# Analysis

Input:

locations of n nodes with their coordinates

Job:

create a network topology(undirected graph) with following properties:

1. Contains all nodes
2. Degree of each vertex is at least 3
3. Diameter of the graph is at most 4(hop-distance)
4. Total cost of the network topology is as low as possible by the total geometric length of all links

Goal:

Implement two different heuristic algorithm(it does not have to guarantee the exact optimum)

E.g: Branch and Bound, Simulated Annealing, Greedy Local Search, Tabu Search, Genetic Algorithm

Creative ideas will be appreciated.

Two algorithms should be sufficiently different to compete in finding good solution.

Tasks:

* Describe the two algorithms.
* Provide reference to the source
* Provide pseudo code with sufficient comments
* Run the program on randomly generated examples(at least 5 examples), pick n random points in the plane, this can be done by generating random numbers in some range and taking them as coordinates, n >= 15
* show result(nodes` position) graphically.
* Draw some conclusion about how the two algorithms compare

Design

1.Given n nodes with coordinates

\* how to identify coordinates.

\* each node will have coordinate variables: x, y

## Random Nodes Generating

The project asks to given the location of n nodes in the plane by their coordinates. Assume there will be 15 nodes with random coordinates be created.

Assume coordinates are in the range from 0 to 9, we could generate random nodes like: n1(0,0), n2(9,0)…., but there should not have duplicate coordinates thus duplication prevention should be considered.As figure 1:

And show them in the graph:

Hashset is used to prevent coordinate duplication of each node, whenever the generated coordinates are unique, then we create new node with the coordinates, and added to node array.

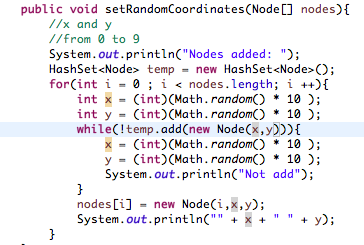


Figure 1

Swing and awt libraries are used to create coordinate system, and draw nodes.



Figure 1.1



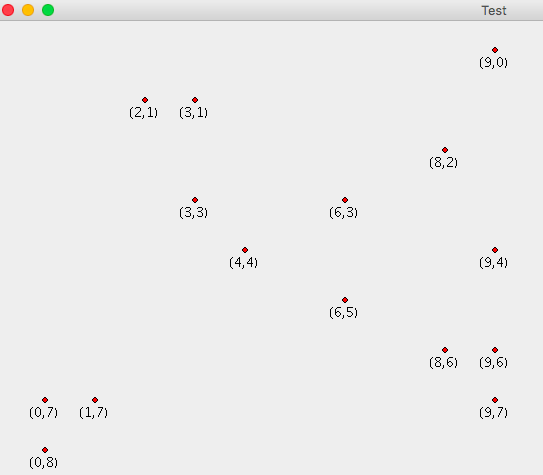


Figure 2

Figure 2 is one random set of nodes created.

## Heuristic Algorithms

What we need to do is to create a network topology that:

1. It will contains all the generated nodes(the graph is complete connected)

2. The diameter of the graph is at most 4(refers to the hop-distance).

3. Total cost of the network topology is as low as possible(measured by geometric length).

### Branch and Bound Algorithm

B : set of all n-dimensional binary vectors(all combinations), here it represent if one node has link to each other nodes.

f(x): maximum value over B, which represent as the connection of each node leads to the final total geometric length of all links.



Bk(b) : subset of B which fixed the first k coordinates, which means the connections of first k nodes has been decided. when k = n, we fixed all the links.

#### Pseudo Code

#### Implementation



Figure 3

First, we assume all nodes are disconnected, each time we pick one unfixed node as starting point, because degree of each vertex in the graph is at least 3, thus each node should has at least 3 outgoing edges.

For each starting point, we pick 3 outgoing links randomly, then compare these 3 links with all other potential links to get 3 links with smallest geometric distance, getGeometricDistance() method will calculate geometric distance between two nodes as Figure 4:



Figure 4

Before we add these 3 links to the final graph, check if graph`s diameter will exceed 4 as Figure 5, if yes, don`t add that link.

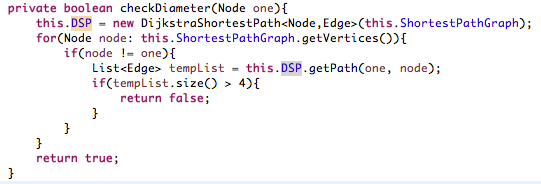
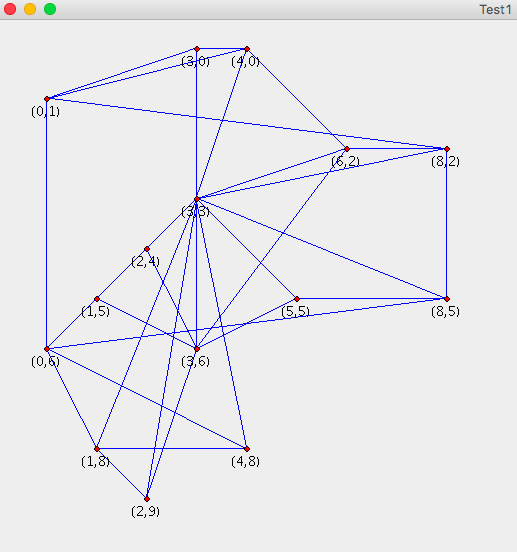


Figure 5

#### 5 Sample Runs





### Genetic Algorithm

#### Pseudo Code

#### Implementation