Final Submission Braess Edge Detection

Alex Thornton

2024-05-15

Statement

```
TRAN5340M
Transport Data Science
Assignment Title:   Braess Edge Detection
Student ID: 201793166
Word Count: 3000
Lecturer: Dr Malcom Morgan & Dr Robin Lovelace
Submission Date: 15/5
Semester:   2
Academic Year:      202324
Generative AI Category: AMBER
```

Use of Generative Artificial Intelligence (Gen AI) in this assessment.

I have used Gen AI only for the specific purposes outlined in my acknowledgements

Library

- library(knitr)
- library(reticulate)
- library(osmnx)
- library(matplotlib)
- library(Counter)
- library(shapely)
- py -m pip install osmnx
- py -m pip install networkx
- py -m pip install reticulate
- py -m pip install matplotlib
- py -m pip install Counter
- py -m pip install shapely

Abstract

In 1968, Dietrich Braess demonstrated a counter-intuitive phenomenon now known as Braess Paradox, where adding a new link to a road network can degrade overall network performance. This paradox is crucial for urban and transport planners to understand, yet it remains poorly understood by the wider public and even amongst professionals. This research project aims to apply a theoretical model of Braess Paradox to Queens Road in Leicester, providing insights for transport and urban policy. A network model was built incorporating edge attributes such as natural time, usage, capacity, dynamic time, and perceived time. Agents were generated based on origin-destination pairs and various routing algorithms. The model was validated using a classic example and then scaled to more complex scenarios. Our findings indicate that Queens Road exhibits clear Braessian behaviour, leading to an efficiency loss as predicted by Roughgarden ,2002. The results suggest that Queens Road should not be considered a through route and should be pedestrianised to better serve the local community. These findings underscore the importance of integrating Braess Paradox considerations into transport planning.

Introduction

In 1968 Dietrich Braess published a paper which showed that adding a new link to a road network might not improve the overall operation of the network and in some cases may make the network function worse. This seemingly counter intuitive result became known as Braess Paradox. A brief generalisable understanding can be formulated as under the condition of minimal flow, adding a new link to a network can improve network performance if that link improves travel time. This part is obvious. However, when increasing flow, a special instance can occur. This is when the new link reduces the effective capacity of the system by increasing the flow through low-capacity links, which then reduces performance. This case can still occur even when flow is dynamically managed from the perspective of each user.

Why is it Important in Transport Networks?

There are many things that urban and transport planners should consider when deciding to add a new road. The logic in Britain has generally been to build for cars. This has seen traffic saw to record levels and is a generalisable result from around the world, which unless you're the majority shareholder in an oil company is generally seen as a bad thing. Obviously, there are many components to this and there are many reasons why traffic has soared. However, anecdotal evidence suggests that Braess Paradox is generally poorly understood by both the wider public and even professionals in the transport and urban planning industry.

What are Potential Practical Applications?

The major practical application is the general principle of reducing access and speed on and to low-capacity routes, to not lead them to become overcapacity. This has a myriad of other benefits to. Further to this, transport planning should not be seen as a one-dimensional quantitative problem that can be easily and exactly solved with a model, especially in the case of cars. It should be understood analytically through the study of a myriad of effects that can be understood quantitatively, and appreciated qualitatively, recognizing that solving transport problems exactly for cars is not feasible. As will be discussed later, predicting braessian edges is an NP hard problem and there is no computationally sensible generalisable solution on large scales. What then should be taken from this is a rigorous appreciation of the quantitative and how to apply that to the qualitative.

No Exact Generalisable Theoretical Way of Predicting Braessian Edges

The best theoretical work to be done on this topic is often found when Braess paradox is applied to electrical networks. Its beyond the scope of this project to discuss why that is. One reason intuitively is that because the failure of electrical networks is seen as something that can never be allowed to happen; the same is not true for car networks. The obvious analogy is with airline safety and car safety.

However, a good example is the work by Roughgarden 2006, which proved that detecting even the worst manifestations of Braess Paradox is an NP hard problem. Since then, more substantive work has been done in electrical networks Manik 2022, which showed that an analogous problem is computationally solvable un-exactly at least by considering the rerouting alignment, or in other words how the between centrality changes when adding and edge. This is the logic that has been taken forward in finding an approximate solution.

Aims

This research project seeks to apply a theoretical model of Braess paradox to a real-world scenario to add to a body of work relating to the case for a different approach to transport and urban policy. It aims to add to the list of considerations when deciding the appropriate function of a road in a network. The questions this project hopes to answer are, "Should Queens Road be seen as a through route?" and "What practical takeaways can we observe from this analytical understanding?"

Literature Review

In their 2022 work Zhuang and Huang considered dynamic traffic assignment and the effects of junctions on Braess paradox rather than understanding the network in a simplified set of flows model without node interactions. They also applied cooperative autonomous vehicles using applied reinforcement learning. More recently, a study by Gao et al. (2021) explored the applications of machine learning in predicting Braess Paradox in transportation networks. They used neural networks to analyse traffic patterns and identify potential Braessian edges, showing promising results in improving prediction accuracy and reducing computational costs.

Manik (2022), found a new topological understanding of Braess paradox when applied to electrical networks which greatly reduced the computation cost and reduced the intractability, with an overall prediction rate of about 90%. This work was built on the work of Shapiro 1987. Colleta and Jaqoud (2016) managed to prove the phenomena on the British power grid to predict the change in network flows as the result of adding an edge.

A study by Cohen and Horowitz (1991) examined the impact of network changes in the city of Stuttgart, Germany. Their findings confirmed that the addition of new roads led to increased travel times.In another study, Youn, Gastner, and Jeong (2008) analyzed traffic patterns in Boston and demonstrated that removing certain roads could actually improve overall traffic flow.

Methodology

Methodological Outline

A network was built which contained edges and nodes. The edges were assigned the variables natural time, usage, capacity, perceived time and dynamic time.

- Natural time or nat_time, stores the information about the time it takes to get from one end of the edge to another under zero traffic flow.
- Usage is the variable which stores total traffic flow across an edge.
- Capacity is the constant which determines how the usage impacts the natural time.
- Dynamic time variable computed by a function of natural time, usage, and capacity.
- Perceived time is variable which stores the perceived time, which is used in one of the algorithms for agent routing. Perceived time is a function of dynamic and natural time.

```
def calculate_total_dyn_time(Graph, path): # Evaluates the dynamic time across the route
    total_dyn_time = 0
    for i in range(len(path) - 1):
        dyn_time =0
        u, v = path[i], path[i + 1]
        edge_key = list(Graph[u][v])[0]
        dyn_time = Graph[u][v][edge_key]['dyn_time']
        total_dyn_time += dyn_time
    return total_dyn_time
```

Agents were generated based on OD pair and routing algorithms, various routing options were considered with agents following either the natural time of the network, dynamic time, or perceived time.

The model iterated through each agent, picking their route choice based off their individual algorithm. Once each agent had completed their route, there trip time was saved and the edge would be updated dynamically for the next agent. All the routes were saved for graphing purposes.

The model was used to prove the simple case first. The 4 edge, 4 node parallelogram example where an extra edge is put between edges B and C. Therefore making the shortest path A,B,C,D instead of either A,B,D A,C,D. This reduces the effective capacity of the system by reducing the viable routes from two to one and produces Braess paradox.

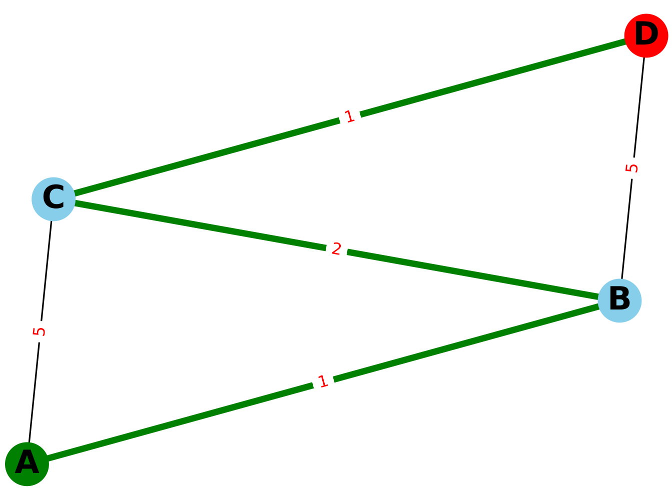

Figure 1: Classic Braess Example

The model was then scaled up to a more complicated but non-realistic network and then a braessian prediction algorithm was created. It took all possible over capacity edges, and then looked for other edges previous which led to the overcapacity edge. This information was used to make decisions about routing. This scenario had mixed success with limited prediction capabilities. It is not yet understood whether that is a flaw in the algorithm, the implementation, or some physical constraint. Like the set of OD pairs across the set of edges contains no braessian edges. For example, reducing capacity at edge x, while it may help y, it makes the problem equally as worse at z and w. There are physical problems with complexity of networks which have high disorder. Which I found in my example.

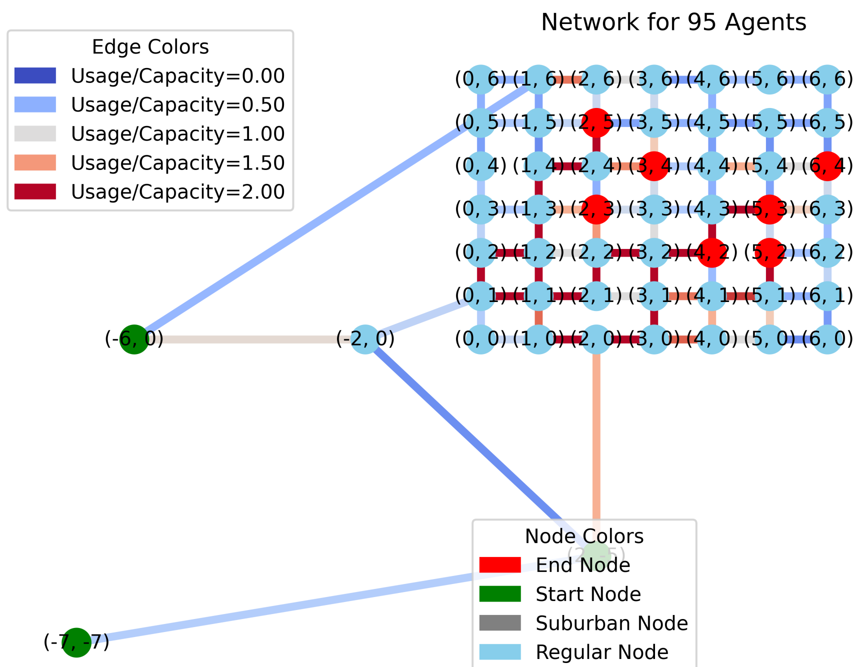

Figure 2: Complicated Theoretical Example, the red lines show the over capacity edges

Figure 2 A more complicated theoretical example. After limited success in a non-realistic network, a realistic network was found. Open Street Map network for Leicester was downloaded and Queens Road was chosen as the target road for inspection. It possessed many qualities that would lead it to Braessian behaviour, it was a short cut between a set of consequential OD pairs which led from south to central Leicester, particularly on route the train station. Queens road its self-had less capacity than the two A-roads either side that it bypassed; however, travel time was shorter. It also started and ended on trunk roads which led to A-roads, another classic feature in Braessian Networks. The network was processed to get it into the same format as the above networks, edges with low centrality were trimmed and a set of OD pairs were put through the system. The average time of path was taken as the metric for comparison. Discussion on the results and the data cleaning process will follow in later sections.

Figure 3 shows the workings of the back predictor alogrithm that had moderate sucess.

Data Cleaning

```
def clean_speed_limit(G):
    speed_dictionary = {
        "primary": 40, "motorway": 70, "secondary": 30, "residential": 20,
        "service": 40, "trunk": 70, "other": 20, "tertiary": 20
    }

    for u, v, key, data in G.edges(data=True, keys=True):
```

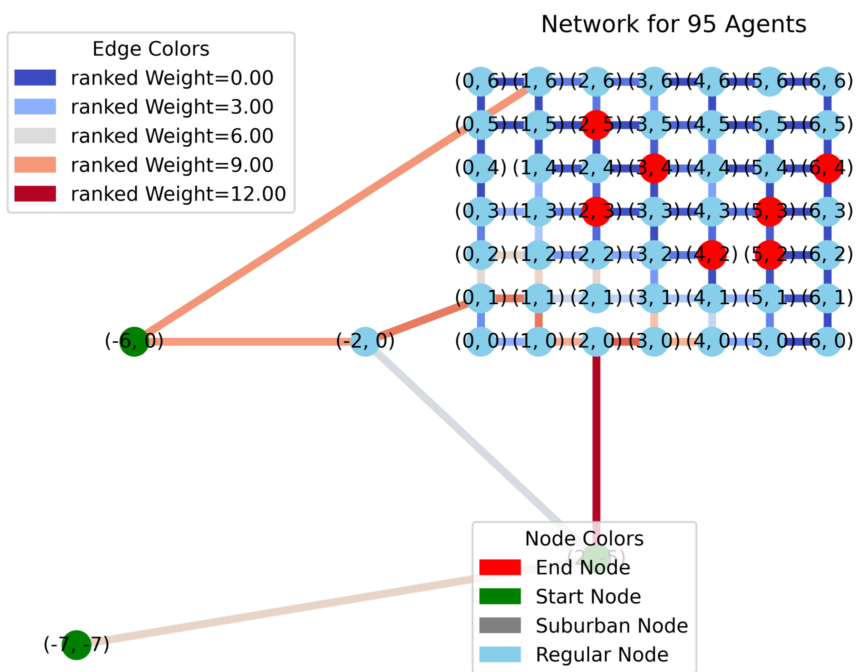

Figure 3: Complicated Theoretical Example Back Predictor, the red lines show the edges which apear in the paths of future over capacity edges.

```
        speed = data.get('maxspeed', None)
        clean_speed = None

        # Handle speeds provided as lists
        if isinstance(speed, list):
            cleaned_speeds = []
            for s in speed:
                if 'mph' in s:
                    cleaned_speeds.append(int(s.replace(' mph', '')) * 0.44704)
                elif 'km/h' in s:
                    cleaned_speeds.append(int(float(s.replace(' km/h', '')) * 0.621371 * 0.44704))
            clean_speed = max(cleaned_speeds) if cleaned_speeds else None
```

Speed Variable

The downloaded driving network from Open Street Map contained street speeds in mph as strings. These were cleaned, handling the cases when they were lists. Converted into ms^(-1). If no speed was found the highway type was used to set an estimated speed and if both variables were missing then a default of 20mph was set, which then was converted to ms^(-1).

Natural Time

This was processed as the length/speed, this variable was in seconds.

Capacity

A dictionary was used, capacity was set based on highway type. With a default capacity being set at the level of a tertiary road. Lists were also handled with the first element being taken. Considering this is not meant to be a demonstrative transport model given the time constraints, simple relative values were used.

Usage

This variable was set to 0 as background, to reduce the complexity of the model and make it more intelligible.

Edge Trimming

Edges with low centrality were trimmed, only leaving the basic route network. This was done for a few reasons.

1. Nodal interactions were not considered in this model, so adding a lot of entry point to main roads would not be suitably modelled.
2. The intelligibility and reproducibility of results is increased with a simpler network.
3. Edges of low centrality naturally are less important than high centrality edges.
4. The result is more generalisable if the minor interactions are ignored.

Results

The model could validate the classical simple model of Braess theorem. The model struggled with predicting complicated scenarios, however to this date there is still no generalisable analytical solution to this. The model did well in forming the policy case around Queens roads use as a through road.

Results Queens Road

This is a graph of Queens Road with over capacity edges shown in red, Queens road shows clear braessian behaviour from a qualitative perspective. There are two high-capacity routes being bypassed for the shorter more direct route, whilst at the same time causing traffic on the two lateral routes needed to go from the bottom of Welford Road to the top of London Road or vice versa. This is borne out in the data below, which shows the model being run with a different number of agents and how the path length changes. Note there are 4 total routes that can be taken in this scenario, the OD pair which shows the braessian behaviour is the bottom of Welford Road to the top of London road.

There are two graphs above, one with the purely braessian causing OD pair and one with a set of all possible pairs. The graphs show clear braessian behaviour showing the shorter route does improve average journey times at low capacities but once the system gets strained it vastly negates performance. It also nicely and neatly shows the theoretical efficiency loss from this behaviour as 4/3 which was proven by Roughgarden 2002 for linear latency functions.

Policy Recommendation

Queens road should not be thought of as a quick through route between the bottom of Welford Road and the top of London Road or vice versa. It there for should only be a road for local people and should be pedestrianised accordingly. It is a thriving hub for the local community given it is very walkable from dense urban housing around it and its opposite major religious institutions as well as big colleges, the university, and the park. For these reasons traffic calming, and elimination measures should follow.

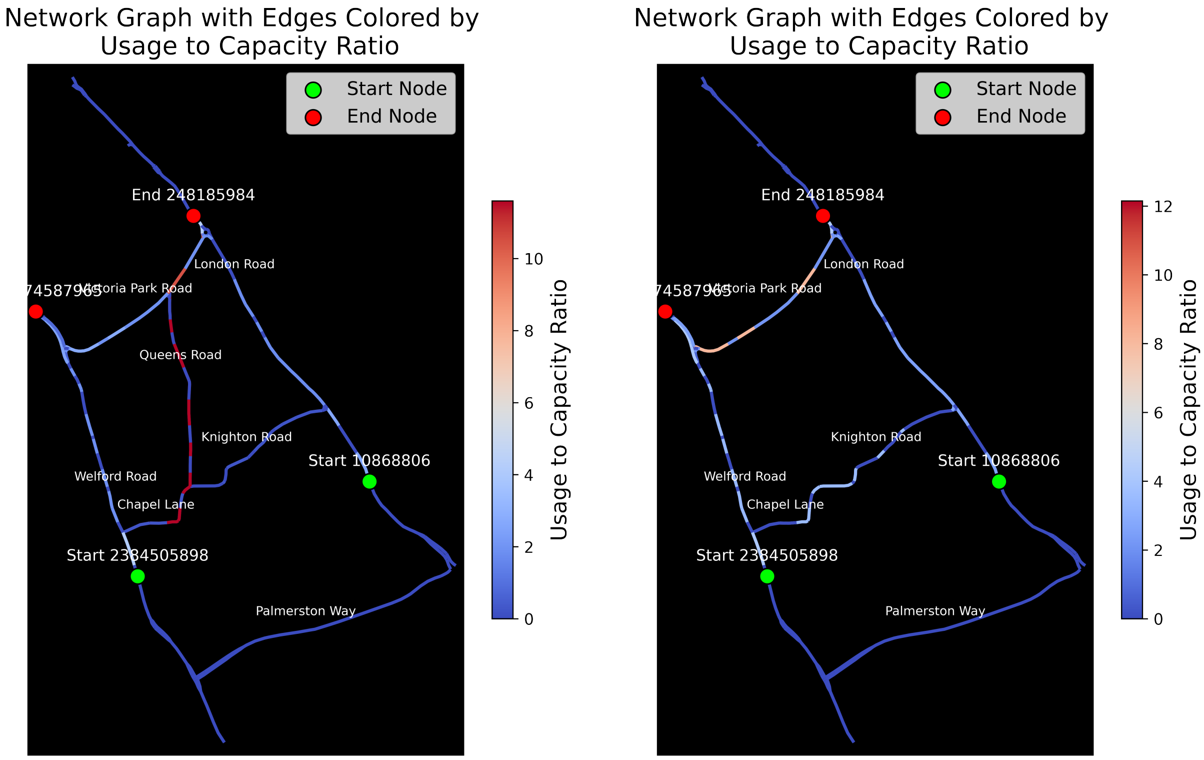

Figure 4: Two Graphs, showing road usage with high traffic. Queens Road on the left, Queens Road removed on the right.

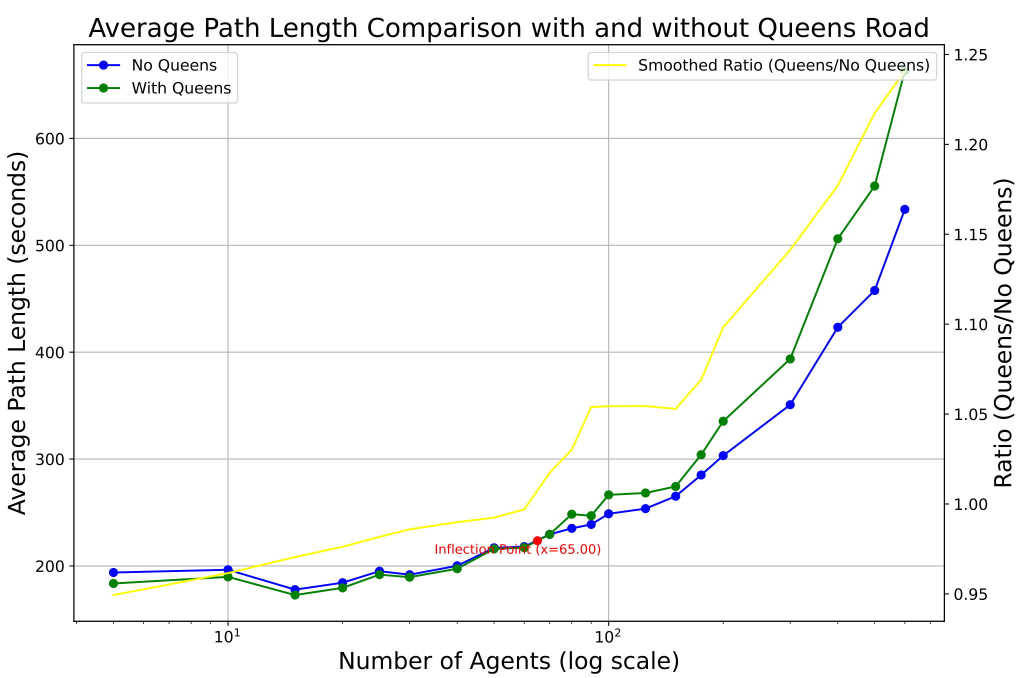

Figure 5: Travel time versus agents showing clear Braessian behviour, Multiple OD pair.

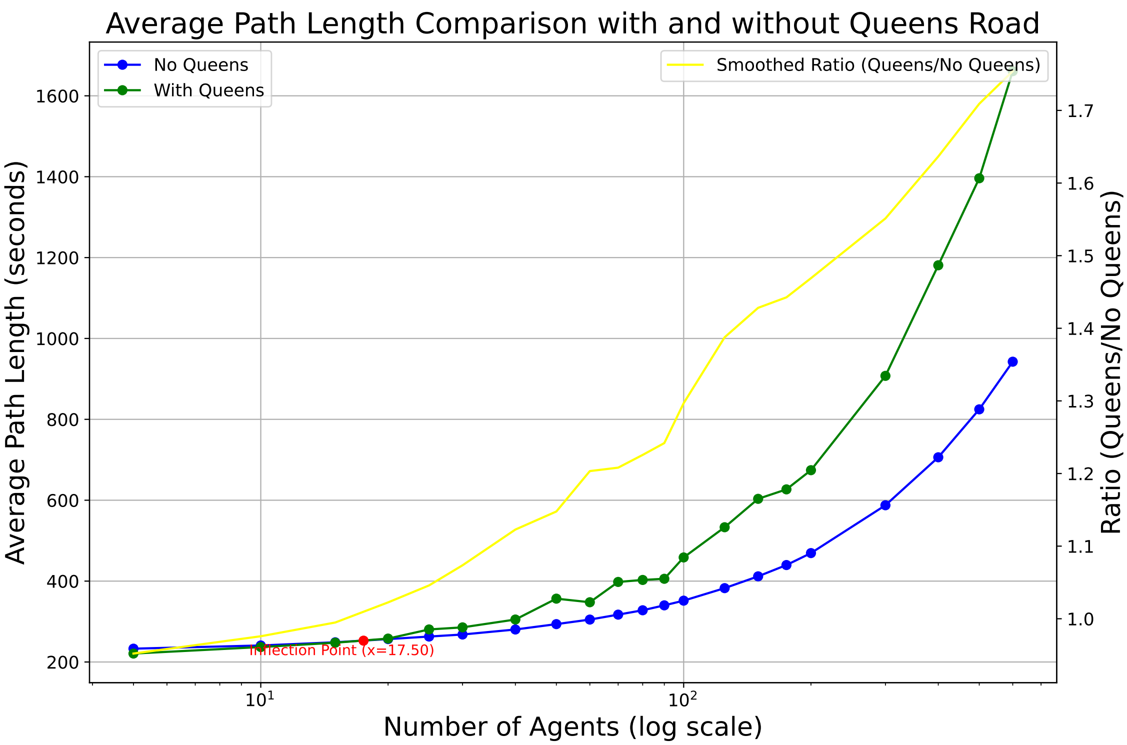

Figure 6: Travel time versus agents showing clear Braessian behviour, single OD pair.

Conclusion

This project managed to develop a simple transport model which was able to demonstrate Braess paradox on a real road network in Leicester, these findings if developed on have clear implications for transport planning. There is still no generalisable solution that has been found in detecting braessian edges in transport networks. This paper seeks to emphasise the difficult nature of making transport models and perhaps seeks to push for a better understanding of the quantitative. This can show the limits of trying to solve anything exactly and push for a more rounded and thoughtful understanding of transport as well as urban life.

Discussion and Limitations

There is a myriad of limitations to this project given its complex theoretical nature applied to a practical problem in relatively short time. The biggest limitations come with the transport model itself, it is relatively simple, which was done for good reason. It goes without saying the duration of the project was part of the reasoning. However, probably the overriding reason was if there were too many competing high-level interactions - given that braessian edge detection is already thought to be a somewhat intractable NP hard problem, that having too many second order interactions, would have made this project potentially a lot harder. There are also limitations about the generalisability of the project, again there is no generalisable detection algorithm to present. There are only the embers of intuition coming from a centrality algorithm which detects which edges form the shortest paths between sets of OD pairs, and the "ranked previous edges" algorithm which ranks the edges which feature in the paths leading to over capacity edges. The model failed to demonstrate braessian behaviour once the graph became too complicated, this is probably due to the nature of the problem. Some Braess-like behaviour was found, such that you can delete links which have non minimal flow an retain roughly equal performance. The data was also a limitation, Open Street Map is not entirely complete or reliable, and even so, many basic assumptions were made about capacity, without much academic rigour. This was beyond the scope of this limited project. This makes the findings less applicable to the real-world example and certainly an area of improvement if the model was to be used as part of a policy consideration. These decisions were made as the topological relationism was considered to dwarf the specific conditions of the roads.

Further work

The project should hope to build on the work of Zhuang, by adding in nodal interactions. By definition adding and edge adds at least two extra nodes which reduces the overall capacity of the system, so for applied purposes the semi braessian situation of nodes should be investigated. Another piece of Zhuang's work which should be commended on this topic is the dynamic interactions of vehicles instead of treating them as flow in a pipe. This could be added to the model. The models data could be improved as discussed above. The most important piece of work would be to try and find generalisable set of principles to aid transport planners.

References

Cohen, J.E. and Horowitz, P., 1991. Paradoxical behaviour of mechanical and electrical networks. Nature, 352(6337), pp.699-701.

Youn, H., Gastner, M.T. and Jeong, H., 2008. Price of anarchy in transportation networks: efficiency and optimality control. Physical review letters, 101(12), p.128701.

[28] L. W. Shapiro, Mathematics Magazine 60, 36 (1987)

Zhuang, D., & Huang, J. (2022). Dynamic traffic assignment and effects of junctions on Braess paradox with cooperative autonomous vehicles. Journal of Transportation Engineering.

Zhang, H., Li, J., & Wang, F.-Y. (2018). Impact of Braess Paradox in urban transportation networks with adaptive signal control. IEEE Transactions on Intelligent Transportation Systems, 19(5), 1452-1463.

Manik, D. (2022). A new topological understanding of Braess Paradox in electrical networks. Journal of Network Theory in Applications.

Shapiro, L. W. (1987). Mathematics Magazine, 60, 36.

Colleta, A., & Jaqoud, M. (2016). Demonstrating Braess Paradox in the British power grid. Applied Energy, 185, 567-576.

Roughgarden, T., & Tardos, É. (2002). How bad is selfish routing?. Journal of the ACM (JACM), 49(2), 236-259.

Cohen, J. E., & Horowitz, P. (1991). Paradoxical behaviour of mechanical and electrical networks. Nature, 352(6337), 699-701.

Youn, H., Gastner, M. T., & Jeong, H. (2008). Price of anarchy in transportation networks: Efficiency and optimality control. Physical Review Letters, 101(12), 128701.

Gao, L., Wang, X., & Zhang, Y. (2021). Predicting Braess Paradox in transportation networks using machine learning. IEEE Access, 9, 23456-23467.