# Report EPA1352 Advanced Simulation Assignment 2

Christiaan Ouwehand (4293053), Ewout ter Hoeven (4493346), Marlou Ceha (4691539), Timo Frazer (4579992), Zoé Huizing (4660455)

## Introduction

The goal of this lab exercises is to analyse the transportation delays and its economic impact due to breakdown of bridges on the N1 from Chittagong to Dhaka in Bangladesh. For that, a Mesa module 0.9.0 with Python 3.10 will be used to generate trucks driving every 5 minutes from the beginning to the end of the road crossing bridges that could break under different probabilistic scenarios. To compare these results, a minimal discreet simulation model will be made.

## Data preparation

The ''WEBSIM_Lab1_cleanedDataset'' has been used to prepare the data. On that data frame the following modifications were made to make it suited for an analysis for the N1. First all roads other than the N1 are deleted. Every bridge length will be categorized to assign them a potential delay (see Table 1).

Several bridges are double in the data frame, this is because they have a left and right road. To avoid adding double roads in the data every bridge with a right variant will be removed. The left bridges have been used in this model because they drive on the left in Bangladesh. Also, the distance between every bridge is calculated to be able to make road length later in the model. Then, every bridge that is closer to the previous bridge than its own length is also removed. This has been done to remove very small bridges and bridges with a length of 0.

To make the model run correctly the data file needs to be fit in the same format as the demo CSV. For that, a column 'model_type' is added to make a distinction between the roads and the bridges. The location (latitude and longitude) of the roads has been located as the middle point of both ends of the roads. The roads have been added by the length of the roads with their latitude and longitude.

# Methodology

## Design of the Mesa model

To implement the delay time of a truck in the different bridge classes a definition *get_delay_value* is created. The diverse types of bridges are categorized as follows:

Table 1: delay per bridge from each category

| Bridge Length | Category Name | Delay time for a truck |
|---|---|---|
| Over 200 m | XL | Triangular (1,2,4) hours |
| Between 50 and 200 m | L | Uniform (45,90) minutes |
| Between 10 and 50 m | M | Uniform (15,60) minutes |
| Under 10 m | S | Uniform (10,20) minutes |

To get the delay time all in the same unit, the delay time of category XL was converted from hours to minutes, resulting in Triangular (60,120,240) minutes. A dictionary was used to obtain the delay times for the different bridge categories.
In the model, every bridge also has distinct categories of conditions for the bridges. If a bridge is in a good condition, it will be categorized as A. The bridges in the worst condition will be categorized as D.

To get a good overview of the economic impact of critical infrastructures, scenarios have been added to the model in the model.py file, by reading a csv file with the different scenarios, an overview of these scenarios is given by Figure A3 from the Appendix.

In the components.py file multiple things have been added. In the bridges class the chance that a bridge gets broken is added. It looks at the chance a particular bridge of a certain condition breaks down in that scenario. If this chance is bigger than the value of a random uniform distribution between 0 and 1, the bridge breaks down and the trucks get delayed.
Also, the speed from a truck is set to 48 kilometres per hour in the model.

To generate the model and the different scenarios, a dictionary is used with the break-down chance for each bridge type. Also, in the file experiments.py the setup of the different scenarios is written. The model is run 5 times over 24 hours with 1 tick per minute. The data from the different scenarios is stored using a pickle file (Python Software Foundation, 2022).

**Limitations of the Mesa Model**

The Mesa model contains many limitations, caused by its discrete approach of an analytical problem. A time step is defined, by default 1 minute, which causes rounding errors throughout the model. If a truck arrives at its destination, no matter where in the minute, it will immediately start driving for another minute, making it drive earlier than it should. But if it does not have to wait, it spends the rest of the minute doing nothing, only after which it will start driving to the next location.

A discrete-event or analytical model wouldn't pose those limitations, since those create event when a truck is expected to arrive or don't acknowledge time anyways, respectively. This way both models would be more accurate and at the same time faster than an agent-based model with a fixed time step.

Benchmarks emphasize this (huge) performance limitation of the Mesa model. Running the Mesa model only once for all 9 scenarios takes about 115.3 seconds. This means if we want to do 125 replications, this is expected to take about around 4 hours.

## Design of the minimal model

Because of the limitations described above, mainly the low performance and high complexity, we decided to develop a new, minimal model. This model solves the problem in an analytical way using much simpler formula's for calculating broken bridges and delays stochastically.

The minimal model can run all scenarios 10 times in about 13.11 seconds, using 1000 trucks per scenario. Since the Mesa Model took 115.3 seconds for all scenarios only once, this means the model is 88 times faster than the minimal model and thus can be ran 88 more times than the Mesa model in the same time span. 125 replications with 1000 trucks takes only about 2.7 minutes, compared to the 4 hours of the Mesa model. After iteration, 125 replications showed to be enough to get normal-like distribution for the average delay time for scenarios 3 to 8 (for scenario 1 and 2 see, the appendix).

Meanwhile the results are more accurate, since driving times of both the road segments (being exactly 571 minutes always for the roads, since a fixed speed of 48 km/h is assumed) and those of the not-broken bridges are calculated analytically once, and thus don't contain rounding errors, and only the delays of broken bridges are calculated from chances. Because of the fast performance, 125 replications were run for each scenario that delivered a broad array of sets of broken bridges, nicely distributed (see the Histograms in Figure A2. These results will be discussed in the next section.

The minimal model is available in minimal_model/minimal.py, and the benchmarking code is in minimal_model/benchmarks.py.

**Limitations of the minimal model**
Since a high number of replications can be easily ran, variations between random seeds are not a important limitation, but still they could be implemented to make the model perfectly distributable. Currently the bridges that are broken are only calculated once for each iteration. Functions that bridge break during the simulation could be implemented. Trucks have all a constant, identical speed, as for the assignment. In the real world this would not be the case, so a distribution of varying average speeds could be implemented.

Note that all those limitations also are present in the Mesa model, the minimal model doesn't contain additional limitations.

## Calculation of economic impact

Economic impact of delay in the trade between India and Bangladesh has been measured by Das & Phit in 2004. They estimate the economic losses on 2,5% of the value of the cargo per day of delay. The model will, for each scenario, calculate the average delay of all trucks in minutes. The economic losses will be calculated in percentage of the total value of the cargo of all the truck with Formula 1:

$$Economic\ loss = Delay_{time} \cdot 2{,}5 : 60 : 24 \quad (1)$$

# Results

Table 2 shows the average travel time and economic loss due to delay per scenario. The first column indicates the number of scenarios. In the model, also a 0 scenario has been added: the base case, where there are no probabilities that the bridges will break down. The second column gives the average delay time in minutes. Each row of the different scenarios represents the delay time plus the delay time of the base case scenario. The third column shows the 95% confidence interval. This gives the possible variation of the delay. The fourth column represents the average economic loss, see the methodology. The last column shows the variation in economic loss. It is the difference between the higher bound and lower bound of the confidence interval divided by the average delay.

Table 2: Average travel time and economic loss due to delay per scenario

| Scenario | Average delay time (minutes) | 95% confidence interval delay | Average Economic loss (in % of the cargo) | Variation in economic loss (%) |
|---|---|---|---|---|
| 0 | 571 travel time | - | - | - |
| 1 | +30.5 | (22.7, 38.2) | 0.05 | ∓ 50.9 |
| 2 | +70.6 | (60.2, 81.1) | 0.12 | ∓ 29.6 |
| 3 | +298.5 | (277.1, 319.9) | 0.52 | ∓ 14.4 |
| 4 | +596.9 | (570.6, 623.3) | 1.04 | ∓ 8.8 |
| 5 | +746.2 | (717.4, 775.0) | 1.30 | ∓ 7.7 |
| 6 | +1458.1 | (1418.1, 1498.1) | 2.53 | ∓ 5.5 |
| 7 | +1827.8 | (1783.3, 1872.3) | 3.17 | ∓ 4.9 |
| 8 | +3624.5 | (3571.5, 3677.4) | 6.29 | ∓ 2.9 |

From Figure A3 can be seen that every scenario has an increase in number of bridges that could break. This is also the case in the experiment, the delay time is increasing per scenario. A delay engenders an economic loss linearly increasing with the delay (Formula 1).

From figure A1 from the Appendix can be seen that the delay time of each truck in each scenario has a normal shape from scenario 3 to 8. In the first two scenarios the shape of the distribution is Parto. This explains why the confidence intervals are very narrow for the first two scenarios and became larger for the other scenarios. This has implication for the variation in economic loss due to uncertainty in the delay. If the delay is low, the average economic loss is low too, but the variation of these losses is high because the size of the confidence interval is low in comparison to the average delay time.

To validate these results, a mathematical approach to calculate the total delay is used:

$$Average\ delay\ =\ P_{break}\cdot\ N_{bridges}\cdot E(\bar{x})_{categorie} \quad (2)$$

The break chance of a bridge in a certain category is taken and multiplied with the corresponding number of bridges in that category times the average value of the distributions of its categories from Table 1. The total amount of bridges in each category is given by table A4 of the Appendix.

Table 3: The expected average delay calculated with Formula 2 for each scenario

| Scenario | Expected delay time (min) |
|---|---|
| 0 | 0 |
| 1 | 31.75 |
| 2 | 63.5 |
| 3 | 289.87 |
| 4 | 579.75 |
| 5 | 726.12 |
| 6 | 1452.25 |
| 7 | 1822 |
| 8 | 3644 |

The calculated delay from Table 3 is entering in the 95% confidence interval of Table 2. This makes the model fit for analysing economic impacts due to delays on the N1 in Bangladesh.

# Conclusion and discussion

The probability of bridges breaking down will under different scenarios generate delays for trucks on the N1 from Chittagong to Dhaka. The economic losses due to these delays are estimated as a percentage of the value of the cargo. If the delay is low the variation of the economic losses is high. The other way around, if the delay is high, the economic losses are more predictable.

This research also has several limitations due to the stochastic approach. By adding 125 replications to the model this will stabilize the results. The mathematically calculated average delays are all entering the confidence intervals of Table 2.
It is seen that in a scenario where bridges have a higher probability of breaking, the results are more specific. This is because it has less impact if one bridge extra breaks or not. This results in a normal distribution of the results in scenarios 3 and above and a Pareto distribution for the first two scenarios causing a widespread in economic impact in scenario 1 and 2.

A limitation of this study is that all the bridges break at the start of the model at the same time. In this model no set up time has been considered, so all the trucks start in the same place. This leads to the possibility that a bridge is already fixed before the first trucks are arriving.

In a follow-up model these limitations could be taken into consideration. For further research it could be modelled that the chance of bridge breaking down is calculated every time step. Also, when a bridge is fixed, the condition of the repaired bridge needs to be in the A category as it now stays in the same category.

# References

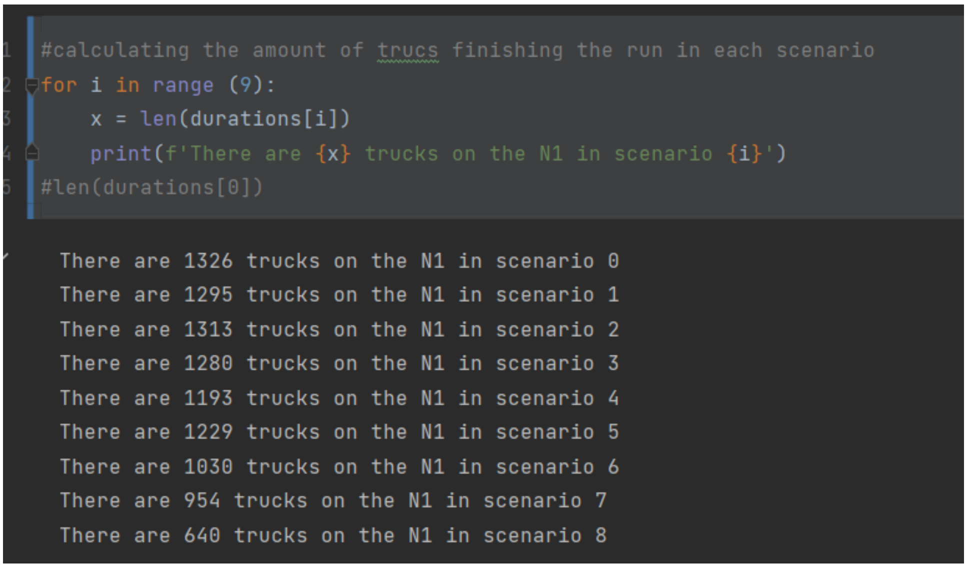

Das, S., & Pohit, S. (2004). *Quantifying transport, regulatory and other costs of India-Bangladesh trade*. National Council of Applied Economic Research.

Python Software Foundation. (2022). pickle — Python object serialization. Python 3.10.2 documentation. Geraadpleegd op 4 maart 2022, van https://docs.python.org/3/library/pickle.html

Kazil, J., Masad, D., & Crooks, A. (2020). Utilizing Python for Agent-Based Modeling: The Mesa Framework. Social, Cultural, and Behavioral Modeling, 308–317. https://doi.org/10.1007/978-3-030-61255-9_30

J., Masad, D., & Crooks, A. (2020). Utilizing Python for Agent-Based Modeling: The Mesa Framework. Social, Cultural, and Behavioral Modeling, 308–317. https://doi.org/10.1007/978-3-030-61255-9_30


# Appendixes



```
#calculating the amount of trucs finishing the run in each scenario
for i in range (9):
    x = len(durations[i])
    print(f'There are {x} trucks on the N1 in scenario {i}')
#len(durations[0])

There are 1326 trucks on the N1 in scenario 0
There are 1295 trucks on the N1 in scenario 1
There are 1313 trucks on the N1 in scenario 2
There are 1280 trucks on the N1 in scenario 3
There are 1193 trucks on the N1 in scenario 4
There are 1229 trucks on the N1 in scenario 5
There are 1030 trucks on the N1 in scenario 6
There are 954 trucks on the N1 in scenario 7
There are 640 trucks on the N1 in scenario 8
```

Figure A1: Amount of trucks in each scenario in the Mesa model.

From Figure A2 can be seen that the amount of trucks finishing the travel on the N1 is decreassing rapidly if the delay scenario are increassing. This because Mesa is only mesuring the travel time if a truck arrives at Dahka.

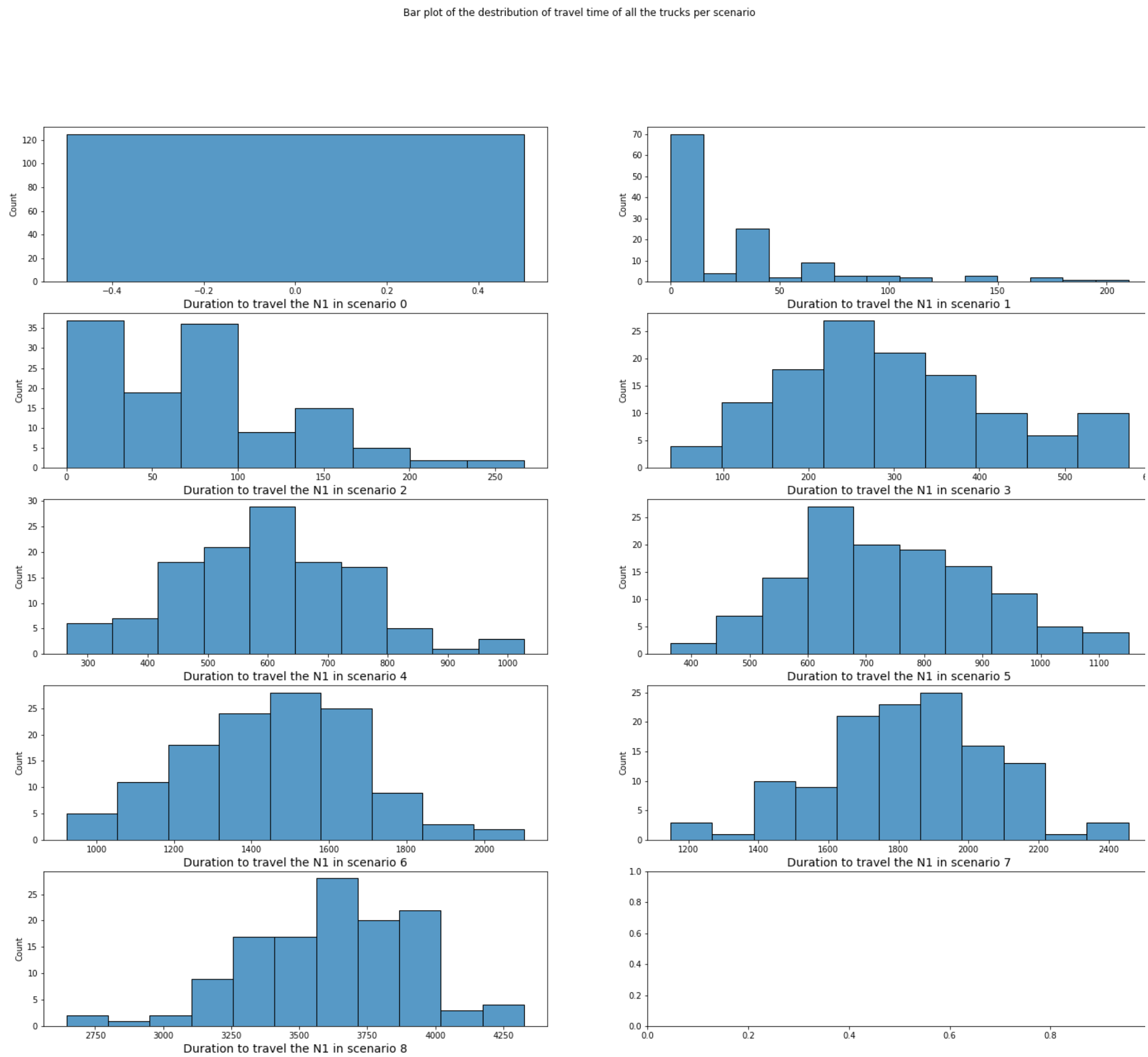



Figure A2: Histograms of the travel time of each truck per scenario.

From Figure A2 can be seen that the distribution of the delay time become normal from scenario 3 to 8. For scenario 0, as expected, a uniform distribution is observed with a low variation. For scenario 1 and 2 a Pareto distribution is observed. This means that the average delay time will be low but that there is a possibility of extremely high out layer values. For scenario 1 the modus delay time lays between 0 and 15 but some trucks have a delay up to 300 minutes, twenty times more.

| Scenario | Cat A % | Cat B % | Cat C % | Cat D % |
|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 5 |
| 2 | 0 | 0 | 0 | 10 |
| 3 | 0 | 0 | 5 | 10 |
| 4 | 0 | 0 | 10 | 20 |
| 5 | 0 | 5 | 10 | 20 |
| 6 | 0 | 10 | 20 | 40 |
| 7 | 5 | 10 | 20 | 40 |
| 8 | 10 | 20 | 40 | 80 |

Figure A3: Scenarios of bridge breaking per categories

Every bridge has a category for its quality. Cat A is the best categories of bridges and Cat D the worst.

| | Cat A | Cat B | Cat C | Cat D |
|---|---|---|---|---|
| S | 274 | 73 | 51 | 4 |
| M | 75 | 36 | 62 | 8 |
| L | 7 | 3 | 13 | 2 |
| XL | 0 | 2 | 4 | 1 |

Figure A4: Amount of each bridge size in each category