IE313 Supply Chain Management

Assignment 3

7.6.2023

Fatmanur Yaman 2019402204

Ömercan Mısırlıoğlu 2020402261

1) *Choose a formulation and a strategy to solve TSP. Download TR81 cities data from the course web page.*

In the formulation, the objective function (1), and constraints (2), (3), and (5) are used. The Miller-Tucker-Zemlin constrains are used for 15 cities subtour elimination.

$$\min \quad \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

Subject to:

$$\sum_{j:j\neq i} x_{ij} = 1 \qquad j = 1, 2, \cdots, n$$
$$\sum_{i:i\neq j} x_{ij} = 1 \qquad i = 1, 2, \cdots, n$$

$$u_i - u_j + n x_{ij} \leq n - 1, \qquad i \neq j, i, j = 2, 3, \cdots n$$

$$x_{ij} \in \{0, 1\} \qquad \forall i, j$$

2) *Solve the TSP for randomly chosen 15 cities. Report your findings.*

The code first extracted the data from the Excel file and transformed it into a data frame in Python. The order of the data frame was then arranged, the indices were changed, and some of the columns were removed to make it easier to work with. In this step, null parts were filled with infinity value. Finally, 15 cities were randomly selected and assigned to a new data frame.

The random 15 cities which found with seed value “1923” are as below:

['ŞANLIURFA', 'SAKARYA', 'KAYSERİ', 'ZONGULDAK', 'VAN', 'BAYBURT', 'AFYON', 'ARTVİN', 'TUNCELİ', 'OSMANİYE', 'SİNOP', 'SİİRT', 'KÜTAHYA', 'İÇEL', 'BİNGÖL']

After arranging the data and randomly selecting 15 cities, the code started to write the mathematical model. The model definition, decision variables, and objective function were written first. Then, the constraints were added. The constraints ensured that each city had exactly one incoming edge and one outgoing edge. This means that each city must be reached from another city, and a city must be selected to go from each city.

The constraints including MTZ (subtour elimination) constraints are as below:

```python
for i in i_index:
    model += lpSum(X[i][j] for j in i_index if i != j) == 1

for i in i_index:
    model += lpSum(X[j][i] for j in i_index if i != j) == 1

    for i in range(1,15):
        for j in range(1,15):
            if i != j:
                model += U[i] - U[j] <=  14 - 15 *  X[i][j]
```

Solving the problem, the resulting optimal tours cost is **4642,** and the tour is shown below:

'ŞANLIURFA' → 'OSMANİYE' → 'İÇEL' → 'KAYSERİ' → 'AFYON' → 'KÜTAHYA' → 'SAKARYA' → 'ZONGULDAK' → 'SİNOP' → 'ARTVİN' → 'BAYBURT' → 'TUNCELİ' → 'BİNGÖL' → 'VAN' → 'SİİRT' → 'ŞANLIURFA'

All of the **codes** for this question are provided in the zip file.

3) *Solve the TSP for 81 cities. Report your findings.*

Initially, an attempt was made to implement the PULP package to solve the optimal tour for 81 cities. However, this took too much time and the result was not obtained. Therefore, another solution was searched for on the internet. As a result, the Gurobi package was found. There are several ways to reach a solution in the Gurobi package. One of them was used. The formulation was changed: instead of using MTZ constraints, standard subtour elimination constraints with S subset were used, as mentioned above.

That part was excluded from the code. Then, the approach was changed to Gurobi. The necessary packages were imported. The academic license was obtained free of charge from Gurobi, and then the code was implemented. The functions to be used were defined, such as the subtour elimination function and the shortest path finding function. The number of cities, n, was assigned as 81. A distance matrix was used, and the model was defined. Then, the constraints and variables were added. Finally, the output and model optimization were obtained.

The resulting cost for 81 cities optimal tour is **9938.**

All of the **codes** for this question are provided in the zip file.

4) The optimal tour for 81 cities is shown below: