IE313 Supply Chain Management

Assignment 3

07.06.2023

BOĞAZİÇİ ÜNİVERSİTESİ

1863

Group 1

Tevfik Buğra Türker 2019402120

Hüseyin Emre Bacak 2021402279

Q1) Choose a formulation and a strategy to solve TSP.

We formulated the desired TSP as sequential formulation of Miller, Tucker, Zemlin (MTZ) along with basic assignment constraints since it is a very large instance of interest regarding 81 cities of Turkey.

Parameters

c_{ij} → the distance between city i and city j

Decision variables

$$x_{ij} = \begin{cases} 1, & \text{if the salesperson visits city j directly after visiting city i} \\ 0, & \text{otherwise} \end{cases}$$

Also it is important to note that x_{ii}'s are not defined.

u_i → to indicate the sequence in which city i is visited (i = 2, …, n)

Also, u_1 is not defined.

Objective function

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

Constraints

$$\sum_{j:j \neq i} x_{ij} = 1 \qquad i = 1, 2, \ldots, n \qquad \textbf{Assignment constraints I}$$

$$\sum_{i:i \neq j} x_{ij} = 1 \qquad j = 1, 2, \ldots, n \qquad \textbf{Assignment constraints II}$$

$u_i - u_j + nx_{ij} \le n-1 \qquad i \ne j, \quad i, j = 2, 3, \dots, n$ **Subtour elimination constraints**

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

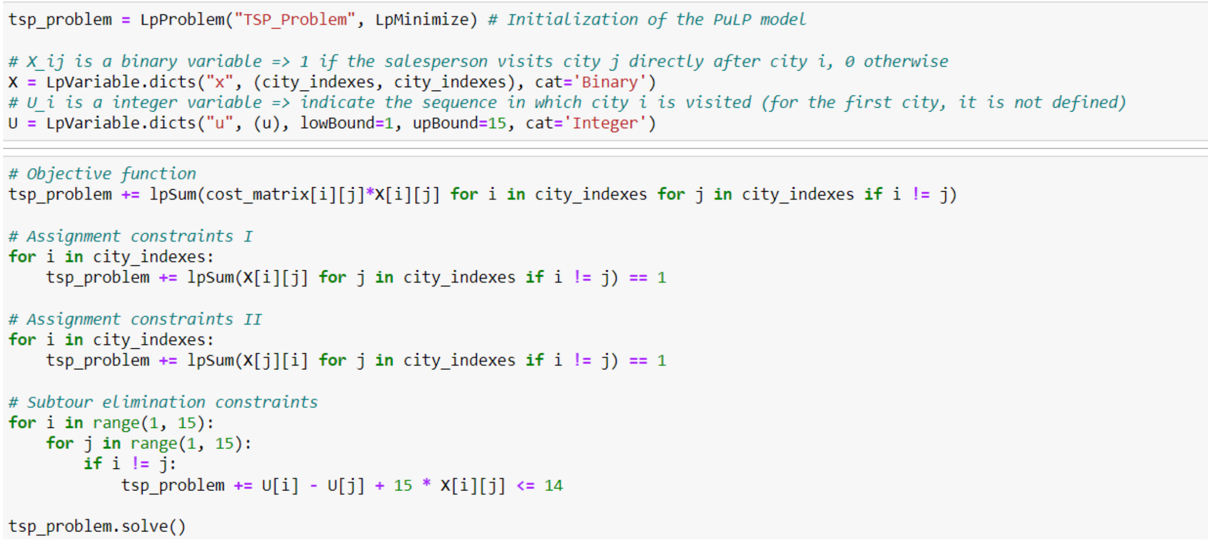
$2 \le u_i \le n \qquad \forall i$

Although it is a “weak formulation”, we reduce the traditional amount of subtour elimination constraints from $O(2^n)$ to $O(n^2)$.

Q2) Solve the TSP for randomly chosen 15 cities. Report your findings.

We converted the raw data in the MS Excel file containing the distances between each cities into a data frame object in Python using “pandas”. We put “np.inf” (a very large float which can be assumed as infinity) to represent distance between a city itself to make x_{ii}’s undefined. After that, we randomly sampled 15 city indexes out of 81 available. In an instance, 15 randomly chosen cities were;
ADIYAMAN, ANKARA, BİTLİS, GÜMÜŞHANE, HATAY, İÇEL, İSTANBUL, ORDU, SİİRT, SİNOP, TOKAT, ŞANLIURFA, BATMAN, ARDAHAN, DÜZCE

After random sampling, we coded the above decision variables, objective function, and constraints into Python. Since 15 cities case is comparably small instance, we used PuLP solver.



```
tsp_problem = LpProblem("TSP_Problem", LpMinimize) # Initialization of the PuLP model

# X_ij is a binary variable => 1 if the salesperson visits city j directly after city i, 0 otherwise
X = LpVariable.dicts("x", (city_indexes, city_indexes), cat='Binary')
# U_i is a integer variable => indicate the sequence in which city i is visited (for the first city, it is not defined)
U = LpVariable.dicts("u", (u), lowBound=1, upBound=15, cat='Integer')
```

```
# Objective function
tsp_problem += lpSum(cost_matrix[i][j]*X[i][j] for i in city_indexes for j in city_indexes if i != j)

# Assignment constraints I
for i in city_indexes:
    tsp_problem += lpSum(X[i][j] for j in city_indexes if i != j) == 1

# Assignment constraints II
for i in city_indexes:
    tsp_problem += lpSum(X[j][i] for j in city_indexes if i != j) == 1

# Subtour elimination constraints
for i in range(1, 15):
    for j in range(1, 15):
        if i != j:
            tsp_problem += U[i] - U[j] + 15 * X[i][j] <= 14

tsp_problem.solve()
```

Figure 1. Defining decision variables, objective function, and constraints in PuLP

Then, we solved the problem and found a optimal tour. The optimal tour for the above 15 cities is;

ADIYAMAN - HATAY - İÇEL - ANKARA - İSTANBUL - DÜZCE -SİNOP - TOKAT - ORDU - GÜMÜŞHANE - ARDAHAN - BİTLİS - SİİRT - BATMAN - ŞANLIURFA - ADIYAMAN

And the total optimal distance travelled for these 15 randomly chosen cities were 4478 kms.

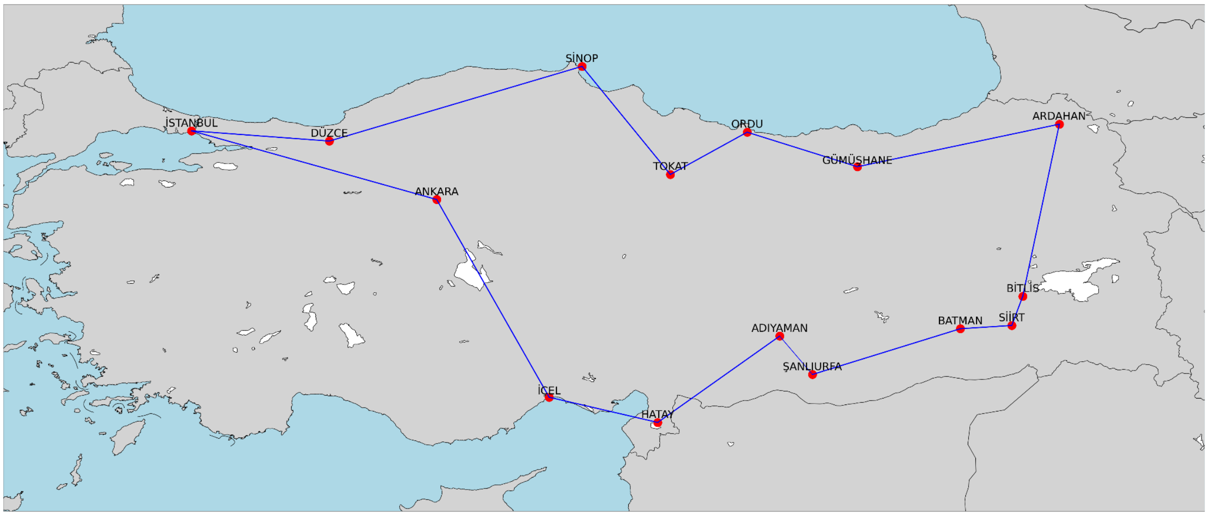
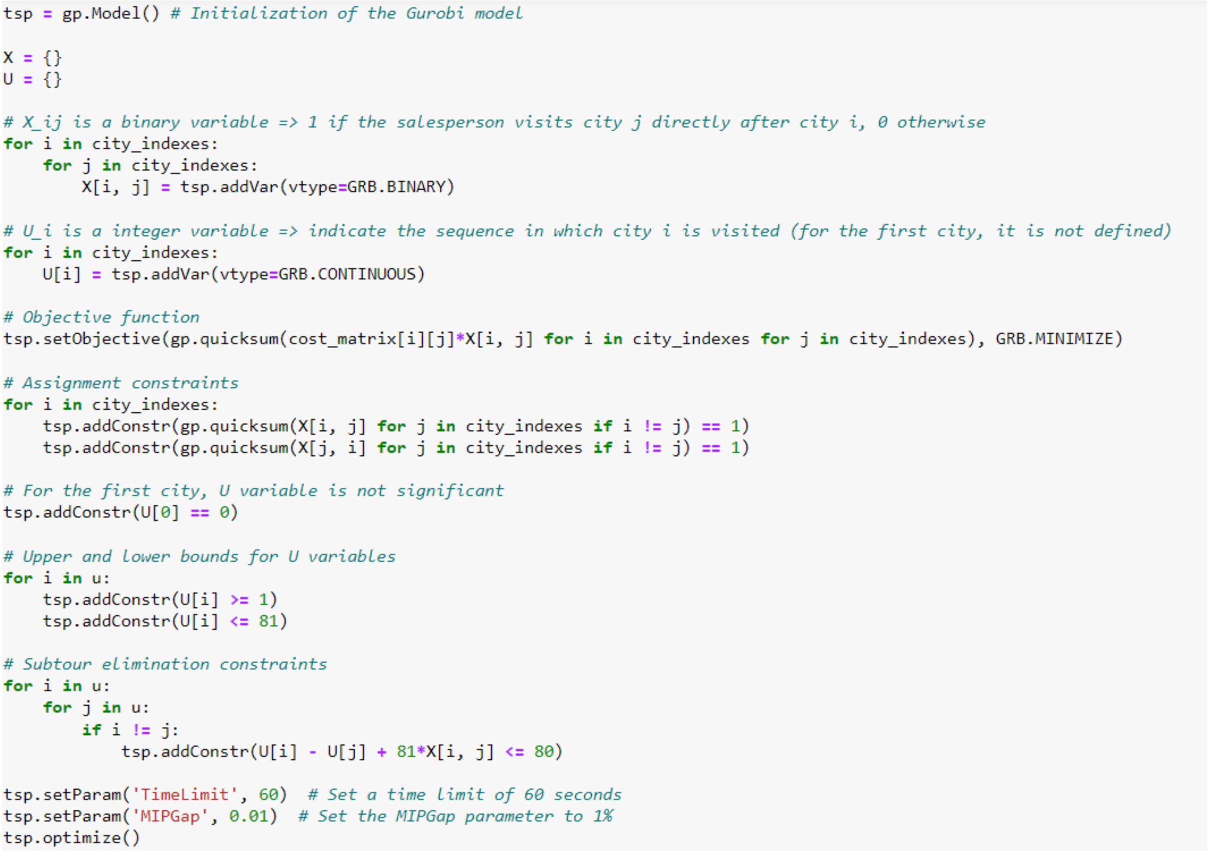


Figure 2. Plot of the optimal tour for the 15 randomly chosen cities of Turkey

Q3) Solve the TSP for 81 cities. Report your findings.

In this case, we first tried to use PuLP solver. However, it takes so much time and we could not get a result since this instance is comparably large. Then, we decided to use Gurobi solver for Python to formulate and solve the TSP for 81 cities.



```
tsp = gp.Model() # Initialization of the Gurobi model

X = {}
U = {}

# X_ij is a binary variable => 1 if the salesperson visits city j directly after city i, 0 otherwise
for i in city_indexes:
    for j in city_indexes:
        X[i, j] = tsp.addVar(vtype=GRB.BINARY)

# U_i is a integer variable => indicate the sequence in which city i is visited (for the first city, it is not defined)
for i in city_indexes:
    U[i] = tsp.addVar(vtype=GRB.CONTINUOUS)

# Objective function
tsp.setObjective(gp.quicksum(cost_matrix[i][j]*X[i, j] for i in city_indexes for j in city_indexes), GRB.MINIMIZE)

# Assignment constraints
for i in city_indexes:
    tsp.addConstr(gp.quicksum(X[i, j] for j in city_indexes if i != j) == 1)
    tsp.addConstr(gp.quicksum(X[j, i] for j in city_indexes if i != j) == 1)

# For the first city, U variable is not significant
tsp.addConstr(U[0] == 0)

# Upper and lower bounds for U variables
for i in u:
    tsp.addConstr(U[i] >= 1)
    tsp.addConstr(U[i] <= 81)

# Subtour elimination constraints
for i in u:
    for j in u:
        if i != j:
            tsp.addConstr(U[i] - U[j] + 81*X[i, j] <= 80)

tsp.setParam('TimeLimit', 60)  # Set a time limit of 60 seconds
tsp.setParam('MIPGap', 0.01)  # Set the MIPGap parameter to 1%
tsp.optimize()
```

Figure 3. Defining decision variables, objective function, and constraints in Gurobi

Then, we found the optimal tour for 81 cities as;

ADANA - İÇEL - KARAMAN - KONYA - AKSARAY - NEVŞEHİR - NİĞDE - KAYSERİ - SİVAS - TOKAT - AMASYA - ÇORUM - YOZGAT - KIRŞEHİR - KIRIKKALE - ÇANKIRI - ANKARA - ESKİŞEHİR - KÜTAHYA - AFYON - UŞAK - ISPARTA - BURDUR - ANTALYA - DENİZLİ - MUĞLA - AYDIN - İZMİR - MANİSA - BALIKESİR - ÇANAKKALE - EDİRNE - KIRKLARELİ - TEKİRDAĞ - İSTANBUL - KOCAELİ - YALOVA - BURSA - BİLECİK - SAKARYA - BOLU - DÜZCE - ZONGULDAK - BARTIN - KARABÜK - KASTAMONU - SİNOP - SAMSUN - ORDU - GİRESUN - TRABZON - RİZE - ARTVİN - ARDAHAN - KARS - IĞDIR - AĞRI - ERZURUM - BAYBURT - GÜMÜŞHANE - ERZİNCAN - TUNCELİ - ELAZIĞ - MALATYA - BİNGÖL - MUŞ - BİTLİS - VAN - HAKKARİ - ŞIRNAK - SİİRT - BATMAN - DİYARBAKIR - MARDİN - ŞANLIURFA - ADIYAMAN - KAHRAMANMARAŞ - GAZİANTEP - KİLİS - HATAY - OSMANİYE - ADANA

And the total optimal distance travelled for 81 cities were 9938 kms.

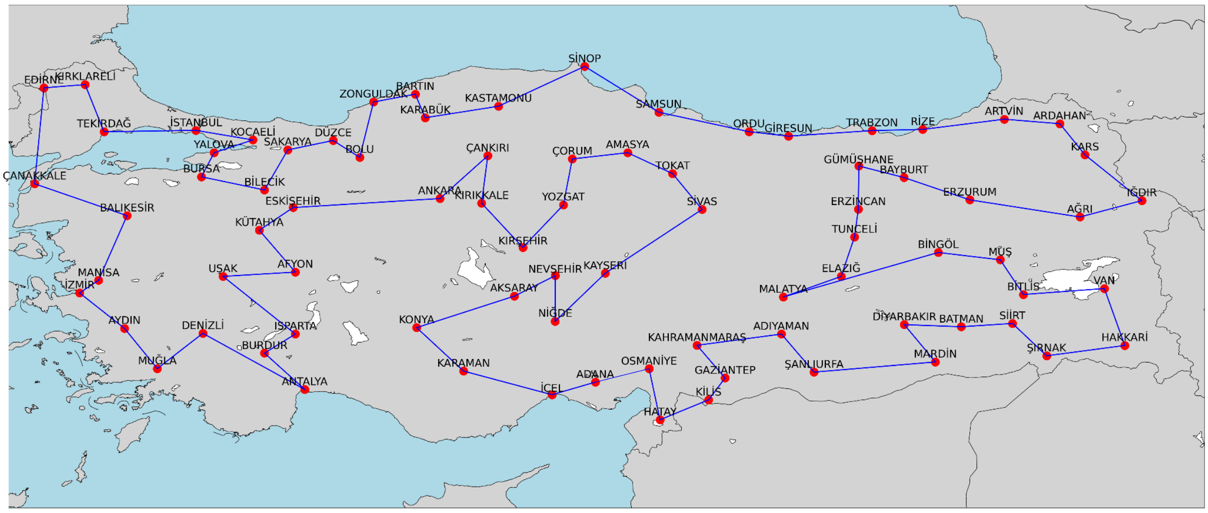


Figure 4. Plot of the optimal tour for whole 81 cities of Turkey