UAS4T Competition

H. Tavafoghi, J. Porter, K. Poolla, P. Varayia
University of California, Berkeley

Outline. We provide a brief overview of the algorithm we developed to estimate the queue length in urban arterial roads based on vehicle traces. We first review the definition of *queue length* and what/when vehicles are considered as part of a queue. We then describe the key idea that underlies the algorithm. We then provide an overview of all the components of the algorithm and the code that enables us to calculate the results from the original dataset. Finally, we present the results of our algorithm for the three regions of interest.

Definition of queue. To estimate the queue length, we use the notion of *back of queue* as defined in Highway Capacity Manual [1]: "The distance between the stop line of a signalized intersection and the farthest reach of an upstream queue. Slowly moving vehicles or people joining the rear of the queue are usually considered part of the queue. The internal queue dynamics can involve starts and stops. The vehicles previously stopped at the front of the queue are counted even if they begin moving." Accordingly, the length of a queue can grow even during the green phase of a traffic light.

Key idea. The above definition of queues is similar to macroscopic shock2 wave model of a queue formation [2]. In the shock wave analysis of a queue, a shock wave is defined as "the motion (or propagation) of an abrupt change (discontinuity) in concentration" [3]. As such, to estimate the queue length using the shock wave model one needs to estimate how far the queuing shock wave keeps propagating into the upstream. Fig. 1 depicts the evolution process of a queue during a cycles, where the thick blue line captures the time-evolution of the queue length.

We use the macroscopic shock wave approach above to develop the key idea behind our algorithm, which operates on the UAS4T dataset that provides microscopic-level information. Fig. 2 depicts the position of vehicles in the most left lane of west bound Leof. Alexandras that slow down significantly ($<33\%$ of average (non-stop) speed $\simeq 5-10$ km/hr).

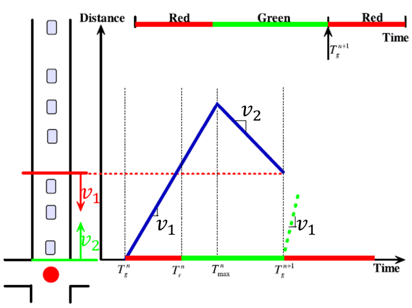

Figure 1: The evolution of queue (shock wave propagation) during one cycle [2].

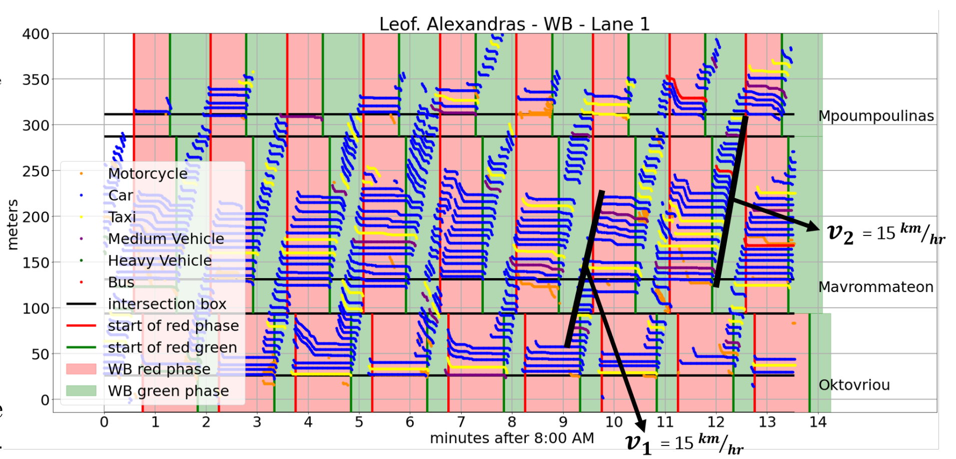

Figure 2: The footprint of (almost) stopped vehicles on the left lane of Leof. Alexandras (west bound)

As seen, the rate the queue starts forming up (charging rate v_1 in Fig. 1) and disappearing (discharge rate v_2 in Fig. 1) are almost equal to 15 km/hr, which appears to be more or less the same over time and intersections. As such, we can utilize the estimated shock wave propagation speed and align (over time) the vehicles that (potentially) belong to the same queue, compensating the delay that vehicles in the back of a queue have compared to vehicles in the front. That is, we time-shift each queued vehicles based on how far along the road they are from a reference point on the road, *i.e.* time-shift equals distance to 0 divided by 15 km/hr. Fig 3 depicts the result of such a time-shift to align the vehicles that potentially belong to the same queue. Accordingly, we use a simple approach to estimate and locate the longest queue length as follows. Using the time-shifted version of (almost) stopped vehicles, searching over (adjusted) time, we look for the longest chain of queued vehicles such that the gap between two consecutive vehicles is not larger than a maximum allowable gap that depends on the type of vehicles before/after each gap.

Note: The red and green phases depicted in Fig. 2 and 3 are only for illustration purposes; they are not used in the code to estimate the longest queue, the lane it occurs, *etc.*

Note: To estimate the length of the longest queue, we do not use the location of intersections. However, to determine the information about spill-backs, we do utilize the location of intersections along the road.

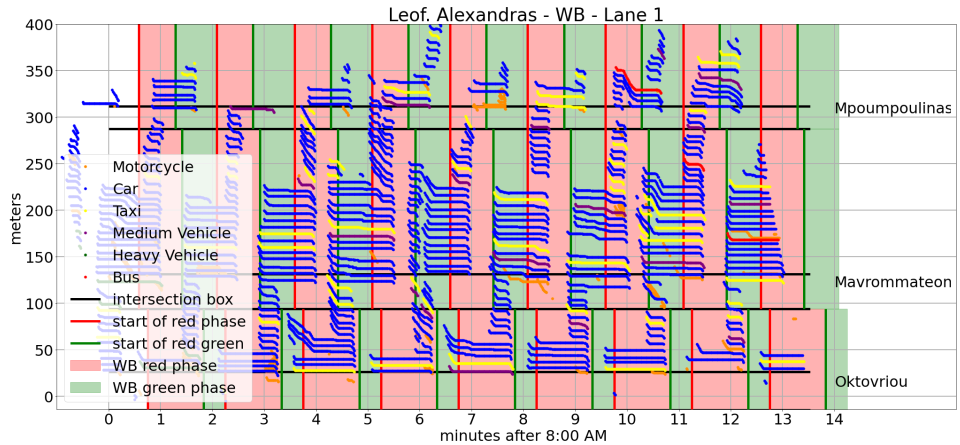

Figure 3: The time-shifted (aligned) footprint of queued vehicles on the left lane of WB Leof. Alexandras.

Code. The algorithm described above is written in Python (Jupyter Notebook). Below we briefly describe the main components of the code and their functionalities.

- **Parsing the dataset:** running `csv2pickle.ipynb`, we parse the competition dataset into two Pandas dataframes: (i) `meta_df` that contains trip-level summaries, (ii) `trace_df` that includes vehicle traces

- **Preprocessing & road projection:** running `projection-road.ipynb`, we project the traces onto different roads in the area. We note that the projection is done for all roads, not only the ones specified as the interest areas. The result contains new columns in dataframe `trace_df`; namely the main new columns are "road": the street name each GPS point is assigned to; "dir": the direction the vehicle is traveling along the road; "X_road": the location of the vehicle along the road from a reference point (depending on the direction of the road; its intersection either with Alexandars or Oktovriou.); & "Y_road": the location of the vehicle across the road (later be used to determine lanes). The projection is done by first estimating a heading angle for each GPS measurement of vehicles, and then calculating a score for each road that depends on the closest distance of the vehicle from the road center line & vehicle heading alignemnt with the road direction. For each point, the road with the best score is selected, and the new columns are generated accordingly. The code takes `street_information.csv` as an input which contains the roads' coordinates. The coordinates are obtain from *Nominatim API* for OpenStreetMap data, and then cross-validated by GoogleMaps data.

- **Algorithm for queue estimation:** running `queue-estimate.ipynb`, we estimate the longest queue using the algorithm described above, and generate visulizations that includes Fig 2 and 3. We describe the main steps in the algorithm below. **(1)** Using function `select_stopped_vehicles`, we select vehicles that can potentially be part of a queue. We filter out vehicles that stop temporarilly in the middle of the road for other reasons (*e.g.* Taxis). **(2)** We determine lanes based on "Y_road" values. The most left lane is labeled as lane "1" and the label index increases accordingly for the lanes on the right. The only exception is for the ramp between NB 28is Oktovriou & EB. Leof Alexandras where the left lane starts as lane "2" to match the final lane on EB Alexandras it leads to. **(3)** Using function `wave_time_delay`, we time-shift queued vehicles according to the idea described above. **(4)** Using function `find_max_Q`, we determine (i) max queue length (in meters & #cars), (ii) lane it occurs, (iii) coordinates of its start & end, (iv) the time max length occurs, (v) which intersection it starts from, whether spillback occurs, and when and at which intersections it occurs. **(5)** Using function `plot_queues`, we generate plots similar to Fig. 2 and 3.

Results. The following results are generated verbatim by running the code described above. The code also generates plots similar to Fig. 2 and 3 for all lanes in each area.

Leof. Alexandras, with direction towards 28is Oktovriou

```
(i) queue length = 284.71 meters, 31.0 #vehicles
(ii) forms in lane = 2.0
(iii) coordinates: from (37.991626,23.732484) to (37.991210,23.735119)
(iv) max queue occurs at 09:43
(v) spillback occurs, queue starts behind intersection with "Mavrommateon"
spillback occurs at intersection with "Mpoumpoulinas" at time 09:16
```

Note: Lane 2 denotes the second lane from the left (from the middle of Leof. Alexandras)

28is Oktovriou, with direction towards Leof. Alexandras

```
(i) queue length = 75.57 meters, 12.0 #vehicles
(ii) forms in lane = 2.0
(iii) coordinates: from (37.991583,23.732122) to (37.991427,23.731421)
(iv) max queue occurs at 08:35
(v) spillback occurs, queue starts behind intersection with "Mavrommateon"
spillback occurs at intersection with "In ramp" at time 08:19
```

Note: Lane 2 denotes the most left lane in the link connecting 28is Oktovrio to Leof. Alexandras, which leads to lane 2 & 3 (varies for different vehicles) on Leof. Alexandras.

28is Oktovriou, above Leof. Alexandras with direction towards the South: From the description, it is not clear whether the left turn lane toward Leof. Alexandras has to be included or not. Thus, we report the result of the code for both scenarios.
Note: The difference between max queue length is marginal between the two scenarios (the lane and occurrence time differ).
Note: Our algorithm reports no spill-back. This is because the spatial coverage of the dataset does not include the upstream intersection with Cheiden. However, it appears very likely that spill-back occurs in both scenarios, and the actual max queue length is greater than 103 meters.

LT lane (lane 1) toward Leof. Alexandras not included

```
(i) queue length = 102.15 meters, 14.0 #vehicles
(ii) forms in lane = 2.0
(iii) coordinates: from (37.991893,23.731268) to (37.992773,23.731496)
(iv) max queue occurs at 05:55
(v) no spillback, queue starts behind intersection of "Alexandras"
```

LT lane (lane 1) toward Leof. Alexandras is included

```
(i) queue length = 102.74 meters, 18.0 #vehicles
(ii) forms in lane = 1.0
(iii) coordinates: from (37.991889,23.731333) to (37.992790,23.731536)
(iv) max queue occurs at 05:51
(v) no spillback, queue starts behind intersection of "Alexandras"
```

References

[1] Manual, H.C., 2000. "Highway Capacity Manual." Transportation Research Board, Washington, DC, fourth edition Edition, ISBN 0-309-06681-6.

[2] Liu, Henry X., Xinkai Wu, Wenteng Ma, and Heng Hu. "Real-time queue length estimation for congested signalized intersections." Transportation research part C: emerging technologies, 2009.

[3] Stephanopoulos, G., Michalopoulos, P.G. "Modelling and analysis of traffic queue dynamics at signalized intersections." Transportation Research, 1979.