Urban Oasis: Utilizing Network Analysis Techniques to Reclaim the Streets of San Francisco

Luda Zhao, Elena Frey, Karen Wang

Stanford University

January 9, 2016

1 Introduction

Half of the world's population currently lives in urban areas, and by 2050 this number is expected to grow to eighty percent.[3] Yet as our cities grow larger and more complex, so do the associated problems such as pollution, traffic, and dissolution of community. It is time that we rethink city planning in order to accommodate human behavior on a more individual scale.

Cities such as Bogotá, Copenhagen, and New York City have begun to take steps in this direction via projects to repurpose streets as bike and pedestrian-only thoroughfares.[1, 5] This reduces the danger for bikers and pedestrians of colliding with vehicles, and leads to less automotive traffic as people are encouraged to walk or bike instead. So far, these projects have only been carried out on a small-scale as test runs. What if we could make it possible to carry this out on a larger scale in any city, by analyzing the city's existing street networks and developing a strategy for street removal based on network properties?

Specifically, this paper examines the street network of San Francisco, which historically has been plagued by traffic congestion, and uses network analysis tools to figure out which blocks can be repurposed as bike and pedestrian pathways while minimizing the effect on overall transportation efficiency. We accomplished this in three stages. First, we identified potential candidates for removal using a variety of heuristics and network analysis algorithms. Second, we simulated a traffic model of San Francisco that captures the traffic patterns within the city. Third, we ran the same traffic model on the repurposed network (after having removed the candidate roads) and measured the impact removing these roads had on traffic.

2 Prior Work

2.1 On road network modeling and analysis

In 2003, Jiang and Claramunt analyzed the urban road networks of Gävle, Munich, and San Francisco from a topological perspective to determine whether the road network exhibited small-world and scale-free characteristics.[7] In the topological model the authors used, rather than modeling the street network geographically, in which the intersections are the vertices and the roads between them are the edges, they used a named-streets-oriented view in which each "named street" is a vertex and the edges are the intersections. This model was explored further in a 2005 paper by Porta et. al. where they dubbed it the "dual" representation of a graph (versus the more intuitive "primal" representation).[12] The dual model preserves the continuity of streets over a plurality of edges and incorporates the idea that people preferentially go straight on a street. Furthermore, the dual model captures more of the specific structures of the real-world network it represents—namely, that streets are continuous entities that do not terminate at a vertex or intersection.

2.2 On modeling traffic and predicting road closure effects

In 2007, Moses and Mtoi identified and evaluated a series of link congestion functions, known as BRP functions, as a parametric model for predicting travel times on arterial links in road networks.[11] These functions have been widely used in transportation planning applications in which the travel time is assumed to have a nonlinear relationship with the volume/capacity ratio.

In 2008, Jenelius proposed a formula for measuring the increase in vehicle travel time during road closures, taking into account alternative routes that a commuter could take.[6] The author also proposed additional heuristics for analyzing the structure of a road network: alpha and beta indices for a region (α_r and β_r) to measure the robustness of a network as defined by link redundancy:

$$\alpha_r = (M_r - N_r + 1)/(2N_r - 5)$$

$$\beta_r = M_r/N_r$$

where M_r and N_r are the number of undirected edges and nodes in region r, respectively. α_r ranges from 0 (trees) to 1 (triangular grids), while β_r tends to be 1 for large trees, 2 for square grids, and 3 for triangular grids. They also proposed the road density of a region R_r as an indicator of the availability of alternative routes in a network, where R_r is the total length of the regional road network divided by the area of the region.

For our research, we converted our data into the dual representation to see what new insights we could gain, and based our traffic modeling algorithm off of the BRP functions.

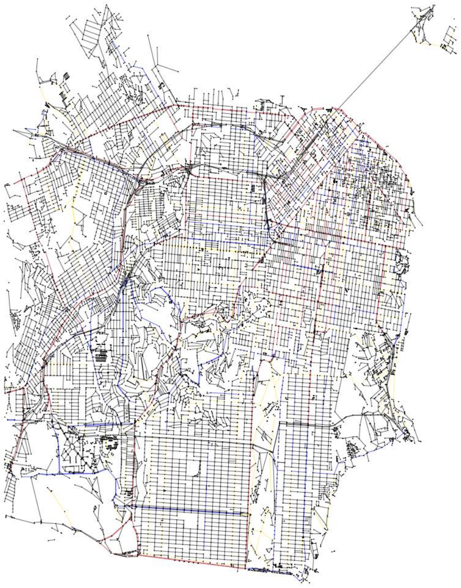


Figure 1: San Francisco Road Network (Primal Graph)

3 Data Collection and Processing

We initially wanted to combine the University of Utah's California road networks dataset with Caltrans's traffic volume data, but we ran into difficulties extracting and merging the two formats.[13, 2] However, upon further research we found OpenStreetMap (OSM) to be a sufficiently well-formatted and comprehensive dataset for our purposes.[9] OSM encodes streets as "ways" representing a collection of nodes. Each node has an ID, longitude/latitude coordinates, and tags encoding information on points of interest. Moreover, ways are also tagged with information such as how many lanes they have and whether they are paved or not, which will be used in our traffic models.

We used the Overpass API to extract OSM XML-formatted data for the city of San Francisco. As we wanted to model streets in both the primal and the dual forms, we then used different extraction algorithms to obtain our datasets:

Primal: OSM nodes are converted into graph nodes. OSM ways are broken down into a series of edges that connect sequentially from node to node. This model accurately captures the geographical properties of the network, but loses information about roads as continuous entities through the city.

Dual: OSM ways are converted into graph nodes. OSM nodes are converted into edges based on the ways that they intersect with at each node. This model forgoes geographic accuracy in exchange for insightful structural information. During our subsequent investigations, this form proved to be more useful in identifying our candidate roads.

Table 1: Summary statistics of network models

Measure	**Primal**	**Dual**
Node count	15,107	10,211
Edge count	22,125	24,274
Average degree	1.50	2.31

We faced various challenges in processing the data. As the OSM data relies heavily on crowd-sourcing, it contained inconsistent entries which we had to manually clean. We also ran into integer overflow issues with node and edge IDs, which required a re-mapping of node and edge IDs for both graphs.

4 Identifying Candidate Roads

4.1 Heuristics

After processing the data, our first task was to identify potential candidate roads to remove from the network in order to repurpose them as cyclist and pedestrian-only roads. We decided that, for an initial heuristic, we should look at the nodes in the dual representation of the graph (in which the nodes represent the edges) that have low betweenness centrality but high closeness centrality. From looking at these measures, we can get road segments that are central to the network in the sense that they have a low total distance to all other roads while not frequently acting as a bridge along the shortest paths between other nodes in the network. (To clarify, we aren't removing entire roads, but rather an entire "way" as stored in the data, which could be anywhere from one to a couple of blocks.) These are necessary conditions to consider in order to get candidates that will make viable cyclist and pedestrian-only roads.

After extracting the dual representation of the graph data into a snap.py representation, we computed the betweenness $C_B(i)$ and closeness centrality $C_C(i)$ of every node as follows:

$$C_B(i) = \frac{\sum_{j<k} \dfrac{g_{jk}(i)}{g_{jk}}}{(n-1)(n-2)/2}$$

where g_{jk} represents the number of shortest paths between nodes j and k and $g_{jk}(i)$ is the number of these shortest paths that pass through node i.

$$C_C(i) = \frac{1}{(n-1)\sum_{j=1}^{n} d(i,j)}$$

After calculating these metrics on every node, we found that there were about 1,600 nodes with a betweenness centrality of 0. We also found the 100 nodes with the highest closeness centrality.

After cross-referencing the list of nodes with betweenness centrality 0 and the nodes with the highest closeness, we found that there were 46 roads satisfying both conditions. Interestingly, upon examining the OSM data for each of these roads, we found that all of their types were listed as either paths, steps, or some other pedestrian or cyclist-only road. Not only that, but some of the nodes in the list of results were listed as proposed cycleways already. This indicated that we were on the right track in identifying potential candidates for removal, but ultimately we were not finding appropriate roads to help us achieve our goal.

We then decided to filter out all of the roads that had a betweenness centrality of 0, since this indicates that the node is not along any shortest paths in the network and is therefore not significant enough to repurpose. Looking at the next 1,000 nodes with lowest betweenness centrality, the intersection of this set with the set of those with the 100 highest closeness centralities yielded one result, a golf course path.

Since it seemed that we were not going to identify any significant roads in the network with this approach, we decided to filter the roads based on the "highway" tag from the OSM data. This tag indicates the importance of each road in the network and ranges from "motorway" and "trunk" as the most significant to "primary," "secondary," "tertiary," "residential," and "service" being the least significant. Now looking at only primary, secondary, and tertiary roads as possible candidates for repurposing, we again took the intersection of the nodes with lowest betweenness (again including those with 0 betweenness since we only looked at significant roads) and the nodes with highest closeness. This resulted in 13 candidates, some of which are displayed below.

Table 2: Candidate road segments for removal

Node ID	**Road Name**
7861	Dolores St.
9458	Golden Gate Ave.
7700	24th St.
8779	Valencia St.
6295	Fulton St.

4.2 Analysis

While the dual graph revealed interesting properties about streets themselves that we used to select potential candidates, the primal graph encodes real-world properties about the network such as connectivity and shortest path length that can allow us to evaluate our policy. Therefore, after retrieving the set of 13 candidate streets from the dual graph analysis, we removed the corresponding road segments from the primal graph in order to evaluate the impact on San Francisco's physical street network. We also removed 13 randomly selected roads and performed the same analysis.

The baseline heuristics we measured on the primal graph before and after street removal were approximate diameter, approximate average shortest path length, and number of weakly-connected components (WCCs).

Our results were as follows:

Table 3: Street removal analysis results

Heuristic	Before	After candidate removal	After random removal
Diameter (approx.)	133	133	134
Avg. shortest path (approx.)	44.55	44.58	44.71
Number of WCCs	10	10	10

We did not see a significant change in the diameter of the primal graph before and after street removal. We interpreted this as a positive sign, as no shortest path between nodes was drastically lengthened due to the removal of our candidate streets.

We also wanted to get an idea of the impact on average shortest path length of the graph. However, due to the high time complexity of computing BFS for a graph of our size, we instead selected 100 nodes at random and computed the average shortest path between them, repeated the process 10 times, and took the average of the result. Again, there was not a significant increase in shortest path length for any of these 100 node samples after removing our candidate roads. However, there was a slightly larger increase after removing the random roads, which suggests that our heuristics are accurately selecting streets that will have less of a disruptive impact on the network.

Finally, we computed the number of WCCs before and after street removal, and found that the number remained constant. We interpreted this as a positive indicator that we didn't accidentally fragment the graph into more disconnected components with our street reclamation policy.

5 Traffic Modeling and Evaluation

5.1 Background & Algorithm

In order to more quantitatively measure the impact of repurposing roads in San Francisco, we needed more than a static analysis of the graph network. Instead, we needed to create an accurate model of traffic flow in San Francisco in order to capture the dynamic impact repurposing roads would have on travel in the city. To create this model, we generated k random start-destination pairs distributed across the city in our primal graph (since this representation provides a more accurate geospatial view of the network).[14] We then applied our implementation of Dijkstra's algorithm between the pairs, using the estimated travel time of each edge as the weights to find the optimal path between each pair of nodes.

The main challenge of traffic modeling based on travel time is that travel time is a function of not only the road distance and the road typology, but also the volume of current traffic flow along the road. Thus, in order to calculate traffic time accurately, when developing our algorithm we decided to use a series of link congestion functions in an iterative process to converge upon the optimal model.

For free-flowing highways without signals, we used a version of the Highway Flow function proposed by Skabardonis and Dowling in 1997[15]:

$$T(q) = T_o \left(1 + \alpha \left(\frac{v}{Lc}\right)^{\beta}\right) \tag{1}$$

Here, T_o is the free-flowing travel time (edge distance divided by speed limit), v is the volume (number of cars currently using the link), L is the segment length, and c is the capacity (estimated in our model by the number of lanes in the road). α and β are congestion parameters specific to the area, and we used the values $\alpha = 0.2$ and $\beta = 3$ from Moses and Mtoi, adjusted based on a sampling of paths cross-referenced by Google Maps traffic time estimates. (Note that these are different from the network link-redundancy heuristics α_r and β_r mentioned earlier in the prior work section.)

For roads with signals and other stoppages, we will use a modified version of the function above to account for the extra delay incurred by the signals:

$$T(q) = T_o' \left(1 + \alpha \left(\frac{v}{Lc}\right)^{\beta}\right)$$

in which T_o' is the modified free-flow travel time affected by traffic intersections.

To derive T_o', we used the work of Moses and Mtoi that related free-flowing vehicular speed and traffic-modified vehicular speed:

$$U_o' = \frac{L}{\dfrac{L}{U_o} + \dfrac{dN}{60}}$$

where U_o is the free-flow speed, U_o' is the modified speed, L is the edge length, d is the average in delay between all intersections (in seconds, and regardless of whether a car stopped at a red light), and N is the number of traffic lights (assumed to be proportional to the number of intersections). Since we know that $U_o = \dfrac{L}{T_o}$ and $U_o' = \dfrac{L}{T_o'}$, we can substitute to obtain an expression for T_o' in terms of T_o:

$$\frac{L}{T_o'} = \frac{L}{\dfrac{L}{L/T_o} + \dfrac{dN}{60}}$$

Solving for T_o', we have:

$$T_o' L = L \left(\frac{L}{L/T_o} + \frac{dN}{60}\right)$$

$$T_o' = T_o + \frac{dN}{60}$$

Thus, we have:

$$T(q) = \left(T_o + \frac{dN}{60}\right)\left(1 + \alpha \left(\frac{v}{Lc}\right)^{\beta}\right) \tag{2}$$

We used $d = 18.10$ s, from real-time delay measurement work done in Wolshon and Taylors.[16] As before, we used the parameters fitted from real traffic datasets of $\alpha = 0.2$ and $\beta = 3$.

5.2 Implementation

We originally wanted to generate a number of pairs equal to the population of San Francisco. However, finding an optimal route for each pair turned out to be computationally prohibitive. Instead, we grouped our pairs into $k = 1000$ batches to speed up calculations.

The pseudocode of our algorithm to generate the model is as follows:

```
Initialize graph
Initialize edge time_weight, edge vehicle_volume to 0
For each iteration i:
  For each pair p in randomly_generated_pairs:
    shortest_path = dijkstras(p, graph, time_weights)
    For each edge e in shortest_path:
      if e is highway:
        updateHighwayTimeWeights(e)
      else:
        updateLocalWayTimeWeights(e)
      updateVehicleVolume(e)
  calculatePercentageChange(prevPaths, currPaths)
  if percentageChange < convergence threshold:
    break
return graph, edge time_weights
```

Note that the edge weights and the edge volumes of the graph are updated “stochastically” after **each path reassignment**, not after each iteration through the generated pairs. We found that if we updated the edge weights after each iteration and completed route reassignment in batches, the paths would not converge—instead, the paths would bounce back and forth between a few preferred paths. Thus, we took this incremental update approach to ensure our model would converge to a stable state.

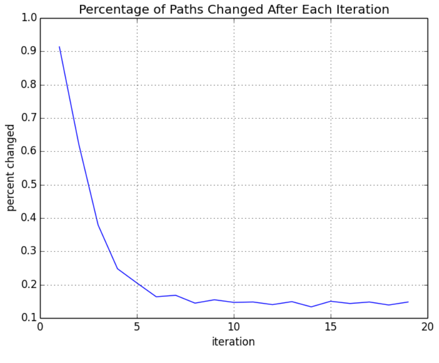
As we wanted to capture as close to real-life traffic conditions as possible, we took specific care in making sure our parameters in equations 1 and 2 were consistent. These parameters include:

– Edge distance: To compute the distance between two longitude and latitude coordinates, we used a python library called geopy that provides an implementation of the Vincenty distance calculation. These formulae are more accurate than the great-circle distance calculation as they assume the Earth is an oblate spheroid rather than a perfect sphere.

– One way streets: In our original calculations we assumed that all roads could be traveled along bidirectionally. However, after comparing a few of our shortest paths with the routes suggested by Google and observing the prevalence of one way roads in San Francisco, we concluded that it is necessary to take this into account to get the most accurate results. We were able to pull this data from the OSM tags for each road.

– Road width and speed: Where available, we were able to use the exact number of lanes and maximum speed for each road as provided in the OSM dataset. However, since some of this information is missing from the data, we used the road type (“motorway”, “primary”, “secondary”, etc.) to fit approximate values for the speed and number of lanes so that we still captured an appropriate representation of the road.

We used the percentage of routes that changed between each iteration as the metric to determine when the model had converged. During our testing, the model was able to converge to less than 15% of routes changing after 20 iterations. The graph to the right shows the convergence of the model:



To evaluate if our model accurately captured the real-world mechanics of traffic flow, we calculated some metrics based on our randomly generated start and end destinations for each of our graphs. These statistics seem to be realistic for randomly-generated routes in San Francisco.

Table 4: Traffic metrics for model

Route Metric	Value
Avg. Distance (miles)	5.47
Avg. Distance Std. Dev.	3.21
Max Distance	16.16
Min Distance	0.15
Avg. Time (minutes)	22.51
Avg. Time Std. Dev.	9.11
Max Time	47.52
Min Time	1.43

We also hand-sampled a few of our paths and calculated both the optimal route and the estimated travel time using our converged traffic model. We then compared this with travel times using Google Maps. While the model is not as accurate for longer paths, it does very well for most of the paths we checked. An example path from *2080 Market St* to *1071 Harrison St* is shown below:

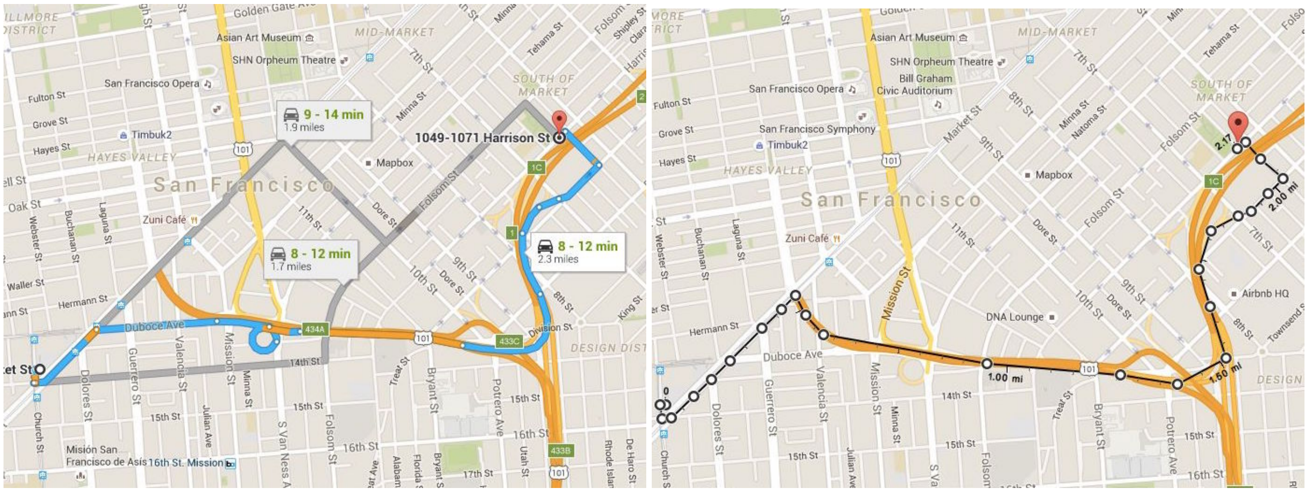


Figure 2: The route and estimated travel time from Google Maps is shown on the left. On the right is the optimal route chosen by our traffic model. The model estimated the time of this route to be 11.9 minutes, which falls inside the range estimated by Google Maps.

5.3 Evaluation of Repurposed Road Candidates

In order to quantify the impact of road repurposing on traffic, we ran our traffic model on an altered graph with the candidate roads removed. We then measured the percentage travel time increase:

$$\Delta t_{avg} = \frac{\sum_{i=1}^{N} t_i' - \sum_{i=1}^{N} t_i}{\sum_{i=1}^{N} t_i}$$

where t_i is the travel time along route i for the initial graph, and t_i' is the time along route i for the modified graph. We decided to weight each route equally in our calculation under the assumption that every route has equal importance regardless of distance.

Similarly, we calculated the percentage travel distance increase:

$$\Delta d_{avg} = \frac{\sum_{i=1}^{N} d_i' - \sum_{i=1}^{N} d_i}{\sum_{i=1}^{N} d_i}$$

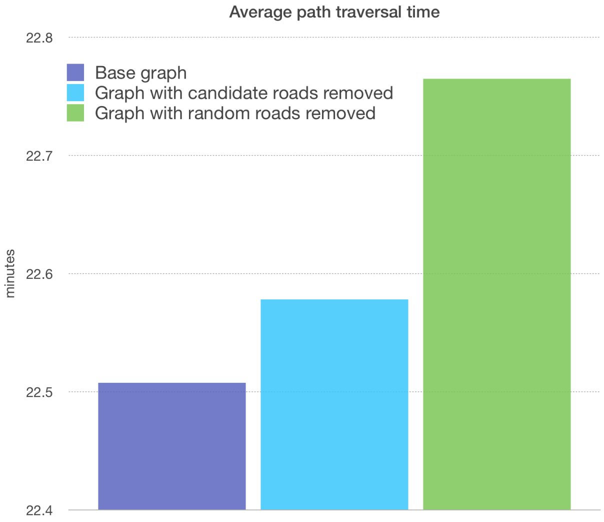
We also ran our traffic model on the graph with the same number of **random** edges removed. This comparison measured the effectiveness of our selection criteria by comparing against a random baseline. To reduce noise between our comparisons, we used the same set of start and end destinations for all three runs of the model, and we used the same metrics to measure performance.

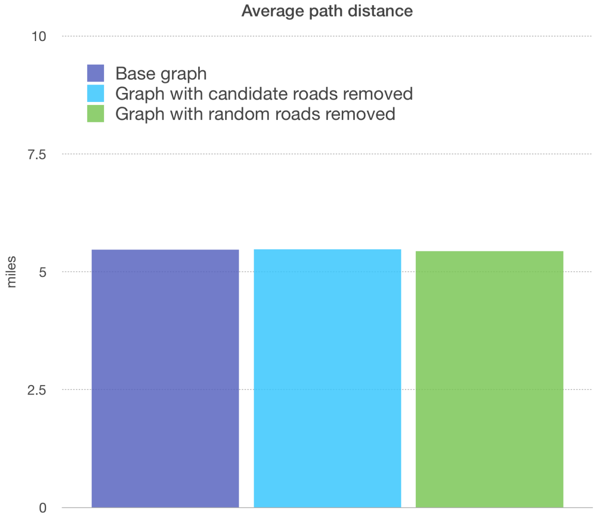
Here are our results:

Table 5: Comparison of traffic metrics for altered graphs

Route Metric	Base graph	With candidate roads removed	With random roads removed
Avg. Distance (miles)	**5.47**	**5.48**	**5.44**
Avg. Distance Std. Dev.	3.21	3.21	3.15
Max Distance	16.16	16.16	16.16
Min Distance	0.15	0.15	0.15
Avg. Time (minutes)	**22.51**	**22.58**	**22.76**
Avg. Time Std. Dev.	9.11	9.14	9.26
Max Time	47.52	47.66	48.04
Min Time	1.43	1.43	1.43

The following graphs show the average path traversal time and average path length for each of the three graphs after the traffic model converged.





From these results, we can see that the average travel time increased by 0.3% from the base graph after removing our proposed nodes. However, there was a 1.1% increase in travel time when removing a random set of nodes. This promising result shows that our heuristic for selecting candidate roads outperformed random selections. Additionally, we can see that the average path distance was virtually unchanged between each version of the graph. As our path-finding algorithm used travel time and not distance as weights, it was interesting to observe that the average distance of routes did not increase as we removed roads from the graph.

6 Conclusion

From our results, we can conclude that central roads in San Francisco can be repurposed while having negligible impact on traffic congestion, at least based on the traffic model we built in this study. Our results suggest that a simple heuristics algorithm can be used to identify candidate road segments that have a demonstrably smaller impact on traffic when removed than if random roads are selected. Although further analysis is needed to provide a complete picture of how such street repurposement would affect city life overall, our result serves as compelling evidence that such programs are logistically sound in regards to traffic.

Future directions to investigate include experimenting with additional heuristics for identifying candidate roads, as well as improving on our traffic model. Alternative non-network-based heuristics we could have looked at for our road reclamation policy include taking into account real-world traffic data, as measured by traffic volumes (data on average annual daily traffic for California state highways is available online [2]). Moreover, we have considered a number of modifications to tune our traffic model in order to capture more real-world phenomena. One, a penalty could be introduced to the pathfinding algorithm based on the number of roads traversed (as an indicator of how many turns were made) to capture the longer amount of time required to make a turn rather than to continue straight. Two, additional geospatial orientation parameters could be incorporated to penalize left turns more than right turns (since drivers can turn right on red lights but not left, these take more time). It would be quite feasible to compute these parameters based on the changes in longitude and latitude coordinates at each turn. Finally, this study opens up possibilities of more interdisciplinary applications, as it is now apparent that network analysis techniques in computer science can be used to effectively address questions related to city planning.

Reclaimed road segments provide public areas for the community and safe thoroughfares to increase safety and accessibility for bikers and pedestrians. They act as plazas where people can eat, rest, and socialize, and they increases spatial efficiency, creating a more livable, communal environment in modern urban areas such as San Francisco. By applying network analysis techniques to model this program, we were able to obtain concrete evidence in support of these programs. We hope that this and future work will provide guidance for street reclamation programs and help bring them more into the public realm.

References

[1] Brown, L. (2010, May 21). Reclaiming the streets. Retrieved November 17, 2015, from http://grist.org/article/reclaiming-the-streets/

[2] Caltrans GIS Data. (n.d.). Retrieved November 17, 2015, from http://www.dot.ca.gov/hq/tsip/gis/datalibrary/

[3] Dalsgaard, A. (Director). (2014). *The Human Scale* [Motion picture]. NFP.

[4] Davis, G. A., & Xiong, H. (2007). Access to Destinations: Travel Time Estimation on Arterials.

[5] Goodyear, S. (2012, April 25). Why the Streets of Copenhagen and Amsterdam Look So Different From Ours. Retrieved November 17, 2015, from http://www.citylab.com/commute/2012/04/why-streets-copenhagen-and-amsterdam-look-so-different-ours/1849/

[6] Jenelius, E. (2008). Network structure and travel patterns: explaining the geographical disparities of road network vulnerability.

[7] Jiang, B., & Claramunt, C. (2003). Topological analysis of urban street networks. *Environment and Planning B*, 151-162.

[8] Li, F., Cheng, D., Hadjieleftheriou, M., Kollios, G., & Teng, S. H. (2005). On trip planning queries in spatial databases. In *Advances in Spatial and Temporal Databases* (pp. 273-290). Springer Berlin Heidelberg.

[9] Main Page. (2015, November 17). In *OpenStreetMap.org*. Retrieved 04:58, November 17, 2015, from https://wiki.openstreetmap.org/wiki/

[10] Matson, J. (2010, June 15). Auto Immune: Cities Convert Streets into Pedestrian, Cyclist and Mass Transit Thoroughfares. Retrieved November 17, 2015, from http://www.scientificamerican.com/article/car-free-streets/

[11] Moses, R. & Mtoi, E. (2013). Development of Street Models for Improving Travel Forecasting and Highway Performance Evaluation

[12] Porta, S., Crucitti, P., & Latora, V. (2005). The network analysis of urban streets: a dual approach. *Physica A: Statistical Mechanics and its Applications*, 853-866.

[13] Real Datasets for Spatial Databases: Road Networks and Category Points. (n.d.). Retrieved November 17, 2015, from https://www.cs.utah.edu/∼lifeifei/SpatialDataset.htm

[14] San Francisco County QuickFacts from the US Census Bureau. (n.d.). Retrieved November 17, 2015, from http://quickfacts.census.gov/qfd/states/06/06075.html

[15] Skabardonis, Alexander, and Richard Dowling. Improved speed-flow relationships for planning applications. *Transportation Research Record: Journal of the Transportation Research Board 1572 (1997)*, 18-23.

[16] Wolshon, B. & Taylor W. (1999). Analysis of intersection delay under real-time adaptive signal control. *Transportation Research Part C: Emerging Technologies*, 53-72.

*Karen was also in CS229 this quarter and her group used the OSM dataset. Other than that there was no overlap between the projects and all of the work presented here was done solely for the purposes of this class.