สุดในการผลิตให้เสร็จสิ้นที่มากที่สุด ที่มีข้อจำกัดด้านปริมาณความต้องการของงานในแต่กลุ่ม (q_i) ที่ครบถ้วนหรือมากกว่า โดยจำนวนงานในตารางเวลาของกลุ่มงานที่ i บนเครื่องจักรที่ j (s_{ij}) เมื่อรวมกันในทุกเครื่องจักรที่ j แล้วจะต้องมีปริมาณไม่น้อยกว่าปริมาณความต้องการของงานในแต่กลุ่ม (q_i) ดังสมการที่ (4) ส่วนข้อจำกัดด้านเวลาที่เสร็จสิ้นในการผลิตของแต่ละเครื่องจักรที่ j นั้น เวลาการผลิตรวมของงานกลุ่มที่ i นั้น ($p_i s_{ij}$) จะต้องรวมแล้ว มีเวลาในการผลิตน้อยกว่าเวลาในการผลิตให้เสร็จสิ้นที่มากที่สุด (C_{max}) ดังสมการที่ (5) ส่วนข้อจำกัดด้านตัวแปรตัดสินใจนั้น จำนวนงานในตารางเวลาของกลุ่มงานที่ i บนเครื่องจักรที่ j (s_{ij}) และเวลาในการผลิตให้เสร็จสิ้นที่มากที่สุด (C_{max}) จะต้องมีค่าเป็นจำนวนเต็มบวกเสมอ ดังสมการที่ (6)

2.2 ปัญหาการตัดวัสดุ (Cutting stock problem)

เป็นการตัดวัสดุขนาดมาตรฐานให้ได้ตามจำนวน d_i ชิ้น ที่มีขนาดความยาว l_i; i = 1, . . . , n โดยที่ต้องการตัดวัสดุขนาดมาตรฐานในจำนวนที่น้อยที่สุด นั่นก็คือต้องการให้เกิดการสูญเสียวัสดุต่ำที่สุดนั่นเอง ดังรูปที่ 1 Gilmore และ Gomory [6] ได้เสนอวิธีการใช้โปรแกรมเชิงเส้นตรง โดยการใช้วิธีการในปัญหาแน็ปแซค (Knapsack problem) ในการสร้างสดมภ์ (Column generation) โดยกำหนดให้ วัสดุขนาดมาตรฐานมีความยาว L; $\max_{1 \leq i \leq n} l_i \leq L$ ที่มีเวคเตอร์ของรูปแบบการตัดเป็น $(a_1, a_2, \ldots, a_n)$ ที่ประกอบด้วยขนาดความยาว l_1 จำนวน a_1 ชิ้น ขนาดความยาว l_2 จำนวน a_2 ชิ้น เป็นต้น ดัชนี j กำหนดลำดับของรูปแบบการตัด และมีรูปแบบจำลองทางคณิตศาสตร์ดังต่อไปนี้

a_{ij} = จำนวนชิ้นของขนาดความยาว l_i ที่ตัดโดยรูปแบบการตัดที่ j

x_j = จำนวนครั้งที่รูปแบบการตัดที่ j ถูกใช้

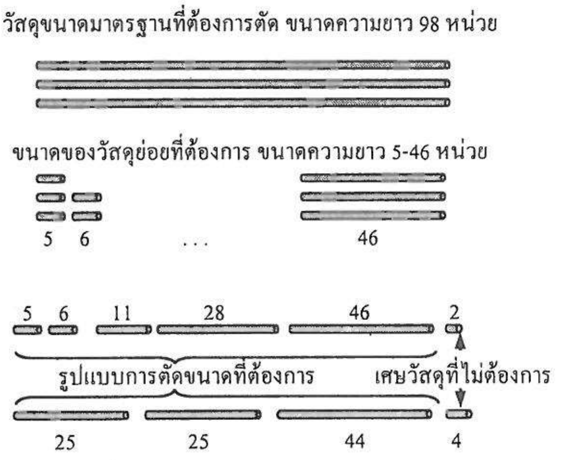
d_i = จำนวนที่ต้องการของขนาดความยาว l_i

$$Z = \min \sum_{j=1}^{m} c_j x_j \tag{7}$$

ข้อจำกัด

$$\sum_{j=1}^{m} a_{ij} x_j \geq d_i \tag{8}$$

$$x_j \geq 0 \text{ ; เลขจำนวนเต็ม} \tag{9}$$



รูปที่ 1 ลักษณะของการตัดวัสดุมิติเดียว

โดยที่ $a_{ij} x_j$ = จำนวนชิ้นของขนาดความยาว l_i ที่ได้จากการตัดรูปแบบ j และมีความต้องการใช้จำนวนของวัสดุขนาดมาตรฐานให้ต่ำที่สุด ถึงแม้ว่าจำนวนของรูปแบบการตัดและจำนวนความต้องการจะมากมาย จนทำให้ปัญหามีขนาดใหญ่มากก็ตาม โปรแกรมเชิงเส้นตรงก็สามารถแก้ปัญหา (7) – (9) ได้โดยง่ายด้วยวิธีซิมเพลกซ์ ส่วนการสร้างสดมภ์นั้นเป็นปัญหาย่อยสำหรับสร้างสัมประสิทธิ์ a_{ij} ด้วยหลักการของปัญหาแน็ปแซค โดยที่สัมประสิทธิ์ต้นทุนของตัวแปรนอนเบสิค x_j คือ $\overline{C}_j = 1 - \sum_{i=1}^{n} \pi_i a_{ij}$ โดยที่ π_i เป็นซิมเพลกซ์มัลติไพรเออร์จาก (8) ในปัญหาค่าต่ำสุดดังกล่าว ดังนั้นเพื่อหาต้นทุนลด (Reduced Cost) ที่เป็นลบมากที่สุด จึงสามารถสร้างปัญหาย่อยได้เป็น k ปัญหาดังนี้

$$\nu^k = \min_{1\le j\le m} \left(1 - \sum_{i=1}^{n} \pi_i^k a_{ij}\right) \qquad (10)$$

สำหรับแต่ละรูปแบบการตัดนั้นต้องมีข้อจำกัด $\sum_{i=1}^{n} l_i a_i \le L$ และ $a_i \ge 0$; จำนวนเต็ม ดังนั้นปัญหาย่อยจึงสามารถลดปริมาณการคำนวณลงได้ โดยสัมประสิทธิ์ a_{ij} ของรูปแบบการตัดใหม่จะทำให้ (10) ลดลงหรือไม่ก็เท่าเดิม โดยกำหนดให้เป็น

$$Z = \max \sum_{i=1}^{n} \pi_i a_i \qquad (11)$$

ข้อจำกัด $$\sum_{i=1}^{n} l_i a_i \le L \qquad (12)$$

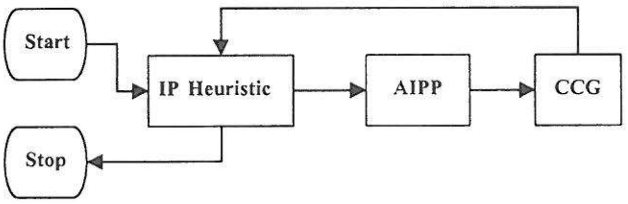
$$a_i \ge 0 \text{ ; เลขจำนวนเต็ม} \qquad (13)$$

รูปแบบของปัญหานี้เป็นปัญหาแบบแน็ปแซค ซึ่งถ้าค่าสูงสุด (11) เป็น ν ในกรณีที่ $\nu > 1$ จะสามารถกำหนดรูปแบบการตัดเพิ่มเติมได้ แล้วทำการคำนวณปัญหาหลักใหม่ แต่ถ้า $\nu \le 1$ แล้วนั้นแสดงว่าคำตอบที่ได้ดีที่สุดแล้ว สำหรับวิธีการสร้างสดมภ์ในปัญหานี้มีประสิทธิภาพมากในการหาคำตอบที่ดีได้ ถึงแม้ว่าวิธีการนี้จะต้องสร้างสดมภ์จำนวนมากก็ตาม จึงทำให้มีงานวิจัยที่จะลดปัญหาดังกล่าวนี้ตามมาอย่างมากมาย เริ่มต้นโดยผู้ริเริ่มวิธีการนี้เอง Gilmore และ Gomory [7] ได้แนะนำวิธีใหม่ที่รวดเร็วกว่าเดิม โดยการสร้างรูปแบบการตัดที่ดีในเบื้องต้นก่อน ต่อมา Haessler [11] ได้แนะนำวิธีการสร้างรูปแบบการตัดด้วยวิธีการสุ่มจัดลำดับ ซึ่งเป็นวิธีการเลือกสร้างรูปแบบการตัดในเบื้องต้นที่คำนึงถึงการสูญเสียที่ต่ำสุดจำนวนมากที่สุด และเก็บรูปแบบบางส่วนไว้เป็นส่วนผสมของรูปแบบต่อ ๆ ไป ต่อมา [12] ได้แนะนำการควบคุมการเปลี่ยนรูปแบบการตัด โดยการกำหนดค่าความสูญเสียให้กับการเปลี่ยนรูปแบบการตัดและให้มีมูลค่าเพิ่มสำหรับการใช้รูปแบบการตัดที่มีอยู่แล้วร่วมกับวิธีการสุ่มจัดลำดับข้างต้น ซึ่งพบว่าวิธีการนี้ไม่เหมาะสมในกรณีที่มีขนาดที่ต้องการแตกต่างกับขนาดมาตรฐานมาก ๆ ในปีถัดไป Golden [8] ได้ทำการศึกษาวิธีการแก้ปัญหาการตัด ซึ่งมีลักษณะคล้ายกับปัญหาการบรรจุกล่องและปัญหาการบรรทุก ที่สามารถหาคำตอบได้ภายใต้การเพิ่มขึ้นของเวลาแบบโพลีโนเมียล หลังจากนั้น ในปี ค.ศ. 1980 Haessler [13] ได้เสนอวิธีการควบคุมการสร้างรูปแบบการตัด เพื่อลดจำนวนรอบในการประมวลผลและการเปลี่ยนรูปแบบการตัด ในปีถัดไป Dyckhoff [2] ได้ปรับปรุงวิธีการดั้งเดิมด้วยเทคนิคพิเศษในการสร้างสดมภ์ โดยการใช้รูปแบบการตัดที่ง่าย ๆ อย่างพลวัตที่สามารถแทนรูปแบบการตัดที่มากมายได้ ซึ่งเหมาะสำหรับการประยุกต์ใช้กับกรณีที่มีจำนวนขนาดที่ต้องการตัดและขนาดที่ต้องการจำนวนมาก ๆ ต่อมาในปี ค.ศ. 1990 Farley [4] มีความเห็นว่าลักษณะของปัญหานี้ไม่มีความจำเป็นที่จะต้องประมวลผลจนกระทั่งได้ค่าที่ดีที่สุด การกำหนดขอบเขตให้กับวิธีการนี้จึงเป็นทางเลือกหนึ่งเพื่อลดเวลาในการประมวลผลลง Vanderbeck และ Wolsey [19] ได้เสนอวิธีการสร้างสดมภ์ในโปรแกรมเชิงจำนวนเต็มที่ประกอบไปด้วยสดมภ์จำนวนมาก โดยผสมผสานระหว่างวิธีแตกกิ่งและการประยุกต์ขอบเขตต่ำสุดของปัญหาย่อยเข้าด้วยกัน Vanderbeck [17] พบว่าจากตัวอย่าง 5-30 ขนาดที่ทำการศึกษาใช้เวลาในการประมวลผลเพียงไม่กี่วินาที ด้วยวิธีการ Branch-and-Price ซึ่งเป็นการผสมผสานระหว่างวิธี Branch-and-Bound ควบคู่ไปกับวิธีการสร้างสดมภ์ นั่นคือวิธีการสร้างสดมภ์ สามารถลดจำนวนของปัญหาย่อยลงให้น้อยที่สุดได้ นอกจากจะลดเวลาในการประมวลผลลงแล้ว ในทางปฏิบัติวิธีนี้ยังสามารถลดเวลา และต้นทุนที่เกิดจากการเตรียมงานในแต่ละรูปแบบการตัดอีกด้วย ต่อมาในปี 2000 [18] ได้เสนอวิธีการลดจำนวนของการเตรียมงานซึ่งเป็นปัญหาย่อยของปัญหานี้ให้น้อยที่สุด เนื่องจากการเปลี่ยนจากรูปแบบการตัดหนึ่งไปสู่อีกรูปแบบการตัดนั้นจะต้องมีการเตรียมงานใหม่เกิดขึ้น โดยใช้วิธีการสร้างสดมภ์ในการแก้ปัญหาโปรแกรมเชิงจำนวนเต็มที่มีปัญหาย่อยเป็นโปรแกรมเชิงจำนวนเต็มที่ไม่เป็นเส้นตรง ซึ่งจะถูกแยกสู่ปัญหาแน็ปแซคที่มีหลายขอบเขตในแต่ละ Branch-and-Bound ของโปรแกรมเชิงเส้นตรงแบบผ่อนปรนจะถูกจำกัด โดยอสมการที่เพิ่มเข้าไปด้วยการแตกกิ่งอย่างสมดุล ประกอบกับวิธีโดยประมาณ (Heuristic) ทำให้ผลของวิธี Branch-and-Price-and-Cut นี้สามารถหาค่าที่ดีที่สุดหรือไม่ก็ใกล้เคียงสำหรับปัญหาในการผลิตจริง จากข้างต้นทำให้เห็นถึงพัฒนาการของวิธีการสร้างสดมภ์ ที่ยังคงมีรูปแบบหลักเช่นเดิมจากแนวคิดของ

Gilmore และ Gomory ส่วนการพัฒนานั้นจะเป็นทางด้านวิธีการสร้างรูปแบบเบื้องต้นและการสร้างสดมภ์เท่านั้น

จากปัญหาการตัดวัสดุมิติเดียวข้างต้น ถ้าปัญหามีวัตถุประสงค์ที่คล้ายคลึงกัน เช่นปัญหาการบรรจุกล่องมิติเดียวที่ต้องการหาจำนวนที่น้อยที่สุดของกล่องที่มีความสูงเท่ากัน ปัญหาการจัดลำดับและตารางเวลาที่ต้องการหาจำนวนที่น้อยที่สุดของหน่วยผลิตที่มีเวลาในการผลิตคงที่ ก็สามารถประยุกต์ใช้วิธีการในการแก้ปัญหาการตัดมิติเดียวได้ เช่นเดียวกัน โดยถ้าเป็นปัญหาการจัดลำดับและตารางเวลานั้น จะเป็นการหาเวลาในการผลิตที่ต่ำที่สุด ในขณะที่มีจำนวนหน่วยผลิตที่คงที่ ซึ่งเป็นปัญหาหน่วยผลิตแบบขนาน โดยที่งานเป็นอิสระต่อกันและไม่มีข้อจำกัดด้านลำดับการทำงาน มีเวลาเสร็จสิ้นต่ำที่สุดเป็นการวัดสมรรถภาพซึ่งจัดอยู่ในประเภท $P//C_{max}$ ดังนั้นการประยุกต์ใช้วิธีการแก้ปัญหาการตัดวัสดุที่มีประสิทธิภาพในปัญหาการจัดลำดับและตารางเวลา ย่อมจะเป็นวิธีการที่มีประสิทธิภาพด้วยเช่นกัน

3. วิธีโปรแกรมเชิงจำนวนเต็มและสร้างสดมภ์ต่อเนื่อง



รูปที่ 2 แผนภาพการไหลของวิธีการ IPCCG

รูปแบบของวิธีการ IPCCG (Integer programming and continuous column generation) มีขั้นตอนดังแผนภาพการไหล รูปที่ 2 เป็นการผสมผสานระหว่างวิธี CCG ที่ประยุกต์มาจากวิธีการสร้างสดมภ์ที่ทรงประสิทธิภาพในปัญหาการตัดวัสดุ ร่วมกับวิธีการจัดลำดับตารางเวลาโดยใช้วิธีโปรแกรมเชิงจำนวนเต็มโดยประมาณ ซึ่งเป็นการใช้ประโยชน์จากผลลัพธ์ที่เป็นไปได้ ภายใต้การกำหนดขอบเขตในการประมวลผลจากโปรแกรมสำเร็จรูป LINGO 6 ที่ 2×10^5 ขั้นตอนการประมวลผลในการสร้างรูปแบบเบื้องต้นของตารางเวลา เนื่องจากการทดลองประมวลผลพบว่าผลลัพธ์จาก LINGO 6 ที่ 2×10^5 ขั้นตอนการประมวลผลนั้นมีผลลัพธ์ที่ดีกว่าผลลัพธ์จากวิธี LPT หลังจากนั้นจึงทำการปรับปรุงรูปแบบของตารางเวลาดังกล่าวให้อยู่ในขอบเขตล่าง หรือ C^*_{max} (2) ซึ่งเรียกขั้นตอนนี้ว่าการปรับรูปแบบเบื้องต้นของตารางเวลา (Adjusted IP Patterns หรือ AIPP) หลังจากนั้นทำการสร้างรูปแบบของตารางเวลาเพิ่มเติมด้วยวิธี CCG โดยโปรแกรมสำเร็จรูปไมโครซอฟท์เอ็กเซลร่วมกับโปรแกรมสำเร็จรูป LINDO 6.1 เพื่อตรวจสอบว่ารูปแบบของตารางเวลาอยู่ในขอบเขตล่างหรือไม่ ถ้าไม่อยู่ในขอบเขตดังกล่าวก็ทำการขยายขอบเขตล่าง แล้วจึงดำเนินวิธีการนี้เช่นเดิมจนกระทั่งสามารถทำการสร้างรูปแบบของตารางเวลาอยู่ภายในขอบเขตที่กำหนดได้เป็นอันสิ้นสุดวิธีการ ดังขั้นตอนต่อไปนี้

ขั้นตอนที่ 1 สร้างรูปแบบเบื้องต้นของตารางเวลา $S_j = \{s_{ij}\}$ โดยใช้วิธีโปรแกรมเชิงจำนวนเต็ม จากรูปแบบทางคณิตศาสตร์ (3) - (6)

ขั้นตอนที่ 2 ปรับปรุงรูปแบบของตารางเวลาให้อยู่ในขอบเขตล่าง ดังนี้

ขั้นตอนที่ 2.1 คำนวณขอบเขตล่าง (C^*_{max}) โดยกำหนดให้ $C = \sum_{i=1}^{n} p_i q_i$ แล้ว $C^*_{max} = \lceil C/m \rceil$; เลขจำนวนเต็ม

ขั้นตอนที่ 2.2 คำนวณเวลาเสร็จสิ้นของแต่ละเครื่องจักรที่ j ซึ่ง $C(S_j) = \sum_{i=1}^{n} p_i s_{ij}$

ขั้นตอนที่ 2.3 ตรวจสอบรูปแบบของตารางเวลา ว่าอยู่ในขอบเขตล่างหรือไม่ โดยเริ่มจาก $j = 1$ ถ้า $C(S_j) > C^*_{max}$ ให้ไปยังขั้นตอนที่ 2.4 นอกเหนือจากนี้ให้ $j = j + 1$ และทำการตรวจสอบรูปแบบดังกล่าวในขั้นตอนนี้จนกระทั่ง $j = m$ แล้วจึงหยุดขั้นตอนที่ 2 นี้

ขั้นตอนที่ 2.4 ทำการปรับปรุงรูปแบบของตารางเวลา ให้อยู่ภายในขอบเขตล่าง โดยเริ่มจาก $i = n$ ถ้า $s_{ij} > 0$ แล้ว $s_{ij} = s_{ij} - 1$ และไปยังขั้นตอนที่ 2.3 นอกเหนือจากนี้ให้ $i = n - 1$ แล้วทำการปรับปรุงรูปแบบดังกล่าวในขั้นตอนนี้จนกระทั่ง $i = 0$

ขั้นตอนที่ 3 สร้างรูปแบบปัญหาเช่นเดียวกับปัญหาการตัดวัสดุ (7 – 13) และทำการสร้างสดมภ์ จนกระทั่ง $V \leq 1$ (10) โดยกำหนดให้ $C_{LB} = C^*_{max}$

ปัญหาหลัก $Z = \min \sum_{j=1}^{m} x_j$ (14)

ข้อจำกัด $\sum_{j=1}^{m} s_{ij} x_j \geq q_i$; $\forall\ i = 1, 2, 3, \ldots, n$ (15)

$x_j \geq 0$ (16)

ปัญหาย่อย $Z = \max \sum_{i=1}^{n} \pi_i s_{ij}$ (17)

ข้อจำกัด $\sum_{i=1}^{n} p_i s_{ij} \leq C_{LB}$ (18)

$s_{ij} \geq 0$ และเป็นจำนวนเต็ม (19)

ขั้นตอนที่ 4 ตรวจสอบการสร้างสดมภ์ หรือรูปแบบของตารางเวลากับขอบเขตล่าง ถ้า $C_{LB} = C^*_{max}$ ให้ไปยังขั้นตอนที่ 5 นอกเหนือจากนี้ให้ $C_{LB} = C_{LB} + 1$ แล้วกลับไปยังขั้นตอนที่ 2

ขั้นตอนที่ 5 ลดรูปจำนวนของแต่ละรูปแบบของตารางเวลา (x_j) ให้เป็นจำนวนเต็มโดยการปัดลง และทำการเลือกรูปแบบของตารางเวลาดังกล่าวเก็บไว้ แล้วไปยังขั้นตอนที่ 6 แต่ถ้ารูปแบบของตารางเวลาที่ถูกเลือก $\sum_{j=1}^{m} x_j = 0$ ให้ $C_{LB} = C_{LB} + 1$ แล้วกลับไปยังขั้นตอนที่ 2

ขั้นตอนที่ 6 คำนวณปริมาณงาน (q_i) และจำนวนเครื่องจักร (m_j) ที่เหลือ

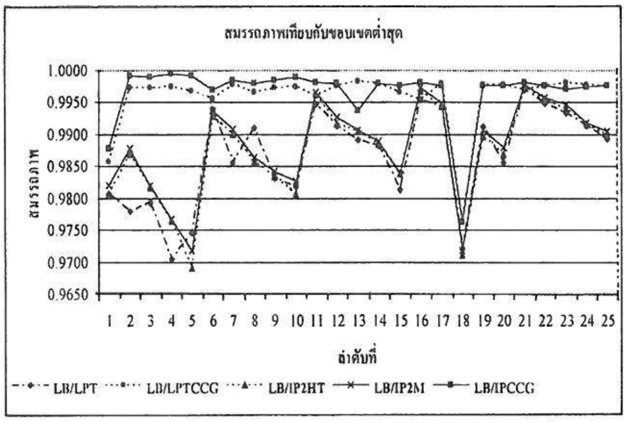
ขั้นตอนที่ 7 สร้างรูปแบบของตารางเวลาสำหรับปริมาณงาน (q_i) และจำนวนเครื่องจักร (m_j) ที่เหลือ โดยการใช้วิธีโปรแกรมเชิงจำนวนเต็มเช่นเดียวกับขั้นตอนที่ 1 ถ้า $C(S_j) \leq C_{LB}$ เป็นการสิ้นสุดวิธีการ นอกเหนือจากนี้ให้ $C_{LB} = C_{LB} + 1$ แล้วกลับไปยังขั้นตอนที่ 2

4. การประเมินสมรรถภาพของวิธีการ

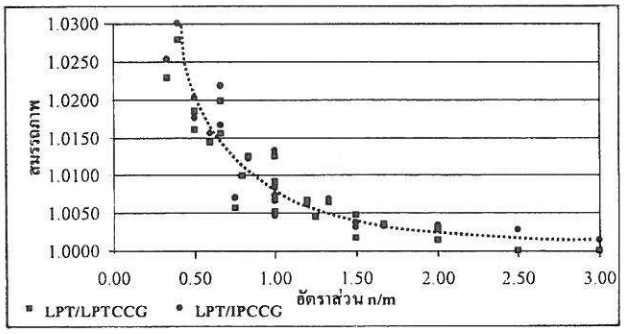
จากการประมวลผลภายใต้ขอบเขตที่ประกอบด้วย ตัวแปรหลัก 4 ตัวแปร คือจำนวนกลุ่มของงาน (n) จำนวนเครื่องจักร (m) เวลาในการผลิตต่อหน่วยสำหรับงานแต่ละกลุ่ม (p_i) และปริมาณความต้องการในการผลิตสำหรับงานแต่ละกลุ่ม (q_i) สำหรับจำนวนกลุ่มของงานและจำนวนเครื่องจักรนั้นมีจำนวน 10, 15, 20, 25 และ 30 ส่วนเวลาในการผลิตต่อหน่วยสำหรับงานแต่ละกลุ่มและปริมาณความต้องการในการผลิตสำหรับงานแต่ละกลุ่มนั้น ได้จากตัวเลขสุ่ม U[50,100] และU[10,50] ตามลำดับ ดังนั้นจึงมีกลุ่มตัวอย่าง 25 กลุ่ม ซึ่งในแต่ละกลุ่มมี 10 ตัวอย่าง ดังนั้นจึงมีตัวอย่างรวมทั้งสิ้น 250 ตัวอย่าง ทำการประมวลผลด้วยโปรแกรมที่ประยุกต์ใช้ภาษาวิชวลเบสิค ในโปรแกรมไมโครซอฟท์เอ็กเซลร่วมกับโปรแกรม LINDO 6.1 ในขั้นตอนการสร้างสดมภ์ต่อเนื่องและใช้โปรแกรม LINGO 6 ในขั้นตอนการสร้างรูปแบบของตารางเวลาด้วยโปรแกรมเชิงจำนวนเต็ม

จากการประมวลผลได้ผลลัพธ์ดังแสดงไว้ในตารางที่ 2 ที่แสดงรายละเอียดของอัตราส่วน n/m เวลาที่น้อยที่สุดในการผลิตที่มากที่สุดของวิธีการดังนี้ LB, LPT, LPTCCG, IP2HT (การประมวลผลด้วยโปรแกรม LINGO 6 ที่ 2×10^5 ขั้นตอน), IP2M (การประมวลผลด้วยโปรแกรม LINGO 6 ที่ 2×10^6 ขั้นตอน) และ IPCCG รวมทั้งเวลาในการประมวลผลของของวิธี LPTCCG และ IPCCG การเปรียบเทียบสมรรถภาพของวิธีการต่าง ๆ กับขอบเขตต่ำสุดแสดงไว้ในรูปที่ 3 ซึ่งมีสมรรถภาพเรียงลำดับจากมากไปหาน้อยดังนี้ IPCCG, LPTCCG, IP2M, IP2HT และ LPT ตามลำดับ

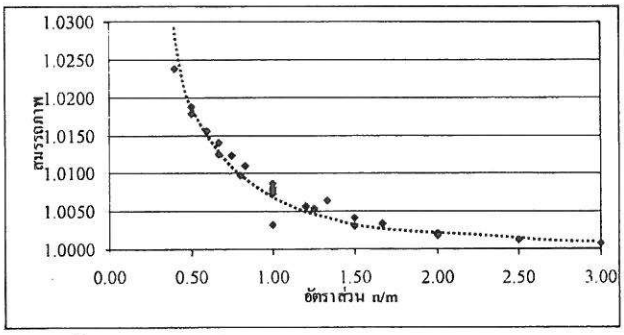
สำหรับอัตราส่วน n/m นั้นมีอิทธิพลต่อสมรรถภาพของวิธีการ LPTCCG และ IPCCG เมื่อเปรียบเทียบกับ LPT ดังรูปที่ 4 ซึ่งเห็นได้ว่าในขณะที่อัตราส่วน n/m มีค่าเพิ่มมากขึ้นจะทำให้สมรรถภาพทั้งของวิธีการ LPTCCG และ IPCCG ลดลงอย่างรวดเร็ว เพื่อแสดงผลการปรับปรุงรูปแบบของตารางเวลาด้วยวิธีการสร้างสดมภ์ จึงเปรียบเทียบความสัมพันธ์ระหว่างอัตราส่วน n/m กับสมรรถภาพของ IPCCG เปรียบเทียบกับ IP 2×10^5 ขั้นตอน ดังรูปที่ 5 เห็นได้ว่าอัตราส่วน n/m นั้นมีอิทธิพลต่อสมรรถภาพของวิธีการ IPCCG ในลักษณะเดียวกัน สำหรับความสัมพันธ์ระหว่างจำนวนกลุ่มของงาน กับเวลาในการประมวลผล IPCCG ดังรูปที่ 6 เห็นได้ว่าการเพิ่มขึ้นของจำนวนกลุ่มของงานนั้น แทบจะไม่มีผลต่อเวลาในการประมวลผลของ IPCCG แต่ในทางกลับกันเวลาในการประมวลผลโดยเฉลี่ยกลับลดลง ส่วนความสัมพันธ์ระหว่างจำนวนจำนวนเครื่องจักรกับเวลาในการประมวลผล IPCCG ดังรูปที่ 7 นั้น เห็นได้ว่าการเพิ่มขึ้นของเครื่องจักรมีผลต่อเวลาในการประมวลผลของ IPCCG อย่างมาก และสุดท้ายเป็นการเปรียบเทียบเวลาในการประมวลผลของวิธี LPTCCG และ IPCCG ดังแสดงในรูปที่ 8 เห็นได้ว่าเวลาในการประมวลผลของวิธี IPCCG นั้นใช้เวลาน้อยกว่าวิธี LPTCCG มากนัก



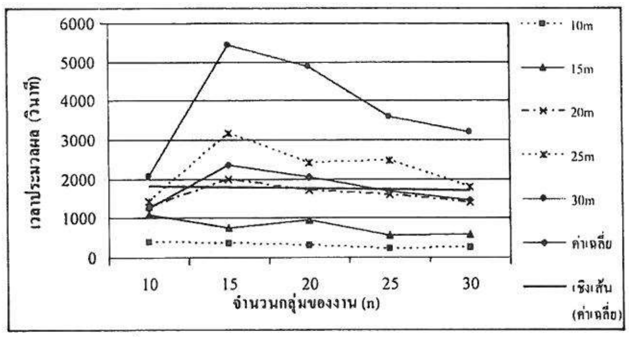
รูปที่ 3 สมรรถภาพของวิธีการต่าง ๆ เปรียบเทียบกับขอบเขตต่ำสุด



รูปที่ 4 ความสัมพันธ์ระหว่าง n/m กับสมรรถภาพวิธี LPTCCG, IPCCG



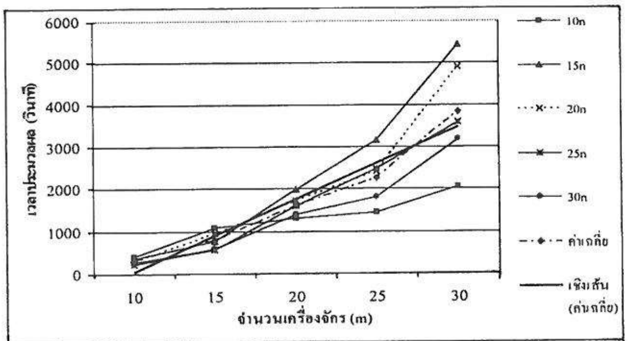
รูปที่ 5 ความสัมพันธ์ระหว่าง n/m กับสมรรถภาพของ IPCCG เทียบกับ IP 2×10^5 ขั้นตอน



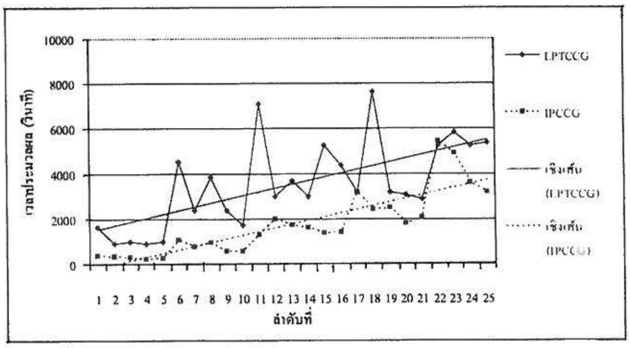
รูปที่ 6 ความสัมพันธ์ระหว่างจำนวนกลุ่มของงานกับเวลาในการประมวลผลของวิธี IPCCG

ตารางที่ 2 ผลลัพธ์ของเวลาที่น้อยที่สุดในการผลิตที่มากที่สุดและเวลาในการประมวลผล

ลำดับที่	จำนวนของ กลุ่มงาน (n)	จำนวนของ เครื่องจักร (m)	อัตราส่วน n/m	Cmax LB	Cmax LPT	Cmax LPTCCG	Cmax IP2HT	Cmax IP2M	Cmax IPCCG	เวลาประมวลผล(วินาที) LPTCCG	เวลาประมวลผล(วินาที) IPCCG
1	10	10	1.0000	2332.80	2378.70	2366.60	2378.80	2375.70	2361.50	1664.60	396.60
2	10	15	0.6667	1406.30	1438.10	1410.20	1425.00	1423.80	1407.50	925.00	359.50
3	10	20	0.5000	1226.30	1252.30	1229.70	1249.30	1248.70	1227.50	997.20	303.80
4	10	25	0.4000	988.60	1018.80	991.20	1012.60	1012.20	989.10	925.00	217.90
5	10	30	0.3333	781.90	802.30	784.40	806.90	804.60	782.50	989.60	262.30
6	15	10	1.5000	3477.50	3499.00	3493.00	3502.70	3499.30	3488.30	4535.90	1090.00
7	15	15	1.0000	2135.00	2166.40	2139.60	2156.60	2154.70	2138.10	2374.80	766.20
8	15	20	0.7500	1709.30	1724.80	1715.20	1733.90	1732.80	1712.70	3830.30	943.00
9	15	25	0.6000	1346.60	1369.60	1350.20	1369.50	1368.20	1348.60	2374.80	568.90
10	15	30	0.5000	1124.70	1145.60	1127.60	1146.90	1144.50	1125.80	1711.00	575.70
11	20	10	2.0000	4300.40	4323.00	4317.10	4316.40	4315.40	4308.30	7106.20	1302.10
12	20	15	1.3333	2911.10	2936.70	2918.00	2935.30	2932.30	2916.80	2987.60	1981.80
13	20	20	1.0000	2352.70	2378.40	2356.80	2375.20	2374.80	2367.50	3691.20	1708.30
14	20	25	0.8000	1937.20	1960.30	1941.10	1960.00	1958.70	1941.20	2987.60	1602.00
15	20	30	0.6667	1432.50	1459.70	1437.30	1456.00	1455.90	1435.90	5260.40	1398.10
16	25	10	2.5000	5731.20	5758.10	5758.10	5749.60	5746.40	5742.40	4368.50	1436.70
17	25	15	1.6667	3834.40	3856.10	3842.40	3856.30	3854.10	3843.70	3188.30	3162.70
18	25	20	1.2500	2727.40	2806.00	2793.80	2808.60	2806.50	2793.70	7621.50	2418.90
19	25	25	1.0000	2195.60	2215.20	2200.40	2218.50	2216.50	2200.80	3188.30	2476.00
20	25	30	0.8333	1878.00	1905.40	1882.00	1903.00	1901.00	1882.40	3047.50	1795.50
21	30	10	3.0000	6544.10	6564.70	6564.70	6560.90	6559.00	6555.90	2859.40	2062.10
22	30	15	2.0000	4359.40	4381.50	4369.70	4379.20	4378.00	4369.70	5241.00	5445.70
23	30	20	1.5000	3296.70	3318.60	3302.90	3316.40	3313.90	3306.30	5807.20	4891.70
24	30	25	1.2000	2623.40	2646.50	2629.00	2645.20	2644.80	2630.30	5241.00	3594.20
25	30	30	1.0000	2285.70	2310.50	2291.10	2309.10	2307.70	2291.20	5371.90	3188.30
ค่าเฉลี่ย				2597.55	2624.65	2608.48	2622.88	2621.18	2606.71	3531.83	1757.92



รูปที่ 7 ความสัมพันธ์จำนวนจำนวนเครื่องจักรกับเวลาในการประมวลผลของวิธี IPCCG



รูปที่ 8 การเปรียบเทียบเวลาในการประมวลผลวิธี LPTCCG กับ IPCCG

5. สรุป

สมรรถภาพของแต่ละวิธีการเทียบกับขอบเขตต่ำสุด เรียงลำดับจากมากไปหาน้อยดังนี้ IPCCG, LPTCCG, IP2M, IP2HT และ LPT ตามลำดับ สำหรับอัตราส่วน n/m นั้นเมื่อมีค่ามากขึ้นจะทำให้สมรรถภาพทั้งของวิธีการ LPTCCG และ IPCCG ลดลงอย่างรวดเร็ว อีกทั้งยังมีอิทธิพลต่อสมรรถภาพของวิธีการ IPCCG ในลักษณะเดียวกันในกรณีของการปรับปรุงรูปแบบของตารางเวลาด้วย CCG ส่วนความสัมพันธ์ระหว่างอัตราส่วน n/m กับสมรรถภาพของ IP ที่ 2×10^6 ขั้นตอนเปรียบเทียบกับ IP ที่ 2×10^5 ขั้นตอนนั้นไม่มีรูปแบบที่แน่นอนนัก สำหรับความสัมพันธ์ระหว่างจำนวนกลุ่มของงาน กับสมรรถภาพของ IPCCG โดยเฉลี่ยนั้นไม่เห็นแนวโน้มของความสัมพันธ์ชัดเจนนักแต่ก็มีแนวโน้มที่สูงขึ้น ส่วนความสัมพันธ์ระหว่างจำนวนเครื่องจักรกับสมรรถภาพของ IPCCG โดยเฉลี่ยก็เช่นเดียวกัน แต่มีแนวโน้มที่ต่ำลง สำหรับความสัมพันธ์ระหว่างจำนวนกลุ่มของงาน กับเวลาในการประมวลผล IPCCG นั้นมีแนวโน้มที่ค่อนข้างชัดเจน การเพิ่มขึ้นของจำนวนกลุ่มของงานนั้น แทบจะไม่มีผลต่อเวลาในการประมวลผล แต่ในทางกลับกันเวลาในการประมวลผลโดยเฉลี่ยกลับลดลง ส่วนความสัมพันธ์ระหว่างจำนวนเครื่องจักรกับเวลาในการประมวลผลนั้น การเพิ่มขึ้นของเครื่องจักรมีผลต่อการเพิ่มขึ้นเวลาในการประมวลผลอย่างมาก และสุดท้ายเวลาในการประมวลผลของวิธี IPCCG นั้นใช้เวลาน้อยกว่าวิธี LPTCCG มากนัก สำหรับการวิจัยเพิ่มเติมนั้นในแต่ละกลุ่มงานจะต้องรวมข้อจำกัดทางด้านเวลาเตรียมการผลิต (Setup time) และกำหนดเวลาส่งมอบ (Due date) เข้าไปด้วย

เอกสารอ้างอิง

[1] E.G. Coffman, M.R. Garey and D.S. Johnson. "An application of bin packing to multiprocessor scheduling", SIAM Computing, Vol.7, 1978, pp.1-17.

[2] H. Dyckhoff, "A new linear programming approach to the cutting stock problem", Operations Research, Vol.29, 1981, pp. 1092-1104.

[3] __________, "A typology of cutting and packing problems", European J. of Operational Research, Vol.44, 1990, pp.145-159.

[4] A.A. Farley, "A note on bounding a class of linear programming problems including cutting stock problems", Operations Research, Vol.38, No.5, 1990, pp.922-923.

[5] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Company, San Francisco, CA, 1979.

[6] P.C. Gilmore and R.E. Gomory, "A linear programming approach to the cutting-stock problem", Operations Research, Vol.9, 1961, pp.849-859.

[7] P.C. Gilmore and R.E. Gomory, "A linear programming approach to the cutting-stock problem: Part II", Operations Research, Vol.11, 1963, pp.863-888.

[8] B.L. Golden, "Approach to the cutting stock problem", AIIE Transactions, Vol.8, 1976, pp.265-274.

[9] R.L. Graham, "Bounds on multiprocessor timing anomalies", SIAM J. of Applied Mathematics, Vol.17, 1969, pp.416-429.

[10] J.N.D. Gupta and J. Ruiz-Torres, "A LISTFIT heuristic for minimizing makespan on identical parallel machines", Production Planning & Control, Vol.12, 2001, pp.28-36.

[11] R.W. Haessler, "A heuristic programming solution to a nonlinear cutting stock problem", Management Science, Vol.17, 1971, pp.793-802.

[12] __________, "Controlling pattern changes in one-dimensional cutting stock problems", Operations Research, Vol.23, 1975, pp.483-493.

[13] __________, "A note on computational modification to the Gilmore-Gomory cutting stock problem", Operations Research, Vol.28, 1980, pp.1001-1005.

[14] L.V. Kantorovich, "Mathematical methods of organization and planning production", Management Science, Vol.6, 1960, pp.366-442.

[15] C.Y. Lee and J.D. Massey, "Multiprocessor scheduling: combining LPT and MULTIFIT", Discrete Applied Mathematics, Vol.20, 1988, pp.233-242.

[16] S. Sukto and P. Charnsethikul, "A column generation approach for scheduling of N job-groups through M parallel machines with minimum makespan", Proceeding of Symposium in Production and Quality Engineering for Competitive Business Environment, Kasetsart University, Bangkok, Thailand, Jun. 2002, pp.131-145.

[17] F. Vanderbeck, "Computational study of a column generation for bin packing and cutting stock problems", Math. Programming Ser, Vol.86, 1999, pp.565-594.

[18] __________, "Exact algorithm for minimizing the number of setups in the one-dimensional cutting stock problem", Operations Research, Vol.48, No.6, 2000, pp.915-926.

[19] __________ and L.A. Wolsey. "An exact algorithm for IP column generation", Operations Research Letters, Vol.19, No.4, 1996, pp.151-159.