Weilin Ye
120090513

Homework 1

Problem 1

(a) Assume that the company produces x units of the first product and y units of the second product. Thus,

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

(b)

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

$$\begin{aligned} \iff - \underset{x,y,s_1,s_2}{\text{minimize}} \quad & -(7.8x + 7.1y) \\ \text{subject to} \quad & \frac{1}{8}x + \frac{1}{4}y + s_1 = 90, \\ & \frac{1}{2}x + \frac{1}{6}y + s_2 = 80, \\ & x, y, s_1, s_2 \geq 0. \end{aligned}$$

(c) Let the company schedule z hours of overtime assembly labor. Thus,

$$\begin{aligned} - \underset{x,y,z,s_1,s_2,s_3}{\text{minimize}} \quad & -(7.8x + 7.1y - 8z) \\ \text{subject to} \quad & \frac{1}{8}x + \frac{1}{4}y + s_1 = 90, \\ & \frac{1}{2}x + \frac{1}{6}y + s_2 = 80, \\ & z + s_3 = 40, \\ & x, y, z, s_1, s_2, s_3 \geq 0. \end{aligned}$$

(d) Optimal solution: $x = 48$, $y = 336$. Optimal value: $+2760$
i.e., the company is supposed to produce 48 units of the first product and 336 units of the first product. There will be a profit of \$2760.

```
cvx_begin
    variables x(1) y(1),
    maximize ((9-1.2)*x + (8-0.9)*y)
    subject to
        1/8 *x + 1/4 *y ≤ 90
        1/2 *x + 1/6 *y ≤ 80
        0 ≤ x
        0 ≤ y
cvx_end
```

Problem 2

(a) For the sake of simplicity, we have

$$C = \begin{bmatrix} 999 & 20 & 13 & 11 & 28 \\ 20 & 999 & 18 & 8 & 46 \\ 13 & 18 & 999 & 9 & 27 \\ 11 & 8 & 9 & 999 & 20 \\ 28 & 46 & 27 & 20 & 999 \end{bmatrix}, \quad a = \begin{bmatrix} 40 \\ -135 \\ 200 \\ -220 \\ -220 \end{bmatrix},$$

where C_{ij} indicates the cost of moving a car from i to j and a_i indicates the needed cars in region i. Thus, the formulation is,

$$\begin{aligned} \underset{W \in \mathbb{R}^{5\times 5}}{\text{minimize}} \quad & \|C_{ij} \cdot W_{ij}\|_1 \\ \text{subject to} \quad & \sum_{i=1}^{5} W_{ik} - \sum_{j=1}^{5} W_{kj} \geq a_k, \quad \forall k = 1, 2, \ldots, 5, \\ & W_{ij} \geq 0, \quad \forall i, j = 1, 2, \ldots, 5., \\ & \text{diag}(W) = 0. \end{aligned}$$

(b) MATLAB code and output:

```
C = [999, 20, 13, 11, 28;
    20 , 999, 18, 8, 46;
    13, 18, 999, 9, 27;
    11, 8, 9, 999, 20;
    28, 46, 27, 20, 999]
cvx_begin quiet
    variables W(5, 5)
    minimize sum(sum(C.*W))
    subject to
        sum(W(:,1))-sum(W(1,:))≥40
        sum(W(:,2))-sum(W(2,:))≥-135
        sum(W(:,3))-sum(W(3,:))≥200
        sum(W(:,4))-sum(W(4,:))≥-220
        sum(W(:,5))-sum(W(5,:))≥-220
        W(:,:) ≥ 0
cvx_end
W
```

```
W =

         0    0.0000    0.0000    0.0000    0.0000
    0.0000         0    0.0000   20.0000    0.0000
    0.0000    0.0000         0    0.0000    0.0000
   40.0000    0.0000  200.0000         0    0.0000
    0.0000    0.0000    0.0000    0.0000         0
```

Problem 3

(a) For the sake of simplicity, we have

$$C = \begin{bmatrix} 999 & 5 & 4 & 999 & 999 & 999 & 999 & 999 \\ 5 & 999 & 999 & 3 & 999 & 7 & 999 & 999 \\ 4 & 999 & 999 & 999 & 1 & 2 & 999 & 999 \\ 999 & 3 & 999 & 999 & 2 & 999 & 999 & 999 \\ 999 & 999 & 1 & 2 & 999 & 999 & 2 & 5 \\ 999 & 7 & 2 & 999 & 999 & 999 & 999 & 3 \\ 999 & 999 & 999 & 999 & 2 & 999 & 999 & 1 \\ 999 & 999 & 999 & 999 & 5 & 3 & 1 & 999 \end{bmatrix},$$

where C_{ij} indicates the distance between i and j, 999 means there is no edge between two nodes. Thus, the formulation is

$$
\begin{aligned}
\underset{W \in \mathbb{R}^{8\times 8}}{\text{minimize}} \quad & \|C_{ij} \cdot W_{ij}\|_1 \\
\text{subject to} \quad & \sum_{i=1}^{8} W_{1i} = 1, \\
& \sum_{i=1}^{8} W_{i8} = 1, \\
& \operatorname{diag}(W) = 0, \\
& \sum_{i=1}^{8} W_{ik} = \sum_{j=1}^{8} W_{kj}, \quad && \forall k = 1, 2, \dots, 8, \\
& 0 \le W_{ij} \le 1. && \forall i, j = 1, 2, \dots, 8.
\end{aligned}
$$

```
C = [999 5 4 999 999 999 999 999;
        5 999 999 3 999 7 999 999;
        4 999 999 999 1 2 999 999;
        999 3 999 999 2 999 999 999;
        999 999 1 2 999 999 2 5;
        999 7 2 999 999 999 999 3;
        999 999 999 999 2 999 999 1;
        999 999 999 999 5 3 1 999]

cvx_begin quiet
    variables W(8, 8)
    minimize sum(sum(C.*W))
    subject to
        sum(W(1,:)) == 1
        sum(W(:,8)) == 1
        for i = 2:7
            sum(W(:,i)) == sum(W(i,:))
        end
        W(:,:) ≥ 0
        W(:,:) ≤ 1
        diag(W) == 0
cvx_end
W
```

```
>> format rat
>> hw1_3
W =

        0     *     1     *     *     *     *     *
        *     0     *     *     *     *     *     *
        *     *     0     *     1     *     *     *
        *     *     *     0     *     *     *     *
        *     *     *     *     0     *     1     *
        *     *     *     *     *     0     *     *
        *     *     *     *     *     *     0     1
        *     *     *     *     *     *     *     0
```

Hence, the shortest path is $1 \to 3 \to 5 \to 7 \to 8$.