Large-Scale Shared Autonomous Vehicles Dial-a-Ride Problem

Chijia LIU[1], Alain QUILLIOT[1], Hélène TOUSSAINT[1], Dominique FEILLET[2]

[1] Université Clermont Auvergne, CNRS, UMR 6158 LIMOS, 63178 Aubière France
{chijia.liu, alain.quilliot, helene.toussaint}@uca.fr

[2] Mines Saint-Etienne, Univ Clermont Auvergne, INP Clermont Auvergne, CNRS, UMR 6158 LIMOS, F - 42023 Saint-Etienne France
feillet@emse.fr

Keywords : *shared autonomous vehicles, dial-a-ride problem, filtering mechanism.*

1 Introduction

We design a new transportation system where dial-a-ride services are provided in urban and rural districts with SAVs. The system aims to satisfy a city-sized passenger request set within a large road network, so an efficient dispatching algorithm is indispensable. Many work deals with a large-scale DARP by introducing some accelerating techniques ([1], [2]). We propose here a filtering mechanism to quickly identify the most worth exploring candidate SAVs to serve the request in need as well as its corresponding insertion positions. With the filtering modules combined to an adaptive best-fit insertion heuristic whose objective is to minimize the fleet size to serve all the requests, experiment results have shown that the processing time can be saved by almost 96% compared to the basic module without filters, while keeping a good dispatching quality.

2 Filtering mechanism

We highlight in this section the basic idea of our filtering mechanism, which is composed of two modules: the fleet screening module and the insertion position filtering module.

The road network and service period are partitioned into zones and sub-periods. We introduce an index matrix $M[\mathcal{Z} \times \mathcal{H}]$, serving as a fleet filter, where $\mathcal{Z}$ is the set of zones and $\mathcal{H}$ is the period set. Each cell $M[z, h]$ stocks all the SAVs that could potentially reach the zone z during the period h. Given a new passenger query, the fleet screening module explores simultaneously from the origin and the destination sides all the cells with zones and periods related to the query, take all the concerned SAVs, sort them by score and form the candidate set.

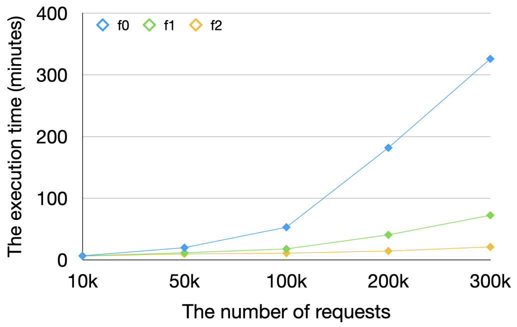
For each SAV v, a structure $M^v_{aux}[\mathcal{Z} \times \mathcal{H}]$ is used to serve as an insertion position filter. Each cell $M^v_{aux}[z, h]$ contains a set of points in the route of v, departing from which v could pass the zone z during the period h. Given a candidate SAV v, the insertion positions for the query's origin and the destination are then selected from the related cells of M^v_{aux}. In addition, a position is regarded as an eligible candidate only if it could arrive at the target zone during the time window imposed by the query.

Finally, only the candidate SAVs and the corresponding candidate insertion positions can be passed to the downstream scheduling process which consists of more strict and time-consuming feasibility tests and constraints propagation along the route due to the insertion of the new request.

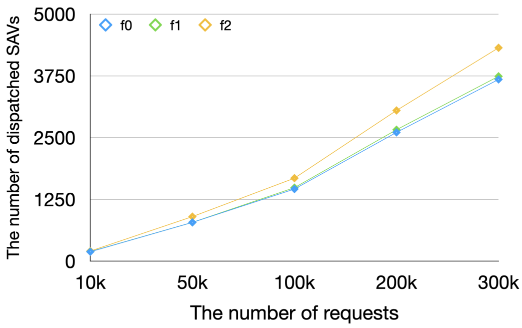
3 Experiment results

We consider the network of the city of Clermont-Ferrand (in France) and its suburbs, with 13,839 nodes and 31,357 arcs. And the service time lasts 24h. The number of requests to be processed in a day varies between 10,000 and 300,000.

In terms of the final fleet size (see Figure 1b), the gaps between the non-filter version and the solutions with filters become bigger when the system process more passenger requests, but remain acceptable, especially for f_1. In terms of the execution time (see Figure 1a), the advantage of our filtering system becomes more and more obvious with the problem size growing. According to all the experiments we have done, the CPU time can be saved by up to 96% with certain filter configurations, at the cost of dispatching more SAVs.



(a) The CPU times taken to arrange request sets with different size by using different filters



(b) The final fleet sizes used to serve the given request sets by using different filters

FIG. 1: Effectiveness of the filtering mechanism and performance of different filters (f_0: basic model without filter; f_1: all the candidate SAVs are passed to the scheduling process; f_2: only 10% of the eligible candidate SAVs are passed to the scheduling process)

4 Conclusions and perspectives

We proposed a filtering mechanism combined to a best-fit insertion heuristic to efficiently solve the large-scale SAV DARP. From the macro SAV fleet to the micro insertion position, different filters and screening techniques are applied sequentially like a waterfall to avoid unnecessary time-consuming calculations. As shown by the experiment results, thanks to the filtering mechanism, the execution time can be reduced by up to 96%, while maintaining a relatively good performance, not only in terms of the final fleet size but also the global service qualities regarding the passengers' ride time.

In the future, we may try mining more information from the massive historical data and obtain the system's travel patterns, which will help to furthermore make fast decisions with good quality in a real-time dynamic scenario. Besides, it is of great significance to make the system aware of the real-time traffic conditions and self-adjust to the congestion in future work.

References

[1] Xiao Liang, Gonçalo Homem de Almeida Correia, Kun An, and Bart van Arem. Automated taxis' dial-a-ride problem with ride-sharing considering congestion-based dynamic travel times. *Transportation Research Part C: Emerging Technologies*, 112:260–281, March 2020.

[2] Shuo Ma, Yu Zheng, and O. Wolfson. T-share: A large-scale dynamic taxi ridesharing service. In *2013 IEEE 29th International Conference on Data Engineering (ICDE)*, pages 410–421, Brisbane, QLD, April 2013. IEEE.