E4004 2021 Final's solution

Professor Cédric Josz

Problem 1: (14 points) Given $c \in \mathbb{R}^n$, show that the following mathematical program

$$\min_{x \in \mathbb{R}^n} \quad c^T x \quad \text{subject to} \quad 1 \leqslant \|x\|_\infty \leqslant 2 \tag{1}$$

can be cast as a mixed-integer linear program, where $\|x\|_\infty := \max\{|x_1|, \ldots, |x_m|\}$.

Solution: Fist we may write $\min\{c^T x \mid 1 - y_i \leqslant |x_i| \leqslant 2,\ y_i \in \{0,1\},\ y_1 + \ldots + y_n \geqslant 1\}$. Since the feasible set is bounded, we can then use the big M method: $\min\{c^T x \mid 1 - y_i \leqslant x_i + M z_i,\ 1 - y_i \leqslant -x_i + M(1 - z_i),\ z_i \in \{0,1\},\ x_i \leqslant 2,\ y_i \in \{0,1\},\ y_1 + \ldots + y_n \geqslant 1\}$.

Problem 2: (12 points) We would like to install temporary healthcare facilities in counties in New Mexico so that each county is within at most 10km of a facility. The distances between counties can be found in Table 1. Suppose we would like to build as few facilities as possible. Formulate this problem as an integer program and solve it using python. Where should one install the facilities?

Solution: A solution is to install a facility at Catron, Luna and Lea.

County	Catron	Eddy	Luna	Mora	Taos	Lea
Catron	0	10	40	25	20	10
Eddy	10	0	25	20	35	20
Luna	40	25	0	10	25	5
Mora	25	20	10	0	35	15
Taos	20	35	25	35	0	10
Lea	10	20	5	15	10	0

Table 1: Distance between counties in km

See Figure 1 and 2.

Problem 3: (14 points) Imagine that you are helping a friend to move into their new place. Your friend only has one box of 28kg to move their belongings, so you propose to lend your box of 22kg to help your friend. The weight and the value of your friend's belongings are given in Table 2. Provide an integer programming formulation that maximizes the value of the items packed. Without solving the integer program, is it the case that one of the items will definitely be packed no matter what? Justify your answer.

Solution:

```python
import gurobipy as gp
from gurobipy import GRB

# Create a new model
m = gp.Model("cover")

# Create variables
x1 = m.addVar(vtype=GRB.INTEGER,name="x1")
x2 = m.addVar(vtype=GRB.INTEGER,name="x2")
x3 = m.addVar(vtype=GRB.INTEGER,name="x3")
x4 = m.addVar(vtype=GRB.INTEGER,name="x4")
x5 = m.addVar(vtype=GRB.INTEGER,name="x5")
x6 = m.addVar(vtype=GRB.INTEGER,name="x6")


# Set objective
m.setObjective(x1+x2+x3+x4+x5+x6, GRB.MINIMIZE)

# Add constraint
m.addConstr(x1 + x2                + x6 >= 1)
m.addConstr(x1 + x2                     >= 1)
m.addConstr(          x3 + x4 +      x6 >= 1)
m.addConstr(          x3 + x4           >= 1)
m.addConstr(                    x5 + x6 >= 1)
m.addConstr(x1      + x3 +      x5 + x6 >= 1)

# Optimize model
m.optimize()

for v in m.getVars():
    print('%s %g' % (v.varName, v.x))

print('Obj: %g' % m.objVal)
```

Figure 1: Python code

Item	#1	#2	#3	#4	#5	#6	#7	#8
Weight (kg)	10	9	15	3	3	6	11	4
Value ($)	5	2	7	6	8	6	1	6

Table 2: Belongings

$$\begin{array}{rc} \max & 5(x_{11}+x_{21})+2(x_{12}+x_{22})+7(x_{13}+x_{23})+6(x_{14}+x_{24})+ \\ & 8(x_{15}+x_{25})+6(x_{16}+x_{26})+x_{17}+x_{27}+6(x_{18}+x_{28}) \\ \text{s.t.} & 10x_{i1}+9x_{i2}+15x_{i3}+3x_{i4}+3x_{i5}+6x_{i6}+11x_{i7}+4x_{i8} \leqslant 22,\ i=1, \\ & 10x_{i1}+9x_{i2}+15x_{i3}+3x_{i4}+3x_{i5}+6x_{i6}+11x_{i7}+4x_{i8} \leqslant 28,\ i=2, \\ & x_{i1}+x_{i2} \leqslant 1,\ i=1,2, \\ & x_{i1},\ldots,x_{i8} \in \{0,1\},\ i=1,2, \end{array} \tag{2}$$

Item 5 will definitely be packed since in any feasible solution one can remove any item and replace by item 5 and increase the objective value.

Problem 4: (14 points) A gala is being organized for the students of a masters program. To promote exchanges, it is desired that no more than three students from one country sit at the same table, and no more than six from the same continent. Use a maximum flow formulation to determine whether one can find a seating arrangement. There are eight countries with respectively $s_1, \ldots, s_8$ students. The first three countries are in Asia, the next three countries are in Europe, and the remaining two are in North America. There are n tables with respectively $t_1, \ldots, t_n$ seats.

Solution: See Figure 3.

Problem 5: (14 points) Given a list of relative integers such as in Table 3, we would like to the find the greatest consecutive sum of numbers. For example, the second, third and fourth numbers in Table 3 add up to eleven, while the last four numbers add up to ten. Propose a dynamic programming

```
Gurobi Optimizer version 9.1.2 build v9.1.2rc0 (mac64)
Thread count: 4 physical cores, 8 logical processors, using up to 8 threa
ds
Optimize a model with 6 rows, 6 columns and 16 nonzeros
Model fingerprint: 0x87e8507f
Variable types: 0 continuous, 6 integer (0 binary)
Coefficient statistics:
  Matrix range     [1e+00, 1e+00]
  Objective range  [1e+00, 1e+00]
  Bounds range     [0e+00, 0e+00]
  RHS range        [1e+00, 1e+00]
Found heuristic solution: objective 3.0000000
Presolve removed 6 rows and 6 columns
Presolve time: 0.00s
Presolve: All rows and columns removed

Explored 0 nodes (0 simplex iterations) in 0.01 seconds
Thread count was 1 (of 8 available processors)

Solution count 1: 3

Optimal solution found (tolerance 1.00e-04)
Best objective 3.000000000000e+00, best bound 3.000000000000e+00, gap 0.0
000%
x1 1
x2 -0
x3 1
x4 -0
x5 -0
x6 1
Obj: 3
```

Figure 2: Gurobi output

approach to solve this problem. How does the computational burden compare with simply computing every possible sum?

Solution: Let $v_n := \max\{\sum_{k=i}^{j} a_i \mid 1 \leqslant i \leqslant j \leqslant n\}$ where $a_1, \ldots, a_n$ are

10	-15	6	20	-13	9	-16	10	7

Table 3: List of relative numbers

the relative integers. We have $v_1 = a_1$ and $v_l = \max\{v_{l-1}, a_1 + \ldots + a_l, a_2 + \ldots + a_l, \ldots, a_l\}$ which basically requires l summations at every step plus finding the maximum of l terms. The dynamic program thus requires $n(n+1)/2$ summations and finding the maximum of that many terms (roughly). The brute force approach would require 2^n summations instead and finding a maximum among that many terms.

Problem 6: (30 points) Consider the following optimization problem:

$$\inf_{x_1, x_2 \in \mathbb{R}} \quad f(x_1, x_2) \tag{3}$$

where $f : \mathbb{R}^2 \longrightarrow \mathbb{R}$.

1. Assume that $f(x_1, x_2) := x_1^2 + (1 - x_1 x_2)^2$. Write the first and second order necessary optimality conditions at a point (x_1, x_2). What can you deduce about the set of minimizers? Is the infimum reached? What is the infimum equal to?

2. Assume that $f(x_1, x_2) := x_1^4 + x_2^4 + x_1 x_2 - x_1^2 + x_2$. Write the first and second order necessary optimality conditions at a point (x_1, x_2). Prove that $|x_1| \geqslant \sqrt{6}/6$.

3. Assume that $f(x_1, x_2) := 2x_1^3 + x_1^2 + 1/4 x_1 x_2 + x_2^2 - x_2/2 + 1/16$ and that we would like to find the infimum on the ball $x_1^2 + x_2^2 \leqslant 1$. Write

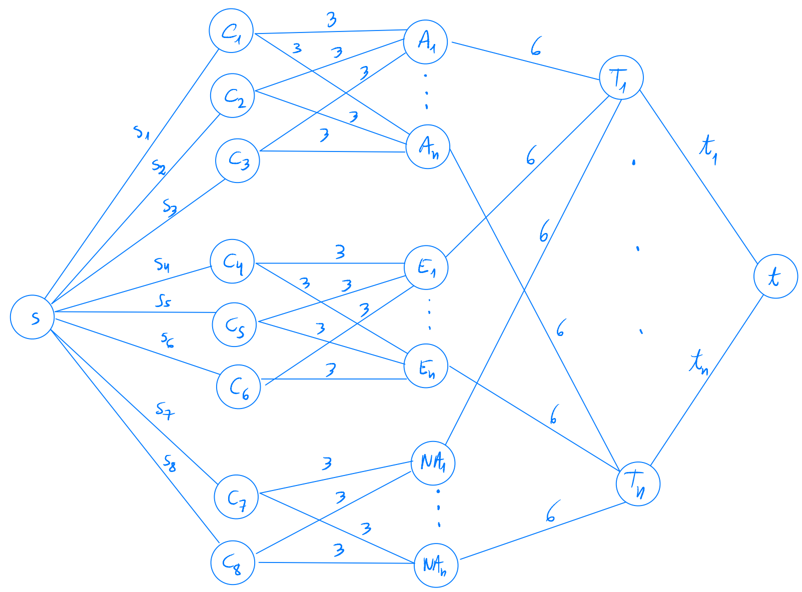

Figure 3: Gala: maximum flow formulation

the first order necessary optimality conditions (i.e., KKT) at a point (x_1, x_2).

Solution:

1. We have
$$\nabla f(x_1,x_2) = \begin{pmatrix} 2x_1 - 2x_2(1-x_1x_2) \\ -2x_1(1-x_1x_2) \end{pmatrix} \tag{4}$$
$$\nabla^2 f(x_1,x_2) = \begin{pmatrix} 2(1+x_2^2) & 2(2x_1x_2-1) \\ 2(2x_1x_2-1) & 2x_1^2 \end{pmatrix} \tag{5}$$
$\nabla f(x_1, x_2) = 0$ iff $x_1 = x_2 = 0$ and
$$\nabla^2 f(0,0) = \begin{pmatrix} 2 & -2 \\ -2 & 0 \end{pmatrix} \tag{6}$$
which is not positive semi-definite. Thus no local minimum exists. The infimum is thus not reached. The infimum is equal to zero (one can take the minimizing sequence defined by $(x_1^n, x_2^n) = (1/(1+n), n+1)$.

2. We have
$$\nabla f(x_1,x_2) = \begin{pmatrix} 4x_1^3 + x_2 - 2x_1 \\ 4x_2^3 + x_1 + 1 \end{pmatrix} \tag{7}$$
$$\nabla^2 f(x_1,x_2) = \begin{pmatrix} 12x_1^2 - 2 & 1 \\ 1 & 12x_2^2 \end{pmatrix} \tag{8}$$
Since the trace and the determinant of the Hessian should be non-negative, we have $x_1^2 + x_2^2 \geqslant 1/6$ and $(x_1^2 - 1/6)x_2^2 \geqslant 0$. We can distinguish whether $x_2 = 0$; in both cases $x_1^2 \geqslant 1/6$.

3. We have $L(x_1, x_2, \lambda) = 2x_1^3 + x_1^2 + 1/4x_1x_2 + x_2^2 - x_2/2 + 1/16 + \lambda(x_1^2 + x_2^2 - 1)$ and

$$\begin{cases} 6x_1^2 + 2x_1 + x_2/4 + 2\lambda x_1 = 0 \\ x_1/4 + 2x_2 - 1/2 + 2\lambda x_2 = 0 \\ x_1^2 + x_2^2 - 1 \leqslant 0 \\ \lambda \geqslant 0 \\ (1 - x_1^2 - x_2^2)\lambda = 0. \end{cases} \tag{9}$$