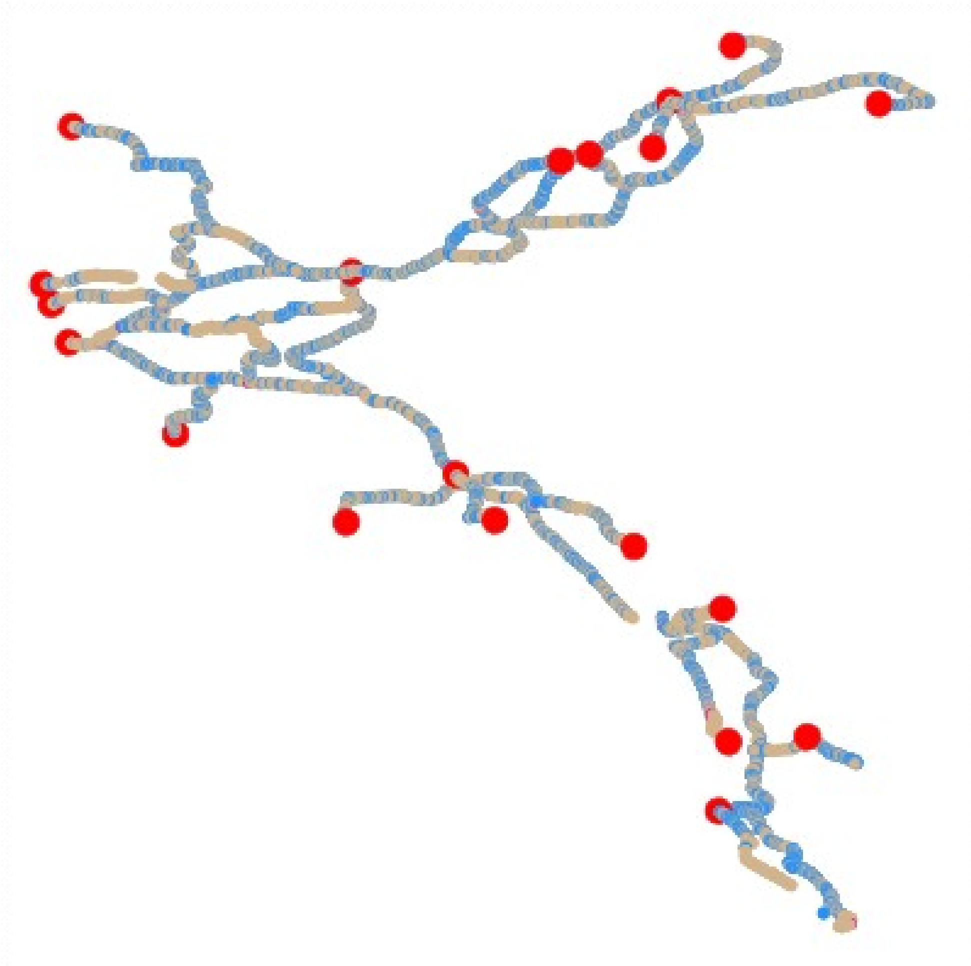
# Bangladesh’s Expanding Transport Infrastructure:

## Criticality & Vulnerability in Network Analysis N1 - N2

Group 22
EPA1352 - Assignment 4


| | |
|---|---|
| *Anna Noteboom* | *4564979* |
| *Auriane Técourt* | *5397243* |
| *Floris Boendermaker* | *4655605* |
| *Job Onkenhout* | *4595769* |
| *Zara-Vé van Tetterode* | *4577701* |

# Content

# 1 Introduction

The extremely densely populated country of Bangladesh suffers frequently from the negative effects of climate change, such as natural disasters. Bangladesh can be seen as one of the most vulnerable countries to the effects of climate change (Kulp & Strauss, 2019). These frequent natural disruptions can cause large problems in transportation networks (Jenelius & Mattson, 2015). Infrastructural policies and projects can play a big part in increasing the country's resilience, making it less vulnerable to the effects of climate change (Ministry of Environment and Forests Government of the People's Republic of Bangladesh, 2008).

Bangladesh's economically critical roads, the N1 and the N2, as well as national roads over 25km long that cross them, will be analyzed in this report as a transportation network. Strong transportation networks enable the movement of goods and people, which results in socioeconomic development and increasement in resilience (He et al., 2020). Therefore, identifying the strengths and weaknesses in terms of criticality and vulnerability of this road network can be of great value to policy-making regarding Bangladesh's transport infrastructure.

To analyze this, the concepts of *vulnerability* and *criticality* need to be defined. With these definitions, both concepts can be translated into operational metrics suitable for modelling the Bangladesh transport infrastructure network. The network is constructed by generating a Mesa model and a NetworkX model, the former relying on the latter. Multiple scenarios will be introduced to analyze the network, using different experiments where network and traffic data will be assessed in terms of criticality and vulnerability. Based on the experimental results, insights will be translated into specific policy recommendations.

# 2 Criticality and Vulnerability

The concepts criticality and vulnerability may seem to have obvious definitions, but in terms of transport infrastructures or networks, it is necessary to concretely define how these concepts will be assessed.

## 2.1 Criticality and importance

Jenelius & Mattsson (2015) define criticality in road networks as the combination of importance and disruption probability of each element of the network. Importance is defined as the impact of the disruption of a given element (Jenelius & Mattsson, 2015; Nicholson and Du, 1994): the greater the impact of a disruption, the higher the importance of the element. This paper assumes that this importance therefore is dependent on how busy certain network components (roads and bridges) are in terms of traffic. Criticality adds the *probability of disruption* element to this definition, by taking into account the chance that certain network components (in our case only bridges) will break down and cause delays.

## 2.2 Vulnerability

Jenelius & Mattsson (2015) define vulnerability in road network systems as society's risk of transport disruptions and degradations. They argue that vulnerability does not necessarily focus on the transport network in its entirety, but rather on the use of roads, bridges and network components by people, services and businesses. Therefore, this research will assess vulnerability from the user perspective rather than the component perspective. When analyzing vulnerability, the different aspects of disruption impacts for users need to be compared and aggregated under multiple scenarios (Jenelius & Mattson, 2015). Therefore, the probability of bridges breaking down will be taken into account when assessing vulnerability in this research. Not taking this into account would lead to the concept 'exposure' (Jenelius & Mattson, 2015), which will be left out of our analysis.

## 2.3 Utilitarian and Egalitarian perspectives

Jafino et al. (2020) make a distinction between egalitarian and utilitarian perspectives, where the utilitarian view takes into account that not every element is equal in a network when analyzing vulnerability. Thus, when more people live in a certain area, this needs to be taken into account when assessing the vulnerability of the components within or surrounding that area. The egalitarian perspective does not take this into account and sees the aforementioned as equal. Both perspectives are relevant for analysis in terms of vulnerability, therefore the concept vulnerability is split up into Utilitarian vulnerability and Egalitarian vulnerability.

# 3 Operationalization of metrics

Using the definitions for criticality and vulnerability, this section will operationalize these concepts into measurable outputs in terms of the Bangladesh N1-N2-sideroads network model. Based on the aforementioned concept definitions, the following operationalised definitions were derived:

***Importance*** *= traffic flow per bridge*

To be able to measure the traffic flow per bridge, the number of trucks that pass each bridge during the model simulation will be measured, regardless of scenario and probability of bridges breaking down. This is also done for all links.

***Criticality*** *= importance * probability of bridge breaking down*

For measuring criticality, the probability of the bridges breaking down per scenario will be taken into account. Multiplying importance with the probability results in measuring the highest delay caused by a bridge in each scenario, this will not be done for links.

***Utilitarian vulnerability*** *=*

*Absolute delay of each source * probability of bridge breaking down*

*OR*

*Absolute delayed trucks of each source * probability of bridge breaking down*

Sources are the origin of the vehicles that drive in the transport network. They can be assumed to represent towns or population hubs. Bigger towns in terms of population are more important in the utilitarian perspective (Jafino et al., 2020). For each source, the total delay time of the trucks it generates in every scenario will be counted. This will be the first type of utilitarian vulnerability. For a second type of utilitarian vulnerability, the number of trucks from each source that experience a delay (cumulative) is counted.

***Egalitarian vulnerability*** *=*

*Relative delay of each source * probability of bridge breaking down*

*OR*

*Proportion delayed trucks of each source * probability of bridge breaking down*

This is similar to the utilitarian vulnerability explained above. However, the **average** delay of trucks from the same source and the **proportion** of trucks from the same source being delayed are taken into account. This removes population size from the importance of a source, thus making all cities equal.

The conceptualization of the definitions with their associated model measurements are graphically portrayed in Figure 1.

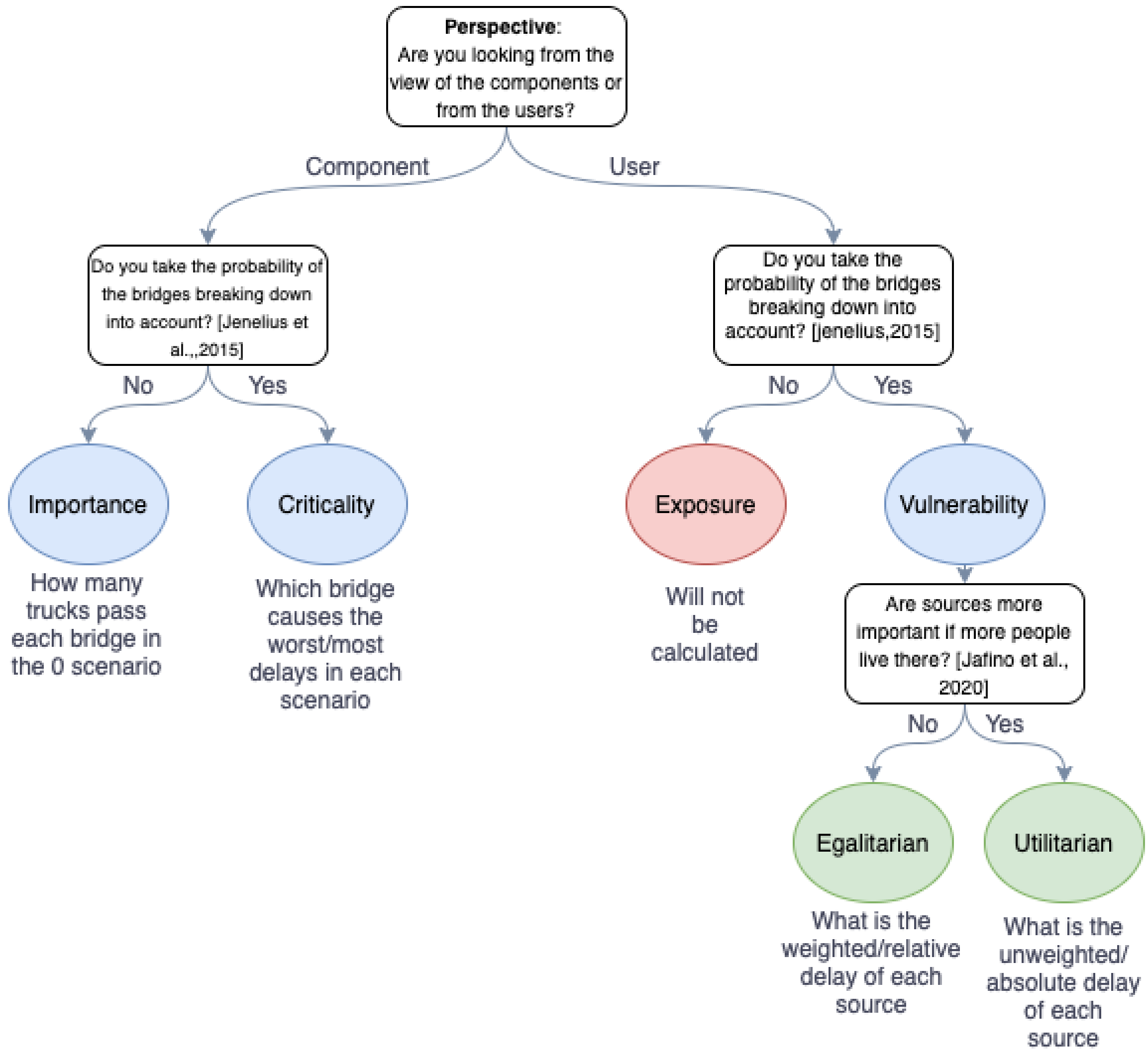



*Figure 1: Conceptualization of criticality and vulnerability with associated metrics*

In order to further the scope of the study, the betweenness centrality of each bridge in regard to each city will also be calculated. This allows for a study of the importance of bridges for each city of the network. This will be done by computing the betweenness centrality of all bridges on a subset of paths for each source. The subset contains the city of interest (sourcesink) as a starting point, all other sourcesinks as destinations, and all paths joining the source to the destinations. This gives the betweenness centrality for every bridge looking from one specific source. Multiplying this with the acquired egalitarian and utilitarian metrics allows us to determine the most vulnerable bridges.

# 4 Data Preparation & Model Design

In order to run experiments, the data available to us in the roads and the bridges datasets (*_roads3.csv* and *BMMS_overview.xlsx*) needed to be cleaned and the model needed to be expanded.

## 4.1 Cleaning process

The cleaning process is done in the file *G22-04-data_processing.py*, using functions defined in *G22-04-data_processing_components.py*. The inputs of the functions may be changed according to a user's needs, thus ensuring a generalised cleaning process.

The data cleaning consists of three steps, each based on one newly-defined function. The datasets are filtered for relevant information and a first DataFrame with all elements of all relevant roads is created. That DataFrame is then modified to add intersections and traffic information where necessary.

Creating a DataFrame that will allow for intersections to be defined in the second step is done by the function *get_bridge_road_data*. It filters the datasets mentioned above for elements of the roads relevant to the study, merges the filtered data of both datasets into a single dataset, gives each element a unique id and finally assigns an agent subclass sourcesink, bridge or link to each of these elements. The inputs of the function are:

- the datasets for roads and bridges information (in our case, *_roads3.csv* and *BMMS_overview.xlsx)*
- the "main" roads (in our case, N1 and N2), or the roads of which the side- and crossroads must be considered
- the type of side- and crossroad one must take into account (in our case, only N-Roads)
- the minimum required length in km of a side- or crossroad to be included in the study (in our case, 25).

The function's output is a pandas DataFrame with all elements of all relevant roads to the study, all of which have a unique id and an agent type for the model (sourcesink, link, bridge).The id and the model_type may be changed in the subsequent steps.

The addition of intersections is done by the function *add_intersections*. It takes the output of the previous function as input. In order to find and create intersections, it goes through the input DataFrame, searches documented intersections (column "type" from the original datasets indicating a "CrossRoad" of a "SideRoad") and creates them (i.e, changes "model_type" to "intersection" and gives the intersection the same id on both roads) at the corresponding points if the intersection is documented on both concerned roads. This process doesn't find all intersections due to an incompleteness of the original datasets. There are thus two further mechanisms to find intersections that may be missing. If the intersection is documented on one road only, then the second road of that intersection is scanned for its closest point to the documented intersection using euclidean projection. That point is then partially overwritten by the documented intersection, thus creating an intersection. If all else fails, the entire DataFrame (bar bridges and existing intersections) is searched through for points of different roads that are less than 100m away from each other using euclidean projection.

Traffic data is added with the function *add_traffic_.* This function takes two inputs: the output of the previous function and the path to a folder containing all *.htm* files relevant to this study, namely the *X.traffic.htm* files, where X stands for the name of each road relevant to this study. Due to our experimental design (see relevant section for details), each sourcesink must be given an outgoing and an incoming traffic. More specifically, outgoing traffic is the proportion of total trucks that must be generated at the sourcesink, and incoming traffic is the proportion of total trucks that must have that sourcesink as destination. The absolute traffic numbers of each sourcesink are taken from the corresponding *X.traffic.htm* file using their road to pick the relevant file and their chainage to find the relevant row. Once the absolute numbers of outgoing and incoming traffic are found for all sourcesinks, they are used to calculate the total number of trucks on the road and are then overwritten with the proportion of trucks created at and sent to each sourcesink. The output of this function is the final Dataframe.

All three functions are called sequentially in the *G22-04-data_processing.py* file, and the resulting DataFrame is saved to the *N1_N2_plus_sideroads.csv* file. This process can, due to the defined functions, easily be repeated for other roads and types of side- and crossroads.

## 4.2 MESA model design

A few changes needed to be made to the model from the last assignment in order to make traffic more realistic: different sources must create different numbers of trucks, and destinations cannot be randomly assigned anymore. For the former, the column “out” of the input file is used for each source as a probability of creating a truck at each time step. The change was made in the *Source* class in the *components.py* file. For the latter, the *get_route* function in the model (*model.py* file) was changed: destinations are still randomly assigned but now with a weighted probability, based on the “in”column of the input file.

Furthermore, in order to retrieve the necessary information after the experiments, a new function *compute_traffic* was defined in *model.py*. This function is used as a model reporter in the batchrunner. It returns eight characteristics of the model, as dictionaries, all of which are of importance for the evaluation of results:

- *generated*: the number of trucks created at each source during the run. **This is used as population size**.
- *removed*: the number of trucks that reached each sink during the run
- *delay_time_abs*: the accumulated delay of all trucks from the same source, for all sources. **This is used as one type of utilitarian vulnerability**.
- *delay_time_rel* : the average delay of a truck created at each source. **This is used as one type of egalitarian vulnerability, dividing the relevant utilitarian vulnerability by the population size**.
- *delay_freq_abs*: the number of trucks from each source that were delayed (regardless of the length of the delay). **This is used as the other type of utilitarian vulnerability**.

- *delay_freq_rel*: the proportion of trucks from each source that were delayed (regardless of the length of the delay). **This is used as the other type of egalitarian vulnerability, dividing the relevant utilitarian vulnerability by the population size**.
- *traffic_bridges*: the number of trucks that passed each bridge. **This is used for the importance and criticality of bridges**.
- traffic_links: the number of trucks that passed each link. **This is used for the importance of links**.

These outputs are all based on the following model and agent attributes, of which some already existed in the demo model and others were added:

- model attribute *self.sources*: a list of the ids of all sources and sourcesinks in the model
- model attribute *self.sinks*: a list of the ids of all sinks and sourcesinks in the model
- model attribute *sinks_in:* a dictionary using the id of a sink or a sourcesink as key and the proportion of trucks going towards that sink or sourcesink as value.
- model attribute *self.schedule.agents*: the scheduler, used as a list of all agents of the model
- attribute *generated_traffic* for *Source* and *SourceSink* agents: the number of trucks each source or sourcesink creates over the run
- attribute *removed_traffic* for *Sink* and *SourceSink* agents: the number of trucks each sink or sourcesink removes from the model over the run
- attribute *waiting_times* for *Source* and *SourceSink* agents: total number of waiting times that all trucks generated by that agent experienced. Truck specific waiting times are added to this number each time a truck reaches the destination, before it gets removed from the model.
- attribute *waiting_freqs* for *Source* and *SourceSink* agents*:* total number of how many times each truck generated by that agent was delayed on its route. Truck specific waiting occurrences are added to this number each time a truck reaches the destination, before it gets removed from the model.
- attribute *trucks_passed* for *Bridge* and *Link* agents*:* number of trucks, regardless of origin or destination, that pass each *Bridge* or *Link* agent. For *Bridge* agents, this number increases by 1 each time the *get_delay_time* function is called. For *Link* agents, this number increases by 1 each time the agent is the next point the truck is driving towards (called *next_infra* in the *Vehicle* agent class)

Changes in model attributes are done in the file *model.py*, changes in the agent attributes are done in the *components.py* file.

## 4.3 Summary

In order for traffic to be represented more realistically, the input file needs to have information about traffic going to and from all sourcesinks. This is done in the data cleaning. The traffic information was added as the probability of any truck to be generated at each sourcesink (outgoing traffic) and as the probability of each sourcesink to be assigned as a destination (incoming traffic). These probabilities are normalized over all sourcesinks.

In order to make the most of this information, the relevant agent classes and model functions needed to be modified:

- Each minute, each source generates ten random numbers between 0 and 1 and creates as many trucks as there are generated numbers below the specified probability of a truck being generated at this source. Generating random numbers instead of only one ensures that a high amount of trucks is created.
- Every truck picks its destination based on the probability of each sourcesink to be assigned as a destination. This is done by saving all the sinks to a dictionary with the specified probability as values. In the get_route function, a truck now picks a sink as a destination from this dictionary. In order to get the right distribution, the random.choices() function is used, weighted by the probabilities as specified in the dictionary.

# 5 Experimental Set-Up

Due to the changes made in the model, which affect how many trucks are created at each source and how destinations are assigned, the traffic on the N1, N2 and their sideroads is modeled more realistically than in the previous assignment. In order to assess the criticality of bridges and the vulnerability of cities/bridges, it is essential to have a realistic probability of breaking down for each bridge category. Since there is no data available on how each bridge category behaves, not even in relation to one another, the model is run under different scenarios, each scenario with a different probability of breaking down for each bridge category. All scenarios are shown in Table 1. Every scenario was run 30 times.

*Table 1: Scenarios used in the experiment*

| Scenario | A | B | C | D |
|---|---|---|---|---|
| 0 | No delays | No delays | No delays | No delays |
| 1 | 10% | 10% | 10% | 10% |
| 2 | 10% | 15% | 20% | 25% |
| 3 | 10% | 20% | 30% | 40% |
| 4 | 10% | 15% | 25% | 40% |

Scenario 0 is a control, where no bridges break down. This represents a perfect day. Scenario 1 assumes that all bridges have a 10% probability of breaking down, thus blurring differences between bridge categories and rather focussing on the traffic that passes by them. In scenarios 2 to 4, bridges have a higher probability of breaking down if they belong to a lower-quality category. The difference in breakdown probabilities between bridge categories becomes bigger in each consecutive scenario. In scenario 2 and 3, the difference in breakdown probability is the same from one category to the next, while it doubles in scenario 4.

# 6 Results

In this section the results will be presented and discussed. First a general output metric, *Average driving time* per scenario will be presented, for interpretation and verification purposes. Afterwards, the model output in terms of the pre-defined network metrics *importance, criticality* and *vulnerability (egalitarian* and *utilitarian)* will be shown.

## 6.1 Average driving times per scenario

In figure 2 it can be seen that the driving times have no significant difference per scenario, except for scenario 0.

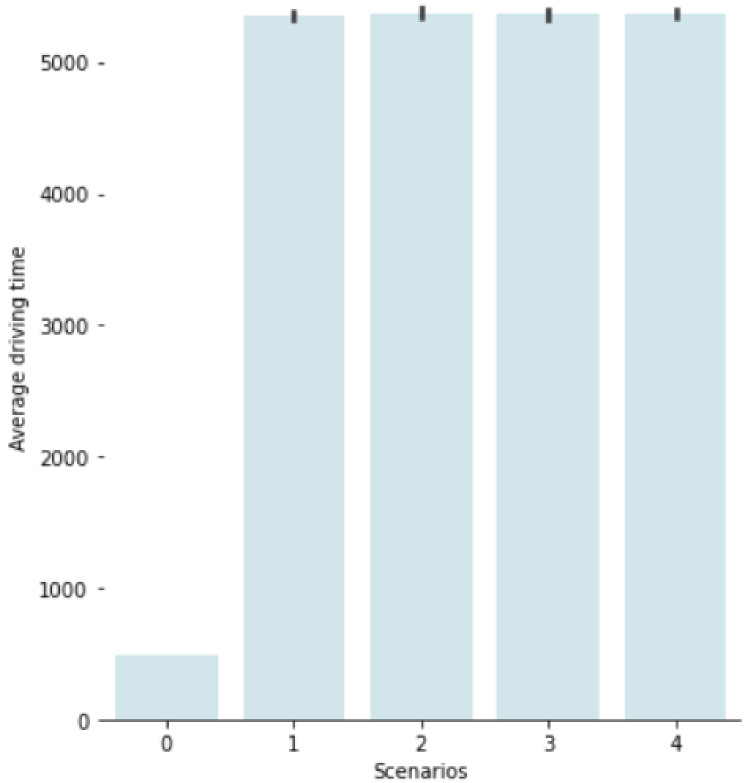


*Figure 2: Average driving time over scenarios*

The driving times for scenarios 1 through 4 are not exactly equal to each other, however they show very little difference. This result does not correspond with the behavior which was initially expected, because scenarios influence breakdown probabilities drastically and this should have affected the driving times. However, when observing the data, it can immediately be seen what causes this behavior (Table2).

*Table 2: Occurrences of bridges in category*

| ***Condition*** | ***#occurrences*** |
|---|---|
| A | 970 |
| B | 249 |
| C | 179 |
| D | 13 |

The vast majority of the bridges in the network are category A and the other bridge categories get somewhat 'overshadowed' by the presence of category A bridges in the network. This implies that delays for A-bridges mostly determine the output of the scenarios and because this bridge category's breakdown probability doesn't get varied in the experiments (10% for all scenarios 1-4), the average driving times per scenario are thus very similar.

What this means for the rest of the results is that there are no significant differences in terms of delays (traffic flows) for the scenarios, meaning that all scenarios will be analyzed as one for the criticality and vulnerability analysis. Scenario 4 is used for the resulting output in the following sections, the other scenario outputs can be seen in the *Data_Analysis.py* notebook

## 6.2 Importance (for scenario 0)

In scenario 0, in every run, the bridges occur a certain amount of times. The bridges that are the most important, averaged out over all runs, are listed in Table 3 and their location within the network is visualized in Figure 3.

*Table 3: Importance*

| ***Bridge*** | ***Importance*** *(#occurrences in scenario 0)* |
|---|---|
| Corepai Box Culvert | 45726.2 |
| Nimshar Slab Culvert | 45725.3 |
| Tipra Bazar Box Culvert | 45724.3 |
| Kabila Dubarchar Slab Culvert | 45723.8 |
| Cori Pai Box Culvert | 45723.1 |

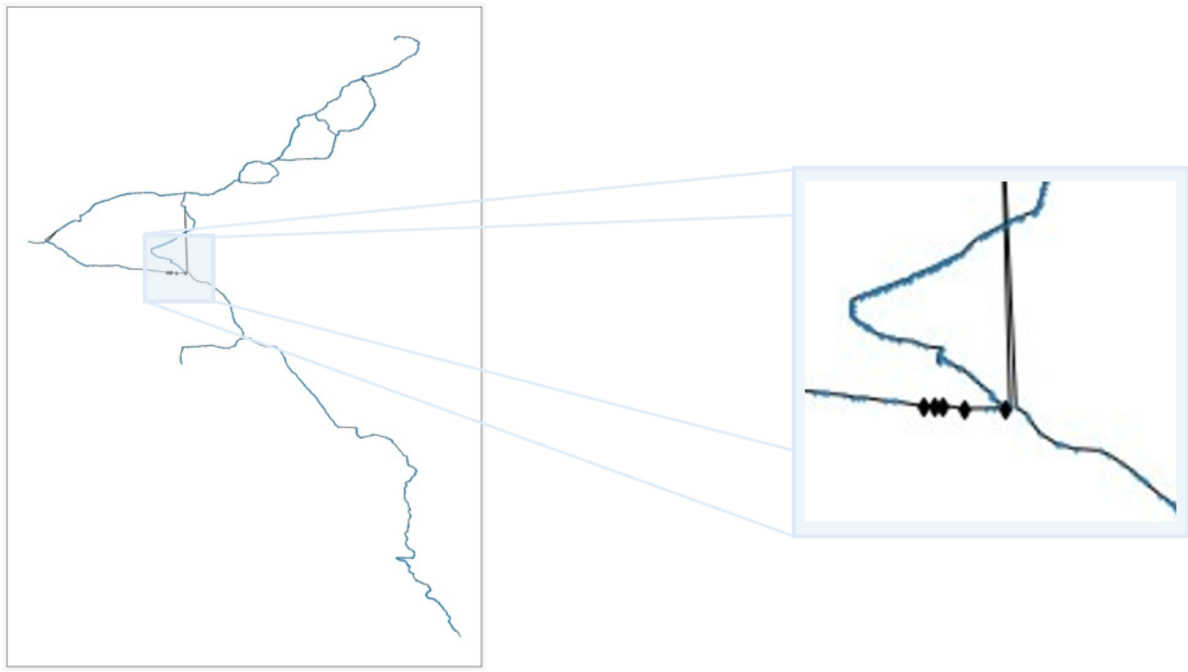


*Figure 3: Important bridges on map*

The bridges shown in table 3 and figure 3 represent the bridges that get passed the most times in the simulation of the network, therefore having the highest traffic flow. These bridges could perhaps be deemed as critical if these bridges also have high breakdown probabilities, which will be investigated in the next subsection

## 6.3 Criticality

The critical bridges are defined as bridges that are important and have high breakdown probabilities. This is translated into bridges that cause the most and/or worst delays. They are depicted in Table 4 and visualized in Figure 4.

*Table 4: Criticality*

| **Bridge** | **Worst Bridge Delay (#)** |
|---|---|
| Narayan Pur Rcc Gider Bridge | 31 |
| Mundabad Box Culvert | 19 |
| Lala Bazar Start | 13 |
| Narayerpur Culvert | 13 |
| Lala Bazar | 9 |

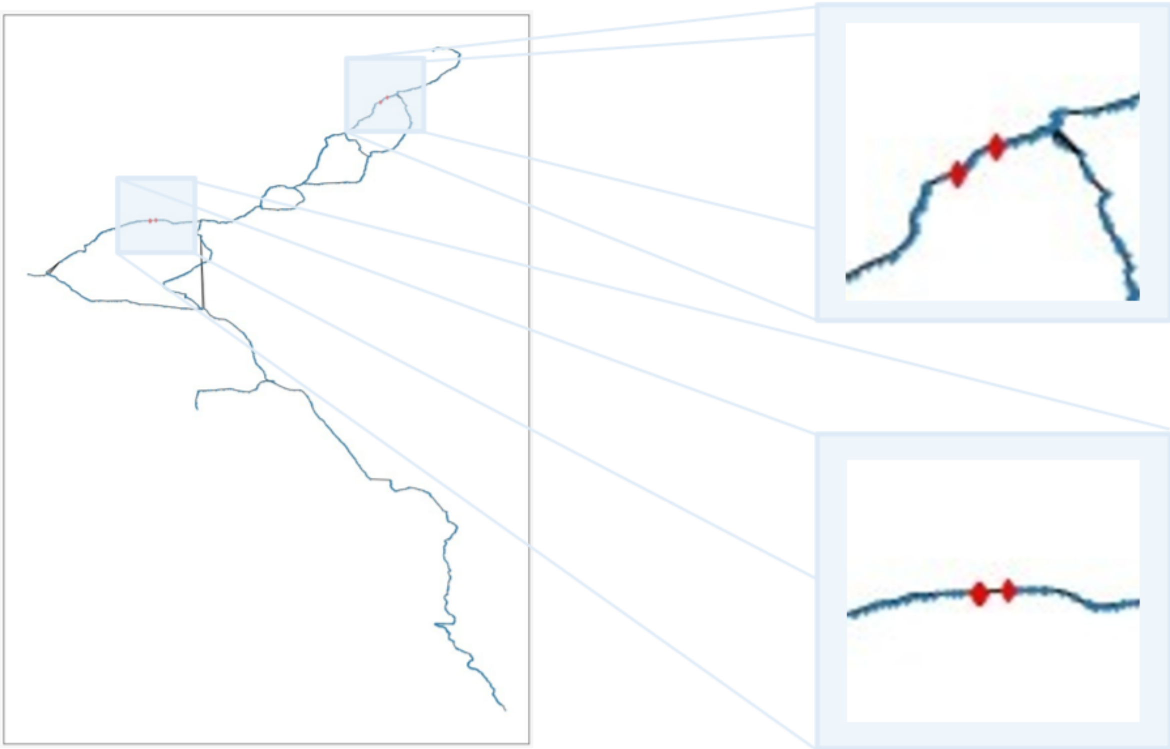


*Figure 4: Critical bridges on map*

What can directly be observed is that these bridges do not correspond with the bridges that have the highest importance.

## 6.4 Utilitarian Vulnerability

This metric gives information on the most vulnerable bridges seen per city/sourcesinks if all cities are seen as unequal. The unweighted absolute delay of each source is multiplied by the partial betweenness centrality of that specific sourcesink/city, towards all bridges, which results in the most vulnerable bridges per city. This is determined based on the frequency of delays (figure 5) and the delay times (figure 6) per sourcesink/city

**Frequency**: how often do trucks coming from that sourcessink/city cause a delay, absolute per city

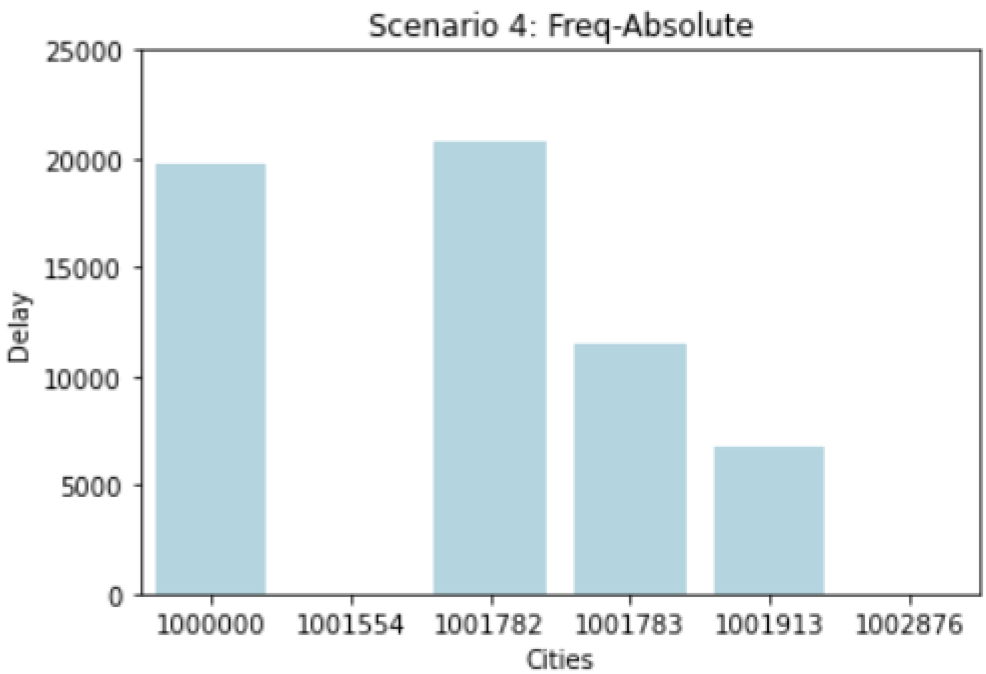



*Figure 5: Scenario 4: Absolute Frequency*

**Delay time absolute**: how much delay time is caused by the bridge breakdowns for that sourcesink, per city

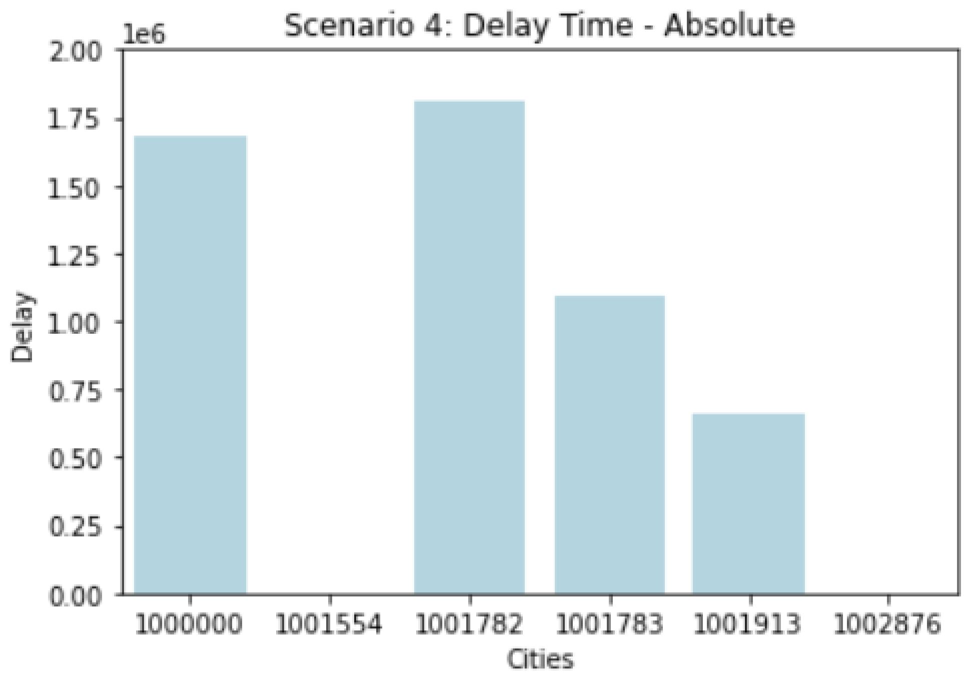



*Figure 6: Scenario 4: Absolute Time*

What can directly be observed is that even though the values for delay time and frequency obviously differ, the relationship the cities have with each other is similar. This seems logical, the frequency of experienced delays correlates highly with the absolute delay time. More delays often means a longer total delay time.

## 6.5 Egalitarian Vulnerability

This metric gives information on the most vulnerable bridges seen per city/sourcesinks if all cities are seen as equal. The weighted relative delay of each source is multiplied by the partial betweenness centrality of that specific sourcesink/city, towards all bridges, which results in the most vulnerable bridges per city. This is determined based on the frequency of delays (figure 7) and the delay times (figure 8) per sourcesink/city

**Frequency**: how often do trucks coming from that sourcessink/city cause a delay (relative to the size of the sourcesink/city)

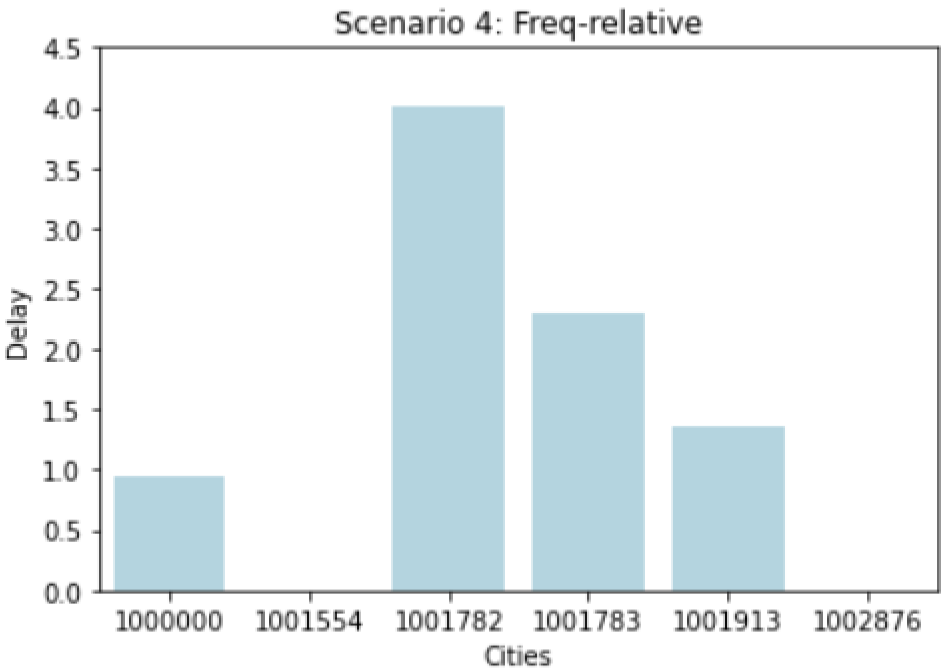



*Figure 7: Scenario 4: Relative Frequency*

**Delay time relative** (how much delay time is caused for that sourcesink, relative to the size of the population in that sourcesink/city)

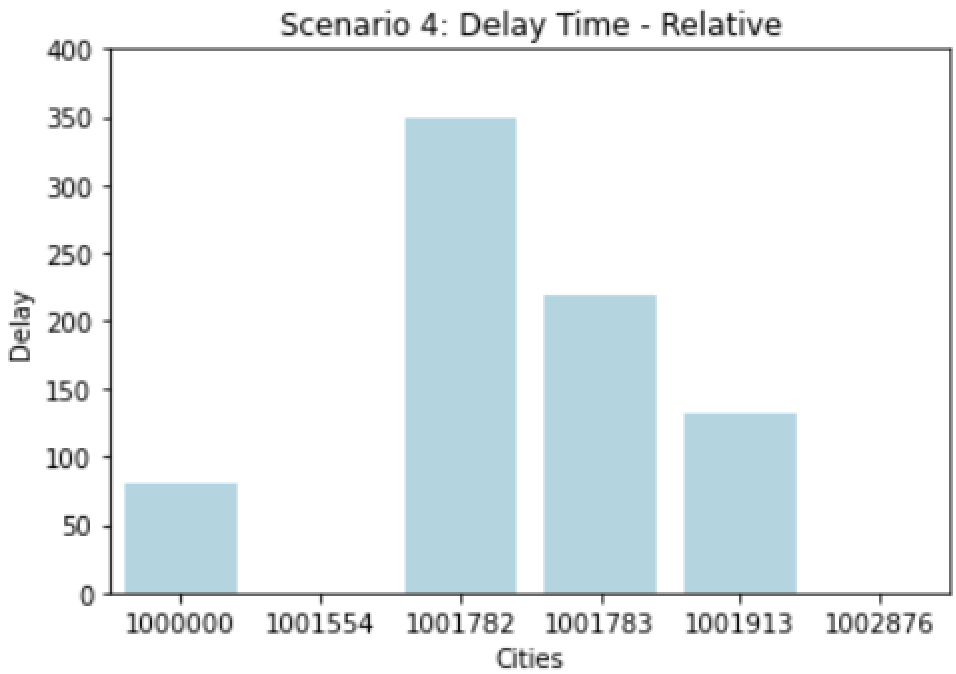



*Figure 8: Scenario 4: Relative Time*

The same phenomenon as mentioned in the previous paragraph occurs here.

## 6.6 Vulnerability

The betweenness centrality metric is necessary to identify the most vulnerable bridges for each sourcesink/city and combining this with the results from the previous two sections enables the identification of the most vulnerable bridges.

The weighted value with the betweenness centrality, for all metrics used for vulnerability (egalitarian and utilitarian, in terms of frequency and delay time) results in the values in Table 5. Table 6 gives the top seven of bridges. Because the seven bridges had the same values, it is chosen to include a top seven instead of a top five. The locations of the bridges in the network are visualized in Figure 9.

*Table 5: Values for variables (same output for egalitarian and utilitarian)*

| ***Measure*** | ***Scenario 1*** | ***Scenario 2*** | ***Scenario 3*** | ***Scenario 4*** |
|---|---|---|---|---|
| Absolute Delay Time | 1.648115 | 1.635759 | 1.639791 | 1.639803 |
| Relative Delay Time | 0.000318 | 0.000316 | 0.000317 | 0.000317 |
| Absolute Delay Frequency | 0.019001 | 0.018851 | 0.018889 | 0.0188888 |
| Relative Delay Frequency | 0.000004 | 0.000004 | 0.000004 | 0.000004 |

Table 6: Vulnerable bridges (same output for egalitarian and utilitarian)

| ***Bridge*** |
|---|
| Molabi Bari Culvert |
| Ghatara Bridge |
| Suhil Pur Box Culvert |
| Suhilpur Mirhati Box Culvert |
| North Portala Bridge |
| Uttor Shohil Pur Culvert |
| Bathbaria Box Culvert |

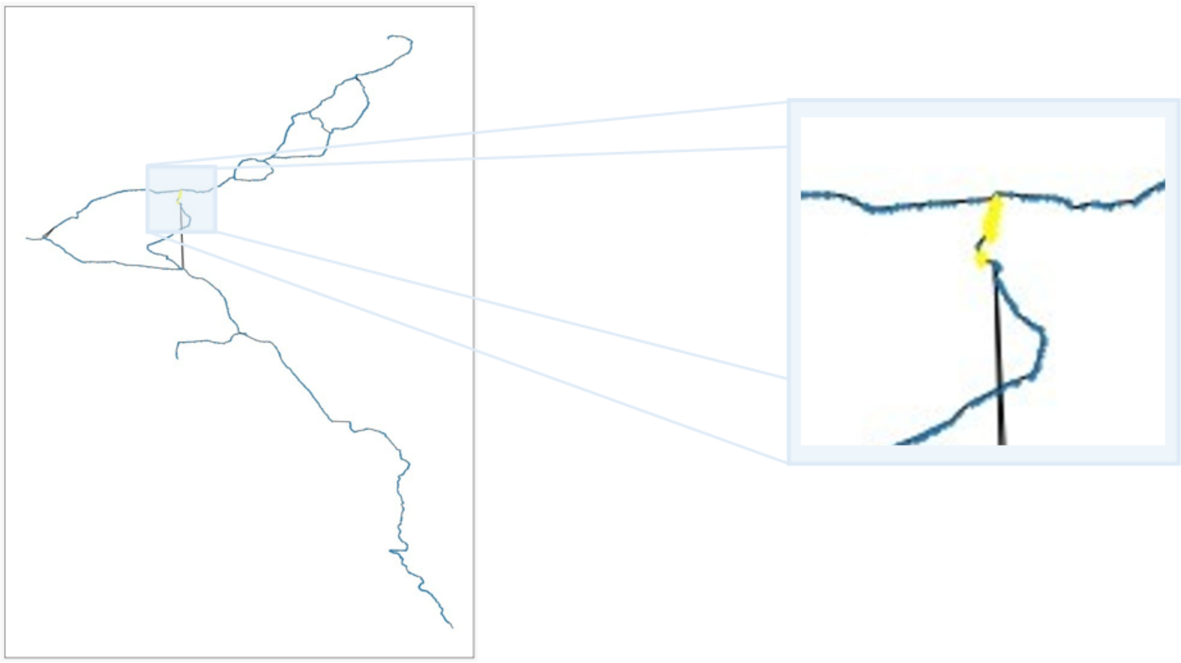


Figure 9: Vulnerable bridges on map

What stands out is that there is  no difference in vulnerable bridges between the egalitarian perspective and the utilitarian perspective. Both metrics generate the same bridges as most vulnerable.

# 7 Conclusion & Discussion

This chapter discusses the conclusion of the results and reflects upon them.

## 7.1 Importance

In terms of importance, this research has identified the bridges that have the highest traffic flow. These bridges however, do not correspond with the identified critical bridges. Therefore, these bridges should be deemed as ‘very relevant in this network’ by the Bangladesh government and should be closely monitored in terms of breakdown probability (which is assumed to be linked to bridge quality). In future scenarios, if these important bridges decrease in quality and start causing delays, this could have major implications for the entire network, as these bridges then become highly critical. The bridges with the highest importance for the network are:

1. Corepai Box Culvert
2. Nimshar Slab Culvert
3. Tipra Bazar Box Culvert
4. Kabila Dubarchar Slab Culvert
5. Cori Pai Box Culvert

## 7.2 Criticality

The most critical bridges cause the most impactful delays in this network. The transportation of goods and people in this network depend largely on the quality of these identified bridges, which is not sufficient at this moment. Local policy makers should consider renovating these bridges so that the flow of traffic in this network will be more fluent and less delay is experienced in the network as a whole. The bridges with the highest criticality for the network are:

1. Narayan Pur Rcc Gider Bridge
2. Mundabad Box Culvert
3. Lala Bazar Start
4. Narayerpur Culvert
5. Lala Bazar

These bridges are closely located to each other, meaning that the logistics of the renovation investments should not be too difficult.

## 7.3 Vulnerability

The most vulnerable bridges are researched in terms of egalitarian and utilitarian vulnerability. The outcome is that in the bridges taken into account, there is no difference between the bridges to repair for either egalitarian purposes or utilitarian purposes. The bridges with the highest vulnerability are:

1. Molabi Bari Culvert
2. Ghatara Bridge
3. Suhil Pur Box Culvert
4. Suhilpur Mirhati Box Culvert
5. North Portala Bridge
6. Uttor Shohil Pur Culvert
7. Bathbaria Box Culvert

As Bangladesh's resilience is concerningly low, it is essential that the government invests in the above mentioned bridges. Increasing Bangladesh's resilience, especially in its constant battle against the devastating effects of climate change, is a necessity that should have top priority. What this research has shown is that egalitarian and utilitarian perspectives do not seem to make significant differences on the N1-N2 network in terms of vulnerability. This will be further discussed in the reflection.

Interestingly, all measures (importance, criticality and vulnerability) output different bridges as the worst bridges.

## 7.5 Reflection

Two things that stand out in this paper is that there is no significant difference in the ranking for egalitarian and utilitarian bridges and there are very little differences in delays across scenarios. Both remarks are probably due to the number of roads looked at. The model ignores most cities in Bangladesh. Maybe those cities would have collectively (or even individually) tipped the scales and made some bridges more important, maybe even more critical in different scenarios. Furthermore, only important roads (N1, N2 and their N-sideroads) are taken into account. This causes them to have  mostly good bridges, thus making the bridge analysis not so interesting, and it makes the study inherently utilitarian. A study that is remotely egalitarian would need the inclusion of more, less important roads. More boldly stated, this paper currently only looks at the most important cities of Bangladesh (Dhaka, Chittagong) and at its most important road (N1). These cities already have much more resources and opportunities than all other cities, but only host about 10% of the population. Any egalitarian study should include more cities, or at least Khulna and Rajshahi, the third and fourth biggest cities of Bangladesh (Bangladesh Bureau of Statistics, 2018).

Lastly, it should be taken into account that every result in this research is in some form relative. This means that the identified most vulnerable, critical and important bridges are all 'the most' vulnerable/critical/important compared to the rest of the bridges in the network. Thus, if the recommended policy measurements would be implemented and the model is run again, new worst bridges in terms of the metrics will be outputted and the same recommendations would follow for new bridges. Exactly how much more vulnerable/etc. the most vulnerable/etc. bridges are compared to the other bridges in the dataset, is difficult to say because of the definition and operationalization of the metrics in this research.

# 8 Limitations

- A large limitation of this research is the experimental set-up in relationship to the amount of bridges per category present in the cleaned dataset. Category A bridges somewhat overshadow the other bridges (B, C, D). This resulted in a limited, almost non-existing impact of the different scenarios on the simulation. All results presented are those of scenario 4, but these are almost entirely the same for scenario 1, 2 and 3. In hindsight, with the knowledge of the distribution of bridge categories being skewed towards category A, the experimental set-up should have entailed varying breakdown possibilities for category A bridges as well, as it was now 10% for each scenario.

- The network, as specified in the model, is still not 100% correct. Some outliers in the location of some point are still present. This could influence shortest path calculations that determine what route trucks take.

- Traffic is generated according to acquired traffic distributions. While the fractions of traffic at end and begin points of roads are close to reality, the exact amount of generated trucks doen not.

- The use of the betweenness centrality metric, for measuring the vulnerability of the bridges, implicates the assumption that every truck/person has the desire to drive towards every bridge equally. This might be a very impactful assumption and future research could experiment with weighted betweenness centrality metrics.

# 9 References

Bangladesh Bureau of Statistics. (2018, April). STATISTICAL POCKET BOOK BANGLADESH 2017. *Ministry of Planning.* http://bbs.portal.gov.bd/sites/default/files/files/bbs.portal.gov.bd/page/d6556cd1_dc6f_41f5_a766_042b69cb1687/PocketBook2017.pdf

Jenelius, E., & Mattsson, L. G. (2015). Road network vulnerability analysis: Conceptualization, implementation and application. *Computers, Environment and Urban Systems*, *49*, 136-147.

Kulp, S. A., & Strauss, B. H. (2019). New elevation data triple estimates of global vulnerability to sea-level rise and coastal flooding. *Nature Communications, 10(1)*. https://doi.org/10.1038/s41467-019-12808-z

Ministry of Environment and Forests Government of the People's Republic of Bangladesh. (2008). Bangladesh Climate Change Strategy and Action Plan. ISBN 978-984-8574-25-6.

He, Y., Thies, S., Avner, P., & Rentschler, J. (2020). The Impact of Flooding on Urban Transit and Accessibility.

Huang, Y. (2013). Automated Simulation Model Generation, *Delft University of Technology*