

# A Transportation System Model Based on Smart Autonomous Vehicles with Distributed Coordination

Amanda C. N. de Oliveira<sup>1</sup>, Amit Bhaya<sup>2</sup>, Daniel R. Figueiredo<sup>1</sup>

<sup>1</sup>Programa de Engenharia de Sistemas e Computação – COPPE/UFRJ  
Rio de Janeiro – RJ – Brazil

<sup>2</sup>Programa de Engenharia Elétrica – COPPE/UFRJ  
Rio de Janeiro – RJ – Brazil

{amandacno, daniel}@cos.ufrj.br, amit@nacad.ufrj.br

**Abstract.** *Public transportation in urban centers is of fundamental importance, being a widely investigated topic. Smart autonomous vehicles (SAVs) present a great potential in revolutionizing transportation systems in urban areas, providing more flexible and efficient solutions. This work proposes a new transportation model based on SAVs that provides a station-based, point-to-point service, with distributed coordination. The model offers two different modes of operation, one with exclusive rides, and the other with ride sharing between clients. A simulator has been developed, through which the system's characteristics are analyzed, and the two modes of operation compared. It was observed that with the increase in the system client demand over time the ride sharing mode gets more efficient than the mode with exclusive rides, both in terms to the average time required to deliver clients and the total distance traveled.*

## 1. Introduction

Public transportation is of fundamental value worldwide, since it is the basis of human mobility in most urban areas, in particular in large cities. With the recent technology advancements, formerly unrealistic concepts are now becoming possible. Such effect can be seen in the transportation systems, where new transport models are being developed and implemented. Some examples are bike renting through apps in cities, like Bike Itaú, but also systems of ride hailing, ride sharing, car sharing, etc., like Uber, Waze Carpool, Grab, Lift, Car2Go, and others, that are quite popular nowadays.

It is also the case of autonomous vehicles (AVs), which have been more and more studied. There have been many proposals to incorporate AVs into public transportation systems, models of which have been reviewed and compared in [Berrada and Leurent 2017] and [Fourie et al. 2020]. There have been studies of how smart autonomous vehicles (SAVs) could result in competitive advantages with respect to human operated systems, in the form of reduced cost and possibly also added convenience, as discussed in [Liu et al. 2019]. With SAVs the service cost does not need to include human labor in operating the vehicle, and neither do clients need to make trips to access and return the vehicles.

Related works also characterize the effects of autonomous driving on the traffic management level, like [Wagner 2016], which shows that AVs have the potential to generate time gains at traffic signals in urban areas. The need for specific legislation for

AVs has been discussed in the past, with [van Dijke et al. 2012] suggesting a certification procedure for automated transportation systems, as a way to guarantee the safety of such services, and [Koopman et al. 2019] proposing a safety standard approach for fully autonomous vehicles.

Mathematical models for real-time ride sharing for autonomous vehicles have also been recently proposed [Alonso-Mora et al. 2017]. Such models can be used to generate optimal routes with respect to online demand and vehicle locations scaling to a large number of passengers and trips, as well as different vehicle capacities. Similarly, this current paper also proposes a model for real-time ride sharing, however with the fundamental difference of relying on a distributed coordination, in which vehicles do not have the full knowledge of the trips and vehicle locations, relying solely on local information.

This work proposes a transportation model that uses SAVs to offer a station-based point-to-point service. Two modes of operation are considered: an exclusive service mode, in which the system offers private rides to clients, each vehicle servicing one client at a time; and a shared service mode, in which ride sharing among clients is provided. A similar concept has been proposed in [Lam et al. 2016], which presents a ride sharing transportation system that manages a fleet of AVs to accommodate transportation requests, optimizing scheduling and admission control. However, the aforementioned model uses a control center to manage the vehicles and, as specified before, this work proposes a system with distributed coordination, in which vehicles must communicate with each other to decide the scheduling, without prior knowledge of how many vehicles are available and their current schedule.

Fleet of autonomous vehicles with distributed coordination has also been recently investigated in a constrained industrial scenario to move objects [Mahdavi and Carvalho 2019]. In this setting, vehicles operate with limited information, implementing a cooperative path finding algorithm, communicating their intentions. Despite the different context, the work proposed in this paper implements similar concepts, with vehicles communicating with each other in order to decide their assignments (client requests) and routes to travel.

A simulator of the proposed system has been designed and implemented, in order to ascertain the system's main characteristics, as well as compare the performances between the two proposed modes of operation. The simulator is publicly available at [de Oliveira 2019]. The parameters used in the analysis are based on other works in the field, like [Canca et al. 2014], which implements railways timetable determination optimization, as well as [voytek 2014], which optimizes a vehicle dispatch system.

The remainder of this work is organized as follows. Section 2 presents the proposed transportation system model, with details of the scheduling decision mechanism in 2.1 and on the behavior of the two modes of operation in 2.2. Section 3 explores the system behavior through experiments performed with the system simulator, inferring its characteristics from the obtained results. Lastly, section 4 presents the concluding remarks, with a summary and comments about the obtained results and discussion on future studies.

## 2. The Transportation System

The proposed transportation system is capable of servicing individual passengers, offering point-to-point service without a centralized dispatch center. The system contains a fleet of autonomous vehicles, which are capable of processing and servicing client requests. These vehicles run in dedicated pathways, and can only be boarded and/or exited at specific locations, the stations. All vehicles have full knowledge of the map of tracks and stations. Clients may solicit transportation from any station to any other station. Since the pathways are not shared with other means of transportation, there is no consideration for the possibility of traffic jams, and the vehicles calculate the shortest path between two points using Dijkstra's shortest path algorithm [Dijkstra 1959].

In order to be able to handle requests without a dispatch center, it is assumed that vehicles within a certain distance from each other are able to communicate. In this system, broadcast single-hop communication for both vehicle-vehicle and vehicle-client message exchange is assumed, forming an ad-hoc network. The broadcast range is of high importance in this model, since a too small broadcast range might mean that no communication is established. The system is engineered assuming that no loss of messages when communicating is possible, and messages are delivered in the order they were sent.

### 2.1. Election Procedure

Vehicles use message exchange to execute a leader election algorithm, which will determine the vehicle that will service each particular client request. In other words, this procedure is the mechanism through which the system implements scheduling. Its goal is to guarantee that one and only one vehicle will service each client. The algorithm implemented in this work is a variation of the Bully Algorithm [Garcia-Molina 1982]. The original algorithm was adapted to select the vehicle with the shortest path to service the client, thus minimizing the client waiting time<sup>1</sup> and, therefore, increasing system quality of service. For the algorithm to work, it is assumed that each vehicle has a unique identifier (e.g. the license plate number).

The election procedure begins with the service request by the client, which is received by the vehicles within its broadcast range. Each vehicle that received the message calculates the distance to service the client, and sends an *election* message to other vehicles, containing this distance. When a vehicle receives such a message, it compares the received distance with its own distance, arriving in one of the three:

1. The received distance is larger than the vehicle's own. In this case, the vehicle has a shorter path and has won this comparison. It sends then an *election acknowledgment* message to the other vehicle, and sends its own *election* message if it has not previously done so.
2. The received distance is smaller than the vehicle's own. In this case it has lost this comparison and, therefore, the election. It quits the procedure with no further action.
3. The received distance is the same as the vehicles own. In this case, it will compare the vehicle identification number with its own, and the one with the highest value wins the comparison. It then proceeds with one of the two previous actions accordingly.

---

<sup>1</sup>Since there is no traffic consideration, smaller distance is equivalent to less time.

After sending its *election* message, a vehicle waits for an answer. If it times out without receiving any *election acknowledgment* message, or any *election* with a smaller distance, it understands that it has won the election and proceeds to service the client. It sends a *leader* message to other vehicles as a means of guaranteeing consensus between vehicles.

## 2.2. Modes of Operation

The system offers two different modes of operation: single client (exclusive service), and multiple clients (shared service). In the single client mode, made relevant in the context of a pandemic, only one client can be picked up at a time, making it a fully exclusive system. In the multiple clients mode, however, there is the possibility of ride sharing, since vehicles can pick up new clients before having finished delivering current ones.

It is of note that ride sharing is allowed only when there is no route alteration to the current client. That means that a vehicle will pick up a new client before delivering the previous one if and only if the new client's location is already part of the planned route. If the vehicle needs to alter its route to pick up a client, it will proceed like in the single client mode, and only service the new client after the current client has been delivered. Note that the election procedure still applies, and having the pick-up location on its path does not guarantee that the vehicle will be the one with the shortest path. Therefore the vehicle to serve this client is the one that is closest to delivering it, and not necessarily the one initially closer to it. The advantage of such ride sharing implementation lies on its simplicity, allowing resource usage optimization with little added complexity.

## 3. System Evaluation

The system's performance is analyzed through two different metrics: the *average total service time* and the *total traveled distance*.

**Average total service time  $\overline{T}_n$ :** Let  $t_s(i)$  and  $t_f(i)$  denote the instant of time that the  $i$ -th client makes a request and the instant he/she arrives at his/her destination. Thus,  $t_t(i) = t_f(i) - t_s(i)$  is the total service time of the  $i$ -th client request. The average total service time of the first  $n$  clients of the system is denoted as:

$$\overline{T}_n = \frac{1}{n} \sum_{i=1}^n t_t(i)$$

**Total traveled distance  $D(t)$ :** Considering  $t$  to be a time instant,  $n_v$  the number of available vehicles and  $d_i(t)$  the distance traveled by the  $i$ -th vehicle up until instant  $t$ , the system total traveled distance for time  $t$  is the sum of the distances traveled by each of the  $n_v$  vehicles until  $t$ , being defined as:

$$D(t) = \sum_{i=1}^{n_v} d_i(t)$$

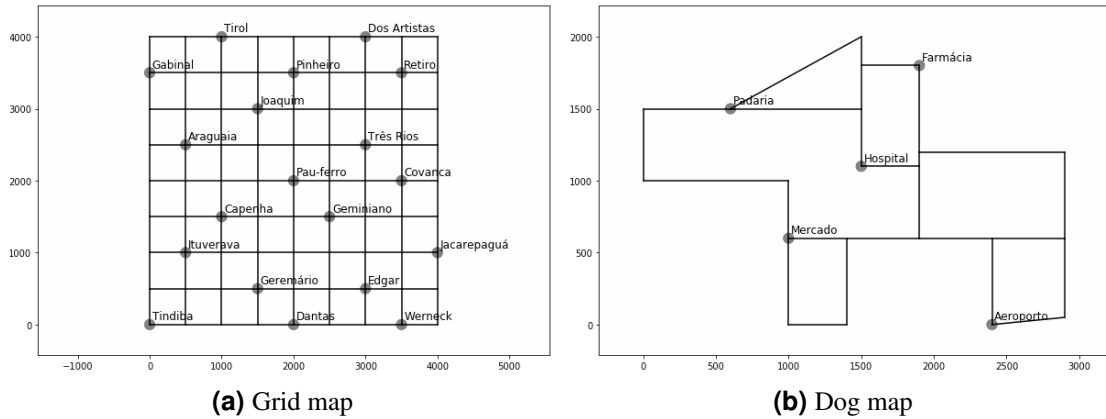
In what follows, the expected values for  $\overline{T}_n$  and  $D(t)$  will be estimated using independent simulations runs, comparing the differences in the values obtained for each

scenario considered. In the experiments, the time  $t$  used to obtain  $D(t)$  will be equivalent to the instant in which the last of the  $n$  clients is delivered at its destination, that is:  $t = \max_i \{t_f(i)\}$ . Henceforth this parameter will be represented by the symbol  $D_n$ , denoting the total distance traveled by the vehicles to deliver the first  $n$  clients of the system.

A simulator was developed for the proposed system. This simulator works with discrete time, iterating the clock counter at each simulation step. At each step, all vehicle and client objects process events and update their status sequentially, and therefore there are no race conditions in the program. The simulator receives as input the desired system conditions under which the simulation is to be performed, and calculates the average total service time and the total distance traveled for a given simulation.

In order to ascertain the model characteristics, several experiments were performed using the simulator. In particular, the average  $\bar{T}_n$  and  $D_n$  are evaluated as function of the system parameters, comparing the behavior of the different modes of operation. The main parameters taken into account are the number of vehicles  $n_v$  itself, the communication broadcast radius  $r$ , the demand intensity and the map of available tracks and stations. The maps used for said experiments are shown in Figure 1.

The client arrival process also influences performance significantly and is random in nature, causing variability in the obtained results and demanding statistical analysis.



**Figure 1. Maps of tracks and stations (represented by the dots in the maps, each with a respective name) used in the simulation experiments**

The first scenario analyzed is the system response to variations in  $n_v$  and  $r$ , for each mode of operation. The vehicle broadcast radius  $r_v$  was set to be  $3r$ , while the client broadcast radius  $r_c$  is equal to the reference  $r$ . This is considered enough to guarantee that all vehicles are able to communicate during an election procedure, even if the vehicles are going away from each other at full speed from diametrically opposed positions with respect to the client.<sup>2</sup>

An uniform client demand distribution, in both time and space, is assumed. The

<sup>2</sup>The simulation establishes a maximum speed of 20 m/s, and the election procedure does not take longer than a minute. For the considered distances, the distance a vehicle can travel in such time is not enough to put it out of the range of the other vehicles in the election.

same client arrival process is used for all the different sets of parameters evaluated. Note, however, that the initial position of the vehicles on the map is random. For each simulation performed, 60 clients were generated and delivered by the system ( $n = 60$ ), and every simulation condition was repeated 40 times, each with a different client arrival process realization, in order to ascertain the statistics fluctuation of the measurements. As expected, varying the client arrivals in the system causes the time and distance outputs to change significantly.

### 3.1. Number of Vehicles and Broadcast Radius

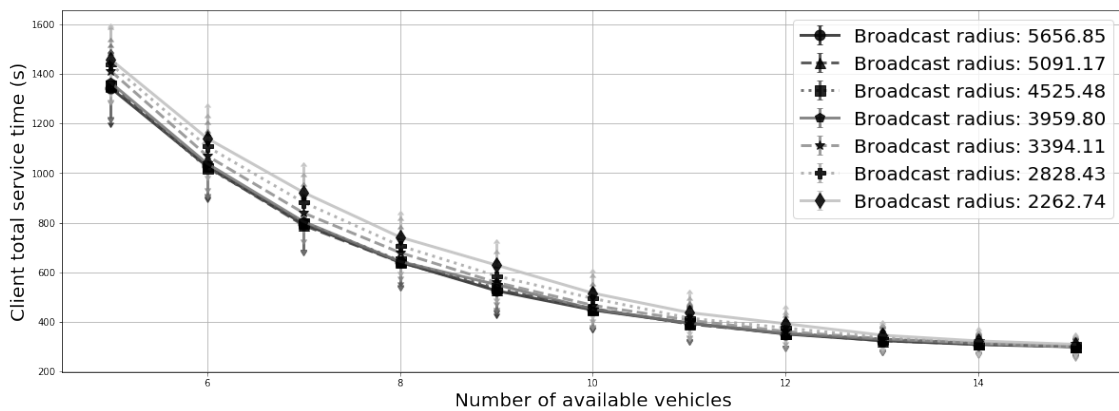
The first set of experiments evaluates the impact of the number of vehicles  $n_v$  and of the broadcast radius  $r$  in the system performance. In this first set of experiments, there is 5% of probability of a single client appearing in the system at every second.

Tests were performed with 7 different broadcast radii, starting with the maximum  $r$ ,  $r_{\max}$ , which corresponds to smallest radius that covers the whole map from any position in it, and progressively diminishing it, reducing 10% of  $r_{\max}$  at a time. The experiments showed that for an  $r$  of at least 40% of  $r_{\max}$ , the system is guaranteed to work, but for 30% or less communication failures may occur.

It was also observed that the broadcast radius does not interfere much in the results, as long as it has a minimum value to ensure that the system works correctly, especially with respect to the time measurements. This behavior can be seen in Figure 2, which shows the average  $\overline{T}_n$  results for the single client mode, tested on the Grid map. It can be observed that the variation of  $r$  does not overly impact the resulting curves, and the differences shown between them are within the statistics uncertainties of the measurements.

Due to the relatively low variations between different broadcast radii, subsequent experiments will present only two  $r$  values for each scenario. These values will be the largest and the smallest ones tested in the simulations, that is, with the full map being covered by the client from any position, the maximum broadcast radius  $r_{\max}$ , and 40% of

**Average client total service time versus number of available vehicles**

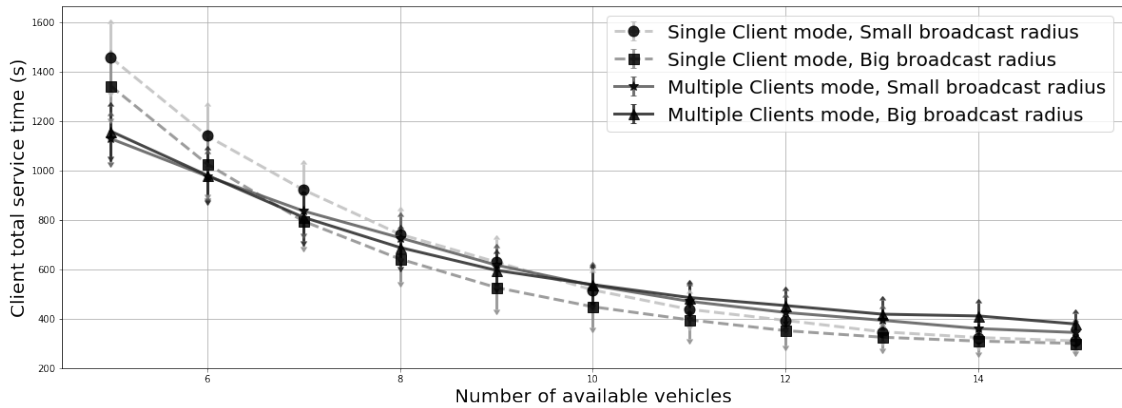


**Figure 2. Average total service time  $\overline{T}_n$ , for single client mode, with respect to the variation in the number of vehicles  $n_v$ , with the standard deviation portrayed in the error bars. The different lines represent different broadcast radii  $r$ , specified in the legend.**

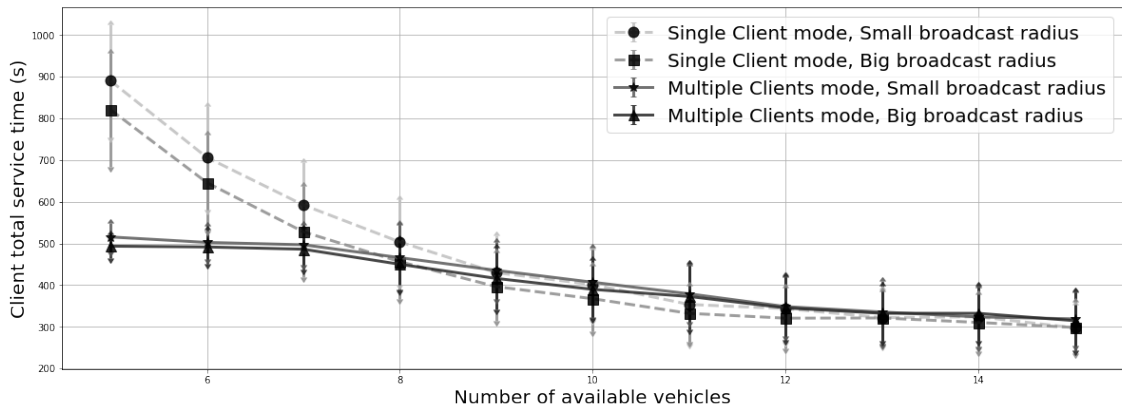
$r_{max}$ . In the case of the Grid map these radii are equivalent to 5656.85 m and 2262.74 m respectively, as shown in Figure 2.

Although the broadcast radius did not impact the system overall behavior, varying the number of vehicles has a very strong influence in both the time and distance measurements, as can be observed in Figure 2. Note that going from  $n_v = 5$  to  $n_v = 10$  there is a reduction of more than half the average  $\overline{T}_n$ . The comparison of the total service time with respect to the different modes of operation can be seen in Figure 3, for both maps considered.

### Average client total service time versus number of available vehicles



(a) Grid map. The big broadcast radius has 5657 m (full map reach from any position,  $r_{max}$ ), while the small one has 2263 m (40% of  $r_{max}$ ).



(b) Dog map. The big broadcast radius has 3523 m (full map reach from any position,  $r_{max}$ ), while the small one has 1409 m (40% of  $r_{max}$ ).

**Figure 3. Average total service time  $\overline{T}_n$ , with respect to the variation in the number of vehicles  $n_v$ , with the standard deviation portrayed in the error bars. The dashed lines correspond to single client mode results, while the solid lines represent the multiple clients mode results, each having two different broadcast radius values  $r$ , differentiated by color and marker.**

From the figures, it is clear that in general having more vehicles leads to a smaller client service time. Moreover, there is the general pattern that with few vehicles, the multiple clients mode is more efficient in delivering the clients than the single client mode (it achieves smaller  $\overline{T}_n$  values in such conditions), and with many available vehicles the

single client mode is more efficient with regards to the average values of  $\overline{T}_n$ . Note, however, that even if the average times are smaller in the single client mode, they are still very similar, especially considering the uncertainties associated.

Observe that the map has a strong impact on the results, especially in the case of multiple clients mode's experiments. While the performance of the single client mode only suffers a change in scale, the multiple clients mode changes its behavior significantly, showing a proportionally much smaller increase in the total service time as the number of available vehicles decreases.

Figure 4 shows the total distance traveled for the same experiments. The expected behavior for the system would be that, the more vehicles there are in the system, the smaller the total combined distance they must travel to deliver all clients. This is observed in the single client mode. For multiple clients mode, however, that is not always the case. Multiple clients mode shows a surprising increase in the total traveled distance when the number of vehicles increases from  $n_v = 5$  to  $n_v = 8$ . Although perhaps counter intuitive at first, this result can be understood when one compares the behaviors of both modes of operation.

The analysis of both client waiting times and vehicles distances traveled leads to the conclusion that, with the increase in the number of vehicles, there is not much difference in the efficiency values of each mode. Therefore, in such cases, there is little to no ride sharing occurring in the system. It is the general perception that for 8 or more vehicles the modes behave essentially in the same way, for both maps considered. However, with less vehicles the system is under greater stress, and vehicles must each attend to more clients, increasing the chances that ride sharing occurs. With the increase of vehicles, the probability that there is a vehicle free to service a client straight away also increases.

With the decrease of ride sharing, it is also logical that the combined distances of all vehicles will increase. If a vehicle was to drive two clients at the same time, it optimizes the distance traveled to deliver the second client, due to the own nature of the implemented ride sharing functionality. If ride sharing is to occur, it will do so when the client is already in the route of the vehicle (and therefore there is no increase in distance traveled to pick this client up). Moreover, the total distance for this vehicle to deliver the client along the ride is smaller than the distance any other vehicle would have to travel.

This notion also corroborates the observation that there is little to no ride sharing occurring with a higher number of vehicles. Note that not only the graphs show very similar overall behavior for the different modes of operation for  $n_v \geq 8$  (see Figures 3 and 4)<sup>3</sup>, but also the average  $D_n$  comes to be even slightly higher for multiple clients mode than for single clients mode in the Dog map, for some of the scenarios tested (see Figure 4(b)). This behavior should not be observed when there is ride sharing occurring often.

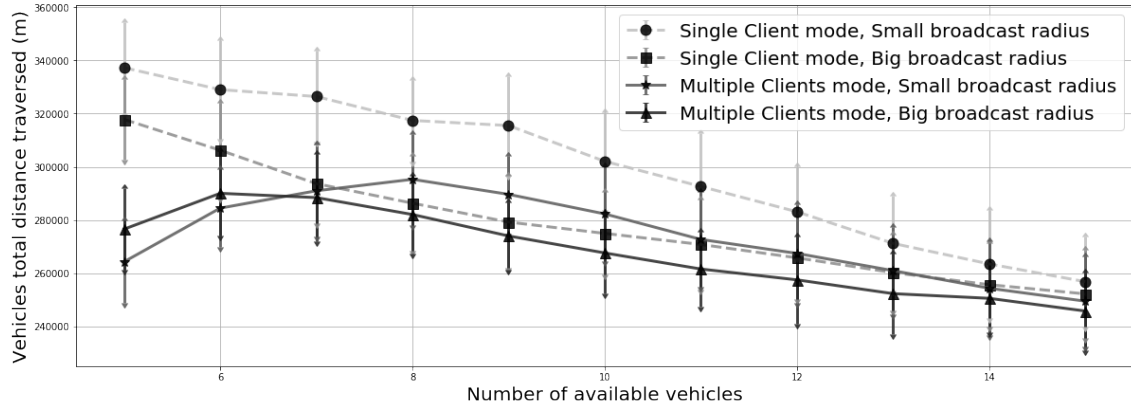
Another conclusion is that the map of available tracks and stations has a major influence in the system behavior. It is especially true in the multiple clients mode, which

---

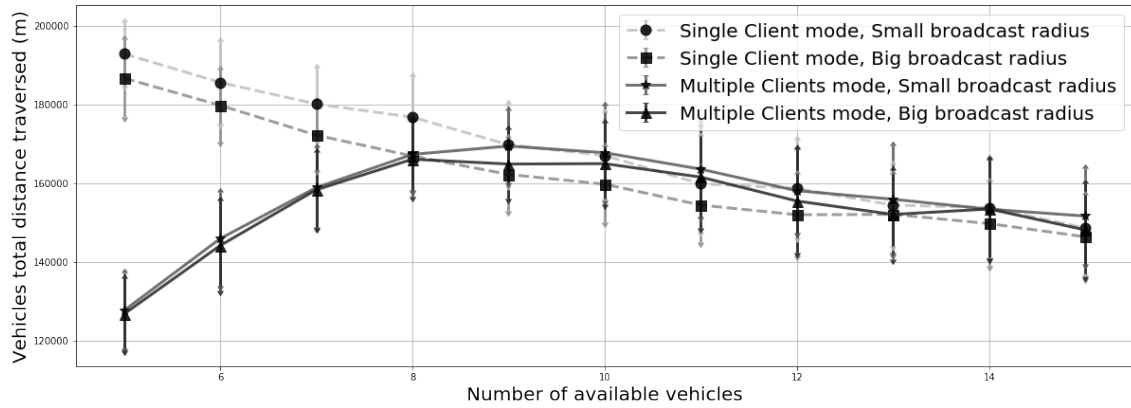
<sup>3</sup>The term *overall behavior* is used to take into account both the total service time and total traveled distance. Therefore this behavior is only truly similar between modes when  $n_v$  has 8 or more vehicles, despite the fact that the  $\overline{T}_n$  graphs (in figure 3) converge sooner.



## Average total traveled distance versus number of available vehicles



(a) Grid map. The big broadcast radius has 5657 m (full map reach from any position,  $r_{max}$ ), while the small one has 2263 m (40% of  $r_{max}$ ).



(b) Dog map. The big broadcast radius has 3523 m (full map reach from any position,  $r_{max}$ ), while the small one has 1409 m (40% of  $r_{max}$ ).

**Figure 4. Average total traveled distance  $D_{n_v}$ , with respect to the variation in the number of vehicles  $n_v$ , with the standard deviation portrayed in the error bars. The dashed lines correspond to single client mode results, while the solid lines represent the multiple clients mode results, each having two different broadcast radius values  $r$ , differentiated by color and marker.**

shows that the Dog map (represented in plots (b) of the figures) is much more favorable to the occurrence of ride sharing than the Grid map. This is understandable, for in that map there are much less stations available, with smaller distances and less paths in general. For this reason, it is more likely that when a client places a request, some vehicle will be near its station, and possibly even going to its destination as well, without altering its path. And thus the system can more easily deliver clients optimizing the distances vehicles need to travel.

### 3.2. Demand Intensity

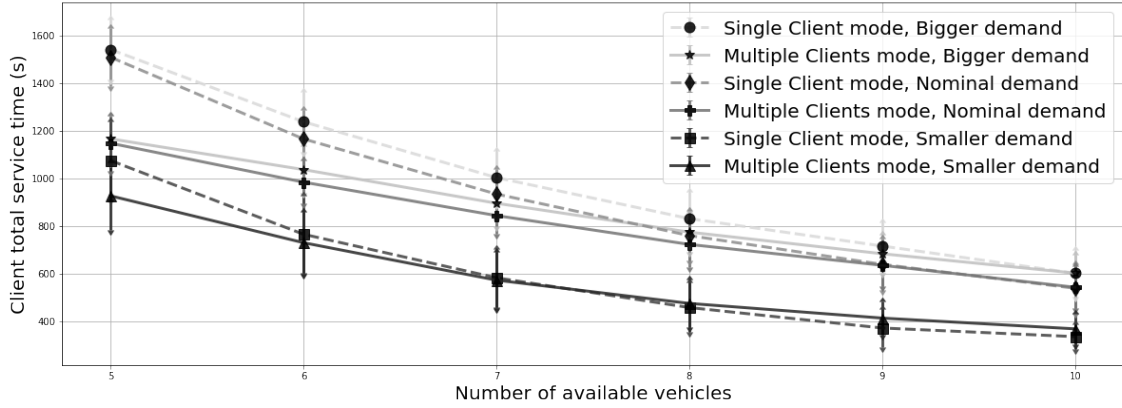
Although the experiments so far have used a constant and uniform distribution of the demand for the service, this is hardly representative of real operation conditions in a transportation system. It is expected that, if nothing else, the intensity of the demand would vary with time, so that in some periods there would be more clients requesting

services.

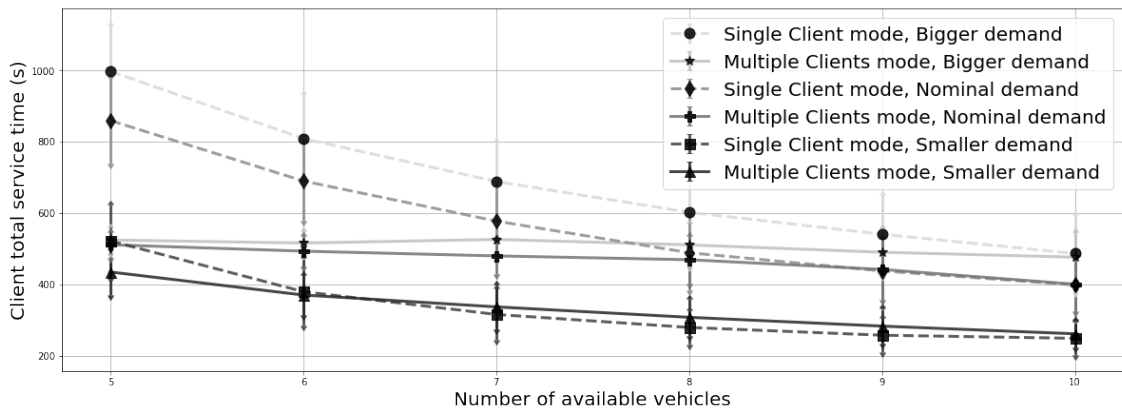
To evaluate how the system behaves under an increase and decrease of the rate of client appearance, further simulations were conducted. The following results compare the performance of the system for 3% and 6% probability of a client appearance per second in the system, where the previously used rate was 5%. Figures 5 and 6 show the average  $\overline{T}_n$  and  $D_n$ , respectively, for the two modes of operation, three different demand intensity rates and the two different maps.

As with the previous experiments, these were performed by simulating the appearance and delivery of 60 clients in the system ( $n = 60$ ), and their appearance is given by an arrival process that was generated a priori, guaranteeing that the different scenarios have the same client arrival patterns. For each scenario the simulation was repeated 40 times to account for statistics fluctuations (each with a different client arrival pattern).

### Average client total service time versus number of available vehicles



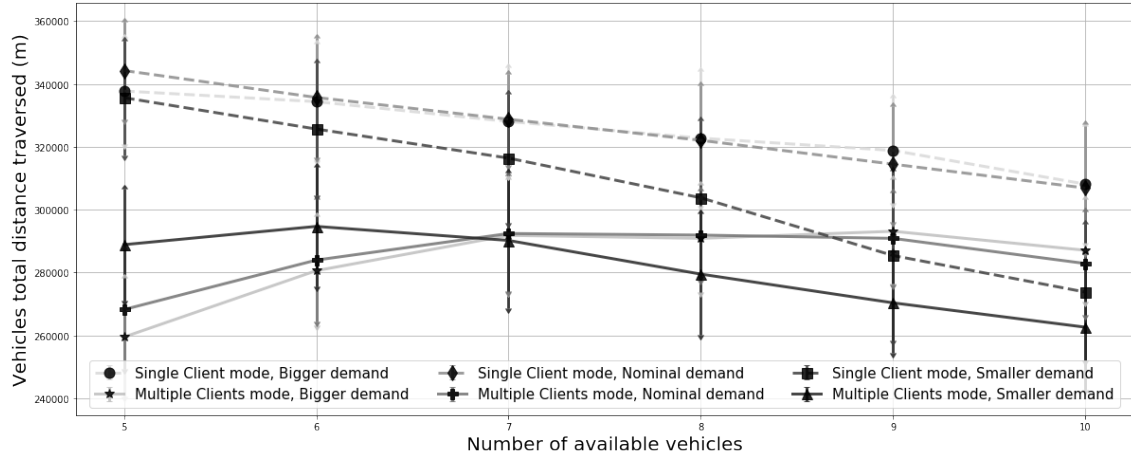
(a) Grid map. The bigger demand corresponds to the 6% probability rate, the nominal one to 5% and the smaller one to 3%.



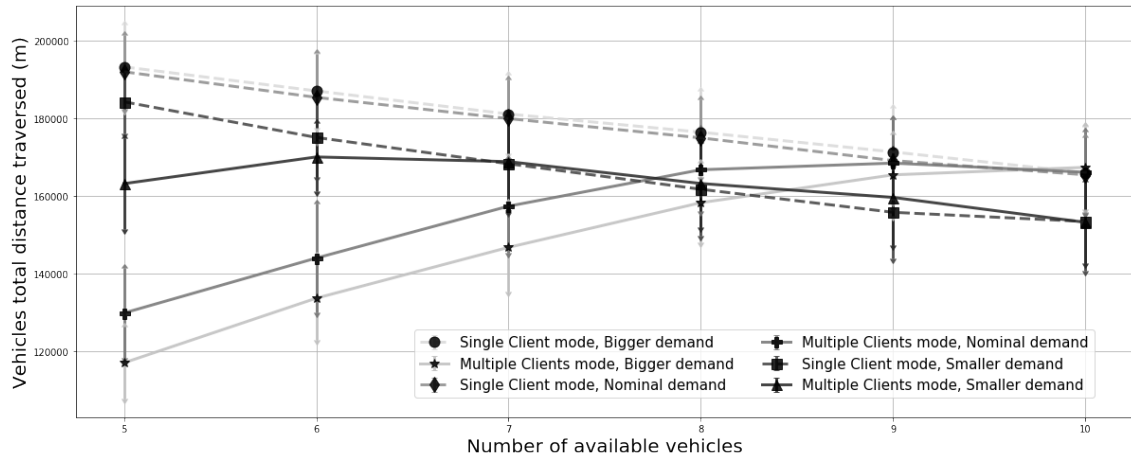
(b) Dog map. The bigger demand corresponds to the 6% probability rate, the nominal one to 5% and the smaller one to 3%.

**Figure 5. Average total service time  $\overline{T}_n$ , with respect to the variation in the number of vehicles  $n_v$ , with the standard deviation portrayed in the error bars. The dashed lines correspond to single client mode results, while the solid lines represent the multiple clients mode results, each having three different demand intensity rates, differentiated by color and marker.**

## Average total traveled distance versus number of available vehicles



(a) Grid map. The bigger demand corresponds to the 6% probability rate, the nominal one to 5% and the smaller one to 3%.



(b) Dog map. The bigger demand corresponds to the 6% probability rate, the nominal one to 5% and the smaller one to 3%.

**Figure 6. Average total traveled distance  $D_n$ , with respect to the variation in the number of vehicles  $n_v$ , with the standard deviation portrayed in the error bars. The dashed lines correspond to single client mode results, while the solid lines represent the multiple clients mode results, each having three different demand intensity rates, differentiated by color and marker.**

These results used a broadcast radius  $r$  of 40% of full map reach ( $0.4r_{max}$ ), maintaining the previous associations of  $r_c = r$  and  $r_v = 3r$ .

As expected, with the decrease in the demand the average  $\overline{T}_n$  also decreases, and clients take less time to arrive to their destinations (shown in Figure 5). However, note that the average  $D_n$  does not always follow the same pattern. In multiple clients mode, it is noticeable that the decrease in the demand generally increases the average total traveled distance by the vehicles.

The explanation for such a behavior is that when the system is under less stress (less clients per vehicle ratio), there is less ride sharing occurring, and therefore less route

optimization to deliver the clients. Moreover, a higher demand of clients per second means that the system is under more stress, the vehicles are under more duress, and there is a higher chance of ride sharing.

Again the Dog map shows this behavior with more emphasis, and it can be seen that the increase in the number of vehicles (and consequent decrease in the client-vehicle ratio) clearly leads to the convergence of multiple clients mode into the single clients mode curves, for both performance metrics (shown in Figures 5(b) and 6(b)). For 10 vehicles, there is no more sensible difference between the modes, which indicates that multiple clients mode functions just as single client does, without ride sharing.

In comparison, the Grid map shows a similar trend, but it is less pronounced than in the Dog map. Especially the average  $D_n$  graphs show that there is still ride sharing occurring even when the total service time graphs show barely any difference (see Figures 5(a) and 6(a)). Since this map is larger and has many more stations and longer routes between them, it is clear that there would be a need of more vehicles before they saturate the system into behaving like the single client mode. A similar argument also explains why the decrease in the demand intensity leads to closer single client and multiple clients mode curves. Even if multiple clients mode is still more efficient for the scenario of 5 vehicles, this improved efficiency is less pronounced when the system is under less stress, especially considering the  $\overline{T}_n$  values (see Figure 5).

#### 4. Conclusions

This work has presented a novel transportation system based on SAVs with distributed coordination, and two different modes of operation: exclusive service, named single client mode, and shared service, named multiple clients mode. The system was analyzed using a simulator designed and implemented for this transportation system [de Oliveira 2019]. Its performance was evaluated through the average total time it takes the system to deliver a client at its destination ( $\overline{T}_n$ ) and the distance traveled by vehicles to fulfill a group of client requests ( $D_n$ ), as well as how these efficiency parameters vary with the different system conditions. In particular, the two modes of operation were compared.

Results have consistently shown that when the system is under stress, that is, when it has a high client per vehicle ratio over time, the multiple clients mode presents a significantly better performance, decreasing both the total service time and the total traveled distance. These stress situations were observed when experiments were made with fewer available vehicles, or with an increase in the demand (rate of clients arriving per second). It was also observed that when the system has no stress, multiple clients mode functioned very similarly to single clients mode, and they tend to converge to having the same average performance.

Although the results were capable of portraying the general performance of the system, they are still preliminary. For example, the rate of ride sharing between clients was not measured, which would be interesting to validate the assumptions of the system. There is also no metric of the vehicle idleness throughout the simulations. It is expected that the more vehicles there are, the more time the vehicles stay idle. This is an important parameter when one wants to determine how many vehicles are needed in the system. Also no traffic has been taken into account, which can occur even if the pathways are exclusive for the system, if there are enough vehicles, or if there are intersections with other

transportation systems. Moreover, although a station-based approach has been chosen, it would be to its advantage to compare its performance with a system with no stations, where clients could board and disembark vehicles at any point.

Lastly, it should be pointed out that the proposed simple communication protocol has a scalability problem, since as the size of the map increases, the broadcast radius must also increase for the system to work. Note that the Grid map (Figure 1(a)) has larger  $r$  values than the Dog map (Figure 1 (b)). One solution would be to implement multi-hop broadcast communication, and artificially set a range around the client to select vehicles that would take part in any given election. Another solution is to divide the map into smaller zones, and assign each vehicle to work in a specific zone. Prior work [Miao et al. 2016] could be used to manage the vehicle distribution through zones, making the system scalable (and possibly more efficient) for large urban centers.

## References

- Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., and Rus, D. (2017). On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences*, 114(3):462–467.
- Berrada, J. and Leurent, F. (2017). Modeling transportation systems involving autonomous vehicles: A state of the art. *Transportation Research Procedia*, 27(1):215–221.
- Canca, D., Barrena, E., Algaba, E., and Zarzo, A. (2014). Design and analysis of demand-adapted railway timetables. *Journal of Advanced Transportation*, 48(2):119–137.
- de Oliveira, A. C. N. (2019). V.A. Project repository. Available at: <https://github.com/AmieOliveira/V.A..git>. Accessed in: 2019-03-12.
- Dijkstra, E. W. (1959). A note on two problems in connection with graphs. *Numerische Mathematik*, 1(1):269–271.
- Fourie, P. J., Jittrapirom, P., Binder, R. B., Tobey, M. B., Medina, S. O., Maheshwari, T., and Yamagata, Y. (2020). Chapter 6 - modeling and design of smart mobility systems. In *Urban Systems Design*, pages 163 – 197.
- Garcia-Molina, H. (1982). Elections in a distributed computing system. *IEEE Transactions on Computers*, c-31(1):48–59.
- Koopman, P., Ferrell, U., Fratrick, F., and Wagner, M. (2019). A safety standard approach for fully autonomous vehicles. In *Computer Safety, Reliability, and Security*, pages 326–332.
- Lam, A. Y. S., Leung, Y., and Chu, X. (2016). Autonomous-vehicle public transportation system: Scheduling and admission control. *IEEE Transactions on Intelligent Transportation Systems*, 17(5):1210–1226.
- Liu, J., Jones, S., and Adanu, E. K. (2019). Challenging human driver taxis with shared autonomous vehicles: a case study of chicago. *Transportation Letters*, 0(0):1–5.
- Mahdavi, A. and Carvalho, M. (2019). Distributed coordination of autonomous guided vehicles in multi-agent systems with shared resources. In *2019 SoutheastCon*, pages 1–7.

- Miao, F., Han, S., Lin, S., Stankovic, J. A., Zhang, D., Munir, S., Huang, H., He, T., and Pappas, G. J. (2016). Taxi dispatch with real-time sensing data in metropolitan areas: A receding horizon control approach. *IEEE Transactions on Automation Science and Engineering*, 13(2):463–478.
- van Dijke, J., van Schijndel, M., Nashashibi, F., and de la Fortelle, A. (2012). Certification of automated transport systems. *Procedia - Social and Behavioral Sciences*, 48(1):3461 – 3470.
- voytek (2014). Uber newsroom, optimizing a dispatch system using an AI simulation framework. <https://www.uber.com/newsroom/semi-automated-science-using-an-ai-simulation-framework>. Accessed in: 2019-11-01.
- Wagner, P. (2016). Traffic control and traffic management in a transportation system with autonomous vehicles. In *Autonomous Driving: Technical, Legal and Social Aspects*, pages 301–316.