SYSEN 6000: Foundations of Complex Systems

Deterministic Global Optimization

Nick Kunz [NetID: nhk37] nhk37@cornell.edu

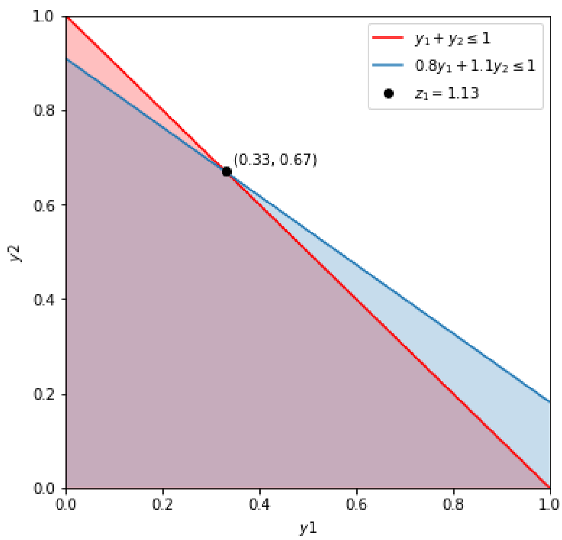
October 31, 2022

Integer Programming

Given that:

$$
\begin{aligned}
\max \quad & y_1 + 1.2y_2 \\
\text{s.t.} \quad & y_1 + y_2 \leq 1 \\
& 0.8y_1 + 1.1y_2 \leq 1 \\
& y_1, y_2 = \{0, 1\}
\end{aligned} \tag{1}
$$

When the binary variables are relaxed to continuous ones, such that $y_1, y_2 \in [0, 1]$, the contours of the objective and the feasible region can be exhibited as:



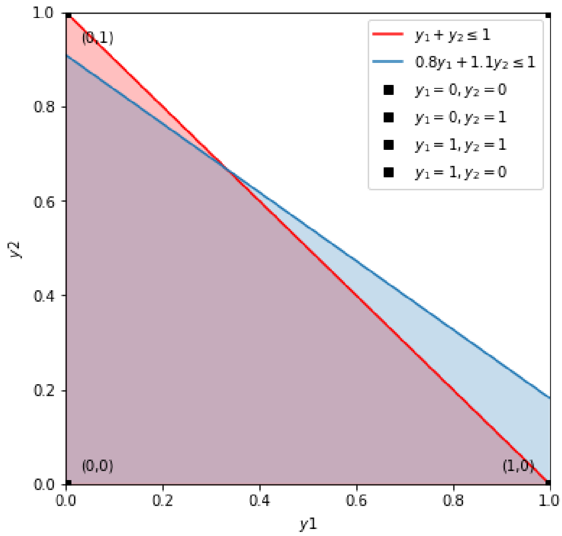
From inspection, the optimal solution to the relaxed problem, such that $y_1, y_2 \in [0, 1]$, is:

$$
\begin{gathered}
y_1 = \frac{1}{3}, y_2 = \frac{2}{3} \\
z_1 = 1.13
\end{gathered} \tag{2}
$$

Enumerating all y_1, y_2 integer combinations $[0,1]$ the solution space can be expressed as:

$$
\begin{aligned}
y_1 &= 0, y_2 = 0 \\
y_1 &= 0, y_2 = 1 \\
y_1 &= 1, y_2 = 1 \\
y_1 &= 1, y_2 = 0
\end{aligned} \tag{3}
$$

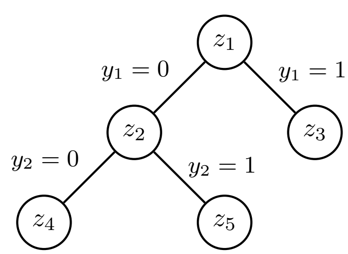
which can be exhibited as:



Therefore, the optimal solution is:

$$y_1 = 1, y_2 = 0 \tag{4}$$

The branch and bound method for solving LP subproblems can be exhibited as:



where:

$$
\begin{aligned}
z_1 &= 1.13 \\
z_2 &= 1.091 \\
z_3 &= 1 \\
z_4 &= 0 \\
z_5 &= \cdot
\end{aligned} \tag{5}
$$

Therefore, the optimal solution is:

$$\begin{gathered} y_1 = 1, y_2 = 0 \\ z_3 = 1 \end{gathered} \tag{6}$$

Gomory Cut

Given the inequality:

$$y_2 + y_3 + 2y_4 \leq 6 \tag{7}$$

and the region:

$$X = \{y \in Z_4 : 4y_1 + 5y_2 + 9y_3 + 12y_4 \leq 34\} \tag{8}$$

It is provably valid by:

$$\frac{4}{5}y_1 + \frac{5}{5}y_2 + \frac{9}{5}y_3 + \frac{12}{5}y_4 \leq \frac{34}{5} \Longrightarrow 0\frac{4}{5}y_1 + y_2 + 1\frac{4}{5}y_3 + 2\frac{2}{5}y_4 \leq \frac{34}{5} \tag{9}$$

when considering:

$$\sum_j a_{ij}^* y_j \leq b_i, y_j \in \{0, 1\} \tag{10}$$

which is given by:

$$\sum_j \lfloor a_{ij} \rfloor y_j \leq \lfloor b_i \rfloor \tag{11}$$

and when applied is:

$$0 \cdot y_1 + y_2 + 1 \cdot y_3 + 2 \cdot y_4 \leq 6 \tag{12}$$

which results in the given inequality:

$$y_2 + y_3 + 2y_4 \leq 6 \tag{13}$$

Mixed-Integer Non-Linear Programming

Given that:

$$\begin{aligned} \min \quad & f = y_1 + 1.5y_2 + 0.5y_3 + x_1^2 + x_2^2 \\ \text{s.t.} \quad & (x_1 - 2)^2 - x_2 \leq 0 \\ & x_1 - 2y_1 \geq 0 \\ & x_1 - x_2 - 4(1 - y_1) \leq 0 \\ & x_1 - (1 - y_1) \geq 0 \\ & x_2 - y_2 \geq 0 \\ & x_1 + x_2 \geq 3y_3 \\ & y_1 + y_2 + y_3 \geq 1 \\ & 0 \leq x_1 \leq 4, 0 \leq x_2 \leq 4 \\ & y_1, y_2, y_3 = \{0, 1\} \end{aligned} \tag{14}$$

the inequalities can be reformulated as:

$$\begin{aligned} \text{s.t.} \quad & (x_1-2)^2 - x_2 \le 0 \\ & -(x_1 - 2y_1) \le 0 \\ & x_1 - x_2 - 4(1-y_1) \le 0 \\ & -x_1 - (1-y_1) \le 0 \\ & -(x_2 + y_2) \le 0 \\ & -(x_2 + y_2 - 3y_3) \le 0 \\ & -(y_1 + y_2 + y_3 - 1) \le 0 \end{aligned} \tag{15}$$

Let $y_1 = y_2 = y_3 = 1$, so that:

$$\begin{aligned} \min \quad & f = 3 + x_1^2 + x_2^2 \\ \text{s.t.} \quad & (x_1-2)^2 - x_2 \le 0 \\ & -(x_1 - 2\cdot 1) \le 0 \\ & x_1 - x_2 - 4(1-1) \le 0 \\ & -x_1 - (1-1) \le 0 \\ & -(x_2 + 1) \le 0 \\ & -(x_2 + 1 - 3\cdot 1) \le 0 \\ & -(1+1+1-1) \le 0 \\ & 0 \le x_1 \le 4, 0 \le x_2 \le 4 \\ & y_1, y_2, y_3 = \{0, 1\} \end{aligned} \tag{16}$$

which results in $x_1^* = 2, x_2^* = 2, UB_1 = 5$, where we can continue to solve for:

$$\nabla f = \begin{bmatrix} 2\cdot 2 \\ 2\cdot 2 \end{bmatrix} \implies \nabla f^* = \begin{bmatrix} 4 \\ 4 \end{bmatrix} \tag{17}$$

$$\begin{aligned} f_{Lin} &= f(x^*) + \nabla f(x^*) + (x - x^*) \\ &= 8 + \begin{bmatrix} 4 \\ 4 \end{bmatrix} \begin{bmatrix} 2-2 \\ 2-2 \end{bmatrix} \\ &= 8 + 4(2-2) + 4(2-2) + \\ & \quad (y_1 + 1.5y_2 + 0.5y_3) \end{aligned} \tag{18}$$

$$\nabla g = \begin{bmatrix} 2(2-2) \\ -1 \end{bmatrix} \implies \nabla g^* = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \tag{19}$$

$$\begin{aligned} g_{Lin} &= g(x^*) + \nabla g(x^*) + (x - x^*) \\ &= -2 + \begin{bmatrix} 0 \\ -1 \end{bmatrix} \begin{bmatrix} 2-2 \\ 2-2 \end{bmatrix} \\ &= -2 - (2-2) \le 0 \end{aligned} \tag{20}$$

Hence $x_1^* = 1, x_2^* = 1, y_1 = 0, y_2 = 1, y_3 = 0, LB_1 = 3.5$, when min α, s.t. $\alpha \ge f_L in, g_L in \le 0$, and the reformulated constraints, where the $UB_1 > LB_1$.

Now let $y_1 = 0, y_2 = 1, y_3 = 0$, so that:

$$\begin{aligned}
\min \quad & f = 1.5 + x_1^2 + x_2^2 \\
\text{s.t.} \quad & (x_1 - 2)^2 - x_2 \leq 0 \\
& -x_1 \leq 0 \\
& x_1 - x_2 - 4 \leq 0 \\
& -x_1 - 1 \leq 0 \\
& -x_2 + 1 \leq 0 \\
& -x_1 - x_2 \leq 0 \\
& -x_2 \leq 0 \\
& x_1 - 4 \leq 0 \\
& x_2 - 4 \leq 0 \\
& 0 \leq x_1 \leq 4, 0 \leq x_2 \leq 4 \\
& y_1, y_2, y_3 = \{0, 1\}
\end{aligned} \tag{21}$$

Hence $x_1^* = 1, x_2^* = 1, y_1 = 0, y_2 = 1, y_3 = 0, UB_2 = 3.5$, when min α, s.t. $\alpha \geq f_L in, g_L in \leq 0$, and the reformulated constraints, where $UB_2 = LB_1$.

Here the stopping criteria $UB = LB$ has been satisfied, where the optimal solution is:

$$\begin{gathered}
x_1 = 1, x_2 = 1 \\
y_1 = 0, y_2 = 1, y_3 = 0 \\
z_3 = 3.5
\end{gathered} \tag{22}$$

Mixed-Integer Linear Fractional Programming

Given that:

$$\begin{aligned}
\max \quad & \frac{2x + 4y_1 + 3y_2}{x + y_1 + 1} \\
\text{s.t.} \quad & 4x - y_1 - 4y_2 \leq 1 \\
& 5x + 4y_1 + 2y_2 \leq 3 \\
& x \geq 0 \\
& y_1, y_2 \in \{0, 1\}
\end{aligned} \tag{23}$$

The parametric algorithm applies q, which reformulates into:

$$\begin{aligned}
\max \; & (2x + 4y_1 + 3y_2) - q \cdot (x + y_1 + 1) \\
\text{s.t.} \; & x + y_1 + 1 > 0 \\
& 4x - y_1 - 4y_2 \leq 1 \\
& 5x + 4y_1 + 2y_2 \leq 3 \\
& x \geq 0 \\
& y_1, y_2 \in \{0, 1\}
\end{aligned} \tag{24}$$

that when solved, the optimal solution is:

$$\begin{aligned} x &= 0 \\ y_1 &= 0, y_2 = 1 \end{aligned} \tag{25}$$

The reformulation-linearization algorithm applies g and z, which reformulates into:

$$\begin{aligned} \max\ & 2 \cdot z + 4 \cdot g \cdot y_1 + 3 \cdot g \cdot y_2 \\ \text{s.t.}\ \ & z + g \cdot y_1 + 1 \cdot g = 1 \\ & 4 \cdot z - g \cdot y_1 - 4 \cdot g \cdot y_2 \leq 1 \\ & 5 \cdot z + 4 \cdot g \cdot y_1 + 2 \cdot g \cdot y_2 \leq 3 \\ & z \geq 0, g \geq 0 \\ & y_1, y_2 \in \{0, 1\} \end{aligned} \tag{26}$$

and with w_1 and w_2, further into:

$$\begin{aligned} \max\ & 2 \cdot z + 4 \cdot w_1 + 3 \cdot w_2 \\ \text{s.t.}\ \ & z + w_1 + 1 \cdot g = 1 \\ & 4 \cdot z - w_1 - 4 \cdot w_2 - g \leq 0 \\ & 5 \cdot z + 4 \cdot w_1 + 2 \cdot w_2 - 3 \cdot g \leq 0 \\ & 0 \leq w_1 \leq M \cdot y_1 \\ & 0 \leq w_2 \leq M \cdot y_2 \\ & w_1 \leq g \\ & w_2 \leq g \\ & w_1 \geq g - M(1 - y_1) \\ & w_2 \geq g - M(1 - y_2) \\ & z \geq 0 \\ & g \geq 0 \\ & w_1 \geq 0 \\ & w_2 \geq 0 \\ & y_1, y_2 \in \{0, 1\} \end{aligned} \tag{27}$$

where:

$$\begin{aligned} z &= g \cdot x \\ g &= \frac{1}{x + y_1 + 1} \\ w_1 &= g \cdot y_1, w_2 = g \cdot y_2 \end{aligned} \tag{28}$$

that when solved, the optimal solution is:

$$\begin{aligned} z &= 0 \\ g &= 1 \\ w_1 &= 0, w_2 = 1 \\ y_1 &= 0, y_2 = 1 \end{aligned} \tag{29}$$

Appendix

1. Integer Programming

```python
## librarires
import numpy as np
from scipy.optimize import linprog
import matplotlib.pyplot as plt

## minimize objective
## (inversed original maximize)
obj = [-1.0, -1.2]

## constraints
## left side inequalities
lft_ieq = [
    [1.0, 1.0],
    [0.8, 1.1]
]

## right side inequalities
rgt_ieq = [
    1.0,
    1.0
]

## global bounds
gbl_bds = [
    [0.0, 1.0], ## y1
    [0.0, 1.0] ## y2
]

## optimize
opt = linprog(
    c = obj,
    A_ub = lft_ieq,
    b_ub = rgt_ieq,
    bounds = gbl_bds,
    method = 'simplex'
)

## solution
y1 = round(
    number = opt.x[0],
    ndigits = 2
)

y2 = round(
    number = opt.x[1],
    ndigits = 2
)

z = round(
    number = -opt.fun,
    ndigits = 2
)

print('y1 = {x}'.format(x = y1))
print('y2 = {x}'.format(x = y2))
print('z = {x}'.format(x = z))
```

Python 3: Integer Programming Solution to the Relaxed Problem

2. Continuous Contours

```python
## librarires
import numpy as np
import matplotlib.pyplot as plt

## space
x = np.linspace(0, 1, 100)

## constraints
y1 = (1 - x)
y2 = (0.8 / -1.1) * x + (-1.0 / -1.1)

## plot
plt.figure(figsize=(6,6))
plt.plot(x, y1,
    color = 'red',
    label = r'$y_1 + y_2 \leq 1$'
)
plt.plot(
    x, y2,
    label = r'$0.8y_1 + 1.1y_2 \leq 1$'
)

plt.plot(
    0.33, 0.67, 'o',
    color = 'black',
    label = r'$z_1 = 1.13$'
)

plt.annotate(
    s = '(0.33, 0.67)',
    xy = (0.33, 0.67),
    textcoords = "offset points",
    xytext = (5, 5),
    ha = 'left'
)

plt.fill_between(
    x, y1,
    color = 'red',
    alpha = 0.25
)

plt.fill_between(
    x, y2,
    alpha = 0.25
)

plt.xlabel(r'$y1$')
plt.ylabel(r'$y2$')
plt.xlim(0, 1)
plt.ylim(0, 1)

plt.legend(
    bbox_to_anchor = (1, 1),
    loc = 0,
    borderaxespad = 0.5
)

plt.show()
```

Python 3: Contours of the Relaxed Problem

3. Enumerated Combinations

```
## librarires
import numpy as np
import matplotlib.pyplot as plt

## space
x = np.linspace(0, 1, 100)

## constraints
y1 = (1 - x)
y2 = (0.8 / -1.1) * x + (-1.0 / -1.1)

## plot
plt.figure(figsize=(6,6))
plt.plot(
    x, y1,
    color = 'red',
    label = r'$y_1 + y_2 \leq 1$'
)

plt.plot(
    x, y2,
    label = r'$0.8y_1 + 1.1y_2 \leq 1$'
)

plt.plot(
    0, 0, 's',
    color = 'black',
    label = r'$y_1 = 0, y_2 = 0$'
)

plt.plot(
    0, 1, 's',
    color = 'black',
    label = r'$y_1 = 0, y_2 = 1$'
)

plt.plot(
    1, 1, 's',
    color = 'black',
    label = r'$y_1 = 1, y_2 = 1$'
)

plt.plot(
    1, 0, 's',
    color = 'black',
    label = r'$y_1 = 1, y_2 = 0$'
)

plt.annotate(
    s = '(0,0)',
    xy = (0, 0),
    textcoords = "offset points",
    xytext = (10, 10),
    ha = 'left'
)

plt.annotate(
    s = '(0,1)',
    xy = (0, 1),
    textcoords = "offset points",
    xytext = (10, -20),
    ha = 'left'
)
```

```python

plt.annotate(
    s = '(1,0)',
    xy = (1, 0),
    textcoords = "offset points",
    xytext = (-10, 10),
    ha = 'right'
)

plt.fill_between(
    x, y1,
    color = 'red',
    alpha = 0.25
)

plt.fill_between(
    x, y2,
    alpha = 0.25
)

plt.xlabel(r'$y1$')
plt.ylabel(r'$y2$')
plt.xlim(0, 1)
plt.ylim(0, 1)

plt.legend(
    bbox_to_anchor = (1, 1),
    loc = 0,
    borderaxespad = 0.5
)

plt.show()

## happy halloween!
```

Python 3: Enumerated Combinations for the Optimal Solution