Daena: International Journal of Good Conscience. A22.20(1)1-12. Enero-Junio 2025.
ISSN 1870-557X / ISSN 3065-7814

# Centralized Multiagent System Approach to Reduce Bus Bunching

M.S. Jesús Ángel Patlán Castillo, Dr. Francisco Torres Guerrero and Dr. Romeo Sánchez Nigenda

Universidad Autonoma de Nuevo León, Nuevo León, México
jesus.patlancstl@uanl.edu.mx
romeo.sanchezng@uanl.edu.mx
francisco.torresgrr@uanl.edu.mx

**Abstract.** The buses routes most of the times face spontaneous changes in the environment during its daily work: the traffic dynamism, sudden broke of a bus, the dynamism of the passengers arriving rates at a stop, etc. These changes provokes what is called the bus bunching problem, in which two or more buses are relatively too close to each other, leading to a lesser efficiency of the buses to pick up passengers on the stops. The bus bunching problems affects the daily life of numerous passengers everyday, affecting the performance of them in their daily activities in work. This paper addresses the bus bunching problem with a centralized multiagent system, in which these dynamism can be controlled by monitoring the environment and provides information to the buses to act according to the current state of the environment. The results show a positive impact on reducing the bunching between buses in the simulations provided.

**Keywords:** Bus Bunching Problem · Centralized Multiagent System · Linear Programming.

## 1 Introduction

In recent years, the use of public transportation has increased, as the population in cities and displacement from job centers to suburban areas has risen. The use of public transport in the US increased by 28% since 1995, which is more than the 23% of the population growth rate [4]. There are approximately 6800 companies providing public transportation services in the US, which have invested in research to increase the effectiveness of their resources to provide better performance and quality of service for their users. Metrics of performance rely on maximizing the number of passengers that can travel in a single transport unit and minimizing the traveling time that passengers take. One important study area of bus operating systems is the real-time control, which involves maintaining the bus operations along a period of time to minimize passenger inconvenience [3]. Buses in the routes are scheduled to arrive at each stop in each certain period of time, yet the changes of passengers flow, traffic, and other dynamics factors that the route has through the time impacts on the performance of the buses arriving rate in each stop, and even provokes that the buses becomes too close too each other, producing a low arriving rate of the buses on the stops and a poor performance of the buses overall. The problem when two or more buses are relatively too close to each other is what it is known as the bus bunching problem. The impact that the bus bunching has on the route increases as buses get closes to each other in a specific part of the route, since the amount of passengers on other parts of the route increases and when a single bus passes by the stops in these highly demanded parts it can't pick up all the passengers waiting, increasing the passengers waiting time [11]. A high passenger waiting time impacts on the performance of the public transport users, because it provokes that employees, students, and all the public transport users a worse performance in their daily activities. This is why the bus bunching problem has become a well known problem that has been addressed using different strategies to reduce the headway between buses:

– **Bus holding** This strategy involves holding the buses in certain stops to adjust the headways of nearby buses. Programming models has been used to obtain the holding times of the buses [**?**,9].
– **Speed regulation** By the increasing or decreasing of the bus speed, the buses try to maintain a headway tolerance range [7].
– **Skip stop** The bus skip a stop to reduce the headway between the next bus and reduce it from the rear bus [14,5,10].
– **Deadheading** By marking a stop with a deadhead, buses tends to skip the bus a certain amount of times to keep their speed to reach other stops. The stops marked with deadheading tends to have a low passengers arriving rate [6,8].

These strategies have all positive impacts on reducing the bus bunching in the route, nevertheless most of the mathematical models solutions that uses these strategies do not take into account the dynamic factors that arises in the route through the time, specifically, it does not support emergent plans to deal with these spontaneous problems that occurs in the route. Besides, since the bus bunching problem is part of the real time control area, the impact of these strategies increases when they are applied in the needed time, which is complex to model mathematically in the needed time as the instance problem of the model takes time to be solved and to be applied by the buses on the routes.

Since the bus bunching problem involve all these dynamic factors in the environment through time, we decide to solve it by using a multiagent system. Multiagent system is defined as a system which involve the interaction of multiple intelligent agents that interact to solve a problem [12]. Each agent has the capability to act and react to the environment changes by applying the best (based on their perspective) action to solve the problem, in the case of bus bunching problem, to reduce the bunching between the buses in the route. The use of multiple agents helps to deal with the dynamism of a bus route because each agent can react with emergent plans to deal with bus bunching. Since it is possible to develop emergent plans to multiple scenarios, multiagent system can simulate a great variety of scenarios that can happen in the route in a real case scenario.

There has been approaches using multiagent system: using bus holding strategy with multiagent reinforcement learning [1], a distributed multiagent system involving stop agents and buses agents to establish the bus holding time [15], an adaptive control scheme to cooperate between nearby buses to perform a speed regulation [2]. There has been a preference in the use of multiagent system to face the bus bunching problem, since multiagent system addresses the problem with an environment that simulates the dynamism that the buses routes faces.

However, there has not been a multiagent system which incorporates multiples strategies to face the bus bunching problem. A multiagent system allows the use of an agent which can generate the plans for the buses to maintain an acceptable headway between the route with multiple strategies.

There are two kinds of multiagent system architecture: a centralized multiagent system develops plans that allows all the agents of the system to act collectively, meanwhile a decentralized, often called distributed multiagent system, allows that some groups of agents to develop plans by communication and this permits that the whole system can act more freely between groups of agents (or even individually for each agent) [13]. However, we decide to use a centralized architecture since groups of agents (represented as buses in the system) may be separated between them, but now between all the agents of the route, which can generate bunching between them in the route if there is not a ”leader” (which we will call the Control Point Agent) that can moderate all the agents or group of agents in the route.

## 2 Methodology

A public bus system consists of a route of buses, which is a set of buses each one with a capacity between 36-60 seats and a speed that varies between 50km/h to 90km/h depending on the avenue that travels. Every bus travels through a set of stops that have a different flow of passengers that enters and leaves the buses that pass by them. All buses keep traveling through the stops until finishing the working day. To simulate these characteristics,in the simulation we have a set of buses $B$ each one with a maximum capacity $c$ of passengers. Since bus speed varies between avenues, we use a relative value for the speed, this is, the speed of the bus varies between 0-100%. We have a set of stops $S$, each one with a different arriving and descending rate of passengers. Each stop is separated between a given distance, and between each stop there is a speed limit to simulate the possible traffic in the route and the maximum speed limit that each avenue has. It is possible to allow a circular route, meaning that the bus can continue their travel at the last stop by continuing on the first stop. There is also the possibility to allow buses to overtake or not. With these two options, it is possible to run instances as a bus rapid transit model or use a common public transport bus service.

All of these characteristics can be configurable to create different instances and test the multiagent system performance. The centralized multiagent system have two types of agents:

- **Control Point Agent** The control point agent (CP-Agent) is in charge of planning the actions that the bus agents must perform to reduce the headway.
- **Bus Agent** The bus agent (B-Agent) is aboard on every bus, its function is to inform the control point agent about its current state: speed, passengers aboard and position. It performs the actions that the control point agent requests. We call these ”Dummy B-Agent” since it does not think rationally about the actions that the CP-Agent request to do.

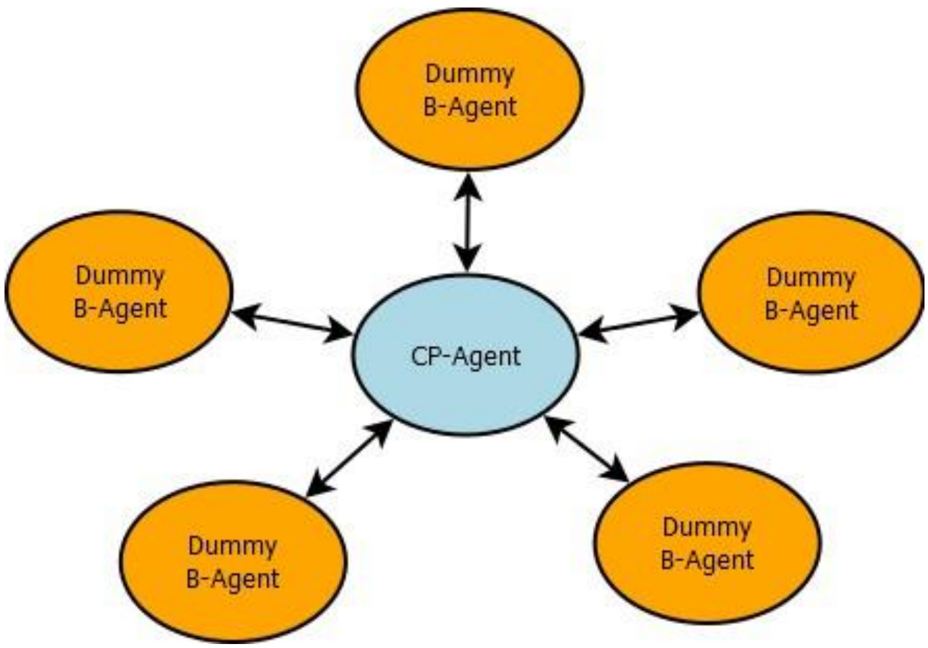



Fig. 1: Centralized Multiagent Communication flow

The communication flow between agents is represented graphically in figure 1, where the B-agents and the CP-agents sends message between them for the CP-agent to develop a better plan based on the current environment characteristics. This is an important characteristic that can adapt more to a real case of a bus system, since in the real case no one can know the real current state of the route until it is reported someone. We represent each B-agent as a dummy since they don't reason about the instructions that are given from the CP-agent, they just execute the actions commanded by it.

The process that the control point agent follows to plan is based on a headway tolerance range (HTR), in which the agent catalogues every bus based on its position to specify an action to perform. The headway tolerance range is relative to the distance between the rear and the front bus of every bus, with a percentage that can vary between 0 and 50%. The actions that the control point agent commands are specified in figure 2.

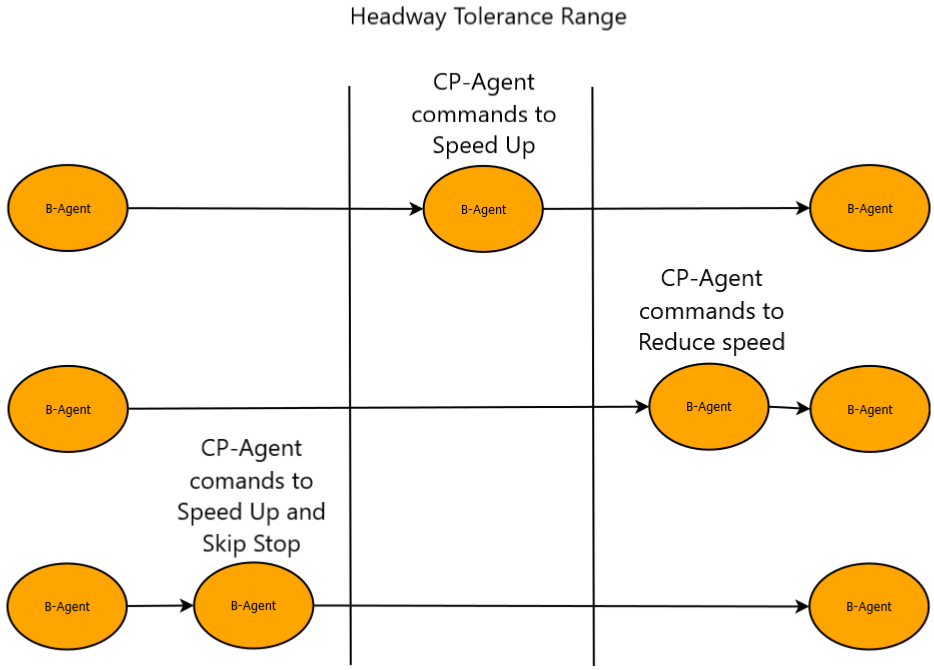



Fig. 2: Control point agent planning

When the position of the B-Agent is too close to the rear bus, the CP-Agent commands it to increase its speed by 10% and to skip the next stop if possible. If the position of the bus is in the headway tolerance range, then the B-Agent is asked to increase its speed to reach the next stop as fast as possible. Finally, if the B-Agent is too close to the front bus, then it is requested to reduce it speed by 10%. Additionally, the CP-Agent performs an specified number of calls to a linear programming model to determine if any bus should perform a bus holding on the next stops. The model used in this paper was modelled by Citlali Olvera in its work to solve a bus rapid transit model [9]. We use also a relaxed version of this model by removing the restriction of overtake between buses to solve the instances in which overtaking is allowed.

The call of the bus holding solver and the simulation of the bus route environment is asynchronous, this is, the CP-Agent calls for the solver at time $t_0$, the solution is received at time $t_0 + t_i$, and when received by the CP-Agent, it makes the request for every bus to perform the bus holding if specified by the solution.

## 3 Experimental study

The implementation of the multiagent system was developed in Java, using the libraries of Jason for the multiagent environment and Gurobi as the solver for the linear programming model for the bus holding. The simulations where tested in a computer with the following characteristics:

- **Processor** Intel i7 8550U
- **RAM** 8GB
- **Storage** Solid State Disk 512GB

We used 3 kinds of instances: 2 for a bus rapid transit (BTR) and 1 for a public bus transport service (PBT), this is, 2 instances in which overtaking is not allowed and 1 where overtaking is. We used dummy data for the 1 of the BTR and the PBT, since we obtained information about a real case of a BTR used in Nuevo León called "Ecovia" (Thanks to Citlali Maryuri for sharing the data). Between these 3 instances we variate the HTR between 10%, 20% and 30% to analyze if the planning algorithm of the CP-Agent affects the performance of the route. The parameters of the instances we used are the number of stops in the route, the number of buses in the route, each one having a maximum capacity of 75 passengers, the number of calls to the bus holding solver during the simulation, the bus alight, which is the time that takes to a passenger to abord the bus, the bus dwell which is the time that takes to a passenger to descend from the bus, if overtake and/or circular route was enabled in the instance and the HTR percentage that the instance has. We use the average headway obtained during all the simulation to represent the efficiency of the multiagent system to coordinate their action to reduce the bus bunching through the route. The simulation is prepared to simulated the rate of passengers that arrive at each stop using a Poisson distribution with a defined mean, however, to analyze these instances we decided to run a previous simulation of every instance to save the values of the number of passengers that arrive at each stop and with this we used these same values for every instance, with the idea that these probabilistic distribution doesn't interfere with the results of the simulation. Tables 1, 2 and 3 resume the results.

| | | | |
|---|---|---|---|
| **Stops** | 10 | 10 | 10 |
| **Buses** | 7 | 7 | 7 |
| **Bus holding solver calls** | 15 | 15 | 15 |
| **Bus Alight** | 0.15 | 0.15 | 0.15 |
| **Bus Dwell** | 0.25 | 0.25 | 0.25 |
| **Overtake** | TRUE | TRUE | TRUE |
| **Circular** | TRUE | TRUE | TRUE |
| **HTR** | 10% | 20% | 30% |
| **Average Headway** | 4.103 | 2.613 | 1.724 |

Table 1: Experiment 1: Public Bus Transport

In this table 1 we can see a clear difference in the average headway between instance with HTR of 10% and HTR of 30%, showing us that the HTR has a great impact on the simulation of public bus transport systems.

| | | | |
|---|---|---|---|
| **Stops** | 10 | 10 | 10 |
| **Buses** | 7 | 7 | 7 |
| **Bus holding solver calls** | 15 | 15 | 15 |
| **Bus Alight** | 0.15 | 0.15 | 0.15 |
| **Bus Dwell** | 0.25 | 0.25 | 0.25 |
| **Overtake** | FALSE | FALSE | FALSE |
| **Circular** | TRUE | TRUE | TRUE |
| **HTR** | 10% | 20% | 30% |
| **Average Headway** | 3.882 | 3.517 | 2.213 |

Table 2: Experiment 2: Bus Rapid Transit

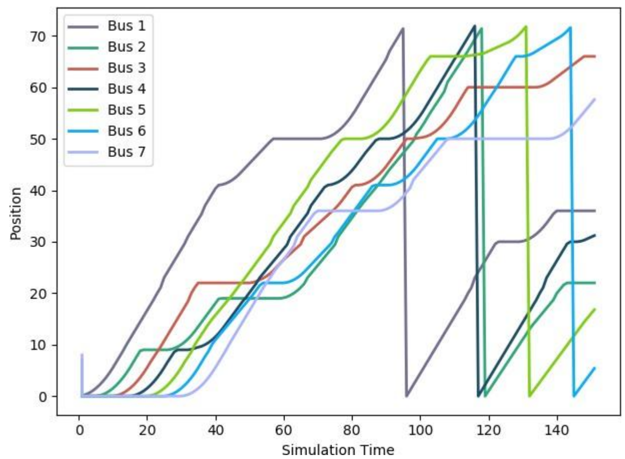
Table 2 shows also a significant impact in the simulation when we use a lower percentage of HTR in the instances of bus rapid transit.

| | | | |
|---|---|---|---|
| **Stops** | 10 | 10 | 10 |
| **Buses** | 7 | 7 | 7 |
| **Bus holding solver calls** | 15 | 15 | 15 |
| **Bus Alight** | 0.15 | 0.15 | 0.15 |
| **Bus Dwell** | 0.25 | 0.25 | 0.25 |
| **Overtake** | FALSE | FALSE | FALSE |
| **Circular** | TRUE | TRUE | TRUE |
| **HTR** | 10% | 20% | 30% |
| **Average Headway** | 9.35 | 9.314 | 8.157 |

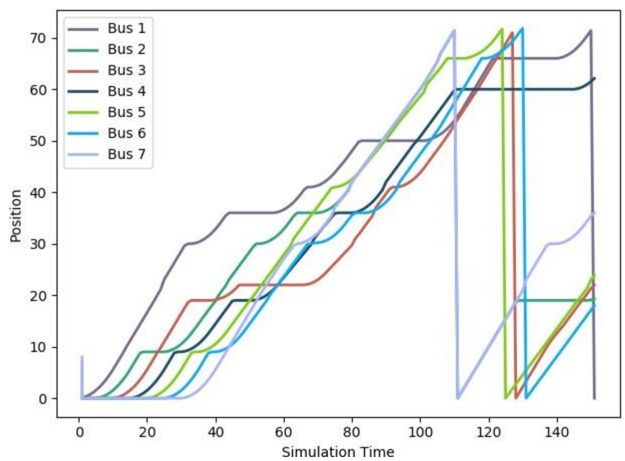
Table 3: Experiment 3: Bus Rapid Transit Ecovia

Using real data from a bus rapid transit, the table 2 shows too that the HTR has a positive impact on the route, which is an important result considering we are using and simulating a real case scenario.

From these results, we can observe that the average headway between the buses in the route during the simulation increases the lower the headway tolerance range is, implying that the planning from the CP-Agent was effective to reduce the bus bunching between the buses. The figures 3a and 3b represents the behaviour of the buses during the simulation of the PBT, figures 4a and 4b represents it for the BRT, and figures 5a and 5b for the Ecovia simulation.

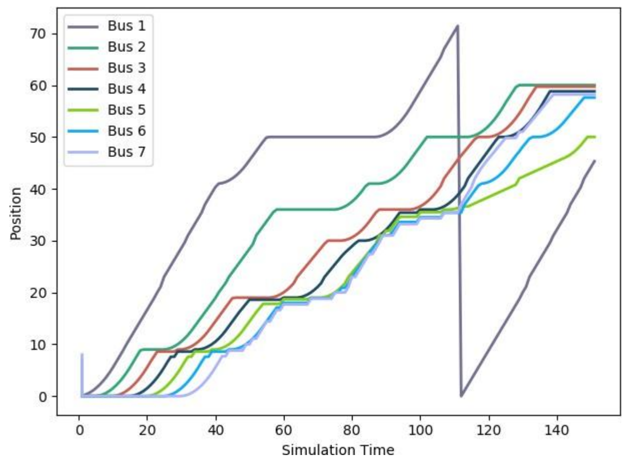



(a) PBT: 10% HTR

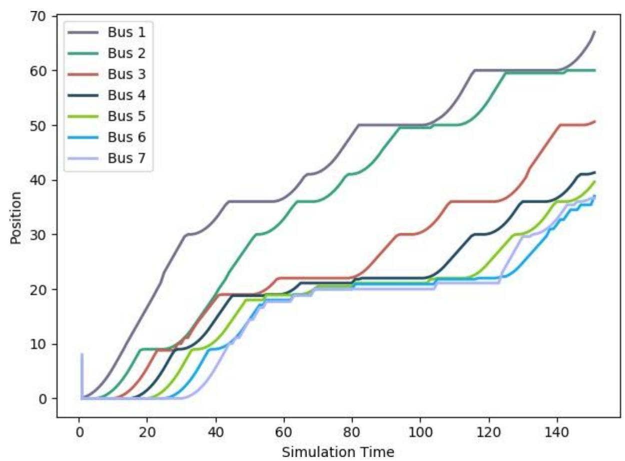



(b) PBT: 30% HTR

In figure 3a we can observe that the buses tends to separate between each other more as times passes when we consider a HTR of 10%, compared with the results shown by the simulation of 30% HTR in figure 3b. We can observe that at the end of the simulation, buses 2, 3, 5 and 6 are too close to each other in figure 3b, provoking the bus bunching problem, which is not present in the figure 3a.

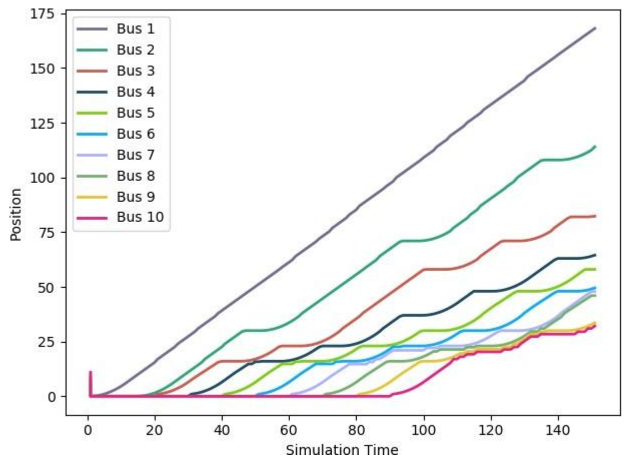



(a) BRT: 10% HTR

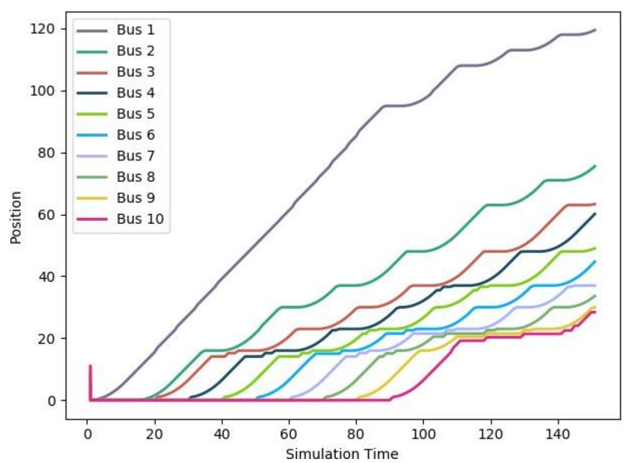



(b) BRT: 30% HTR

In figure 4a we can observe that bus 1 travels around all the stops by time 100 and by the end of the simulation it reaches at position 40, which implies that the HTR of 10% allows it to pass through some of the stops to pick up passengers that are waiting on stops that are far away from the initial stop. Other buses do the same process of skipping some stops, and by the end of the simulation the others buses are around position 60. Compared to the results with a HTR 30%, we observe in figure 4b that no bus passed through all the stops by the end of the simulation, and most of the buses reach only around the position 30, provoking that these buses stay together for almost all the simulation time.




(a) ECOVIA: 10% HTR




(b) ECOVIA: 30% HTR

In these both figures 5a and 5b, we can see a combination of the results shown in the previous instances. In the case of the instance with a HTR of 10%, we can see that buses reaches to a farest position compared to the buses in the instance of 30% HTR, and we can notice also that buses tends to be more separated in the instance with figure 5a.

As seen in the results from the tables, the graphs also shown better performance of the routes when having a lower HTR.

## 4 Conclusion

The use of a centralized multiagent system had a positive impact in the performance of the buses in the route by applying multiple strategies, specially by having a shorter tolerance headway range. The algorithm used to plan the buses routes by the control point agent is effective even if we consider the dynamism of the environment.

## 5 Future work

With this centralized multiagent system, we are developing a distributed multiagent system by allowing the buses agents to perform actions with the real state of the local environment they can observe. With this new strategy that the bus can develop, the control point agent will still command the agents action with the information that it has of the environment, but the bus agent might not perform the action that it received, since it can develop a better plan according to the real state of the environment.

## References


1. Chen, C., Chen, W., Chen, Z.: A multi-agent reinforcement learning approach for bus holding control strategies. Advances in Transportation Studies (2015)
2. Daganzo, C.F., Pilachowski, J.: Reducing bunching with bus-to-bus cooperation. Transportation Research Part B: Methodological **45**(1), 267–277 (2011)
3. Desaulniers, G., Hickman, M.D.: Public transit. Handbooks in operations research and management science **14**, 69–127 (2007)
4. Dickens, M.: American public transportation association: 2021 public transportation fact book (May 2021), https://www.apta.com/wp-content/uploads/APTA-2021-Fact-Book.pdf
5. Fu, L., Liu, Q., Calamai, P.: Real-time optimization model for dynamic scheduling of transit operations. Transportation research record **1857**(1), 48–55 (2003)
6. Furth, P.G.: Alternating deadheading in bus route operations. Transportation science **19**(1), 13–28 (1985)
7. He, S.X., Dong, J., Liang, S.D., Yuan, P.C.: An approach to improve the operational stability of a bus line by adjusting bus speeds on the dedicated bus lanes. Transportation Research Part C: Emerging Technologies **107**, 54–69 (2019)
8. Liu, Z., Yan, Y., Qu, X., Zhang, Y.: Bus stop-skipping scheme with random travel time. Transportation Research Part C: Emerging Technologies **35**, 46–56 (2013)
9. Olvera Toscano, C.M.: Modelo matemático e implementación de un simulador basado en un sistema de transporte urbano. Ph.D. thesis, Universidad Autónoma de Nuevo León (2018)
10. Sun, A., Hickman, M.: The real–time stop–skipping problem. Journal of Intelligent Transportation Systems **9**(2), 91–109 (2005)
11. Verbich, D., Diab, E., El-Geneidy, A.: Have they bunched yet? an exploratory study of the impacts of bus bunching on dwell and running times. Public Transport **8**(2), 225–242 (2016)

12. Weiss, G.: Multiagent systems: a modern approach to distributed artificial intelligence. MIT press (1999)
13. Xuan, P., Lesser, V.: Multi-agent policies: from centralized ones to decentralized ones. In: Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 3. pp. 1098–1105 (2002)
14. Xuan, Y., Argote, J., Daganzo, C.F.: Dynamic bus holding strategies for schedule reliability: Optimal linear control and performance analysis. Transportation Research Part B: Methodological **45**(10), 1831–1845 (2011)
15. Zhao, J., Bukkapatnam, S., Dessouky, M.M.: Distributed architecture for real-time coordination of bus holding in transit networks. IEEE Transactions on Intelligent Transportation Systems **4**(1), 43–51 (2003)