Aditya Shankar, Patrick Wilson, Alex Lancaster
EECS 586 - Final Project

Algorithmic Deep Dive - The Shortest Path Problem

Project Overview

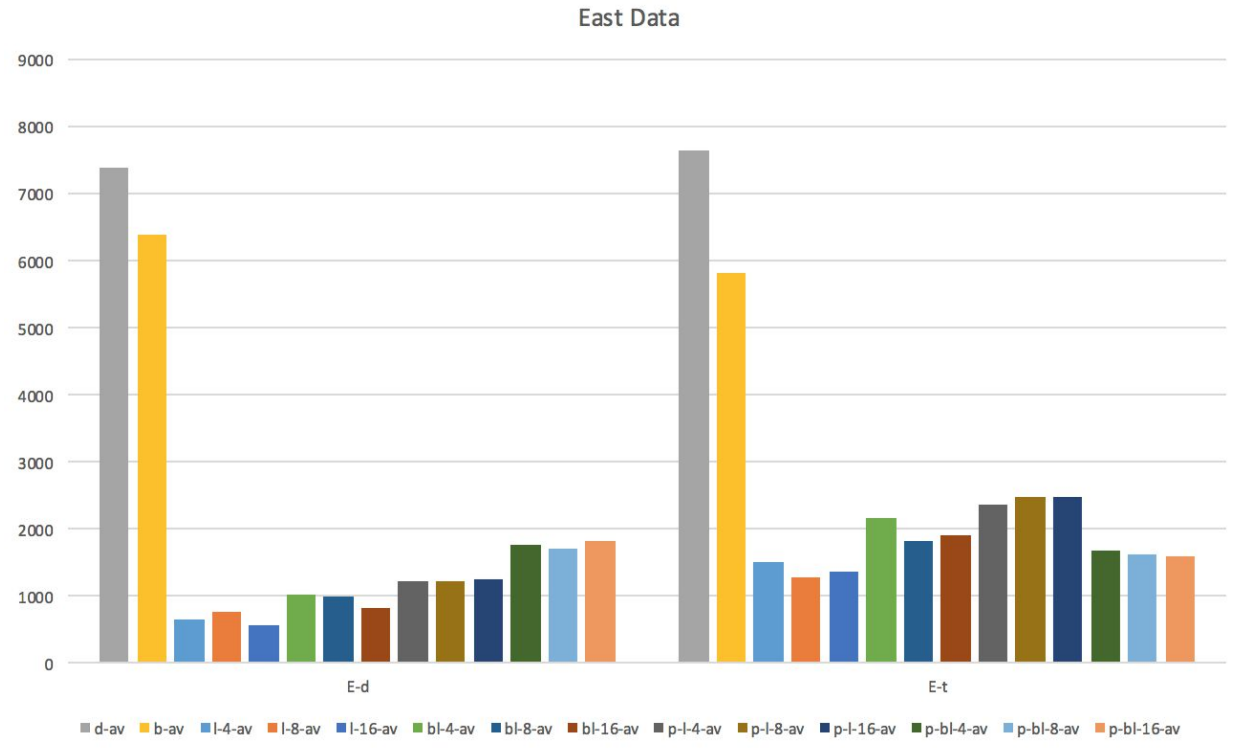
For our project we chose to investigate different Shortest Path Algorithms that are at the forefront of current algorithms research. Our work consisted of reading about and understanding the different algorithms, implementing them and then filming a YouTube video explaining the algorithms such that they could be understood by someone at the middle school level. Additionally we created a visualization tool to better understand how the algorithms interacted with physical transportation networks and explored shortest path calculation optimization research in the context of highway hierarchies.

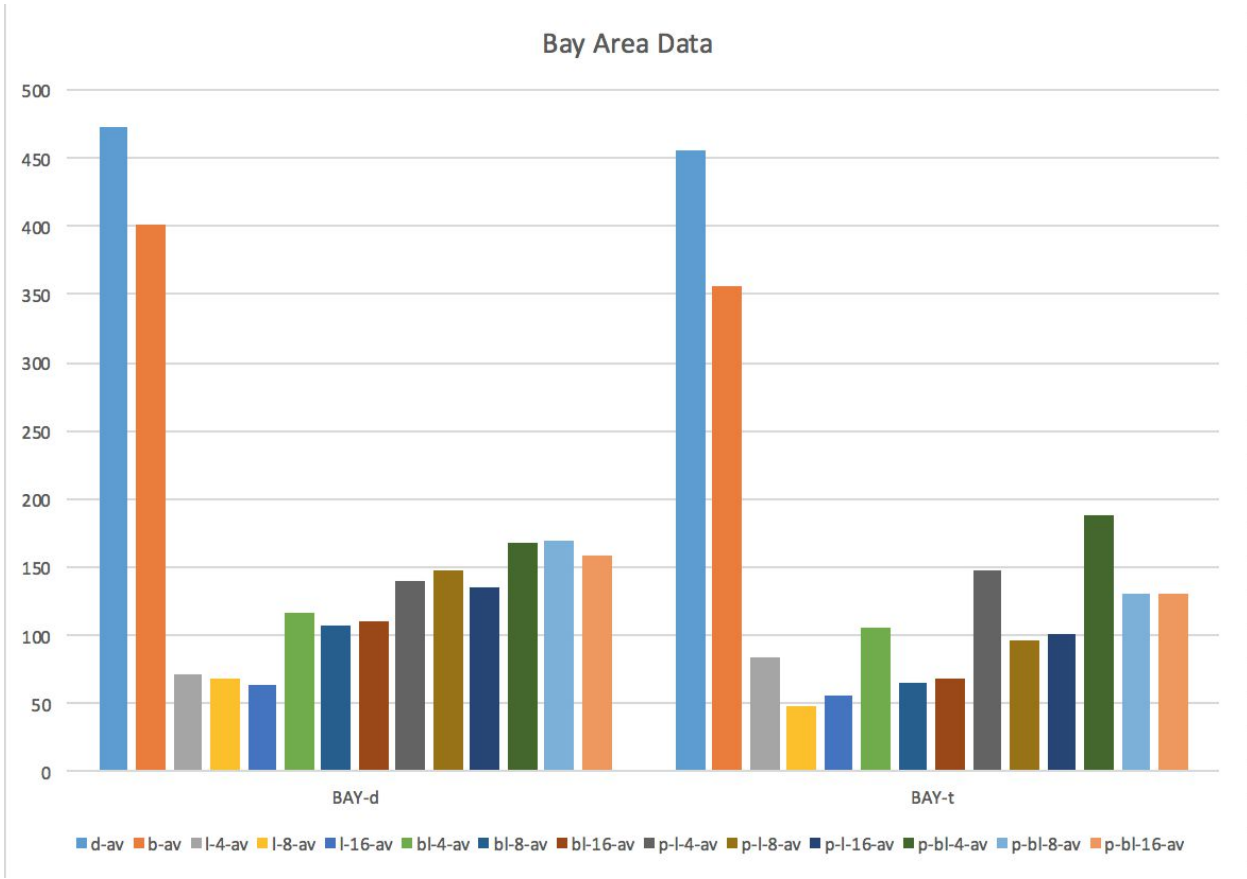
Algorithms Implemented

We explored the ALT Algorithm (both forward and Bidirectional), the Reach Algorithm as well as the concept of highway hierarchies. The ALT algorithms were the ones we spent the most time on as they were more complex and influential in research. For the purpose of comparison we also implemented Dijkstra's algorithm and Bidirectional Dijkstra's. We used the DIMACS Benchmark Map Datasets for our testing purposes.

Benchmarks:

Key: d = dijkstras, b = bidirectional dijkstras, l = landmark (ALT), bl = bidirectional landmark (bidirectional ALT), p = picked one landmark to use up front rather than iterating through all them each time, #'s = how many landmarks we chose


East Data
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
E-d
E-t
d-av
b-av
l-4-av
l-8-av
l-16-av
bl-4-av
bl-8-av
bl-16-av
p-l-4-av
p-l-8-av
p-l-16-av
p-bl-4-av
p-bl-8-av
p-bl-16-av


Bay Area Data
500
450
400
350
300
250
200
150
100
50
0
BAY-d
BAY-t
d-av
b-av
l-4-av
l-8-av
l-16-av
bl-4-av
bl-8-av
bl-16-av
p-l-4-av
p-l-8-av
p-l-16-av
p-bl-4-av
p-bl-8-av
p-bl-16-av

Northwest (1.6M vertices), random queries, 16 landmarks.

method	preprocessing		query		
	minutes	MB	avgscan	maxscan	ms
Bidirectional Dijkstra	—	28	518 723	1 197 607	340.74
ALT	4	132	16 276	150 389	12.05
Reach	1 100	34	53 888	106 288	30.61

Averages Table														
	d-av	b-av	l-4-av	l-8-av	l-16-av	bl-4-av	bl-8-av	bl-16-av	p-l-4-av	p-l-8-av	p-l-16-av	p-bl-4-av	p-bl-8-av	p-bl-16-av
BAY-d	472.6	401.2	71.6	67.6	64	116.6	107.6	110.4	140.6	147	135.2	167.8	170	159
BAY-t	455	356	84.4	48.8	55.6	105	64.6	68.6	147.6	96.8	100.8	187.4	129.8	129.8
E-d	7394.8	6385.8	668	756.8	560.2	1031	985.4	828.6	1235.2	1227.6	1252.4	1777.4	1714.2	1821.6
E-t	7652.4	5809	1497.8	1271.6	1355.4	2161.8	1814.8	1900.4	2379.8	2493.2	2490.6	1682.4	1613.8	1586

Shortest Path Visualization Tool

In order to better visually understand the behavior and optimizations of the different shortest path calculation algorithms in the context of the physical world, we created a framework that sits on top of the Google Maps API and allows us to upload a run of one of our shortest path calculation algorithms and display its progress on the map. We wrote the tool primarily in javascript and html and it has a number of parameters that can be tweaked to better visualize what it is that you are looking for. To run the tool, we select an output file from the C++ algorithm driver code and take that as input to the animation.html file and view in the browser. Code can be accessed in our git repo here.

Youtube Video Summarization

We wanted the research that we had conducted to persist for future students to be able to digest if they were interested in exploring the various leading methods. With this is mind, we created an instructional video that takes the viewer from the basic outline the shortest path problem to the very leading optimizing algorithms in academia including Dijkstra's, Bidirectional Dijkstra's, ALT, Reach, and Highway Hierarchies. We also include visualizations and common examples throughout the footage to better guide a viewer to understanding. The video is fairly comprehensive in regards to our total personal explorations into shortest path algorithms and may be the only video on youtube that addresses ALT, reach, and elements of bidirectional dijkstra's. That video can be found here and we would love to answer any questions on the work if anything comes up.

Summary

Overall, we found a significant performance improvement from Dijkstra's algorithm to the ALT algorithm. It was interesting learning about the use of landmarks for the computing a heuristic in a shortest path search. Overall, we saw that the ALT algorithms gave a significant performance boost in comparison to Dijkstra's. It was particularly cool visualizing this performance boost by

animating the algorithms in conjunction with Google Maps. We saw that the ALT algorithms touched significantly fewer nodes and edges than did the two Dijkstra's algorithms. We learned that heuristic functions and upper/lower bounds really play a huge role in allowing effective real time shortest path calculations in large transportation networks. Thanks for a great semester, and for allowing us to explore this interesting problem set.