Delft University of Technology 2024/2025
Dr. rer. nat. Özge Şahin (O.Sahin@tudelft.nl)
Dr. Dorota Kurowicka (D.Kurowicka@tudelft.nl)

Traffic congestion management using network modeling

Modelling 2A [AM2050-A TWN]

1 Background

Traffic congestion is a significant issue on the roads connecting different cities. The congestion leads to increased travel times, pollution, and economic costs. In the Netherlands, cities such as Delft (City 1) and Rotterdam (City 2) are closely connected by road networks that frequently experience heavy traffic, especially during peak hours. Efficient traffic management strategies are crucial for solving these problems and ensuring smooth mobility between the cities.

In this project, we will tackle simplified road networks between different cities in the Netherlands as a graph and use various algorithms to study and mitigate traffic congestion. By focusing on intercity traffic, we can explore real-world traffic problems and propose solutions using mathematical and computational techniques.

Let us first look at a very simple example of the problem.

2 Basic model and traffic management strategy

Consider a directed graph $G = (V, E)$ with nodes V and edges E representing the road network between City 1 and City 2, where V represents intersections or important junctions, and E represents the roads (see Figure 1).

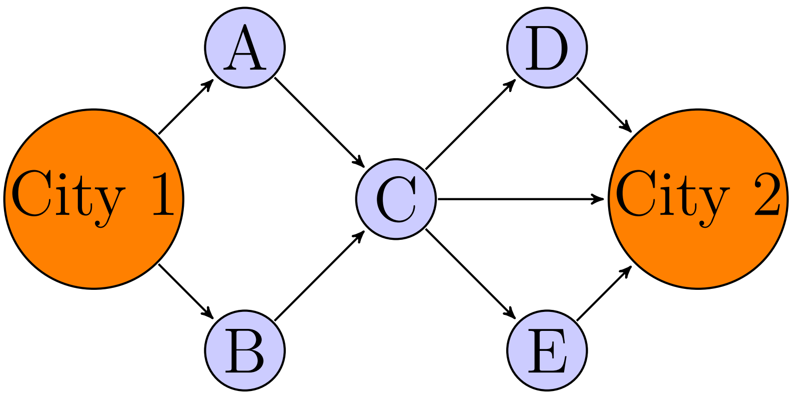

Figure 1: A network representation of road connections between Cities 1 and 2.

The graph represents the road infrastructure, which should also include information about the length, capacity, speed limit, etc., of each road (edge).

Vehicles enter the network from City 1 and move towards City 2 (we observe them at discrete time steps $t = 1, 2, 3, ..., t_{max}$ in minutes), with travel times on each edge influenced by the number of vehicles present. Let $N(e,t)$ denote the number of vehicles on edge e at time t. Even though more realistic assumptions can be made, for simplicity, we will assume that vehicles enter the network only from City 1 and exit the network as soon as and only when they reach City 2. Vehicles can only enter an edge if the edge capacity $C(e)$, representing the maximum number of vehicles on the road, is not reached, hence when the edge is available. Otherwise, vehicles must wait till there is some space on the road. For instance, if the edges $(City1 \to A)$ and $(City1 \to B)$ are at their full capacity at a given time step, the vehicles wait to enter at the node City 1. The waiting time increases their total traveling time. Furthermore, if there is more than one (path) that a vehicle can follow, a strategy describing such choice needs to be designed.

Travel time on a road segment $tt_i(e,t)$. In this project, we model the travel time of a vehicle on each edge of the road network using the Bureau of Public Roads (BPR) function. The BPR function is widely used in traffic engineering to estimate travel times based on traffic volume on a road segment. The travel time $tt_i(e,t)$ of a vehicle i on an edge e at time t is given by the BPR function with a stochastic term as:

$$tt_i(e,t) = tt_0(e)\left(1 + \alpha\left(\frac{N(e,t)}{C(e)}\right)^{\beta}\right) + \epsilon_i, \quad t = 0, 1, 2, ..., t_{max}$$

where $tt_0(e)$ is the free-flow travel time on edge e, calculated as the length of the road segment divided by the speed limit, $N(e,t)$ is the number of vehicles on edge e at time t, $C(e)$ is the capacity of edge e, and α and β are parameters that determine the sensitivity of travel time to changes in the number of vehicles in traffic. Commonly used values for these parameters are $\alpha = 0.15$ and $\beta = 4$. The term ϵ_i represents random variations in travel time for vehicle i, which can be modeled as a normally distributed random variable with mean zero and standard deviation σ, i.e., $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. In this simple example, we assume that $\sigma = 2$. In more realistic scenarios, choices depending on, e.g., speed, length of e, or the type of vehicle can be considered.

Capacity of a road segment $C(e)$. Let ℓ_{car} be the average length of a car (in meters), d_{spacing} be the minimum required distance between cars (in meters), $L(e)$ be the length of edge e (in meters), and $n_{\text{lanes}}(e)$ be the number of lanes on edge e. The capacity $C(e)$ of edge e can then be calculated using the formula:

$$C(e) = \left\lfloor \frac{n_{\text{lanes}}(e) \times L(e)}{\ell_{\text{car}} + d_{\text{spacing}}} \right\rfloor.$$

This formula provides a calculation of the road segment's capacity by rounding down to the nearest integer, reflecting the physical constraints of vehicle length and safe driving practices. Here, we assume an average car length of $\ell_{\text{car}} = 4.5$ meters and a minimum vehicle spacing of $d_{\text{spacing}} = 55$ meters to calculate the capacity of each road segment.

The free-flow travel time of a road segment $tt_0(e)$. The free-flow travel time $tt_0(e)$ on each edge e (in minutes) is calculated as:

$$tt_0(e) = \frac{L(e)}{v_{\text{limit}}(e)} \times \frac{60}{1000},$$

where $L(e)$ is the length of the road segment corresponding to edge e (in meters), $v_{\text{limit}}(e)$ is the speed limit on edge e (in kilometers per hour).

Vehicle movement. Vehicles move through the network according to a probabilistic model. At each time stamp the remaining travel time for vehicle i on edge e at time t, denoted as $rt_i(e,t)$ is computed:

$rt_i(e,t)$= {travel time of i on e} - {time i stayed on e up to t}.

If this remaining travel time is smaller than a time step, then this vehicle is ready to leave e and proceed to the next edge on the path.

Which edge? When a vehicle is ready to enter the next edge (is at node u), then

- If all outgoing edges (edges to children of u) are not available - wait.
- Prioritize the edges based on their average remaining times. That is, compute the average remaining times of each possible edge choice (edges to children of u).

  The probability $P(u \to v, t)$ that a vehicle chooses an edge $u \to v$ at time t is inversely proportional to the average travel time on the edge:

  $$P(u \to v, t) = \frac{\frac{1}{\frac{1}{N(u\to v,t)}\sum_{j\in(u\to v)} tt_j(u\to v,t)}}{\sum_{w\in \text{ch}(u)} \frac{1}{\frac{1}{N(u\to w,t)}\sum_{j\in(u\to w)} tt_j(u\to w,t)}},$$

  where $\text{ch}(u)$ is the set of children nodes (nodes that can be reached directly from u and are not at their full capacity). The numerator represents the attractiveness of choosing edge $(u \to v)$, with lower travel times being more attractive. The denominator is a normalization factor ensuring that the probabilities sum to 1. This approach ensures that vehicles are more likely to choose edges with lower travel times, reflecting more realistic decision-making processes in the presence of varying traffic conditions.

Assumptions and parameters for the simple model.

1. **Traffic flow assumptions:**
   - Peak hours (8-9 AM and 5-6 PM): sample from $\mathcal{N}(10,\ 1)$ and round the nearest integer per minute at City 1.
   - Off-peak hours: sample from $\mathcal{N}(4,\ 1.5)$ and round the nearest integer per minute at City 1.
   - No entrance and exit from other nodes.

2. **Parameters:**

   - Time steps: 1 minute per step.
   - Initial conditions: The initial conditions on the road sections are required for the model. These might be known but if they are not known (as in our simple example) the usual solution is to start the model with empty roads and run it for, say 120 minutes, to get initial conditions on the roads.
   - Simulation duration: 1440 minutes, where 120 minutes are for initialization.
   - Edge information: All edges are the same type.

Edge	$L(e)$ **(in meters)**	$v_{\mathbf{limit}}(e)$ **(km/h)**	$n_{\mathbf{lanes}}(e)$
City 1 → A	3000	100	2
City 1 → B	5000	100	2
A → C	4000	100	2
B → C	6000	100	2
C → D	2000	100	2
C → E	3000	100	2
C → City 2	7000	100	2
D → City 2	4000	100	2
E → City 2	5000	100	2

Mitigate congestion. To mitigate congestion, we propose a simple traffic management strategy. The strategy involves identifying the most congested edge in the network and increasing its capacity. This can be done by adding more lanes to a road or improving traffic signal timings at an intersection. We simulate the initial traffic flow and identify the edge with the most vehicles over all time steps. Let e^* be the most congested edge, with $C(e^*)$ denoting its capacity. We then simulate an increase in its capacity with the factor δ:

$$C(e^*) \leftarrow \delta \times C(e^*).$$

After enhancing the capacity of the most congested edge, we re-simulate the traffic flow to observe the impact of this intervention.

Output. Simulate the traffic model based on the defined parameters and rules using a constructed network graph representing connections between two theoretical cities in Figure 1. Track individual vehicle travel times from City 1 to City 2 and calculate the travel time per vehicle throughout the simulation. Present results of the simulations in the form of distributions of travel time between City 1 and City 2 in pick and off-pick hours. Fit parametric distributions (if possible) to travel times and perform goodness-of-fit tests.

The simulation should include a sensitivity analysis to explore how variations in key parameters such as vehicle entry rates, maximum road capacity, and initial vehicle distribution impact traffic flow and congestion.

Visualization. The simple example presented so far must be implemented and visualized. You are encouraged to implement a nice visualization of the simulation that will be used to present your results and also help debug the simulation program. Prepare your visualization so that it can be applied to more general networks.

3 More realistic example

To conduct a comprehensive traffic analysis using real data in the Netherlands, you must construct a detailed graph of a road network between two /three cities. This involves obtaining data on road types, lengths, and speed limits for all road segments. The road types are crucial for assessing the realistic capacity of each edge, as different road types have varying maximum vehicle capacities. For instance, highways generally allow for higher speed limits and thus have a higher capacity than urban roads. This means that highways will typically have lower free-flow travel times and higher capacities compared to other road types. The length and speed limit of each road segment will allow for calculating average travel times and estimating maximum vehicle flow. This information can typically be sourced from the National Data Warehouse for Traffic Information (NDW) and open data portals. After constructing the network, it is important to calibrate the model to ensure that the parameters accurately reflect real-world conditions. Then, you should make simulation analyses with your data and network. This comprehensive approach tests the traffic model's robustness and enhances its applicability to actual traffic systems and management strategies.

The simple example and a more realistic example, together with simple congestion mitigation strategies and visualization, form the basis of the project.

4 Extensions

To make your model more realistic and improve your grade, you are encouraged to consider some extensions. You have a lot of freedom in which direction you want to go. Some examples of possible extensions are:

- **Scaling to larger networks:** Expand the simulation to include a larger network of roads and intersections, potentially considering a metropolitan area in the Netherlands. This will require increased computational resources and possibly more sophisticated software tools and parameter calibration to manage the complexity and size of the data.
- **Implementation of Networks-in-Networks approach:** Apply the methodology, called Networks-in-Networks, from the provided research paper[1] to enhance the current traffic simulation model.
- **Personalized extensions:** After consulting with your supervisor, you can propose your extensions to the project.

[1] Hackl, J., & Adey, B. T. (2019). Estimation of traffic flow changes using networks in networks approaches. *Applied Network Science, 4(1)*, 28. https://appliednetsci.springeropen.com/articles/10.1007/s41109-019-0139-y

4.1 Detailed Schedule

Week 1: Form groups and prepare a group work plan until Sep 6, 2024. For this, read the document on Brightspace on 'Working in groups' and perform the steps presented in the section 'Getting started.' In particular, discuss the project to see if everything is clear and make a plan for the project. Also, discuss each other's strengths and allocate the work according to the strengths of the group members. Choose software that you would like to use in this project. If time permits, make a first implementation of the network as discussed in Section 2.

Week 2-3: The goal in these weeks is to implement the basic model and strategies discussed in Section 2. This includes defining the graph and simulating the traffic flow using the given model. Implement the visualization of the simulations. Generate figures of traffic flow and congestion patterns at different time steps. Estimate the distribution of travel times and perform a statistical analysis of your results.

Week 3-4: Apply the knowledge you acquired in the basic model to the real traffic data from the Netherlands. Construct a more realistic example.

Week 5: Deliver the first version of the report. This version does not need to be complete but should at least contain the initial model results from Section 2. The report's structure should indicate which sections will be in the final report. Include a first version of the introduction, a literature review etc.

Week 6-8: Work on extensions. Address interesting research questions related to these extensions. At this point, you should have a realistic idea of how much time you can spend on these extensions.

Week 9: Finalize the report and prepare the final presentation. Ensure the report includes all sections, with detailed explanations of the model, management strategies, results, and extensions. Include visualizations and a thorough statistical/probabilistic analysis. Prepare a video with the results.

4.2 Deliverables

For this project, you are required to deliver the following:

- Final report (20-25 pages). It should be written and does not need to contain your computer code.
- Computer code that you used to generate the figures and calculations in the report.
- Movie presenting what you have done in this project (max 5 minutes). Be creative! This movie will be judged on content, clarity, entertainment and creativity.