Math 346, HW10 Solution

7.1.1 (b)

We solve the distribution problem using the algorithm described in Section 7.1. Note that the total supply is $4 + 6 + 7 + 8 = 25$, and the total demand is $4+3+6+5+6 = 24$, we can add a new destination with demand $25-24 = 1$, and the links from every origin to this destination is defined to be $+\infty$.

We first greedily fill in the numbers:

3	3	2	1	0		1		5		∞		4
1	1	0		3	3	2	2	2		∞		6
1		2	2	3	3	3	2	0		∞		7
0		1		4		4	1	1	1	∞	1	8
4		3		6		5		6	5	1		

Note that the fourth row has surplus and the fifth column's demand has not been met (with a deficit of 5). Now we can start the row/column labelling process. We first label column 2, 3, 4, 6 by row 4. Then row 1 by columns 2, row 2 by column 2, 3, 4, row 3 by column 3,4 and row 4 by column 4. Then we can label column 5 which corresponds to the demand not yet met. In particular this gives us a path Row 4 $-$ Column 4 $-$ Row 2 $-$ Column 5, where we could decrease the deficit by 2. The new table is:

3	3	2	1	0		1		5		∞		4
1	1	0		3	3	2		2	2	∞		6
1		2	2	3	3	3	2	0		∞		7
0		1		4		4	3	1	1	∞	1	8
4		3		6		5		6	3	1		

Similarly we can find Row 4 - Column 2 - Row 1 -Column 5, which gives:

3	3	2		0		1		5	1	∞		4
1	1	0		3	3	2		2	2	∞		6
1		2	2	3	3	3	2	0		∞		7
0		1	1	4		4	3	1	1	∞	1	8
4		3		6		5		6	2	1		

We can find Row 4 - Column 4 - Row 3 - Column 1 - Row 1 - Column 5, which gives:

3	2	2		0		1		5	2	∞		4
1	1	0		3	3	2		2	2	∞		6
1	1	2	2	3	3	3	1	0		∞		7
0		1	1	4		4	4	1	1	∞	1	8
4		3		6		5		6	1	1		

We can label column 3, 6 by row 4, row 2 and 3 by column 3, column 4 by row 2 and 3. The labeling process terminates here. And note that we have not reached the column having a deficit (that is column 5), which means the problem is infeasible.

One can also check it is infeasible by looking at the labelled rows $R = \{2, 3, 4\}$, and the labelled columns $C = \{3, 4, 6\}$. Using Theorem 7..1.2, we can check that $\sum_{j \in C'} b_j = b_1 + b_2 + b_5 = 4 + 3 + 6 = 13$, $\sum_{i \in R'} a_i = a_1 = 4$, $\sum_{j \in C'} \sum_{i \in R} k_{ij} = 8$. Since $13 > 4 + 8$ so the problem is infeasible.

7.2.1 (a)

We start by consider the dual of the transportation problem. And we can take $u_1 = \min\{5, 6, 7, 5\} = 5$, $u_2 = 6$, $u_3 = 2$. Then $v_1 = \min\{c_{11} - u_1, c_{21} - u_2, c_{31} - u_3, c_{41} - u_4\} = 0$, $v_2 = 1$, $v_3 = 2$, $v_4 = 0$. We can check for which i, j, $u_i + v_j$ is equal to c_{ij}, and create a distribution problem with these links having capacity ∞ and all other links having capacity 0. The distribution problem is infeasible since the greedy assignment and the labeling process give $R = \{2\}$ and $C = \{4\}$. Now we modify $\{u_i\}$ and $\{v_j\}$ according to step 4 in the algorithm (on page 267). $d = \min_{i \in R, j \notin C} c_{ij} - (u_i + v_j) = 1$. So now $u_1 = 5$, $u_2 = 7$, $u_3 = 2$, and $v_1 = 0$, $v_2 = 1$, $v_3 = 2$, $v_4 = -1$.

We create a new distribution problem according to the new $\{u_i\}$ and $\{v_j\}$ and solve it (with $k_{12} = k_{13} = k_{14} = k_{23} = k_{24} = k_{34} = \infty$ and the rest link capacity being zero). The greedy assignment and the labeling process gives $R = \{2, 3\}$ and $C = \{3, 4\}$. Now d is equal to 1, and the new $u_1 = 5$, $u_2 = 8$, $u_3 = 3$, and $v_1 = 0$, $v_2 = 1$, $v_3 = 1$, $v_4 = -2$.

Again we create a new distribution problem according to the new $\{u_i\}$ and $\{v_j\}$ and solve it (with $k_{12} = k_{13} = k_{23} = k_{24} = k_{31} = k_{32} = k_{33} = \infty$ and the rest link capacity being zero). This distribution problem has a feasible solution with $x_{11} = 9$, $x_{12} = 3$, $x_{23} = 6$, $x_{24} = 8$, $x_{31} = 0$, $x_{32} = 5$, $x_{33} = 5$. This gives an optimal solution to the transportation problem, with minimum cost being: $\sum_{i,j} c_{ij} x_{ij} = 5 \times 9 + 6 \times 3 + 9 \times 6 + 6 \times 8 + 4 \times 5 + 4 \times 5 = 45 + 18 + 54 + 48 + 20 + 20 = 205$.

7.3.23 (b)

We use the Hungarian algorithm to solve this assignment problem. First we add an extra row of zeros to make the number of persons equal to number jobs. Then we subtract the minimum entry for each row from the entire row (note that we don't have to do this for columns since the minimum entry for each column is now zero), we get:

3	3	4	5	0
1	3	1	0	2
0	3	2	3	1
1	3	2	3	0
0	0	0	0	0

We need to cover the zeros by at least 4 lines, as can be seen from the cover of Row 5, Column 5, Row 2, Row 3, and the observation that there are four zeros at the positions $(1,5)$, $(2,4)$, $(3,1)$, $(5,2)$, none of which lie on the same row or column.

Now we consider the uncovered rows and columns, the smallest element is equal to 1, so we subtract 1 from the uncovered rows and add 1 to the covered columns, we get:

2	2	3	4	0
1	3	1	0	3
0	3	2	3	2
0	2	1	2	0
0	0	0	0	1

Now for this new assignment problem, we repeat the Hungarian algorithm: the minimum cover of zeros would be Column 1,4,5 and row 5. Now again we modify the matrices by subtracting from the uncovered rows and add to the covered columns:

2	1	2	4	0
1	2	0	0	3
0	2	1	3	2
0	1	0	2	0
1	0	0	1	2

We apply the algorithm one more time:

Now we can find an optimal assignment consisting of only zeros (1-5, 2-4, 3-1, 4-3, 5-2). This is also the optimal assignment for the original problem, which gives minimum cost: $3 + 1 + 6 + 2 = 12$.

7.3.35

This proof is via creating an auxiliary graph. But you can also state it in non-graph-theoretic languages. First we assume that the number of machines is equal to the number of jobs (otherwise we add extra machines or jobs and take the value to be zero), both equaling n. We create a bipartite graph such that each side contains n edges, on the left every vertex corresponds to a machine, and on the right every vertex corresponds to a job. We connect two vertices by a green edge if this is in the optimal assignement. We also add a red edge from every vertex on the left to its best job on the right. Now we prove by contradiction, suppose every machine is not assigned its best job, we attempts to improve this assignment which would contradicts its optimality. This could be done by proving the existence of a red-green alternating cycle, say $M_{i_1} - J_{j_1} - M_{i_2} \cdots - J_{j_k}$, where the first edge is red, second edge green, third edge red, etc. Since every vertex on the left is incident to some red edge, and the green edges form a matching, we know we can greedily create such sequence until we hit one of the vertex on the left visited before. So such color-alternating cycle always exists, and by reassigning the jobs J_{j_x} to M_{i_x} for every $1 \leq x \leq k$, we could improve the assignment which results in contradiction.