Implementation of a macroscopic simulation in Lyon area, based on Macroscopic Fundamental Diagram theory

VICTOR BOULANGER - 03/09/18 - LICIT
VA TRANSPORT - MASTER MEGA
UNDER MAHENDRA PAIPURI

During this study, we developed a macroscopic traffic simulator based on MFD theory for Lyon city. MFD computation is based on a previous master thesis work, he built the MFD by combining loop detectors & taxis trajectories data.

This kind of simulator is not the first one. Nevertheless, hasn’t applied for real networks of a city scale. The method is transparent; every step is well detailed. This present report can be presented as “MFD-simulator guidebook”.

The approach developed here is a “step by step method”, we first collect all the inputs necessary to launch a raw simulation. Based on these results, we investigate the main sources of error and question the biggest assumptions: (i) data, (ii) the clustering, (iii) production estimation method, (iv) the demand, (v) MFD fits and (vi) trip length computation.

THE DATA
First, loop detectors are scattered all over the network, they are concentrated in certain zones. Then, mean speed is computed with taxis trajectories. Obviously, with more taxis data we have more accurate will be the mean speed. The number of taxis is higher during peaks hours and so on the mean speed is more accurate. Nevertheless, taxis have specifics ways and access to bus lines, where the traffic is in free flow thus the taxis may be faster than other vehicles during peaks.

THE CLUSTERING
MFD computation is based on this step and so on the whole simulation, that’s why this part is very important. Furthermore, this step is time consuming. In fact, even if we start from existing clusters, the main work was realized by hand. Mistakes could remain. Finally, I couldn’t perform too small cluster. For example, the reservoirs close to the network boundaries are quite big. Indeed, the data didn’t allow me to build small cluster here: MFD were highly scattered.
THE DEMAND
First of all, we started by ignoring the demand coming from outside the network. But, we quickly understood that most of the demand is, indeed, coming from the outside: we couldn’t neglect it. That’s why I added ghost reservoirs, designed like entry / exit ways to simulate the external demand. Then, we implemented the demand by OD thus assignment procedure was necessary, and it brings a lot of errors. That’s why we decided to give the demand directly for each macro path, based on the micro routes we add (from IRIS to IRIS). The demand is coming from a 4-steps model but we compare MFD-simulator with empirical results. Besides, accumulation-based model is not efficient for fast demand variation. It could explain why we have a shift between demand peaks. Using the trip-based model could improve this point.

MFD FITS
Fitting the MFD is one of the most important step as the whole simulation is based on that. The lack of points in congested part and near the critical production/accumulation led to complications. Furthermore, the MFD approximation implemented into the simulator allows two degrees of freedom. But with only two degrees we’re not able to define a flat zone near the critical production/accumulation. Besides, I used a trial and error “method” to find the best fits. My objective was to match MFD-simulator mean speed results with empirical mean speed data, as we saw before this output is more trustable.

BROADLY SPEAKING
I assume that the traffic data available will increase over the coming years. With the mobile phone data, the Bluetooth detectors and the connected vehicles. This data rise should make traffic evaluation more accurate, same for the traffic states or the MFD.

Finally, mobility needs specific regulations to be efficient. MFD-based simulators present huge application since the level of aggregation is higher than microscopic or mesoscopic simulator. Besides, the computation time will also decrease with this kind of model. The boundary control seems to be a great application. In fact, by playing with the entry flow into the downtown it could avoid gridlock process efficiently.