Greener Pastures: Reducing Small-Business Emissions Through Route Optimization

Lizzie Healy

Georgetown University Data Science and Analytics

Problem Statement

One of the biggest contributors to climate change is the excessive emissions caused by transportation. For this project, I aim to tackle the problem of emissions in a small real-world example.

This entails making a flower shop 'greener' by attempting to optimize delivery routes for the local small business. The goal will be to determine if utilizing route optimization algorithms is a viable solution to unnecessary emissions from inefficient delivery routes.

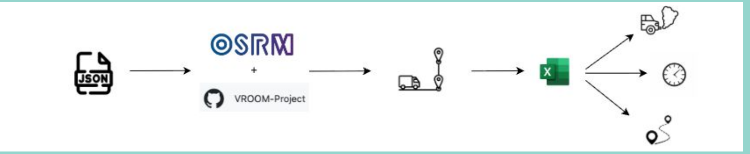
While this project is not computationally heavy, the importance is focusing on the real world aspect of and the scalability to other similar small and medium sized businesses, which together could make a measured difference in emissions and operating costs from the delivery portion of the transportation sector.

Data

The data was collected directly from a local flower shop in Pennsylvania, called Media Florist. This flower shop takes orders from their personal website, online FTD, and through phone calls. The orders are all manually inputted into a computer system, which houses the computerized tickets that include the location of delivery, personal information about the individual who placed the order, and specifications for the floral arrangement.

For the purposes of this research, the delivery location in the form of a street address was utilized. Deliveries from Mother's Day 2025 are selected, due to it being a particularly busy day. To supplement this data I worked backwards to determine each of the individual trips taken by the delivery drivers, and thus the specific routes that were taken on May 11. The number of stops, the number of trips, the total miles traveled and the total time taken were calculated.

The second portion of the data comes from the output of the VROOM route optimization algorithm, which takes in the same delivery addresses and utilized OSRM to produce its own version of optimized delivery routes. This data was in the form of a JSON that includes the trip number and the order of deliveries in that trip. It was converted to human readable output in the form of a excel spreadsheet with the address, the coordinates, the trip number, and the stop number.



Methodology

To determine the optimal route for the deliveries the VROOM-project was utilized, which is an open source route optimization coding algorithm, publicly available through GitHub. This project is an optimization engine meant to solve complex vehicle routing problems. To supplement VROOM, project open source routing machine (OSRM) was downloaded and ran on Docker in the background. OSRM is an open-source routing machine that will be queried by VROOM to provide distance and time measurements between locations. With these queries, the VROOM algorithm calculated the optimal route plan based on minimizing time and distance.

Once both engines were up and running, the JSON file of delivery addresses was inputted, as well as the number of vehicles, and the number of max deliveries per vehicle. The algorithm provided the optimized route plans to meet the requirements and complete the deliveries. This output was converted to an excel format for ease of read.

Again, the total mileage for delivery completion, the number of trips takens, the number of stops, and the total time taken were calculated.While the algorithm ws running, the carbon emissions from the computation were calculated utilizing Code Carbon. The location of the server was documented to understand how carbon footprint can differ by location.

An average estimate of the carbon emissions per mile for the fleet of cars was created based on MIT's online Carbon Counter, which provided the lifecycle emissions for a specific vehicle based on vehicle production, fuel production, and fuel use. The fuel usage emission value for some of the utilized cars was isolated and the average emissions among these cars was calculated, creating a rough estimate of the emissions per mile of driving and emissions per minute of idling for the vehicles in the delivery fleet.

Evaluation

To evaluate the algorithmic routing compared to the manual routing, a series of equations was employed to assess its overall feasibility and benefits (or disadvantages). These equations calculate the emissions from each method, the cost of fuel, the cost of labor, and compare the time, trips, distance, emissions, and cost.

$$tot\_emit_{manual} = \sum_{i=1}^{n} [(EPM_i \times Miles_i) + (EF_i \times Stops \times \overline{d})]$$

where

- n = is the number of total trips
- EPM_i = emissions per mile on trip i
- $Miles_i$ = total number of miles travelled on trip i
- EF_i = emissions per minute while idling for trip i
- $Trips_i$ = number of trips made
- $\overline{d}$ = average number of dwell minutes at each stop

$$tot\_emit_{algo} = \sum_{i=1}^{n} ((EPM_v \times Miles_i) + (EF_v \times Trips_i \times \overline{d})) + compute\_emissions$$

where

- $compute\_emissions$ = carbon emissions from the algorithmic computation process

$$FuelCost = P_{fuel} \times \sum_{i=1}^{n} \frac{Miles_i}{MPG_i}$$

$$LaborCost = w \times \sum_{i=1}^{n} T_i$$

$$TotalCost_{manual} = FuelCost + LaborCost$$

$$TotalCost_{algo} = FuelCost + LaborCost + AlgoCost$$

$$reduction_{cost} = TotalCost_{manual} - TotalCost_{algo}$$

Results

Two different configurations of the VROOM algorithm were created and tested, one with 10 maximal deliveries per vehicle and one with 8 maximal deliveries per vehicle. For reference, the absolute maximum deliveries per vehicle is around 13 for mini vans, while a normal sized car averages 7-8 deliveries.

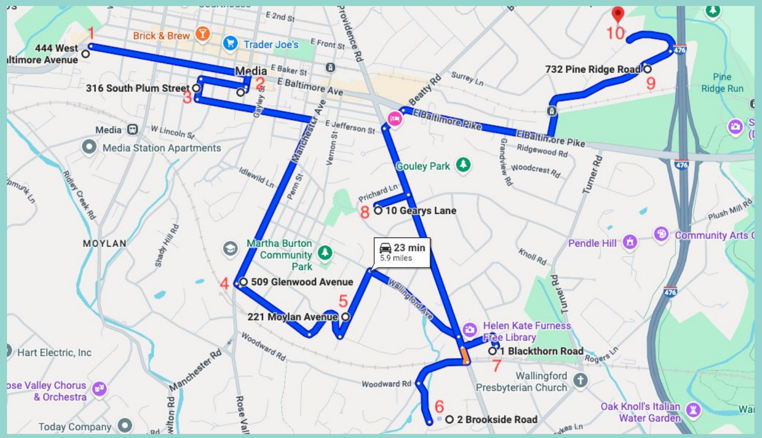
The results of the manual routing, and algorithm 1 (10 max) and algorithm 2 (8 max) as shown in the table below. In addition, an example of one of the optimized and feasible routes is shown to the right.

	Time (mins)	Distance (mi)	Trips	Emit (pd CO2eq)	Cost ($)
Manual	583	230.2	14	319.89	2686.746
Algo 1	494	163.4	10	246.27	2174.382
Algo 2	515	173.5	13	257.39	2277.005

As seen above, the results of VROOM produced two optimized route plans that proved to be more efficient than the manual version. Most importantly, the routes were logical and completely possible in terms of the physical driving route. The algorithms both improved upon the mileage, the number of trips, the total time elapsed for a day of delivering, the estimated carbon emissions (even when considering computational emissions), and overall costs from labor and gas. The algorithm using 10 deliveries max per vehicle outperformed both manual and the 8 max algorithm and, in the end, was able to save…

1h29
66.8 mi
4 trips
73.62 pd CO2eq
$512.36

```
[codecarbon INFO @ 16:43:10] Energy consumed for RAM : 0.000003 kWh. RAM Power : 6.0 W
[codecarbon INFO @ 16:43:10] Energy consumed for all CPUs : 0.000022 kWh. Total CPU Power : 42.5 W
[codecarbon INFO @ 16:43:10] 0.000025 kWh of electricity used since the beginning.
Estimated O₂ emissions: 0.000010 kg
```



Discussions and Limitations

Overall, the algorithm works reasonably well, in that it decreases emissions without incurring additional costs. Furthermore, this particular case study could be utilized in a real-world scenario for the florist shop and other delivery businesses.

However, this algorithm is not suitable for smaller delivery days, which are basically any days of the year that don't fall near a major holiday. So, in conclusion, for Mother's Day, Christmas, and Valentine's Day this algorithm could save time, money, and emissions, but for the rest of the days, a manual method is more feasible.

The other main limitation to this work is the lack of adjustability of the algorithm. The initial work set out to test the route optimization using a variety of different real-world delivery scenarios including specifying when an order must be delivered by, specifying more fine-grained limits for each specific vehicle to account for vehicle sizes and larger floral arrangements, and assigning specific routes to specific cars. Instead the algorithm works with the number of cars and provides that number of optimized routes, only providing one route for each car. In theory this can be tuned by looking at this number instead as the number of delivery routes and hyperparameter tuning to find the optimized number of routes and then in the real scenario dividing these routes among the drivers, however, this adds an additional layer of complexity to the calculation and more computational emissions.

Carbon Cost

Overall, Code Carbon estimated 0.000010 kg for each run of the VROOM optimization algorithm. During the process of coding and tweaking the algorithm and full set-up, I ran the full pipeline about 20 times. In addition, I ran smaller versions about 10 times. This equates to about 0.3 grams of CO2 equivalent.

In addition, during the process I estimated about 10 queries to Chat GPT, which each produce ~3 grams of CO2. This translates to 30 grams of CO2.

Finally, utilizing a laptop emits ~10g CO2eq per hour and, by my estimates, this project in its entirety took about 40 hours, for a total of 400 gCO2eq.

The final rough estimate is about **430.3 gCO2eq** from start to finish.