Assignment 2 – modelling the N1 highway in Bangladesh
Group 13
EPA 1351 Advanced Discrete Simulation

Sybe Andringa 4558448
Marceau Mertens 4236343
Wessel Groot 4243714
Kim van Vliet 5185114

1 - Data cleaning

For this assignment, two datafiles were provided: the roads and bridges files. In order to use them both, it is necessary to merge them. For this, a jupyter notebook with markdown was created and can be found in the working directory of this project.

First of all, the data frames were filtered on N1 roads and N1 bridges. For the filtered bridges data frame, a function (which was made in assignment 1) to clean the duplicates and bridges that are alike was used to get a clean data frame.

Secondly, the desired columns were selected in both data frames in order to merge the two. Two extra columns 'what' and 'delay' were created to distinguish the road points from the bridges (in order to label them in Simio) and to assign the processing time provided in the assignment, whenever the bridges are broken. This is based on the length of the bridge. While merging, duplicate road points were not appended, since they were present in the bridges data frame.

Lastly, the newly created data frame was sorted on LRP name. Figure 1 gives a short representation of the created dataframe.

1	lrp	lat	lon	length	condition	structureN	chainage	what	Delay
2	LRP010b	236.943.906.141.361	905.375.740.774.869	6.3	A	112531.0	10.88	bridge	Random.Uniform(10,20)
3	LRP010c	236.943.023.869.158	905.377.073.719.626	6.3	A	112532.0	10.897	bridge	Random.Uniform(10,20)
4	LRP011	23.696.638.600.000.000	9.053.480.520.000.000				10.503	roadpoint	0
5	LRP011a	236.923.601.518.691	905.409.183.897.196	8.3	A	101110.0	11.296	bridge	Random.Uniform(10,20)
6	LRP011b	236.922.774	905.410.552				11.312	roadpoint	0
7	LRP011c	236.909.119.315.151	905.454.624.957.575	10.6	B	101114.0	11.808	bridge	Random.Uniform(15,60)
8	LRP012	236.918.052	905.426.108				11.497	roadpoint	0
9	LRP012a	236.884.123.260.204	905.485.585.290.816	9.3	A	101117.0	12.239	bridge	Random.Uniform(10,20)
10	LRP012b	236.883.197.331.632	90.548.649.807.653	6.1	A	101119.0	12.253	bridge	Random.Uniform(10,20)

Figure 1: short representation of merged data frame

With this newly created data frame, a linkage data frame was made. This linkage data frame specifies the order of nodes and servers in Simio, and from what point to the next the links should be established. A short representation can be found in figure 2. This data frame will be referred to as 'Destinations' in Simio.

1	Object	Destination
2	Ouput@LRPS	LRPSa
3	LRPSa	LRPSb
4	LRPSb	LRP001
5	LRP001	LRP002
6	LRP002	LRP002a
7	LRP002a	LRP003
8	LRP003	LRP004
9	LRP004	LRP004a

Figure 2: short representation of links data frame

Limitation

By analyzing the first dataframe, an inconsistency can be found. As can be seen in figure 3, the chainage and the coordinates from the bridges are not always in line with the chainages and coordinates from the roads file (rows 5, 6, 7 and 8). This limitation can be assigned to the provided data files which are not consistent with each other. However, this is inevitable since both of the provided data frames mostly consist of interpolated points and chainages.

A solution would be to interpolate the outlying points (mostly bridges) in the newly created merged dataframe. However, for the use of our Simio model, it is not necessary to do this, since it will have no significant influence on the waiting time. Therefore, and for the matter of time, the points are not interpolated and fixed for the merged dataframe.

1	lrp	lat	lon	length	condition	structureN	chainage	what	Delay
2	LRP009	23.705.027.800.000.000	905.193.327				8.503	roadpoint	0.0
3	LRP009a	23.706.083.300.000.000	905.215.271				8.763	roadpoint	0.0
4	LRP010	23.702.527.800.000.000	905.273.882				9.503	roadpoint	0.0
5	LRP010a	237.015.278	905.281.938				9.615	roadpoint	0.0
6	LRP010b	236.943.906.141.361	905.375.740.774.869	6.3	A	112531.0	10.88	bridge	Random.Uniform(10,20)
7	LRP010c	236.943.023.869.158	905.377.073.719.626	6.3	A	112532.0	10.897	bridge	Random.Uniform(10,20)
8	LRP011	23.696.638.600.000.000	9.053.480.520.000.000				10.503	roadpoint	0
9	LRP011a	236.923.601.518.691	905.409.183.897.196	8.3	A	101110.0	11.296	bridge	Random.Uniform(10,20)

Figure 3: limitation in merged data frame

2 - Model construction

In order to properly use the cleaned data into the model, the tables were imported in Simio. Using auto-create instance, from this table 4 different objects were created, their types defined in the column 'what'. A startpoint is a source, by which trucks are spawned. This is the first waypoint of the N1-road. An endpoint is the sink (the last waypoint of the road), through which all trucks disappear from the model again. This is the last waypoint of the N1-road. A roadpoint is an actual roadpoint, and is simulated as a transfer node. A bridge is a real-life bridge, and is simulated as a server. Each roadpoint and bridge has an x and y-coordinate, in order for it to appear on the correct location. This is defined in and linked to the Object property column called 'Name', which gives each point on the road a unique name based on its LRP. Note that these names are only unique for this specific road; would the model need to be expanded to multiple roads, the road name should be in this column as well.

Each bridge additionally has a ‘Condition’ and a ‘Delay’. Based on its condition, each bridge has a certain probability of breaking down, and a certain delay time whenever it in fact breaks down. The latter has been added in the table through Python, and is therefore a ‘given’. However, as the probability of breaking down for different bridge conditions should be variable parameters, they have been defined as model controls, called ‘BreakingProbabilityA’ and similarly for each other bridge condition. A process inside the Bridge object, initiated at the start of the run, defines whether a certain bridge is going to be broken or not during that run. This process first determines what kind of bridge it is (using a property that gets its value from the corresponding row in the ‘Locations’ table), then determines whether this bridge is going to be broken based on the controllable probability, and finally assigns the state variable ‘NewProcessingTime’ the value of ‘MyDelayTime’ whenever this bridge is broken. MyDelayTime is a property that gets the ‘Delay’ value from the corresponding row in the ‘Locations’ table, and ‘NewProcessingTime’ is a state, that has initial value ‘0’, so that when the bridge is not broken, the processing time is 0. The bridge processing time is linked to be equal to this state value.

Trucks are simulated as entities in this model. They spawn from the source ‘Startpoint’ every 5 minutes, and have a speed of 48 km/h (random triangular(46,48,50)), based on the average real-life speed of driving on this road. The trucks get their next destination from the roadpoints or bridges where they currently are. ‘Entity Destination Type’ for the roadpoints and the output nodes of the bridges have been linked to the ‘Destinations’ table, which for each node contains information about what the next node is the trucks need to travel to from there. In this way, the trucks know where to go.

3 - Experiment design

To analyse the system, different scenarios were formulated. These scenarios represent different levels of disaster, in which percentages of different qualities of bridges break down to see how that affects the mobility within the country. The percentage of bridges of a certain quality that break down in each scenario is shown in table 1.

Table 1 - Breakdown odds of different categories of bridges

Scenario	**Cat A%**	**Cat B%**	**Cat C%**	**Cat D%**
1	0	0	0	0
2	0	0	0	5
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

To measure the difference between the scenarios the key performance indicator ‘average driving time’ was used, which was accounted as the average time a truck entity spent in the system. To account for randomness because of the distributions used in the model, ten replications will be ran with a runtime of 5 times 24 hours. No warm-up period is used, because currently the trucks that are already in the model do not have an effect on the model and therefore it is not necessary to have an initial number of trucks already in the model via a warm-up period.

4 - Results

The results (figure 4) show that when no bridges break down, the average driving time from Dhaka to Chittagong (the entire N1 highway) is 7.4 hours. Furthermore, the results indicate that the average time spent in system increases dramatically when more bridges break down. The cumulative and relative percentage increase in travelling time has been indicated in table 2. This shows that a breakdown of 5% of all category D-bridges only causes a minor increase in travelling times. However, in the worst case scenario in which 80% of D bridges, 40% of C bridges, 20% of B bridges and 10% of A bridges break down, travelling times increase to 2,5 days or by 723% compared to the base case scenario. Furthermore, every increase in breakdown odds increases the travelling time further; apart from a 5% breakdown of category D bridges, every subsequent increase causes an additional delay of 20% travelling time or more.

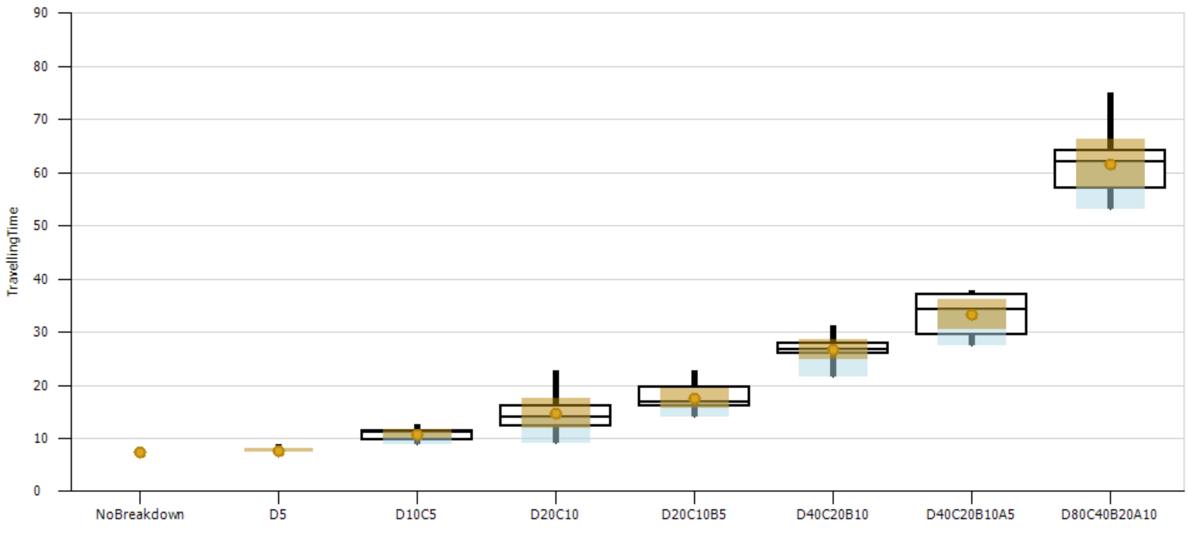

Figure 4. Truck travelling time on the N1 in the different scenarios

Table 2 - relative and cumulative increase of time of trucks in the system

Scenario	Average time in system (h)	Cumulative increase (compared to scenario 1) (%)	Relative increase (%)
1	7,5	-	-
2	7,7	3%	3%
3	10,8	44%	40%
4	14,7	96%	36%
5	17,6	135%	20%
6	26,8	257%	52%
7	33,3	344%	24%
8	61,7	723%	85%

5 - Conclusions & discussion

The results of the model indicate that the gravity of the disaster and the concordant number of broken bridges matters greatly for the travelling time over the N1, with a possible increase of travelling time by 723%. This number calls for more research to create a better overview of which bridges are vulnerable to natural disasters and which bridges demand immediate repair. A model that includes all main roads and bridges in Bangladesh, in which trucks have an actual destination, could provide insight in disaster risk mitigation and bridge repair.

Furthermore, some minor points of attention could be covered by further models. First of all, the average travelling time of 7.4 hours over the N1 is remarkable, since this means that the N1 road should be 355 km long. However, the chainage of the roads data indicates that the road should be 466 km long, whereas the assignment indicates that the road should have a length of 287 km. When it is assumed that the chainage indication is correct, the lower length indicated by the model could be due to the modelling of the road; trucks travel in a straight line from LRP to LRP, which could greatly reduce the road length in the model if the road is, in fact, curved. This is a semantic accuracy error; however, the pragmatic accuracy is still sufficient, because the objective of the model is to investigate bridge breakdown, not road length.

Secondly, the model could be improved further by cleaning the dataset even more; at this moment, there are certain bridges that come before certain roadpoints, while in the dataset they are depicted as being later on the road. This causes trucks to sometimes turn around for a short time and then continue their journey again, which causes the total travel time to be higher than in real-life.

Thirdly, whenever bridges are broken, they keep being broken during the entire run. While this is pragmatically sufficient, this is not realistic. In real-life, this is more dynamic during a run, instead of being a static true/false during the entire run.

Finally, an implication of the way the system has currently been modelled is that every truck has the same delay time over a particular bridge. This has been done because of time constraints, and because it is thought to not have a very large impact on the final results, when a large amount of runs and a large amount of bridges are concerned. This would however be the next step in modelling, and could be achieved by making a new process inside the bridge that does not fire at the start of the run, but every time a new truck enters this bridge. It could then pick a different delay time from a probability distribution every time. This process would only run if the bridge is indeed broken (which could be determined through a boolean state variable inside the bridge); otherwise, the delay time is always 0.

These improvements could make the model more valid. However, this simple model has already proven its worth, by showing that broken bridges during a natural disaster in Bangladesh merit more attention and repair plans should be in place, or else the travelling time of trucks with relief goods could be severely delayed.