ใบรับรองวิทยานิพนธ์
บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมอุตสาหการ)
ปริญญา

วิศวกรรมอุตสาหการ	วิศวกรรมอุตสาหการ
สาขา	**ภาควิชา**

เรื่อง ปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน

Two-Stage Single Constrained Linear Fractional Programming Problem

นามผู้วิจัย นางสาวจารุวรรณ วิวัฒนาคม

ได้พิจารณาเห็นชอบโดย

ประธานกรรมการ ....
(รองศาสตราจารย์พีรยุทธ์ ชาญเศรษฐิกุล, Ph.D.)

กรรมการ ....
(ผู้ช่วยศาสตราจารย์จุฑา พิชิตลำเค็ญ, Ph.D.)

กรรมการ ....
(ผู้ช่วยศาสตราจารย์ธีรสิทธิ์ เกษตรเกษม, Ph.D.)

หัวหน้าภาควิชา ....
(ผู้ช่วยศาสตราจารย์อนันต์ มุ่งวัฒนา, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

....
(รองศาสตราจารย์วินัย อาจคงหาญ, M.A.)
คณบดีบัณฑิตวิทยาลัย

วันที่ 24 **เดือน** มีนาคม **พ.ศ.** 2549

วิทยานิพนธ์

เรื่อง

ปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน

Two-Stage Single Constrained Linear Fractional Programming Problem

โดย

นางสาวจารุวรรณ วิวัฒนาคม

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์
เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมอุตสาหการ)
พ.ศ. 2549

ISBN 974-16-1371-7

จารุวรรณ วิวัฒนาคม 2549: ปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมอุตสาหการ) สาขาวิชาวิศวกรรมอุตสาหการ ภาควิชาวิศวกรรมอุตสาหการ ประธานกรรมการที่ปรึกษา: รองศาสตราจารย์พีรยุทธ์ ชาญเศรษฐิกุล, Ph.D. 180 หน้า

ISBN 974-16-1371-7

งานวิจัยนี้เสนอการพัฒนาวิธีในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอนที่มีค่าข้างขวามือของข้อจำกัดเป็นเชิงสุ่ม โดยอาศัยพื้นฐานทฤษฎีกำหนดการเชิงเส้น กำหนดการเศษส่วนเชิงเส้น และกำหนดการเชิงเส้นสองขั้นตอน ลักษณะของปัญหาเป็นรูปแบบการหาค่าผลิตภาพที่เหมาะสมที่สุด ภายใต้ทรัพยากรที่จำกัดอันหนึ่ง ในการศึกษาครั้งนี้ ผู้วิจัยทำการพัฒนาวิธีการแก้ปัญหาเพื่อให้เกิดประสิทธิภาพและประสิทธิผล โดยทำการเปรียบเทียบผลลัพธ์ที่ได้จากวิธีการคำนวณโดยตรง ที่พัฒนาขึ้นซึ่งเป็นวิธีการหาค่าผลเฉลยทางคณิตศาสตร์ กับผลลัพธ์ที่ได้จากวิธีการใช้คำสั่งลินพร็อก (Linprog) ซึ่งเป็นวิธีการแก้ปัญหาแบบเชิงเส้น ในกล่องเครื่องมือของซอฟต์แวร์แมทแลป (MATLAB Software) โดยใช้เวลาในการคำนวณของแต่ละวิธีเป็นตัวเปรียบเทียบ ทำการศึกษาลักษณะปัญหาสองแบบคือ แบบที่ไม่มีการกำหนดขอบเขตของตัวแปรและแบบที่มีการกำหนดขอบเขตของตัวแปร เมื่อจำนวนเงื่อนไข (m) และจำนวนตัวแปรของปัญหา (n) มีการเปลี่ยนแปลงไป ผลจากการศึกษาพบว่า กรณีที่ปัญหาไม่มีการกำหนดขอบเขตวิธีที่ทำการพัฒนาขึ้นมาเฉพาะมีประสิทธิภาพดีกว่าในทุกกรณี ถึงแม้ว่าปัญหาจะมีขนาดใหญ่มากขึ้น ส่วนกรณีที่ปัญหามีการกำหนดขอบเขตของตัวแปร จะพบว่าวิธีการแก้ปัญหาโดยใช้คำสั่งลินพร็อกสามารถทำการแก้ปัญหาได้ดีกว่า เมื่อปัญหามีขนาดใหญ่ขึ้น

CHARUWAN W.

ลายมือชื่อนิสิต

ลายมือชื่อประธานกรรมการ

20 / 03 / 49

Charuwan Wiwattanakom 2006: Two-Stage Single Constrained Linear Fractional Programming Problem. Master of Engineering (Industrial Engineering), Major Field: Industrial Engineering, Department of Industrial Engineering. Thesis Advisor: Associate Professor Peerayuth Charnsethikul, Ph.D. 180 pages.
ISBN 974-16-1371-7

This research proposes a methodology development for solving a class of two-stage single constrained linear fractional programming problem with stochastic right-hand sides using basic theoretical linear programming, linear fractional programming and two-stage linear programming. The problem characteristic is a model for determining the optimal productivity under a restricted resource. In this study, an efficient and effective method is developed and compared with the direct method using linear programming solved by the command "linprog" available in the "Optimization" tool book in MATLAB software. Computational times by both methods are used for comparison. Two problem types, unbounded and bounded variables, are investigated by varying m (the number of right hand side alternatives) and n (the number of variables). The results indicate that for the case of unbounded variables, the proposed procedure is more efficient in all cases especially for large problems while in case of bounded variables; the direct method is more efficient as the problem size grows.

CHARUWAN W. ______ / ______ 20 / 03 / 06

Student's signature | Thesis Advisor's signature

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ ด้วยความกรุณา ความเอาใจใส่อย่างดี และความช่วยเหลือ แนะนำแนวทางต่าง ๆ รวมทั้งให้โอกาสได้ทำงานวิจัยในครั้งนี้จาก รศ.ดร.พีรยุทธ์ ชาญเศรษฐิกุล ประธานกรรมการ ผศ.ดร. ธีรสิทธิ์ เกษตรเกษม กรรมการสาขาวิชารอง ในการให้คำแนะนำและแนวทางในการทำงานวิจัย รวมไปถึง ผศ.ดร.จุฑา พิชิตลำเค็ญ กรรมการวิชาเอก และ รศ.ดร.อัศนี ก่อตระกูล ผู้แทนบัณฑิตวิทยาลัยที่ได้กรุณาให้คำแนะนำปรึกษา ตลอดจนแก้ไขวิทยานิพนธ์ให้สมบูรณ์ยิ่งขึ้น ต้องขอกราบพระคุณเป็นอย่างสูง ณ ที่นี้

ขอขอบคุณ เพื่อน ๆ ภาควิชาวิศวกรรมอุตสาหการ และ ภาควิชาวิศวกรรมเครื่องกลทุกท่าน และขอขอบคุณทุกท่านที่มิได้เอ่ยนาม ที่ได้ให้ความช่วยเหลือ และคำแนะนำในการแก้ไขปัญหาต่าง ๆ จากการทำงานวิจัยนี้ ให้สำเร็จลุล่วงได้ด้วยดี ข้าพเจ้าขอขอบคุณทุกคนในครอบครัวของข้าพเจ้าที่คอยเป็นกำลังใจและสนับสนุนในการทำงานวิจัย จนทำให้การศึกษานี้สำเร็จด้วยดี

จารุวรรณ วิวัฒนาคม

มีนาคม 2549

สารบัญ

สารบัญ (ต่อ)

สารบัญตาราง

สารบัญตาราง (ต่อ)

ตารางผนวกที่ **หน้า**

สารบัญตาราง (ต่อ)

สารบัญภาพ

สารบัญภาพ (ต่อ)

ปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน

Two-Stage Single Constrained Linear Fractional Programming Problem

คำนำ

งานวิจัยในด้านกำหนดการทางคณิตศาสตร์ (Mathematical Programming) มักจะทำการสมมุติข้อมูลที่ต้องการใช้เป็นค่าที่แน่นอน แต่ในเหตุการณ์จริงข้อมูลที่ได้มา บางครั้งไม่สามารถรู้ค่าที่แน่นอนได้ ดังนั้นเพื่อแก้ปัญหาการตัดสินใจภายใต้ความไม่แน่นอนดังกล่าว จึงจำเป็นต้องคำนึงถึง ความไม่แน่นอนที่จะต้องเผชิญ อิทธิพลที่มีผลต่อการตัดสินใจที่เหมาะสม และพิจารณาว่าแบบจำลองทางคณิตศาสตร์มีความเหมาะสมกับปัจจัยที่มีความไม่แน่นอนเข้ามาเกี่ยวข้องหรือไม่ ซึ่งในการแก้ปัญหาสามารถทำการพิจารณาออกเป็น 2 ส่วน คือ ส่วนที่เป็นเงื่อนไขของปัญหา (Constraints) และส่วนวัตถุประสงค์ของปัญหา (Objective Function)

โดยทั่วไปปัญหากำหนดการเชิงเส้น (Linear Programming) จะนิยมใช้วิธีแก้ปัญหาคือ วิธีซิมเพล็กซ์ (The Simplex Method) แต่บางวิธีเป็นวิธีการเฉพาะ เช่น ปัญหาการขนส่ง ปัญหาการจัดงาน (Scheduling) หรือปัญหาข่ายงาน (Network Flow Problem) เป็นต้น เพื่อเพิ่มประสิทธิภาพในการหาคำตอบ ดังนั้น ในการทำงานวิจัยนี้ เป็นการศึกษาหาวิธีการแก้ปัญหากำหนดการเชิงเส้นที่มีรูปแบบเฉพาะซึ่งมีลักษณะรวมกันของ ปัญหากำหนดการเศษส่วนเชิงเส้น (Linear Fractional Programming) และ ปัญหากำหนดการเชิงเส้นสองขั้นตอนแบบหนึ่งเงื่อนไข (Single Constrained Two-Stage Linear Programming) เพื่อค้นหาค่าคำตอบที่เหมาะสมของฟังก์ชันเป้าหมายของปัญหาที่สนใจ และมีหนึ่งเงื่อนไข แทนการแก้ปัญหาโดยใช้วิธีซิมเพล็กซ์โดยตรง และเมื่อข้อกำหนดของปัญหามีความซับซ้อนมากขึ้น เช่นจำนวนตัวแปรของปัญหา (n) มีค่ามาก ๆ และค่าจำนวนสมการ (m) ผันแปรตามจำนวนค่าที่เป็นไปได้ทางขวามือ จึงทำให้เกิดเป็นปัญหามีขนาดใหญ่ (Large-Scale Problem)

วิธีแก้ปัญหานี้สามารถนำไปประยุกต์ใช้ได้กับการแก้ปัญหาจริง ทั้งด้านการผลิตตลอดจนการขนส่งและด้านการเงิน เช่น ปัญหาการตัดแบ่งสต๊อก ปัญหาการผสมสินแร่ ปัญหาการจัดลำดับขนส่ง นโยบายที่เหมาะสมสำหรับลูกโซ่มาร์คอฟ (Markovian Chain) ปัญหาความไวของกำหนดการเชิงเส้น ค่าเหมาะสมของอัตราส่วนในแต่ละสถานการณ์ หรือเพื่อศึกษาประสิทธิภาพที่

มีความสัมพันธ์ในด้านที่แตกต่างกัน เช่น การศึกษา การดำเนินการของโรงพยาบาล ระบบสนามกีฬา การทำนุบำรุง สาขาย่อยธนาคาร เป็นต้น ซึ่งได้กล่าวไว้ใน Craven (1988)

ขอบเขตของงานวิจัย

1. งานวิจัยนี้นำเสนอวิธีการแก้ปัญหา 2 แบบคือ วิธีการหาค่าผลเฉลยสร้างขึ้นใน M-file โดยใช้คำสั่ง linprog ในกล่องเครื่องมือของโปรแกรมสำเร็จรูป MATLAB และวิธีตรง ซึ่งใช้การคำนวณทางคณิตศาสตร์โดยวิธีกำหนดการเชิงเส้นในการหาค่าผลเฉลยของปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เขียนขึ้นใน M-file ของโปรแกรมสำเร็จรูป MATLAB

2. ทำการทำลองกับปัญหา 2 ลักษณะ คือ เมื่อปัญหากำหนดตัวแปรแบบไม่มีขอบเขต (Unbounded Variables) และเมื่อปัญหามีการกำหนดตัวแปรแบบมีขอบเขต (Bounded Variables)

3. ทำการทดสอบวิธีการในการแก้ปัญหา กับปัญหาที่สร้างมาจากตัวเลขสุ่ม

ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะได้รับจากผลการทำวิจัย ทั้งในด้านของการแก้ปัญหา และประโยชน์ของงานที่จะนำไปปรับใช้คือ

1. ช่วยในการตัดสินใจในเรื่องการคำนวณหาค่าผลเฉลยของปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน ที่รวดเร็วขึ้น และมีประสิทธิภาพดีกว่าใช้วิธีตรง

2. สามารถนำไปใช้เป็นฐานความรู้ของในการแก้ปัญหาที่มีความซับซ้อนสูงขึ้น เช่นกรณีที่มีเงื่อนไขมากขึ้น หรือการเพิ่มเงื่อนไขของตัวแปรเลขจำนวนเต็ม เป็นต้น

3. เป็นพื้นฐานการนำเสนอแนวทางการแก้ปัญหาให้มีความเหมาะสมกับปัญหาในภาคอุตสาหกรรมมากขึ้น เนื่องจากความซับซ้อนของปัญหา และการแก้ปัญหาที่ต้องการความรวดเร็วในการตัดสินใจ

การตรวจเอกสาร

การศึกษานี้มีการพิจารณาแบบจำลองการตัดสินใจ ซึ่งมุ่งเน้นการหาแนวทางในการแก้ปัญหาเพื่อหาคำตอบของปัญหาอย่างมีประสิทธิภาพ โดยการศึกษา และ เรียนรู้การออกแบบวิธีการแก้ปัญหากำหนดการเชิงเส้นเฉพาะทางตามรูปแบบ ที่มีเอกลักษณ์ และความซับซ้อนมากขึ้น การศึกษาแนวความคิดในการหาวิธีแก้ปัญหาเฉพาะทาง สำหรับปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน ศึกษาหาปัจจัยที่มีผลกระทบต่อวิธีการที่พัฒนาขึ้น เพื่อประเมินค่าประสิทธิภาพ และประสิทธิผล ในการวิจัยนี้จึงได้อธิบายทฤษฎี และ หลักการพื้นฐานความรู้ทางคณิตศาสตร์ ได้แก่ กำหนดการเศษส่วนเชิงเส้น กำหนดการเชิงเส้นสองขั้นตอน และข้อปัญหาถุงเป้ที่นำไปสู่การพัฒนาปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน

1. <u>กำหนดการเศษส่วนเชิงเส้น (Linear Fractional Programming)</u>

ลักษณะของปัญหากำหนดการเศษส่วน เป็นอัตราส่วนของสองฟังก์ชัน สามารถนำไปใช้สำหรับการหาค่าผลเฉลยของปัญหาผลิตภาพ ประสิทธิภาพในการทำงาน ซึ่งรูปทั่วไปของกำหนดการเศษส่วนเชิงเส้นคือ (Bazaraa *et al*., 1979; Chakraborty and Gupta, 2002)

$$\begin{aligned} \text{Max} \quad & \frac{c^T x + \alpha}{d^T x + \beta} \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0, x \in R^n \\ & c, d \in R^n \\ & A \in R^{m \times n} \\ & \alpha, \beta \in R \end{aligned} \tag{1}$$

สำหรับบางค่าของ x ซึ่ง $d^T x + \beta$ มีค่าเท่ากับศูนย์ ในการหลีกเลี่ยงกรณีเช่นนี้ จะต้องทำการกำหนด ถ้า $[x \geq 0, Ax = b]$ แล้ว $[d^T x + \beta > 0]$ หรือ ถ้า $[x \geq 0, Ax = b]$ แล้ว $[d^T x + \beta < 0]$ เพื่อความสะดวก จะทำการสมมุติค่ากำหนดการเศษส่วนเชิงเส้นที่เงื่อนไขดังกล่าวนี้

$$[x \geq 0, Ax = b] \Rightarrow [d^T x + \beta > 0] \tag{2}$$

นิยามบท (Chakraborty and Gupta, 2002) ปัญหากำหนดการคณิตศาสตร์ 2 ปัญหา

1. ค่ามากที่สุดของฟังก์ชัน $F(x)$ โดยเงื่อนไขที่ x เป็นสมาชิกของ Δ
2. ค่ามากที่สุดของฟังก์ชัน $G(x)$ โดยเงื่อนไขของ x เป็นสมาชิกของ Γ จะกล่าวได้ว่า เป็นค่าเทียบเท่าถ้ามีการจับคู่หนึ่งต่อหนึ่ง $q(\cdot)$ ของเซตคำตอบที่เป็นไปได้ของ (i) บนเซตคำตอบที่เป็นไปได้ของ (ii) เช่น $F(x) = G(q(x))$ สำหรับทุกค่า x ที่เป็นสมาชิกของ Δ

ทฤษฎีบท สมมุติว่าไม่มีจุด $(y,0)$ ที่ y มากกว่าหรือเท่ากับศูนย์ เป็นคำตอบที่เป็นไปได้สำหรับกำหนดการเชิงเส้น (Chakraborty and Gupta, 2002) ดังนี้

$$\begin{aligned} \text{Max} \quad & c^T y + \alpha t \\ \text{s.t.} \quad & d^T y + \beta t = 1 \\ & Ay - bt = 0 \\ & t \geq 0, y \geq 0, y \in R^n, t \in R \end{aligned} \tag{3}$$

โดยสมมุติเงื่อนไข (2) เช่น ถ้า $[x \geq 0, Ax = b]$ แล้ว $[d^T x + \beta > 0]$แล้ว (1) จะมีค่าเทียบเท่ากับกำหนดการเชิงเส้น

พิสูจน์ (Chakraborty and Gupta, 2002) ค่าสัมบูรณ์ของ x เป็นคำตอบที่เป็นไปได้สำหรับ (1) กำหนดให้ $q(x) = (y,t)$ ด้วย $t = (d^T x + \beta > 0)^{-1}$ และ $y = tx$ ให้ $y \geq 0$, $t > 0$, $Ay - tb = t(Ax - b) = 0$, $d^T y + tb = t(d^T x + \beta) = 1$ ดังนั้น (y,t) เป็นคำตอบที่เป็นไปได้สำหรับกำหนดการเชิงเส้น (3) ในทางกลับกัน ถ้า (y,t) เป็นคำตอบที่เป็นไปได้สำหรับ (3) และไม่มีจุด $(y,0)$ เป็นคำตอบที่เป็นไปได้สำหรับ (3) แล้ว $t > 0$ และ $x = \dfrac{y}{t}$ โดย $x \geq 0$, $Ax - b = (Ay - tb)/t = 0$ ส่งผลให้ $/q(\cdot)/$ การจับคู่เซตคำตอบที่เป็นไปได้ของ (1) แบบหนึ่งต่อหนึ่งบนเซตคำตอบที่เป็นไปได้สำหรับ (3) นอกจากนี้ฟังก์ชันวัตถุประสงค์จะมีความสัมพันธ์โดย พิสูจน์ค่าเทียบเท่ากันแล้วได้

$$\frac{\left(c^T x + \alpha\right)}{\left(d^T x + \beta\right)} = \frac{\left(c^T y + \alpha t\right)}{\left(d^T y + \beta t\right)} = \frac{\left(c^T y + \alpha t\right)}{1}$$

วิธีของ Charnes and Cooper (1962) พิจารณาดังนี้

$$\text{Min} \quad \frac{p^T x + \alpha}{q^T x + \beta}$$
$$\text{s.t.} \quad Ax \le b$$
$$x \ge 0$$

สมมุติให้เซตของ $S = \{x : Ax \le b, x \ge 0\}$ และ $q^T x + \beta > 0$ สำหรับทุกค่า $x \in S$ กำหนดให้ $z = \dfrac{1}{q^T x + \beta}$ และ $y = zx$ ปัญหาข้างต้นสามารถแปลงเป็น

$$\text{Min} \quad p^T y + \alpha \ z$$
$$\text{s.t.} \quad Ay - bz \le 0$$
$$q^T y + \beta \ z = 1$$
$$y \ge 0$$
$$z \ge 0$$

ถ้า (y, z) เป็นคำตอบที่เป็นไปได้ของปัญหาข้างต้น แล้ว $y > 0$ ดังนั้น ถ้า $(\bar{y}, \bar{z})$ เป็นคำตอบที่เหมาะสมที่สุดสำหรับกำหนดการเชิงเส้น แล้ว $\bar{x} = \bar{y} / \bar{z}$ เป็นคำตอบที่เหมาะสมที่สุดสำหรับกำหนดการเศษส่วน ซึ่ง $A\bar{x} \le b$ และ $\bar{x} \ge 0$ ตามเงื่อนไขเบื้องต้น

ถ้า $q^T x + \beta < 0$ สำหรับทุกค่า $x \in S$ จะกำหนดให้ $-z = \dfrac{1}{q^T x + \beta}$ และ $y = zx$ ปัญหาข้างต้นสามารถแปลงเป็น

$$\text{Min} \quad -p^T y - \alpha z$$
$$\text{s.t.} \quad Ay - bz \le 0$$
$$-q^T y - \beta z = 1$$
$$y \ge 0$$
$$z \ge 0$$

ในทำนองเดียวกัน ถ้า $(\bar{y}, \bar{z})$ เป็นคำตอบที่เหมาะสมที่สุดสำหรับกำหนดการเชิงเส้น แล้ว $\bar{x} = \bar{y} / \bar{z}$ เป็นคำตอบที่เหมาะสมที่สุดสำหรับกำหนดการเศษส่วน ซึ่ง $A\bar{x} \le b$ และ $\bar{x} \ge 0$ ตามเงื่อนไขเบื้องต้น

ดังนั้นสามารถสรุปได้ว่า กำหนดการเศษส่วนเชิงเส้นสามารถหาคำตอบได้ด้วยปัญหากำหนดการเชิงเส้นเพิ่มหนึ่งตัวแปรและเพิ่มหนึ่งเงื่อนไข

2. <u>กำหนดการเชิงเส้นสองขั้นตอน (Two - Stage Linear Programming)</u>

กำหนดการเชิงเส้นสองขั้นตอนเป็นกำหนดการสโตแคสติก (Stochastic Programming) ซึ่งมีรูปแบบในการตัดสินใจปัญหาที่ข้อมูลมีความไม่แน่นอนเข้ามาเกี่ยวข้อง โดยพิจารณาความไม่แน่นอนของข้อมูลที่เป็นไปได้ของค่าสัมประสิทธิ์ของปัญหากำหนดการเชิงเส้น ลักษณะปัญหาชนิดนี้ได้ถูกกำหนดขึ้นในปี 1950 โดย Dantzig, Beale, Tintner, Charnes and Cooper

Wagner (1975) เริ่มทำการพิจารณาการให้เหตุผลสำหรับความไม่แน่นอนของความน่าจะเป็นของปัญหากำหนดการเชิงเส้น และ นำเสนองานในหลายส่วนเพื่อเป็นแนวทางสำหรับแก้ปัญหา ที่เรียกว่า แบบจำลองกำหนดการเชิงเส้นสโตแคสติก (Stochastic Linear Programming Model) และ แนะแนวทางในการทำแบบจำลองสองขั้นตอน สองอย่างคือ การทำแบบจำลองกำหนดการสโตแคสติกอย่างง่าย ให้ได้ปัญหากำหนดการเชิงเส้นทั่ว ๆ ไป และการขยายขนาดของปัญหา

พื้นฐานของกำหนดการเชิงเส้นสองขั้นตอนมาจากปัญหาหนึ่งขั้นตอน โดยสมมุติให้แบบจำลองของปัญหาที่แน่นอน โดย m เป็นจำนวนเงื่อนไขที่สามารถเกิดขึ้นได้ และ n เป็นจำนวนตัวแปร หรือปัจจัย ซึ่งสามารถเขียนเป็นรูปทั่วไป (Wagner, 1979) ดังนี้

$$
\begin{aligned}
\text{Max} \quad & \sum_{j=1}^{n} c_j x_j \\
\text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j = b_i \qquad ; \ i = 1,2,\ldots,m \\
& x_j \ge 0 \qquad ; \ j = 1,2,\ldots,n
\end{aligned}
$$

ขั้นแรกให้สมมติค่าสัมประสิทธิ์ของฟังก์ชันวัตถุประสงค์ (c_j) ไม่แน่นอน โดยทำการสุ่มหาค่าจริง ซึ่งสถานการณ์อาจจะขึ้นอยู่กับการวางแผน หรือ การพยากรณ์ แต่รู้ค่าที่แน่นอนของค่าสัมประสิทธิ์ของเงื่อนไข (a_{ij}) และ สัมประสิทธิ์ของค่าข้างขวา (b_i) ซึ่งในขั้นนี้การค่า x_j หาได้ไม่ยาก วัตถุประสงค์ที่เหมาะสม คือค่ามากที่สุดของค่าคาดหวังตาม ทฤษฎีบทของความเท่ากัน ความแน่นอนเชิงเส้น เมื่อ ค่า c_j ไม่แน่นอน แต่รู้ค่า a_{ij} และ b_i ที่แน่นอนจะได้

$$\text{Max} \quad \mathrm{E}\left[\sum_{j=1}^{n} c_j x_j\right]$$

$$\text{Max} \quad \sum_{j=1}^{n} \mathrm{E}\left[c_j\right] x_j$$

ดังนั้น ถ้ามีเพียงค่าสัมประสิทธิ์ของตัวแปรในวัตถุประสงค์ที่เป็นเชิงสุ่ม และเป็นอิสระต่อกัน คำตอบที่เหมาะสมสามารถหาได้จากกำหนดการเชิงเส้นที่เทียบเท่ากัน แต่ถ้าค่าตัวแปรตัวอื่นมีความไม่แน่นอนเข้ามาเกี่ยวข้อง จะทำให้ไม่สามารถหาค่าคำตอบได้ง่าย สมมติให้ค่าสัมประสิทธิ์ของปัญหาแบบสุ่ม เนื่องจากสถานการณ์จริงที่ค่าสัมประสิทธิ์ของตัวแปรต่าง ๆ ไม่สามารถรู้ค่าที่แน่นอนได้ ทำให้ยากที่จะตัดสินใจกำหนดค่าที่เหมาะสมของตัวแปรที่สนใจได้

โดยการแก้ปัญหาในรูปทั่วไปที่เรียกว่า แบบจำลองเชิงเส้นสองขั้นตอน (Two-Stage Linear Model) จากหลักการเบื้องต้นสามารถสรุปในรูปแบบสโตแคสติกได้ ดังนี้

1. ค่าของสมาชิกการสุ่มทุกตัวเป็นอิสระต่อกันของทุกค่า x_j
2. ระดับของ x_j สำหรับ $j = 1,2,...,k < n$ ถูกกำหนดที่ขั้นตอนแรก (The First Stage) ก่อนจะรู้ค่าแน่นอนของสมาชิกเชิงสุ่ม
3. เงื่อนไข $i = 1,2,...,g$ กำหนด เพียงตัวแปรเป็นขั้นตอนแรก (The First Stage Variables) และ รู้ค่าที่แน่นอนของ a_{ij} และ b_i โดย g ในที่นี้เป็นจำนวนตัวแปรในขั้นตอนแรก
4. ระดับที่เหมาะสมสำหรับตัวแปรขั้นตอนที่สอง (The Second Stage Variables) x_j สำหรับ $j = k+1,...,n$ ซึ่งจะถูกสร้างขึ้นหลังจากรู้ค่าตัวแปรเชิงสุ่ม
5. มีจำนวนที่นับได้ Q ของเซตที่เป็นไปได้ของค่า c_j สำหรับ $j = k+1,...,n$ และสำหรับค่า a_{ij} และ b_i เมื่อ $i = g+1,...,m$ และเมื่อ $j = 1,...,n$ ตั้งชื่อเซตนี้โดย c_{qj}, a_{qij}, b_{qj} และความน่าจะเป็นของสิ่งที่เกิดเป็น p_q สำหรับ $q = 1,2,...,Q$ โดยที่ q เป็นเซตของ Q

แสดงกฎการตัดสินใจที่เหมาะสม โดยโปรแกรมเชิงเส้นดังนี้

$$\text{Max} \quad \sum_{j=1}^{k} \mathrm{E}\left[c_j\right] x_j + \sum_{q=1}^{Q} p_q \left[\sum_{j=k+1}^{n} c_{qj} x_{qj} \right]$$

$$\text{s.t.} \quad \sum_{j=1}^{k} a_{ij} x_j = b_i \qquad ; \; i = 1,2,\ldots,g \quad \text{(ขั้นตอนแรก)}$$

$$\sum_{j=1}^{k} a_{qij} x_j + \sum_{j=k+1}^{n} a_{qij} x_{qj} = b_{qj} \quad \text{(กฎการตัดสินใจขั้นตอนที่สอง)}$$

สำหรับ ค่า $i = g+1,\ldots,m$ และ $q = 1,2,\ldots,Q$

$$\forall x_j \geq 0, \; \forall x_{aj} \geq 0$$

นอกจากนี้ได้เสนอการแปลงปัญหากำหนดการเชิงเส้นสโตแคสติกหลายขั้นตอน ในรูปของปัญหากำหนดการเชิงเส้นที่ใหญ่มากขึ้น นอกจากนี้แล้วยังนำเสนอการแก้ปัญหากำหนดการเชิงเส้นสองขั้นตอน ในแบบจำลองกำหนดการเชิงเส้นมาตรฐานที่เทียบเท่าแล้วแสดงกระบวนการตัดสินใจดังถัดไปนี้

วิธีของ Witchakul et al. (2004) เสนอดังนี้

การศึกษากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอนได้นำลักษณะการแก้ปัญหากำหนดการเชิงเส้นสองขั้นตอน ในแบบจำลองกำหนดการเชิงเส้นมาตรฐาน ซึ่งมีพื้นฐานของปัญหามากจาก Witchakul et al. (2004) นำเสนอปัญหากำหนดการเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต โดยมีลักษณะดังนี้

1. วัตถุประสงค์ และเงื่อนไขของปัญหามีลักษณะเป็นแบบเชิงเส้น
2. ปัญหามีหนึ่งเงื่อนไข แต่ค่าด้านขวาของเงื่อนไขสามารถมีได้หลายค่า
3. ค่าตัวแปรของปัญหาเป็นตัวแปรที่มีขอบเขต (Bounded variables)
4. การแก้ปัญหาทดลองแก้ปัญหาโดยสร้างปัญหาจากตัวเลขสุ่ม และทำการเพิ่มขนาดของปัญหาจนกลายเป็นปัญหาที่มีขนาดใหญ่ และซับซ้อน

รูปทั่วไปของกำหนดการเชิงเส้นสองขั้นตอนแบบหนึ่งเงื่อนไขและมีขอบเขต (Two - Stage Bounded Variables and Single Constrained Linear Programming) คือ

$$\text{Min } Z = \sum_{i=1}^{n} c_j x_j + \sum_{i=1}^{m} (g_i u_i + h_i v_i) \qquad ; c_j, g_i, h_i \geq 0$$

$$\text{s.t.} \quad \sum_{j=1}^{n} a_j x_j + u_i - v_i = b_i \qquad (i = 1,2,...,m \text{ และ } j = 1,2,...,n)$$

$$0 \leq x_j \leq t_j$$

$$u_j, v_i \geq 0$$

สมมุติให้

$$b_i \leq b_{i+1} \qquad (i = 1,2,...,m-1)$$

$$c_j / a_j \leq c_{j+1} / a_{j+1} \qquad (j = 1,2,...,n \text{ และ } j \neq k)$$

ถ้าคำตอบที่เหมาะสมของกำหนดการเชิงเส้นสองขั้นตอนหนึ่งเงื่อนไขเป็น

$$\begin{aligned} x_1 &= t_1 \\ x_2 &= t_2 \\ &\vdots \\ x_{k-1} &= t_{k-1} \\ x_k &= \frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \leq t_k \\ x_{k+1} &= 0 \\ &\vdots \\ x_n &= 0 \end{aligned}$$

กำหนดค่า k เป็นค่าที่ $0 \leq \dfrac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \leq t_k$ กำหนดค่า p เป็น $\left(-c_k / a_k + \sum_{i=p}^{m} g_i - \sum_{i=1}^{p-1} h_i\right) > 0$ และ $\left(c_k / a_k + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i\right) > 0$ โดย

$$v_i = b_p - b_i \text{ และ } u_i = 0 \qquad (i = 1,2,..., p-1)$$

$$u_i \text{ และ } v_i = 0 \qquad (i = p)$$

$$u_i = b_i - b_p \text{ และ } v_i = 0 \qquad (i = p+1, p+2,..., m)$$

พิสูจน์ เมื่อแทนค่า k และ p ตามเงื่อนไขข้างต้นจะได้

$$
\begin{aligned}
Z &= \sum_{\substack{j=1\\ j\neq k}}^{n} c_j x_j + c_k x_k + \sum_{i=1}^{p-1} g_i u_i + \sum_{i=1}^{p-1} h_i v_i + \sum_{i=p+1}^{m} g_i u_i \\
&\quad + \sum_{i=p+1}^{m} h_i v_i + g_p u_p + h_p v_p \qquad (4) \\
x_k &= \left(b_p - u_p + v_p - \sum_{j=1}^{k-1} a_j t_j - \sum_{j=k+1}^{n} a_j x_j \right) / a_k \\
v_i &= \sum_{\substack{j=1\\ j\neq k}}^{n} a_j x_j + a_k x_k + u_i - b_i \\
&= \sum_{j=1}^{k-1} a_j t_j + a_k \left(\left(b_p - u_p + v_p - \sum_{j=1}^{k-1} a_j t_j - \sum_{j=k+1}^{n} a_j x_j \right) / a_k \right) \\
&\quad + \sum_{j=k+1}^{n} a_j x_j + u_i - b_i \\
&= b_p - b_i + u_i - u_p + v_p \qquad (i = 1,2,\ldots, p-1) \\
u_i &= b_i - b_p + v_i + u_p - v_p \qquad (i = p+1, p+2,\ldots, m)
\end{aligned}
$$

แทนค่า u_i และ v_i ในสมการ (4) จะได้

$$
\begin{aligned}
Z &= \sum_{j=1}^{k-1} c_j t_j + c_k \left(\left(b_p - u_p + v_p - \sum_{\substack{j=1\\ j\neq k}}^{n} a_j x_j \right) / a_k \right) + \sum_{j=k+1}^{n} c_j x_j \\
&\quad + \sum_{i=1}^{p-1} g_i u_i + \sum_{i=1}^{p-1} h_i \left(b_p - b_i + u_i - u_p + v_p \right) + \sum_{i=p+1}^{m} h_i v_i \\
&\quad + \sum_{i=p+1}^{m} g_i \left(b_i - b_p + v_i + u_p - v_p \right) + g_p u_p + h_p v_p \\
&= \sum_{j=1}^{k-1} c_j t_j + c_k \left(\left(b_p - \sum_{j=1}^{k-1} a_j t_j \right) / a_k \right) + \sum_{i=1}^{p-1} h_i \left(b_p - b_i \right) \\
&\quad + \sum_{i=p+1}^{m} g_i \left(b_i - b_p \right) + \sum_{\substack{j=1\\ j\neq k}}^{n} \left(c_j - c_k a_j / a_k \right) x_j + \sum_{i=1}^{p-1} \left(g_i + h_i \right) u_i \\
&\quad + \sum_{i=p+1}^{m} \left(h_i + g_i \right) v_i + \left(-c_k / a_k + g_P + \sum_{i=p+1}^{m} g_i - \sum_{i=1}^{p-1} h_i \right) u_p
\end{aligned}
$$

$$+\left(c_k / a_k + h_p + \sum_{i=1}^{p-1} h_i - \sum_{i=p+1}^{m} g_i\right)v_p$$

จากสมมุติฐาน เมื่อ $c_j / a_j \le c_{j+1} / a_{j+1}$ ($j = 1,2,\ldots,n-1$) แล้ว $c_j - c_k a_j / a_k \ge 0 \rightarrow \sum_{j=k+1}^{n}\left(c_j - c_k a_j / a_k\right)x_j \ge 0$ เพื่อหาค่าที่น้อยที่สุดของ Z จะได้ x_j ($j = k+1, k+2,\ldots,n$) เท่ากับศูนย์

เนื่องจาก $g_i, h_i \ge 0, \sum_{i=1}^{p-1}\left(g_i + h_i\right)u_i \ge 0$ และ $\sum_{i=p+1}^{m}\left(h_i + g_i\right)v_i \ge 0$ เพื่อหาค่าที่น้อยที่สุดของ Z ได้ $u_i = 0$ ($i = 1,2,\ldots, p-1$) และ $v_i = 0$ ($i = p+1, p+2,\ldots,m$) ดังนั้น เลือก p ที่ $\left(-c_k / a_k + \sum_{i=p}^{m} g_i - \sum_{i=1}^{p-1} h_i\right)$ และ $\left(c_k / a_k + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i\right)$ มากกว่าศูนย์

3. <u>ข้อปัญหาถุงเป้ (Knapsack Problem)</u>

เนื่องด้วยลักษณะของปัญหาถุงเป้ มีข้อจำกัดที่คล้ายกับปัญหาที่ทำการศึกษาคือ มีข้อจำกัดเดียว และ ลักษณะของปัญหาแบบนี้เป็นลักษณะของปัญหากำหนดการสองขั้นตอนอย่างง่ายที่สุด ซึ่งปัญหาถุงเป้สามารถอธิบายได้โดย เมื่อพิจารณาถุงเป้ด้วยความสามารถความจุ c และ เซตของ n วัตถุ และวัตถุ j มีผลตอบแทน p_j และน้ำหนัก w_j กำหนดเวกเตอร์ของตัวแปรฐานสอง x_j ดังนี้ (Dantzig, 1963; Ahuja *et al*., 1993)

- ถ้าวัตถุ j ถูกเลือก $x_j = 1$
- ถ้าวัตถุ j ไม่ถูกเลือก $x_j = 0$ (สำหรับ $j = 1,2,3,\ldots,n$)

ข้อปัญหาถุงเป้จะทำการเลือกวัตถุด้วยค่าผลตอบแทนที่มากที่สุด โดยน้ำหนักไม่เกินความสามารถในการจุ c ข้อปัญหาถุงเป้สามารถเขียนในรูปทั่วไปได้ดังนี้

$$\text{Max } Z = \sum_{j=1}^{n} p_j x_j$$

$$\text{s.t.} \quad \sum_{j=1}^{n} w_j x_j \le c$$

$x_j = 0$ หรือ $x_j = 1$ (Integer Variables) เมื่อ $j = 1,2,3,...,n$

ความหลากหลายของข้อปัญหาถุงเป้ (Simm, 1997) ประกอบด้วย

1. ข้อปัญหาถุงเป้หลายตัวเลือก การแบ่งเซตวัตถุเป็นเซตย่อย และ หยิบวัตถุที่มากที่สุดในแต่ละเซตย่อย

2. ข้อปัญหาถุงเป้มีขอบเขต สำหรับทุก ๆ ค่า j มีปริมาณ b_j วัตถุ ของผลประโยชน์ในค่า p_j และ ขนาด w_j

3. ข้อปัญหารวมเซตย่อย ข้อปัญหาถุงเป้แบบทวิภาค โดย $p_j = w_j$ สำหรับทุกค่า j

4. ข้อปัญหาถุงเป้พหุคูณ ข้อปัญหาถุงเป้ด้วย m ถุงเป้ ความสามารถในการจุ c_j โดย $j = 1,2,3,...,n$

5. ข้อปัญหาถุงเป้สโทแคสติก การเข้ามาและการออกไปจากถุงเป้แบบสุ่ม ด้วยค่าตอบแทนสโทแคสติก และ เป้าหมายของความเป็นไปได้ที่มากที่สุดของการบรรลุค่าตอบ แทนทั้งหมด

ข้อปัญหาถุงเป้ ต่อมาได้มีผู้ศึกษาและทำการวิจัย ลักษณะของวัตถุประสงค์เป็นเศษส่วนของฟังก์ชันเชิงเส้นสองฟังก์ชัน โดยที่มีเงื่อนไขเพียงหนึ่งเงื่อนไข และเรียกปัญหานี้ว่า ข้อปัญหาถุงเป้เชิงเศษส่วน (Fractional Knapsack Problem)

นอกจากหลักการ และวิธีการ ซึ่งเป็นทฤษฎีเบื้องต้นในการศึกษาปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสอง ยังมีงานวิจัยหลายงานที่ได้ทำการศึกษาปัญหาที่เป็นแนวทางเป็นฐาน รวมไปถึงวิธีการแก้ปัญหาในวิธีต่าง ๆ โดยจะขอนำเสนอดังต่อไปนี้

Gilmore and Gomory (1963) อธิบายปัญหาการตัดสต๊อกในอุตสาหกรรมกระดาษ และแสดงให้เห็นว่าภายใต้สถานการณ์กำหนดเป็นจำนวนวัตถุดิบที่เสียที่น้อยที่สุด ซึ่งเกี่ยวกับความสำคัญของอัตราส่วนระหว่างวัตถุดิบที่สูญเสีย และ จำนวนวัตถุดิบที่ถูกใช้

Luhandjula (1984) แก้ปัญหา Multi-Objective Function Programming Problem: MOFPP โดยใช้หลักวิธีคลุมเคลือ และใช้ด้านภาษาในการศึกษาการแก้ปัญหา MOLFPP โดยตัวแปรของปัญหา แสดงถึงการกำหนดการตัดสินใจ

Birge and Holmes (1992) ศึกษาเทคนิคที่น่าสนใจการกระจายตำแหน่งบีคิว (the BQ decomposition) สำหรับกำหนดการเชิงเส้นมุมขอบคู่ควบ (Dual block-angular linear programs) ซึ่งพิจารณา 32 สถานการณ์ถึงการลดความพยายามในการหามุมภายในที่เป็นคำตอบ แต่การคำนวณมีความลำบากสำหรับเหตุการณ์ที่เกิดขึ้น ความน่าจะเป็นที่เป็นไปได้คือ เทคนิคการแบ่งคอลัมม์ (the column splitting technique) ซึ่งแบ่งคอลัมม์ที่มีความหนาแน่นออกเป็นส่วนน้อย ๆ ซึ่งการนำไปประยุกต์ใช้เป็นการศึกษาของ Lustig *et al.* (1991)

Bazaraa *et al.* (1993) อธิบายกำหนดการเศษส่วนเชิงเส้น (Linear Fractional Programming) โดยกล่าวถึงลักษณะปัญหาที่มีลักษณะเป็นเศษส่วน และ มีข้อจำกัด เป็นเชิงเส้น เสนอทฤษฎีบทที่เกี่ยวข้องและพิสูจน์โดยอ้างอิง Gilmore and Gomory และวิธีของ Charnes and Cooper (1962)

Birge and Louveaux (1997) แนะนำกำหนดการสโทแคสติก (Stochastic Programming) และยกตัวอย่างปัญหาที่เกี่ยวข้อง จัดแบ่งปัญหา และ แนะนำแบบจำลองกำหนดการสโทแคสติก รวมถึงอธิบายลักษณะสมบัติของปัญหา วิธีในการแก้ปัญหากำหนดการสโทแคสติกด้วยวิธีต่าง ๆ

Meszaros (1997) นำการประยุกต์ใช้วิธีจุดภายใน (Interior point method: IPM) ในการแก้ปัญหาคำตอบแน่นอนที่เทียบเท่ากับปัญหากำหนดการเชิงเส้นไม่แน่นอนสองขั้นตอน (Two-stage stochastic linear programming) โดยทำการพิสูจน์ว่าวิธีจุดภายในกับระบบที่มีการขยาย ซึ่งให้คำตอบของปัญหาที่ดีแต่นำไปใช้ได้ยาก และ มีประสิทธิภาพต่ำ พร้อมทั้งทำการเสนอวิธีใหม่สำหรับการแตกแยกย่อยของระบบที่มีการขยาย ด้วยปัญหาที่มีลักษณะเฉพาะเพื่อความรวดเร็วและง่ายในการแก้ปัญหากำหนดการเชิงเส้นไม่แน่นอนสองขั้นตอนที่มีขนาดใหญ่ พร้อมทั้งทำการเปรียบเทียบผลการทดลองกับวิธีฟอลวก (Folaug) ซึ่งเป็นวิธีที่ได้มีการเสนอก่อนหน้านี้

Calvete and Gale (1999) ศึกษากำหนดการสองระดับ (Bilevel programming) ซึ่งเกี่ยวกับปัญหาในการหาค่าที่เหมาะสมสองปัญหา โดยพื้นที่เงื่อนไขของปัญหาในระดับแรกปรากฏอยู่ในปัญหาระดับถัดไป โดยทำการศึกษาเป็นปัญหากำหนดการเชิงเส้นสองระดับ กับปัญหากำหนดการเศษส่วนเชิงเส้น โดยฟังก์ชันวัตถุประสงค์ในระดับแรกเป็นเชิงเส้น และ ฟังก์ชันวัตถุประสงค์ระดับที่สองเป็นเศษส่วนเชิงเส้น โดยพื้นที่ของคำตอบเป็นรูปหลายเหลี่ยม (Polyhedron) ในปัญหานี้ทำการหาค่าที่จุดมุมของรูปหลายเหลี่ยม นอกจากนี้ยังทำการศึกษาความสัมพันธ์ระหว่างคำตอบที่

เป็นไปได้ของปัญหา บนพื้นฐานของเทคนิคเมตริกซ์ย่อยของค่าสัมประสิทธิ์ (Coefficient submatrix) ที่มีความสัมพัทธ์กับตัวแปรระดับที่สอง

Sakawa and Nishizaki (2001) เสนอกำหนดการคลุมเครือ (Fuzzy programming)เกี่ยวกับปัญหาการหาค่าที่ดีที่สุดแบบเส้นตรงซึ่งเป็นปัญหาที่เป็นเศษส่วน ใช้วิธีฟัซซีสำหรับปัญหากำหนดการเศษส่วนเชิงเส้นสองระดับ หลังจากพิจารณาเป้าหมายคลุมเครือ ของการตัดสินในที่ระดับบน (Upper level) ที่มีประสิทธิภาพ แล้วจึงพิจารณาระดับที่พอใจเพิ่มขึ้นของระดับการตัดสินใจในเบื้องต้น ในกระบวนการศึกษาการแก้ปัญหาที่เหมาะสม สำหรับปัญหากำหนดการการคำนวณได้รวมวิธีแบ่งครึ่ง (Bisection method) วิธีเฟสหนึ่งของกำหนดการเชิงเส้น และการแปลงตัวแปร (Variable transformation) ของ Charnes *et al.* (1962)

Billionnet (2002) หาวิธีการแก้ปัญหาเพื่อหาค่าที่เหมาะสม เมื่อวัตถุประสงค์ของปัญหาเป็นอัตราส่วนของสองฟังก์ชันเชิงเส้น โดยทำการพัฒนาขั้นตอนวิธีสำหรับปัญหาเริ่มต้น และประมาณค่า ด้วยวิธี ขั้นตอนการประมาณแบบ 1/2 (1/2 – Approximation algorithm) สำหรับปัญหาข้อถุงเป้ 0–1 แบบเศษส่วน (Fractional 0–1 knapsack problem) และวิธีขั้นตอนประมาณพหุนาม (Fully polynomial– time approximations schemes) แบบเต็ม ทดสอบกับปัญหาที่ไม่มีขอบเขต

Chakravorty and Gupta (2002) ได้ทำการศึกษาการแก้ปัญหาของปัญหากำหนดการเศษส่วนเชิงเส้นเชิงเส้นหลายวัตถุประสงค์ และ เปรียบเทียบผลเฉลยของปัญหาโดยนำวิธีคลุมเครือ (Fuzzy method) เข้าใช้ในการแก้ไขปัญหา สำหรับการหาผลเฉลยของปัญหาหลายๆ วัตถุประสงค์

Chang (2002) ทำการศึกษาปัญหากำหนดการเศษส่วนโพสิโนเมียล (Posynomial fractional programming : PFP) ซึ่งเกิดจากการสรุปค่าที่น้อยที่สุดของผลหารหลาย ๆ เทอม เป็นส่วนประกอบของเทอมโพสิโนเมียลในฟังก์ชันวัตถุประสงค์ ในการคำนวณปัญหาอยู่บนพื้นฐานเทคนิคเชิงเส้นแบบช่วง (Piecewise) โดยทำการแปลงค่าโพสิโนเมียลเข้าสู่ผลรวมของค่าสัมบูรณ์ การศึกษาได้พยายามแก้ปัญหาเพื่อให้ค่าที่ได้เป็นคำตอบที่เหมาะสมแท้จริง (Global optimum)

Benati (2004) ศึกษาปัญหาทฤษฎีความเสี่ยง เพื่อใช้ความเสี่ยงเป็นตัววัดความสัมพันธ์ของฟังก์ชันการแจกแจงความน่าจะเป็น เพื่อแสดงผลประโยชน์ที่แย่ที่สุดของตัวแปรสุ่ม ประมาณการขาดทุนโดยใช้เครื่องมือทางการวิจัยดำเนินการ ซึ่งจะทำการแก้ปัญหากำหนดการเชิงจำนวนเต็ม

แบบเศษส่วน ด้วยเงื่อนไขเชิงเส้นหนึ่งเงื่อนไข เมื่อปัญหามีความซับซ้อนและเป็นกำหนดการไม่เชิงเส้น โดยทำการแก้ปัญหาแบบแจกแจง และนำวิธีปัญหาถุงเป้มาประยุกต์ใช้ ในการแก้ปัญหาแบบแตกขอบเขตและสาขา

Witchakul *et al.* (2004) เสนอวิธีการแก้ปัญหากำหนดการเชิงเส้นสองขั้นตอนโดยใช้วิธีให้ผลเฉลยโดยแปลงตัวแปรวิธีกำหนดการเชิงเส้นเป็นวิธีทางคณิตศาสตร์ แล้วทำการทำทดสอบโดยเขียนขั้นตอนวิธีในโปรแกรมสำเร็จรูป MATLAB เพื่อหาคำตอบที่รวดเร็ว และ ง่าย โดยกำหนดขนาดของปัญหาให้มีขนาดใหญ่ขึ้น พร้อมทั้งศึกษาหาปัจจัยที่มีผลต่อการหาคำตอบของแบบจำลองที่สร้างขึ้น

ในการศึกษานี้จะพิจารณารูปแบบใหม่โดยใช้หลักการ และ ทฤษฎีดังกล่าวเป็นพื้นฐาน ซึ่งมีชื่อว่ากำหนดการเศษส่วนเชิงเส้นแบบสองขั้นตอนหนึ่งเงื่อนไข ซึ่งทำการศึกษาตัวแปรที่ไม่กำหนดขอบเขต และ ตัวแปรที่มีการกำหนดค่าขอบเขตตามลำดับ

อุปกรณ์และวิธีการ

อุปกรณ์

1. เครื่องคอมพิวเตอร์จำนวน 1 ชุด
2. ซอฟต์แวร์สำเร็จ Microsoft Windows
3. ซอฟต์แวร์สำเร็จ Microsoft Office
4. ซอฟต์แวร์สำเร็จ Adobe Acrobat
5. ซอฟต์แวร์สำเร็จ MATLAB

วิธีการ

กำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน (Two-Stage Single Constrained Linear Fractional Programming) มีรูปทั่วไป คือ

$$\text{Min } Z = \frac{\sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m}\left(g_i u_i + h_i v_i\right)}{\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m}\left(p_i u_i + q_i v_i\right)} \quad ; \; j = 1,2,3,\ldots,n$$

$$\text{s.t.} \quad \sum_{j=1}^{n} a_j x_j + u_i - v_i = b_i \quad ; \; i = 1,2,3,\ldots,m$$

$$0 \le x_j \le t_j$$

$$u_i, v_i \ge 0$$

$$c_j, g_i, h_i \ge 0$$

$$d_j, p_i, q_i \ge 0$$

ลักษณะของปัญหาข้างต้น สามารถยกตัวอย่างการแก้ปัญหาทั่วไป เช่นการหาผลิตภาพระหว่างค่าใช้จ่ายในการจัดเก็บ และค่าใช้จ่ายในการผลิตสินค้า โดยมีเงื่อนไขในการจัดเก็บเพื่อให้เห็นภาพของปัญหาทั่วไปที่เกิดขึ้น ได้ชัดเจนขึ้น ดังนี้

x_j เป็นสินค้าชนิดที่ j ($j = 1,2,3,\ldots,n$)

c_j เป็นต้นทุนการจัดเก็บในการเลือกสินค้าชนิดที่ j (Storage Cost per unit)

d_j เป็นต้นทุนการผลิตในการเลือกสินค้าชนิดที่ j (Production Cost per unit)

a_j เป็นพื้นที่ในการจัดเก็บในการเลือกสินค้าชนิดที่ j (Production Cost per unit)

b_i เป็นห้องที่ใช้ในการจัดเก็บมีขนาดแตกต่างกันตามห้องที่ i ($i = 1,2,3,\ldots,m$)

u_i เป็นขนาดพื้นที่ที่เหลือจากการจัดเก็บขนาดที่ i (Penalty)

v_i เป็นขนาดพื้นที่ที่ไม่พอสำหรับการจัดเก็บขนาดที่ i (Penalty)

g_i, p_i เป็นต้นทุนเมื่อมีพื้นที่ที่เหลือจากการจัดเก็บ สำหรับการจัดเก็บและการผลิต ตามขนาดที่ i ตามลำดับ

h_i, q_i เป็นต้นทุนพื้นที่ที่ไม่พอสำหรับการจัดเก็บ สำหรับการจัดเก็บและการผลิต ขนาดที่ i ตามลำดับ

t_j เป็นจำนวนของสินค้าชนิดที่ j ที่สามารถผลิตได้

ในงานวิจัยจะนำเสนอการแก้ปัญหา 2 วิธีการ โดยใช้วิธีในการแปลงรูปทั่วไปของปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน เข้าสู่รูปของกำหนดการเชิงเส้น หาค่าผลเฉลยด้วยคำสั่ง Linprog ในกล่องเครื่องมือของโปรแกรมสำเร็จรูป MATLAB และ วิธีที่นำหลักการทางคณิตศาสตร์มาใช้ในการแก้ปัญหา ซึ่งจะเรียกว่าวิธีตรงในการคำนวณหาค่าผลเฉลยของปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน โดยพัฒนาในโปรแกรมสำเร็จรูป MATLAB พร้อมทั้งมีการศึกษากับปัญหาที่เงื่อนไขของตัวแปรไม่มีการกำหนดขอบเขต และเงื่อนไขของตัวแปรมีการกำหนดขอบเขตตามลักษณะของปัญหาจริงที่เกิดขึ้น

1. การแก้ปัญหาโดยใช้คำสั่ง linprog

ในการหาคำตอบของการศึกษานี้ ได้ทำการเขียนอัลกอริทึมในซอฟต์แวร์สำเร็จ MATLAB และทดสอบความสามารถในการแก้ปัญหาโดยกำหนดขนาดของปัญหาให้มีขนาดใหญ่ขึ้น เมื่อ n เป็นจำนวนตัวแปร และ m เป็นจำนวนเงื่อนไขที่สามารถเกิดขึ้นได้ แบบจำลองรูปทั่วไป

$$\text{Min } Z = \frac{\sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \left(g_i u_i + h_i v_i\right)}{\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m} \left(p_i u_i + q_i v_i\right)} \quad ; \; j = 1,2,3,\ldots,n$$

s.t. $\sum_{j=1}^{n} a_j x_j + u_i - v_i = b_i$; $i = 1,2,3,...,m$

$0 \le x_j \le t_j$

$u_i, v_i \ge 0$

$c_j, g_i, h_i \ge 0$

$d_j, p_i, q_i \ge 0$

ให้ $y_0 = \dfrac{1}{\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m} (p_i u_i + q_i v_i)}; y_0 > 0$

จะได้ $y_j = x_j \cdot y_0$

$r_i = u_i \cdot y_0$

$s_i = v_i \cdot y_0$

เปลี่ยนเป็นแบบจำลองในรูปแบบกำหนดการเชิงเส้น จะได้ระบบสมการเป็น

Min Z = $\sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} (g_i r_i + h_i s_i)$

s.t. $\sum_{j=1}^{n} a_j y_j + r_i - s_i = b_i y_0$

เพิ่มสมการเงื่อนไข $\sum_{j=1}^{n} d_j y_j + \sum_{i=1}^{m} (p_i r_i + q_i s_i) = 1$

$0 \le y_j \le t_j y_0$

$r_i, s_i \ge 0$

$c_j, g_i, h_i \ge 0$

$d_j, p_i, q_i \ge 0$

กรณีที่ไม่มีการกำหนดขอบเขตบนของตัวแปรจะให้ค่า t_j มีค่าอนันต์ (Infinite) ดังนั้นจะส่งผลให้สมการมีเงื่อนไขเป็น $x_j \ge 0$ และ $y_j \ge 0$ ตามลำดับ

ในการแก้ปัญหาจะพบว่า ในการสุ่มค่าเพื่อสร้างปัญหาในการทดสอบ จะต้องทำการปรับค่าสัมประสิทธิ์ของตัวแปร x_j ในสมการวัตถุประสงค์ เนื่องจากค่าสัมประสิทธิ์ที่ไม่เหมาะสมทำให้ วิธีนี้มีผลเฉลยที่ผิดจากเงื่อนไขเริ่มต้น ดังเช่นตัวอย่างต่อไปนี้

เมื่อกำหนดค่า $m = 10$ และ $n = 10$ ในกรณีที่ไม่มีขอบเขต เมื่อทำการสุ่มค่าเพื่อสร้างปัญหาจะได้

a_j = [56.1467 49.3389 99.3977 2.5676 25.9516 32.7163 12.2876 74.5932 62.9361 19.6595]

b_i = [4.3183 4.1590 7.1581 8.7649 2.6225 0.3044 6.4332 8.2456 7.7863 4.7526]

c_j = [40.0443 76.7515 38.3863 89.1410 40.7792 37.6830 8.8450 98.2577 92.1696 46.1905]

d_j = [46.3769 26.5954 0.3622 11.5757 55.8916 89.2269 90.1206 69.9883 23.3777 52.9502]

g_i = [0.8433 0.2426 0.3700 0.9646 0.4691 0.5329 0.0540 0.6223 0.6652 0.7595]

h_i = [0.6755 0.3271 0.9843 0.5123 0.8174 0.3945 0.4819 0.7229 0.5792 0.2225]

p_i = [0.2259 0.3554 0.6421 0.6430 0.0521 0.9925 0.3879 0.2991 0.2476 0.0504]

q_i = [0.5542 0.6391 0.2129 0.2264 0.0665 0.5469 0.3505 0.0594 0.6470 0.9731]

คำตอบที่ได้จากคำสั่ง Linprog คือ

y_j = [0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000]

r_i = [0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.5970 0.0000 0.0000]

s_i = [0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.5970 0.0000 0.0000]

y_0 = [0.0000]

Z = 0.3847

เช็คคำตอบของสมการ โดยแทนค่าในคำตอบในรูปทั่วไปของปัญหาเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอน โดยแทนคำตอบทุกคำตอบในสมการเงื่อนไข ทำให้เงื่อนไขไม่เป็นจริง ดังนี้

$x_j = y_j \cdot y_0$ = [0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000]

$u_i = r_i \cdot y_0$ = [0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000]

$v_i = s_i \cdot y_0$ = [0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000]

จาก $$\sum_{j=1}^{n} a_j x_j + u_i - v_i = b_i$$

แทนค่าจะได้

$0 \neq 4.3183$

$0 \neq 4.1590$

$0 \neq 7.1581$

$0 \neq 8.7649$

$0 \neq 2.6225$

$0 \neq 0.3044$

$0 \neq 6.4332$

$0 \neq 8.2456$

$0 \neq 7.7863$

$0 \neq 4.7526$

ในกรณีที่มีตัวแปรมีขอบเขต หรือมีค่า t_j ก็สามารถเกิดคำตอบในลักษณะเช่นเดียวกันทำให้เมื่อแทนคำตอบเข้าสู่ปัญหา และเงื่อนไขของปัญหาเริ่มต้นแล้ว ส่งผลให้ระบบสมการของปัญหาเริ่มต้นไม่เป็นจริง เพราะฉะนั้นในการคำนวณวิธีนี้จึงจำเป็นต้องคำนึงถึงสัมประสิทธิ์ ในที่นี้จึงทำการปรับค่าสัมประสิทธิ์ของตัวแปร x_j ในสมการวัตถุประสงค์ โดยได้เสนอการปรับค่า c_j และ d_j ไว้ในตารางภาคผนวกที่ ก1 และ ตารางภาคผนวกที่ ก2

ในการศึกษาวิธีนี้จะใช้ เครื่องมือของซอฟต์แวร์สำเร็จรูป MATLAB ที่ชื่อ linprog ในการนำมาแก้ปัญหา โดยเขียนเป็นโปรแกรมในรูปของเอ็มไฟล์ (M-file) พร้อมทั้งอิงคำตอบจากวิธีนี้เป็นหลัก

2. <u>กำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน (Two-Stage Single Constrained Linear Fractional Programming) โดยวิธีตรง</u>

รูปทั่วไปของกำหนดการเศษส่วนเชิงเส้นสองขั้นตอน คือ

$$\text{Min } Z = \frac{\sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m}\left(g_i u_i + h_i v_i\right)}{\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m}\left(p_i u_i + q_i v_i\right)} \qquad ; \ j = 1,2,3,\ldots,n$$

$$\text{s.t.} \quad \sum_{j=1}^{n} a_j x_j + u_i - v_i = b_i \qquad ; \ i = 1,2,3,\ldots,m$$

$$x_j, u_i, v_i \ge 0$$

$$c_j, g_i, h_i \ge 0$$

$$d_j, p_i, q_i \ge 0$$

สมการฟังก์ชันวัตถุประสงค์ต้องมี $\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m}\left(p_i u_i + q_i v_i\right) \ge 0$ ถ้าคำตอบที่เหมาะสมคือ

$$x_k = \frac{b_p}{a_k} \qquad ; \ p \in i$$

$i \neq p$ $\qquad a_k \dfrac{b_p}{a_k} + u_i - v_i = b_i$

$u_i = b_i - b_p + v_i$

$b_i > b_p, v_i = 0$ $\qquad u_i = b_i - b_p \qquad (i = p+1, p+2, \ldots, m)$

$b_i < b_p, u_i = 0$ $\qquad v_i = b_p - b_i \qquad (i = 1,2,\ldots, p-1)$

$i = p$ $\qquad u_p - v_p = 0 \quad \rightarrow \quad u_p, v_p = 0$

กำหนดให้
$$y_0 = \frac{1}{\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m} (p_i u_i + q_i v_i)} \quad ; \ y_0 > 0$$

$$y_j = x_j y_0 \quad ; \ y_j \geq 0$$

$$r_i = u_i y_0 \quad ; \ r_i \geq 0$$

$$s_i = v_i y_0 \quad ; \ s_i \geq 0$$

จะได้ระบบสมการเป็น

$$\text{Min } Z = \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} (g_i r_i + h_i s_i)$$

นำ y_0 คูณเงื่อนไขได้
$$\sum_{j=1}^{n} a_j y_j + r_i - s_i = b_i y_0$$

กำหนด
$$b_i \leq b_{i+1} \quad ; \ i = 1,2,\ldots, m-1$$

$$\frac{c_k}{a_k} \leq \frac{c_j}{a_j} \quad ; \ j > k$$

ถ้าคำตอบคือ

$$x_k = \frac{b_p}{a_k} \quad \rightarrow \quad y_k = \frac{b_p y_0}{a_k}$$

ดังนั้นจะได้

$$\text{Min } Z = \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} (g_i r_i + h_i s_i) \quad ; \ j = 1,2,3,\ldots, n$$

$$\sum_{j=1}^{n} a_j y_j + r_i + s_i = b_i y_0 \qquad ; \ i = 1,2,3,\ldots,m \qquad (5)$$

เพิ่มสมการเพื่อให้ระบบสมการสมบูรณ์

$$\sum_{j=1}^{n} d_j y_j + \sum_{i=1}^{m} \left(p_i r_i + q_i s_i \right) = 1 \qquad (6)$$

$$y_k = \frac{b_p y_0 - r_p + s_p - \sum\limits_{\substack{j=1 \\ j \neq k}}^{n} a_j y_j}{a_k}$$

$$s_i = \left(b_p - b_i\right) y_0 + r_i - r_p + s_p \qquad ; \ b_i < b_p$$

$$r_i = \left(b_i - b_p\right) y_0 + s_i + r_p - s_p \qquad ; \ b_i > b_p$$

จาก (6)

$$\sum_{\substack{j=1 \\ j \neq k}}^{n} d_j y_j + d_k y_k + \sum_{i=1}^{p-1} \left(p_i r_i + q_i s_i \right) + \sum_{i=p+1}^{m} \left(p_i r_i + q_i s_i \right) + p_p r_p + q_p s_p = 1$$

แทน y_k, s_i และ r_i

$$\sum_{\substack{j=1 \\ j \neq k}}^{n} d_j y_j + \frac{d_k}{a_k} \left(b_p y_0 - r_p + s_p - \sum_{\substack{j=1 \\ j \neq k}}^{n} a_j y_j \right) + \sum_{i=1}^{p-1} \left[p_i r_i + q_i \left\{ \left(b_p - b_i\right) y_0 + r_i - r_p + s_p \right\} \right]$$

$$\sum_{i=p+1}^{m} \left[p_i \left\{ \left(b_i - b_p\right) y_0 + s_i + r_p - s_p \right\} + q_i s_i \right] + p_p r_p + q_p s_p = 1$$

$$\frac{d_k b_p}{a_k} y_0 + \sum_{\substack{j=1 \\ j \neq k}}^{n} \left(d_j - \frac{a_j d_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} \left[\left(b_p - b_i\right) q_i y_0 \right] + \sum_{i=p+1}^{m} \left[\left(b_i - b_p\right) p_i y_0 \right] + \sum_{i=1}^{p-1} \left(p_i + q_i \right) r_i$$

$$\sum_{i=p+1}^{m} \left(p_i + q_i \right) s_i + \left(\frac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i \right) s_p + \left(-\frac{d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i \right) r_p = 1$$

กำหนดให้

$$z_0 = \sum_{\substack{j=1 \\ j \neq k}}^{n} \left(d_j - \frac{a_j d_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} \left(p_i + q_i \right) r_i + \sum_{i=p+1}^{m} \left(p_i + q_i \right) s_i + \left(\frac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i \right) s_p$$

$$+\left(-\frac{d_k}{a_k}-\sum_{i=1}^{p-1}q_i+\sum_{i=p}^{m}p_i\right)r_p$$

$$y_0=\left[(1-z_0)/\left(\frac{d_kb_p}{a_k}+\sum_{i=1}^{p-1}(b_p-b_i)q_i+\sum_{i=p+1}^{m}(b_i-b_p)p_i\right)\right]$$

แทนค่าในฟังก์ชันวัตถุประสงค์จะได้

$$\text{Min } Z=\sum_{\substack{j=1\\ j\neq k}}^{n}c_jy_j+c_ky_k+\sum_{i=1}^{p-1}(g_ir_i+h_is_i)+\sum_{i=p+1}^{m}(g_ir_i+h_is_i)+g_pr_p+h_ps_p$$

$$=\sum_{\substack{j=1\\ j\neq k}}^{n}c_jy_j+\frac{c_k}{a_k}\left(b_py_0-r_p+s_p-\sum_{\substack{j=1\\ j\neq k}}^{n}a_jy_j\right)+\sum_{i=1}^{p-1}g_ir_i+\sum_{i=p+1}^{m}h_is_i+g_pr_p$$

$$+h_ps_p+\sum_{i=1}^{p-1}h_i\left(b_py_0-b_iy_0+r_i-r_p+s_p\right)$$

$$+\sum_{i=p+1}^{m}g_i\left(b_iy_0-b_py_0+s_i+r_p-s_p\right)$$

$$=\frac{c_kb_py_0}{a_k}+\sum_{i=1}^{p-1}h_i\left(b_py_0-b_iy_0\right)+\sum_{i=p+1}^{m}g_i\left(b_iy_0-b_py_0\right)+\sum_{\substack{j=1\\ j\neq k}}^{n}\left(c_j-\frac{a_jc_k}{a_k}\right)y_j$$

$$+\sum_{i=1}^{p-1}(g_i+h_i)r_i+\sum_{i=p+1}^{m}(g_i+h_i)s_i+\left(\frac{-c_k}{a_k}+g_p-\sum_{i=1}^{p-1}h_i+\sum_{i=p+1}^{m}g_i\right)r_p$$

$$+\left(\frac{c_k}{a_k}+h_p+\sum_{i=1}^{p-1}h_i-\sum_{i=p+1}^{m}g_i\right)s_p$$

แทนค่า y_0

$$=\left[\begin{array}{l}\left\{\dfrac{c_kb_p}{a_k}+\displaystyle\sum_{i=1}^{p-1}h_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}g_i\left(b_i-b_p\right)\right\}\times\\ \left\{\dfrac{1-z_0}{\dfrac{d_kb_p}{a_k}+\displaystyle\sum_{i=1}^{p-1}(b_p-b_i)q_i+\sum_{i=p+1}^{m}(b_i-b_p)}\right\}\end{array}\right]$$

$$+\sum_{\substack{j=1\\ j\neq k}}^{n}\left(c_j-\frac{a_jc_k}{a_k}\right)y_j+\sum_{i=1}^{p-1}(g_i+h_i)r_i+\sum_{i=p+1}^{m}(g_i+h_i)s_i$$

$$+\left(\frac{-c_k}{a_k}-\sum_{i=1}^{p-1}h_i+\sum_{i=p}^{m}g_i\right)+\left(\frac{c_k}{a_k}+\sum_{i=1}^{p}h_i-\sum_{i=p+1}^{m}g_i\right)s_p$$

$$= \left[\left\{ \frac{\dfrac{c_k b_p}{a_k} + \sum_{i=1}^{p-1} h_i (b_p - b_i) + \sum_{i=p+1}^{m} g_i (b_i - b_p)}{\dfrac{d_k b_p}{a_k} + \sum_{i=1}^{p-1} (b_p - b_i) q_i + \sum_{i=p+1}^{m} (b_i - b_p)} \right\} \times 1 - z_0 \right]$$

$$+ \sum_{\substack{j=1 \\ j \neq k}}^{n} \left(c_j - \frac{a_j c_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} (g_i + h_i) r_i + \sum_{i=p+1}^{m} (g_i + h_i) s_i$$

$$+ \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i \right) r_p + \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i \right) s_p$$

กำหนดให้

$$z = \left\{ \frac{\dfrac{c_k b_p}{a_k} + \sum_{i=1}^{p-1} h_i (b_p - b_i) + \sum_{i=p+1}^{m} g_i (b_i - b_p)}{\dfrac{d_k b_p}{a_k} + \sum_{i=1}^{p-1} q_i (b_p - b_i) + \sum_{i=p+1}^{m} p_i (b_i - b_p)} \right\}$$

เปลี่ยน i เป็น e ($e = 1,2,3,\ldots,m$) ดังนั้น

$$z = \left\{ \frac{\dfrac{c_k b_p}{a_k} + \sum_{e=1}^{p-1} h_e (b_p - b_e) + \sum_{e=p+1}^{m} g_e (b_e - b_p)}{\dfrac{d_k b_p}{a_k} + \sum_{e=1}^{p-1} q_e (b_p - b_e) + \sum_{e=p+1}^{m} p_e (b_e - b_p)} \right\}$$

จัดเทอมในรูปของ z และ z_0

$$\text{Min } Z = z \times \left[1 - \left(\begin{array}{l} \sum_{\substack{j=1 \\ j \neq k}}^{n} \left(d_j - \dfrac{a_j d_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} (p_i + q_i) r_i + \sum_{i=p+1}^{m} (p_i + q_i) s_i \\ + \left(\dfrac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i \right) s_p + \left(-\dfrac{d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i \right) r_p \end{array} \right) \right]$$

$$+ \sum_{\substack{j=1 \\ j \neq k}}^{n} \left(c_j - \frac{a_j c_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} (g_i + h_i) r_i + \sum_{i=p+1}^{m} (g_i + h_i) s_i$$

$$+ \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i \right) r_p + \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i \right) s_p$$

$$= z + \sum_{\substack{j=1 \\ j \neq k}}^{n} y_j A_j + \sum_{i=1}^{p-1} r_i U_i + \sum_{i=p+1}^{m1} s_i V_i + s_p V_p + r_p U_p$$

โดยกำหนดให้

$$A_j = \left(c_j - \frac{a_j c_k}{a_k}\right) - \left(d_j - \frac{a_j d_k}{a_k}\right) z$$

$$U_i = (g_i + h_i) - (p_i + q_i) z \qquad ; i = 1,2,\ldots, p-1$$

$$V_i = (g_i + h_i) - (p_i + q_i) z \qquad ; i = p+1, p+2,\ldots, m$$

$$V_p = \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i\right) - \left(\frac{d_k}{a_k} - \sum_{i=1}^{p} q_i + \sum_{i=p+1}^{m} p_i\right) z$$

$$U_p = \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i\right) - \left(\frac{-d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i\right) z$$

จากข้อกำหนด $a_j, c_j, d_j, y_j \geq 0$ แล้ว

$$\sum_{\substack{j=1 \\ j \neq k}}^{n} \left[\left(c_j - \frac{a_j c_k}{a_k}\right) - \left(d_j - \frac{a_j d_k}{a_k}\right) \left\{ \frac{\dfrac{c_k b_p}{a_k} + \sum_{e=1}^{p-1} h_e (b_p - b_e) + \sum_{e=p+1}^{m} g_e (b_e - b_p)}{\dfrac{d_k b_p}{a_k} + \sum_{e=1}^{p-1} q_e (b_p - b_e) + \sum_{e=p+1}^{m} p_e (b_e - b_p)} \right\} \right] \geq 0$$

เพื่อให้ค่าวัตถุประสงค์ที่น้อยที่สุด ทำให้ $y_j = 0$ สำหรับ $j \neq k, j = 1,2,..,n$

จากข้อกำหนด $g_i, h_i, r_i, s_i \geq 0$ แล้ว

$$\sum_{i=1}^{p-1} \left[(g_i + h_i) - (p_i + q_i) \left\{ \frac{\dfrac{c_k b_p}{a_k} + \sum_{e=1}^{p-1} h_e (b_p - b_e) + \sum_{e=p+1}^{m} g_e (b_e - b_p)}{\dfrac{d_k b_p}{a_k} + \sum_{e=1}^{p-1} q_e (b_p - b_e) + \sum_{e=p+1}^{m} p_e (b_e - b_p)} \right\} \right] \geq 0,$$

$$\sum_{i=p+1}^{m} \left[(g_i + h_i) - (p_i + q_i) \left\{ \frac{\dfrac{c_k b_p}{a_k} + \sum_{e=1}^{p-1} h_e (b_p - b_e) + \sum_{e=p+1}^{m} g_e (b_e - b_p)}{\dfrac{d_k b_p}{a_k} + \sum_{e=1}^{p-1} q_e (b_p - b_e) + \sum_{e=p+1}^{m} p_e (b_e - b_p)} \right\} \right] \geq 0$$

ค่าวัตถุประสงค์ที่น้อยที่สุด ทำให้ $r_i = 0$ สำหรับ $i = 1,2,.., p-1$ และ $s_i = 0$ สำหรับ $i = p+1, p+2,.., m$ เช่นเดียวกันกับ ค่าวัตถุประสงค์ที่น้อยที่สุด ที่ทำให้ $r_p, s_p = 0$ เมื่อ

$$\left(\frac{c_k}{a_k}+\sum_{i=1}^{p}h_i-\sum_{i=p+1}^{m}g_i\right)-\left(\frac{d_k}{a_k}-\sum_{i=1}^{p}q_i+\sum_{i=p+1}^{m}p_i\right)\left\{\frac{\frac{c_k b_p}{a_k}+\sum_{e=1}^{p-1}h_e\left(b_p-b_e\right)+\sum_{e=p+1}^{m}g_e\left(b_e-b_p\right)}{\frac{d_k b_p}{a_k}+\sum_{e=1}^{p-1}q_e\left(b_p-b_e\right)+\sum_{e=p+1}^{m}p_e\left(b_e-b_p\right)}\right\}\geq 0,$$

$$\left(\frac{-c_k}{a_k}-\sum_{i=1}^{p-1}h_i+\sum_{i=p}^{m}g_i\right)-\left(\frac{-d_k}{a_k}-\sum_{i=1}^{p-1}q_i+\sum_{i=p}^{m}p_i\right)\left\{\frac{\frac{c_k b_p}{a_k}+\sum_{e=1}^{p-1}h_e\left(b_p-b_e\right)+\sum_{e=p+1}^{m}g_e\left(b_e-b_p\right)}{\frac{d_k b_p}{a_k}+\sum_{e=1}^{p-1}q_e\left(b_p-b_e\right)+\sum_{e=p+1}^{m}p_e\left(b_e-b_p\right)}\right\}\geq 0$$

เมื่อ $i = p$ และ $j = k$

$$y_k=\left[b_p y_0-r_p+s_p-\sum_{\substack{j=1\\ j\neq k}}^{n}a_j y_j\right]/a_k \quad\rightarrow\quad y_k=b_p y_o/a_k$$

$$s_i=\left(b_p-b_i\right)y_0+r_i-r_p+s_p \quad\rightarrow\quad s_i=\left(b_p-b_i\right)y_0$$

$$r_i=\left(b_i-b_p\right)y_0+s_i+r_p-s_p \quad\rightarrow\quad r_i=\left(b_i-b_p\right)y_0$$

คำตอบที่เหมาะสมคือ

$$\text{Min } Z = \left\{\frac{\frac{c_k b_p}{a_k}+\sum_{i=1}^{p-1}h_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}g_i\left(b_i-b_p\right)}{\frac{d_k b_p}{a_k}+\sum_{i=1}^{p-1}q_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}p_i\left(b_i-b_p\right)}\right\}$$

$$x_k=b_p/a_k$$

$$u_i=b_i-b_p \quad\rightarrow\quad u_p=0$$

$$v_i=b_p-b_i \quad\rightarrow\quad v_p=0$$

$$y_0=\frac{1}{d_k x_k+\sum_{i=1}^{p-1}p_i\left(b_i-b_p\right)+\sum_{i=p+1}^{m}q_i\left(b_p-b_i\right)}$$

จากระบบสมการข้างต้นจะเห็นได้ว่า เมื่อแปลงเป็นระบบสมการเชิงเส้น สมการเงื่อนไขที่เพิ่มขึ้น (10) เป็นปัญหากำหนดการขั้นแรก เนื่องจากค่าสัมประสิทธิ์ทุกค่าสามารถรู้ค่าแน่นอน แต่จะเห็นว่าสมการเดิม (9) เป็นปัญหากำหนดการขั้นที่สอง เนื่องจากค่าข้างขวาไม่สามารถทราบค่าที่แน่นอนได้

จากการแก้ปัญหาค่าผลเฉลยข้างต้น สามารถนำมาเขียนเป็นขั้นตอนวิธีการแก้ปัญหาของกำหนดการเชิงเส้นเศษส่วนหนึ่งเงื่อนไขสองขั้นตอน เป็นขั้นตอนดังนี้

1. กำหนดจำนวนของค่า n และ m สำหรับสุ่มค่า

$$a_j, c_j, d_j \qquad ; \; j = 1,2,3,\ldots,n$$
$$b_i, g_i, h_i, p_i, q_i \qquad ; \; i = 1,2,3,\ldots,m$$

2. เรียงค่า b_i จากน้อยไปมาก กำหนดให้มีค่าเท่ากับ b_{si} และ เรียงตำแหน่งโดยกำหนดให้เป็นเวกเตอร์ f แล้วใช้ตำแหน่งของเวกเตอร์ f ในการเรียงค่า g_i, h_i, p_i, q_i เป็น $g_{si}, h_{si}, p_{si}, q_{si}$ ตามลำดับ

3. ทำการเช็คเงื่อนไขของสมการข้างต้น เพื่อลดขั้นตอนการทำงาน จึงเริ่มจาก เงื่อนไขโดยกำหนดเป็นการตรวจคำตอบ chk1 และ chk2 ตามลำดับ

$$\left(\frac{c_k}{a_k}+\sum_{i=1}^{p}h_i-\sum_{i=p+1}^{m}g_i\right)-\left(\frac{d_k}{a_k}-\sum_{i=1}^{p}q_i+\sum_{i=p+1}^{m}p_i\right)\left\{\frac{\dfrac{c_k b_p}{a_k}+\sum_{i=1}^{p-1}h_i(b_p-b_i)+\sum_{i=p+1}^{m}g_i(b_i-b_p)}{\dfrac{d_k b_p}{a_k}+\sum_{i=1}^{p-1}q_i(b_p-b_i)+\sum_{i=p+1}^{m}p_i(b_i-b_p)}\right\}\geq 0$$

$$\left(\frac{-c_k}{a_k}-\sum_{i=1}^{p-1}h_i+\sum_{i=p}^{m}g_i\right)-\left(\frac{-d_k}{a_k}-\sum_{i=1}^{p-1}q_i+\sum_{i=p}^{m}p_i\right)\left\{\frac{\dfrac{c_k b_p}{a_k}+\sum_{i=1}^{p-1}h_i(b_p-b_i)+\sum_{i=p+1}^{m}g_i(b_i-b_p)}{\dfrac{d_k b_p}{a_k}+\sum_{i=1}^{p-1}q_i(b_p-b_i)+\sum_{i=p+1}^{m}p_i(b_i-b_p)}\right\}\geq 0$$

ทำการแทนค่า k ($k = 1,2,3,\ldots,n$) และ หาค่า p ($p = 1,2,3,\ldots,m$) ที่ทำให้สมการเป็นจริง ถ้าสามารถหาค่าคู่ k และ p ที่ทำให้สมการเป็นจริงได้จึงจะทำการตรวจสอบเงื่อนไขถัดไป แต่ถ้าไม่สามารถหาคู่ระหว่างค่า k และ p ใด ๆ ได้ แสดงว่าไม่สามารถเลือกตัวแปร x ใด ๆ ได้ แสดงว่าไม่มีคำตอบที่ทำให้ระบบสมการเป็นจริง โปรแกรมจะออกจากการคำนวณ

4. เมื่อ chk1 และ chk2 มีคำตอบ จึงทำการ ตรวจคำตอบโดยตรวจสอบเงื่อนไข chk3

$$\sum_{\substack{j=1\\ j\neq k}}^{n}\left[\left(c_j-\frac{a_jc_k}{a_k}\right)-\left(d_j-\frac{a_jd_k}{a_k}\right)\left\{\frac{\dfrac{c_kb_p}{a_k}+\sum_{i=1}^{p-1}h_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}g_i\left(b_i-b_p\right)}{\dfrac{d_kb_p}{a_k}+\sum_{i=1}^{p-1}q_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}p_i\left(b_i-b_p\right)}\right\}\right]\geq 0$$

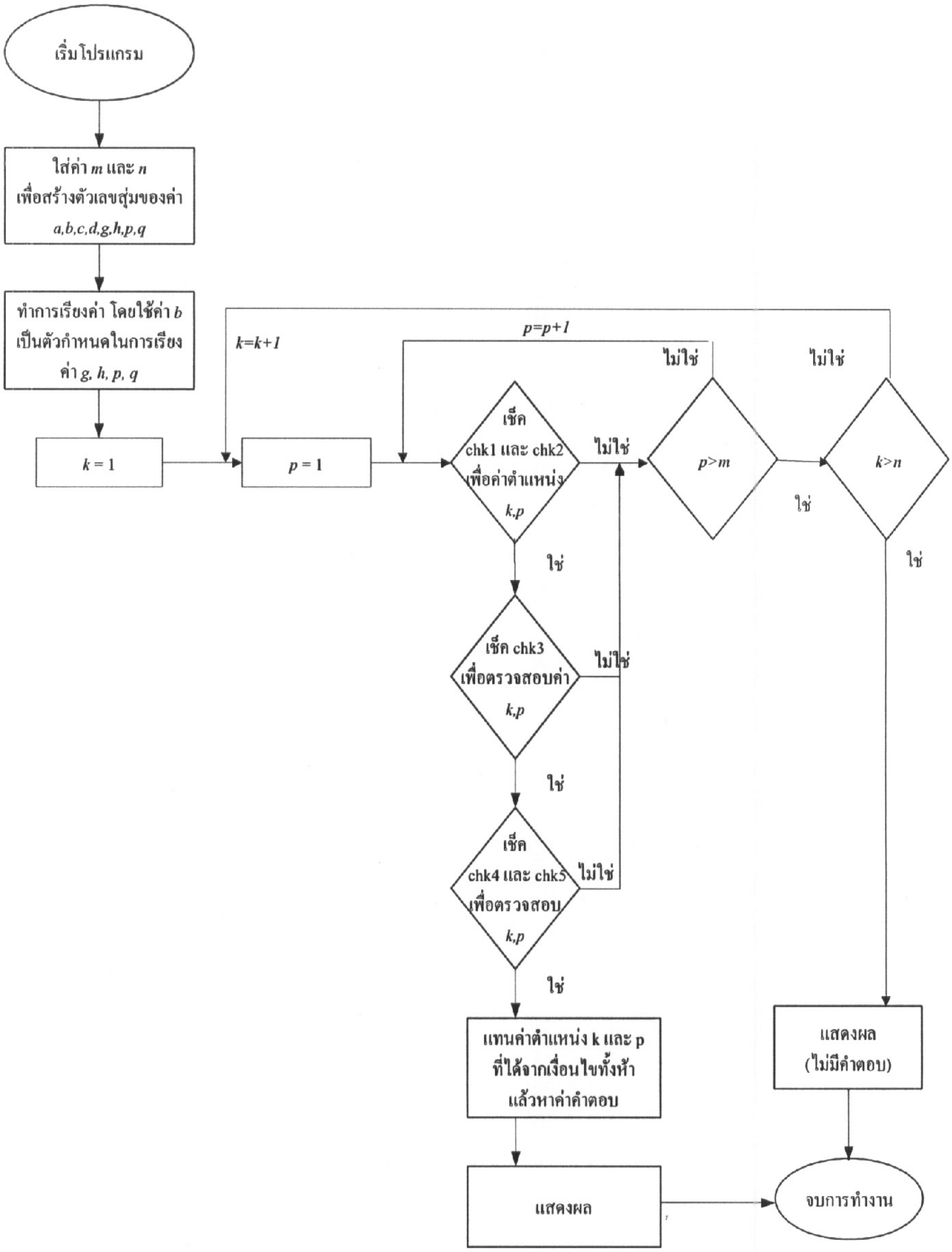
ทำการแทนค่าผลลัพธ์ k และค่า p ที่ได้จากสมการเงื่อนไขก่อนหน้า และ ทำการตรวจสอบเงื่อนไขที่ค่า k และ p คู่ดังกล่าวสามารถทำให้สมการเงื่อนไขข้างบนเป็นจริงหรือไม่ เมื่อแทนค่าแล้วทำให้เงื่อนไข chk3 เป็นจริงให้ทำการทดสอบในข้อถัดไป แต่ถ้าไม่ทำให้สมการเงื่อนไขเป็นจริง แสดงว่าไม่สามารถเลือกตัวแปร x ใด ๆ ได้ แสดงว่าไม่มีคำตอบที่ทำให้ระบบสมการเป็นจริง โปรแกรมจะออกจากการคำนวณ

5. เมื่อ chk1 chk2 และ chk3 แล้วยังคงมีคำตอบ ให้ทำการตรวจคำตอบโดยตรวจสอบเงื่อนไข chk4 และ chk5 ดังสมการเงื่อนไขด้านล่าง ตามลำดับ

$$\sum_{i=1}^{p-1}\left[\left(g_i+h_i\right)-\left(p_i+q_i\right)\left\{\frac{\dfrac{c_kb_p}{a_k}+\sum_{i=1}^{p-1}h_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}g_i\left(b_i-b_p\right)}{\dfrac{d_kb_p}{a_k}+\sum_{i=1}^{p-1}q_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}p_i\left(b_i-b_p\right)}\right\}\right]\geq 0$$

$$\sum_{i=p+1}^{m}\left[\left(g_i+h_i\right)-\left(p_i+q_i\right)\left\{\frac{\dfrac{c_kb_p}{a_k}+\sum_{i=1}^{p-1}h_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}g_i\left(b_i-b_p\right)}{\dfrac{d_kb_p}{a_k}+\sum_{i=1}^{p-1}q_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}p_i\left(b_i-b_p\right)}\right\}\right]\geq 0$$

ทำการตรวจสอบเงื่อนไขที่ค่า k และ p คู่คำตอบที่ได้จากเงื่อนไขก่อนหน้า ตรวจสอบเงื่อนไขโดยแทนค่าคำตอบแต่ละคู่ ถ้าไม่สามารถทำให้สมการเงื่อนไขข้างบนเป็นจริง แสดงว่าไม่มีคำตอบที่ทำให้ระบบสมการเป็นจริง โปรแกรมจะออกจากการคำนวณ แต่ถ้าเมื่อแทนค่าแล้วทำให้เงื่อนไขข้างบนเป็นจริงแล้ว โปรแกรมจะแทนค่า กลับเพื่อหาค่าวัตถุประสงค์ที่น้อยที่สุด และแสดงผล โดยสามารถแสดงเป็นแผนภาพดังนี้



ภาพที่ 1 แผนภาพขั้นตอนในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบวิธีตรง

3. <u>กำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต (Two-Stage Single Constrained Linear Fractional Programming With Bounded Variables) โดยวิธีตรง</u>

เมื่อตัวแปรของปัญหากำหนดการเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต สามารถแสดงได้โดย

$$\text{Min } Z = \frac{\sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m}\left(g_i u_i + h_i v_i\right)}{\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m}\left(p_i u_i + q_i v_i\right)} \qquad (7)$$

$$\text{s.t.} \quad \sum_{j=1}^{n} a_j x_j + u_i - v_i = b_i \qquad (8)$$

$$0 \le x_j \le t_j \qquad ; \ j = 1,2,3,\ldots,n$$

$$c_j, g_i, h_i \ge 0 \qquad ; \ i = 1,2,3,\ldots,m$$

$$d_j, p_i, q_i \ge 0$$

$$u_i, v_i \ge 0$$

สำหรับค่าฟังก์ชันวัตถุประสงค์ สมการต้องเป็น $\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m}\left(p_i u_i + q_i v_i\right) \ge 0$ และกำหนดให้

$$y_0 = \frac{1}{\sum_{j=1}^{n} d_j x_j + \sum_{i=1}^{m}\left(p_i u_i + q_i v_i\right)} \qquad ; \ y_0 \ge 0$$

$$y_j = x_j y_0 \qquad ; \ y_j \ge 0$$

$$r_i = u_i y_0 \qquad ; \ r_i \ge 0$$

$$s_i = v_i y_0 \qquad ; \ s_i \ge 0$$

ดังนั้น $\text{Min } Z = \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m}\left(g_i r_i + h_i s_i\right)$ (9)

จาก (8) นำ y_0 คูณตลอดจะได้

$$\sum_{j=1}^{n} a_j y_j + r_i - s_i = b_i y_0 \tag{10}$$

เพิ่มสมการเงื่อนไขเพื่อทำให้ระบบสมการสมบูรณ์

$$\sum_{j=1}^{n} d_j y_j + \sum_{i=1}^{m} \left(p_i r_i + q_i s_i\right) = 1 \tag{11}$$

$$0 \le y_j \le t_j y_0$$

$$c_j, g_i, h_i \ge 0$$

$$d_j, p_i, q_i \ge 0$$

$$r_i, s_i \ge 0$$

จากเงื่อนไขของคำตอบ

$y_j \le t_j y_o$ และ $y_j \ge 0$; $j = 1,2,3,\ldots,n$

การหาคำตอบปัญหามีความซับซ้อนมากไม่สามารถเรียงค่า j ที่แน่นอนได้ในทุกค่าของ k และ p จึงต้องทำการหาค่าคำตอบทุกค่าที่เป็นไปได้จะได้ โดยขั้นแรก หาค่า k และ p ที่เป็นคำตอบจะเป็นค่าน้อยที่สุดที่อยู่ภายใต้เงื่อนไขของคำตอบด้านบน ดังนั้นจะได้คำตอบคือ

ขั้นที่ 1

$$y_k = \frac{b_p y_o}{a_k} \qquad ;\left(k \in j \mid k = 1,2,3,\ldots,n\right)$$

ขั้นที่ 2

$$y_1 = t_1 y_o$$

$$y_k = \frac{b_p y_0 - a_1 t_1 y_0}{a_k}$$

$$y_{k+1} = 0$$

$$\vdots$$

$$y_n = 0$$

ขั้นที่ 3

$$y_1 = t_1 y_o$$

$$y_2 = t_2 y_o$$

$$y_k = \frac{b_p y_0 - \sum_{j=1}^{2} a_j t_j y_0}{a_k}$$

$$y_{k+1} = 0$$

$$\vdots$$

$$y_n = 0$$

ขั้นที่ 4

$$y_1 = t_1 y_o$$

$$y_2 = t_2 y_o$$

$$y_3 = t_3 y_o$$

$$y_k = \frac{b_p y_0 - \sum_{j=1}^{3} a_j t_j y_0}{a_k}$$

$$y_{k+1} = 0$$

$$\vdots$$

$$y_n = 0$$

ขั้นที่ 5

$$y_1 = t_1 y_o$$

$$y_2 = t_2 y_o$$

$$y_3 = t_3 y_o$$

$$y_4 = t_4 y_o$$

$$y_k = \frac{b_p y_0 - \sum_{j=1}^{n} a_j t_j y_0}{a_k}$$

$$y_{k+1} = 0$$

$$\vdots$$

$$y_n = 0$$

ขั้นทั่วไป

$$y_1 = t_1 y_o$$

$$y_2 = t_2 y_o$$
$$\vdots$$
$$\vdots$$
$$y_{k-1} = t_{k-1} y_o$$
$$y_k = \frac{b_p y_0 - \sum_{j=1}^{n} a_j t_j y_0}{a_k}$$
$$y_{k+1} = 0$$
$$\vdots$$
$$\vdots$$
$$y_n = 0$$

เมื่อ k เป็นตำแหน่งที่ $\dfrac{b_p y_0 - \sum_{j=1}^{k-1} a_j t_j y_0}{a_k} \le t_k y_0$ และ p เป็นตำแหน่งที่

$$\left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i\right)$$

$$-\left(\frac{d_k}{a_k} - \sum_{i=1}^{p} q_i + \sum_{i=p+1}^{m} p_i\right)\left\{\frac{\sum_{j=1}^{k-1} c_j t_j + c_k\left(\dfrac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k}\right) + \sum_{i=1}^{p-1} h_i\left(b_p - b_i\right) + \sum_{i=p+1}^{m} g_i\left(b_i - b_p\right)}{\sum_{j=1}^{k-1} d_j t_j + d_k\left(\dfrac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k}\right) + \sum_{i=1}^{p-1} q_i\left(b_p - b_i\right) + \sum_{i=p+1}^{m} p_i\left(b_i - b_p\right)}\right\}$$

และ

$$\left(\frac{-c_k}{a_k}-\sum_{i=1}^{p-1}h_i+\sum_{i=p}^{m}g_i\right)$$

$$-\left(\frac{-d_k}{a_k}-\sum_{i=1}^{p-1}q_i+\sum_{i=p}^{m}p_i\right)\left\{\frac{\sum_{j=1}^{k-1}c_jt_j+c_k\left(\frac{b_p-\sum_{j=1}^{k-1}a_jt_j}{a_k}\right)+\sum_{i=1}^{p-1}h_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}g_i\left(b_i-b_p\right)}{\sum_{j=1}^{k-1}d_jt_j+d_k\left(\frac{b_p-\sum_{j=1}^{k-1}a_jt_j}{a_k}\right)+\sum_{i=1}^{p-1}q_i\left(b_p-b_i\right)+\sum_{i=p+1}^{m}p_i\left(b_i-b_p\right)}\right\}$$

ต้องมีค่ามากกว่าหรือเท่ากับศูนย์

$i \neq p$; $r_i = 0$ และ $s_i = b_p y_0 - b_i y_0$ $\left(i = 1,2,\ldots, p-1\right)$

$r_i = b_i y_0 - b_p y_0$ และ $s_i = 0$ $\left(i = p+1, p+2,\ldots, m\right)$

$i = p$; $r_i = 0$ และ $s_i = 0$

ดังนั้นจะสามารถจัดรูปแบบของตัวแปรให้อยู่ในรูปของ k และ p ดังนี้

เมื่อ $y_j = t_j y_o - w_j$; $w_j \geq 0$ (j = 1, 2, 3, …, k-1)

จาก (10) จัดเทอม y_k

$$y_k = \left(b_p y_0 - r_p + s_p - \sum_{j=1}^{k-1}a_j\left(t_j y_0 - w_j\right) - \sum_{j=k+1}^{n}a_j y_j\right)/a_k \qquad (12)$$

แทนค่า y_k กลับในสมการ

$$\sum_{j=1}^{k-1}a_j\left(t_j y_0 - w_j\right) + a_k\left(b_p - r_p + s_p - \sum_{j=1}^{k-1}a_j\left(t_j y_0 - w_j\right) - \sum_{j=k+1}^{n}a_j y_j\right)/a_k$$

$$+\sum_{j=k+1}^{n}a_j y_j + r_i - s_i = b_i y_0$$

$i \neq p$;

$s_i = 0$ และ $r_i = b_i y_0 - b_p y_0 + s_i + r_p - s_p$; $b_i y_0 > b_p y_0$ (13)

$r_i = 0$ และ $s_i = b_p y_0 - b_i y_0 + r_i - r_p + s_p$; $b_i y_0 < b_p y_0$ (14)

$$i = p\,; \qquad r_p - s_p = 0 \quad \rightarrow \quad r_p, s_p = 0 \tag{15}$$

สมมติให้

$$b_i \le b_{i+1} \qquad ; \ (i = 1,2,\dots,m-1)$$

จาก (11)

$$\sum_{j=1}^{k-1} d_j y_j + d_k y_k + \sum_{j=k+1}^{n} d_j y_j + \sum_{i=1}^{p-1}(p_i r_i + q_i s_i) + \sum_{i=p+1}^{m}(p_i r_i + q_i s_i) + p_p r_p + q_p s_p = 1$$

แทนค่าเทอม y_k, s_i และ r_i จะได้

$$\sum_{j=1}^{k-1} d_j (t_j y_0 - w_j) + \frac{d_k}{a_k}\left(b_p y_0 - r_p + s_p - \sum_{j=1}^{k-1} a_j (t_j y_0 - w_j) - \sum_{j=k+1}^{n} a_j y_j \right) + \sum_{j=k+1}^{n} d_j y_j$$
$$+ \sum_{i=1}^{p-1}\left[p_i r_i + q_i \left((b_p - b_i) y_0 + r_i - r_p + s_p \right) \right] + p_p r_p + q_p s_p$$
$$+ \sum_{i=p+1}^{m}\left[p_i \left((b_i - b_p) y_0 + s_i + r_p - s_p \right) + q_i s_i \right] = 1$$

$$\sum_{j=1}^{k-1} d_j (t_j y_0 - w_j) + d_k \left(\frac{b_p y_0 - \sum_{j=1}^{k-1} a_j (t_j y_0 - w_j)}{a_k} \right) + \sum_{j=k+1}^{n} \left(d_j - \frac{a_j d_k}{a_k} \right) y_j$$
$$+ \sum_{i=1}^{p-1} q_i (b_p y_0 - b_i y_0) + \sum_{i=p+1}^{m} p_i (b_i y_0 - b_p y_0) + \sum_{i=1}^{p-1} (p_i + q_i) r_i$$
$$+ \sum_{i=p+1}^{m} (p_i + q_i) s_i + \left(\frac{-d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i \right) r_p + \left(\frac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i \right) s_p = 1$$

กำหนดให้

$$z_0 = \sum_{j=1}^{k-1} \left(\frac{a_j d_k}{a_k} - d_j \right) w_j + \sum_{j=k+1}^{n} \left(d_j - \frac{a_j d_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} (p_i + q_i) r_i + \sum_{i=p+1}^{m} (p_i + q_i) s_i$$
$$+ \left(\frac{-d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i \right) r_p + \left(\frac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i \right) s_p$$

$$y_0 = \left[(1-z_0) / \left(\sum_{j=1}^{k-1} d_j t_j + d_k \left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \right) + \sum_{i=1}^{p-1} q_i (b_p - b_i) + \sum_{i=p+1}^{m} p_i (b_i - b_p) \right) \right] \tag{16}$$

แทนค่าในสมการวัตถุประสงค์จะได้

$$\begin{aligned}
\text{Min } Z &= \sum_{j=1}^{k-1} c_j y_j + c_k y_k + \sum_{j=k+1}^{n} c_j y_j + \sum_{i=1}^{p-1} (g_i r_i + h_i s_i) + \sum_{i=p+1}^{m} (g_i r_i + h_i s_i) + g_p r_p + h_p s_p \\
&= \sum_{j=1}^{k-1} c_j (t_j y_0 - w_j) + \frac{c_k}{a_k} \left(b_p y_0 - r_p + s_p - \sum_{j=1}^{k-1} a_j (t_j y_0 - w_j) - \sum_{j=k+1}^{n} a_j y_j \right) \\
&\quad + \sum_{j=k+1}^{n} c_j y_j + \sum_{i=1}^{p-1} g_i r_i + \sum_{i=1}^{p-1} h_i (b_p y_0 - b_i y_0 + r_i - r_p + s_p) \\
&\quad + \sum_{i=p+1}^{m} g_i (b_i y_0 - b_p y_0 + s_i + r_p - s_p) + \sum_{i=p+1}^{m} h_i s_i + g_p r_p + h_p s_p \\
&= \sum_{j=1}^{k-1} c_j (t_j y_0 - w_j) + c_k \left(\frac{b_p y_0 - \sum_{j=1}^{k-1} a_j (t_j y_0 - w_j)}{a_k} \right) + \sum_{j=k+1}^{n} \left(c_j - \frac{a_j c_k}{a_k} \right) y_j \\
&\quad + \sum_{i=1}^{p-1} h_i (b_p y_0 - b_i y_0) + \sum_{i=p+1}^{m} g_i (b_i y_0 - b_p y_0) + \sum_{i=1}^{p-1} (g_i + h_i) r_i + \sum_{i=p+1}^{m} (g_i + h_i) s_i \\
&\quad + \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i \right) r_p + \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i \right) s_p
\end{aligned}$$

จัดรูปให้อยู่ในรูปเทอม y_0

$$\begin{aligned}
&= \left[\sum_{j=1}^{k-1} c_j t_j + c_k \left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \right) + \sum_{i=1}^{p-1} h_i (b_p - b_i) + \sum_{i=p+1}^{m} g_i (b_i - b_p) \right] y_0 \\
&\quad + \sum_{j=1}^{k-1} \left(\frac{a_j c_k}{a_k} - c_j \right) w_j + \sum_{j=k+1}^{n} \left(c_j - \frac{a_j c_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} (g_i + h_i) r_i + \sum_{i=p+1}^{m} (g_i + h_i) s_i \\
&\quad + \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i \right) r_p + \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i \right) s_p
\end{aligned}$$

นำ y_0 จาก (16) มาแทนค่า

$$
= \left[\left\{ \sum_{j=1}^{k-1} c_j t_j + c_k \left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \right) + \sum_{i=1}^{p-1} h_i (b_p - b_i) + \sum_{i=p+1}^{m} g_i (b_i - b_p) \right\} \times \left\{ \frac{1 - z_0}{\sum_{j=1}^{k-1} d_j t_j + d_k \left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \right) + \sum_{i=1}^{p-1} q_i (b_p - b_i) + \sum_{i=p+1}^{m} p_i (b_i - b_p)} \right\} \right]
$$

$$
+ \sum_{j=1}^{k-1} \left(\frac{a_j c_k}{a_k} - c_j \right) w_j + \sum_{j=k+1}^{n} \left(c_j - \frac{a_j c_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} (g_i + h_i) r_i + \sum_{i=p+1}^{m} (g_i + h_i) s_i
$$

$$
+ \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i \right) r_p + \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i \right) s_p
$$

จะได้

$$
= \left[\left\{ \frac{\sum_{j=1}^{k-1} c_j t_j + c_k \left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \right) + \sum_{i=1}^{p-1} h_i (b_p - b_i) + \sum_{i=p+1}^{m} g_i (b_i - b_p)}{\sum_{j=1}^{k-1} d_j t_j + d_k \left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \right) + \sum_{i=1}^{p-1} q_i (b_p - b_i) + \sum_{i=p+1}^{m} p_i (b_i - b_p)} \right\} \times \left\{ 1 - \left(\begin{array}{l} \sum_{j=1}^{k-1} \left(\frac{a_j d_k}{a_k} - d_j \right) w_j + \sum_{j=k+1}^{n} \left(d_j - \frac{a_j d_k}{a_k} \right) y_j + \sum_{i=1}^{p-1} (p_i + q_i) r_i \\ + \sum_{i=p+1}^{m} (p_i + q_i) s_i + \left(\frac{-d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i \right) r_p \\ + \left(\frac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i \right) s_p \end{array} \right) \right\} \right]
$$

$$+\sum_{j=1}^{k-1}\left(\frac{a_j c_k}{a_k}-c_j\right)w_j+\sum_{j=k+1}^{n}\left(c_j-\frac{a_j c_k}{a_k}\right)y_j+\sum_{i=1}^{p-1}(g_i+h_i)r_i$$

$$+\sum_{i=p+1}^{m}(g_i+h_i)s_i+\left(\frac{-c_k}{a_k}-\sum_{i=1}^{p-1}h_i+\sum_{i=p}^{m}g_i\right)r_p+\left(\frac{c_k}{a_k}+\sum_{i=1}^{p}h_i-\sum_{i=p+1}^{m}g_i\right)s_p$$

กำหนดให้

$$\mathrm{z}=\left\{\frac{\sum_{j=1}^{k-1}c_j t_j+c_k\left(\frac{b_p-\sum_{j=1}^{k-1}a_j t_j}{a_k}\right)+\sum_{i=1}^{p-1}h_i(b_p-b_i)+\sum_{i=p+1}^{m}g_i(b_i-b_p)}{\sum_{j=1}^{k-1}d_j t_j+d_k\left(\frac{b_p-\sum_{j=1}^{k-1}a_j t_j}{a_k}\right)+\sum_{i=1}^{p-1}q_i(b_p-b_i)+\sum_{i=p+1}^{m}p_i(b_i-b_p)}\right\}$$

แทน i เป็น f $(f=1,2,3,\ldots,m)$ และ j เป็น e $(e=1,2,3,\ldots,n)$ ดังนั้น

$$\mathrm{z}=\left\{\frac{\sum_{e=1}^{k-1}c_e t_e+c_k\left(\frac{b_p-\sum_{e=1}^{k-1}a_e t_e}{a_k}\right)+\sum_{f=1}^{p-1}h_f(b_p-b_f)+\sum_{f=p+1}^{m}g_f(b_f-b_p)}{\sum_{e=1}^{k-1}d_e t_e+d_k\left(\frac{b_p-\sum_{e=1}^{k-1}a_e t_e}{a_k}\right)+\sum_{f=1}^{p-1}q_f(b_p-b_f)+\sum_{f=p+1}^{m}p_f(b_f-b_p)}\right\}$$

จะได้สมการเป็น

$$\text{Min } Z = z+\sum_{j=k+1}^{n}w_j A_j+\sum_{j=k+1}^{n}y_j B_j+\sum_{i=1}^{p-1}r_i U_i+\sum_{i=p+1}^{m}s_i V_i+r_p U_p+s_p V_p$$

เมื่อกำหนดให้

$$A_j=\left(\frac{a_j c_k}{a_k}-c_j\right)-\left(\frac{a_j d_k}{a_k}-d_j\right)z \qquad ;\ j=k+1,k+2,\ldots,n$$

$$B_j=\left(c_j-\frac{a_j c_k}{a_k}\right)-\left(d_j-\frac{a_j d_k}{a_k}\right)z \qquad ;\ j=k+1,k+2,\ldots,n$$

$$U_i = (g_i + h_i) - (p_i + q_i)z \qquad ;\ i = 1,2,\ldots, p-1$$
$$V_i = (g_i + h_i) - (p_i + q_i)z \qquad ;\ i = p+1, p+2,\ldots, m$$
$$U_p = \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i\right) - \left(\frac{-d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i\right)z$$
$$V_p = \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i\right) - \left(\frac{d_k}{a_k} - \sum_{i=1}^{p} q_i + \sum_{i=p+1}^{m} p_i\right)z$$

จากสมการวัตถุประสงค์สามารถสรุปได้ว่า สำหรับข้อกำหนด $a_j, c_j, d_j, w_j \geq 0$ แล้ว

$$\sum_{j=1}^{k-1}\left[\left(\frac{a_j c_k}{a_k} - c_j\right) - \left(\frac{a_j d_k}{a_k} - d_j\right)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f (b_p - b_f) + \sum_{f=p+1}^{m} g_f (b_f - b_p)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f (b_p - b_f) + \sum_{f=p+1}^{m} p_f (b_f - b_p)}\right\}\right]$$

มากกว่าหรือเท่ากับศูนย์แล้ว เพื่อค่าวัตถุประสงค์ที่น้อยที่สุด จึงให้ $w_j = 0 \ (j = 1,2,3,\ldots, k-1)$

สำหรับข้อกำหนด $a_j, c_j, d_j, y_j \geq 0$ แล้ว

$$\sum_{j=k+1}^{n}\left[\left(c_j - \frac{a_j c_k}{a_k}\right) - \left(d_j - \frac{a_j d_k}{a_k}\right)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f (b_p - b_f) + \sum_{f=p+1}^{m} g_f (b_f - b_p)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f (b_p - b_f) + \sum_{f=p+1}^{m} p_f (b_f - b_p)}\right\}\right]$$

มากกว่าหรือเท่ากับศูนย์แล้ว เพื่อค่าวัตถุประสงค์ที่น้อยที่สุด จึงให้ $y_j = 0$ หรือ $t_j y_0 = 0$ $(j = k+1, k+2, ..., n)$

สำหรับข้อกำหนด $g_i, h_i, r_i, s_i \geq 0$ แล้ว

$$\sum_{i=1}^{p-1}\left[(g_i + h_i) - (p_i + q_i)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f (b_p - b_f) + \sum_{f=p+1}^{m} g_f (b_f - b_p)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f (b_p - b_f) + \sum_{f=p+1}^{m} p_f (b_f - b_p)}\right\}\right] \geq 0$$

$$\sum_{i=p+1}^{m}\left[(g_i + h_i) - (p_i + q_i)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f (b_p - b_f) + \sum_{f=p+1}^{m} g_f (b_f - b_p)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f (b_p - b_f) + \sum_{f=p+1}^{m} p_f (b_f - b_p)}\right\}\right] \geq 0$$

เพื่อค่าวัตถุประสงค์ที่น้อยที่สุด จึงให้ $r_i = 0$ $(i = 1,2,..., p-1)$ และ $s_i = 0$ $(i = p+1, p+2,..., m)$ เช่นเดียวกันกับ

$$\left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i\right)$$

$$-\left(\frac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i\right)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f (b_p - b_f) + \sum_{f=p+1}^{m} g_f (b_f - b_p)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f (b_p - b_f) + \sum_{f=p+1}^{m} p_f (b_f - b_p)}\right\}$$

มากกว่าหรือเท่ากับศูนย์ และ

$$\left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i\right) - \left(\frac{-d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i\right)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f\left(b_p - b_f\right) + \sum_{f=p+1}^{m} g_f\left(b_f - b_p\right)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f\left(b_p - b_f\right) + \sum_{f=p+1}^{m} p_f\left(b_f - b_p\right)}\right\}$$

มากกว่าหรือเท่ากับศูนย์ เพื่อค่าวัตถุประสงค์ที่น้อยที่สุด จึงให้ $r_p, s_p = 0$

เมื่อ $i = p$ และ $j = k$

$$y_k = \left(b_p y_0 - r_p + s_p - \sum_{j=1}^{k-1} a_j t_j y_0 - \sum_{j=k+1}^{n} a_j t_j y_0\right) / a_k \quad \text{จะได้} \quad y_k = \frac{b_p y_0 - \sum_{j=1}^{k-1} a_j t_j y_0}{a_k}$$

$$s_i = \left(b_p - b_i\right) y_0 + r_i - r_p + s_p \quad \text{จะได้} \quad s_i = \left(b_p - b_i\right) y_0$$

$$r_i = \left(b_i - b_p\right) y_0 + s_i + r_p - s_p \quad \text{จะได้} \quad r_i = \left(b_i - b_p\right) y_0$$

จะได้คำตอบคือ

$$\text{Min } Z = \left\{\frac{\sum_{j=1}^{k-1} c_j t_j + c_k\left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k}\right) + \sum_{i=1}^{p-1} h_i\left(b_p - b_i\right) + \sum_{i=p+1}^{m} g_i\left(b_i - b_p\right)}{\sum_{j=1}^{k-1} d_j t_j + d_k\left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k}\right) + \sum_{i=1}^{p-1} q_i\left(b_p - b_i\right) + \sum_{i=p+1}^{m} p_i\left(b_i - b_p\right)}\right\}$$

$$y_k = \frac{b_p y_0 - \sum_{j=1}^{k-1} a_j t_j y_0}{a_k} \le t_k y_0 \quad \rightarrow \quad x_k = \frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \le t_k$$

$$s_i = (b_p - b_i)y_0 \rightarrow v_i = b_p - b_i$$
$$r_i = (b_i - b_p)y_0 \rightarrow u_i = b_i - b_p$$
$$r_p = 0 \text{ and } s_p = 0 \rightarrow u_p = 0 \text{ and } v_p = 0$$
$$y_0 = \frac{1}{\sum_{j=1}^{k-1} d_j t_j + d_k \left(\frac{b_p - \sum_{j=1}^{k-1} a_j t_j}{a_k} \right) + \sum_{i=1}^{p-1} q_i (b_p - b_i) + \sum_{i=p+1}^{m} p_i (b_i - b_p)}$$

จากการแก้ปัญหาหาค่าผลเฉลยข้างต้นคำตอบส่วนใหญ่ในการทดลองสุ่มและสร้างปัญหาขึ้นมา มักจะเกิดคำตอบของค่า x_j ไม่เกิน5 คำตอบในแต่ละปัญหา รวมทั้งในการหาค่าคำตอบทั้งหมดระดับ 5 ระดับ ต้องใช้ หน่วยความจำที่ค่อนข้างมาก เมื่อเจอปัญหาที่ใหญ่ขึ้นก็ไม่สามารถหาคำตอบได้เนื่องจากหน่วยความจำไม่เพียงพอ จึงทำการตัดสินใจสร้างขั้นตอนในการแก้ปัญหาที่ 5 ระดับ ดังนั้นจะสามารถนำมาเขียนเป็นขั้นตอนวิธีการแก้ปัญหาของกำหนดการเชิงเส้นเศษส่วนหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขตโดยทำการคำนวณหาค่าคำตอบของปัญหา เป็น 5 ระดับ ได้เป็นขั้นตอนดังนี้

1. กำหนดจำนวนของค่า n และ m สำหรับสุ่มค่า

$$a_j, c_j, d_j, t_j \quad ; \; j = 1,2,3,\ldots,n$$
$$b_i, g_i, h_i, p_i, q_i \quad ; \; i = 1,2,3,\ldots,m$$

2. เรียงค่า b_i จากน้อยไปมาก กำหนดให้มีค่าเท่ากับ $b_{s\,i}$ และ เรียงตำแหน่งโดยกำหนดให้เป็นเวกเตอร์ f แล้วใช้ตำแหน่งของเวกเตอร์ f ในการเรียงค่า g_i, h_i, p_i, q_i เป็น $g_{s\,i}, h_{s\,i}, p_{s\,i}, q_{s\,i}$ ตามลำดับ และ $a_j, c_j, d_j, t_j = a_{s\,j}, c_{s\,j}, d_{s\,j}, t_{s\,j}$ ตามลำดับ

3. ทำการเช็คค่า x_j โดยหา $testx(j) = b_s(1)/a_s(j)$ ทุกค่า j แล้วนำค่า $testx(j)$ ที่ไม่เท่ากับศูนย์ไปเช็คเงื่อนไขในข้อถัดไป

4. ทำการเช็คเงื่อนไขของสมการข้างต้น เพื่อลดขั้นตอนการทำงาน จึงเริ่มจาก เงื่อนไขโดยกำหนดเป็นการตรวจคำตอบ chk1 และ chk2 ตามลำดับ

$$\text{chk1} = \left(\frac{c_k}{a_k} + \sum_{i=1}^{p} h_i - \sum_{i=p+1}^{m} g_i\right)$$

$$-\left(\frac{d_k}{a_k} + \sum_{i=1}^{p} q_i - \sum_{i=p+1}^{m} p_i\right)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f\left(b_p - b_f\right) + \sum_{f=p+1}^{m} g_f\left(b_f - b_p\right)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f\left(b_p - b_f\right) + \sum_{f=p+1}^{m} p_f\left(b_f - b_p\right)}\right\}$$

$$\geq 0$$

$$\text{chk2} = \left(\frac{-c_k}{a_k} - \sum_{i=1}^{p-1} h_i + \sum_{i=p}^{m} g_i\right)$$

$$-\left(\frac{-d_k}{a_k} - \sum_{i=1}^{p-1} q_i + \sum_{i=p}^{m} p_i\right)\left\{\frac{\sum_{e=1}^{k-1} c_e t_e + c_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} h_f\left(b_p - b_f\right) + \sum_{f=p+1}^{m} g_f\left(b_f - b_p\right)}{\sum_{e=1}^{k-1} d_e t_e + d_k\left(\frac{b_p - \sum_{e=1}^{k-1} a_e t_e}{a_k}\right) + \sum_{f=1}^{p-1} q_f\left(b_p - b_f\right) + \sum_{f=p+1}^{m} p_f\left(b_f - b_p\right)}\right\}$$

$$\geq 0$$

ทำการแทนค่า k ($k = 1,2,3,\ldots,n$) และ หาค่า p ($p = 1,2,3,\ldots,m$) ที่ทำให้สมการเป็นจริง ถ้าสามารถหาค่าคู่ k และ p ที่ทำให้สมการเป็นจริงได้จึงจะทำการตรวจสอบเงื่อนไขถัดไป ถ้าไม่สามารถหาค่า k และ p ใด ๆ ได้ แสดงว่าไม่สามารถเลือกตัวแปร x ใด ๆ ได้ แสดงว่าไม่มีคำตอบที่ทำให้ระบบสมการเป็นจริง โปรแกรมจะออกจากการคำนวณในระดับที่ 1 (ค่า x มีเพียงคำตอบเดียว) แล้วทำการเช็คคำตอบในระดับที่สอง (ค่า x มี 2 คำตอบ) ตามขั้นตอนในข้อที่ 8

5. เมื่อ chk1 และ chk2 มีคำตอบ จึงทำการ ตรวจคำตอบโดยตรวจสอบเงื่อนไข chk3 และ chk4 ตามลำดับ

$$\sum_{j=1}^{k-1}\left[\begin{array}{l}\left(\dfrac{a_j c_k}{a_k}-c_j\right)\\ -\left(\dfrac{a_j d_k}{a_k}-d_j\right)\left\{\dfrac{\sum_{e=1}^{k-1}c_e t_e+c_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_e t_e}{a_k}\right)+\sum_{f=1}^{p-1}h_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}g_f\left(b_f-b_p\right)}{\sum_{e=1}^{k-1}d_e t_e+d_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_e t_e}{a_k}\right)+\sum_{f=1}^{p-1}q_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}p_f\left(b_f-b_p\right)}\right\}\end{array}\right]$$

$$\sum_{j=k+1}^{n}\left[\begin{array}{l}\left(c_j-\dfrac{a_j c_k}{a_k}\right)\\ -\left(d_j-\dfrac{a_j d_k}{a_k}\right)\left\{\dfrac{\sum_{e=1}^{k-1}c_e t_e+c_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_e t_e}{a_k}\right)+\sum_{f=1}^{p-1}h_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}g_f\left(b_f-b_p\right)}{\sum_{e=1}^{k-1}d_e t_e+d_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_e t_e}{a_k}\right)+\sum_{f=1}^{p-1}q_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}p_f\left(b_f-b_p\right)}\right\}\end{array}\right]$$

ทำการแทนค่าผลลัพธ์ k และค่า p ที่ได้จากสมการเงื่อนไขก่อนหน้า และทำการตรวจสอบเงื่อนไขที่ค่า k และ p คู่ดังกล่าวสามารถทำให้สมการเงื่อนไขข้างบนมีค่ามากกว่าหรือเท่ากับศูนย์ทั้งสองสมการ เมื่อแทนค่าแล้วทำให้เงื่อนไขเป็นจริงให้ทำการทดสอบในข้อถัดไป แต่ถ้าไม่ทำให้สมการเงื่อนไขเป็นจริง แสดงว่าไม่สามารถเลือกตัวแปร x ใด ๆ ได้ แสดงว่าไม่มีคำตอบที่ทำให้ระบบสมการเป็นจริง โปรแกรมจะออกจากการคำนวณในระดับที่ 1 แล้วทำการเช็คคำตอบในระดับที่ 2

6. เมื่อ chk1 chk2 และ chk3 chk4 มีคำตอบ ให้ทำการตรวจคำตอบโดยตรวจสอบเงื่อนไข chk5 และ chk6 ดังสมการเงื่อนไขด้านล่าง ตามลำดับ

$$\sum_{i=1}^{p-1}\left[\begin{array}{l}(g_i+h_i)\\ -(p_i+q_i)\left\{\dfrac{\sum_{e=1}^{k-1}c_et_e+c_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_et_e}{a_k}\right)+\sum_{f=1}^{p-1}h_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}g_f\left(b_f-b_p\right)}{\sum_{e=1}^{k-1}d_et_e+d_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_et_e}{a_k}\right)+\sum_{f=1}^{p-1}q_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}p_f\left(b_f-b_p\right)}\right\}\end{array}\right]\geq 0$$

$$\sum_{i=p+1}^{m}\left[\begin{array}{l}(g_i+h_i)\\ -(p_i+q_i)\left\{\dfrac{\sum_{e=1}^{k-1}c_et_e+c_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_et_e}{a_k}\right)+\sum_{f=1}^{p-1}h_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}g_f\left(b_f-b_p\right)}{\sum_{e=1}^{k-1}d_et_e+d_k\left(\dfrac{b_p-\sum_{e=1}^{k-1}a_et_e}{a_k}\right)+\sum_{f=1}^{p-1}q_f\left(b_p-b_f\right)+\sum_{f=p+1}^{m}p_f\left(b_f-b_p\right)}\right\}\end{array}\right]\geq 0$$

ทำการตรวจสอบเงื่อนไขที่ค่า k และ p คู่คำตอบที่ได้จากเงื่อนไขก่อนหน้า ตรวจสอบเงื่อนไขโดยแทนค่าคำตอบแต่ละคู่ ถ้าไม่สามารถทำให้สมการเงื่อนไขข้างบนเป็นจริง $Obj(k)=0$ แสดงว่าไม่มีคำตอบที่ทำให้ระบบสมการเป็นจริง โปรแกรมจะออกจากการคำนวณในระดับนี้ แล้วทำการเพิ่มจำนวนคำตอบ แล้วทำการทดสอบในขั้นตอนที่คำตอบเพิ่มขึ้นในข้อถัดไป

7. แต่ถ้าแทนค่าแล้วทำให้เงื่อนไขข้างบนเป็นจริงแล้ว โปรแกรมจะแทนค่า กลับเพื่อหาค่าวัตถุประสงค์ที่น้อยที่สุด และแสดงผล ซึ่งจะได้คำตอบ $Obj(k)$, ตำแหน่ง i ของค่า b , และ $x(k)=\dfrac{b_s(i)}{a_s(k)}$ และเก็บค่าคำตอบในระดับนี้ไว้ แล้วทำการทดสอบในขั้นถัดไป

8. เมื่อทดสอบคำตอบในระดับ 1 เสร็จ ทำการเช็คคำตอบในระดับที่ 2 โดยให้ $j=1,2,..,n-1$ และ $k=j+1,j+2,..,n$ แล้วเช็คค่า x_k โดยหา $testx2(j,k)=\dfrac{b_s(m)-t_s(j)a_s(j)}{a_s(k)}$ ทุกค่า j,k แล้วนำค่า $testx2(j,k)$ ที่ไม่เท่ากับศูนย์ไปเช็คเงื่อนไข chk1, chk2, chk3, chk4, chk5 และ chk6 ตามที่เช็คในการแก้ปัญหาระดับที่ 1 แต่ถ้า $testx2(j,k)=0$ ทุกค่าให้ออกจากการคำนวณ

9. ในกรณีที่เช็คเงื่อนไขแล้วเป็นไปตามเงื่อนไข จะได้คำตอบ $Obj2(j,k)$, ตำแหน่ง i ของค่า b, $x(j)=t(j)$ และ $x(k)=\dfrac{b_s(i)-t_s(j)a_s(j)}{a_s(k)}$ แล้วเก็บค่าคำตอบในระดับนี้ แล้วแก้ปัญหาโดยหาคำตอบในระดับที่ 3 (เพิ่มจำนวนคำตอบของ x เป็น 3 คำตอบ)

10. ถ้าเช็คแล้วไม่เป็นไปตามเงื่อนไขจะได้คำตอบในระดับนี้เป็น $Obj2(j,k)=0$, $x(j)=0$ และ $x(k)=0$ ให้ไปแก้ปัญหาโดยหาคำตอบในระดับที่ 3

11. ทำการเช็คคำตอบในระดับที่ 3 โดยให้ $j1=1,2,..,n-2$, $j2=1,2,..,n-1$ และ $k=j2+1, j+2,..,n$ แล้วเช็คค่า x_k โดยหา $testx3(j1,j2,k)$ = $\dfrac{b_s(m)-t_s(j1)a_s(j1)-t_s(j2)a_s(j2)}{a_s(k)}$ ทุกค่า $j1, j2$ และ k แล้วนำค่า $testx3(j1,j2,k)$ ที่ไม่เท่ากับศูนย์ ไปเช็คเงื่อนไข chk1, chk2, chk3, chk4, chk5 และ chk6 ตามที่เช็คในการแก้ปัญหาระดับที่ 1 แต่ถ้า $testx3(j1,j2,k)=0$ ทุกค่า ให้ออกจากการคำนวณ

12. ในกรณีที่เช็คเงื่อนไขแล้วเป็นไปตามเงื่อนไข จะได้คำตอบ $Obj3(j1,j2,k)$, ตำแหน่ง i ของค่า b, $x(j1)=t(j1)$, $x(j2)=t(j2)$ และ $x(k)=$ $\dfrac{b_s(i)-t_s(j1)a_s(j1)-t_s(j2)a_s(j2)}{a_s(k)}$ แล้วเก็บค่าคำตอบในระดับนี้ แล้วแก้ปัญหาโดยหาคำตอบในระดับที่ 4 (คำตอบมีค่า x มี 4 คำตอบ)

13. ถ้าเช็คแล้วไม่เป็นไปตามเงื่อนไขจะได้คำตอบในระดับนี้เป็น $Obj3(j1,j2,k)=0$ ให้ไปแก้ปัญหาโดยหาคำตอบในระดับที่ 4

14. ทำการเช็คคำตอบในระดับที่ 4 โดยให้ $j1=1,2,..,n-3$, $j2=1,2,..,n-2$, $j3=1,2,..,n-1$ และ $k=j3+1, j+2,..,n$ แล้วเช็คค่า x_k โดยหา $testx4(j1,j2,j3,k)=$ $\dfrac{b_s(m)-t_s(j1)a_s(j1)-t_s(j2)a_s(j2)-t_s(j3)a_s(j3)}{a_s(k)}$ ทุกค่า $j1, j2, j3, k$ แล้วนำค่า $testx4(j1,j2,j3,k)$ ที่ไม่เท่ากับศูนย์ไปเช็คเงื่อนไข chk1, chk2, chk3, chk4, chk5 และ chk6 ตามที่เช็คในการแก้ปัญหาระดับที่ 1 แต่ถ้า $testx4(j1,j2,j3,k)=0$ ทุกค่า ให้ออกจากการคำนวณ

15. ในกรณีที่เช็คเงื่อนไขแล้วเป็นไปตามเงื่อนไข จะได้คำตอบ $Obj4(j1, j2, j3, k)$, ตำแหน่ง i ของค่า b, $x(j1) = t(j1)$, $x(j2) = t(j2)$, $x(j3) = t(j3)$ และ $x(k) = \dfrac{b_s(i) - t_s(j1)a_s(j1) - t_s(j2)a_s(j2) - t_s(j3)a_s(j3)}{a_s(k)}$ แล้วเก็บค่าคำตอบในระดับนี้ แล้วแก้ปัญหาโดยหาคำตอบในระดับที่ 5 (คำตอบมีค่า x มี 5 คำตอบ)

16. ถ้าเช็คแล้วไม่เป็นไปตามเงื่อนไขจะได้คำตอบในระดับนี้เป็น $Obj4(j1, j2, j3, k)$ เท่ากับศูนย์ ให้ไปแก้ปัญหาโดยหาคำตอบในระดับที่ 5

17. ทำการเช็คคำตอบในระดับที่ 5 โดยให้ $j1 = 1,2,..,n-4$, $j2 = 1,2,..,n-3$, $j3 = 1,2,..,n-2$, $j4 = 1,2,..,n-1$ และ $k = j4+1, j+2,..,n$ แล้วเช็คค่า x_k โดยหา $testx5(j1, j2, j3, j4, k)$ = $\dfrac{b_s(m) - t_s(j1)a_s(j1) - t_s(j2)a_s(j2) - t_s(j3)a_s(j3) - t_s(j4)a_s(j4)}{a_s(k)}$ ทุกค่า $j1, j2, j3, j4, k$ แล้วนำค่า $testx5(j1, j2, j3, j4, k)$ ที่ไม่เท่ากับศูนย์ ไปเช็คเงื่อนไข chk1, chk2, chk3, chk4, chk5 และ chk6 ตามที่เช็คในการแก้ปัญหาระดับที่ 1 แต่ถ้า $testx5(j1, j2, j3, j4, k)$ เท่ากับศูนย์ทุกค่า ให้ออกจากการคำนวณ

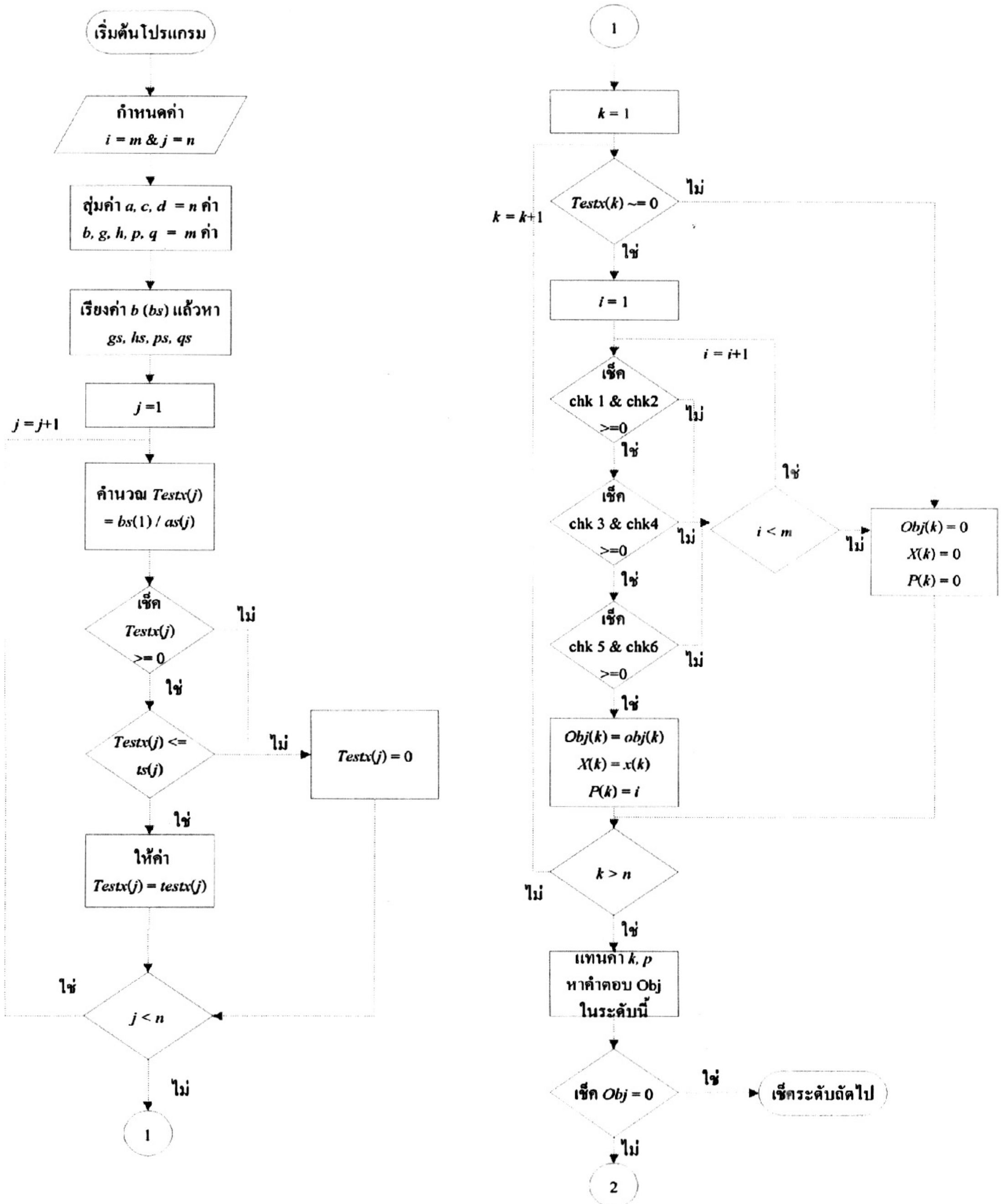
18. ในกรณีที่เช็คเงื่อนไขแล้วเป็นไปตามเงื่อนไข จะได้คำตอบ $Obj5(j1, j2, j3, j4, k)$, ตำแหน่ง i ของค่า b, $x(j1) = t(j1)$, $x(j2) = t(j2)$, $x(j3) = t(j3)$ $x(j4) = t(j4)$ และ $x(k) = \dfrac{b_s(i) - t_s(j1)a_s(j1) - t_s(j2)a_s(j2) - t_s(j3)a_s(j3) - t_s(j4)a_s(j4)}{a_s(k)}$ แล้วเก็บค่าคำตอบในระดับนี้

19. ถ้าเช็คแล้วไม่เป็นไปตามเงื่อนไขในระดับนี้ จะได้คำตอบในระดับนี้เป็น $Obj5(j1, j2, j3, j4, k)$ เท่ากับศูนย์

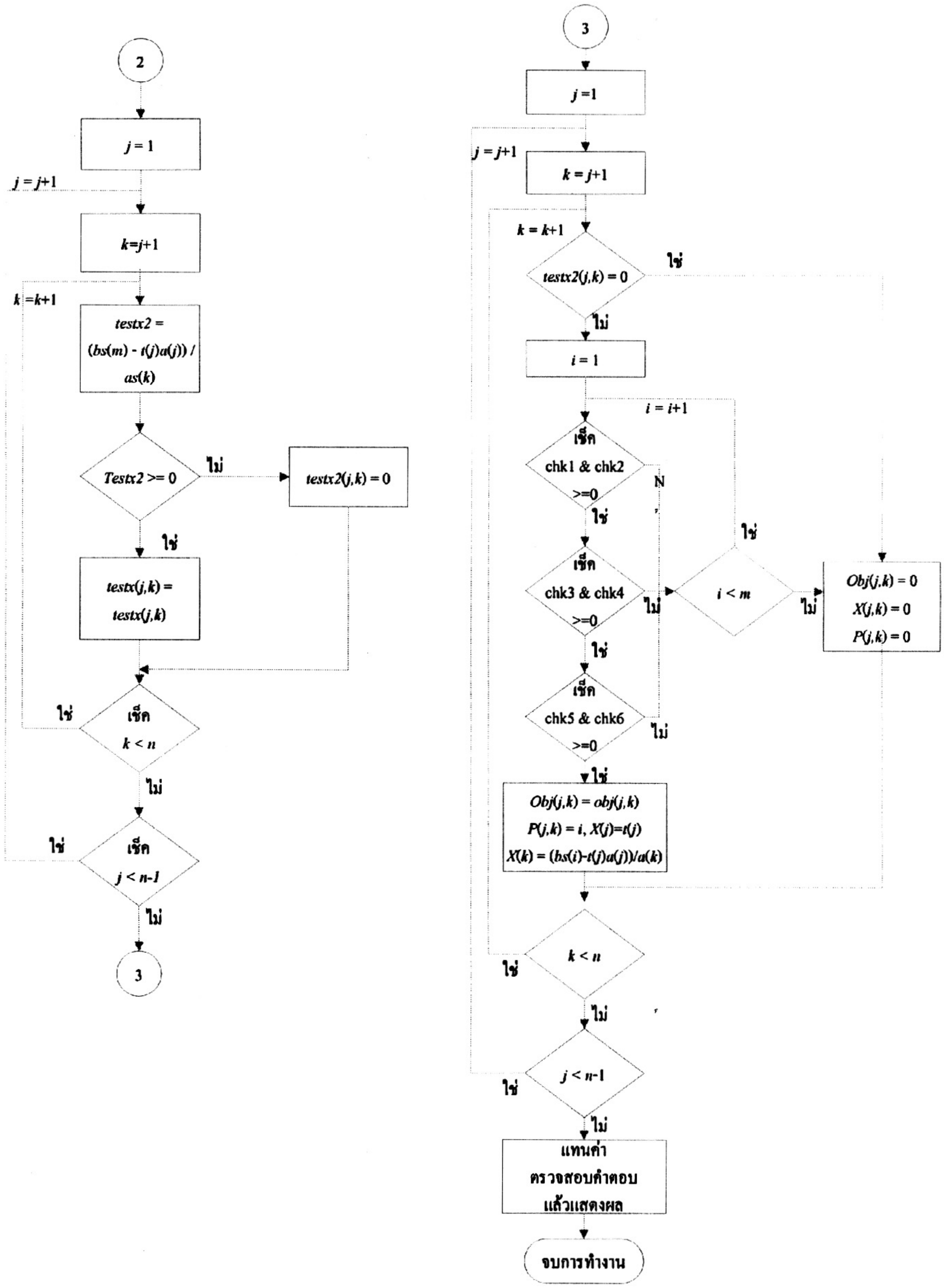
20. ในกรณีที่ $testx$ มีค่าเป็นศูนย์จะทำให้จบการคำนวณ แล้วนำคำตอบที่ได้ทั้งหมดมาเปรียบเทียบกัน คำตอบจะมีการตรวจสอบค่า Obj ที่ไม่เท่ากับศูนย์ โปรแกรมจะแสดงคำตอบที่น้อยที่สุด แต่ไม่เท่ากับศูนย์ แต่ถ้าได้ค่า Obj ทุกระดับที่หาได้มีค่าเท่ากับศูนย์ จะแสดงว่าไม่มีคำตอบที่เหมาะสม จากหลักการข้างบนสามารถสร้างแผนภูมิดังภาพที่ 2 แสดงการแก้ปัญหาใน 2 ระดับขั้นตอนการแก้ปัญหาหน้าถัดไป

จากการแก้ปัญหา ที่มีขนาดต่าง ๆ ในแต่ละวิธีให้ค่าคำตอบสำหรับเวลาที่ใช้ในการคำนวณเท่ากัน จึงได้ทำการตรวจสอบวิธีในการคำนวณ เพื่อดูประสิทธิภาพของการคำนวณของแต่ละวิธี โดยทำการตรวจสอบด้วยวิธีทางสถิติ ด้วยการทดสอบค่าที (t-Test) ซึ่งแต่ละขนาดจะมีค่าเวลาที่ได้จากการแก้ปัญหาในแต่ละวิธี แต่เนื่องจากว่าไม่รู้ว่าความแปรปรวนของทั้งสองวิธีเท่ากันใหม่จึงต้องทำการทดสอบค่าเอฟ (F-Test) ก่อน เพื่อจะกำหนดได้ว่าความแปรปรวนของวิธีสองวิธีมีค่าเท่าหรือไม่เท่ากัน เพื่อกำหนดวิธีการทดสอบค่าที ซึ่งจะต้องทำการแยกศึกษาเป็นการทดสอบค่าทีเมื่อค่าความแปรปรวนของสองข้อมูลเท่ากัน และ การทดสอบค่าทีเมื่อค่าความแปรปรวนของสองข้อมูลไม่เท่ากัน ดูรายละเอียดได้ในการคำนวณตารางผนวก ก16 เป็นต้นไป

ในการหาคำตอบของการศึกษานี้ ได้ทำการเขียนอัลกอริทึมในกำหนดการสำเร็จรูป MATLAB และทดสอบความสามารถในการแก้ปัญหาโดยกำหนดขนาดของปัญหาให้มีขนาดใหญ่ขึ้น การทดลองในงานวิจัยใช้คอมพิวเตอร์ Intel pentium M centrino processor 1.4 GHz และ RAM 256 MB โดยทำการทดลองกับชุดของปัญหาจำนวน 49 ขนาด ที่มีขนาดของปัญหาแตกต่างกัน ซึ่งได้มาจากการสร้างปัญหาแบบสุ่ม โดยให้จำนวนตัวแปร (x) อยู่ในช่วง 10-50,000 ตัวแปร โดยมีเงื่อนไขในการเลือก (b) อยู่ในช่วง 10 – 10,000 ทางเลือก สำหรับการแก้ปัญหา โดยใช้ซอฟต์แวร์ MATLAB คำสั่ง linprog และการพัฒนาขั้นตอนวิธีเฉพาะในการแก้ปัญหาสำหรับปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน ทั้งกรณีที่ไม่มีขอบเขต และ กรณีที่มีขอบเขต โดยกำหนดเวลาในการแก้ปัญหาเท่ากับ 5,000 วินาที เนื่องจากการแก้ปัญหา โดยใช้ซอฟต์แวร์ MATLAB ในการหาคำตอบที่ดีที่สุดสำหรับบางปัญหาใช้เวลานาน เวลาที่ได้จะเป็นเวลาเฉลี่ยในการแก้ปัญหาแต่ละวิธี



<u>ภาพที่ 2</u> แผนภาพขั้นตอนในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนมีขอบเขต โดยวิธีตรง


2
j = 1
j = j+1
k=j+1
k =k+1
testx2 =
(bs(m) - t(j)a(j)) /
as(k)
Testx2 >= 0
ไม่
testx2(j,k) = 0
ใช่
testx(j,k) =
testx(j,k)
ใช่
เช็ค
k < n
ไม่
ใช่
เช็ค
j < n-1
ไม่
3
3
j =1
j = j+1
k = j+1
k = k+1
testx2(j,k) = 0
ใช่
ไม่
i = 1
i = i+1
เช็ค
chk1 & chk2
>=0
N
ใช่
ใช่
เช็ค
chk3 & chk4
>=0
ไม่
i < m
ไม่
Obj(j,k) = 0
X(j,k) = 0
P(j,k) = 0
ใช่
เช็ค
chk5 & chk6
>=0
ไม่
ใช่
Obj(j,k) = obj(j,k)
P(j,k) = i, X(j)=t(j)
X(k) = (bs(i)-t(j)a(j))/a(k)
k < n
ใช่
ไม่
j < n-1
ใช่
ไม่
แทนค่า
ตรวจสอบคำตอบ
แล้วแสดงผล
จบการทำงาน

ภาพที่ 2 (ต่อ)

ผลและวิจารณ์

1. ผลการเปรียบเทียบเวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน (Two-Stage Single Constrained Linear Fractional Programming)

ในการทดลองแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน โดยใช้วิธีในการแก้ปัญหา 2 วิธี คือ การแก้ปัญหาโดยใช้คำสั่ง linprog และการแก้ปัญหาโดยใช้วิธีตรง ในการทดลองจะพบว่าคำตอบที่ได้เป็นคำตอบเดียวกัน ดังนั้นจึงทำการเก็บค่าเวลาในการคำนวณของเวลาในแต่ละวิธีดังตารางข้างล่างนี้

ตารางที่ 1 ผลการแก้ปัญหาแบบไม่มีขอบเขต ด้วยคำสั่ง linprog (หน่วย : วินาที)

	m =						
n =	10	50	100	500	1000	5000	10000
10	0.3603	0.3906	0.4777	0.7242	1.2438	15.6023	37.4028
50	0.3766	0.4386	0.6369	2.5055	6.4683	220.4459	1137.8853
100	0.4015	0.5239	0.8813	6.5804	21.6238	1101.8016	4961.5353
500	0.5287	2.0891	6.1097	215.2506	1087.5247	>5000	n/a
1000	1.0705	6.5752	22.0808	1098.2793	4842.1096	>5000	n/a
5000	7.1031	216.0088	1107.3905	>5000	n/a	n/a	n/a
10000	25.4006	1124.5561	>5000	n/a	n/a	n/a	n/a

หมายเหตุ n/a = ไม่รู้ค่าเวลาในการหาผลเฉลย

ตารางที่ 1 (ต่อ)

	m =		
n =	20000	30000	50000
10	188.9126	482.4368	1736.3745
50	>5000	n/a	n/a
100	>5000	n/a	n/a
500	n/a	n/a	n/a
1000	n/a	n/a	n/a
5000	n/a	n/a	n/a
10000	n/a	n/a	n/a

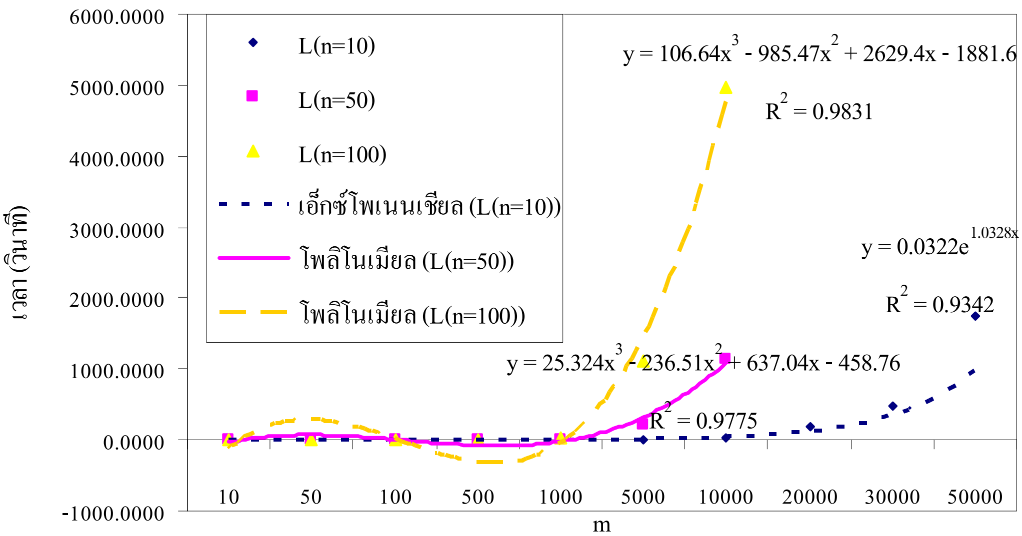
ตารางที่ 2 ผลการแก้ปัญหาแบบไม่มีขอบเขต โดยวิธีตรง (หน่วย : วินาที)

	m =						
n =	10	50	100	500	1000	5000	10000
10	0.0380	0.0570	0.0750	0.3033	0.7329	10.9492	31.1207
50	0.0601	0.1172	0.1862	1.5252	3.8144	65.7760	327.1053
100	0.0701	0.1822	0.3333	3.0696	7.4878	138.9127	1432.9163
500	0.1814	0.6900	1.6044	11.5466	39.7161	628.5129	3068.0335
1000	0.3906	1.3391	3.0384	26.3458	97.4813	942.2940	>5000
5000	3.6682	8.9347	15.5343	136.0346	474.6338	>5000	n/a
10000	12.9874	22.1129	40.6716	263.3957	850.8304	>5000	n/a

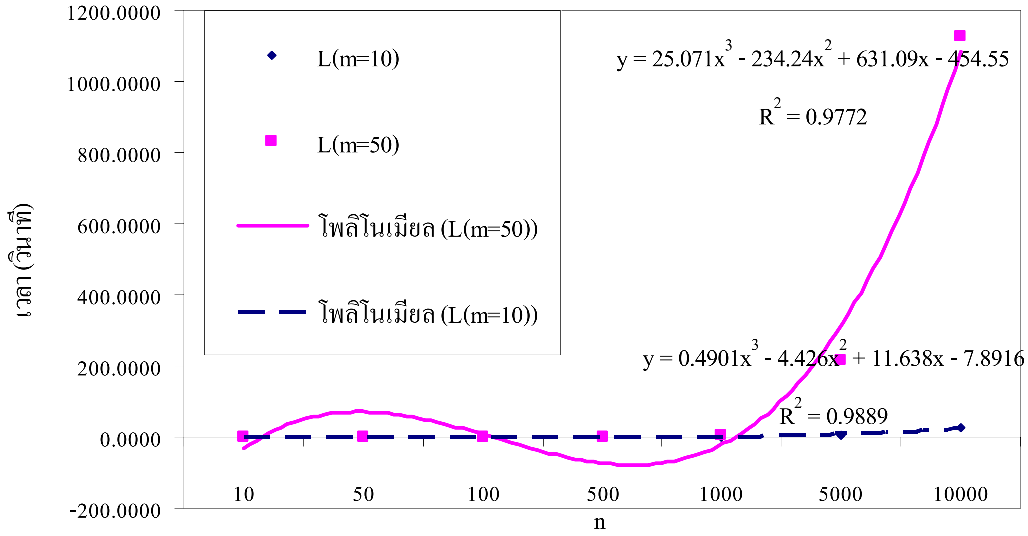
หมายเหตุ n/a = ไม่รู้ค่าเวลาในการหาผลเฉลย

<u>ตารางที่ 2</u> (ต่อ)

	m =		
n =	20000	30000	50000
10	176.9143	441.7252	1281.2362
50	1448.9363	2299.5077	>5000
100	2162.0679	>5000	n/a
500	>5000	n/a	n/a
1000	n/a	n/a	n/a
5000	n/a	n/a	n/a
10000	n/a	n/a	n/a



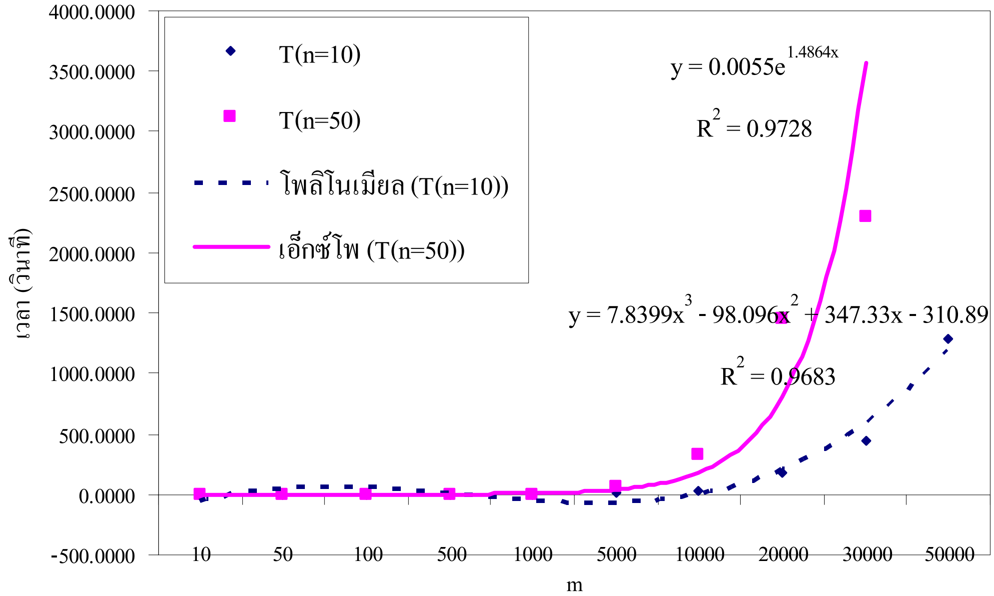
<u>ภาพที่ 3</u> กราฟแสดงแนวโน้มเวลาสำหรับการใช้คำสั่ง linprog เมื่อเปลี่ยนแปลงขนาดของตัวแปร



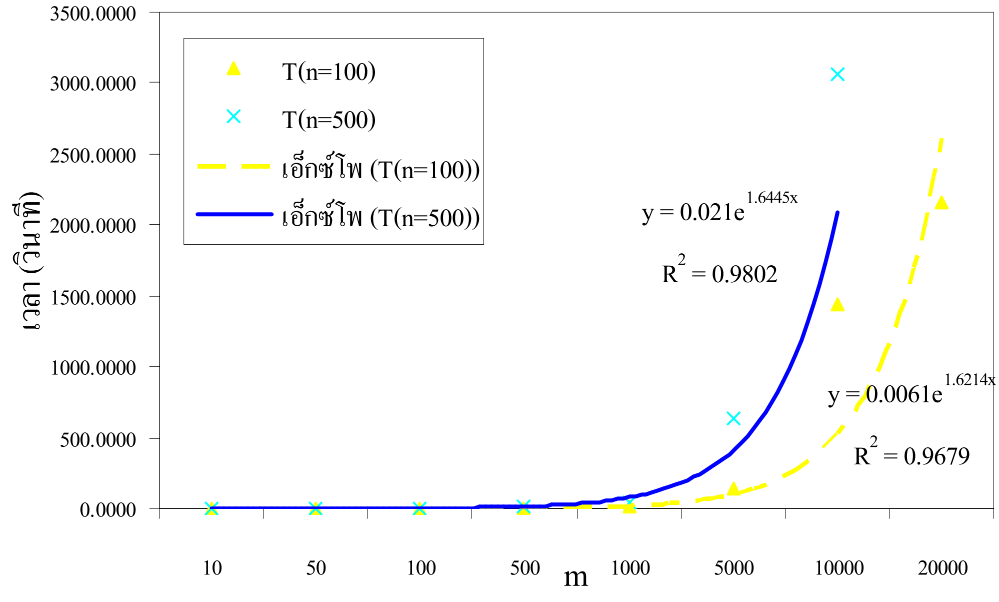
<u>ภาพที่ 4</u>   กราฟแสดงแนวโน้มเวลาสำหรับการใช้คำสั่ง linprog เมื่อเปลี่ยนแปลงจำนวนเงื่อนไข

จากตารางที่ 1 และภาพที่ 3 และภาพที่ 4 จะเห็นได้ว่าในการใช้วิธีการแก้ปัญหาด้วยคำสั่ง linprog เมื่อพิจารณาสำหรับทุก ๆ ค่า n เวลาในการคำนวณจะเพิ่มขึ้นตามค่าของ m และเมื่อพิจารณาค่าของ m จะได้ว่าเวลาในการคำนวณเพิ่มขึ้นเมื่อ n เพิ่มขึ้นเช่นเดียวกัน โดยที่ลักษณะเวลาที่เพิ่มขึ้นนั้นไม่แน่นอน ซึ่งจะเห็นได้ว่าเมื่อสร้างกราฟแนวโน้มสำหรับค่า n และ m จะได้กราฟที่เป็นฟังก์ชันพหุนาม (Polynomial function)หรือฟังก์ชันเลขชี้กำลัง (Exponential function) ที่มีแนวโน้มเพิ่มขึ้น

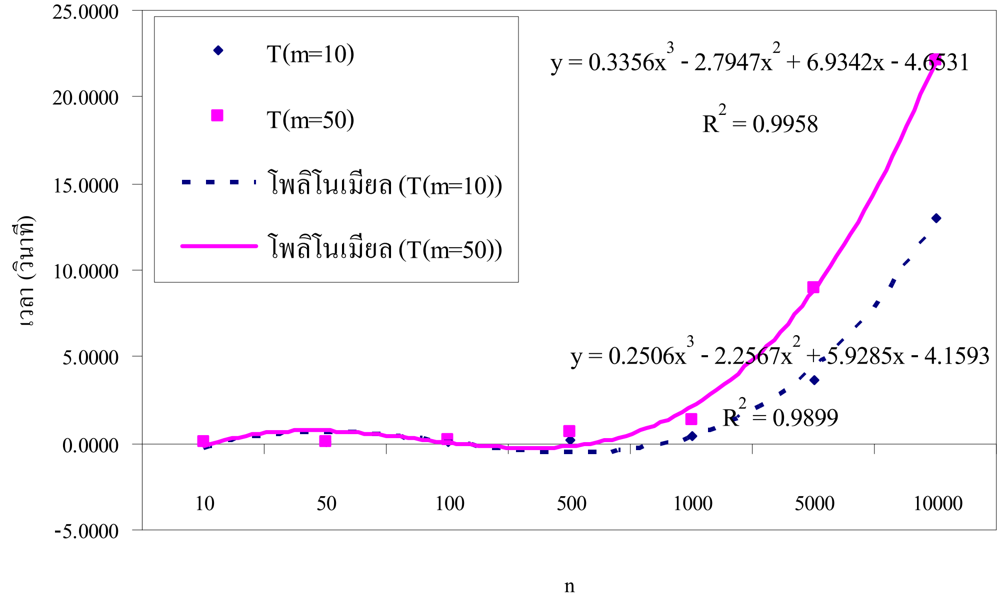
จากตารางที่ 2 และภาพที่ 5, 6 และ 7 ซึ่งแสดงเวลาในการแก้ปัญหาแบบวิธีตรง จะเห็นได้ว่าทุก ๆ ค่า n เวลาในการคำนวณจะเพิ่มขึ้นตามค่าของ m ในลักษณะฟังก์ชันเลขชี้กำลัง หรือฟังก์ชันโพลิโนเมียล ด้วยแนวโน้มที่เพิ่มขึ้น และเมื่อพิจารณาค่าของ m จะเห็นได้ว่าเวลาในการคำนวณเพิ่มขึ้นเมื่อ n เพิ่มขึ้น โดยจะสังเกตเห็นได้ว่าช่วงแรกที่ปัญหามีขนาดไม่ใหญ่นัก เวลาที่ใช้ในการคำนวณมีค่าใกล้เคียงกันแต่เมื่อปัญหาเริ่มมีขนาดใหญ่เวลาที่ได้จะมีค่ามากขึ้น จากค่าความชันที่เพิ่มขึ้น ซึ่งอาจจะเป็นแบบฟังก์ชันพหุนาม หรือ ฟังก์ชันเลขชี้กำลัง



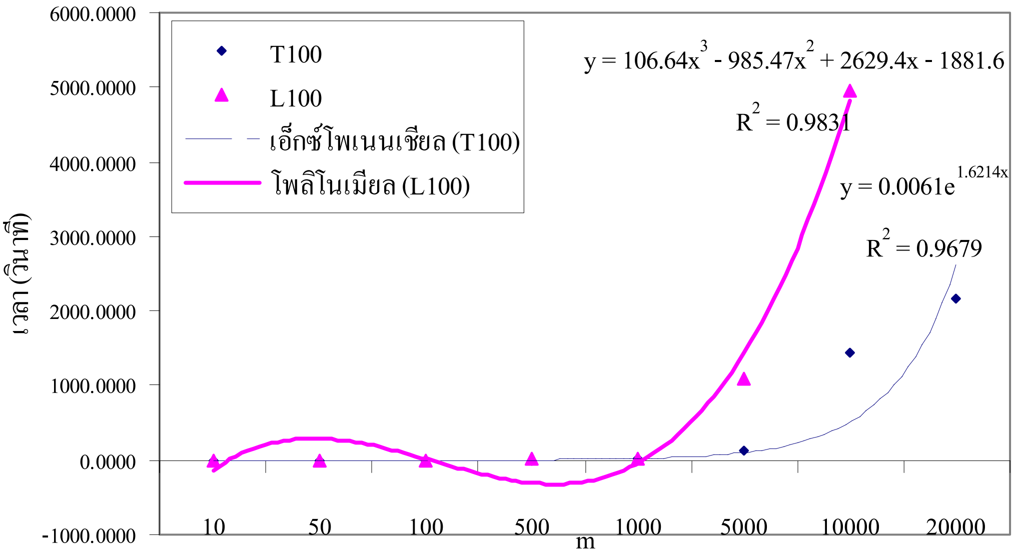
ภาพที่ 5 กราฟแสดงแนวโน้มเวลาสำหรับวิธีตรง เมื่อเขนาดของตัวแปรเท่ากับ 10 และ 50



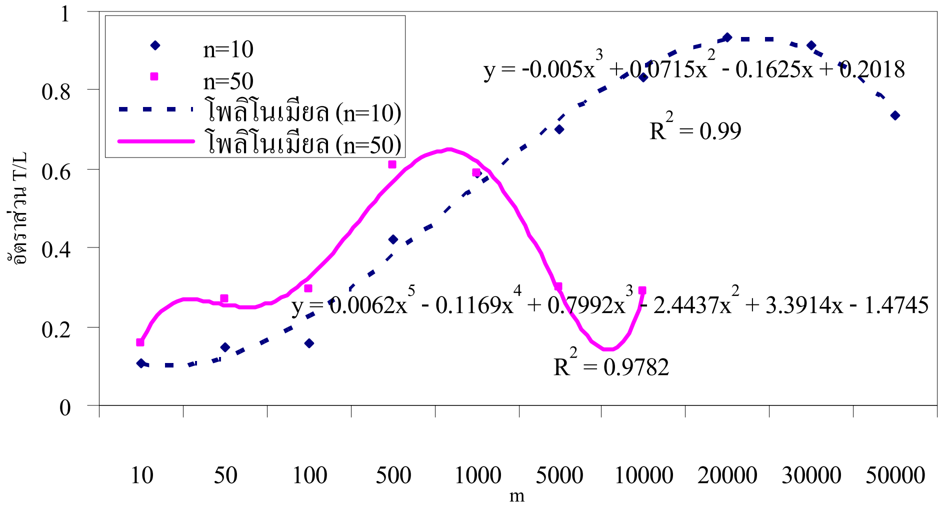
ภาพที่ 6 กราฟแสดงแนวโน้มเวลาสำหรับวิธีตรง เมื่อขนาดของตัวแปรเท่ากับ 100 และ 500



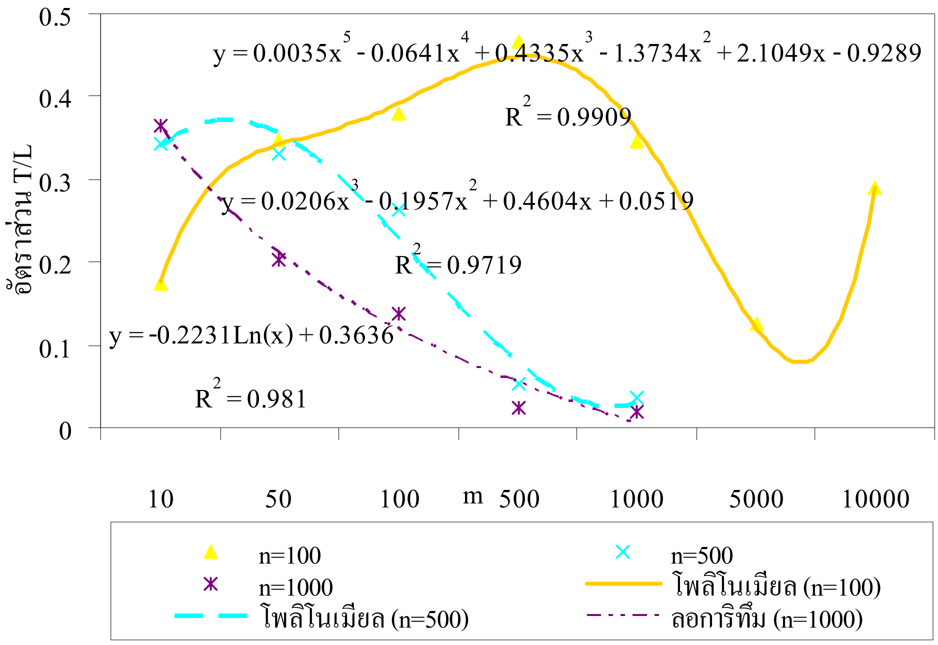
ภาพที่ 7 กราฟแสดงแนวโน้มเวลาสำหรับวิธีตรง เมื่อจำนวนเงื่อนไขเท่ากับ 10 และ 50



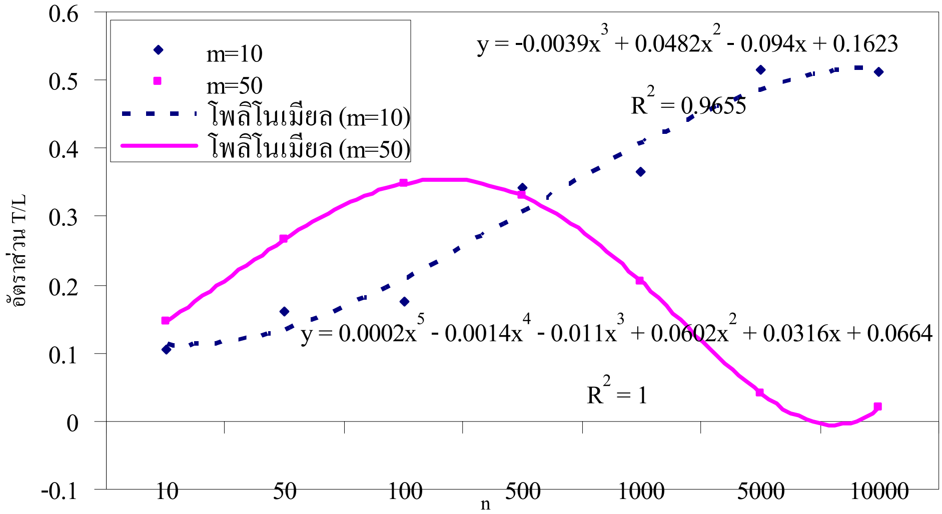
ภาพที่ 8 กราฟแสดงแนวโน้มเวลาระหว่าง วิธีแก้ปัญหาโดยใช้คำสั่ง linprog (L) และ วิธีตรง (T) ที่ขนาดของตัวแปรมีค่าเท่ากับ 100



<u>ภาพที่ 9</u> กราฟแสดงอัตราส่วนแนวโน้มเวลาระหว่าง วิธีแก้ปัญหาโดยใช้คำสั่ง linprog (L) และ วิธีตรง (T) ที่จำนวนตัวแปรเท่ากับ 10 และ 50



<u>ภาพที่ 10</u> กราฟแสดงอัตราส่วนแนวโน้มเวลาระหว่าง วิธีแก้ปัญหาโดยใช้คำสั่ง linprog (L) และ วิธีตรง (T) ที่จำนวนตัวแปรเท่ากับ 100 500 และ 1000



ภาพที่ 11 กราฟแสดงอัตราส่วนแนวโน้มเวลาระหว่าง วิธีแก้ปัญหาโดยใช้คำสั่ง linprog (L) และวิธีตรง (T) เมื่อเปรียบเทียบจำนวนเงื่อนไขเท่ากับ 10 และ 50

เมื่อทำการเปรียบเทียบสองวิธีในการแก้ปัญหาเศษส่วนเชิงเส้นแบบหนึ่งเงื่อนไขสองขั้นตอนแบบไม่มีขอบเขต เมื่อทำการเพิ่มขนาดของปัญหาทั้งจำนวนตัวแปรหรือ จำนวนเงื่อนไขย่อมส่งผลให้เวลาที่ใช้ในการแก้ปัญหามีแนวโน้มเพิ่มขึ้น เมื่อพิจารณาจากภาพที่ 8 จะเห็นได้ว่าเมื่อเปรียบเทียบจำนวนตัวแปรที่เท่ากัน วิธีการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน โดยวิธีตรง ใช้เวลาน้อยกว่าการแก้ปัญหาด้วยคำสั่ง linprog และเมื่อทำการเปรียบเทียบเวลาเป็นอัตราส่วนของเวลาในการแก้ปัญหาของวิธีตรงและวิธีคำสั่ง linprog จะเห็นได้ว่าวิธีตรงให้ค่าที่น้อยกว่า 1 นั่นแสดงว่าใช้เวลาน้อยกว่าทุกกรณี ไม่ว่าจะทำการเปรียบเทียบในลักษณะของจำนวนตัวแปร หรือจำนวนเงื่อนไข เมื่อทำการสร้างกราฟแนวโน้ม จะเห็นได้ว่าลักษณะการเพิ่มเวลาเมื่อเปรียบเทียบกับขนาดของปัญหา อาจเป็นได้ทั้งฟังก์ชันแบบพหุนาม และ ฟังก์ชันเลขชี้กำลัง ดังภาพที่ 9 ภาพที่10 และ ภาพที่11

2. ผลเปรียบเทียบเวลาในการแก้ปัญหาของปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต (Two-Stage Single Constrained Linear Fractional Programming With Bounded Variables)

ในการทดลองแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต โดยทำการเปรียบเทียบประสิทธิภาพในการแก้ปัญหา 2 วิธี คือ การแก้ปัญหาโดยใช้คำสั่ง

linprog และการแก้ปัญหาโดยใช้วิธีตรง ในการทดลองจะพบว่าคำตอบที่ได้เป็นคำตอบเดียวกัน ดังนั้นจึงทำการเก็บค่าเวลาในการคำนวณของเวลาในแต่ละวิธีดังตารางข้างล่างนี้

ตารางที่ 3 ผลการแก้ปัญหาแบบมีขอบเขต ด้วยคำสั่ง linprog (หน่วย : วินาที)

	m =						
n =	10	50	100	500	1000	5000	10000
10	0.5115	0.5378	0.7039	1.2899	3.4239	36.9311	80.8846
50	0.6269	0.7019	1.0435	3.5681	12.4191	280.0847	1242.4182
100	0.7280	1.1128	1.7345	9.1591	30.5777	1193.1544	>5000
500	3.6492	8.3080	15.8537	265.9173	1218.8726	>5000	n/a
1000	20.2782	35.4439	60.5169	1314.8335	>5000	n/a	n/a
5000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
10000	n/a	n/a	n/a	n/a	n/a	n/a	n/a

หมายเหตุ n/a = ไม่รู้ค่าเวลาในการหาผลเฉลย

ตารางที่ 3 (ต่อ)

	m =		
n =	20000	30000	50000
10	224.266	553.6100	>5000
50	>5000	n/a	n/a
100	n/a	n/a	n/a
500	n/a	n/a	n/a
1000	n/a	n/a	n/a
5000	n/a	n/a	n/a
10000	n/a	n/a	n/a

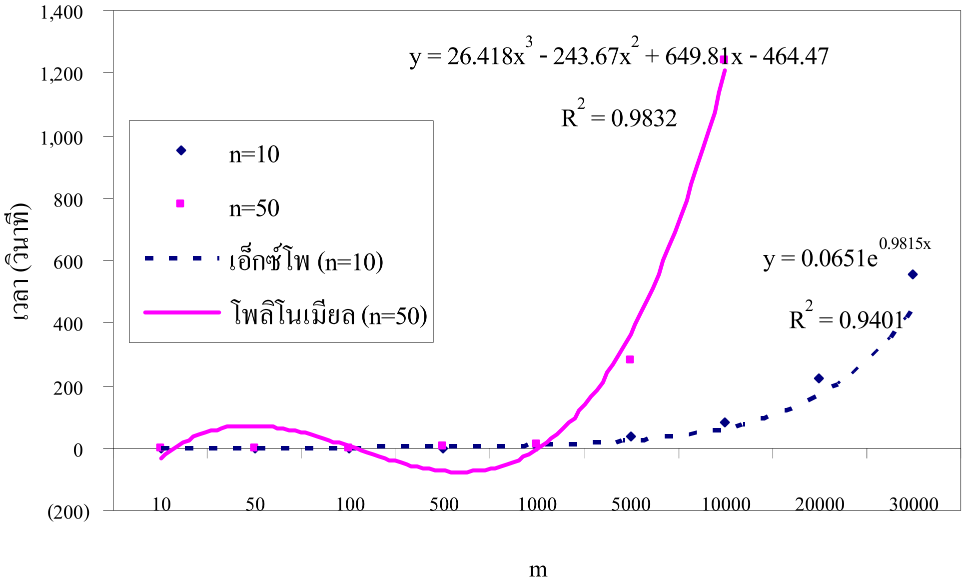
ตารางที่ 4 ผลการแก้ปัญหาแบบมีขอบเขต โดยวิธีตรง (หน่วย : วินาที)

	m =						
n =	10	50	100	500	1000	5000	10000
10	0.3007	0.4098	0.5130	1.8576	6.1188	190.0702	755.0986
50	n/a	n/a	n/a	n/a	n/a	n/a	n/a
100	n/a	n/a	n/a	n/a	n/a	n/a	n/a
500	n/a	n/a	n/a	n/a	n/a	n/a	n/a
1000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
5000	n/a	n/a	n/a	n/a	n/a	n/a	n/a
10000	n/a	n/a	n/a	n/a	n/a	n/a	n/a

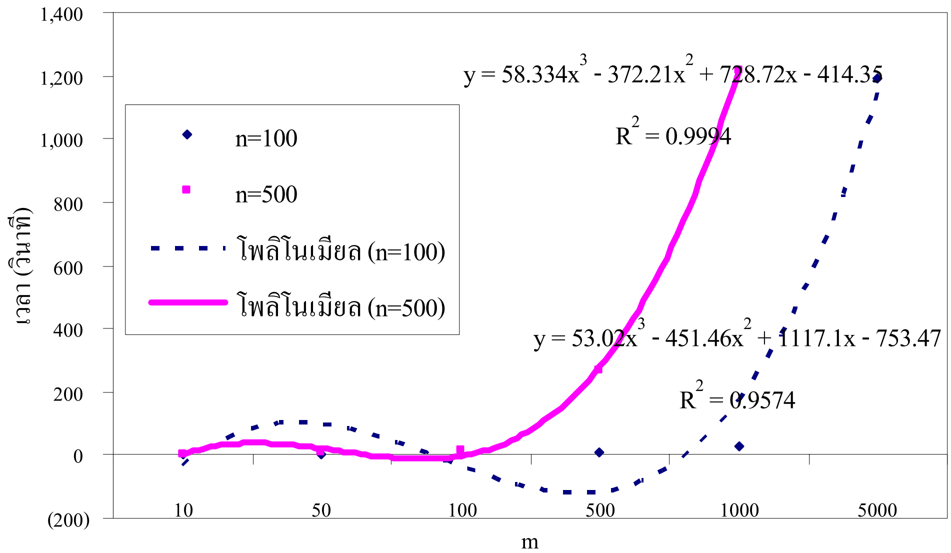
หมายเหตุ n/a = ไม่รู้ค่าเวลาในการหาผลเฉลย

ตารางที่ 4 (ต่อ)

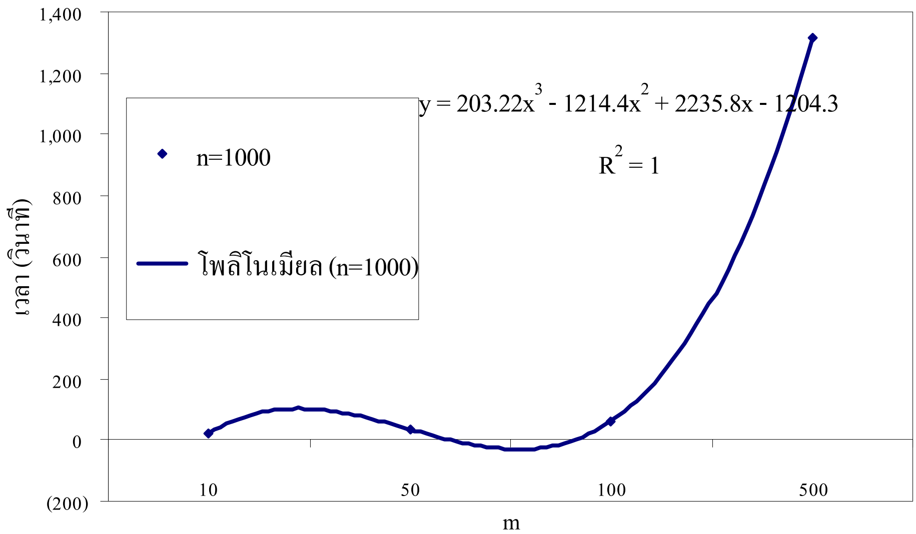
	m =		
n =	20000	30000	50000
10	>5000	n/a	n/a
50	n/a	n/a	n/a
100	n/a	n/a	n/a
500	n/a	n/a	n/a
1000	n/a	n/a	n/a
5000	n/a	n/a	n/a
10000	n/a	n/a	n/a



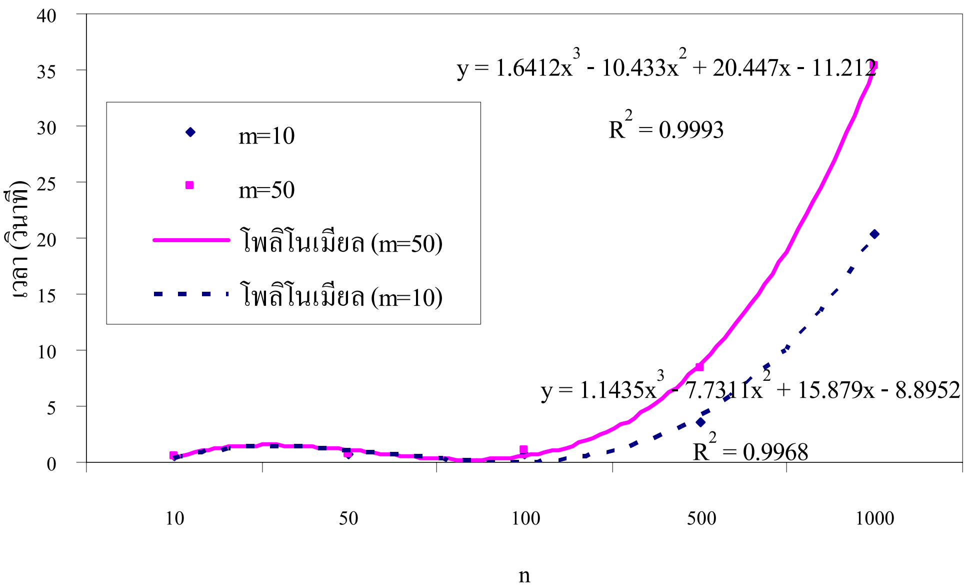
ภาพที่ 12 กราฟแสดงแนวโน้มเวลาสำหรับการใช้คำสั่ง linprog กับปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อขนาดของตัวแปรเท่ากับ 10 และ 50



ภาพที่ 13 กราฟแสดงแนวโน้มเวลาสำหรับการใช้คำสั่ง linprog กับปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อขนาดของตัวแปรเท่ากับ 100 และ 500



<u>ภาพที่ 14</u> กราฟแสดงแนวโน้มเวลาสำหรับการใช้คำสั่ง linprog กับปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อขนาดของตัวแปรเท่ากับ 1000

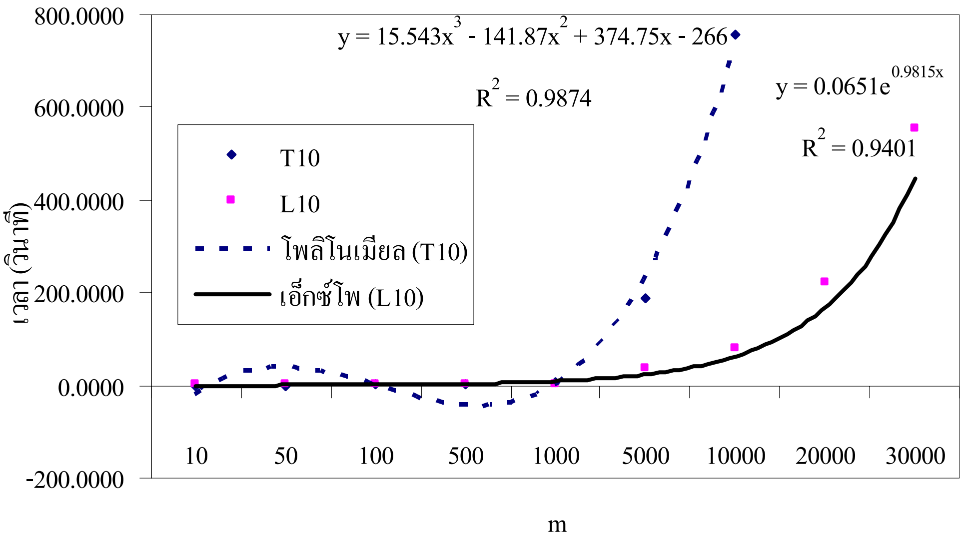


<u>ภาพที่ 15</u> กราฟแสดงแนวโน้มเวลาสำหรับการใช้คำสั่ง linprog กับปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อเปรียบเทียบจำนวนเงื่อนไข

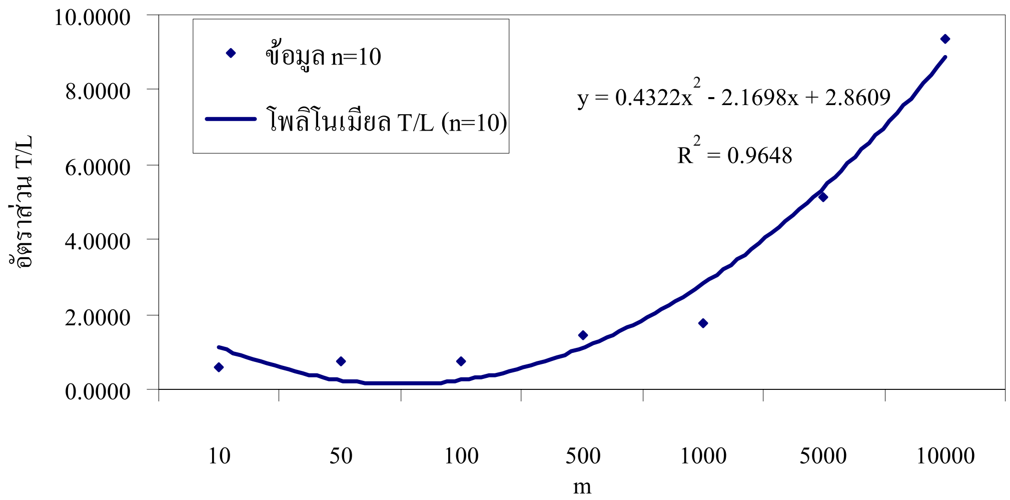
จากตารางที่ 3 และภาพที่ 12 ถึง ภาพที่ 15 เป็นการแสดงเวลาที่ใช้ในการหาคำตอบด้วยวิธีคำสั่ง linprog สำหรับทุก ๆ ค่า n เวลาในการคำนวณจะเพิ่มขึ้นตามค่าของ m และเมื่อพิจารณาค่า

ของ m จะเห็นได้ว่าเวลาในการคำนวณเพิ่มขึ้นเมื่อ n เพิ่มขึ้นเช่นเดียวกัน ความแตกต่างของเวลาเมื่อขนาดของปัญหาใหญ่ขึ้นมีแนวโน้มในการเพิ่มขึ้นเป็นฟังก์ชันพหุนาม และ ฟังก์ชันเลขชี้กำลัง

จากตารางที่ 4 สำหรับทุก ๆ ค่า n เวลาในการคำนวณจะเพิ่มขึ้นตามค่าของ m แต่คำตอบสามารถหาได้ที่ $n = 10$ เท่านั้น เนื่องจากเมื่อเพิ่มค่า $n = 50$ โปรแกรมจะใช้หน่วยความจำมากจนกระทั่งหน่วยความจำหมด จึงไม่สามารถหาค่าออกมาได้ และเมื่อพิจารณาเวลาที่ใช้ในการแก้ปัญหาโดยวิธีตรงเมื่อ $n = 10$ มีแนวโน้มในการเพิ่มขึ้น เมื่อกำหนดปัญหาให้มีเงื่อนไขเพิ่มขึ้นเป็นฟังก์ชันของพหุนามดีกรี3 ดังภาพที่ 16



ภาพที่ 16 กราฟแสดงแนวโน้มเวลาในการแก้ปัญหาโดยใช้คำสั่ง linprog (L) และ วิธีตรง (T) ที่ขนาดของตัวแปรเท่ากับ 10 กับปัญหาแบบมีขอบเขต



<u>ภาพที่ 17</u> กราฟแสดงแนวโน้มอัตราส่วนระหว่างวิธีแก้ปัญหาโดยใช้คำสั่ง linprog (L) และ วิธีตรง (T) ที่ขนาดของตัวแปรมีค่าเท่ากับ 10 กับปัญหาแบบมีขอบเขต

จากการทดลองจะได้ว่าช่วงแรกที่ปัญหามีขนาด 10 x10 ถึง 10 x 100 นั้นวิธีตรงจะใช้เวลาน้อยกว่า แต่เมื่อขนาดใหญ่กว่านั้น วิธีการแก้ปัญหาแบบเชิงเส้น หรือการใช้คำสั่ง linprog สามารถทำการแก้ปัญหาได้ดีกว่า ดังตารางที่ 3 และ ตารางที่ 4 รวมถึง ภาพที่ 16 และ ภาพที่ 17 ที่เป็นเช่นนี้เนื่องจากในการทำปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนกรณีที่มีขอบเขตของตัวแปรแบบวิธีตรง ยังไม่สามารถจัดรูปแบบในการหาคำตอบได้เป็นที่แน่นอน ต้องตรวจสอบค่าทุกค่าที่เป็นไปได้เพื่อหาคำตอบที่ดีที่สุดในการคำนวณ ซึ่งมีผลต่อหน่วยความจำของคอมพิวเตอร์ที่ใช้ในการทดลอง จึงทำให้ผลการทดลองในที่นี้ไม่สามารถทำการหาค่าได้

เมื่อทำการวิเคราะห์กราฟของทั้งสองวิธี เมื่อทำการเปรียบเทียบจำนวนตัวแปรหรือจำนวนเงื่อนไข จะเห็นได้ว่าเวลาที่ใช้ในการแก้ปัญหาแปรผันตามขนาดของข้อมูล จากการสร้างปัญหาโดยการสุ่ม จะได้ว่าแนวโน้มที่เพิ่มขึ้นของเวลาในการทดลองนี้ ส่วนใหญ่เป็นแนวโน้มในลักษณะฟังก์ชันพหุนาม แต่ในกราฟแนวโน้มบางเส้นมีการเพิ่มเป็นฟังก์ชันเลขชี้กำลัง

3. <u>ผลการเปรียบเทียบเวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน (Two-Stage Single Constrained Linear Fractional Programming) ทางสถิติของแต่ละวิธี</u>

จากผลการทดลอง ทำการเก็บเวลาขนาดปัญหาละ 10 ตัวอย่าง เพื่อทดลองแก้ปัญหาด้วยคำสั่ง linprog และ วิธีตรง แล้วหาค่าเฉลี่ยของเวลาที่ใช้ในการแก้ปัญหานั้นในแต่ละขนาด และวิธี เพื่อศึกษาประสิทธิภาพความสามารถในการคำนวณการแก้ปัญหาของสองวิธี ด้วยวิธีทางสถิติ รายละเอียดในการคำนวณแสดงในตารางผนวก ก16 ถึง ตารางผนวก ก22 โดยกำหนดดังนี้

กำหนดให้ ค่าสมมติฐานหลัก คือ ค่าเฉลี่ยเวลาที่ได้จากการแก้ปัญหาโดยวิธีคำสั่ง linprog (μ_L) และ ค่าเฉลี่ยเวลาที่ได้จากการแก้ปัญหาโดยวิธีตรง (μ_T) เท่ากัน $(H_0 : \mu_L = \mu_T)$ สมมติฐานรองคือ ความสามารถหรือเวลาเฉลี่ยในการแก้ปัญหาของสองวิธีไม่เท่ากัน $(H_1 : \mu_L \neq \mu_T)$ ที่ 95 เปอร์เซ็นต์ระดับความเชื่อมั่น $(\alpha = 0.05)$ จะสรุปได้ดังนี้

<u>ตารางที่ 5</u> ผลการเปรียบเทียบความสามารถการแก้ปัญหาของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ 10

n = 10								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ(ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าทีคำนวณ	ค่าทีวิกฤต	P-value	ผล
10	0.038	0.008	0.360	0.015	-60.884	2.101	2.67E-22	ปฏิเสธ H_0
50	0.057	0.013	0.391	0.012	-59.677	2.101	3.82E-22	ปฏิเสธ H_0
100	0.075	0.008	0.478	0.031	-40.083	2.228	2.23E-12	ปฏิเสธ H_0
500	0.303	0.086	0.724	0.013	-15.242	2.262	9.82E-08	ปฏิเสธ H_0
1000	0.733	0.249	1.244	0.026	-6.450	2.262	1.18E-04	ปฏิเสธ H_0
5000	10.949	3.493	15.602	2.165	-3.581	2.101	0.002	ปฏิเสธ H_0
10000	31.121	8.102	37.403	0.543	-2.446	2.262	0.037	ปฏิเสธ H_0
20000	176.914	92.059	188.913	2.029	-0.412	2.262	0.690	ปฏิเสธ H_0 ไม่ได้
30000	441.73	114.25	482.44	146.57	-0.693	2.101	0.497	ปฏิเสธ H_0 ไม่ได้
50000	1281.24	636.15	1736.37	21.48	-2.261	2.262	0.050	ปฏิเสธ H_0 ไม่ได้

ตารางที่ 6 ผลการเปรียบเทียบความสามารถการแก้ปัญหาของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ 50

n = 50								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ (ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าทีคำนวณ	ค่าทีวิกฤต	P-value	ผล
10	0.0601	0.0105	0.3766	0.0185	-47.0575	2.1009	2.68E-20	ปฏิเสธ H_0
50	0.1172	0.0094	0.4386	0.0226	-41.5310	2.1788	2.46E-14	ปฏิเสธ H_0
100	0.1862	0.0237	0.6369	0.0295	-37.6303	2.1009	1.44E-18	ปฏิเสธ H_0
500	1.5252	0.2876	2.5055	0.1420	-9.6664	2.1604	2.66E-07	ปฏิเสธ H_0
1000	3.8144	1.0554	6.4683	0.9070	-6.0307	2.1009	1.06E-05	ปฏิเสธ H_0
5000	65.7760	52.1569	220.4459	2.2480	-9.3689	2.2622	0.0000	ปฏิเสธ H_0
10000	327.1053	157.946	1137.885	7.7296	-16.2134	2.2622	0.0000	ปฏิเสธ H_0

ตารางที่ 7 ผลการแก้ปัญหาเปรียบเทียบความสามารถการแก้ปัญหาของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ 100

n = 100								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ (ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าทีคำนวณ	ค่าทีวิกฤต	P-value	ผล
10	0.0701	0.0094	0.4015	0.0463	-22.1958	2.2281	7.73E-10	ปฏิเสธ H_0
50	0.1822	0.0132	0.5239	0.0217	-42.5265	2.1009	1.63E-19	ปฏิเสธ H_0
100	0.3333	0.0448	0.8813	0.0781	-19.2447	2.1009	1.87E-13	ปฏิเสธ H_0
500	3.0696	0.5288	6.5804	0.3288	-17.8301	2.1009	6.93E-13	ปฏิเสธ H_0
1000	7.4878	1.3748	21.6238	0.3709	-31.3920	2.2281	2.53E-11	ปฏิเสธ H_0
5000	138.9127	59.9067	1101.802	2.8043	-50.7722	2.2622	0.0000	ปฏิเสธ H_0
10000	1432.916	436.47	4961.535	24.7248	-25.5241	2.2622	0.0000	ปฏิเสธ H_0

ตารางที่ 8 ผลการแก้ปัญหาเปรียบเทียบความสามารถการแก้ปัญหาของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ 500

n = 500								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ (ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าทีคำนวณ	ค่าทีวิกฤต	P-value	ผล
10	0.1814	0.0121	0.5287	0.0186	-49.3658	2.1009	1.14E-20	ปฏิเสธ H_0
50	0.6900	0.0787	2.0891	0.0723	-41.4097	2.1009	2.62E-19	ปฏิเสธ H_0
100	1.6044	0.1769	6.1097	0.3068	-40.2315	2.1009	4.39E-19	ปฏิเสธ H_0
500	11.5466	1.9083	215.2506	6.1712	-99.7240	2.2010	1.29E-17	ปฏิเสธ H_0
1000	39.7161	10.5853	1087.525	7.9737	-250.025	2.1009	2.52E-33	ปฏิเสธ H_0

ตารางที่ 9 ผลการแก้ปัญหาเปรียบเทียบความสามารถการแก้ปัญหาของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ1000

n = 1000								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ (ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าทีคำนวณ	ค่าทีวิกฤต	P-value	ผล
10	0.3906	0.0344	1.0705	0.7539	-2.8491	2.2622	1.91E-02	ปฏิเสธ H_0
50	1.3391	0.1896	6.5752	1.6960	-9.7026	2.2622	4.60E-06	ปฏิเสธ H_0
100	3.0384	0.2577	22.0808	0.8533	-67.5579	2.2010	9.27E-16	ปฏิเสธ H_0
500	26.3458	5.1425	1098.279	18.9655	-172.504	2.2281	1.05E-18	ปฏิเสธ H_0
1000	97.4813	30.6998	4842.110	10.5019	-462.419	2.2010	6.07E-25	ปฏิเสธ H_0

ตารางที่ 10 ผลการแก้ปัญหาเปรียบเทียบความสามารถการแก้ปัญหาของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ 5000

n = 5000								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ (ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าที คำนวณ	ค่าที วิกฤต	P-value	ผล
10	3.6682	0.2691	7.1031	0.6323	-15.8078	2.1788	2.13E-09	ปฏิเสธ H_0
50	8.9347	0.9435	216.0088	1.3943	-388.965	2.1009	8.85E-37	ปฏิเสธ H_0
100	15.5343	2.0399	1107.391	8.8008	-382.190	2.2281	3.70E-22	ปฏิเสธ H_0

ตารางที่ 11 ผลการแก้ปัญหาเปรียบเทียบความสามารถการแก้ปัญหาของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ 10000

n = 10000								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ (ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าที คำนวณ	ค่าที วิกฤต	P-value	ผล
10	12.9874	0.4448	25.4006	0.6637	-49.1332	2.1009	1.24E-20	ปฏิเสธ H_0
50	22.1129	1.6992	1124.556	6.8723	-492.456	2.2281	2.93E-23	ปฏิเสธ H_0

ในการทดสอบประสิทธิภาพในการแก้ปัญหาแบบไม่มีขอบเขต ที่ระดับความเชื่อมั่น 95 เปอร์เซ็นต์ จะได้ว่า ค่าที่ที่ได้จากการคำนวณส่วนใหญ่มีค่าอยู่ในบริเวณวิกฤต ทำให้สามารถปฏิเสธสมมติฐานหลักได้ จึงสรุปได้ว่าค่าเฉลี่ยของเวลาที่ใช้ในการแก้ปัญหาในแต่ละวิธีไม่เท่ากัน หรือประสิทธิภาพของแต่ละวิธีในการหาคำตอบมีค่าไม่เท่ากัน และมีเพียง 3 ขนาดปัญหาที่ทดสอบแล้วสรุปค่าเฉลี่ยที่ได้มีค่าเท่ากัน หรือหมายความว่ามีใช้เวลาในการแก้ปัญหาไม่แตกต่างกัน แต่จากการเปรียบเทียบค่าเฉลี่ยในเบื้องต้น จะได้ว่าวิธีตรงใช้เวลาแก้ปัญหาน้อยกว่าวิธีคำสั่ง linprog ทุกกรณี ดังนั้นจากการเปรียบเทียบทางสถิติในการเปรียบเทียบความสามารถในการแก้ปัญหาของสองวิธีจึงสรุปได้ว่าวิธีตรงให้ประสิทธิภาพในการแก้ปัญหาดีกว่าวิธีคำสั่ง linprog ในทุกกรณีด้วยเช่นกัน

4. <u>ผลเปรียบเทียบเวลาในการแก้ปัญหาของปัญหาเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขตทางสถิติ (Two-Stage Single Constrained Linear Fractional Programming With Bounded Variables)</u>

จากการศึกษาประสิทธิภาพในการใช้เวลาคำนวณเพื่อแก้ปัญหาแบบมีขอบเขตสองวิธีทางสถิติ รายละเอียดในตารางผนวก ก23 โดยกำหนด ค่าสมมติฐานหลัก คือ ความสามารถในการแก้ปัญหาวิธีคำสั่ง linprog และวิธีตรงเท่ากัน $\left(H_0 : \mu_L = \mu_T\right)$ สมมติฐานรองคือ ความสามารถในการแก้ปัญหาสองวิธีไม่เท่ากัน $\left(H_1 : \mu_L \neq \mu_T\right)$ ที่ 95 เปอร์เซ็นต์ระดับความเชื่อมั่น $(\alpha = 0.05)$ ทำการทดสอบสรุปได้ดังนี้

<u>ตารางที่ 12</u> ผลการแก้ปัญหาเปรียบเทียบความสามารถการแก้ปัญหาแบบมีขอบเขตของสองวิธี เมื่อจำนวนตัวแปรเท่ากับ 10

n = 10								
	วิธีตรง (2Stagefrac)		วิธี Linprog		ค่าสถิติ (ความเชื่อมั่น=0.05)			
m =	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าเฉลี่ย	ค่าส่วนเบียงเบน	ค่าที คำนวณ	ค่าที วิกฤต	P-value	ผล
10	0.301	0.114	0.512	0.031	-5.64273	2.2281	0.0002	ปฏิเสธ H_0
50	0.410	0.152	0.538	0.037	-2.59242	2.2281	0.0268	ปฏิเสธ H_0
100	0.513	0.293	0.704	0.045	-2.03629	2.2622	0.0722	ปฏิเสธ H_0 ไม่ได้
500	1.858	0.940	1.290	0.139	1.88904	2.2622	0.0915	ปฏิเสธ H_0 ไม่ได้
1000	6.119	2.109	3.424	1.953	2.964856	2.1009	0.0083	ปฏิเสธ H_0
5000	190.070	79.815	36.931	3.305	6.062171	2.2622	0.0002	ปฏิเสธ H_0
10000	755.099	297.625	80.885	16.115	7.1531	2.2622	0.0001	ปฏิเสธ H_0

การทดสอบประสิทธิภาพโดยค่า n เท่ากับ 10 จะได้การเปรียบเทียบ เมื่อ m เท่ากับ 10 และ 50 วิธีตรงใช้เวลาน้อยกว่า มีประสิทธิภาพในการแก้ปัญหาดีกว่า แต่เมื่อ m เท่ากับ 100 และ 500 เวลาที่ใช้ของสองวิธีไม่แตกต่างกัน และ m เท่ากับ 100 ถึง 10000 ความสามารถในการแก้ปัญหาของแต่ละวิธีไม่เท่ากันโดยวิธีคำสั่ง linprog มีประสิทธิภาพมากกว่า และเมื่อขนาดของปัญหามีขนาดใหญ่ขึ้นวิธีคำสั่ง linprog จะมีประสิทธิภาพในการแก้ปัญหาดีกว่าวิธีตรงทุกกรณี

สรุป

การแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน โดยกำหนดให้ค่าสัมประสิทธิ์ทุกตัวมีค่าเป็นบวก มีตัวแปรตัดสินใจคือ การเลือกสินค้าหรือสิ่งของ (x_j) และค่าด้านขวาที่เปลี่ยนแปลงได้ โดยมีวัตถุประสงค์เพื่อหาค่าที่เหมาะสมที่ต่ำสุดของอัตราส่วนในการเลือกสินค้าหรือสิ่งของนั้น ๆ ลักษณะในการแก้ปัญหานี้เป็นการแก้ปัญหาเฉพาะ โดยงานวิจัยนี้ได้ทำการศึกษาการแก้ปัญหาที่ไม่สามารถแก้ได้อย่างรวดเร็วเมื่อปัญหามีขนาดใหญ่ขึ้น เพื่อให้ได้คำตอบที่มีความรวดเร็วแม่นยำมากยิ่งขึ้นในการหาค่าผลเฉลยที่ดีที่สุด โดยใช้วิธีคำนวณแก้ปัญหาทางคณิตศาสตร์ พัฒนาเป็นโปรแกรมในการหาค่าผลเฉลย ผลจากการทดลองนี้จะเห็นได้ว่าวิธีการคำนวณหาค่าผลเฉลยแล้วนำมาพัฒนาสร้างเป็นโปรแกรมในการหาค่าผลเฉลย ซึ่งเรียกว่าเป็นวิธีตรงกับปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน แบบไม่มีการกำหนดขอบเขตของตัวแปร สามารถทำการหาค่าได้อย่างมีประสิทธิภาพมากกว่าการใช้วิธีเชิงเส้น หรือวิธีการใช้คำสั่ง linprog ในการคำนวณหาค่าผลเฉลย ถึงแม้ว่าปัญหาจะมีขนาดใหญ่ ประสิทธิภาพในการหาค่าผลเฉลยของวิธีตรงก็ยังคงดีกว่า ซึ่งเห็นได้จากเวลาในการหาค่าผลเฉลยของวิธีตรงที่น้อยกว่าวิธีการแก้ปัญหาแบบเชิงเส้น แต่ในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน แบบมีขอบเขต จะเห็นได้ว่าค่าขอบเขตจะทำให้เงื่อนไขเพิ่มขึ้น และมีความซับซ้อนในการคำนวณ ซึ่งมีผลต่อการคำนวณอย่างมาก ในการพัฒนาวิธีการหาค่าผลเฉลยจากการแก้ปัญหาโดยวิธีทางคณิตศาสตร์แล้วยังต้องใช้หลักการตรวจสอบคำตอบที่สามารถเป็นไปได้ทุกกรณี สำหรับทุกค่าคำตอบที่เป็นไปได้ เนื่องจากไม่สามารถจัดเรียงลำดับการเลือกสิ่งของ (x_j) ได้เหมือนกับกรณีปัญหากำหนดการเชิงเส้นสองขั้นตอนแบบมีเงื่อนไข ทำให้ไม่สามารถหาค่าผลเฉลยที่ดีออกมาได้ในเวลาที่จำกัด จึงทำให้วิธีการแก้ปัญหาแบบเชิงเส้นมีประสิทธิภาพดีกว่า

จากผลการทดลองซอฟต์แวร์สำเร็จ MATLAB ในแต่ละวิธี เวลาที่ใช้ในการแก้ปัญหาขึ้นอยู่กับขนาด หรือจำนวนของการเลือกตัวแปร และ เงื่อนไขในแต่ละครั้ง ดังนั้นทำให้ไม่สามารถประมาณเวลาที่แน่นอน ในการแก้ปัญหาขนาดต่างๆ เมื่อปัญหามีขนาดใหญ่ขึ้นเวลาที่ใช้จะมากขึ้นตามด้วย เมื่อทำการสร้างกราฟแนวโน้มจะได้กราฟที่มีลักษณะเป็นฟังก์ชันพหุนาม และฟังก์ชันเลขชี้กำลัง โดยมีแนวโน้มที่เพิ่มขึ้นตามขนาดของปัญหา แต่จากผลการทดลอง ทำให้รู้ว่าในกรณีปัญหาแบบไม่มีขอบเขต วิธีการแก้ปัญหาโดยตรงจะสามารถแก้ปัญหาได้เร็วกว่า ซึ่งแสดงถึงประสิทธิภาพและประสิทธิผลในการแก้ปัญหาที่ดีกว่าทุกกรณี เมื่อทำการเปรียบเทียบเวลาเฉลี่ยที่ใช้ในการแก้ปัญหา หรือทำการทดสอบด้านสถิติในระยะเวลาที่จำกัด แต่ในกรณีปัญหาแบบมี

ขอบเขต วิธีการแก้ปัญหาเชิงเส้นยังคงดีกว่าเพิ่มขนาดของปัญหาใหญ่และซับซ้อนมากขึ้น เนื่องจากวิธีตรงที่ใช้การคำนวณแล้วนำวิธีคิดนี้ไปพัฒนาในการหาค่าผลเฉลยนั้น ยังไม่สามารถจัดเรียงค่าสำหรับการหาได้ ซึ่งทำให้การหาคำตอบนั้นล่าช้า และต้องคำนวณทุกคำตอบที่จะสามารถเป็นไปได้ซึ่งส่งผลต่อหน่วยความจำในการคำนวณ

จากการศึกษา จะเห็นได้ว่าในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนสามารถนำวิธีที่ได้พัฒนาขึ้น หรือวิธีตรง ไปใช้ในการแก้ปัญหาที่มีลักษณะเฉพาะ ไม่ว่าจะเป็นภาคอุตสาหกรรม การเงิน หรือการบริหารในส่วนต่าง ๆ เพื่อให้เกิดความสะดวกรวดเร็วและแม่นยำ และยังแก้ปัญหาที่มีขนาดใหญ่ได้ดีกว่า ส่วนการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน แบบมีขอบเขตนั้น จะเห็นได้ว่าวิธีแก้ปัญหาแบบเชิงเส้นยังสามารถแก้ปัญหาได้ดีอยู่ แต่ถ้าสามารถพัฒนาการแก้ปัญหาแบบวิธีตรงในส่วนการจัดเรียงค่าในการแทนคำตอบที่เหมาะสมจะทำให้การหาค่าผลเฉลย มีความรวดเร็ว แม่นยำ และมีประสิทธิภาพที่ดียิ่งขึ้น

เอกสารและสิ่งอ้างอิง

Ahuja, R.K., T.L. Magnanti and J.B. Orlin. 1993. **Network Flows**. Prentice Hall, Inc., New Jersey.

Billionnet, A. 2002. Approximation algorithms for fractional knapsack problems. **Operations Research Letters** vol. 30 (2002): 336-342.

Birge, J. and D. Holmes. 1992. Efficient solution of two-stage stochastic linear programs using interior point methods. **Computational Optimization and Applications** vol. 1 (1992): 245-276.

Birge, J.R. and F. Louveaux. 1997. **Introduction to Stochastic Programming.** Springer-Verlag New York, Inc., New York.

Bazaraa, M.S., H.D. Sherali and C.M. Shetty. 1979. **Nonlinear Programming: Theory and Algorithms**. 2nd edition. John Wiley & Sons, Inc, USA.

Benati, S. 2004. The computation of the worst conditional expectation. **European Journal Operational Research** vol. 155 (2004): 414-425.

Calvete, H.I. and C. Gale. 1999. Theory and Methodology: The bilevel linear/linear fractional programming problem. **European Journal Operational Research** vol. 114 (1999): 188-197.

Chakraborty, M. and S. Gupta. 2002. Fuzzy mathematical programming for multi objective linear fractional programming problem. **Fuzzy Sets and Systems** vol. 125 (2002) 335-342.

Chang, C. 2002. Continuous Optimization: On the posynomial fractional programming problems. **European Journal Operational Research** vol. 143 (2002) 42-52.

Charnes, A., W.W. Cooper and E. Rohdes. 1962. Programming with linear fractional functional. **Nav. Research Logistics Quarterly** vol. 9 (1962): 181-186.

Craven, B.D. 1988. **Fractional Programming**. Heldermann Verlag Berlin.

Dantzig, G.B. 1963. **Linear Programming and Extensions**. Princeton University Press, Princeton, New Jersey.

Gilmore, P.C. and R.E. Gomory. 1963. A linear programming approach to the cutting stock problem-II. **Operations Research** vol. 11 (1963): 863-888.

Hillier, F.S. and G. J. Lieberman. 1990. **Introduction to Operations Research**. 5th edition. McGraw-Hill, New York, USA.

Luhandjula, M.K. 1984. Fuzzy approaches for multiple objective linear fractional optimizations. **Fuzzy Sets and Systems** vol. 13 (1984): 11-23.

Lustig, I.J., J.M. Mulvey and T.J. Carperter. 1991. Formulating two-stage stochastic programs for interior point methods. **Operations Research** vol. 39/5 (1991):757-770.

Meszaros, C.-T. 1997. The augmented system variant of IPMs in two-stage stochastic linear programming computation. **European Journal Operational Research** vol. 101 (1997): 317-327.

Nash, S.G. and A. Sofer. 1996. **Linear and Nonlinear Programming**. International editions. McGraw-Hill.

Nemhauser, G.L., A.H.G. Rinnooy Kan and M.J. Todd. 1989. **Handbook in Operations Research and Management Science: Vol.1**. Elsevier Science B.V, Netherlands.

Sakawa, M. and I. Nishizaki. 2001. Interactive fuzzy programming for two-level linear fractional programming problems. **Fuzzy Sets and Systems** vol. 119 (2001): 31-40.

Simm, A.E. 1997. **A Stochastic Approach to Modeling Aviation Security Problems Using the Knapsack Problem**. M.S. thesis, Virginia Polytechnic Institute and State University, Virginia.

Wagner, H.M. 1975. **Principle of Operations Research**. 2nd edition. Pentice-Hall, Inc., Englewood cliffs, New Jersey, USA.

Witchakul, S., P. Sudasna-na-Ayudthya and P. Charnsethikul. 2004. A Fast Method for Two-Stage Bounded Variables and Single Constrained Linear Programming. **การประชุมวิชาการด้านการวิจัยดำเนินงาน ประจำปี พ.ศ. 254**7: 237-245.

ภาคผนวก

ภาคผนวก ก

ตารางผนวกแสดงรายละเอียดข้อมูลต่าง ๆ ในการทดลองการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน

ตารางผนวกที่ ก1 การปรับค่าสัมประสิทธิ์ $(c_j : d_j)$ ในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อใช้คำสั่ง linprog

	m =				
n	10	50	100	500	1000
10	1 : 10	1 : 100	1 :1000	1 : 10000	1 : 100000
50	1 : 10	1 : 10	1 : 100	1 : 1000	1 : 20000
100	10 : 10	1 : 10	1 : 100	1 : 1000	1 : 10000

ตารางผนวกที่ ก2 การปรับค่าสัมประสิทธิ์ $(c_j : d_j)$ ที่เหมาะสมกับการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน แบบมีขอบเขต เมื่อใช้คำสั่ง linprog

	m =					
n	10	50	100	500	1000	5000
10	10 : 100	10 : 400	10 : 1000	10 : 10000	10 : 100000	10 : 1000000
50	10 : 100	10 : 400	10 : 1000	10 : 8000	10 : 20000	10 : 200000
100	10 : 100	10 : 100	10 : 400	10 : 4000	10 : 10000	10 : 100000

ตารางผนวกที่ ก3 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อกำหนด n = 10 (หน่วย : วินาที)

	n = 10 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	0.050	0.350	0.040	0.370	0.070	0.480
2	0.040	0.360	0.050	0.401	0.080	0.481
3	0.050	0.400	0.070	0.401	0.080	0.471
4	0.040	0.360	0.070	0.380	0.070	0.471
5	0.040	0.360	0.050	0.410	0.070	0.461
6	0.040	0.361	0.060	0.381	0.080	0.451
7	0.030	0.350	0.070	0.400	0.090	0.461
8	0.030	0.360	0.040	0.391	0.070	0.470
9	0.030	0.351	0.050	0.391	0.060	0.561
10	0.030	0.351	0.070	0.381	0.080	0.470
average	0.0380	0.3603	0.0570	0.3906	0.0750	0.4777
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	0.301	0.711	0.581	1.232	12.247	14.47
2	0.230	0.711	0.911	1.242	6.219	14.001
3	0.291	0.721	0.450	1.232	16.033	13.749
4	0.230	0.721	0.941	1.242	14.030	16.574
5	0.420	0.712	0.992	1.261	6.800	13.88
6	0.231	0.741	0.321	1.221	7.241	14.34
7	0.480	0.732	0.861	1.202	9.885	20.66
8	0.340	0.741	0.931	1.242	9.786	16.503
9	0.260	0.741	0.470	1.272	12.480	14.741
10	0.250	0.711	0.871	1.292	14.771	17.105
average	0.3033	0.7242	0.7329	1.2438	10.9492	15.6023

<u>ตารางผนวกที่ ก3</u> (ต่อ)

no.	n = 10 , m = 10000 2stagefrac	10000 linprog	20000 2stagefrac	20000 linprog	30000 2stagefrac	30000 linprog
1	16.724	37.003	256.428	190.904	323.375	522.652
2	23.744	37.634	272.552	189.403	341.481	510.434
3	25.446	37.915	123.608	192.106	239.825	519.868
4	43.162	36.953	256.519	187.339	537.863	600.454
5	30.063	37.234	181.791	187.089	355.841	511.095
6	39.767	37.714	99.903	186.528	504.155	134.814
7	32.938	37.944	67.397	189.502	515.371	342.342
8	29.683	38.145	225.685	190.013	517.254	624.037
9	39.477	37.003	22.322	190.164	515.712	459.080
10	30.203	36.483	262.938	186.078	566.375	599.592
average	31.1207	37.4028	176.9143	188.9126	441.7252	482.4368

no.	50000 2stagefrac	50000 linprog
1	1806.527	1776.634
2	1931.117	1762.264
3	1192.575	1726.062
4	2119.987	1721.806
5	295.786	1722.527
6	1195.069	1724.119
7	1491.605	1718.230
8	1660.417	1724.069
9	448.795	1761.402
10	670.484	1726.632
average	1281.2362	1736.3745

<u>ตารางผนวกที่ ก4</u> เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อกำหนด n = 50 (หน่วย : วินาที)

	n = 50 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	0.040	0.391	0.110	0.450	0.190	0.631
2	0.060	0.371	0.110	0.481	0.190	0.661
3	0.060	0.361	0.101	0.460	0.181	0.651
4	0.070	0.360	0.120	0.431	0.170	0.661
5	0.061	0.380	0.130	0.410	0.230	0.611
6	0.050	0.370	0.110	0.431	0.140	0.611
7	0.050	0.371	0.120	0.411	0.200	0.591
8	0.070	0.421	0.120	0.441	0.190	0.631
9	0.070	0.361	0.120	0.451	0.170	0.691
10	0.070	0.380	0.131	0.420	0.201	0.630
average	0.0601	0.3766	0.1172	0.4386	0.1862	0.6369
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	2.023	2.834	3.115	7.110	39.968	219.876
2	1.913	2.293	2.213	6.400	36.156	218.785
3	1.602	2.413	2.694	6.619	52.315	220.407
4	1.362	2.453	3.986	6.599	32.557	222.780
5	1.573	2.473	5.067	6.700	40.818	225.915
6	1.132	2.453	5.608	6.860	60.948	219.846
7	1.462	2.584	3.074	7.030	204.474	219.595
8	1.171	2.584	3.956	6.740	33.989	218.624
9	1.622	2.484	4.145	3.956	63.281	219.516
10	1.392	2.484	4.286	6.669	93.254	219.115
average	1.5252	2.5055	3.8144	6.4683	65.7760	220.4459

ตารางผนวกที่ ก4 (ต่อ)

	n = 50 , m =					
	10000		20000		30000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	225.284	1147.620	1029.701	>5000	985.457	n/a
2	386.613	1147.901	1324.474	>5000	1955.542	n/a
3	229.971	1149.814	497.914	>5000	2896.295	n/a
4	503.454	1131.046	1173.077	>5000	965.799	n/a
5	243.450	1132.829	1545.652	>5000	1902.486	n/a
6	364.664	1139.678	264.871	>5000	2798.144	n/a
7	471.078	1135.002	1211.923	>5000	2586.789	n/a
8	108.075	1131.628	3567.800	>5000	3408.113	n/a
9	580.678	1131.717	3032.17	>5000	3658.560	n/a
10	157.786	1131.618	841.781	>5000	1837.892	n/a
average	327.1053	1137.8853	1448.9363	>5000	2299.5077	n/a

	50000	
no.	2stagefrac	linprog
1	>5000	n/a
2	>5000	n/a
3	>5000	n/a
4	>5000	n/a
5	>5000	n/a
6	>5000	n/a
7	>5000	n/a
8	>5000	n/a
9	>5000	n/a
10	>5000	n/a
average	>5000	n/a

ตารางผนวกที่ ก5 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อกำหนด n = 100 (หน่วย : วินาที)

	n = 100 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	0.080	0.531	0.190	0.521	0.401	1.061
2	0.080	0.390	0.201	0.510	0.341	0.871
3	0.060	0.390	0.171	0.561	0.360	0.912
4	0.070	0.381	0.190	0.501	0.350	0.792
5	0.060	0.401	0.180	0.521	0.240	0.882
6	0.060	0.391	0.200	0.511	0.360	0.851
7	0.071	0.390	0.160	0.561	0.301	0.871
8	0.080	0.380	0.170	0.501	0.320	0.941
9	0.060	0.391	0.180	0.531	0.360	0.831
10	0.080	0.370	0.180	0.521	0.300	0.801
average	0.0701	0.4015	0.1822	0.5239	0.3333	0.8813
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	2.945	6.900	8.763	22.342	114.304	1103.798
2	2.784	6.389	9.644	21.611	134.354	1104.037
3	4.126	6.889	8.131	22.142	86.625	1105.149
4	3.355	6.189	8.482	21.441	90.620	1099.331
5	2.013	7.241	7.671	21.241	93.454	1099.732
6	3.105	6.399	5.678	21.591	146.220	1103.367
7	3.305	6.549	6.971	21.721	230.682	1104.739
8	3.105	6.269	7.881	21.190	73.806	1097.839
9	2.854	6.459	5.608	21.608	179.097	1098.139
10	3.104	6.520	6.049	21.351	239.965	1101.885
average	3.0696	6.5804	7.4878	21.6238	138.9127	1101.8016

ตารางผนวกที่ ก5 (ต่อ)

	n = 100 , m =					
	10000		20000		30000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	870.051	4983.295	2103.995	>5000	>5000	n/a
2	1638.456	4943.418	1392.953	>5000	>5000	n/a
3	1680.627	4928.407	3579.126	>5000	>5000	n/a
4	1642.471	4980.221	1120.972	>5000	>5000	n/a
5	1736.998	4974.593	1669.490	>5000	>5000	n/a
6	1775.002	5006.569	4856.123	>5000	>5000	n/a
7	1707.615	4943.459	1267.724	>5000	>5000	n/a
8	644.307	4948.315	832.377	>5000	>5000	n/a
9	919.051	4968.124	1409.818	>5000	>5000	n/a
10	1714.585	4938.952	3388.101	>5000	>5000	n/a
average	1432.9163	4961.5353	2162.0679	>5000	>5000	n/a

ตารางผนวกที่ ก6 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อกำหนด n = 500 (หน่วย : วินาที)

	n = 500 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	0.160	0.521	0.711	2.093	1.412	5.878
2	0.181	0.510	0.631	2.063	1.392	5.738
3	0.191	0.540	0.541	2.113	1.682	6.700
4	0.170	0.511	0.721	2.133	1.352	5.959
5	0.190	0.511	0.721	2.183	1.883	5.869
6	0.180	0.541	0.681	2.093	1.653	6.449
7	0.180	0.511	0.701	2.194	1.783	5.988
8	0.190	0.561	0.851	1.993	1.543	6.169
9	0.171	0.550	0.691	2.053	1.622	5.968
10	0.201	0.531	0.651	1.973	1.722	6.379
average	0.1814	0.5287	0.6900	2.0891	1.6044	6.1097
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	12.899	209.211	45.846	1088.916	487.161	>5000
2	12.959	216.591	35.952	1098.529	876.350	>5000
3	13.339	216.922	60.376	1101.034	412.684	>5000
4	9.283	206.768	36.672	1095.245	536.842	>5000
5	11.707	224.543	39.567	1081.085	517.965	>5000
6	12.828	213.017	48.360	1079.892	644.727	>5000
7	12.208	212.485	22.452	1084.980	925.651	>5000
8	12.799	208.970	44.354	1080.293	801.062	>5000
9	8.021	221.679	32.377	1082.006	371.244	>5000
10	9.423	222.320	31.205	1083.267	711.443	>5000
average	11.5466	215.2506	39.7161	1087.5247	628.5129	>5000

ตารางผนวกที่ ก6 (ต่อ)

	n = 500 , m =					
	10000		20000		30000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	2040.885	n/a	>5000	n/a	n/a	n/a
2	3097.885	n/a	>5000	n/a	n/a	n/a
3	3646.754	n/a	>5000	n/a	n/a	n/a
4	2157.212	n/a	>5000	n/a	n/a	n/a
5	5820.259	n/a	>5000	n/a	n/a	n/a
6	2405.869	n/a	>5000	n/a	n/a	n/a
7	2088.763	n/a	>5000	n/a	n/a	n/a
8	3538.348	n/a	>5000	n/a	n/a	n/a
9	1784.776	n/a	>5000	n/a	n/a	n/a
10	4099.584	n/a	>5000	n/a	n/a	n/a
average	3068.0335	n/a	>5000	n/a	n/a	n/a

<u>ตารางผนวกที่ ก7</u> เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อกำหนด n = 1000 (หน่วย : วินาที)

	n = 1000 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	0.391	3.204	1.683	11.386	3.375	24.305
2	0.410	0.822	1.442	6.299	3.304	22.463
3	0.461	0.811	1.282	5.848	2.764	21.131
4	0.380	0.791	1.402	5.878	2.914	21.962
5	0.330	1.052	1.192	6.028	3.145	21.941
6	0.361	0.821	1.112	5.938	3.014	21.651
7	0.381	0.781	1.152	5.979	3.355	22.011
8	0.401	0.771	1.472	6.099	2.644	21.631
9	0.410	0.821	1.152	6.128	2.805	21.731
10	0.381	0.831	1.502	6.169	3.064	21.982
average	0.3906	1.0705	1.3391	6.5752	3.0384	22.0808
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	37.754	1096.337	74.147	4827.361	876.931	>5000
2	23.955	1093.973	73.886	4830.606	906.974	>5000
3	28.000	1092.541	74.357	4851.436	553.025	>5000
4	24.976	1104.678	72.314	4842.884	1088.205	>5000
5	21.991	1140.330	85.453	4847.410	1051.853	>5000
6	19.819	1082.046	134.794	4862.923	724.872	>5000
7	27.920	1120.341	132.330	4841.471	746.904	>5000
8	29.682	1086.342	133.473	4843.044	1851.753	>5000
9	21.781	1086.212	64.122	4832.509	646.610	>5000
10	27.580	1079.993	129.937	4841.452	975.813	>5000
average	26.3458	1098.2793	97.4813	4842.1096	942.2940	>5000

ตารางผนวกที่ ก7 (ต่อ)

	n = 1000 , m =					
	10000		20000		30000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	>5000	n/a	n/a	n/a	n/a	n/a
2	>5000	n/a	n/a	n/a	n/a	n/a
3	>5000	n/a	n/a	n/a	n/a	n/a
4	>5000	n/a	n/a	n/a	n/a	n/a
5	>5000	n/a	n/a	n/a	n/a	n/a
6	>5000	n/a	n/a	n/a	n/a	n/a
7	>5000	n/a	n/a	n/a	n/a	n/a
8	>5000	n/a	n/a	n/a	n/a	n/a
9	>5000	n/a	n/a	n/a	n/a	n/a
10	>5000	n/a	n/a	n/a	n/a	n/a
average	>5000	n/a	n/a	n/a	n/a	n/a

ตารางผนวกที่ ก8 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อกำหนด n = 5000 (หน่วย : วินาที)

	n = 5000 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	3.465	8.843	8.722	214.629	17.485	1123.266
2	4.036	7.170	7.771	215.299	16.354	1109.926
3	3.836	6.900	10.194	215.340	14.861	1107.833
4	3.374	7.061	9.393	214.679	17.505	1118.589
5	3.225	6.789	8.572	215.851	11.507	1112.319
6	3.735	6.960	8.081	215.380	15.612	1100.743
7	3.745	6.860	7.661	218.294	12.899	1099.271
8	3.655	7.030	10.345	215.110	14.881	1096.156
9	3.565	6.589	9.494	218.164	17.035	1105.360
10	4.046	6.829	9.114	217.342	17.204	1100.442
average	3.6682	7.1031	8.9347	216.0088	15.5343	1107.3905
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	115.596	>5000	539.697	n/a	>5000	n/a
2	121.115	>5000	430.620	n/a	>5000	n/a
3	181.491	>5000	481.332	n/a	>5000	n/a
4	110.359	>5000	436.037	n/a	>5000	n/a
5	174.801	>5000	585.232	n/a	>5000	n/a
6	134.183	>5000	410.861	n/a	>5000	n/a
7	161.141	>5000	468.765	n/a	>5000	n/a
8	90.339	>5000	412.653	n/a	>5000	n/a
9	133.272	>5000	310.046	n/a	>5000	n/a
10	138.049	>5000	671.095	n/a	>5000	n/a
average	136.0346	>5000	474.6338	n/a	>5000	n/a

ตารางผนวกที่ ก9 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน เมื่อกำหนด n = 10000 (หน่วย : วินาที)

	n = 10000 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	13.128	27.169	21.230	1126.390	35.371	>5000
2	12.728	25.086	20.910	1128.593	36.583	>5000
3	12.808	25.137	21.201	1128.613	40.609	>5000
4	12.508	25.196	24.506	1129.364	44.454	>5000
5	13.830	24.935	21.882	1128.082	38.685	>5000
6	12.919	24.806	19.658	1119.390	44.924	>5000
7	12.317	25.306	25.146	1114.853	40.549	>5000
8	13.199	25.427	23.324	1118.428	44.744	>5000
9	13.459	25.607	21.641	1116.145	42.201	>5000
10	12.978	25.337	21.631	1135.703	38.596	>5000
average	12.9874	25.4006	22.1129	1124.5561	40.6716	>5000
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	181.000	n/a	616.236	n/a	>5000	n/a
2	385.554	n/a	871.142	n/a	>5000	n/a
3	234.588	n/a	871.974	n/a	>5000	n/a
4	273.874	n/a	799.590	n/a	>5000	n/a
5	293.071	n/a	1053.936	n/a	>5000	n/a
6	306.110	n/a	740.665	n/a	>5000	n/a
7	158.909	n/a	1018.064	n/a	>5000	n/a
8	262.587	n/a	936.727	n/a	>5000	n/a
9	259.062	n/a	739.433	n/a	>5000	n/a
10	279.202	n/a	860.537	n/a	>5000	n/a
average	263.3957	n/a	850.8304	n/a	>5000	n/a

ตารางผนวกที่ ก10 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อกำหนด n = 10 (หน่วย : วินาที)

	n = 10 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	0.241	0.480	0.310	0.521	0.110	0.681
2	0.281	0.510	0.361	0.591	0.852	0.701
3	0.501	0.501	0.341	0.501	0.481	0.741
4	0.271	0.500	0.391	0.521	0.951	0.721
5	0.280	0.571	0.401	0.570	0.471	0.811
6	0.160	0.501	0.230	0.481	0.461	0.701
7	0.511	0.501	0.641	0.521	0.431	0.681
8	0.251	0.480	0.351	0.591	0.872	0.660
9	0.280	0.561	0.721	0.551	0.140	0.681
10	0.231	0.510	0.351	0.530	0.361	0.661
average	0.3007	0.5115	0.4098	0.5378	0.5130	0.7039
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	1.302	1.342	8.291	2.474	203.623	38.205
2	2.273	1.452	9.664	2.303	169.734	31.816
3	3.475	1.262	7.200	2.073	166.490	34.720
4	1.292	1.062	4.777	2.133	245.222	38.866
5	1.743	1.502	4.897	2.113	173.189	42.010
6	1.041	1.302	2.644	6.560	161.682	39.968
7	2.554	1.392	5.708	4.056	161.592	33.829
8	1.362	1.201	5.168	2.834	34.950	35.201
9	0.520	1.122	4.847	2.363	235.759	34.709
10	3.014	1.262	7.992	7.330	348.461	39.987
average	1.8576	1.2899	6.1188	3.4239	190.0702	36.9311

<u>ตารางผนวกที่ ก10</u>(ต่อ)

	n = 10 , m =							
	10000		20000		30000		50000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	1158.556	86.385	>5000	202.281	n/a	533.047	n/a	1755.975
2	582.798	100.535	>5000	196.042	n/a	525.436	n/a	1748.504
3	643.565	84.572	>5000	198.816	n/a	506.418	n/a	1735.365
4	157.256	92.853	>5000	199.387	n/a	551.503	n/a	1774.972
5	904.390	58.634	>5000	196.263	n/a	537.302	n/a	1762.093
6	606.612	99.894	>5000	204.654	n/a	564.392	n/a	1776.764
7	589.007	82.479	>5000	203.873	n/a	526.387	n/a	1748.624
8	848.069	57.523	>5000	334.140	n/a	533.206	n/a	1750.117
9	1067.745	83.811	>5000	204.183	n/a	723.660	n/a	1743.026
10	992.988	62.160	>5000	303.016	n/a	534.749	n/a	1788.542
average	755.0986	80.8846	>5000	224.2655	n/a	553.6100	n/a	1758.3982

ตารางผนวกที่ ก11 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อกำหนด n = 50 (หน่วย : วินาที)

	n = 50 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	0.581	n/a	0.701	n/a	1.082
2	n/a	0.651	n/a	0.961	n/a	1.082
3	n/a	0.651	n/a	0.691	n/a	0.981
4	n/a	0.701	n/a	0.530	n/a	0.981
5	n/a	0.631	n/a	0.661	n/a	1.081
6	n/a	0.601	n/a	0.711	n/a	1.122
7	n/a	0.611	n/a	0.661	n/a	0.851
8	n/a	0.591	n/a	0.651	n/a	1.041
9	n/a	0.621	n/a	0.761	n/a	0.982
10	n/a	0.630	n/a	0.691	n/a	1.232
average	n/a	0.6269	n/a	0.7019	n/a	1.0435
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	3.906	n/a	10.195	n/a	288.124
2	n/a	3.665	n/a	19.398	n/a	282.286
3	n/a	3.284	n/a	8.773	n/a	256.599
4	n/a	3.055	n/a	16.204	n/a	283.207
5	n/a	3.425	n/a	17.235	n/a	288.125
6	n/a	3.986	n/a	8.963	n/a	282.486
7	n/a	2.954	n/a	12.248	n/a	287.483
8	n/a	4.056	n/a	12.909	n/a	291.359
9	n/a	3.725	n/a	9.233	n/a	279.802
10	n/a	3.625	n/a	9.033	n/a	261.376
average	n/a	3.5681	n/a	12.4191	n/a	280.0847

ตารางผนวกที่ ก11(ต่อ)

	n = 50 , m =					
	10000		20000		30000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	1288.162	n/a	>5000	n/a	n/a
2	n/a	1277.477	n/a	>5000	n/a	n/a
3	n/a	1268.504	n/a	>5000	n/a	n/a
4	n/a	1174.358	n/a	>5000	n/a	n/a
5	n/a	1231.450	n/a	>5000	n/a	n/a
6	n/a	1283.014	n/a	>5000	n/a	n/a
7	n/a	1271.548	n/a	>5000	n/a	n/a
8	n/a	1178.154	n/a	>5000	n/a	n/a
9	n/a	1232.542	n/a	>5000	n/a	n/a
10	n/a	1218.973	n/a	>5000	n/a	n/a
average	n/a	1242.4182	n/a	>5000	n/a	n/a

ตารางผนวกที่ ก12 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน แบบมีขอบเขต เมื่อกำหนด n = 100 (หน่วย : วินาที)

	n = 100 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	0.761	n/a	1.201	n/a	1.863
2	n/a	0.691	n/a	1.192	n/a	1.522
3	n/a	0.841	n/a	1.162	n/a	1.943
4	n/a	0.821	n/a	1.052	n/a	1.462
5	n/a	0.761	n/a	1.092	n/a	1.913
6	n/a	0.671	n/a	1.012	n/a	1.833
7	n/a	0.731	n/a	0.991	n/a	1.762
8	n/a	0.661	n/a	0.982	n/a	1.803
9	n/a	0.691	n/a	1.152	n/a	1.632
10	n/a	0.651	n/a	1.292	n/a	1.612
average	n/a	0.7280	n/a	1.1128	n/a	1.7345
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	8.783	n/a	38.074	n/a	1273.671
2	n/a	8.442	n/a	26.839	n/a	1224.600
3	n/a	8.823	n/a	29.152	n/a	1100.923
4	n/a	9.714	n/a	30.413	n/a	1145.177
5	n/a	8.772	n/a	36.002	n/a	1211.221
6	n/a	10.235	n/a	25.526	n/a	1234.635
7	n/a	7.430	n/a	28.861	n/a	1230.138
8	n/a	8.522	n/a	28.671	n/a	1178.835
9	n/a	12.418	n/a	25.446	n/a	1213.044
10	n/a	8.452	n/a	36.793	n/a	1119.300
average	n/a	9.1591	n/a	30.5777	n/a	1193.1544

<u>ตารางผนวกที่ ก12</u>(ต่อ)

	n = 100 , m =					
	10000		20000		30000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	>5000	n/a	n/a	n/a	n/a
2	n/a	>5000	n/a	n/a	n/a	n/a
3	n/a	>5000	n/a	n/a	n/a	n/a
4	n/a	>5000	n/a	n/a	n/a	n/a
5	n/a	>5000	n/a	n/a	n/a	n/a
6	n/a	>5000	n/a	n/a	n/a	n/a
7	n/a	>5000	n/a	n/a	n/a	n/a
8	n/a	>5000	n/a	n/a	n/a	n/a
9	n/a	>5000	n/a	n/a	n/a	n/a
10	n/a	>5000	n/a	n/a	n/a	n/a
average	n/a	>5000	n/a	n/a	n/a	n/a

ตารางผนวกที่ ก13 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อกำหนด n = 500 (หน่วย : วินาที)

	n = 500 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	3.495	n/a	8.532	n/a	14.271
2	n/a	3.615	n/a	8.582	n/a	15.011
3	n/a	3.675	n/a	8.943	n/a	16.735
4	n/a	3.354	n/a	8.192	n/a	16.463
5	n/a	3.925	n/a	7.821	n/a	15.412
6	n/a	3.856	n/a	8.472	n/a	14.240
7	n/a	3.756	n/a	8.452	n/a	16.684
8	n/a	3.625	n/a	7.051	n/a	15.092
9	n/a	3.285	n/a	8.843	n/a	17.074
10	n/a	3.906	n/a	8.192	n/a	17.555
average	n/a	3.6492	n/a	8.3080	n/a	15.8537
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	266.052	n/a	1104.578	n/a	>5000
2	n/a	246.344	n/a	1177.303	n/a	>5000
3	n/a	300.812	n/a	1215.057	n/a	>5000
4	n/a	260.875	n/a	1211.472	n/a	>5000
5	n/a	275.086	n/a	1306.048	n/a	>5000
6	n/a	294.163	n/a	1288.943	n/a	>5000
7	n/a	263.970	n/a	1204.201	n/a	>5000
8	n/a	268.366	n/a	1259.130	n/a	>5000
9	n/a	238.583	n/a	1211.122	n/a	>5000
10	n/a	244.922	n/a	1210.872	n/a	>5000
average	n/a	265.9173	n/a	1218.8726	n/a	>5000

ตารางผนวกที่ ก13(ต่อ)

	n = 500 , m =					
	10000		20000		30000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	n/a	n/a	n/a	n/a	n/a
2	n/a	n/a	n/a	n/a	n/a	n/a
3	n/a	n/a	n/a	n/a	n/a	n/a
4	n/a	n/a	n/a	n/a	n/a	n/a
5	n/a	n/a	n/a	n/a	n/a	n/a
6	n/a	n/a	n/a	n/a	n/a	n/a
7	n/a	n/a	n/a	n/a	n/a	n/a
8	n/a	n/a	n/a	n/a	n/a	n/a
9	n/a	n/a	n/a	n/a	n/a	n/a
10	n/a	n/a	n/a	n/a	n/a	n/a
average	n/a	n/a	n/a	n/a	n/a	n/a

<u>ตารางผนวกที่ ก14</u> เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อกำหนด n = 1000 (หน่วย : วินาที)

	n = 1000 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	19.978	n/a	33.949	n/a	65.384
2	n/a	20.009	n/a	37.904	n/a	60.667
3	n/a	20.819	n/a	35.501	n/a	62.050
4	n/a	20.109	n/a	34.830	n/a	69.350
5	n/a	19.959	n/a	35.691	n/a	59.666
6	n/a	20.410	n/a	33.839	n/a	62.750
7	n/a	20.159	n/a	33.699	n/a	55.810
8	n/a	20.430	n/a	34.850	n/a	56.110
9	n/a	20.710	n/a	36.823	n/a	60.406
10	n/a	20.199	n/a	37.353	n/a	52.976
average	n/a	20.2782	n/a	35.4439	n/a	60.5169
	500		1000		5000	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	1338.825	n/a	>5000	n/a	n/a
2	n/a	1379.243	n/a	>5000	n/a	n/a
3	n/a	1400.593	n/a	>5000	n/a	n/a
4	n/a	1357.932	n/a	>5000	n/a	n/a
5	n/a	1268.745	n/a	>5000	n/a	n/a
6	n/a	1258.019	n/a	>5000	n/a	n/a
7	n/a	1315.031	n/a	>5000	n/a	n/a
8	n/a	1300.350	n/a	>5000	n/a	n/a
9	n/a	1264.398	n/a	>5000	n/a	n/a
10	n/a	1265.199	n/a	>5000	n/a	n/a
average	n/a	1314.8335	n/a	>5000	n/a	n/a

ตารางผนวกที่ ก15 เวลาในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต เมื่อกำหนด n =5000 (หน่วย : วินาที)

	n = 5000 , m =					
	10		50		100	
no.	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
1	n/a	n/a	n/a	n/a	n/a	n/a
2	n/a	n/a	n/a	n/a	n/a	n/a
3	n/a	n/a	n/a	n/a	n/a	n/a
4	n/a	n/a	n/a	n/a	n/a	n/a
5	n/a	n/a	n/a	n/a	n/a	n/a
6	n/a	n/a	n/a	n/a	n/a	n/a
7	n/a	n/a	n/a	n/a	n/a	n/a
8	n/a	n/a	n/a	n/a	n/a	n/a
9	n/a	n/a	n/a	n/a	n/a	n/a
10	n/a	n/a	n/a	n/a	n/a	n/a
average	n/a	n/a	n/a	n/a	n/a	n/a

หมายเหตุ n/a = ไม่รู้ค่าเวลาในการหาผลเฉลย

<u>การทดสอบทางสถิติ</u>

1. การตรวจสอบค่าความแปรปรวน ของแต่ละวิธีโดยการทดสอบค่าเอฟ ในโปรแกรมสำเร็จรูป Microsoft Excel กำหนดให้

- สมมติฐานหลัก ค่าความแปรปรวนของวิธีคำสั่ง linprog $\left(\sigma_L^2\right)$ และค่าความแปรปรวนของวิธีตรง $\left(\sigma_T^2\right)$ เท่ากัน $\left(H_0 : \sigma_L^2 = \sigma_T^2\right)$
- สมมติฐานรอง ค่าความแปรปรวนของทั้งสองวิธีไม่เท่ากัน $\left(H_1 : \sigma_L^2 \neq \sigma_T^2\right)$
- ระดับความเชื่อมั่น 95 % $(\alpha = 0.05)$

เมื่อทำการทดสอบด้วยโปรแกรมสำเร็จ Microsoft Excel สามารถทำการสรุปได้ดังนี้ ถ้าค่าเอฟจากการคำนวณตกอยู่ในช่วงของเอฟวิกฤต (F-critical) ถือว่าปฏิเสธค่าสมมติฐานหลัก หรือทำการเปรียบเทียบค่าความน่าจะเป็น (P-value) ต้องน้อยกว่าหรือเท่ากับค่าระดับความสำคัญจึงจะสามารถปฏิเสธสมมติฐานหลัก เพื่อทราบค่าความแปรปรวนในการกำหนดการทดสอบค่าที ซึ่งสามารถแบ่งการตรวจสอบเป็น การทดสอบค่าทีกรณีที่ค่าความแปรปรวนของสองวิธีมีค่าเท่ากัน หรือการทดสอบค่าทีกรณีที่ค่าความแปรปรวนของสองวิธีมีค่าไม่เท่ากันโดยทำการเปรียบเทียบค่าเวลาที่ใช้ดังข้อถัดไปนี้

2. การตรวจสอบค่าเฉลี่ยแต่ละวิธี โดยการทดสอบค่าทีในโปรแกรมสำเร็จรูป Microsoft Excel ตามเงื่อนไขที่ได้จากการทดสอบค่าความแปรปรวน ซึ่งจะสามารถใช้การทดสอบค่าที เมื่อค่าความแปรปรวนของสองข้อมูลเท่ากัน และ การทดสอบค่าที เมื่อค่าความแปรปรวนของสองข้อมูลไม่เท่ากัน กำหนดให้

- สมมติฐานหลัก ค่าเวลาเฉลี่ยในการแก้ปัญหาของวิธีคำสั่ง linprog (μ_L) และวิธีการแก้ปัญหาด้วยวิธีตรง (μ_T) เท่ากัน $(H_0 : \mu_L = \mu_T)$
- สมมติฐานรอง ค่าเวลาเฉลี่ยในการแก้ปัญหาแต่ละวิธีไม่เท่ากัน $(H_1 : \mu_L \neq \mu_T)$
- ระดับความเชื่อมั่น 95 % $(\alpha = 0.05)$

เมื่อทำการทดสอบด้วยโปรแกรมสำเร็จ Microsoft Excel สามารถทำการสรุปได้ดังนี้ ถ้าค่าทีจากการคำนวณตกอยู่ในช่วงค่าทีวิกฤต (t-critical) ถือว่าปฏิเสธค่าสมมติฐานหลัก หรือทำ

การเปรียบเทียบค่าความน่าจะเป็น ต้องน้อยกว่าหรือเท่ากับค่าระดับความสำคัญจึงสามารถปฏิเสธสมมติฐานหลักได้

ตารางผนวกที่ ก16 ผลการทดสอบทางสถิติ เมื่อ n = 10

ค่าสถิติ	n = 10 , m =					
	10		50		100	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	0.038	0.3603	0.057	0.3906	0.075	0.4777
ค่าความแปรปรวน	6.22E-05	0.000218	0.000157	0.000156	7.22E-05	0.000937
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	0.2854		1.0054		0.0771	
ค่าความน่าจะเป็น	0.0379		0.4969		0.0004	
ค่าเอฟวิกฤต	0.2484		4.0260		0.2484	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance	0.0001		0.0002			
องศาอิสระ	18		18		10	
ค่าทีคำนวณ	-60.8836		-59.6772		-40.0832	
ค่าความน่าจะเป็น	2.6714E-22		3.8234E-22		2.2342E-12	
ค่าทีวิกฤต	2.1009		2.1009		2.2281	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก16 (ต่อ)

ค่าสถิติ	n = 10 , m =					
	500		1000		5000	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	0.3033	0.7242	0.7329	1.2438	10.9492	15.6023
ค่าความแปรปรวน	0.007448	0.000178	0.062085	0.000663	12.19968	4.686271
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	41.9068		93.6021		2.6033	
ค่าความน่าจะเป็น	0.0000		0.0000		0.0851	
ค่าเอฟวิกฤต	4.0260		4.0260		4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance					8.4430	
องศาอิสระ	9		9		18	
ค่าทีคำนวณ	-15.2416		-6.4496		-3.5808	
ค่าความน่าจะเป็น	9.8183E-08		0.0001		0.0021	
ค่าทีวิกฤต	2.2622		2.2622		2.1009	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก16 (ต่อ)

ค่าสถิติ	n = 10 , m =					
	10000		20000		30000	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	31.1207	37.4028	176.9143	188.9126	441.7252	482.4368
ค่าความแปรปรวน	65.64361	0.29477	8474.83	4.116806	13052.82	21482.36
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	222.6942		2058.5935		0.6076	
ค่าความน่าจะเป็น	0.0000		0.0000		0.2347	
ค่าเอฟวิกฤต	4.0260		4.0260		0.2484	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance					17267.5861	
องศาอิสระ	9		9		18	
ค่าทีคำนวณ	-2.4464		-0.4120		-0.6928	
ค่าความน่าจะเป็น	0.0370		0.6899		0.4973	
ค่าทีวิกฤต	2.2622		2.2622		2.1009	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0 ไม่ได้	

ตารางผนวกที่ ก16 (ต่อ)

ค่าสถิติ	n = 10 , m = 50000	
	2stagefrac	linprog
ค่าเฉลี่ย	1281.236	1736.375
ค่าความแปรปรวน	404687.3	461.5081
จำนวนข้อมูล	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน		
องศาอิสระ	9	9
ค่าเอฟคำนวณ	876.8801	
ค่าความน่าจะเป็น	0.0000	
ค่าเอฟวิกฤต	4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย		
Pooled Variance		
องศาอิสระ	9	
ค่าทีคำนวณ	-2.2612	
ค่าความน่าจะเป็น	0.0501	
ค่าทีวิกฤต	2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0 ไม่ได้	

ตารางผนวกที่ ก17 ผลการทดสอบทางสถิติ เมื่อ n = 50

ค่าสถิติ	n = 50 , m =					
	10		50		100	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	0.0601	0.3766	0.1172	0.4386	0.1862	0.6369
ค่าความแปรปรวน	0.000111	0.000341	8.93E-05	0.00051	0.000562	0.000873
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	0.3260		0.1752		0.6440	
ค่าความน่าจะเป็น	0.0552		0.0080		0.2613	
ค่าเอฟวิกฤต	0.2484		0.2484		0.2484	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance	0.0002				0.0007	
องศาอิสระ	18		12		18	
ค่าทีคำนวณ	-47.0575		-41.5310		-37.6303	
ค่าความน่าจะเป็น	2.6814E-20		2.4612E-14		1.4426E-18	
ค่าทีวิกฤต	2.1009		2.1788		2.1009	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก17 (ต่อ)

ค่าสถิติ	n = 50 , m =					
	500		1000		5000	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	1.5252	2.5055	3.8144	6.4683	65.776	220.4459
ค่าความแปรปรวน	0.082687	0.02016	1.113815	0.822734	2720.347	5.053689
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	4.1015		1.3538		538.2893	
ค่าความน่าจะเป็น	0.0236		0.3296		3.3536E-11	
ค่าเอฟวิกฤต	4.0260		4.0260		4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance			0.9683			
องศาอิสระ	13		18		9	
ค่าทีคำนวณ	-9.6664		-6.0307		-9.3689	
ค่าความน่าจะเป็น	2.6635E-07		1.0583E-05		6.1393E-06	
ค่าทีวิกฤต	2.1604		2.1009		2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก17 (ต่อ)

ค่าสถิติ	n = 50 , m = 10000	
	2stagefrac	linprog
ค่าเฉลี่ย	327.1053	1137.885
ค่าความแปรปรวน	24946.9	59.74696
จำนวนข้อมูล	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน		
องศาอิสระ	9	9
ค่าเอฟคำนวณ	417.5425	
ค่าความน่าจะเป็น	1.0477E-10	
ค่าเอฟวิกฤต	4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย		
Pooled Variance		
องศาอิสระ	9	
ค่าทีคำนวณ	-16.2134	
ค่าความน่าจะเป็น	5.7302E-08	
ค่าทีวิกฤต	2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0	

ตารางผนวกที่ ก18 ผลการทดสอบทางสถิติ เมื่อ n = 100

ค่าสถิติ	n = 100 , m =					
	10		50		100	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	0.0701	0.4015	0.1822	0.5239	0.3333	0.8813
ค่าความแปรปรวน	8.9E-05	0.00214	0.000175	0.000471	0.00201	0.006098
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	0.0416		0.3714		0.3297	
ค่าความน่าจะเป็น	2.9916E-05		0.0781		0.0569	
ค่าเอฟวิกฤต	0.2484		0.2484		0.2484	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0 ไม่ได้	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance			0.0003		0.0041	
องศาอิสระ	10		18		18	
ค่าทีคำนวณ	-22.1958		-42.5265		-19.2447	
ค่าความน่าจะเป็น	7.7336E-10		1.6334E-19		1.8702E-13	
ค่าทีวิกฤต	2.2281		2.1009		2.1009	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก18 (ต่อ)

ค่าสถิติ	n = 100 , m =					
	500		1000		5000	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	3.0696	6.5804	7.4878	21.6238	138.9127	1101.802
ค่าความแปรปรวน	0.2796	0.1081	1.8902	0.1375	3588.809	7.8642
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	2.5868		13.7433		456.3501	
ค่าความน่าจะเป็น	0.0865		2.9908E-04		7.0336E-11	
ค่าเอฟวิกฤต	4.0260		4.0260		4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance	0.1939					
องศาอิสระ	18		10		9	
ค่าทีคำนวณ	-17.8301		-31.3920		-50.7722	
ค่าความน่าจะเป็น	6.9250E-13		2.5282E-11		2.2390E-12	
ค่าทีวิกฤต	2.1009		2.2281		2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก18 (ต่อ)

ค่าสถิติ	n = 100 , m = 10000	
	2stagefrac	linprog
ค่าเฉลี่ย	1432.916	4961.535
ค่าความแปรปรวน	190509.4	611.316
จำนวนข้อมูล	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน		
องศาอิสระ	9	9
ค่าเอฟคำนวณ	311.6382	
ค่าความน่าจะเป็น	3.8846E-10	
ค่าเอฟวิกฤต	4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย		
Pooled Variance		
องศาอิสระ	9	
ค่าทีคำนวณ	-25.5241	
ค่าความน่าจะเป็น	1.0470E-09	
ค่าทีวิกฤต	2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0	

ตารางผนวกที่ ก19 ผลการทดสอบทางสถิติ เมื่อ n = 500

ค่าสถิติ	n = 500 , m =					
	10		50		100	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	0.1814	0.5287	0.69	2.0891	1.6044	6.1097
ค่าความแปรปรวน	0.000147	0.000348	0.006188	0.005228	0.031294	0.094111
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	0.4231		1.1837		0.3325	
ค่าความน่าจะเป็น	0.1081		0.4029		0.0583	
ค่าเอฟวิกฤต	0.2484		4.0260		0.2484	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0 ไม่ได้	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance	0.0002		0.0057		0.0627	
องศาอิสระ	18		18		18	
ค่าทีคำนวณ	-49.3658		-41.4097		-40.2315	
ค่าความน่าจะเป็น	1.1395E-20		2.6249E-19		4.3901E-19	
ค่าทีวิกฤต	2.1009		2.1009		2.1009	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก19 (ต่อ)

ค่าสถิติ	n = 500 , m =			
	500		1000	
	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	11.5466	215.2506	39.7161	1087.525
ค่าความแปรปรวน	3.641516	38.08382	112.0491	63.57994
จำนวนข้อมูล	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน				
องศาอิสระ	9	9	9	9
ค่าเอฟคำนวณ	0.0956		1.7623	
ค่าความน่าจะเป็น	0.0009		0.2057	
ค่าเอฟวิกฤต	0.2484		4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้	
การทดสอบค่าทีสำหรับค่าเฉลี่ย				
Pooled Variance			87.8145	
องศาอิสระ	11		18	
ค่าทีคำนวณ	-99.7240		-250.0252	
ค่าความน่าจะเป็น	1.2877E-17		2.5174E-33	
ค่าทีวิกฤต	2.2010		2.1009	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก20 ผลการทดสอบทางสถิติ เมื่อ n = 1000

ค่าสถิติ	n = 1000 , m =					
	10		50		100	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	0.3906	1.0705	1.3391	6.5752	3.0384	22.0808
ค่าความแปรปรวน	0.0012	0.5683	0.0359	2.8764	0.0664	0.7281
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	0.0021		0.0125		0.0912	
ค่าความน่าจะเป็น	5.6234E-11		1.6475E-07		0.0007	
ค่าเอฟวิกฤต	0.2484		0.2484		0.2484	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
องศาอิสระ	9		9		11	
ค่าทีคำนวณ	-2.8491		-9.7026		-67.5579	
ค่าความน่าจะเป็น	0.0191		4.5980E-06		9.2747E-16	
ค่าทีวิกฤต	2.2622		2.2622		2.2010	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก20 (ต่อ)

ค่าสถิติ	n = 1000 , m =			
	500		1000	
	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	26.3458	1098.279	97.4813	4842.11
ค่าความแปรปรวน	26.4453	359.6884	942.4801	110.2902
จำนวนข้อมูล	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน				
องศาอิสระ	9	9	9	9
ค่าเอฟคำนวณ	0.0735		8.5455	
ค่าความน่าจะเป็น	0.0003		0.0019	
ค่าเอฟวิกฤต	0.2484		4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย				
องศาอิสระ	10		11	
ค่าทีคำนวณ	-172.5039		-462.4187	
ค่าความน่าจะเป็น	1.0530E-18		6.0750E-25	
ค่าทีวิกฤต	2.2281		2.2010	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก21 ผลการทดสอบทางสถิติ เมื่อ n = 5000

ค่าสถิติ	n = 5000 , m =					
	10		50		100	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	3.6682	7.1031	8.9347	216.0088	15.5343	1107.391
ค่าความแปรปรวน	0.0724	0.3998	0.8902	1.9440	4.1612	77.4541
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	0.18112		0.45793		0.05372	
ค่าความน่าจะเป็น	0.00900		0.13010		0.00009	
ค่าเอฟวิกฤต	0.24839		0.24839		0.24839	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance			1.4171			
องศาอิสระ	12		18		10	
ค่าทีคำนวณ	-15.8078		-388.9650		-382.1903	
ค่าความน่าจะเป็น	2.1302E-09		8.8490E-37		3.6997E-22	
ค่าทีวิกฤต	2.1788		2.1009		2.2281	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก22 ผลการทดสอบทางสถิติ เมื่อ n = 10000

ค่าสถิติ	n = 10000 , m =			
	10		50	
	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	12.9874	25.4006	22.1129	1124.556
ค่าความแปรปรวน	0.1978	0.4405	2.8874	47.2287
จำนวนข้อมูล	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน				
องศาอิสระ	9	9	9	9
ค่าเอฟคำนวณ	0.4491		0.0611	
ค่าความน่าจะเป็น	0.1244		0.0001	
ค่าเอฟวิกฤต	0.2484		0.2484	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย				
Pooled Variance	0.3191			
องศาอิสระ	18		10	
ค่าทีคำนวณ	-49.1332		-492.4563	
ค่าความน่าจะเป็น	1.2399E-20		2.9332E-23	
ค่าทีวิกฤต	2.1009		2.2281	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก23 ผลการทดสอบทางสถิติสำหรับปัญหามีขอบเขต เมื่อ n = 10

ค่าสถิติ	n = 10 , m =					
	10		50		100	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	0.3007	0.5115	0.4098	0.5378	0.513	0.7039
ค่าความแปรปรวน	0.0130	0.0009	0.0230	0.0014	0.0858	0.0021
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	13.8777		16.7520		41.7640	
ค่าความน่าจะเป็น	0.0003		0.0001		2.8303E-06	
ค่าเอฟวิกฤต	4.0260		4.0260		4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance						
องศาอิสระ	10		10		9	
ค่าทีคำนวณ	-5.6427		-2.5924		-2.0363	
ค่าความน่าจะเป็น	0.0002		0.0268		0.0722	
ค่าทีวิกฤต	2.2281		2.2281		2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0		ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้	

ตารางผนวกที่ ก23 (ต่อ)

ค่าสถิติ	n = 10 , m =					
	500		1000		5000	
	2stagefrac	linprog	2stagefrac	linprog	2stagefrac	linprog
ค่าเฉลี่ย	1.8576	1.2899	6.1188	3.4239	190.0702	36.9311
ค่าความแปรปรวน	0.8838	0.0193	4.4469	3.8149	6370.472	10.9256
จำนวนข้อมูล	10	10	10	10	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน						
องศาอิสระ	9	9	9	9	9	9
ค่าเอฟคำนวณ	45.6775		1.1657		583.0776	
ค่าความน่าจะเป็น	1.9195E-06		0.4116		2.3430E-11	
ค่าเอฟวิกฤต	4.0260		4.0260		4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0		ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย						
Pooled Variance			4.1309			
องศาอิสระ	9		18		9	
ค่าทีคำนวณ	1.8890		2.9649		6.0622	
ค่าความน่าจะเป็น	0.0915		0.0083		0.0002	
ค่าทีวิกฤต	2.2622		2.1009		2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0 ไม่ได้		ปฏิเสธ H_0		ปฏิเสธ H_0	

ตารางผนวกที่ ก23 (ต่อ)

ค่าสถิติ	n = 10 , m = 10000	
	2stagefrac	linprog
ค่าเฉลี่ย	755.0986	80.8846
ค่าความแปรปรวน	88580.68	259.6844
จำนวนข้อมูล	10	10
การทดสอบค่าเอฟสำหรับความแปรปรวน		
องศาอิสระ	9	9
ค่าเอฟคำนวณ	341.1090	
ค่าความน่าจะเป็น	2.5920E-10	
ค่าเอฟวิกฤต	4.0260	
ผลการทดสอบค่าเอฟ	ปฏิเสธ H_0	
การทดสอบค่าทีสำหรับค่าเฉลี่ย		
Pooled Variance		
องศาอิสระ	9	
ค่าทีคำนวณ	7.1531	
ค่าความน่าจะเป็น	0.0001	
ค่าทีวิกฤต	2.2622	
ผลการทดสอบค่าที	ปฏิเสธ H_0	

ภาคผนวก ข

รายละเอียดโปรแกรมที่ใช้ในการแก้ปัญหากำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอน

<u>รายละเอียดโปรแกรมในการหาค่าผลเฉลยของวิธีกำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบไม่มีขอบเขต</u>

```
%The programming was written by Charuwan  Wiwattanakom 46651691.
%Industrial Engineering.
%Function Two-stage Linear fractional Programming.
clc;
clear all;
disp('Min Z = [sum(cjxj)+sum(giui)+sum(hivi)]/[sum(djxj)+sum(piui)+(qivi)]')
disp('S.T.      sum(ajxj)+ui-vi     =       bi')
disp('          xj, ui, vi          >=      0')
disp('          i= 1, 2, 3, ..., m  &   j= 1, 2, 3, ..., n')

n=input('please enter n:');
m=input('please enter m:');
c=rand(1,n)*10;
a=rand(1,n)*10;
d=rand(1,n)*100;    %fractional
b=rand(1,m)*10;
g=rand(1,m)*0.001;
h=rand(1,m)*0.001;
pu=rand(1,m)*0.001;      %('pmum' at fractional)
qv=rand(1,m)*0.001;      %('qmvm' at fractional)

%The fomulation by LinProg of toolbox of Matlab.
tic;
% Create the right hand side.
Rhs=zeros(1,m);
```

```
Rhs(1,m+1)=1;
nv=n+(2*m);
nc=m;
% Create the constrain.
am=sparse(nc,nv);
for i=1:nc
   for j=1:n
      am(i,j)=a(j);
   end
end
ix=n;
for k=1:nc
   ix=ix+1;
   am(k,ix)=1;           %put Ui
   am(k,ix+m)=-1;        %put Vi
   am(k,nv+1)=-b(k);     %put b*y0
end
for i=nc+1;
   for j=1:n
      am(i,j)=d(j);
   end
   for j=1:m
      am(i,n+j)=pu(j);
      am(i,n+m+j)=qv(j);
   end
   am(i,nv+1)=0;
end
```

```
% Create objective function.
for j=1:n
   cj(j)=c(j);
end
for i=1:m
   cj(n+i)=g(i);
   cj(n+m+i)=h(i);
end
cj(n+m+m+1)=0;
l=zeros(nv+1,1);
disp('The method(linprog toolbox) answer can show below : ')
[x,fval]=linprog(cj,[],[],am,Rhs,l)
toc;

%The solution by direct method of 2-Stage Linear Fractional Programming problem.
tic;
cs=c;
as=a;
ds=d;
[bs,f]=sort(b);
gs=g(f);    pus=pu(f);
hs=h(f);    qvs=qv(f);
xs=0;   %ys(j)
s=1;
js=1;
%find position k and p for condition more than and equal to zero.
for js=1:n
```

```
%find position p
chk1=0;
chk2=0;
s=1;
if (chk1<=0)|(chk2<=0)
    while (chk1<=0)|(chk2<=0)
        sumg1=0;  sump1=0;
        sumh1=0;  sumq1=0;
        sumg2=0;  sump2=0;
        sumh2=0;  sumq2=0;
        for i=s:m
          sumg1=sumg1+gs(i);
          sump1=sump1+pus(i);
        end
        for i=1:(s-1)
            if i<1
                sumh1 =0;
                sumq1 =0;
            else sumh1=sumh1+hs(i);
                sumq1=sumq1+qvs(i);
            end
        end
        if (s+1) <= m
           for i=(s+1):m
               sumg2=sumg2+gs(i);
               sump2=sump2+pus(i);
           end
```

```
else sumg2=0;
     sump2=0;
end
for i=1:s
  sumh2=sumh2+hs(i);
  sumq2=sumq2+qvs(i);
end
us=0;   %rs(i)
vs=0;   %ss(i)
us(1:(s-1))=0;
vs(1:(s-1))=bs(s)-bs(1:(s-1));
us(s)=0;
vs(s)=0;
us((s+1):m)=bs((s+1):m)-bs(s);
vs((s+1):m)=0;
puiui=pus.*us;
qvivi=qvs.*vs;
giui=gs.*us;
hivi=hs.*vs;
objU=(cs(js)*bs(s))/as(js)+sum(giui)+sum(hivi);
objL=(ds(js)*bs(s))/as(js)+sum(puiui)+sum(qvivi);
obj=objU/objL;
chk1=(-cs(js)/as(js)+sumg1-sumh1)-((-ds(js)/as(js)+sump1-sumq1)*obj);
chk2=(cs(js)/as(js)-sumg2+sumh2)-((ds(js)/as(js)-sump2+sumq2)*obj);
if (chk1>0)&(chk2>0)
    ps=s;
    xs(js)=bs(ps)/as(js);
```

```
            Pxs(js)=ps;
            Objxs(js)=obj;
        else s=s+1;
        end
                    if s>m
            xs(js)=0;
            Pxs(js)=0;
            Objxs(js)=0;
            chk1 = 1000000;
            chk2 = 1000000;
        end
    end
  end
end

% check k subscribs
for k=1:n
  ck3=0;
  dk3=0;
  if xs(k) ~= 0
    for j=1:n
        if j~=k
            ck3=ck3+(cs(j)-(as(j)*cs(k)/as(k)));
            dk3=dk3+((ds(j)-(as(j)*ds(k)/as(k)))*Objxs(k));
        else ck3=ck3;
            dk3=dk3;
        end
```

```
        end
    else ck3= -10000;
        dk3=0;
    end
    chk3=ck3-dk3;
    if chk3 >= 0
        xs(k)=xs(k);
        Pxs(k)=Pxs(k);
        Objxs(k)=Objxs(k);
    else xs(k)=0;
        Pxs(k)=0;
        Objxs(k)=0;
    end
end

%check i of r and s.
for i=1:n
    if xs(i) ~= 0
        pR=Pxs(i);
        gh4=0;
        pq4=0;
        gh5=0;
        pq5=0;
        if (pR-1) < 1
          gh4 = 0;
          pq4 = 0;
        else for rr=1:(pR-1)
```

```
                gh4=gh4+gs(rr)+hs(rr);
                pq4=pq4+((pus(rr)+qvs(rr))*Objxs(i));
            end
        end
        if (pR+1) > m
          gh5 = 0;
          pq5 = 0;
        else for ss=(pR+1):m
                gh5=gh5+gs(ss)+hs(ss);
                pq5=pq5+((pus(ss)+qvs(ss))*Objxs(i));
            end
        end
        chk4=gh4-pq4;
        chk5=gh5-pq5;
        if chk4 >= 0 & chk5 >= 0
          xs(i)=xs(i);
          Pxs(i)=Pxs(i);
          Obj1(i)=Objxs(i);
        else xs(i)=0;
           Pxs(i)=0;
           Obj1(i)=1000000;
        end
    else xs(i)=0;
         Pxs(i)=0;
         Obj1(i)=1000000;
    end
end
```

```
% Check min and return value then show the answer.
MINZ=min(Obj1);
for i=1:n
   if MINZ==Obj1(i)
      pos=i;
   end
end

if MINZ ~= 1000000
   for k=1:n
      if k==pos
         xs(k)=xs(k);
         Pxs(k)=Pxs(k);
      else xs(k)=0;
         Pxs(k)=0;
      end
   end
   PXS=max(Pxs);
   for k=1:n
      X(k)=xs(k);
      Objxss(k)=Objxs(k);
   end
   us=0;   %rs(i)
   vs=0;   %ss(i)
   us(1:(PXS-1))=0;
   vs(1:(PXS-1))=bs(PXS)-bs(1:(PXS-1));
   us(PXS)=0;
```

```
    vs(PXS)=0;
    us((PXS+1):m)=bs((PXS+1):m)-bs(PXS);
    vs((PXS+1):m)=0;
    for w=1:m
       u(f(w))=us(w);
       v(f(w))=vs(w);
    end
    Pi=find(u==0 & v==0);

    %check min
    fprintf('\n____________________________________________\n\n');
       disp('optimal x=')
       disp(X)
       disp('u(i)=')
       disp(u)
       disp('v(i)=')
       disp(v)
    fprintf('min z = %8.4f\n',MINZ);
    fprintf('p= %8.4f\n',Pi);

else fprintf('\n--------------------------------------------------\n\n');
     disp('The problem is infeasible solution')
     disp('       Min Z = 0 ')
end
toc;
fprintf('\n------------------------------------------------\n\n');
```

<u>รายละเอียดโปรแกรมในการหาค่าผลเฉลยของวิธีกำหนดการเศษส่วนเชิงเส้นหนึ่งเงื่อนไขสองขั้นตอนแบบมีขอบเขต</u>

```
%Main program.
%Function Two-stage Linear fractional Programming with bounded variable.
clc;
clear all;
disp('Min Z = [sum(cjxj)+sum(giui)+sum(hivi)]/[sum(djxj)+sum(piui)+(qivi)]')
disp('S.T.      sum(ajxj)+ui-vi     =       bi')
disp('          xj, ui, vi          >=      0')
disp('          i= 1, 2, 3, ..., m  &   j= 1, 2, 3, ..., n')

n=input('please enter n:');
m=input('please enter m:');
c=rand(1,n)*1;
a=rand(1,n)*10;
d=rand(1,n)*100;    %fractional
t=rand(1,n)*10;
b=rand(1,m)*10;
g=rand(1,m);
h=rand(1,m);
pu=rand(1,m); %('pmum' at fractional)
qv=rand(1,m); %('qmvm' at fractional)

%the formulation by Linprog of Matlab's toolbox.
tic;
disp('The method(linprog toolbox) answer can show below : ')
[x,fval]=linprog2stagefrac(a,b,c,d,g,h,m,n,pu,qv,t);
```

```
y(1:n)=x(1:n)
r(1:m)=x((n+1):(n+m))
s(1:m)=x((n+m+1):(n+m+m))
y0=x(n+m+m+1)
fval=fval
toc;
whos

% start algorithm for the Bounded 2-Stage Linear Fractional Programming and single
constrained.
disp('The answer of Two-stage Linear Fractional Programming can show below : ')
tic;
cs=c;
as=a;
ds=d;
ts=t;
% start algorithm for the first step(One-component of X).
[MINZ1,X1,U1,V1]=boundlevel1(as,b,cs,ds,g,h,m,n,pu,qv,ts);
minz(1,1)=MINZ1;
X(1,(1:n))=X1(1:n);
u(1,(1:m))=U1(1:m);
v(1,(1:m))=V1(1:m);
ta=ts.*a;
chk=min(ta);
if chk<max(X)
%Check upper level : the basic variables of X are 2 components.
[MINZ2,X2,U2,V2,testx2]=boundlevel2(as,b,cs,ds,g,h,m,n,pu,qv,ts);
```

```
minz(1,2)=MINZ2;
X(2,(1:n))=X2(1:n);
u(2,(1:m))=U2(1:m);
v(2,(1:m))=V2(1:m);
chk1=max(max(testx2));
s=2;
if chk1~=0
   %Check the third level : the basic variables of X are 3 components.
   [MINZ3,X3,U3,V3,testx3]=boundlevel3(as,b,cs,ds,g,h,m,n,pu,qv,ts,testx2);
   minz(1,3)=MINZ3;
   X(3,(1:n))=X3(1:n);
   u(3,(1:m))=U3(1:m);
   v(3,(1:m))=V3(1:m);
   chk2=max(max(max(testx3)));
   s=3;
   if chk2~=0
      %Check the forth level : the basic variables of X are 4 components.
      [MINZ4,X4,U4,V4,testx4]=boundlevel4(as,b,cs,ds,g,h,m,n,pu,qv,ts,testx3);
      minz(1,4)=MINZ4;
      X(4,(1:n))=X4(1:n);
      u(4,(1:m))=U4(1:m);
      v(4,(1:m))=V4(1:m);
      chk3=max(max(max(max(testx4))));
      s=4;
      if chk3~=0 & MINZ4<MINZ3
         %Check the fifth level : the basic variables of X are 5 components.
         [MINZ5,X5,U5,V5]=boundlevel5(as,b,cs,ds,g,h,m,n,pu,qv,ts,testx4);
```

```
            minz(1,5)=MINZ5;
            X(5,(1:n))=X5(1:n);
            u(5,(1:m))=U5(1:m);
            v(5,(1:m))=V5(1:m);
            s=5;
        end
    end
end
else s=1;
end
k=0;
obj=min(minz);
if obj ~= 100
    for i=1:s
        if minz(i)==obj
            k=i;
        end
    end
    if obj~=1
        X_ans(1:n)=X(k,(1:n));
        U_ans(1:m)=u(k,(1:m));
        V_ans(1:m)=v(k,(1:m));
        Pi=find(U_ans==0 & V_ans==0);
        fprintf('\n_______________________________________________\n\n');
        fprintf('min z = %8.4f\n',obj);
        disp('The Objective value is')
        disp('optimal x=')
```

```
        disp(X_ans)
        disp('u(i)=')
        disp(U_ans)
        disp('v(i)=')
        disp(V_ans)
        fprintf('position of i = %8.0f\n',Pi);
    else fprintf('\n--------------------------------------------------\n\n');
        disp('Objective have no feasible solution (for 5 components level).')
    end
else fprintf('\n--------------------------------------------------\n\n');
    disp('Objective have no feasible solution (for 5 components level).')
end
toc;
fprintf('\n---------------------------------------------\n\n');
whos

%1st Subprogram .
%Function formulate by Linprog of Matlab's toolbox
function [x,fval]=linprog2stagefrac(a,b,c,d,g,h,m,n,pu,qv,t)
Rhs=zeros(1,m);
Rhs(1,m+1)=1;
nv=n+(2*m);
nc=m;
am=sparse(nc,nv);
for i=1:nc
    for j=1:n
        am(i,j)=a(j);
```

```
    end
end
ix=n;
for k=1:nc
    ix=ix+1;
    am(k,ix)=1;              %put Ui
    am(k,ix+m)=-1;           %put Vi
    am(k,n+m+m+1)=-b(k);     %put b*y0
end
for i=nc+1;
    for j=1:n
        am(i,j)=d(j);
    end
    for j=1:m
        am(i,n+j)=pu(j);
        am(i,n+m+j)=qv(j);
    end
    am(i,n+m+m+1)=0;
end
% bounded constraints.
for i=1:n
    A(i,i)=1;
    A(i,n+m+m+1)=-t(i);
    B(1,i)=0;
end

%Objective function.
```

```
for j=1:n
   cj(j)=c(j);
end
for i=1:m
   cj(n+i)=g(i);
   cj(n+m+i)=h(i);
end
cj(n+m+m+1)=0;
l=zeros(nv+1,1);
[x,fval]=linprog(cj,A,B,am,Rhs,l);
%End of program.

%2 nd subprogram.
%Function of the direct method formulate at the first step.
function [MINZ1,X1,U1,V1]=boundlevel1(as,b,cs,ds,g,h,m,n,pu,qv,ts)
[bs,f]=sort(b);
gs=g(f);
hs=h(f);
pus=pu(f);
qvs=qv(f);
% check k(k=1:n)
for k=1:n
   testx(k)=bs(1)/as(k);
   if testx(k) > ts(k)
      testx(k)=0;
   end
end
```

```
%find position k and p for condition more than and equal to zero.
for js=1:n
   if testx(js)~=0
      %find position p
      chk1=-100;
      chk2=0;
      s=1;
      if (chk1<0)|(chk2<0)
         while (chk1<0)|(chk2<0)
            sumg1=0;  sump1=0;
            sumh1=0;  sumq1=0;
            sumg2=0;  sump2=0;
            sumh2=0;  sumq2=0;
            for i=s:m
               sumg1=sumg1+gs(i);
               sump1=sump1+pus(i);
            end
            for i=1:(s-1)
               sumh1=sumh1+hs(i);
               sumq1=sumq1+qvs(i);
            end
            for i=(s+1):m
               sumg2=sumg2+gs(i);
               sump2=sump2+pus(i);
            end
            for i=1:s
```

```
    sumh2=sumh2+hs(i);
    sumq2=sumq2+qvs(i);
end

xs(1:(js-1))=0;
xs(js)=bs(s)/as(js);
xs((js+1):n)=0;
cjxj=cs.*xs;
djxj=ds.*xs;

us=0;   %rs(i)
vs=0;   %ss(i)
us(1:(s-1))=0;
vs(1:(s-1))=bs(s)-bs(1:(s-1));
us(s)=0;
vs(s)=0;
us((s+1):m)=bs((s+1):m)-bs(s);
vs((s+1):m)=0;
giui=gs.*us;
hivi=hs.*vs;
puiui=pus.*us;
qvivi=qvs.*vs;

objU=sum(cjxj)+sum(giui)+sum(hivi);
objL=sum(djxj)+sum(puiui)+sum(qvivi);
obj=objU/objL;
chk1=(-cs(js)/as(js)+sumg1-sumh1)-((-ds(js)/as(js)+sump1-sumq1)*obj);
```

```
            chk2=(cs(js)/as(js)-sumg2+sumh2)-((ds(js)/as(js)-sump2+sumq2)*obj);
            if (chk1>0)&(chk2>0)
               Xs(js)=xs(js);
               Pxs(js)=s;
               Objxs(js)=obj;
            else s=s+1;
            end
            if s>m
               Xs(js)=0;
               Pxs(js)=0;
               Objxs(js)=0;
               chk1 = 1000000;
               chk2 = 1000000;
            end
         end
      end
   else Xs(js)=0;
       Pxs(js)=0;
       Objxs(js)=0;
   end
   if Objxs(js) < 0 | Xs(js) < 0  | Xs(js)>ts(js)
      Xs(js)=0;
      Pxs(js)=0;
      Objxs(js)=1000000;
   end
end
```

```
% check k of j subscribs
for k=1:n
   ck3=0;  ck4=0;
   dk3=0;  dk4=0;
   if Xs(k) ~= 0
      for j=1:n
         if j~=k
            ck4=ck4+(cs(j)-(as(j)*cs(k)/as(k)));
            dk4=dk4+((ds(j)-(as(j)*ds(k)/as(k)))*Objxs(k));
         else ck4=ck4;
            dk4=dk4;
         end
      end
   else ck3= -10000;
      ck4= -10000;
      dk3= 0;
      dk4= 0;
   end
   chk3=ck3-dk3;
   chk4=ck4-dk4;
   if chk3 < 0 | chk4 < 0
      Xs(k)=0;
      Pxs(k)=0;
      Objxs(k)=0;
   end
end
```

```
%check i of r and s.
for i=1:n
   if Xs(i) ~= 0
      pR=Pxs(i);
      gh5=0;
      pq5=0;
      gh6=0;
      pq6=0;
      for rr=1:(pR-1)
         gh5=gh5+gs(rr)+hs(rr);
         pq5=pq5+((pus(rr)+qvs(rr))*Objxs(i));
      end
      for ss=(pR+1):m
         gh6=gh6+gs(ss)+hs(ss);
         pq6=pq6+((pus(ss)+qvs(ss))*Objxs(i));
      end
      chk5=gh5-pq5;
      chk6=gh6-pq6;
      if chk5 >= 0 & chk6 >= 0
        Xs(i)=Xs(i);
        Pxs(i)=Pxs(i);
        Obj1(i)=Objxs(i);
      else Xs(i)=0;
         Pxs(i)=0;
         Obj1(i)=1000000;
      end
   else Xs(i)=0;
```

```
        Pxs(i)=0;
        Obj1(i)=1000000;
    end
end

MINZ1=min(Obj1);
pos=find(Obj1==MINZ1);

if MINZ1 ~= 1000000
    w=pos;  %Position of X(k).

    X1(1:(w-1))=0;
    Pxs(1:(w-1))=0;
    X1(w)=Xs(w);
    Pxs(w)=Pxs(w);
    X1((w+1):n)=0;
    Pxs((w+1):n)=0;
    PXS=max(Pxs);
    Objf=Objxs;

    us=0;   %rs(i)
    vs=0;   %ss(i)
    us(1:(PXS-1))=0;
    vs(1:(PXS-1))=bs(PXS)-bs(1:(PXS-1));
    us(PXS)=0;
    vs(PXS)=0;
    us((PXS+1):m)=bs((PXS+1):m)-bs(PXS);
```

```
   vs((PXS+1):m)=0;
   %arrange by original
   U1(f(1:m))=us(1:m);
   V1(f(1:m))=vs(1:m);
   Pi=find(U1==0 & V1==0);
else %fprintf('\n--------------------------------------------------\n\n');
   MINZ1=100;
   X1=ones(1,n);
   U1=ones(1,m);
   V1=ones(1,m);
   %disp('Objective have no feasible solution')
end
%End of program.

%3 rd subprogram.
%Function of the direct method formulate at the second step.
function [MINZ2,X2,U2,V2,testx2]=boundlevel2(as,b,cs,ds,g,h,m,n,pu,qv,ts)
% Find answer when x are more than one variable.
[bs,f]=sort(b);
gs=g(f);
hs=h(f);
pus=pu(f);
qvs=qv(f);

%To check upper level (two-components).
for j=1:n
   ta(j)=ts(j)*as(j);
```

```
    for k=1:n
        if k~=j
            testx2(j,k)=(bs(m)-ta(j))/as(k);
            if testx2(j,k)<0
                testx2(j,k)=0;
            end
        end
    end
end
%ta
%testx2
%Find answer of X variable more than and equal to
for j=1:n
    for k=1:n
        if testx2(j,k)~=0
            chk1=-100;
            chk2=0;
            s=1;
            if (chk1<0)|(chk2<0)
                while (chk1<0)|(chk2<0)
                    sumg1=0;  sump1=0;
                    sumh1=0;  sumq1=0;
                    sumg2=0;  sump2=0;
                    sumh2=0;  sumq2=0;
                    for i=s:m
                        sumg1=sumg1+gs(i);
                        sump1=sump1+pus(i);
```

```
end
for i=1:(s-1)
   sumh1=sumh1+hs(i);
   sumq1=sumq1+qvs(i);
end
for i=(s+1):m
   sumg2=sumg2+gs(i);
   sump2=sump2+pus(i);
end
for i=1:s
   sumh2=sumh2+hs(i);
   sumq2=sumq2+qvs(i);
end
for js=1:n
   if js==k
      xs(js)=(bs(s)-ta(j))/as(k);
   elseif js==j
      xs(js)=ts(js);
   else xs(js)=0;
   end
end
cjxj=cs.*xs;
djxj=ds.*xs;

us=0;   %rs(i)
vs=0;   %ss(i)
us(1:(s-1))=0;
```

```
vs(1:(s-1))=bs(s)-bs(1:(s-1));
us(s)=0;
vs(s)=0;
us((s+1):m)=bs((s+1):m)-bs(s);
vs((s+1):m)=0;
giui=gs.*us;
hivi=hs.*vs;
puiui=pus.*us;
qvivi=qvs.*vs;

objU=sum(cjxj)+sum(giui)+sum(hivi);
objL=sum(djxj)+sum(puiui)+sum(qvivi);
obj=objU/objL;
chk1=(-cs(k)/as(k)+sumg1-sumh1)-((-ds(k)/as(k)+sump1-sumq1)*obj);
chk2=(cs(k)/as(k)-sumg2+sumh2)-((ds(k)/as(k)-sump2+sumq2)*obj);
if (chk1>=0)&(chk2>=0)
   Xs(j,k)=xs(k);
   Pxs(j,k)=s;
   Obj2xs(j,k)=obj;
else s=s+1;
end
if s>m
   Xs(j,k)=0;
   Pxs(j,k)=0;
   Obj2xs(j,k)=0;
   chk1 = 1000000;
   chk2 = 1000000;
```

```
            end
          end
        end
      else Xs(j,k)=0;
        Pxs(j,k)=0;
        Obj2xs(j,k)=0;
      end
      if Xs(j,k)>ts(k)| Xs(j,k)<0 | Obj2xs(j,k)<0
        Xs(j,k)=0;
        Pxs(j,k)=0;
        Obj2xs(j,k)=0;
      end
    end
  end

  % check k of j subscribs of X.
  for j=1:n
    for k=1:n
      ck3=0;  ck4=0;
      dk3=0;  dk4=0;
      if Xs(j,k) ~= 0
        for jk=1:n
          if jk==j
            ck3=ck3+((as(jk)*cs(k)/as(k))-cs(jk));
            dk3=dk3+(((as(jk)*ds(k)/as(k))-ds(jk))*Obj2xs(j,k));
          elseif jk==k
            ck3=ck3;
```

```
                dk3=dk3;
                ck4=ck4;
                dk4=dk4;
            else ck4=ck4+(cs(jk)-(as(jk)*cs(k)/as(k)));
                dk4=dk4+((ds(jk)-(as(jk)*ds(k)/as(k)))*Obj2xs(j,k));
            end
        end
        chk3=ck3-dk3;
        chk4=ck4-dk4;
        if chk3 < 0 | chk4 <0
            Xs(j,k)=0;
            Pxs(j,k)=0;
            Objxs(j,k)=0;
        end
    end
  end
end

for j=1:n
  for k=1:n
    if Xs(j,k) ~= 0
        ip=Pxs(j,k);
        gh4=0;
        pq4=0;
        gh5=0;
        pq5=0;
        for rr=1:(ip-1)
```

```
            gh4=gh4+gs(rr)+hs(rr);
            pq4=pq4+((pus(rr)+qvs(rr))*Obj2xs(j,k));
         end
         for ss=(ip+1):m
            gh5=gh5+gs(ss)+hs(ss);
            pq5=pq5+((pus(ss)+qvs(ss))*Obj2xs(j,k));
         end
         chk4=gh4-pq4;
         chk5=gh5-pq5;
         if chk4 >= 0 & chk5 >= 0
            Xs(j,k)=Xs(j,k);
            Pxs(j,k)=Pxs(j,k);
            Obj1(j,k)=Obj2xs(j,k);
         else Xs(j,k)=0;
            Pxs(j,k)=0;
            Obj1(j,k)=1000000;
         end
      else Xs(j,k)=0;
         Pxs(j,k)=0;
         Obj1(j,k)=1000000;
      end
   end
end

MINZ2=100;
for j=1:n
   for k=1:n
```

```
        if MINZ2>Obj1(j,k)
            MINZ2=Obj1(j,k);
            xmin=Xs(j,k);
            pmin=Pxs(j,k);
            pr=j;
            pc=k;
        end
    end
end
if MINZ2 ~= 100
    for ja=1:n
        if ja==pr
            X2(ja)=ts(ja);
            Objx(ja)=0;
        elseif ja==pc
            X2(ja)=xmin;
            Objx(ja)=MINZ2;
        else X2(ja)=0;
            Objx(ja)=0;
        end
    end
    us=0;   %rs(i)
    vs=0;   %ss(i)
    us(1:(pmin-1))=0;
    vs(1:(pmin-1))=bs(pmin)-bs(1:(pmin-1));
    us(pmin)=0;
    vs(pmin)=0;
```

```
    us((pmin+1):m)=bs((pmin+1):m)-bs(pmin);
    vs((pmin+1):m)=0;
    %arrange by original
    U2(f(1:m))=us(1:m);
    V2(f(1:m))=vs(1:m);
    Pi=find(U2==0 & V2==0);
else disp('Objective have no feasible solution in 2 components level.')
     MINZ2=100;
     X2=ones(1,n);
     U2=ones(1,m);
     V2=ones(1,m);
end
%End of program.

%4 th subprogram.
%Function of the direct method formulate at the third step.
function [MINZ3,X3,U3,V3,testx3]=boundlevel3(as,b,cs,ds,g,h,m,n,pu,qv,ts,testx2)
[bs,f]=sort(b);
gs=g(f);
hs=h(f);
pus=pu(f);
qvs=qv(f);
testx3=zeros(n,n,n);

%To check upper level (three-components).
for j1=1:n
    ta1(j1)=ts(j1)*as(j1);
```

```
    for j2=1:n
        ta2(j2)=ts(j2)*as(j2);
        if testx2(j1,j2)~=0
            for k=1:n
                if k~=j1 & k~=j2
                    testx3(j1,j2,k)=(bs(m)-ta1(j1)-ta2(j2))/as(k);
                    if testx3(j1,j2,k)<0
                        testx3(j1,j2,k)=0;
                    end
                else testx3(j1,j2,k)=0;
                end
            end
        end
    end
end

%Find answer of X variable more than and equality.
for j1=1:n
    for j2=1:n
        for k=1:n
            if testx3(j1,j2,k)~=0
                chk1=-100;
                chk2=0;
                s=1;
                if (chk1<0)|(chk2<0)
                    while (chk1<0)|(chk2<0)
                        sumg1=0;  sump1=0;
```

```
sumh1=0;  sumq1=0;
sumg2=0;  sump2=0;
sumh2=0;  sumq2=0;
for i=s:m
   sumg1=sumg1+gs(i);
   sump1=sump1+pus(i);
end
for i=1:(s-1)
   sumh1=sumh1+hs(i);
   sumq1=sumq1+qvs(i);
end
for i=(s+1):m
   sumg2=sumg2+gs(i);
   sump2=sump2+pus(i);
end
for i=1:s
   sumh2=sumh2+hs(i);
   sumq2=sumq2+qvs(i);
end
for js=1:n
   if js==k
      xs(js)=(bs(s)-ta1(j1)-ta2(j2))/as(k);
   elseif js==j1|js==j2
      xs(js)=ts(js);
   else xs(js)=0;
   end
end
```

```
cjxj=cs.*xs;
djxj=ds.*xs;

us=0;   %rs(i)
vs=0;   %ss(i)
us(1:(s-1))=0;
vs(1:(s-1))=bs(s)-bs(1:(s-1));
us(s)=0;
vs(s)=0;
us((s+1):m)=bs((s+1):m)-bs(s);
vs((s+1):m)=0;
giui=gs.*us;
hivi=hs.*vs;
puiui=pus.*us;
qvivi=qvs.*vs;

objU=sum(cjxj)+sum(giui)+sum(hivi);
objL=sum(djxj)+sum(puiui)+sum(qvivi);
obj=objU/objL;
chk1=(-cs(k)/as(k)+sumg1-sumh1)-((-ds(k)/as(k)+sump1-sumq1)*obj);
chk2=(cs(k)/as(k)-sumg2+sumh2)-((ds(k)/as(k)-sump2+sumq2)*obj);
if (chk1>=0)&(chk2>=0)
   Xs(j1,j2,k)=xs(k);
   Pxs(j1,j2,k)=s;
   Obj3xs(j1,j2,k)=obj;
else s=s+1;
end
```

```
                if s>m
                    Xs(j1,j2,k)=0;
                    Pxs(j1,j2,k)=0;
                    Obj3xs(j1,j2,k)=0;
                    chk1 = 1000000;
                    chk2 = 1000000;
                end
            end
        end
      else Xs(j1,j2,k)=0;
          Pxs(j1,j2,k)=0;
          Obj3xs(j1,j2,k)=0;
      end
      if Xs(j1,j2,k)>ts(k)| Xs(j1,j2,k)<0 | Obj3xs(j1,j2,k)<0
          Xs(j1,j2,k)=0;
          Pxs(j1,j2,k)=0;
          Obj3xs(j1,j2,k)=0;
      end
    end
  end
end

%Check k of j subscribs of X.
for j1=1:(n-1)
  for j2=(j1+1):n
    for k=1:n
      ck3=0;  ck4=0;
```

```
        dk3=0;  dk4=0;
        if Xs(j1,j2,k) ~= 0
          for jk=1:n
            if jk==j1 |jk==j2
              ck3=ck3+((as(jk)*cs(k)/as(k))-cs(jk));
              dk3=dk3+(((as(jk)*ds(k)/as(k))-ds(jk))*Obj3xs(j1,j2,k));
            elseif jk==k
              ck3=ck3;
              dk3=dk3;
              ck4=ck4;
              dk4=dk4;
            else ck4=ck4+(cs(jk)-(as(jk)*cs(k)/as(k)));
              dk4=dk4+((ds(jk)-(as(jk)*ds(k)/as(k)))*Obj3xs(j1,j2,k));
            end
          end
          chk3=ck3-dk3;
          chk4=ck4-dk4;
          if chk3 < 0 | chk4 <0
            Xs(j1,j2,k)=0;
            Pxs(j1,j2,k)=0;
            Obj3xs(j1,j2,k)=0;
          end
        end
      end
    end
end
```

```
%Check p of i subscribs of X.
for j1=1:(n-1)
    for j2=(j1+1):n
        for k=1:n
            if Xs(j1,j2,k) ~= 0
                ip=Pxs(j1,j2,k);
                gh4=0;
                pq4=0;
                gh5=0;
                pq5=0;
                for rr=1:(ip-1)
                    gh4=gh4+gs(rr)+hs(rr);
                    pq4=pq4+((pus(rr)+qvs(rr))*Obj3xs(j1,j2,k));
                end
                for ss=(ip+1):m
                    gh5=gh5+gs(ss)+hs(ss);
                    pq5=pq5+((pus(ss)+qvs(ss))*Obj3xs(j1,j2,k));
                end
                chk4=gh4-pq4;
                chk5=gh5-pq5;
                if chk4 >= 0 & chk5 >= 0
                    Xs(j1,j2,k)=Xs(j1,j2,k);
                    Pxs(j1,j2,k)=Pxs(j1,j2,k);
                    Obj1(j1,j2,k)=Obj3xs(j1,j2,k);
                else Xs(j1,j2,k)=0;
                    Pxs(j1,j2,k)=0;
                    Obj1(j1,j2,k)=1000000;
```

```
                end
            else Xs(j1,j2,k)=0;
                Pxs(j1,j2,k)=0;
                Obj1(j1,j2,k)=1000000;
            end
        end
    end
end
MINZ3=100;
for j1=1:(n-1)
    for j2=(j1+1):n
        for k=1:n
            if MINZ3>Obj1(j1,j2,k)
                MINZ3=Obj1(j1,j2,k);
                xmin=Xs(j1,j2,k);
                pmin=Pxs(j1,j2,k);
                pr1=j1;
                pr2=j2;
                pc=k;
            end
        end
    end
end
if MINZ3 ~= 100
    for ja=1:n
        if ja==pr1 | ja==pr2
            X3(ja)=ts(ja);
```

```
            Objx(ja)=0;
        elseif ja==pc
            X3(ja)=xmin;
            Objx(ja)=MINZ3;
        else X3(ja)=0;
            Objx(ja)=0;
        end
    end
    us=0;   %rs(i)
    vs=0;   %ss(i)
    us(1:(pmin-1))=0;
    vs(1:(pmin-1))=bs(pmin)-bs(1:(pmin-1));
    us(pmin)=0;
    vs(pmin)=0;
    us((pmin+1):m)=bs((pmin+1):m)-bs(pmin);
    vs((pmin+1):m)=0;
    %arrange by original
    U3(f(1:m))=us(1:m);
    V3(f(1:m))=vs(1:m);
    Pi=find(U3==0 & V3==0);
else disp('Objective have no feasible solution in this 3 components level.')
    MINZ3=100;
    X3=ones(1,n);
    U3=ones(1,m);
    V3=ones(1,m);
end
%End of program
```

```
%5 th subprogram.
%Function of the direct method formulate at the forth step.
function [MINZ4,X4,U4,V4,testx4]=boundlevel4(as,b,cs,ds,g,h,m,n,pu,qv,ts,testx3)
[bs,f]=sort(b);
gs=g(f);
hs=h(f);
pus=pu(f);
qvs=qv(f);
testx4=zeros(n,n,n,n);

%To check upper level (four-components).
for j1=1:n
   ta1(j1)=ts(j1)*as(j1);
   for j2=1:n
      ta2(j2)=ts(j2)*as(j2);
      for j3=1:n
         ta3(j3)=ts(j3)*as(j3);
         if testx3(j1,j2,j3)~=0
            for k=1:n
               if k~=j1 & k~=j2 & k~=j3
                  testx4(j1,j2,j3,k)=(bs(m)-ta1(j1)-ta2(j2)-ta3(j3))/as(k);
                  if testx4(j1,j2,j3,k)<0
                     testx4(j1,j2,j3,k)=0;
                  end
               end
            end
```

```
            end
        end
    end
end
%Find answer of X variable more than and equal to
for j1=1:n
    for j2=1:n
        for j3=1:n
            for k=1:n
                if testx4(j1,j2,j3,k)~=0
                    chk1=-100;
                    chk2=0;
                    s=1;
                    if (chk1<0)|(chk2<0)
                        while (chk1<0)|(chk2<0)
                            sumg1=0;  sump1=0;
                            sumh1=0;  sumq1=0;
                            sumg2=0;  sump2=0;
                            sumh2=0;  sumq2=0;
                            for i=s:m
                                sumg1=sumg1+gs(i);
                                sump1=sump1+pus(i);
                            end
                            for i=1:(s-1)
                                sumh1=sumh1+hs(i);
                                sumq1=sumq1+qvs(i);
                            end
```

```
for i=(s+1):m
    sumg2=sumg2+gs(i);
    sump2=sump2+pus(i);
end
for i=1:s
    sumh2=sumh2+hs(i);
    sumq2=sumq2+qvs(i);
end
for js=1:n
    if js==k
        xs(js)=(bs(s)-ta1(j1)-ta2(j2)-ta3(j3))/as(k);
    elseif js==j1|js==j2|js==j3
        xs(js)=ts(js);
    else xs(js)=0;
    end
end
cjxj=cs.*xs;
djxj=ds.*xs;

us=0;   %rs(i)
vs=0;   %ss(i)
us(1:(s-1))=0;
vs(1:(s-1))=bs(s)-bs(1:(s-1));
us(s)=0;
vs(s)=0;
us((s+1):m)=bs((s+1):m)-bs(s);
vs((s+1):m)=0;
```

```
                giui=gs.*us;
                hivi=hs.*vs;
                puiui=pus.*us;
                qvivi=qvs.*vs;

                objU=sum(cjxj)+sum(giui)+sum(hivi);
                objL=sum(djxj)+sum(puiui)+sum(qvivi);
                obj=objU/objL;
                chk1=(-cs(k)/as(k)+sumg1-sumh1)-((-ds(k)/as(k)+sump1-
sumq1)*obj);
                chk2=(cs(k)/as(k)-sumg2+sumh2)-((ds(k)/as(k)-
sump2+sumq2)*obj);
                if (chk1>=0)&(chk2>=0)
                  Xs(j1,j2,j3,k)=xs(k);
                  Pxs(j1,j2,j3,k)=s;
                  Obj4xs(j1,j2,j3,k)=obj;
                else s=s+1;
                end
                if s>m
                  Xs(j1,j2,j3,k)=0;
                  Pxs(j1,j2,j3,k)=0;
                  Obj4xs(j1,j2,j3,k)=0;
                  chk1 = 1000000;
                  chk2 = 1000000;
                end
              end
            end
```

```
            else Xs(j1,j2,j3,k)=0;
               Pxs(j1,j2,j3,k)=0;
               Obj4xs(j1,j2,j3,k)=0;
            end
            if Xs(j1,j2,j3,k)>ts(k)| Xs(j1,j2,j3,k)<0 | Obj4xs(j1,j2,j3,k)<0
               Xs(j1,j2,j3,k)=0;
               Pxs(j1,j2,j3,k)=0;
               Obj4xs(j1,j2,j3,k)=0;
            end
         end
      end
   end
end

%Check k of j subscribs of X.
for j1=1:(n-2)
   for j2=(j1+1):(n-1)
      for j3=(j2+1):n
         for k=1:n
            ck3=0;  ck4=0;
            dk3=0;  dk4=0;
            if Xs(j1,j2,j3,k) ~= 0
               for jk=1:n
                  if jk==j1 |jk==j2|jk==j3
                     ck3=ck3+((as(jk)*cs(k)/as(k))-cs(jk));
                     dk3=dk3+(((as(jk)*ds(k)/as(k))-ds(jk))*Obj4xs(j1,j2,j3,k));
                  elseif jk==k
```

```
                    ck3=ck3;
                    dk3=dk3;
                    ck4=ck4;
                    dk4=dk4;
                 else ck4=ck4+(cs(jk)-(as(jk)*cs(k)/as(k)));
                    dk4=dk4+((ds(jk)-(as(jk)*ds(k)/as(k)))*Obj4xs(j1,j2,j3,k));
                 end
              end
              chk3=ck3-dk3;
              chk4=ck4-dk4;
              if chk3 < 0 | chk4 <0
                 Xs(j1,j2,j3,k)=0;
                 Pxs(j1,j2,j3,k)=0;
                 Obj4xs(j1,j2,j3,k)=0;
              end
           end
        end
     end
  end
end

%Check p of i subscribs of X.
for j1=1:(n-2)
  for j2=(j1+1):(n-1)
     for j3=(j2+1):n
        for k=1:n
           if Xs(j1,j2,j3,k) ~= 0
```

```
    ip=Pxs(j1,j2,j3,k);
    gh4=0;
    pq4=0;
    gh5=0;
    pq5=0;
    for rr=1:(ip-1)
        gh4=gh4+gs(rr)+hs(rr);
        pq4=pq4+((pus(rr)+qvs(rr))*Obj4xs(j1,j2,j3,k));
    end
    for ss=(ip+1):m
        gh5=gh5+gs(ss)+hs(ss);
        pq5=pq5+((pus(ss)+qvs(ss))*Obj4xs(j1,j2,j3,k));
    end
    chk4=gh4-pq4;
    chk5=gh5-pq5;
    if chk4 >= 0 & chk5 >= 0
        Xs(j1,j2,j3,k)=Xs(j1,j2,j3,k);
        Pxs(j1,j2,j3,k)=Pxs(j1,j2,j3,k);
        Obj1(j1,j2,j3,k)=Obj4xs(j1,j2,j3,k);
    else Xs(j1,j2,j3,k)=0;
        Pxs(j1,j2,j3,k)=0;
        Obj1(j1,j2,j3,k)=1000000;
    end
else Xs(j1,j2,j3,k)=0;
    Pxs(j1,j2,j3,k)=0;
    Obj1(j1,j2,j3,k)=1000000;
end
```

```
            end
        end
    end
end

MINZ4=100;
for j1=1:(n-2)
    for j2=(j1+1):(n-1)
        for j3=(j2+1):n
            for k=1:n
                if MINZ4>Obj1(j1,j2,j3,k)
                    MINZ4=Obj1(j1,j2,j3,k);
                    xmin=Xs(j1,j2,j3,k);
                    pmin=Pxs(j1,j2,j3,k);
                    pr1=j1;
                    pr2=j2;
                    pr3=j3;
                    pc=k;
                end
            end
        end
    end
end
if MINZ4 ~= 100
    for ja=1:n
        if ja==pr1 | ja==pr2 |ja==pr3
            X4(ja)=ts(ja);
```

```
            Objx(ja)=0;
        elseif ja==pc
            X4(ja)=xmin;
            Objx(ja)=MINZ4;
        else X4(ja)=0;
            Objx(ja)=0;
        end
    end
    us=0;   %rs(i)
    vs=0;   %ss(i)
    us(1:(pmin-1))=0;
    vs(1:(pmin-1))=bs(pmin)-bs(1:(pmin-1));
    us(pmin)=0;
    vs(pmin)=0;
    us((pmin+1):m)=bs((pmin+1):m)-bs(pmin);
    vs((pmin+1):m)=0;
    %arrange by original
    U4(f(1:m))=us(1:m);
    V4(f(1:m))=vs(1:m);
    Pi=find(U4==0 & V4==0);
else disp('Objective have no feasible solution in this 4 components level.')
    MINZ4=100;
    X4=ones(1,n);
    U4=ones(1,m);
    V4=ones(1,m);
end
%End of program.
```

```
%6 th subprogram.
%Function of the direct method formulate at the fifth step.
function [MINZ5,X5,U5,V5]=boundlevel5(as,b,cs,ds,g,h,m,n,pu,qv,ts,testx4)
[bs,f]=sort(b);
gs=g(f);
hs=h(f);
pus=pu(f);
qvs=qv(f);
testx5=zeros(n,n,n,n,n);

%To check upper level (five-components).
for j1=1:n
   ta1(j1)=ts(j1)*as(j1);
   for j2=1:n
      ta2(j2)=ts(j2)*as(j2);
      for j3=1:n
         ta3(j3)=ts(j3)*as(j3);
         for j4=1:n
            ta4(j4)=ts(j4)*as(j4);
            if testx4(j1,j2,j3,j4)~=0
               for k=1:n
                  if k~=j1 & k~=j2 & k~=j3 & k~=j4
                     testx5(j1,j2,j3,j4,k)=(bs(m)-ta1(j1)-ta2(j2)-ta3(j3)-ta4(j4))/as(k);
                     if testx5(j1,j2,j3,j4,k)<0
                        testx5(j1,j2,j3,j4,k)=0;
                     end
```

```
                  else testx5(j1,j2,j3,j4,k)=0;
                  end
                end
              end
            end
          end
        end
      end

%Find answer of X variable more than and equal to
for j1=1:(n-3)
  for j2=(j1+1):(n-2)
    for j3=(j2+1):(n-1)
      for j4=(j3+1):n
        for k=1:n
          if testx5(j1,j2,j3,j4,k)~=0
            chk1=-100;
            chk2=0;
            s=1;
            if (chk1<0)|(chk2<0)
              while (chk1<0)|(chk2<0)
                sumg1=0;  sump1=0;
                sumh1=0;  sumq1=0;
                sumg2=0;  sump2=0;
                sumh2=0;  sumq2=0;
                for i=s:m
                  sumg1=sumg1+gs(i);
```

```
    sump1=sump1+pus(i);
end
for i=1:(s-1)
    sumh1=sumh1+hs(i);
    sumq1=sumq1+qvs(i);
end
for i=(s+1):m
    sumg2=sumg2+gs(i);
    sump2=sump2+pus(i);
end
for i=1:s
    sumh2=sumh2+hs(i);
    sumq2=sumq2+qvs(i);
end
for js=1:n
    if js==k
        xs(js)=(bs(s)-ta1(j1)-ta2(j2)-ta3(j3)-ta4(j4))/as(k);
    elseif js==j1|js==j2|js==j3|js==j4
        xs(js)=ts(js);
    else xs(js)=0;
    end
end
cjxj=cs.*xs;
djxj=ds.*xs;

us=0;   %rs(i)
vs=0;   %ss(i)
```

```
us(1:(s-1))=0;
vs(1:(s-1))=bs(s)-bs(1:(s-1));
us(s)=0;
vs(s)=0;
us((s+1):m)=bs((s+1):m)-bs(s);
vs((s+1):m)=0;
giui=gs.*us;
hivi=hs.*vs;
puiui=pus.*us;
qvivi=qvs.*vs;

objU=sum(cjxj)+sum(giui)+sum(hivi);
objL=sum(djxj)+sum(puiui)+sum(qvivi);
obj=objU/objL;
chk1=(-cs(k)/as(k)+sumg1-sumh1)-((-ds(k)/as(k)+sump1-
sumq1)*obj);
chk2=(cs(k)/as(k)-sumg2+sumh2)-((ds(k)/as(k)-
sump2+sumq2)*obj);
if (chk1>=0)&(chk2>=0)
   Xs(j1,j2,j3,j4,k)=xs(k);
   Pxs(j1,j2,j3,j4,k)=s;
   Obj5xs(j1,j2,j3,j4,k)=obj;
else s=s+1;
end
if s>m
   Xs(j1,j2,j3,j4,k)=0;
   Pxs(j1,j2,j3,j4,k)=0;
```

```
                    Obj5xs(j1,j2,j3,j4,k)=0;
                    chk1 = 1000000;
                    chk2 = 1000000;
                  end
                end
              end
            else Xs(j1,j2,j3,j4,k)=0;
              Pxs(j1,j2,j3,j4,k)=0;
              Obj5xs(j1,j2,j3,j4,k)=0;
            end
            if Xs(j1,j2,j3,j4,k)>ts(k)| Xs(j1,j2,j3,j4,k)<0 | Obj5xs(j1,j2,j3,j4,k)<0
              Xs(j1,j2,j3,j4,k)=0;
              Pxs(j1,j2,j3,j4,k)=0;
              Obj5xs(j1,j2,j3,j4,k)=0;
            end
          end
        end
      end
    end
  end

%Check k of j subscribs of X.
for j1=1:(n-3)
  for j2=(j1+1):(n-2)
    for j3=(j2+1):(n-1)
      for j4=(j3+1):n
        for k=1:n
```

```
            ck3=0;  ck4=0;
            dk3=0;  dk4=0;
            if Xs(j1,j2,j3,k) ~= 0
               for jk=1:n
                  if jk==j1 |jk==j2|jk==j3
                     ck3=ck3+((as(jk)*cs(k)/as(k))-cs(jk));
                     dk3=dk3+(((as(jk)*ds(k)/as(k))-ds(jk))*Obj5xs(j1,j2,j3,j4,k));
                  elseif jk==k
                     ck3=ck3;
                     dk3=dk3;
                     ck4=ck4;
                     dk4=dk4;
                  else ck4=ck4+(cs(jk)-(as(jk)*cs(k)/as(k)));
                     dk4=dk4+((ds(jk)-(as(jk)*ds(k)/as(k)))*Obj5xs(j1,j2,j3,j4,k));
                  end
               end
               chk3=ck3-dk3;
               chk4=ck4-dk4;
               if chk3 < 0 | chk4 <0
                  Xs(j1,j2,j3,j4,k)=0;
                  Pxs(j1,j2,j3,j4,k)=0;
                  Obj4xs(j1,j2,j3,j4,k)=0;
               end
            end
         end
      end
   end
```

```
  end
end

%Check p of i subscribs of X.
for j1=1:(n-3)
  for j2=(j1+1):(n-2)
    for j3=(j2+1):(n-1)
      for j4=(j3+1):n
        for k=1:n
          if Xs(j1,j2,j3,j4,k) ~= 0
            ip=Pxs(j1,j2,j3,j4,k);
            gh4=0;
            pq4=0;
            gh5=0;
            pq5=0;
            for rr=1:(ip-1)
              gh4=gh4+gs(rr)+hs(rr);
              pq4=pq4+((pus(rr)+qvs(rr))*Obj5xs(j1,j2,j3,j4,k));
            end
            for ss=(ip+1):m
              gh5=gh5+gs(ss)+hs(ss);
              pq5=pq5+((pus(ss)+qvs(ss))*Obj5xs(j1,j2,j3,j4,k));
            end
            chk4=gh4-pq4;
            chk5=gh5-pq5;
            if chk4 >= 0 & chk5 >= 0
              Xs(j1,j2,j3,j4,k)=Xs(j1,j2,j3,j4,k);
```

```
                Pxs(j1,j2,j3,j4,k)=Pxs(j1,j2,j3,j4,k);
                Obj1(j1,j2,j3,j4,k)=Obj5xs(j1,j2,j3,j4,k);
             else Xs(j1,j2,j3,j4,k)=0;
                Pxs(j1,j2,j3,j4,k)=0;
                Obj1(j1,j2,j3,j4,k)=1000000;
             end
          else Xs(j1,j2,j3,j4,k)=0;
             Pxs(j1,j2,j3,j4,k)=0;
             Obj1(j1,j2,j3,j4,k)=1000000;
          end
       end
    end
  end
 end
end

MINZ5=100;
for j1=1:(n-3)
 for j2=(j1+1):(n-2)
  for j3=(j2+1):(n-1)
    for j4=(j3+1):n
       for k=1:n
          if MINZ5>Obj1(j1,j2,j3,j4,k)
             MINZ5=Obj1(j1,j2,j3,j4,k);
             xmin=Xs(j1,j2,j3,j4,k);
             pmin=Pxs(j1,j2,j3,j4,k);
             pr1=j1;
```

```
                pr2=j2;
                pr3=j3;
                pr4=j4;
                pc=k;
              end
            end
          end
        end
      end
    end
    if MINZ5 ~= 100
      for ja=1:n
        if ja==pr1 | ja==pr2 |ja==pr3|ja==j4
          X5(ja)=ts(ja);
          Objx(ja)=0;
        elseif ja==pc
          X5(ja)=xmin;
          Objx(ja)=MINZ5;
        else X5(ja)=0;
          Objx(ja)=0;
        end
      end
      us=0;   %rs(i)
      vs=0;   %ss(i)
      us(1:(pmin-1))=0;
      vs(1:(pmin-1))=bs(pmin)-bs(1:(pmin-1));
      us(pmin)=0;
```

```
    vs(pmin)=0;
    us((pmin+1):m)=bs((pmin+1):m)-bs(pmin);
    vs((pmin+1):m)=0;
    %arrange by original
    U5(f(1:m))=us(1:m);
    V5(f(1:m))=vs(1:m);
    Pi=find(U5==0 & V5==0);
else disp('Objective have no feasible solution in this 5 components level.')
    MINZ5=100;
    X5=ones(1,n);
    U5=ones(1,m);
    V5=ones(1,m);
end
%End of program.
```