Open House Route Planner

Alexander Jansing

SUNY Polytechnic Institute
Department of Computer Science

24 April 2019

Inspiration

I have been looking for houses. When I add open houses to my Google Calendar, I am able to request direction to whatever house is open next in time, but I was thinking, *"What if two houses are significantly far apart, open at similar times, and there are other houses in each of their respective neighborhoods that open at different times? Is there a way I can plan my day of house hunting so that I can attend all of the open houses?"*

Inspiration

I have been looking for houses. When I add open houses to my Google Calendar, I am able to request direction to whatever house is open next in time, but I was thinking, *"What if two houses are significantly far apart, open at similar times, and there are other houses in each of their respective neighborhoods that open at different times? Is there a way I can plan my day of house hunting so that I can attend all of the open houses?"*
The answer to this question is, "yes, within reason."

Objective

Given a series of open houses the application should find routes that will allow the user to visit the maximum number of open houses given the constraints of *travel time* and *when the open houses are open*.
After phrasing stating the problem, the problem was divided up into several part:

- where the houses were with respect to each other,
- when the open houses were,
- and try to determine the path I needed to take to visit as many open houses as possible.

I will describe how each of these tasks were accomplished and what other work needed to be done to facilitate that work.

Definitions

- *Safely Query* - querying for existence in MongoDB before querying ArcGIS API for information. This applies to both *geocoding* an address its coordinates on the planet and to gather the directions from one point to another.

Definitions

- *Safely Query* - querying for existence in MongoDB before querying ArcGIS API for information. This applies to both *geocoding* an address its coordinates on the planet and to gather the directions from one point to another.
- geocoding - the process or converting addresses to coordinates on the globe.

Definitions

- *directions matrix* - an array of JSONs containing vertex and edge information. While not strictly a matrix, it supplies the information to populate a matrix of travel times. While a matrix is no longer used, it is a good mental image to have when thinking about the OpenHouseGraph.

```
[{'_id': ObjectId('5cad42f3671c850b358ab86b'), 'url': REDACTED, 'dtstart': '20190414T153000Z',
        ↪ 'dtend': '20190414T170000Z', 'summary': REDACTED, 'description': REDACTED, 'location':
        ↪ {'geometry': {'x': x_0, 'y': y_0, 'spatialReference': {'wkid': 4326, 'latestWkid':
        ↪ 4326}}, 'attributes': {'Loc_name': 'World', 'Status': 'M', 'Score': 100, ... 'X': x_0,
        ↪ 'Y': y_0, 'DisplayX': x_{d0}, 'DisplayY': y_{d0}, 'Xmin': x_{min0}, 'Xmax': x_{max0},
        ↪ 'Ymin': y_{min0}, 'Ymax': y_{max0}, 'ExInfo': '', 'OBJECTID': 1}, 'address': REDACTED},
        ↪ 'address_hash': sha1(location0), 'durations': [[1, 13.85]]}
{'_id': ObjectId('5cac003a671c85002d41afb9'), 'url': REDACTED, 'dtstart': '20190413T150000Z', 'dtend':
      ↪ '20190413T170000Z', 'summary': REDACTED 'description': 'REDACTED, 'location': {'geometry':
      ↪ {'x': x_1, 'y': y_1, 'spatialReference': {'wkid': 4326, 'latestWkid': 4326}}, 'attributes':
      ↪ {'Loc_name': 'World', 'Status': 'M', 'Score': 100, ... 'X': x_1, 'Y': y_1, 'DisplayX':
      ↪ x_{d1}, 'DisplayY': y_{d1}, 'Xmin': x_{min1}, 'Xmax': x_{max1}, 'Ymin': y_{min1}, 'Ymax':
      ↪ y_{max1}, 'ExInfo': '', 'OBJECTID': 1}, 'address': REDACTED},, 'address_hash':
      ↪ sha1(location1), 'durations': [[0, 14.15]]}]
```

Figure 1: Simplified Directions Array (Matrix).

Definitions

- *directions matrix* - an array of JSONs containing vertex and edge information. While not strictly a matrix, it supplies the information to populate a matrix of travel times. While a matrix is no longer used, it is a good mental image to have when thinking about the OpenHouseGraph.
  - Figure 2 is an implied matrix of travel times between locations from Figure 1. $t_{i,j}$ is the travel time between house i and house j. The travel time from house i to house i is given the value -1 as a guard against reflexive traveling.

$$\begin{pmatrix} -1 & t_{0,1} & \cdots & t_{0,n-1} \\ t_{1,0} & -1 & \cdots & t_{1,n-1} \\ \vdots & \ddots & \ddots & \vdots \\ t_{n-1,0} & \cdots & \cdots & -1 \end{pmatrix}$$

Figure 2: Matrix of travel times between locations.

Definitions

- Esri Flask App - REST endpoint that is designed to accept information from MongoOps and query the ArcGIS Developer API for geocoded address information or directions between two geometry points.

Definitions

- Esri Flask App - REST endpoint that is designed to accept information from MongoOps and query the ArcGIS Developer API for geocoded address information or directions between two geometry points.
- ICSParser - Class for parsing ICS files to JSON to be geocoded.

Definitions

- Esri Flask App - REST endpoint that is designed to accept information from MongoOps and query the ArcGIS Developer API for geocoded address information or directions between two geometry points.
- ICSParser - Class for parsing ICS files to JSON to be geocoded.
- MongoOps - Class for:
  - loading data to database,
  - querying the database for geocoded address,
  - querying the database for directions,
  - querying the ArcGIS Developer API for geocoded address,
  - and querying the ArcGIS Developer API for directions.

Definitions

- Esri Flask App - REST endpoint that is designed to accept information from MongoOps and query the ArcGIS Developer API for geocoded address information or directions between two geometry points.
- ICSParser - Class for parsing ICS files to JSON to be geocoded.
- MongoOps - Class for:
  - loading data to database,
  - querying the database for geocoded address,
  - querying the database for directions,
  - querying the ArcGIS Developer API for geocoded address,
  - and querying the ArcGIS Developer API for directions.
- OpenHouseGraph - A graph data structure used for computing routes one might take while visiting open houses.
  - Inspired by: Data Scientists, The one Graph Algorithm you need to know [1] - Basis for the OpenHouseGraph class.

Walkthrough of Algorithm

Acknowledgements

This work is dedicated to my parents. Without their guidance, I would not be the person I am today. I joined the United States Air Force after their suggestion, where I gained a greater appreciation for my education and the opportunities afforded to me.
I would like to thank each of my committee members:
Gerard Aiken supplied the Esri ArcGIS Developer credits required to do this work.
I would also like to acknowledge Jennifer Tran, Sylvia Pericles, Zhushun Cai, and Oliver Medonza for aiding the initial code base where we won the Esri API Prize and the Grand Prize at Hack Upstate XI.
This work was funded by Booz Allen Hamilton tuition assistance program.


R. Agarwal.
To all data scientists - the one graph algorithm you need to know.
https://towardsdatascience.com/to-all-data-scientists-the-one-graph-algorithm-you-need-to-know-59
2019.