MAT3007 Homework 1 Student ID: 121090429

Problem 1 (25pts). A company produces two kinds of products. A product of the first type requires 1/8 hours of assembly labor, 1/4 hours of testing, and \$1.2 worth of raw materials. A product of the second type requires 1/2 hours of assembly, 1/6 hours of testing, and \$0.9 worth of raw materials. Given the current personnel of the company, there can be at most 90 hours of assembly labor and 80 hours of testing each day. Products of the first and second type have a market value of \$9 and \$8 respectively.

(a) Formulate a linear optimization that maximizes the daily profit of the company. (For simplicity, you do not need consider the completeness of products. That is, the variables may take float values.)

(b) Write the standard form of the LP you formulated in part (a).

(c) Consider the following modification to the original problem: Suppose that up to 40 hours of overtime assembly labor can be scheduled, at a cost of \$8 per hour. Can it be easily incorporated into the linear optimization formulation and how?

(d) Solve the LP using MATLAB (for the original problem).

(a) The number of two types of products are X_1, X_2

$$\begin{aligned} \underset{X_1, X_2}{\text{maximize}} \quad & 7.8X_1 + 7.1X_2 \\ \text{subject to} \quad & \tfrac{1}{8}X_1 + \tfrac{1}{2}X_2 \leq 90 \\ & \tfrac{1}{4}X_1 + \tfrac{1}{6}X_2 \leq 80 \\ & X_1, X_2 \geq 0 \end{aligned}$$

(b)

$$\begin{aligned} \underset{X_1, X_2, S_1, S_2}{\text{minimize}} \quad & -7.8X_1 - 7.1X_2 \\ \text{subject to} \quad & \tfrac{1}{8}X_1 + \tfrac{1}{2}X_2 + S_1 = 90 \\ & \tfrac{1}{4}X_1 + \tfrac{1}{6}X_2 + S_2 = 80 \\ & X_1, X_2, S_1, S_2 \geq 0 \end{aligned}$$

(c) New profit is $7.8X_1 + 7.1X_2 - 8\left(\frac{1}{8}X_1 + \frac{1}{2}X_2 - 90\right)^+$

with constraints :
$$\begin{cases} \frac{1}{8}X_1 + \frac{1}{2}X_2 \leq 130 \\ \frac{1}{4}X_1 + \frac{1}{6}X_2 \leq 80 \\ X_1, X_2 \geq 0 \end{cases}$$

Define $m = \left(\frac{1}{8}X_1 + \frac{1}{2}X_2 - 90\right)^+$

The problem can be intercorporated into:

$$\begin{aligned} \underset{X_1, X_2, m}{\text{maximize}} \quad & 7.8X_1 + 7.1X_2 - 8m \\ \text{subject to} \quad & m \geq \left(\tfrac{1}{8}X_1 + \tfrac{1}{2}X_2 - 90\right)^+ \\ & \tfrac{1}{8}X_1 + \tfrac{1}{2}X_2 \leq 130 \\ & \tfrac{1}{4}X_1 + \tfrac{1}{6}X_2 \leq 80 \\ & X_1, X_2 \geq 0 \end{aligned}$$

(d) code:

编辑器 - D:\大二上\MAT3007\assignment\homework1\Homework1_Problem1_121090429.m

untitled.m | Homework1_Problem3_121090429.m | Homework1_Problem2_121090429.m

```
clear;
clc;

cvx_begin
variables x1 x2
minimize -7.8*x1 - 7.1*x2
subject to
1/8*x1 + 1/2*x2 <= 90;
1/4*x1 + 1/6*x2 <= 80;
x1 >= 0;
x2 >= 0;
cvx_end

% print outcome
x1
x2
7.8*x1 + 7.1*x2


```

outcome:

```
------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): -2724


x1 =

  240.0000


x2 =

  120.0000


ans =

   2.7240e+03

fx >>
```

The maximized daily profit is \$2724 with $X_1 = 240$, $X_2 = 120$.

Problem 2 (25pts). The China Railroad Ministry is in the process of planning relocations of freight cars among 5 regions of the country to get ready for the fall harvest. Table1a shows the cost of moving a car between each pair of regions. Table1b shows the current number of cars in each region and the number needed for harvest shipping.

From/To	1	2	3	4	5
1	–	20	13	11	28
2	20	–	18	8	46
3	13	18	–	9	27
4	11	8	9	–	20
5	28	46	27	20	–

(a) Cost of moving a car

	1	2	3	4	5
Present	110	335	400	420	610
Need	150	200	600	200	390

(b) Number of current and needed cars

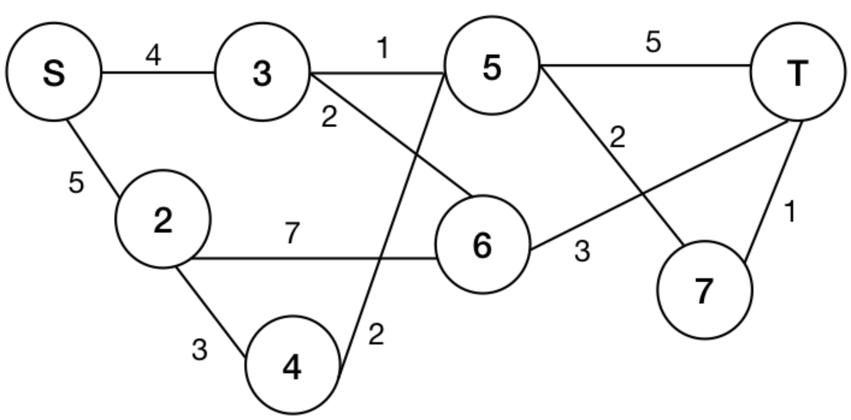

Figure 1: The graph of the shortest path problem

Write down a linear optimization to compute the least costly way to move the cars such us the need is met. Solve the problem using MATLAB.

Create a 5×5 matrix that stands for cars that send out and sent back. Formulate it as a standard form

code:

编辑器 - D:\大二上\MAT3007\assignment\homework1\Homework1_Problem2_121090429.m

untitled.m × | Homework1_Problem3_121090429.m × | Homework1_Problem2_121090429.m × | Homework1

```
clear;
clc;

price = [0 20 13 11 28;
20 0 18 8 46;
13 18 0 9 27;
11 8 9 0 20;
28 46 27 20 0];

cvx_begin
variables x(5,5);
minimize sum(sum(price.*x));
subject to % the sum of cars that sent back and out must confirm the constraints
    x(2,1) + x(3,1) + x(4,1) + x(5,1) - x(1,2) - x(1,3) - x(1,4) - x(1,5) >= 40 ;
    x(1,2) + x(3,2) + x(4,2) + x(5,2) - x(2,1) - x(2,3) - x(2,4) - x(2,5) >= -135 ;
    x(1,3) + x(2,3) + x(4,3) + x(5,3) - x(3,1) - x(3,2) - x(3,4) - x(3,5) >= 200 ;
    x(1,4) + x(2,4) + x(3,4) + x(5,4) - x(4,1) - x(4,2) - x(4,3) - x(4,5) >= -220 ;
    x(1,5) + x(2,5) + x(3,5) + x(4,5) - x(5,1) - x(5,2) - x(5,3) - x(5,4) >= -220 ;
    x >= zeros(5,5);
cvx_end

sum(sum(price.*x))
```

outcome:

```
------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): +2400


ans =

   2.4000e+03

fx >>
```

Problem 3 (20pts). Write a MATLAB code to use linear optimization to solve the shortest path problem. Suppose the input of the problem will be an $n \times n$ matrix of W, where w_{ij} is the length of the path from i to j (in general, w_{ij} does not necessarily equal w_{ji}). In our implementation, we always use 1 to denote the source node (the s node in the lecture slides), and n to denote the terminal node (the t node in the lecture slides). In addition, we assume for any i and j, there is a path, i.e., the set of E is all pairs of nodes. This is without loss of generality since one can set w_{ij} to be an extremely large number if there is no edge between i and j, effectively eliminating it from consideration.

After writing the code, you are asked to solve the concrete problem in the lecture slides, with the given labeling shown in Figure 1. Basically, you need to input the W matrix for this case, then solve it, and then report your solution (the optimal path).

Let W denotes a $n \times n$ matrix that contains the distance between nodes.

minimize the total distance $\sum_{i=1}^{n}\sum_{j=1}^{n} W_{ij}X_{ij}$

subject to: $\sum_{j=1}^{n} X_{1j} - \sum_{j=1}^{n} X_{j1} = 1,$

$\sum_{j=1}^{n} X_{jn} - \sum_{j=1}^{n} X_{nj} = 1,$

$\sum_{j=1}^{n} X_{ij} - \sum_{j=1}^{n} X_{ji} = 0, \ \forall i \neq 1, n$

$X_{ij} \in \{0, 1\}$

code:

编辑器 - D:\大二上\MAT3007\assignment\homework1\Homework1_Problem3_121090429.m

untitled.m | Homework1_Problem3_121090429.m | Homework1_Problem2_121090429.m | Homewo

```matlab
infi = 1000;
% set the distance between the point and itself to be a large enough number
w = [infi,5,4,infi,infi,infi,infi,infi;
    5,infi,infi,3,infi,7,infi,infi;
    4,infi,infi,infi,1,2,infi,infi;
    infi,3,infi,infi,2,infi,infi,infi;
    infi,infi,1,2,infi,infi,2,5;
    infi,7,2,infi,infi,infi,infi,3;
    infi,infi,infi,infi,2,infi,infi,1;
    infi,infi,infi,infi,5,3,1,infi]
[n,m]=size(w)

cvx_begin
    variable x(n,n);
    minimize (sum(sum(w.*x)));
    subject to
        sum(x(1,:)) - sum(x(:,1)) == 1;
        sum(x(:,n)) - sum(x(n,:)) == 1;
        for i = 2:n-1
            sum(x(i,:)) - sum(x(:,i)) == 0;
        end
        max(max(x)) <= 1
        min(min(x)) >= 0
cvx_end
x
```

outcome:

命令行窗口

```
 DIMACS: 2.5e-10  0.0e+00  7.4e-06  0.0e+00  -1.1e-03  2.3e-06
------------------------------------------------------------------

------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): +7.98069


x =

   -0.0000   -0.0000    1.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -0.0000   -0.0000   -0.0000    1.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000    1.0000   -0.0000
   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000    1.0000
   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000   -0.0000

fx >>
```

Problem 4 (30pts). Reformulate the following problems as linear programming problems

(a)

$$\begin{array}{ll} \min & 2x + 3|y - x| \\ \text{s.t.} & |x + 2| + |y| \le 5 \end{array}$$

where $x, y \in \mathbb{R}$.

(b)

$$\begin{array}{ll} \min & c^\top x + f(d^\top x) \\ \text{s.t.} & Ax \ge b \end{array}$$

where $x, c, d \in \mathbb{R}^n, A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$, $f(\alpha) = \max\{\alpha, 0, 2\alpha - 4\}$ for $\alpha \in \mathbb{R}$,

(c)

$$\begin{array}{ll} \min & c^\top x \\ \text{s.t.} & \|Ax - b\|_\infty \le \delta \\ & x \ge 0 \end{array}$$

where $x \in \mathbb{R}^n, A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$.

(a) $\min \quad 2x + 3a$

$$\begin{aligned} \text{s.t.} \quad & a \geq y - x \\ & a \geq x - y \\ & b + c \leq 5 \\ & b \geq x + 2 \\ & b \geq -x - 2 \\ & c \geq y \\ & c \geq -y \end{aligned}$$

(b) $\min \quad c^T x + z$

$$\begin{aligned} \text{s.t.} \quad & Ax \geq b \\ & z \geq d^T x \\ & z \geq 0 \\ & z \geq 2d^T x - 4 \end{aligned}$$

(c) $\min \quad c^T x$

$$\begin{aligned} \text{s.t.} \quad & A_i x - b_i \leq \delta,\ b_i - A_i x \leq \delta,\ \forall i \in [n] \\ & x \geq 0 \end{aligned}$$