รายการอ้างอิง

1. Bazaraa, M. S., and others, Linear Programming And Network Flows. New York: John Wiley & Sons, 1990.
2. Bronson, R. Matrix Methods. New York: Academic Press, 1968.
3. Chong, E. K. P., and Stanislaw Z. H. An Introduction To Optimization. New York: John Wiley & Sons, 2001.
4. Gass, S. I. Linear Programming. 5th ed. New York: McGraw-Hill Book Company, 1994.
5. Klee, V., and Minty, G. J. How Good Is The Simplex Algorithms? In Inequalities. O. Shisha ed. pp. 159-175. New York: Academic Press,1972.
6. Lancaster, P. Theory of Matrices. New York: Academic Press, 1969.
7. Nocedal, J., and Wright, S. J. Numerical Optimization. New York: Springer, 1999.
8. Swanson L. W. Linear Programming. Singapore: McGraw-Hill Book Company, 1987.
9. Wu, N., and Coppins R. Linear Programming. New York: McGraw-Hill BookCompany, 1981.
10. Schnieider, H., and Barker, G. P. Matrices And Linear Algebra. New York: Holt Rinehart and Winston, 1968.

ภาคผนวก

ภาคผนวก ก

ภาคผนวก ก จะกล่าวถึงปัญหากำหนดการเชิงเส้นในสองมิติที่นำไปทดสอบขั้นตอนวิธีใหม่ซึ่งประกอบไปด้วยปัญหา LP01 LP02 และ LP03 ในแต่ละปัญหาจะมีสมบัติต่างกันดังนี้

LP01: ปัญหาที่มีผลเฉลยที่เหมาะที่สุดเพียงจุดเดียว หรือมากกว่าหนึ่งจุด และมีเงื่อนไขบังคับจำนวนมาก

LP02: ปัญหาที่มีผลเฉลยที่เหมาะที่สุดเพียงจุดเดียว หรือมากกว่าหนึ่งจุด มีจุดสุดขีด (extreme point) จำนวนมาก และมีเงื่อนไขบังคับจำนวนมาก

LP03: ปัญหาที่มีผลเฉลยที่เหมาะที่สุดเพียงจุดเดียว หรือมากกว่าหนึ่งจุด มีจุดสุดขีดจำนวนมาก และมีเงื่อนไขบังคับจำนวนมาก และเงื่อนไขบังคับบางชุดมีเวกเตอร์เกรเดียนต์ชี้ไปในทิศทางเดียวกัน

LP01: ปัญหาในกลุ่มนี้มี 7 ปัญหาที่นำมาทดสอบซึ่งในแต่ละปัญหาอยู่ในรูปแบบดังต่อไปนี้

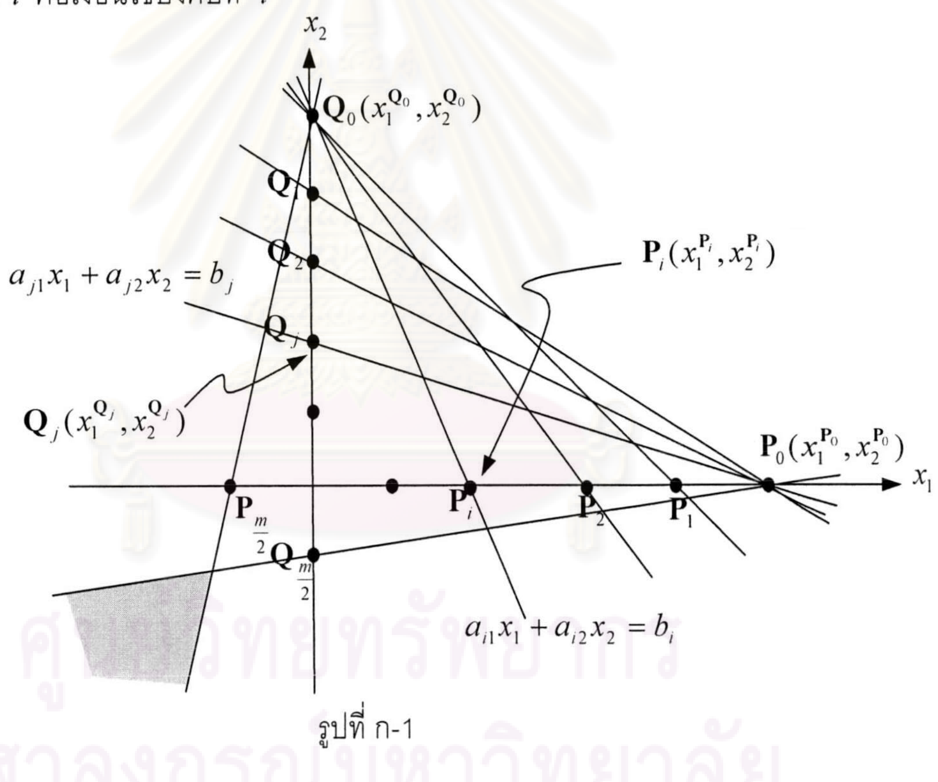
การหาค่าสูงสุดของฟังก์ชันจุดประสงค์ $z = x_1 + x_2$

ภายใต้เงื่อนไข

$$
\begin{array}{ccccc}
a_{11}x_1 & + & a_{12}x_2 & \leq & b_1 \\
a_{21}x_1 & + & a_{22}x_2 & \leq & b_2 \\
 & & & \vdots & \\
a_{m1}x_1 & + & a_{m2}x_2 & \leq & b_m
\end{array}
$$

เมื่อ $m =$ 2,000 4,000 6,000 8,000 10,000 12,000 และ 14,000 คือจำนวนเงื่อนไขบังคับของปัญหาทั้ง 7 ตามลำดับ

โดยที่ a_{i1}, a_{i2} และ b_i สัมพันธ์กับสมการเส้นตรง $a_{i1}x_1 + a_{i2}x_2 = b_i$ ที่สร้างในลักษณะดังรูปที่ ก-1 เมื่อ i คือเงื่อนไขบังคับที่ i



รูปที่ ก-1

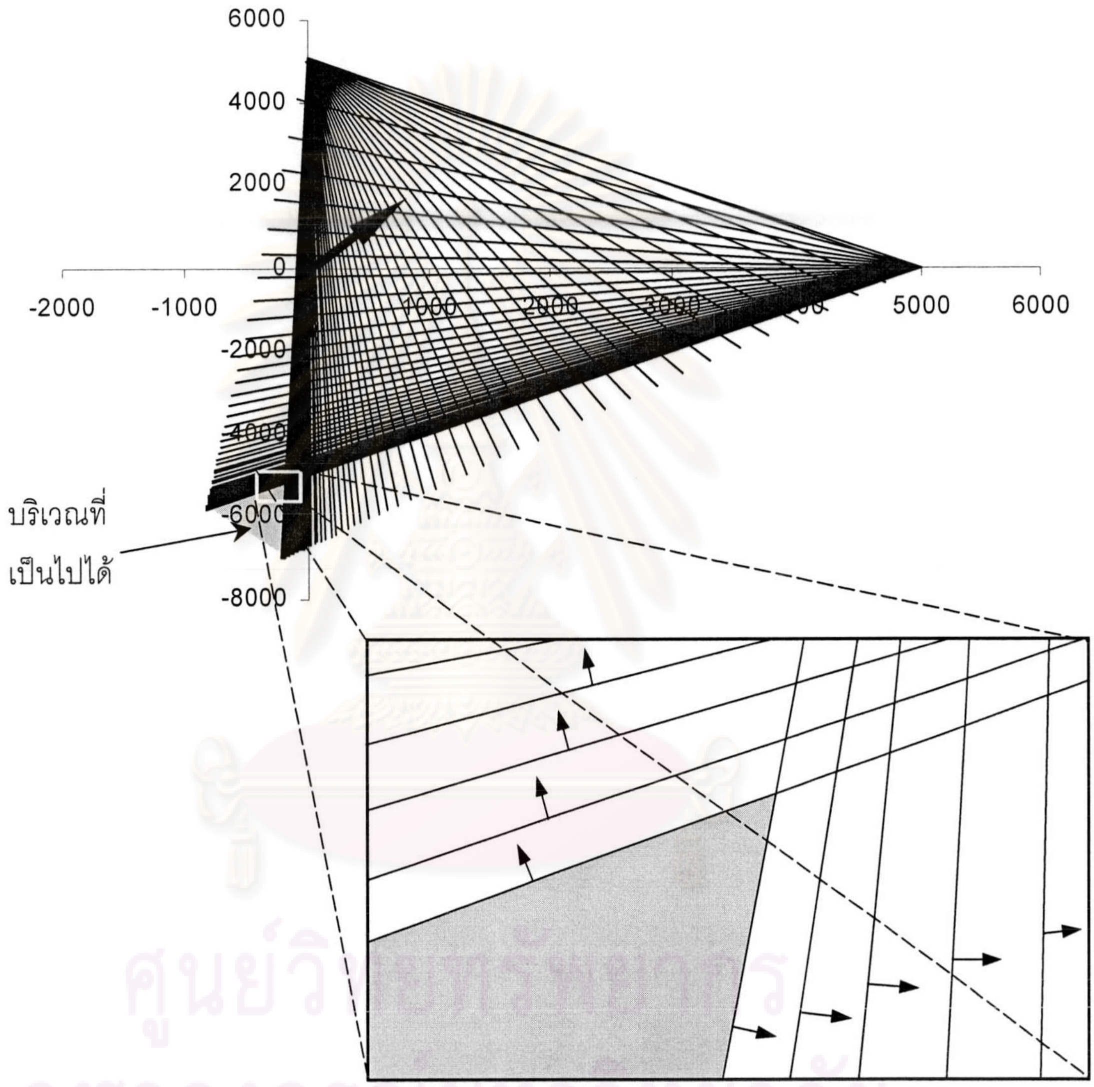
กำหนดให้ m เป็นจำนวนเต็มบวกคู่ และ $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, \cdots, \mathbf{P}_{\frac{m}{2}}$ และ $\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2, \cdots, \mathbf{Q}_{\frac{m}{2}}$ เป็นจุดที่กำหนดให้ สำหรับเงื่อนไขบังคับที่ $i = 1, 2, \cdots, \frac{m}{2}$ จะได้ว่า

$$b_i = (x_2^{\mathbf{P}_i} - x_2^{\mathbf{Q}_0})x_1^{\mathbf{P}_i} - (x_1^{\mathbf{P}_i} - x_1^{\mathbf{Q}_0})x_2^{\mathbf{P}_i}, \quad a_{i1} = x_2^{\mathbf{P}_i} - x_2^{\mathbf{Q}_0}, \quad a_{i2} = x_1^{\mathbf{Q}_0} - x_1^{\mathbf{P}_i}$$

และในทำนองเดียวกันสำหรับเงื่อนไขบังคับที่ $j = 1, 2, \cdots, \frac{m}{2}$ จะได้ว่า

$$b_j = (x_2^{\mathbf{Q}_j} - x_2^{\mathbf{P}_0})x_1^{\mathbf{Q}_j} - (x_1^{\mathbf{Q}_j} - x_1^{\mathbf{P}_0})x_2^{\mathbf{Q}_j}, \quad a_{j1} = x_2^{\mathbf{Q}_j} - x_2^{\mathbf{P}_0}, \quad a_{j2} = x_1^{\mathbf{P}_0} - x_1^{\mathbf{Q}_j}$$

และบริเวณที่เป็นไปได้ของปัญหาอยู่ในลักษณะดังรูปที่ ก-2 ต่อไปนี้



รูปที่ ก-2

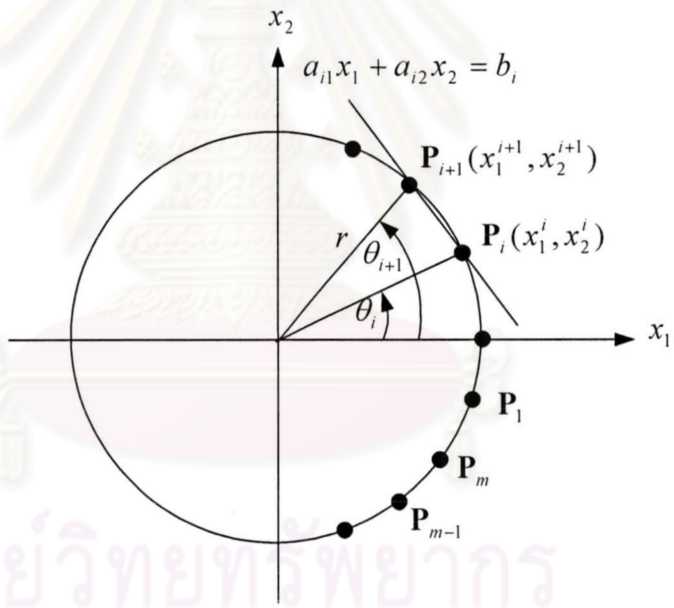
LP02: ปัญหาในกลุ่มนี้มี 7 ปัญหาที่นำมาทดสอบซึ่งในแต่ละปัญหาอยู่ในรูปแบบดังต่อไปนี้

การหาค่าสูงสุดของฟังก์ชันจุดประสงค์ $z = x_1 + x_2$

ภายใต้เงื่อนไข

$$\begin{array}{ccccc} a_{11}x_1 & + & a_{12}x_2 & \le & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & \le & b_2 \\ & & & \vdots & \\ a_{m1}x_1 & + & a_{m2}x_2 & \le & b_m \end{array}$$

เมื่อ $m = 2{,}000$ $4{,}000$ $6{,}000$ $8{,}000$ $10{,}000$ $12{,}000$ และ $14{,}000$ คือจำนวนเงื่อนไขบังคับของปัญหาทั้ง 7 ตามลำดับ โดยที่ a_{i1}, a_{i2} และ b_i คือค่าที่สัมพันธ์กับสมการเส้นตรง $a_{i1}x_1 + a_{i2}x_2 = b_i$ ซึ่งสร้างจากจุด $\mathbf{P}_i, \mathbf{P}_{i+1}$ ที่อยู่บนวงกลมรัศมี $r = 200$ หน่วยดังรูปที่ก-3



รูปที่ ก-3

โดยที่ $x_1^i = r\cos(\theta_i),\ x_2^i = r\sin(\theta_i)$

ดังนั้นจะได้ว่า

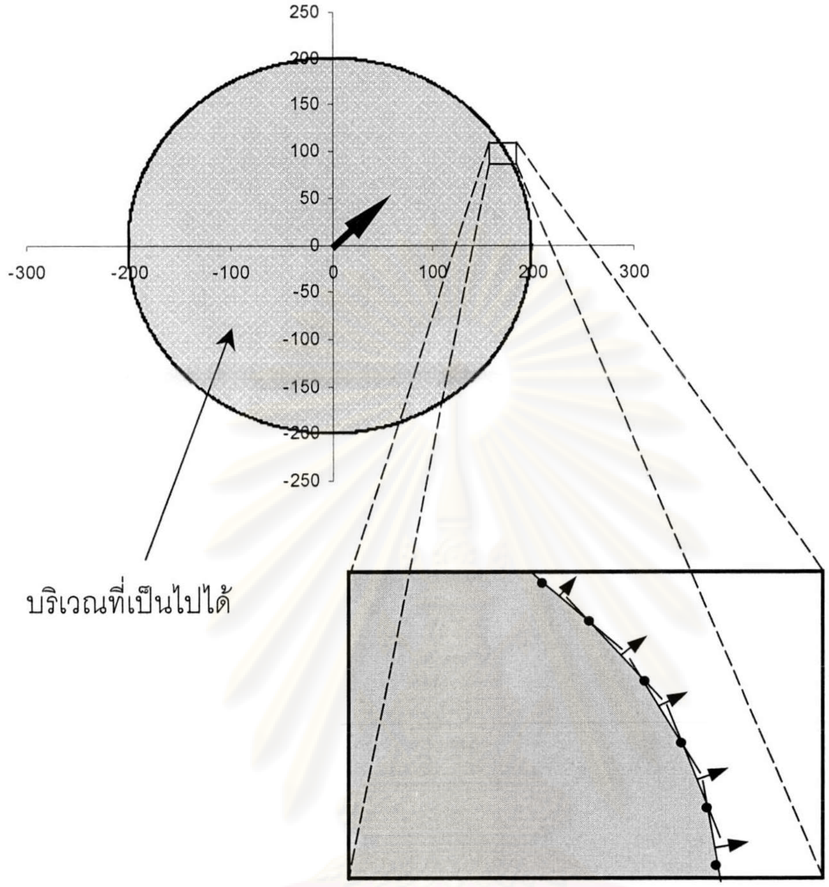
กรณี $x_1^i \ne x_1^{i+1}$ จะได้ว่า $b_i = (x_2^{i+1} - x_2^i)x_1^i - (x_1^{i+1} - x_1^i)x_2^i$

$a_{i1} = x_2^{i+1} - x_2^i$

$a_{i2} = x_1^i - x_1^{i+1}$

กรณี $x_1^i = x_1^{i+1}$ จะได้ว่า $b_i = x_1^i,\ a_{i1} = 1,\ a_{i2} = 0$

และบริเวณที่เป็นไปได้ของปัญหาอยู่ในลักษณะดังรูปที่ ก-4



รูปที่ ก-4

LP03: ปัญหาในกลุ่มนี้มี 7 ปัญหาที่นำมาทดสอบซึ่งในแต่ละปัญหาอยู่ในรูปแบบดังต่อไปนี้

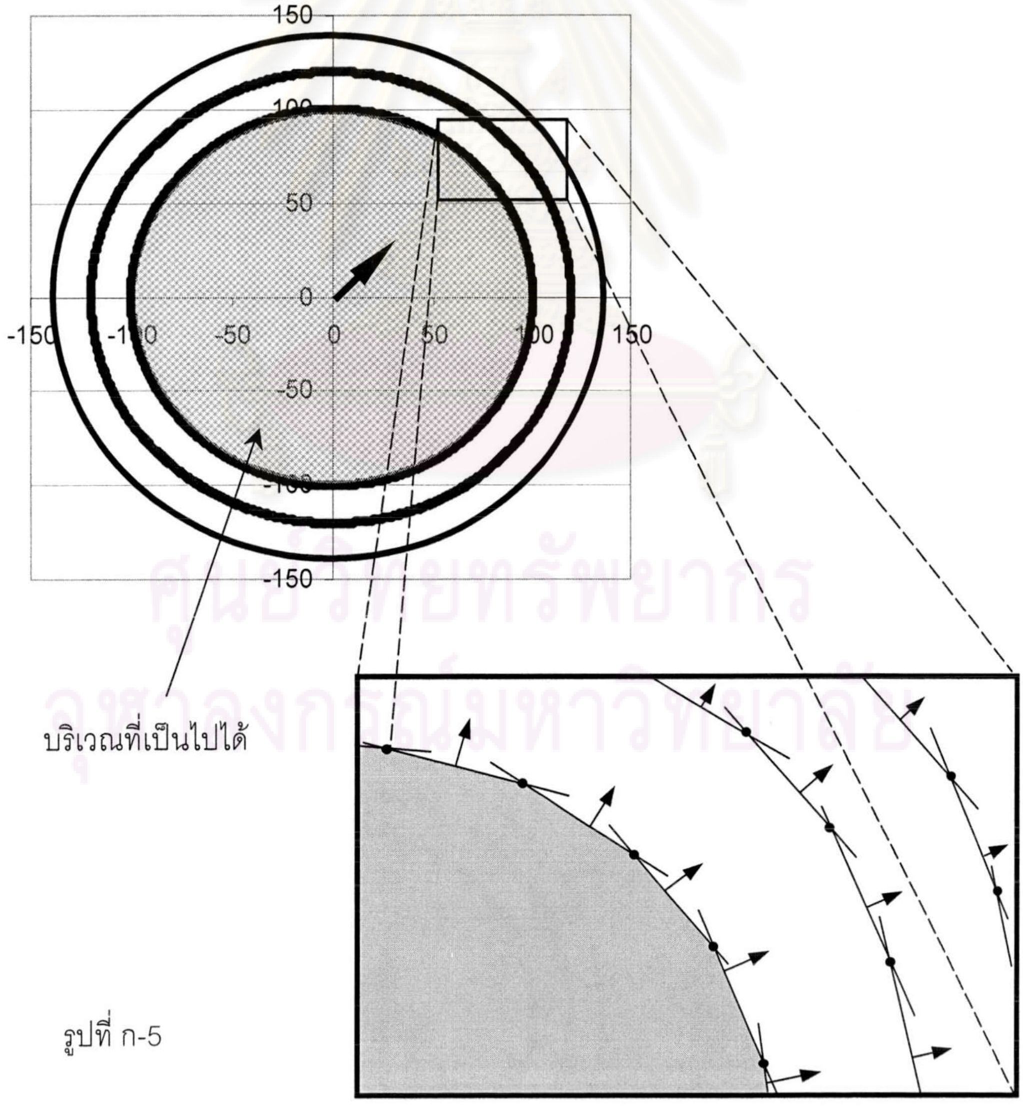
การหาค่าสูงสุดของฟังก์ชันจุดประสงค์ $z = x_1 + x_2$

ภายใต้เงื่อนไข

$$\begin{array}{ccccc} a_{11}x_1 & + & a_{12}x_2 & \leq & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & \leq & b_2 \\ & & & \vdots & \\ a_{m1}x_1 & + & a_{m2}x_2 & \leq & b_m \end{array}$$

โดยที่ a_{i1}, a_{i2} และ b_i สร้างในลักษณะเดียวกันกับปัญหา LP02 แต่วงกลมที่ช่วยในการสร้างมีสามวงคือ $r_1 = 100,\ r_2 = 130$ และ $r_3 = 150$ ดังรูป

เมื่อ $m =$ 2,000 3,000 4,000 5,000 6,000 7,000 และ 8,000 คือจำนวนเงื่อนไขบังคับของปัญหาทั้ง 7 ตามลำดับ และในแต่ละปัญหามีบริเวณที่เป็นไปได้อยู่ในลักษณะดังรูปที่ ก-5



รูปที่ ก-5

ภาคผนวก ข

Redundant Constraint

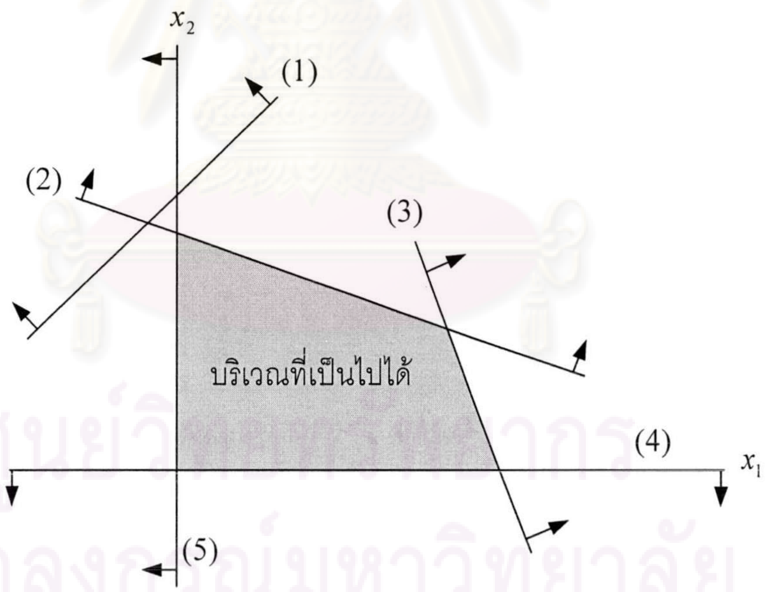
ให้ $X = \{\mathbf{x} | \mathbf{Ax} \le \mathbf{b}\}$ คือเซตของผลเฉลยที่เป็นไปได้ของปัญหากำหนดการเชิงเส้น หรือเรียกว่าเป็นบริเวณที่เป็นไปได้(feasible region)

กำหนดให้ $\mathbf{A}^{\sim}$ คือเมทริกซ์ที่เกิดจากการตัดเวกเตอร์ $\mathbf{A}_{i*}$ ของ $\mathbf{A}$ ออก ขณะที่ $\mathbf{b}^{\sim}$ คือเวกเตอร์ที่ตัด b_i ออกจาก $\mathbf{b}$ และ $X^{\sim} = \{\mathbf{x} | \mathbf{A}^{\sim}\mathbf{x} \le \mathbf{b}^{\sim}\}$

ถ้า $X = X^{\sim}$ แล้วเราจะเรียกเงื่อนไข i ว่าเป็น redundant constraint

กล่าวคือ redundant constraint ของปัญหากำหนดการเชิงเส้น คือเงื่อนไขบังคับที่เมื่อตัดออกแล้วไม่มีผลกระทบต่อบริเวณที่เป็นไปได้ดังเช่นตัวอย่างต่อไปนี้

ตัวอย่าง



รูปที่ ข-1

จากรูปที่ ข-1 จะได้ว่าถ้าตัดเงื่อนไขบังคับ (1) ออกแล้วจะได้ว่าบริเวณที่เป็นไปได้ยังคงเดิมหมายความว่าเงื่อนไขบังคับ (1) เป็น redundant constraint

การวัดเวลาการประมวลคำสั่งของซีพียูด้วยคำสั่งภาษาซี

เวลาที่เครื่องคอมพิวเตอร์ใช้ในการประมวลผลคำสั่งทั้งโปรแกรมเรียกว่าเป็น *processor time* และในการวัดระยะเวลาดังกล่าวจะเรียกใช้ฟังก์ชัน clock() ซึ่งอยู่ในฟังก์ชันมาตรฐานที่ชื่อว่า time.h และถ้ามีการเรียกใช้ฟังก์ชัน clock() แล้วจะให้ค่าข้อมูลกลับคืนมาเป็นข้อมูลตัวเลขชนิด clock_t ซึ่งเป็นข้อมูลชนิด long integer ซึ่งเราจะประกาศตัวแปร start และ end เป็นตัวแปรชนิด clock_t เพื่อเก็บข้อมูลตัวเลขเริ่มต้น และสุดท้ายของการประมวลคำสั่ง และเราคำนวณ *cpu time* ที่ได้จาก start และ end ดังนี้

elapsed = ((double)(end - start))/ CLOCKS_PER_SEC

เมื่อ elapsed คือตัวแปรชนิด double precision มีหน้าที่เก็บค่า *cpu time* และ CLOCKS_PER_SEC คือจำนวนสัญญานนาฬิกาต่อหนึ่งวินาทีซึ่งเป็นค่าคงที่

ขบวนการวัด *cpu time* ทั้งหมดที่กล่าวมาสามารถเขียนให้อยู่ในรูปแบบคำสั่งภาษาซีได้ดังนี้

```
#include <time.h>
clock_t start, end;
double elapsed;

start = clock();     /* จับเวลาเริ่มต้นของการทำงาน
⋮                    /*  คำสั่งในการทำงานที่ต้องการวัดเวลา
end = clock();      /* จับเวลาสุดท้ายของการทำงาน
elapsed = ((double)(end - start))/  CLOCKS_PER_SEC;   /* คำนวณ cpu time
```

GLPK(GNU Linear Programming Kit)

GLPK คือกลุ่มของชุดคำสั่งภาษาซีที่บรรจุไว้ในไฟล์คลังโปรแกรม ซึ่งผู้ใช้สามารถเรียกใช้ได้ ชุดคำสั่งดังกล่าวสร้างขึ้นมาเพื่อวัตถุประสงค์ในการแก้ปัญหากำหนดการเชิงเส้น(LP) ปัญหา mixed integer programming(MIP) และปัญหาอื่นๆที่เกี่ยวกับปัญหากำหนดการเชิงเส้น

สำหรับซอฟต์แวร์ติดตั้ง GLPK ดาวน์โหลดได้จากเวบไซต์

http://www.gnu.org/directory/glpk.html

สำหรับการติดตั้ง ตัวอย่างและคำอธิบายชุดคำสั่ง GLPK สามารถอ่านได้จากคู่มือการติดตั้ง ซึ่งดาวน์โหลดได้จากเวบไซต์ http://ftp.gnu.org/gnu/glpk

พิจารณาปัญหากำหนดการเชิงเส้นที่อยู่ในรูปแบบดังต่อไปนี้

การหาค่าสูงสุดของฟังก์ชันจุดประสงค์ $z = c_1x_1 + c_2x_2 + \cdots + c_nx_n$

ภายใต้เงื่อนไข

$$\begin{array}{ccccccccc}
a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & \le & b_1 \\
a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & \le & b_2 \\
 & & & & & & & \vdots & \\
a_{m1}x_1 & + & a_{m2}x_2 & + & \cdots & + & a_{mn}x_n & \le & b_m
\end{array}$$

$$\begin{array}{ccccc}
l_1 & \le & x_1 & \le & u_1 \\
l_2 & \le & x_2 & \le & u_2 \\
 & & & \vdots & \\
l_n & \le & x_n & \le & u_n
\end{array}$$

เมื่อ z คือฟังก์ชันจุดประสงค์

x_j คือตัวแปรอิสระตัวที่ j เมื่อ $1 \le j \le n$

c_j คือสัมประสิทธิ์ของ x_j ในฟังก์ชันจุดประสงค์เมื่อ $1 \le j \le n$

a_{ij} คือสัมประสิทธิ์ของ x_j ในเงื่อนไขบังคับที่ i เมื่อ $1 \le i \le m$ และ $1 \le j \le n$

b_i คือค่าคงที่ด้านขวามือของเงื่อนไขบังคับที่ i เมื่อ $1 \le i \le m$

l_j และ u_j คือค่าขอบเขตล่างและขอบเขตบนของ x_j ตามลำดับ เมื่อ $1 \le j \le n$

สำหรับตัวแปรอิสระ x_j ใดๆ อาจจะมีค่าขอบเขตล่างและขอบเขตบนเป็นจำนวนจริงทั้งคู่ หรืออย่างใดอย่างหนึ่ง ในกรณีที่ขอบเขตล่างไม่ได้กำหนดหมายความว่าเราสามารถลดค่าของตัวแปร x_j ลงได้อย่างไม่มีขอบเขต ทำนองเดียวกันถ้าขอบเขตบนไม่ได้กำหนดหมายความว่าเรา

สามารถเพิ่มค่าของตัวแปร x_j ขึ้นได้อย่างไม่มีขอบเขตเช่นกัน ดังนั้นตัวแปร x_j อาจจะมีค่าขอบเขตเป็นอย่างใดอย่างหนึ่งดังต่อไปนี้

$$
\begin{array}{ccccc}
-\infty & < & x_j & < & +\infty \\
l_j & \le & x_j & < & +\infty \\
-\infty & < & x_j & \le & u_j \\
l_j & \le & x_j & \le & u_j \\
l_j & = & x_j & = & u_j
\end{array}
$$

การแก้ปัญหากำหนดการเชิงเส้นด้วย GLPK เริ่มต้นด้วยการสร้างชุดคำสั่งภาษาซีซึ่งเป็นชุดคำสั่งหลัก โดยที่ในชุดคำสั่งหลักดังกล่าวจะเรียกใช้ชุดคำสั่งของ GLPK หลังจากเขียนชุดคำสั่งถูกต้องแล้ว เราจะสั่งให้ตัวแปลภาษาซี (C compiler) แปลโปรแกรมและเมื่อสั่งให้โปรแกรมประมวลผลชุดคำสั่งดังกล่าวจะได้ผลเฉลยของปัญหากำหนดการเชิงเส้นดังตัวอย่างต่อไปนี้

การหาค่าสูงสุดของฟังก์ชันจุดประสงค์ $z = 10x_1 + 6x_2 + 4x_3$

ภายใต้เงื่อนไข

$$
\begin{array}{ccccccc}
x_1 & + & x_2 & + & x_3 & \le & 100 \\
10x_1 & + & 4x_2 & + & 5x_3 & \le & 600 \\
2x_1 & + & 2x_2 & + & 6x_3 & \le & 300 \\
 & & & & x_1, x_2, x_3 & \ge & 0
\end{array}
$$

จากปัญหาดังกล่าว กำหนดให้

$$
\begin{aligned}
p &= x_1 + x_2 + x_3 \\
q &= 10x_1 + 4x_2 + 5x_3 \\
r &= 2x_1 + 2x_2 + 6x_3
\end{aligned}
$$

เมื่อ p, q และ r คือตัวแปรที่กำหนดขึ้นมาเพื่อความสะดวกในการอ้างอิง

ดังนั้นจะได้ว่า

$$
\begin{array}{ccccc}
-\infty & < & p & \le & 100 \\
-\infty & < & q & \le & 600 \\
-\infty & < & r & \le & 300
\end{array}
$$

และ

$$
\begin{aligned}
0 &\leq x_1 < +\infty \\
0 &\leq x_2 < +\infty \\
0 &\leq x_3 < +\infty
\end{aligned}
$$

จากข้อมูลของปัญหากำหนดการเชิงเส้นที่ได้ เมื่อนำไปสร้างเป็นชุดคำสั่งภาษาซี พร้อมกับการเรียกใช้ชุดคำสั่งของ GLPK

```
/* sample.c */

#include <stdio.h>
#include <stdlib.h>
#include "glpk.h"

int main(void)
{    LPX *lp;
     int rn[1+9], cn[1+9];
     double a[1+9], Z, x1, x2, x3;

s1:  lp = lpx_create_prob();
s2:  lpx_set_prob_name(lp, "sample");

s3:  lpx_add_rows(lp, 3);

s4:  lpx_set_row_name(lp, 1, "p");
s5:  lpx_set_row_bnds(lp, 1, LPX_UP, 0.0, 100.0);
s6:  lpx_set_row_name(lp, 2, "q");
s7:  lpx_set_row_bnds(lp, 2, LPX_UP, 0.0, 600.0);
s8:  lpx_set_row_name(lp, 3, "r");
s9:  lpx_set_row_bnds(lp, 3, LPX_UP, 0.0, 300.0);

s10: lpx_add_cols(lp, 3);

s11: lpx_set_col_name(lp, 1, "x1");
s12: lpx_set_col_bnds(lp, 1, LPX_LO, 0.0, 0.0);
s13: lpx_set_col_name(lp, 2, "x2");
s14: lpx_set_col_bnds(lp, 2, LPX_LO, 0.0, 0.0);
s15: lpx_set_col_name(lp, 3, "x3");
s16: lpx_set_col_bnds(lp, 3, LPX_LO, 0.0, 0.0);

s17: rn[1] = 1, cn[1] = 1, a[1] =  1.0;
s18: rn[2] = 1, cn[2] = 2, a[2] =  1.0;
s19: rn[3] = 1, cn[3] = 3, a[3] =  1.0;
s20: rn[4] = 2, cn[4] = 1, a[4] = 10.0;
s21: rn[5] = 3, cn[5] = 1, a[5] =  2.0;
s22: rn[6] = 2, cn[6] = 2, a[6] =  4.0;
s23: rn[7] = 3, cn[7] = 2, a[7] =  2.0;
s24: rn[8] = 2, cn[8] = 3, a[8] =  5.0;
s25: rn[9] = 3, cn[9] = 3, a[9] =  6.0;
s26: lpx_load_mat3(lp, 9, rn, cn, a);

s27: lpx_set_obj_dir(lp, LPX_MAX);

s28: lpx_set_col_coef(lp, 1, 10.0);
s29: lpx_set_col_coef(lp, 2, 6.0);
```

```
s30: lpx_set_col_coef(lp, 3, 4.0);

s31: lpx_simplex(lp);

s32: Z = lpx_get_obj_val(lp);
s33: lpx_get_col_info(lp, 1, NULL, &x1, NULL);
s34: lpx_get_col_info(lp, 2, NULL, &x2, NULL);
s35: lpx_get_col_info(lp, 3, NULL, &x3, NULL);

s36: printf("\nZ = %g; x1 = %g; x2 = %g; x3 = %g\n", Z, x1, x2, x3);

s37: lpx_delete_prob(lp);

    return 0;
}
```

/* eof */บรรทัด s1 เรียกใช้ชุดคำสั่งของ GLPK คือ lpx_create_prob ซึ่งเป็นคำสั่งในการสร้างโครงสร้างข้อมูลของปัญหากำหนดการเชิงเส้น

บรรทัด s2 เรียกใช้ชุดคำสั่ง lpx_set_prob_name ซึ่งเป็นคำสั่งในการกำหนดชื่อสัญลักษณ์(symbolic name) ของปัญหากำหนดการเชิงเส้น โดยในตัวอย่างเป็นการกำหนดให้ lp มีชื่อสัญลักษณ์เป็น “sample”

บรรทัด s3 เรียกใช้ชุดคำสั่ง lpx_add_rows ซึ่งเป็นการเพิ่มจำนวนแถว (row) เข้าไปในตัวโครงสร้างของปัญหา โดยในตัวอย่างเป็นการเพิ่มแถวจำนวน 3 แถวเข้าไปใน lp

บรรทัด s4 เรียกใช้ชุดคำสั่ง lpx_set_row_name ซึ่งเป็นคำสั่งในการกำหนดชื่อสัญลักษณ์ของแถว โดยในตัวอย่างเป็นการกำหนดให้แถวแรกชื่อ “p”

บรรทัด s5 เรียกใช้ชุดคำสั่ง lpx_set_row_bnds ซึ่งเป็นคำสั่งที่ใช้ระบุประเภทขอบเขตของตัวแปรที่สัมพันธ์กับแถวดังกล่าวว่ามีขอบเขตเป็นขอบเขตบนหรือขอบเขตล่าง พร้อมทั้งระบุค่าขอบเขตด้วย ในตัวอย่างมีการกำหนดขอบเขตของแถวแรกเป็นแบบ LPX_UP ซึ่งหมายถึงแถวแรกมีขอบเขตบนและค่าขอบเขตบนคือ 100.0

บรรทัด s6 และ s7 เป็นการเรียกใช้คำสั่งในทำนองเดียวกันกับบรรทัด s4 และ s5 ตามลำดับแต่เป็นการกำหนดชื่อสัญลักษณ์ให้กับแถวที่สองโดยให้ชื่อว่า “q” และมีขอบเขตบนคือ 600.0

บรรทัด s8 และ s9 เป็นการเรียกใช้คำสั่งในทำนองเดียวกันกับบรรทัด s4 และ s5 ตามลำดับแต่เป็นการกำหนดชื่อสัญลักษณ์ให้กับแถวที่สามโดยให้ชื่อว่า “r” และมีขอบเขตบนคือ 300.0

บรรทัด s10 เรียกใช้ชุดคำสั่ง lpx_add_cols ซึ่งเป็นการเพิ่มจำนวนสดมภ์ (column) เข้าไปในตัวโครงสร้างข้อมูลของปัญหา โดยในตัวอย่างเป็นการเพิ่มสดมภ์จำนวน 3 สดมภ์เข้าไปใน lp

บรรทัด s11 เรียกใช้ชุดคำสั่ง lpx_set_col_name ซึ่งเป็นคำสั่งในการกำหนดชื่อสัญลักษณ์ของสดมภ์ โดยในตัวอย่างเป็นการกำหนดให้สดมภ์แรกชื่อ “x1”

บรรทัด s12 เรียกใช้ชุดคำสั่ง lpx_set_col_bnds ซึ่งเป็นคำสั่งที่ใช้ระบุประเภทขอบเขตของตัวแปรที่สัมพันธ์กับสดมภ์ดังกล่าวว่ามีขอบเขตเป็นขอบเขตบนหรือขอบเขตล่าง พร้อมทั้งระบุค่าขอบเขตด้วย ในตัวอย่างมีการกำหนดขอบเขตของสดมภ์แรกเป็นแบบ LPX_LO ซึ่งหมายถึงสดมภ์แรกมีขอบเขตล่าง และค่าขอบเขตล่างคือ 0.0

บรรทัด s13 และ s14 เป็นการเรียกใช้คำสั่งในทำนองเดียวกันกับบรรทัด s11 และ s12 ตามลำดับ แต่เป็นการกำหนดชื่อสัญลักษณ์ให้กับสดมภ์ที่สองโดยให้ชื่อว่า “x2” และมีขอบเขตล่างเป็น 0.0

บรรทัด s15 และ s16 เป็นการเรียกใช้คำสั่งในทำนองเดียวกันกับบรรทัด s11 และ s12 ตามลำดับ แต่เป็นการกำหนดชื่อสัญลักษณ์ให้กับสดมภ์ที่สามโดยให้ชื่อว่า “x3” และมีขอบเขตล่างเป็น 0.0

บรรทัด s17 ถึง s25 เป็นการกำหนดค่าของสัมประสิทธิ์ของตัวแปรที่ปรากฏในเงื่อนไขบังคับ ซึ่งสัมประสิทธิ์ดังกล่าวถูกมองให้อยู่ในรูปของเมทริกซ์ ดังนั้นการกำหนดค่าสัมประสิทธิ์จึงมีการอ้างอิงตำแหน่งของสมาชิกเมทริกซ์ โดยที่แถวลำดับ (array) rn จะเก็บตำแหน่งของแถว และแถวลำดับ cn จะเก็บตำแหน่งของสดมภ์ ในขณะที่แถวลำดับ a จะเก็บค่าสัมประสิทธิ์ ณ ตำแหน่งแถวและสดมภ์ที่อ้างอิงโดย rn และ cn

บรรทัด s26 เรียกใช้ชุดคำสั่ง lpx_load_mat3 ซึ่งเป็นคำสั่งในการนำค่าจากแถวลำดับ rn, cn และ a ไปสู่โครงสร้างข้อมูลของปัญหา

บรรทัด s27 เรียกใช้ชุดคำสั่ง lpx_set_obj_dir เพื่อที่จะกำหนดว่าปัญหากำหนดการเชิงเส้นเป็นปัญหาที่ต้องการหาค่ามากที่สุด (maximization) หรือน้อยที่สุด (minimization)

บรรทัด s28 s29 และ s30 เรียกใช้ชุดคำสั่ง lpx_set_col_coef เพื่อกำหนดค่าสัมประสิทธิ์ของตัวแปรอิสระตัวแรก ตัวที่สอง และตัวที่สามตามลำดับ ซึ่งปรากฏในฟังก์ชันจุดประสงค์

ขณะนี้ในโครงสร้างของปัญหามีข้อมูลพร้อมที่จะดำเนินการแก้ปัญหากำหนดการเชิงเส้นแล้ว และในบรรทัด s31 เรียกใช้ชุดคำสั่ง lpx_simplex ซึ่งเป็นชุดคำสั่งในการแก้ปัญหากำหนดการเชิงเส้นดังกล่าว ด้วยวิธีซิมเพล็กซ์

บรรทัด s32 เรียกใช้ชุดคำสั่ง lpx_get_obj_val เพื่อนำค่าของฟังก์ชันจุดประสงค์ที่เหมาะที่สุดที่คำนวณได้จาก s31 ออกมาจากโครงสร้างข้อมูลของปัญหาโดยนำออกมาเก็บค่าไว้ในตัวแปร z

บรรทัด s33, s34 และ s35 เรียกใช้ชุดคำสั่ง lpx_get_col_info เพื่อที่จะนำค่าของตัวแปรอิสระตัวแรก ตัวที่สอง และตัวที่สาม ซึ่งเป็นผลเฉลยของปัญหาที่คำนวณได้จาก s31 ออกมาจากโครงสร้างของปัญหาโดยนำออกมาเก็บค่าไว้ในตัวแปร x_1, x_2 และ x_3 ตามลำดับ

บรรทัด s36 เป็นการนำค่าของผลเฉลยที่เหมาะที่สุดของปัญหากำหนดการเชิงเส้นมาแสดง ซึ่งจากตัวอย่างจะเป็นการนำค่าของ z, x_1, x_2 และ z_3 มาแสดง

บรรทัด s37 เรียกใช้ชุดคำสั่ง lpx_delete_prob ในการลบค่าต่างๆที่กำหนดให้กับโครงสร้างข้อมูลของปัญหา เพื่อให้ระบบพร้อมที่จะทำงานอีกครั้ง

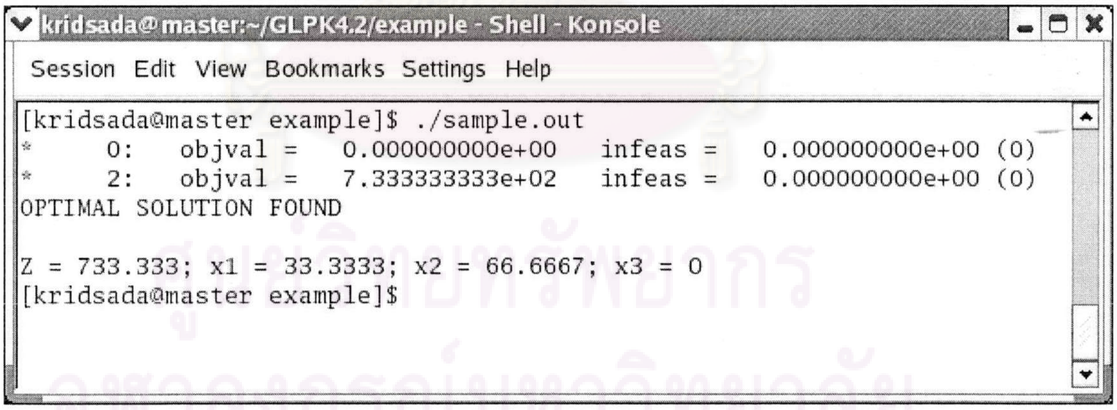
หลังจากได้ชุดคำสั่งหลักแล้วขั้นตอนต่อไปคือสั่งให้ตัวแปลภาษาซีทำการแปลชุดคำสั่งด้วยคำสั่งดังต่อไปนี้

```
$gcc -g -o2 -Iinclude sample.c  ../libglpk.a -lm -o sample.out
```

ผลลัพธ์ที่ได้คือไฟล์ sample.out ซึ่งเป็นไฟล์ที่สามารถเรียกใช้งานได้ และคำสั่งต่อไปนี้เป็นคำสั่งในการเรียกใช้งานไฟล์ sample.out

```
$./sample.out
```

และจะได้ผลลัพธ์คือผลเฉลยของปัญหากำหนดการเชิงเส้น ซึ่งแสดงดังรูปที่ ข-2



รูปที่ ข-2

จากรูปที่ ข-2 จะได้ว่า ปัญหากำหนดการเชิงเส้นมีผลเฉลยที่เหมาะที่สุดคือ $z = 733.333$ ซึ่งเป็นค่าที่ได้จาก $x_1 = 33.3333,\quad x_2 = 66.6667$ และ $x_3 = 0$

ประวัติผู้เขียนวิทยานิพนธ์

ชื่อ-นามสกุล: นายกฤษดา นารอง

วัน-เดือน-ปีเกิด: 16 มิถุนายน พ.ศ. 2518

ภูมิลำเนา: ต.กู่ทอง อ.เชียงยืน จ.มหาสารคาม

สำเร็จการศึกษาวิทยาศาสตรบัณฑิต(คณิตศาสตร์) จากมหาวิทยาลัยขอนแก่น ปี 2540

สถานที่ทำงาน: ภาควิชาคณิตศาสตร์ สถิติและคอมพิวเตอร์ คณะวิทยาศาสตร์
มหาวิทยาลัยอุบลราชธานี

ทุนการศึกษา: ทุนพัฒนาอาจารย์ ปี 2544