Geometric Graphs: Embedding, Matching and Registration

Consider the dataset assigned to your team composed of three observed data sets and three targets, all collected in one archive:

- 'ObservedDataSet1_dist.txt' , 'ObservedDataSet2_dist.txt' , 'ObservedDataSet3_dist.txt'
- 'Target1_coord.txt' , 'Target2_coord.txt' , 'Target3_coord.txt'

The file formats are similar to those in your team homeworks. Specifically, the '*_dist.txt' files have the format:

```
line 1: n m
line 2: i1 j1 d(i1,j1)
line 3: i2 j2 d(i2,j2)
line 4: i3 j3 d(i3,j3)
...
line m+1: im jm d(im,jm)
```

where n denotes the number of vertices of a geometric graph, m denotes the number of available edge distances. Lines 2 to $m+1$ list these distances: $i1, j1, i2, j2, ..., im, jm$ denote vertices (from 1 to n), and $d(i1, j1)$, $d(i2, j2)$, ... , $d(im, jm)$ represent the distances respectively between these pairs of vertices, $(i1, j1)$, $(i2, j2)$, and so on.

The '*_coord.txt' files have the format:

```
line 1: x1 y1 z1
line 2: x2 y2 z2
line 3: x3 y3 z3
line 4: x4 y4 z4
...
line n: xn yn zn
```

where n denotes the number of vertices (points) of a 3D geometric graph, and each line constains the (x, y, z) coordinates of these points.

All your data files refer to geometric graphs with same number of vertices (nodes) n. The number of edges (pairwise distances) in 'ObservedDataSet's may vary from 0 to a maximum of $n(n-1)/2$.

Your dataset has 3 additional files: sgb128_xy.png , sgb128_xy.txt and sgb128_name.txt The file 'sgb128_xy.txt' has coordinates of 128 cities in US whose names are listed in 'sgb128_name.txt'. The coordinates of the first 40 cities are included in the file 'Target1_coord.txt'; the coordinates of the next 40 cities (41 to 80) are included in 'Target2_coord.txt', and the coordinates of followig 40 cities (81 to 120) are included in 'Target3_coord.txt'.

Your project should perform the following tasks:

For each observed data set, 'ObservedDataSet1.dist' to 'ObservedDataSet3.dist':

1. Produce a 3D embedding of the geometric graph using either the full data embedding algorithm (if full data avilable), or the geometric graph partial

data embedding algorithm (if onlly partial data is available). For the latter algorithm try a few values for ε_G parameter; choose the parameter that minimizes the matching error computed at part 3).

2. Determine the optimal full alignment transformation to each of the target graphs, 'Target1.coord' ... 'Target4.coord'.

3. Compute the alignment errors and present the results in a $3x3$ table (known also as, or similar to the "confusion matrix" or "table of confusion"): each entry (i, j) should contain the alignment error of matching the observed data i to target j.

4. For observed data set i, determine the target $\hat{j}$ that has the smallest alignment error. Create a movie (.avi file) that contains the continuous transformation from the embedding produced at 1) to the optimal target graph $\hat{j}$. Use the procedure described in Homework 3 for graph visualization, continuous transformation, and video creation.

Guess which ObservedDataSet each of the following three cities belongs to:

1. Wilmington, DE (from Target 1 Dataset)

2. Tampa, FL (from Target 2 Dataset)

3. San Francisco, CA (from Target 3 Dataset)