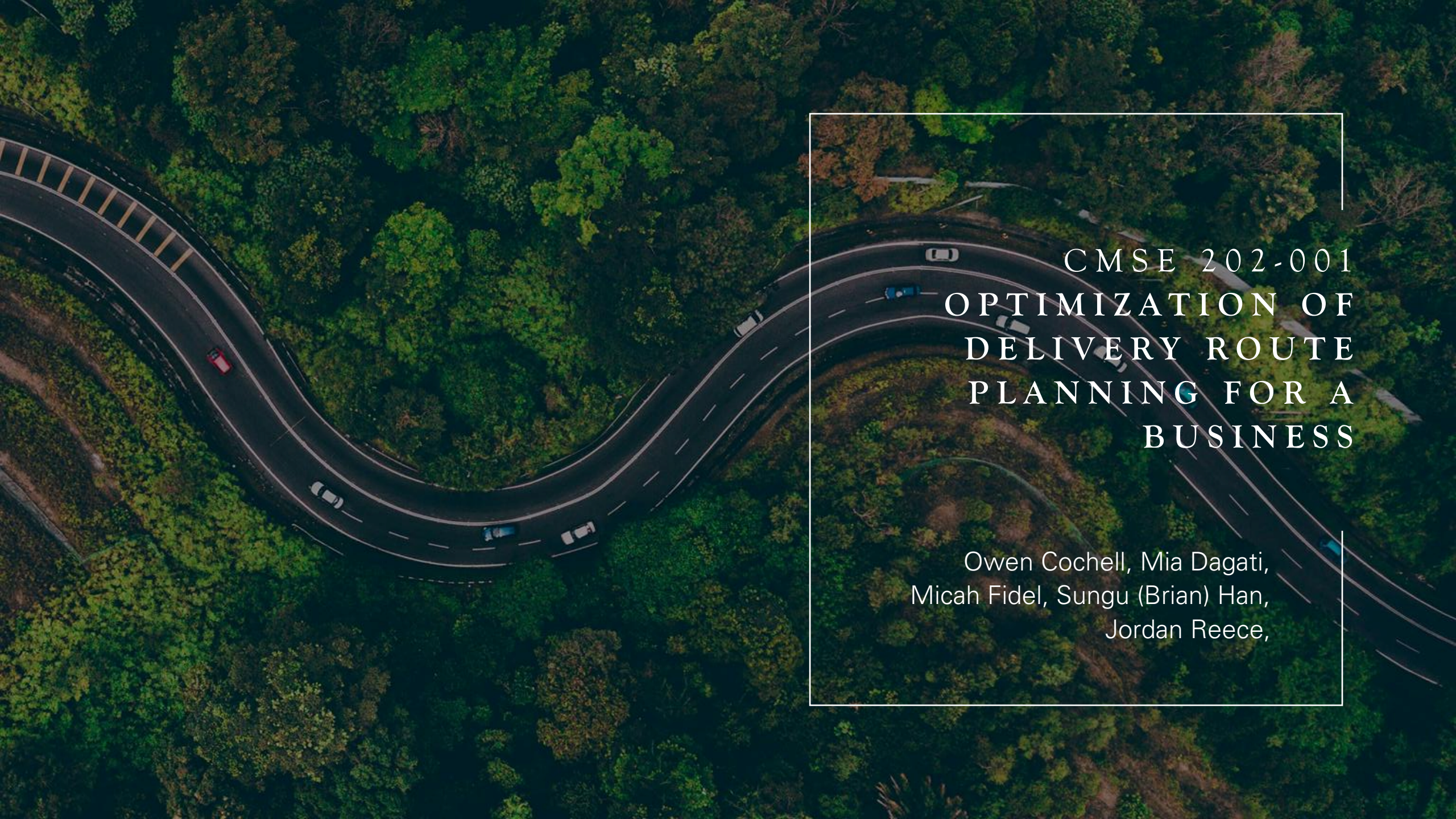


CMSE 202-001

OPTIMIZATION OF DELIVERY ROUTE PLANNING FOR A BUSINESS

Owen Cochell, Mia Dagati,
Micah Fidel, Sungu (Brian) Han,
Jordan Reece,

BACKGROUND

- Efficient routes mean less time on the road, lower fuel costs, and reduced vehicle maintenance—huge for small and mid-sized businesses.
- Timely deliveries = happy customers, in a competitive town like Lansing and East Lansing fast and reliable service can give a competitive edge.
- Fewer miles driven = lower carbon emissions this is also beneficial for businesses wanting to market themselves as ecofriendly.
- Research Question: Given a set of delivery locations, what is the most optimal route to take to hit all locations?

INTRODUCTION

- We intend to create a graphical model of Lansing and East Lansing, MI.
- Given delivery points, nodes, we will generate the optimal delivery route to hit all delivery points in the shortest time.
- Our goal is to model the city as a weighted graph to find efficient routes, using travel time and distance as the weights.

COMPUTATIONAL METHODS

The osmnx library will convert street data in a usable graph

- Queries the OpenStreetMap API
- Creates non-planar directed graphs
- Simplifies network topology and preforms intersection consolidation

The networkx library will perform graph operations

- Represents street graph in workable format
- Preforms optimal path calculation
- Helps determine distance traveled for fuel cost

Matplotlib will visualize the graph

DELIVERY REGION ANALYSIS

Figure 2: Road Map of delivery region

Figure 1: Polygon outline of delivery region

- We will create a model that will traverse through a networkx graph.
- Figure 1 is a polygon that shows the boundary/shape of Lansing and East Lansing, MI.
- Figure 2 takes the boundary in figure 1 maps out the roads within the boundary.

METHODS

- Using the speed limit and distance as the weights
  - Constructed a street network graph from OpenStreetMap data.
  - Graph edges represent roads and are annotated with both speed limits and travel distances.
  - Travel time is calculated by dividing edge length by speed (added via OSMnx utilities).
- Implement a Greedy algorithm (to determine the order of stops)
  - Starting from a specified origin, the algorithm chooses the next best stop based on the shortest available path at each step.
  - It does not look ahead or backtrack just makes locally optimal decisions.
- Taking the best path at every point: The algorithm evaluates all remaining unvisited delivery nodes and selects the one that...
  - Takes the least amount of travel time (for the speed-optimized route)
  - Covers the least amount of physical road distance (for the distance-optimized route)
  - Once at the last point return to origin

RESULTS:

OUT AND BACK SIMULATION

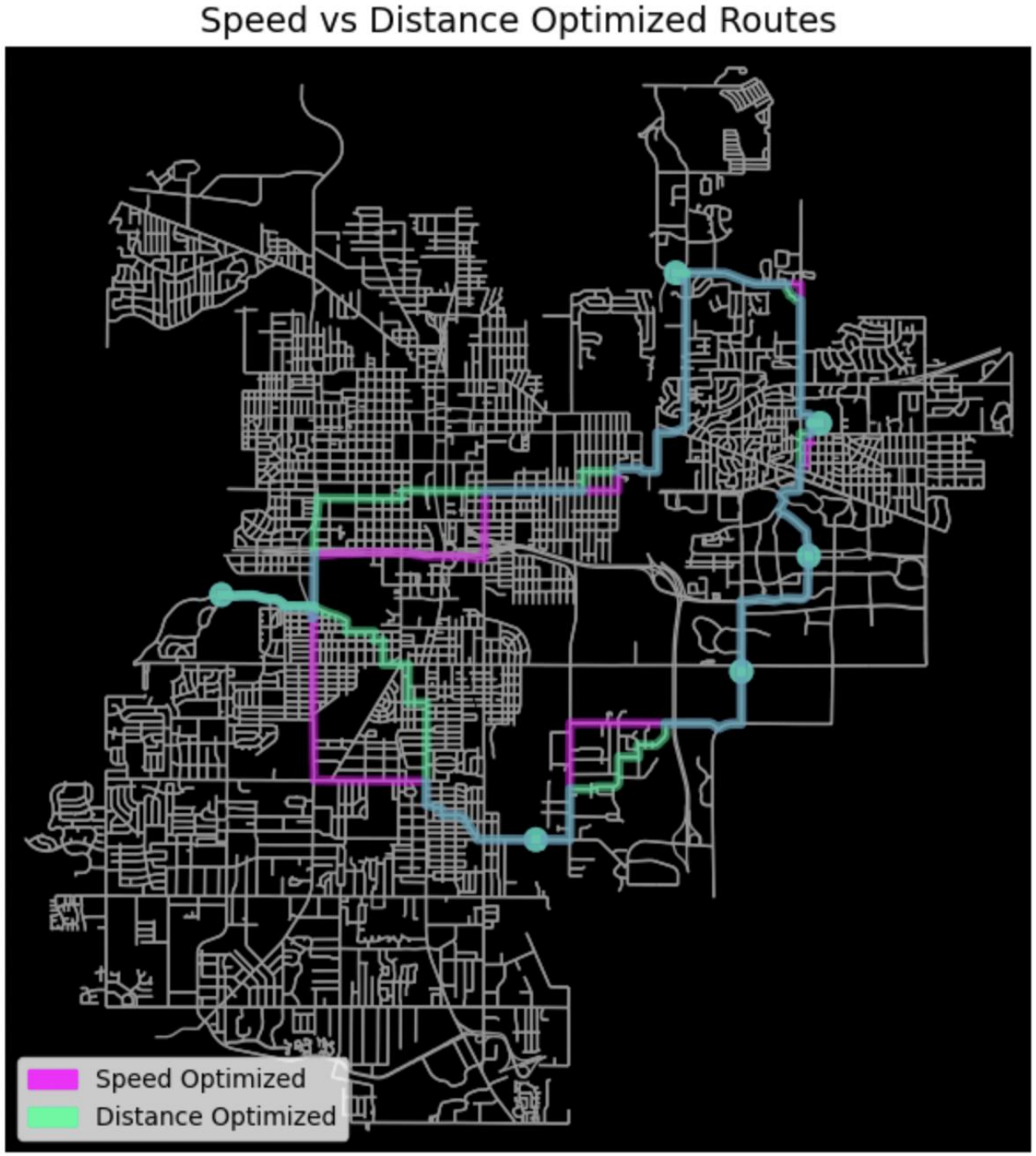
- Magenta – Route A
- Mint Green – Route B
- Light blue – Route A and B
- Route A is 21304 meters and takes 1474 seconds.
- Route B is 23491 meters and takes 1316 seconds.
- This is only for one delivery stop so there is not a huge difference in the routes

RESULTS:

REALISTIC ROUTE SIMULATION

- The speed optimized route is 33419.33 meters (~ **20.8 miles**) and takes 2307.94 seconds (~ **38.5 minutes**).
  - Order: [184250451, 2458329453, 184540619, 184538119, 184525746, 184568959, 184250451]

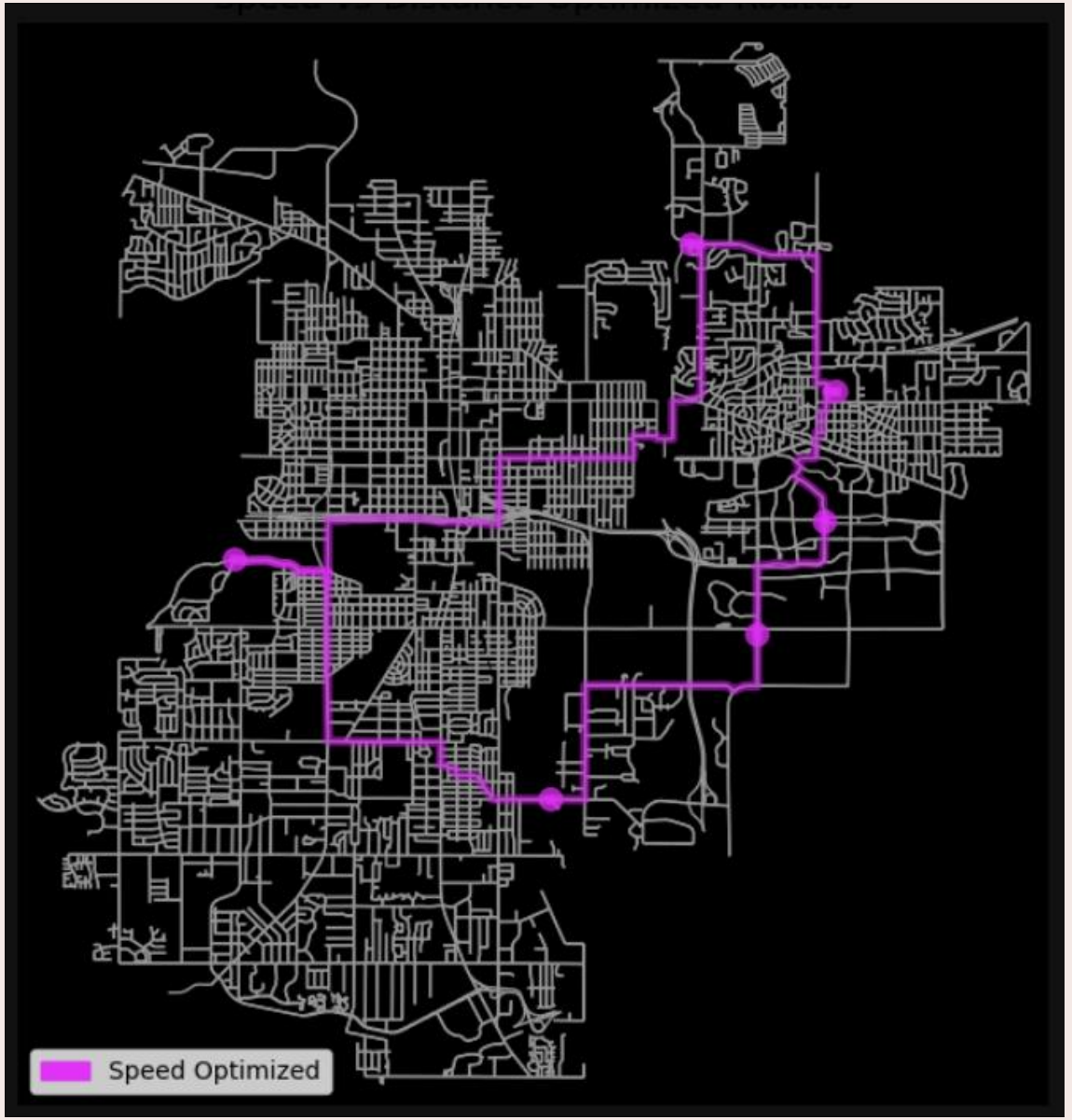
- The distance optimized route is 32996.19 meters (~ **20.5 miles**) and takes 2489.06 seconds (~ **41.5 minutes**).
  - Order: [184250451, 2458329453, 184540619, 184568959, 184525746, 184538119, 184250451]



CONCLUSIONS

Decision time: Which route should be suggested? One based on distance, or one based on speed?

- Distance Route - 51 Intersections
- Speed Route – 37 Intersections
- The **speed-based** route is the better option overall. It tends to be more fuel efficient since maintaining a steady speed uses less energy than frequent stops and starts. Plus, with fewer intersections on average, it reduces delays, making it the faster and more optimal choice overall.



COMPLICATIONS/ FUTURE APPLICATIONS

Struggles

The inclusion of roads on the boarders of the cities

Where roads connect through cities (Mich. Ave)

Improving the model's Accuracy with the factor of Time

Depending on the time of day some routes are not as optimal

Show which roads are most congested and therefore not a part of the optimal route

Include up to date road data for more accurate time estimations

Road closures

Construction

Hazards/Accidents

Usage of other algorithms

Another algorithm we could have used is Dijkstra's Algorithm

THANK YOU FOR LISTENING!

Questions?