School Bus Routing Problem

Yuanqing (Jennifer) Guo
Wan-Yi (Lisa) Shen
Qiuman Xu

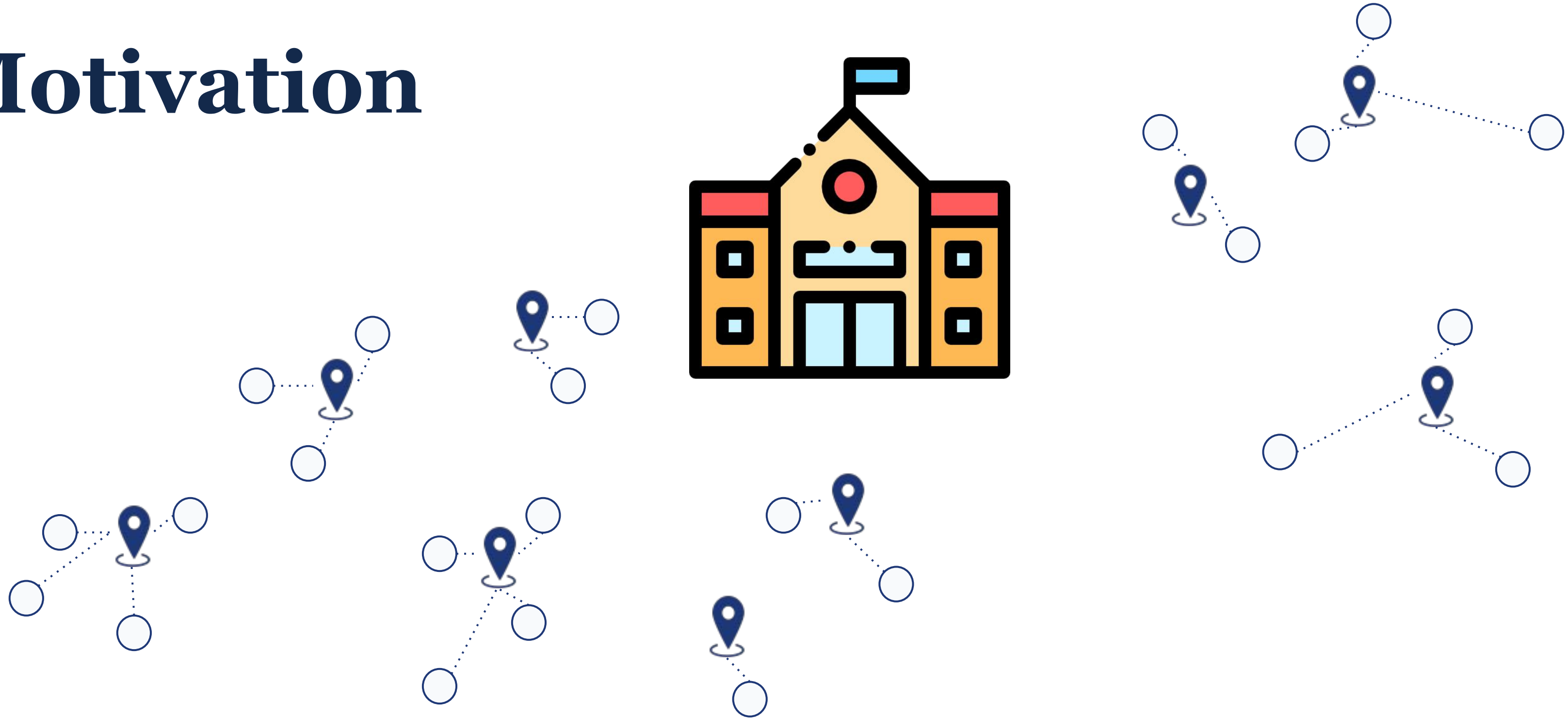
UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Outline

- Introduction
- Network Generation
- Formulation
- Data Processing
- Results/Analysis
- Problems and Solutions
- Improvement
- Conclusion

1. Introduction

1.1 Motivation



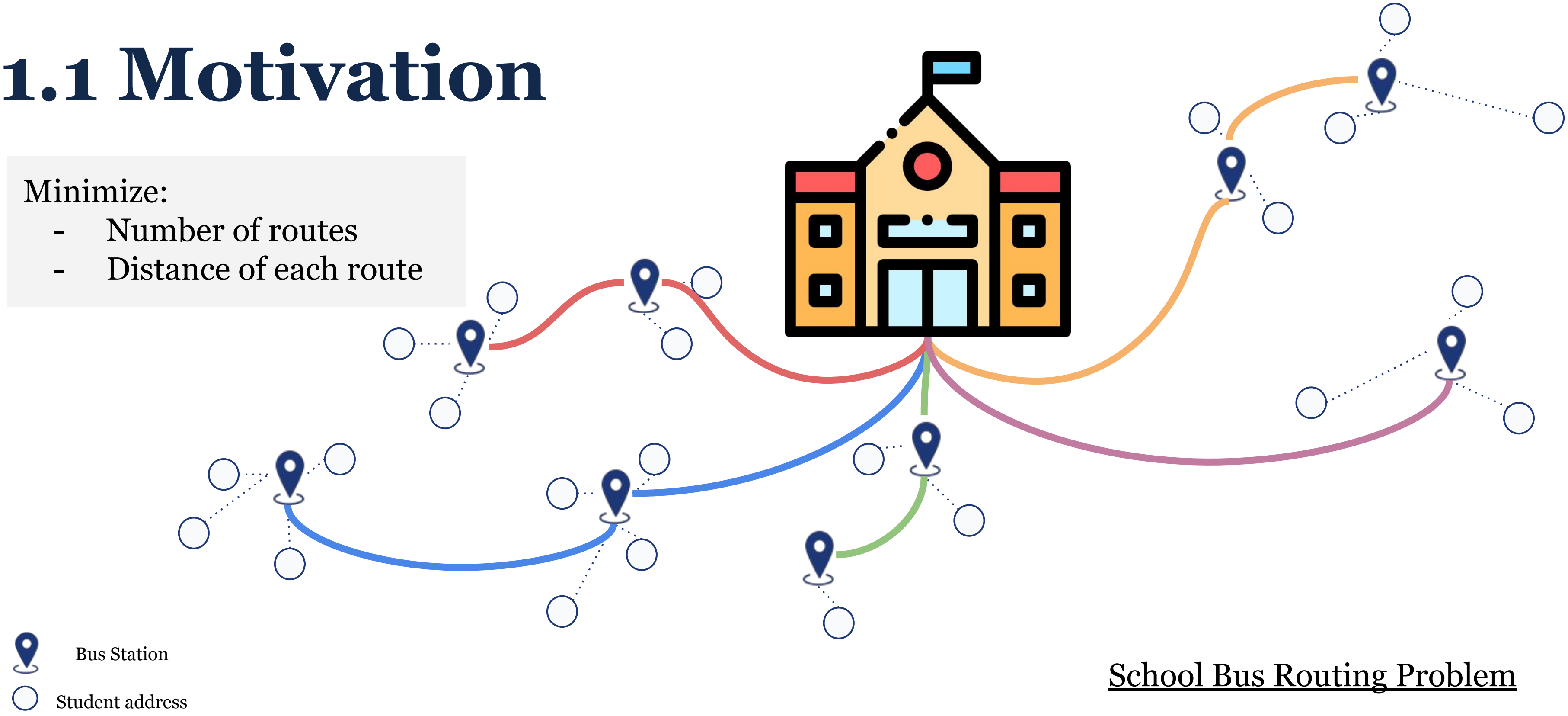
Bus Station

Student address

School Bus Routing Problem

1.1 Motivation

Minimize:
- Number of routes
- Distance of each route



School Bus Routing Problem

1.2 DataSet

- **Total # of stations:** 146 stops
- **Stop_id [0]:** School
- **Stop location:**(latitude,longitude)
- **Number[i]:**

  demand (# of student) on stop[i]

demand

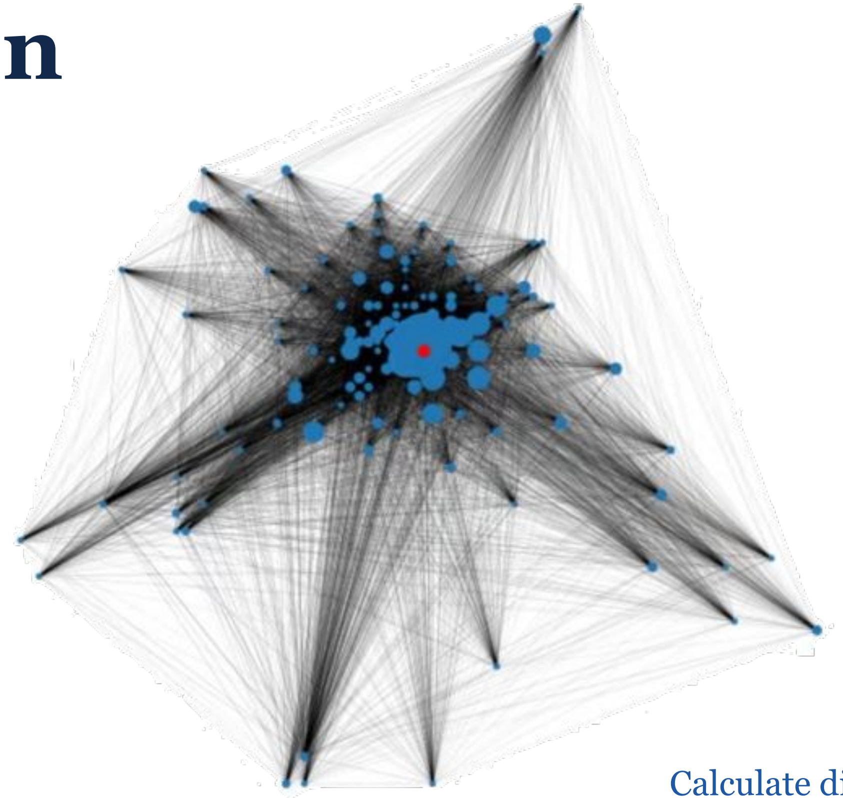
Stop_id	Station	Latitude	Longitude	Number
0	school	31.21	121.49	0
1	七宝镇	31.16	121.35	8
2	万祥镇	30.97	121.82	1
3	三林镇	31.14	121.5	15
4	上钢新村街道	31.17	121.48	7
5	东明路街道	31.14	121.53	2
6	中山街道	31.04	121.25	1
7	九亭镇	31.13	121.33	2
8	书院镇	30.98	121.87	1
9	云莲一居委	31.1	121.29	1
10	五角场街道	31.31	121.52	3

n: 147

⋮

2. Network Generation

Visualization



Blue nodes: stations

Red node: School

Haversine formula

Calculate distance between each station

3. Formulation

Mathematical Model

Data

- $V=\{0,1,2,\ldots,146\}$, set of nodes
- $t(ij)$: time consumed from station i to j | $i,j \in V$
- $c(i)$: # of students boarding on station i
- Q: bus capacity
- T_{max}: time limit
- M: big enough constant

Variable

- $x(ij): \begin{cases} 1, & \text{route from station i to station j is selected} \\ 0, & \text{otherwise} \end{cases}$

- y(i): total number of student on bus after visiting station i

- z(i): total time consumed after visiting station i

Constraint

1) $\sum_{i \in V} x(ij) = 1, \ \forall \ j \in V \mid j \neq 0$ - exactly one bus for each station (except school)
2) $\sum_{j \in V} x(ij) = 1, \ \forall \ i \in V \mid i \neq 0$ - for every station i , the bus will go to a next station j
3) $y(i)+c(j)-y(j) \leq M(1-x(ij)), \ \forall \ i,j \in V \mid j \neq 0$ - capacity relationship between stations
4) $z(i)+t(ij)-z(j) \leq M(1-x(ij)), \ \forall \ i,j \in V \mid j \neq 0$ - time relationship between stations
5) $y(i) \leq Q$ - capacity constraint
6) $z(i) \leq T_{max}$ - time constraint

Objective Function

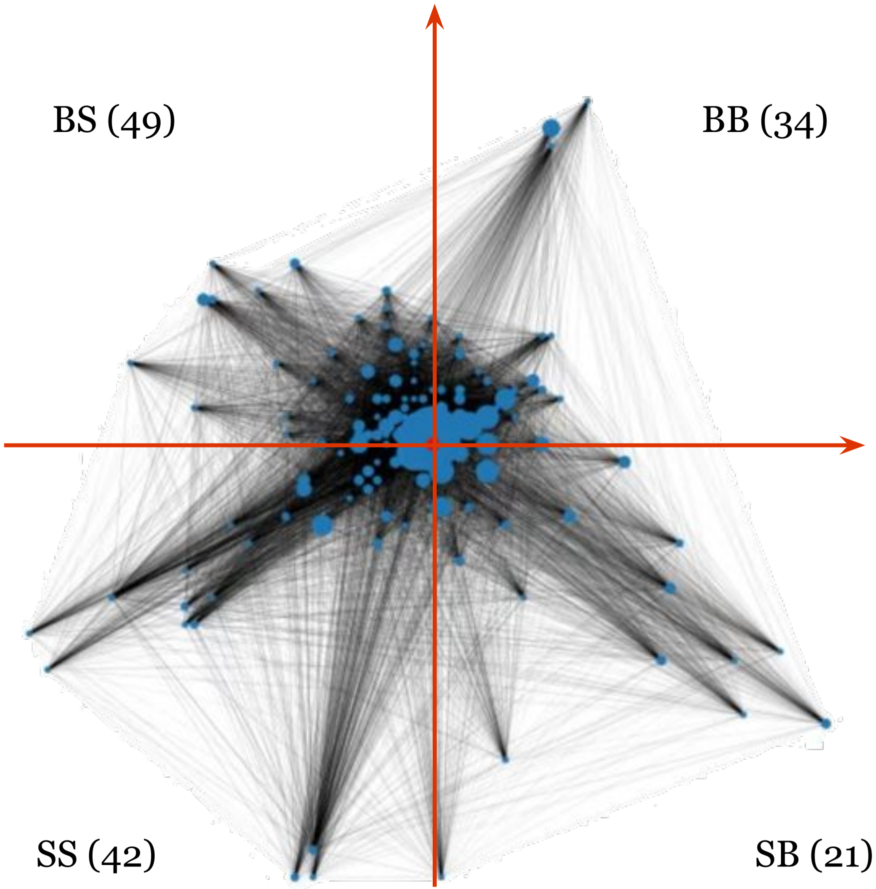
- Min $M \sum_{j \in V, j \neq o} x(oj) + \sum_{i \in V} \sum_{j \in V, j \neq o} t(ij)\, x(ij)$

Minimize:

1. Number of routes
2. Distance of each route

4. Data Processing

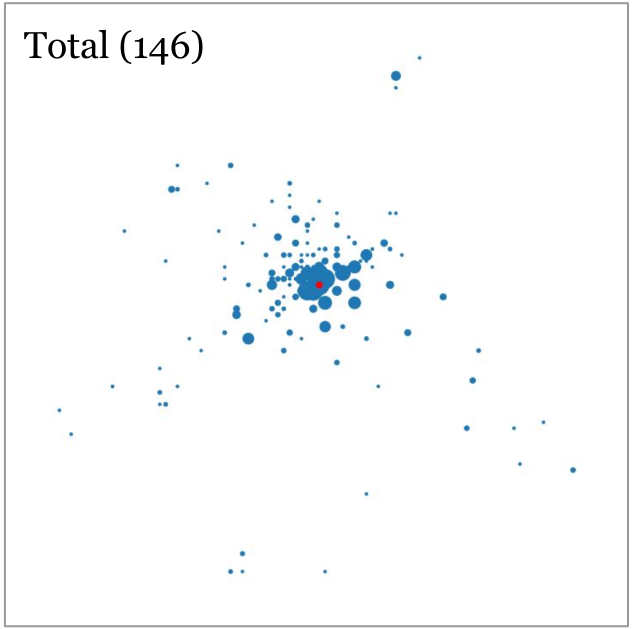
Communities

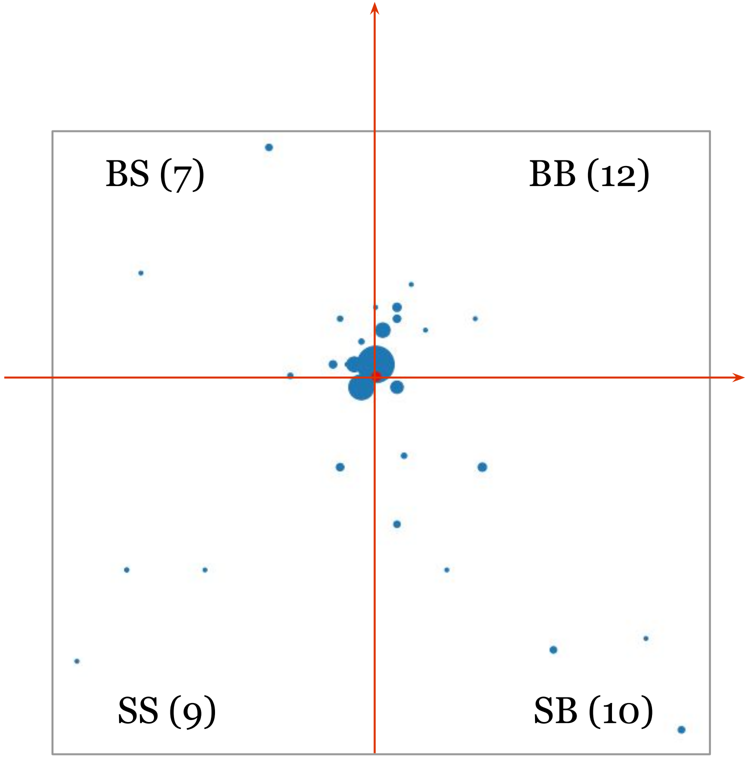


Blue nodes: stations

Red node: School

Small Community



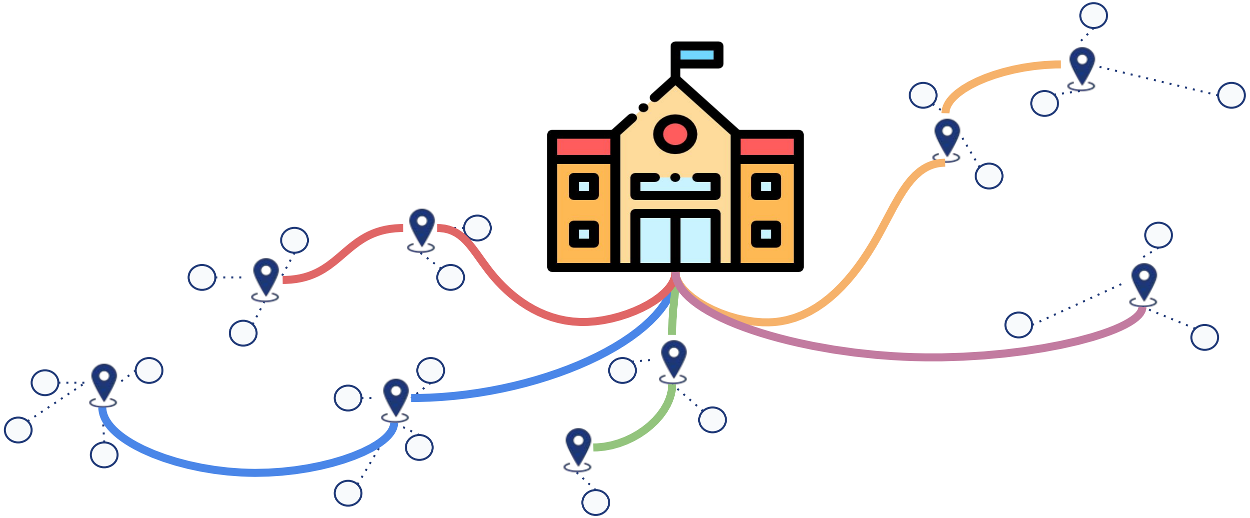


Total (38)

- Blue nodes: stations
- Red node: School

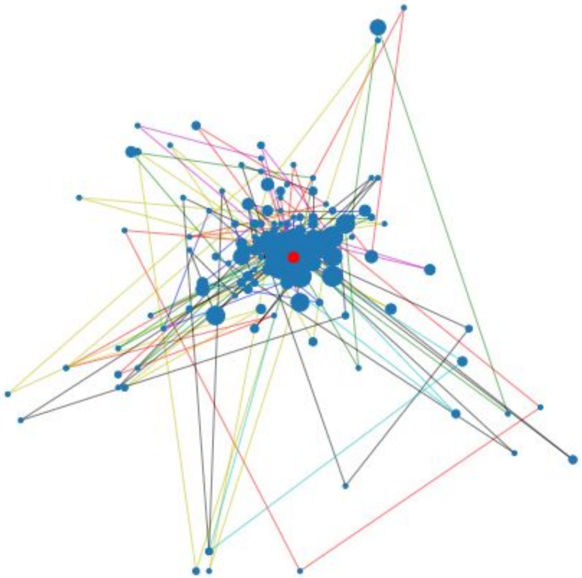
Parameter Settings

- Q= 50 , capacity :50 student/bus
- D = 50, Tmax
- M= 1000

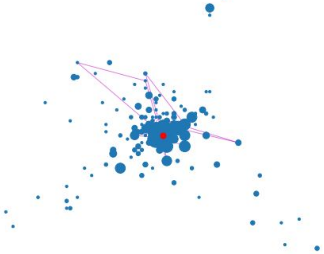


5. Results/Analysis

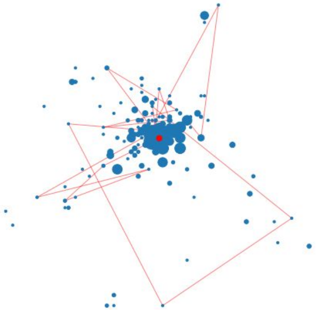
Tentative Result



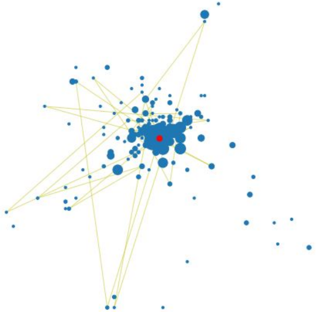
seven routes



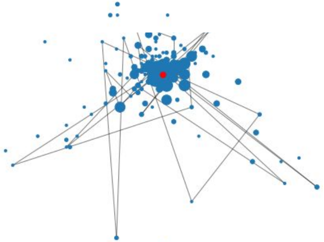
D = 131



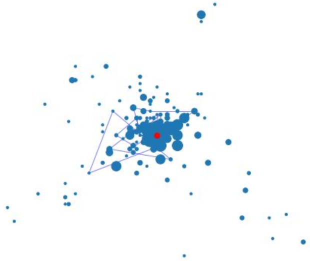
D = 330



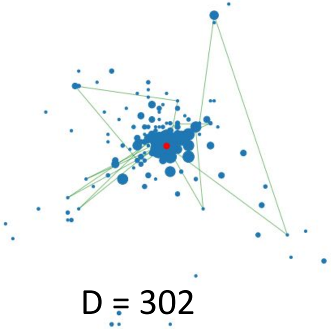
D = 558



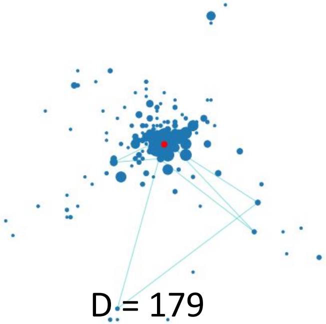
D = 544



D = 122

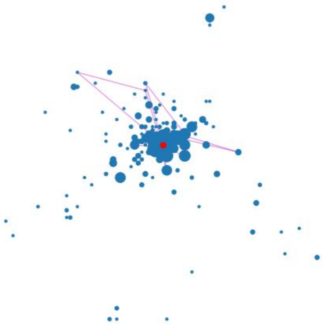


D = 302

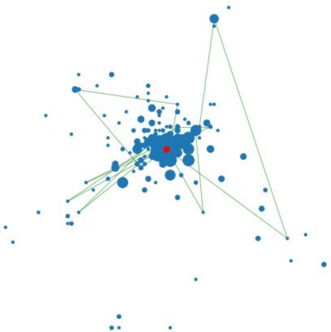


D = 179

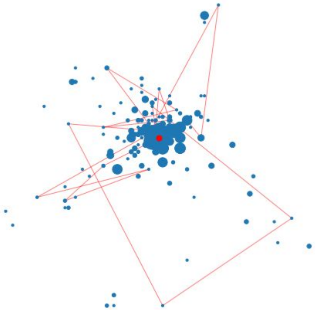
Analysis



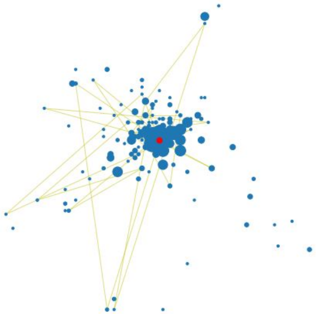
D = 131



D = 302



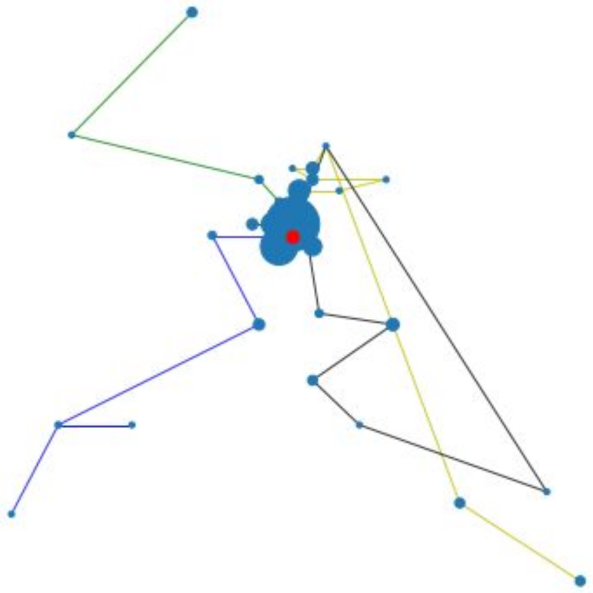
D = 330



D = 558

As we can see from the picture, the school bus with the responsibility of picking more students will have a shorter running path.

Small Community Result



There are eight routes in total.

route 1 : 1.2
route 2 : 52.7
route 3 : 62.0
route 4 : 39.0
route 5 : 82.3
route 6 : 0.7
route 7 : 0.7
route 8 : 0.7 (distance)

The stop that contains 128 are separated into four stations to make the capacity feasible.

6. Problems and Solutions

- **The data set is too large (2^{146}), which is a big workload of CPU**

    ① Divide the data set into cluster according to their latitude and longitude.

    ② Flexibly adjust the dataset number we use to test the upper-bound of CPU

- **Locate the start station of our cluster**

    ① Find the farthest station from school in this cluster and treat it as a starting point.

- **Two objective need to be optimized**

    ① Set a big enough number M to give priority for those two objective function.

    ② Use Guroubi rather than Networkx to satisfy the constraint.

7. Improvement

In our project:

Minimize

- Number of routes
- Distance of routes

Method

- Integer Programming (Gurobi)

① Global solution for small community

② Local solution for whole route planning

Improvement:

Minimize

- Number of buses in fixed routes
- Distance of each route

Method

① Integer Programming

② Heuristic Method (ACA, ES, ANN) for a whole route planning

8. Conclusion

Through this project, we found that school bus planning path is a very complex problem which is known as **SBRP problem** and is well studied during recent years. In addition to find the shortest path from each starting point to reduce costs, we also need to ensure that each station is connected, and these two problems restricts each other. Therefore, we continuously reduce the data size and appropriately simplify the model. Even so, we still handle millions of levels of data.

From the initial consideration of using Networkx to find the shortest path, to the later use of Gurobi for linear programming, the proposal and application of each equation is the consolidation of the knowledge points in this lesson. Professor Chrys gave us a lot of help and thank you so much!

The End