# Linear Bus Holding Model for Real-Time Traffic Network Control

**Leonardo G. Hernández-Landa, Miguel L. Morales-Marroquín, Romeo Sánchez Nigenda and Yasmín Á. Ríos-Solís**

**Abstract** One of the most annoying problems in urban bus operations is *bus bunching*, which happens when two or more buses arrive at a stop nose to tail. Bus bunching reflects an unreliable service that affects transit operations by increasing passenger-waiting times. This work proposes a linear mathematical programming model that establishes bus holding times at certain stops along a transit corridor to avoid bus bunching. Our approach needs real-time input, so we simulate a transit corridor and apply our mathematical model to the data generated. Thus, the inherent variability of a transit system is considered by the simulation, while the optimization model takes into account the key variables and constraints of the bus operation. Our methodology reduces overall passenger-waiting times efficiently given our linear programming model, with the characteristic of applying control intervals just every 5 min.

## 1 Introduction and Problem Description

The study of complex bus operating systems is usually divided into two main areas, *line planning* and *real-time control* [3, 8]. The *line planning* process involves strategic, tactical, and operational decisions. Strategic problems relate to long-term network design decisions. Tactical and operational decisions ultimately define the service offered to the public; for example, frequency of buses, definition of stops, bus

---

L.G. Hernández-Landa · M.L. Morales-Marroquín ·
R.S. Nigenda · Y.Á. Ríos-Solís (✉)
Graduate Program in Systems Engineering, Universidad Autónoma de Nuevo León (UANL), San Nicolás, Mexico
e-mail: yasmin.riossls@uanl.edu.mx

M.L. Morales-Marroquín
e-mail: morales.mike@gmail.com

R.S. Nigenda
e-mail: romeo.sanchezng@uanl.edu.mx

L.G. Hernández-Landa
e-mail: leogabrielhdz@gmail.com



M. Mujica Mota et al. (eds.), *Applied Simulation and Optimization*,
DOI 10.1007/978-3-319-15033-8_10

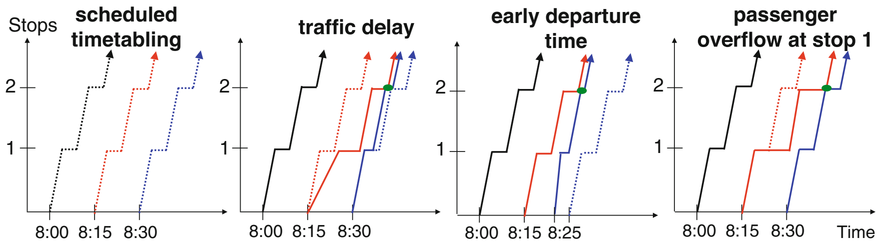



**Fig. 1** Causes of bus bunching (modified from Ceder [3])

timetabling, vehicle scheduling, driver scheduling, maintenance scheduling, among other problems.

*Real-time control* tries to maintain the bus system operational along the day in order to minimize passenger inconvenience caused by the inherent stochastic dynamics of the network or traffic situations [8]. Although bus frequency is planned for each stop in the network, changes in the passenger flow, traffic, or even in the timetabling, produce perturbations that give rise to one of the most annoying problems in urban transportation operations, the *bus bunching problem* (BBP) that happens when two or more buses arrive at a stop nose to tail. BBP is one of the most common customer complaints in today's networks since it reflects an unreliable service that affects transit operations by increasing passenger-waiting times.

In Fig. 1, we show the causes of bus bunching for a single bus line with three trips, which have the following timetable: 8:00, 8:15, and 8:30. For the four graphs, time is represented by the x-axis, while the first two stops are represented by the y-axis. The first graph shows how the planning should look like if everything were deterministic. We can see that the lines of the three trips are *parallel*, so the time differences between them (called headways) are of exactly 15 min. The second graph shows the perturbations that arise when a traffic delay hits the second trip between the depot and the first stop. The dotted lines are the planned schedules, while the plain lines are the real executed delayed plans. Since the 8:15 bus takes longer to arrive at stop 1, there are more passengers waiting to board it. When the bus that departed at 8:30 arrives at stop 1, many of the passengers that should have boarded it have already boarded the 8:15 bus. Then, these two buses will bunch close to stop 2. Graph three represents bunching situations when the departure time of a trip is moved earlier. Similarly to the second case, there will be less passengers at stop 1 so the bus will go faster and catch the 8:15 bus around stop 2. Finally, the fourth graph considers the case of passenger overflow. This graph shows that since there are extra passengers at stop 1 the dwell time of the second bus at that stop will be longer. In other words, the second bus is taking passengers who would be normally assigned to the third bus. By the time the second and third buses arrive at stop 2, they are generating a bus bunching situation.

In this work, we provide solutions to the *bus bunching problem* by maintaining *congruent headways*. Furthermore, we show that maintaining congruent headways implicitly reduces passenger-waiting times. As mentioned, the headway is a quality measure given to the time difference between two consecutive buses. A bus line could have equally distant headways or different ones for each pair of buses Ceder [2], Ibarra-Rojas and Rios-Solis [14], Ibarra-Rojas et al. [15]. We say that *headways are congruent* if the real-time differences between buses are nearly identical to the originally planned. Headway congruence does not necessarily comply with planned timetables. Indeed, the time when a bus arrives at a stop may not be the planned one, but if the distance to its predecessor is almost the planned headway, then it will be a congruent headway. Congruent headways reflect a reliable service, especially for cases when timetables are not intended for the public so the users only know estimated headways for the lines as in Monterrey, Mexico, and many Latino-American cities.

Our methodology interleaves optimization and real-time data retrieving to maintain congruent headways and solve BBP along the day. During the optimization phase, a linear programming model is built and solved to exactly determine the holding times of the buses at the stops in order to maintain congruent headways. The real-time data retrieving phase indicates, at every interval of time, the positions of the buses along a single corridor where only one line operates at a given frequency. In Fig. 2, we can observe how optimization and real-time data retrieving interleave. Real-time (or simulated) data are acquired from the bus corridor to obtain the distance between each bus and its last visited stop, together with the number of passengers waiting at each stop. Then, these data are used to populate our linear programming model, which yields the optimal holding times for each bus in the corridor.

Most of the works in the literature base their quality measure on the waiting times of the passengers, or the variance between the departure times of the buses at the stops, which are generally modeled with quadratic functions that are harder to solve and therefore difficult to operate by real-time systems. By using a linear objective function that minimizes the penalties arising when headways are not congruent, our methodology returns optimal solutions in a short time. One of the main contributions of this work is that by maintaining congruent headways, we implicitly reduce the overall passenger waiting and travel times, as our experimental results will demonstrate.

The remainder of this chapter is structured as follows. A brief revision of the state of the art is presented in Sect. 2. In Sect. 3, we present our new linear programming

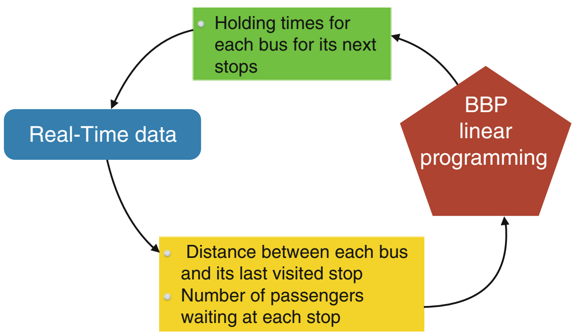



**Fig. 2** Framework for interleaving optimization (BBP LP modeling) and real-time data retrieving (or simulation)

model inspired in earliness and tardiness penalties of just-in-time scheduling problems, which determines the optimal holding times of the buses at the stops. Then, Sect. 4 shows the efficiency of our model on a discrete event simulation of a single corridor. Finally, Sect. 5 presents our conclusions, and discusses open research questions that arise from this work.

## 2 State of the Art Research in Real-Time Bus Operations

Most of the literature related to real-time bus operations deals with models that have nonlinear objective functions. Therefore, the holding times that each bus must be held at the stations are approximations. Work by Zhao et al. [24] minimizes the average waiting cost of passengers, including both off-bus and on-bus costs that are nonlinear, when there is no capacity imposed on the buses. Eberlein et al. [11] minimize the variance between the departure times, which is a quadratic function, and therefore propose heuristic solutions. Sun and Hickman [23] propose a convex quadratic programming problem to minimize the variance between the departure times. A closer work to ours is proposed by Ding and Chien [10], since they consider the minimization of the total variance of headways between buses at all stops.

Daganzo [4] and Daganzo and Pilachowski [5] propose adaptive control schemes aiming to provide quasi-regular headways, while maintaining as high commercial speed as possible. In Daganzo and Pilachowski [5] the authors continuously adjust bus cruising speed based on a cooperative two-way-based approach that considers the headways of the previous and later buses. Bartholdi III and Eisenstein [1] abandon the idea of any *a priori* target headway, allowing headways to dynamically self-equalize by implementing a simple holding rule at a control point. It is worth noting that the aim of the previously mentioned studies is to maintain headways equally, so they do not consider timetables where the headways may be different for each pair of buses and they are not apt for situations when the buses reach their capacities.

Our work deals with the capacity of the vehicles as Zolfaghari et al. [25] do, where the authors minimize the waiting time of passengers at every stop by taking into account the variance between the departure times. These authors propose heuristics to circumvent the complexity of the proposed model. Puong and Wilson [18] propose a nonlinear mixed-integer linear programming for a real-time disruption response model with emphasis on the train holding strategy. In Delgado et al. [6, 7] the aim is to minimize the total waiting times experienced by passengers in the system using a quadratic model.

Our work aims at maintaining congruent headways considering capacity of the vehicles, and in doing so, we expect to reduce passenger-waiting times in the bus corridors. We improve the work of Delgado et al. [7] by reducing the number of variables in the model and the number of times the model is used in real-time scenarios, obtaining exact solutions for the holding times. Moreover, in order to reduce the waiting times of the passengers we bound the holding times of the buses. Another

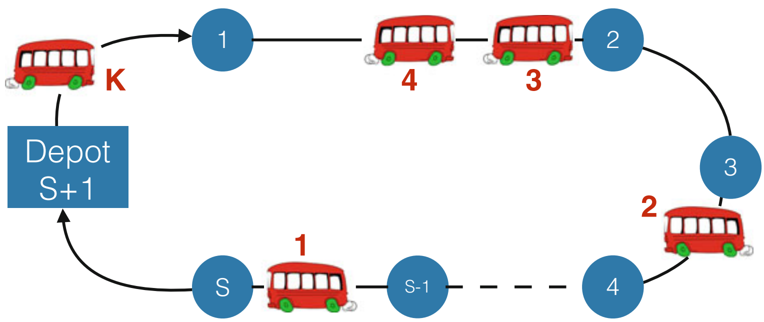



**Fig. 3** Transit bus line model: each bus $k$ leaves the depot according to an established timetable, serving stops 1 to $S$ before coming back to the depot where all the remaining passengers must alight

advantage of our proposal is that it adapts easily to cases where the headways are equal or different during different planning horizons along the day.

## 3 Methodology and Approach

As mentioned earlier, the core of our methodology consists of interleaving optimization and real-time data retrieving of the bus lines in a rolling horizon planning. The optimization phase of our approach builds and solves efficiently a linear model to maintain congruent headways along the bus line. Our model is used at every given time interval[1] to decide how long the buses should be held at the bus stops. Our model requires a real-time data estimation of the state of the system to operate. Such data are provided by the real-time retrieving phase, which in our case of study is supported via simulation. The simulation of the system provides data related to the position of the buses, number of passengers aboard each bus, and the number of passengers waiting at the stops to build our model.

More precisely, the Bus Bunching Problem, BBP, consists of $K$ buses, each with its own capacity and speed that serve all $S$ stops of a single bus corridor. We can see in Fig. 3 that each bus $k$ leaves the depot according to an established timetable, serving stops 1–$S$ before coming back to the depot where all the remaining passengers must alight. Notice that overtaking is not permitted. For the optimization phase, we consider that travel times between stops, and $\lambda_s$ (passengers arrival rate per minute) are deterministic during the period of interest. Moreover, each stop has a dwell time function depending linearly on the number of passengers that board ($boardT$ minutes per passenger).

The characteristics of the line are as follows. Parameter $cap_k$ corresponds to the capacity of bus $k$, $dist_s$ is the distance in meters between stops $s$ and $s-1$, $speed_{ks}$

[1] The time interval is a parameter in our model that could be specified by the control unit of the bus company.

is the operating speed in meters per minute of bus $k$ between stops $s$ and $s-1$ while the bus is moving, and $OD_{kss'}$ is the fraction of passengers boarding bus $k$ at stop $s$ whose destination is stop $s'$ (for all $s < s'$). The headway between buses $k$ and $k-1$ in this line must be between the interval $[minHead_k, maxHead_k]$ to be considered congruent, which is specified as an input parameter for our model.

At time $t^0$, instant when the holding decisions are needed, we assume that we have the following state of the transit corridor:

- $d_k^0$ distance between bus $k$ and its last visited stop at time $t^0$. If the bus is still at a stop, then $d_k^0 = 0$.
- $s(k)$ indicates the last stop that bus $k$ has visited at time $t_0$. If bus $k$ is at stop $s'$, then $s(k) = s' - 1$. In Fig. 3, $s(2) = 3$ and $s(3) = 1$, and to simplify the notation, $s(K) = 0$, but $s(1) + 2 = S + 1$.
- $c_s^0$ is the number of passengers waiting at stop $s$ at time $t^0$.

Decision variables of our model are the holding times for each bus $k$ at control point $s$, denoted by $h_{ks}$. There are auxiliary variables that depend on $h_{ks}$, like the departure times of bus $k$ at stop $s$ that is denoted as $td_{ks}$. If the departure times at stop $s$ of buses $k$ and $k-1$ are between $[minHead_k, maxHead_k]$, then we consider that they are complying with the established headways. Nevertheless, if this difference in departure times is outside this interval, we use the concepts of *earliness* and *tardiness* which is frequent in just-in-time scheduling theory [19–22]. The *earliness of the headway* between buses $k$ and $k-1$ at stop $s$ is defined as $E_{ks} = \max(minHead_k - (td_{ks} - td_{k-1s}), 0)$ which can be linearized as follows:

$$E_{ks} \geq minHead_k - (td_{ks} - td_{k-1s}),\ k = 2, \ldots, K, s = s(k)+1, \ldots, S \quad (1)$$

$$E_{ks} \geq 0,\ k = 2, \ldots, K, s = s(k)+1, \ldots, S. \quad (2)$$

While the *tardiness of the headway* is $T_{ks} = \max((td_{ks} - td_{k-1s}) - maxHead_k, 0)$ which is equivalent to

$$T_{ks} \geq (td_{ks} - td_{k-1s}) - maxHead_k,\ k = 2, \ldots, K, s = s(k)+1, \ldots, S \quad (3)$$

$$T_{ks} \geq 0,\ k = 2, \ldots, K, s = s(k)+1, \ldots, S. \quad (4)$$

Then, the objective function of BBH is the minimization of the sum of all early and tardy headways:

$$\min \sum_{k=2}^{K} \sum_{s=s(k)+1}^{S} \psi E_{ks} + \epsilon T_{ks}, \quad (5)$$

where $\psi$ and $\epsilon$ are linear penalization for the earliness and the tardiness, respectively, subject to constraints (1)–(4). Additionally, the departure times of each bus $k$ at each stop $s$ are defined with two different sets of restrictions. The first one is the case where the bus $k$ at time $t^0$ is between stops $s(k)$ and $s(k)+1$ (in Fig. 3 this case would apply for bus 2 that is between stops 3 and 4). Here, the departure time of $k$

at $s(k)+1$ is the time that needs the bus to arrive at the stop, plus the dwelling time $dwell_{ks(k)+1}$ (that will be computed later) plus the time the model decides that this bus will hold. This situation is reflected by constraints (6). The second case is similar but considers that the bus has not yet reached stop $s-1$ (constraints (7)). Restrictions (8) impose a limit of $maxHold$ to each holding time to guarantee a certain traveling time quality of the passengers.

$$td_{ks(k)+1} = t_0 + \frac{dist_{s(k)} - d_k^0}{speed_{ks(k)}} + dwell_{ks(k)+1} + h_{ks(k)+1}, \quad k \in K \tag{6}$$

$$td_{ks} = td_{ks-1} + \frac{dist_{s-1}}{speed_{ks-1}} + dwell_{ks} + h_{ks}, \quad k \in K, s = s(k)+2, \ldots, S-1. \tag{7}$$

$$h_{ks} \leq maxHold, \quad k \in K \backslash \{1\},\ s = s(k)+1, \ldots, s(k-1). \tag{8}$$

From the state variables of the system, we can compute the total number of passengers that will be at stop $s$ when bus $k$ will reach this stop, denoted as $pass_{ks}$ in (9) and (10), as the number of passengers who are actually in the stop plus the ones that will arrive. The number of passengers who will be in bus $k$ at stop $s$ is equal to the passengers who want to board bus $k$, $pass_{ks}$, minus the proportion of the passengers that left the bus before stop $s$ (restrictions (11)). In this manner, we can compute the dwell times of bus $k$ at $s$ (restrictions (12)). Notice that alighting and friction between the passengers who stay inside the bus could be easily included in the last restriction set.

$$pass_{ks} = c_s^0 + \lambda_s (td_{ks} - t_0), \quad k \in K,\ s = s(k)+1, \ldots, s(k-1) \tag{9}$$

$$pass_{1s} = c_s^0 + \lambda_s (td_{1s} - td_{Ks}),\ s = s(K)+1, \ldots, S \tag{10}$$

$$passBus_{ks} = \min\left( \sum_{i=1}^{s-1} pass_{ki} \left(1 - \sum_{j=i+1}^{s-1} OD_{kij}\right), cap_k \right), \quad k \in K,\ s = s(k)+1, \ldots, S \tag{11}$$

$$dwell_{ks} = passBus_{ks} boardT, \quad k \in K, s = s(k)+1, \ldots, S. \tag{12}$$

The following restrictions are the different cases that need to be considered in order to avoid bus overtaking:

$$td_{ks} - td_{k-1s} \geq 0, \quad k \in K \backslash \{1\},\ s = s(k-1)+1, \ldots, S \tag{13}$$

$$td_{1s} - td_{Ks} \geq 0, \quad s = s(k)+1, \ldots, s(1) \tag{14}$$

$$td_{k-1s} - td_{ks} \geq 0,\ k \in K \backslash \{1\},\ s = s(k)+1, \ldots, s(k-1). \tag{15}$$

The LP for BBP is then

$$\begin{aligned} \min \quad & \textstyle\sum_{k=2}^{K} \sum_{s=s(k)+1}^{S} \psi E_{ks} + \epsilon T_{ks} \\ \text{s.t.} \quad & \\ & E_{ks}, T_{ks}, h_{ks} \geq 0, \quad k \in K, s \in S. \end{aligned}$$

Notice that all variables are required to be positive but not integer, so LP can be solved by the simplex method or by a polynomial barrier algorithm. Indeed, the main variables $h_{ks}$ represent a time interval so we can consider them as continuous variables. One of the main advantage of LP, besides the fast computational time, is that we could use linear programming sensitivity analysis. Nevertheless, the holding times that are going to be transmitted to the drivers at the bus stations should be integer. Then, variables $h_{ks}$ should be in seconds or in minutes and therefore integer variables. Preliminary results showed no drastic increment in the computing times when bus holding variables $h_{ks}$ are integer [1, 4, 6, 15].

Our model improves and differs from the model of Delgado et al. [7] in the following aspects.

- Our objective function is linear so we can obtain optimal solutions for our model.
- The departure times of the buses are according to their established headway or timetable. Only perturbations that arise along the trip are taken into account.
- We only take into account the possible holding times of a bus from its actual position up to the depot instead of considering the holding times for all stops. This reduces the number of variables and makes the problem more realistic.
- We bound the amount of time that a bus can be held at a stop.
- We may have different headways for every pair of buses. In this way, recent synchronization timetables can be benefited by our approach and dealing with different planning periods (e.g., rush hour, night time) is natural.
- We do not need to call the model every time a bus arrives at a stop, we can do it at each fixed interval of time. This fact is more realistic for a bus company. In our case of study, the company retrieves data of the buses every 2 min.

## 4 Experimental Results

The BBP LP model described in the previous section needs data to be populated. Data can be retrieved through the use of monitoring technologies like Global Positioning Systems (GPS) and Automatic Vehicle Location systems (AVL) in real-time during the execution of the bus corridor. However, to study the impact of our model under different scenarios in the traffic corridor we consider a discrete event simulation.

The single corridor is simulated using the discrete event and stochastic simulator ExtendSim AT version 9.0 [9, 17]. The simulator triggers an event at every fixed amount of time, in which the positions of the buses and their loads, and the passengers waiting at the stops, together with their traveling destinations, are updated.

Our BBP LP model uses deterministic functions to forecast demands and travel times. Nevertheless, we use stochastic processes in the simulation to reflect a real system. We use a single corridor of 10 km with 30 stops and one depot uniformly distributed, like in Delgado et al. [7]. There are only 30 stretches, since the last stop is merged with the depot. Travel times of the buses between each pair of stops are distributed as Lognormal with a mean of 0.77 min and variance of 0.4 [13, 24]. At each stop, passengers arrive randomly using a Poisson distribution with rate equal to one [16]. The mean of the distributions are the parameters used by our model.

When passengers arrive at a bus stop, a destination is assigned to them. Passengers wait in line to board the bus in a first-in/first-out manner. Boarding and alighting times of passengers are set to 2.5 and 1.5 s respectively, since all buses have two doors, one for boarding and another for alighting. If passengers cannot enter a bus because it reached its capacity, they will wait in the stop until the next bus with free space arrives. This waiting time is denoted as $W_{first}$. The headway time windows are set to $[minHead_k, maxHead_k] = [0.3, 0.46]$ minutes for all the buses. Note that these time windows are easily adjustable for cases where there are different periods along the day, and for the synchronization timetables that favor transfers. We can measure the waiting and travel times of the passengers and the buses in the simulation since we have modeled these structures as individual agents.

We use a fleet of 60 buses with a maximum capacity of 100 passengers per bus. At every fixed amount of time *interval*, we determine the actions that should be followed by creating the BBP LP model in Java, and solving it with the linear package of Gurobi 5.6. The solution generated contains the holding times for all the buses for all the future stops up to the depot. If after a time *interval* a new solution is generated, then the holding times are updated using a rolling horizon scheme.

Even if we base our scenarios on the ones generated by Delgado et al. [7], there is no fair comparison since our methodologies consider different assumptions. Nevertheless, we can observe that our approach indeed improves the overall waiting and travel times of the passengers.

The scenarios for the simulation are divided into two parts: *time interval* scenarios and the *parameters setting* scenarios; and they are described in the following subsections.

## 4.1 Time Interval Scenarios

The aim of the time interval scenarios is to determine the optimal policy for controlling when new holding times must be computed and given to the system.

In our case study for the city of Monterrey, México, the bus company updates at every 2 min the positions and all the related data of the buses in the transit corridor. Following this policy, Table 1 shows the time interval scenarios in which we test our approach. The first column in Table 1 identifies the scenarios while the second column sets the time intervals (in minutes) in which our BBP LP model is constructed and solved to introduce the resulting holding times to the system. We vary these control

**Table 1** *Time interval* scenarios with earliness and tardiness penalties $\psi = \epsilon = 1$

| Scen | Control (min) | $maxHold$ (min) | $W_{first}$ (min) | Travel (min) | Pass | $W_{first}$/ pass | Travel/ pass |
|---|---|---|---|---|---|---|---|
| $TI_0$ | ✗ | ✗ | 1798.0 | 12035.8 | 1713.3 | 1.0 | 7.0 |
| $TI_1$ | 2 | ✗ | 1115.88 | 17045.40 | 1703.1 | 0.66 | 10.01 |
| $TI_2$ | 5 | ✗ | 1136.92 | 18256.20 | 1746.2 | 0.65 | 10.45 |
| $TI_3$ | 7 | ✗ | 1222.66 | 18907.13 | 1705.8 | 0.72 | 11.08 |
| $TI_4$ | 10 | ✗ | 1362.68 | 18652.68 | 1708.5 | 0.80 | 10.92 |
| $TI_5$ | 2 | 0.38 | 1219.52 | 13112.15 | 1721.4 | 0.71 | 7.62 |
| $TI_6$ | 5 | 0.38 | 1330.89 | 13171.48 | 1737.4 | 0.77 | 7.58 |
| $TI_7$ | 7 | 0.38 | 1463.25 | 12851.01 | 1725.6 | 0.85 | 7.45 |
| $TI_8$ | 10 | 0.38 | 1424.52 | 12450.34 | 1697.7 | 0.84 | 7.33 |

values from 2 to 10 min. Scenario $TI_0$ does not have any control, and we use it as a baseline to compare the performance of our BBP LP model. The third column is an indicator if restriction (8) is applied; that is, if the holding times are bounded. For these scenarios, we set the earliness and the tardiness penalties $\psi = \epsilon = 1$. The fourth column, $W_{first}$, corresponds to the total average waiting time (in minutes) of a passenger to board a bus. The fifth column (Travel) represents the total average travel time of passengers in minutes, while the column Pass indicates the average number of passengers in the system during the simulation time. The last two columns indicate the normalized waiting and travel times of each passenger.

Ten simulation runs were executed for every scenario, each of them corresponding to one hour of bus operations. Each run has the same initial conditions initialized with random numbers. At the beginning of the simulation the buses are placed evenly spaced along the corridor. For each simulation run, we let the system evolve freely for 5 min before making any holding. Indeed, 5 min is enough to observe several bus bunching situations to arise.

We observe an increase in the passenger riding time, and potentially operation costs because of the introduction of holding times in the corridor. This behavior is expected, and in concordance with other works [12]. Nevertheless, the passenger-waiting times for the first bus are always reduced, which in fact is what we wanted to show in the first place. Indeed, by controlling the headway we can also control the passenger-waiting times, without the need for using a quadratic objective function in the model.

We can also observe that the best passenger-waiting times are for cases where the holding controls are applied every 2–5 min, and without the bounds on the holding times. However, the bounds on the holding times induce a reduction in the travel times, which is an important asset. Figure 4 shows the differences in performance when the control (8) ($maxHold$) is applied. It shows the percentage of increase in the passenger-waiting times when bounds are applied and the percentage of increase in the travel times when they are not applied. As mentioned, we observe that even if there

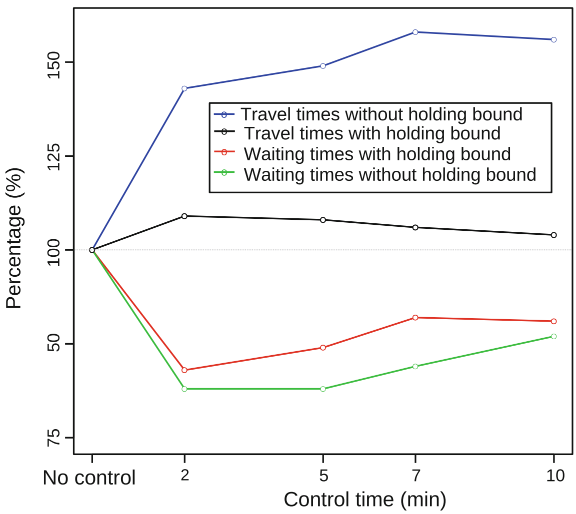



**Fig. 4** Decrease in the waiting times and increase in the travel times for the *time interval* scenarios with earliness and tardiness penalties $\psi = \epsilon = 1$ and applying bound to holding time

is an increase in the passenger waiting times when the holding times are bounded, the benefit on the passenger travel times is considerable. Then, maintaining congruent headways reduces the overall travel time of passenger along the whole network.

For a bus company, the less the traffic controller has to give holding orders to the system (i.e., to the bus drivers), the better. Therefore, from Table 1 and Fig. 4, we conclude that the best policy is to consider bounds on the holding times, and apply the controls to the system at every 5 min, like in the $TI_6$ scenarios.

In Fig. 5, we show two histograms of the length of the holding times (x-axis in minutes) for the time interval scenarios with earliness and tardiness penalties

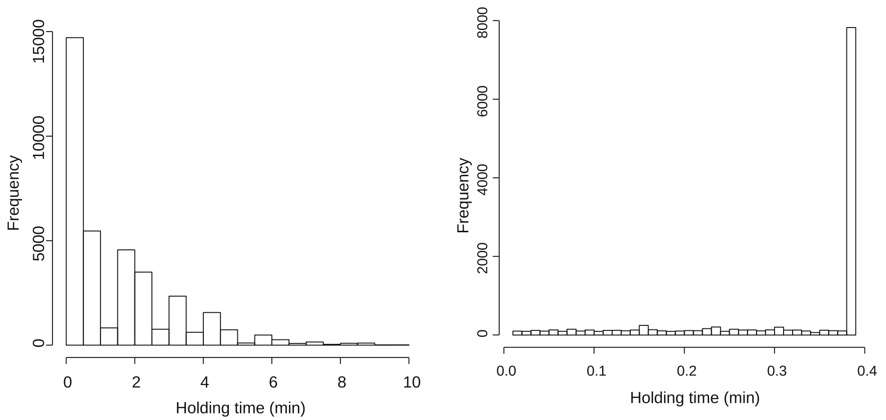



**Fig. 5** Holding times histogram without bounds (*left* histogram) and with bounds (*right* histogram) for the *time interval* scenarios with earliness and tardiness penalties $\psi = \epsilon = 1$, and a control of 5 min

$\psi = \epsilon = 1$, and a control of 5 min with and without bounds on the holding times. On the y-axis, we have the frequency the BBP LP model is called for all the simulations of class $TI_6$. Notice that not all of the holding times are applied, since the rolling horizon may modify several of them. The case when there are limits on the holding times shows that the model either chooses to apply the holding times close to these limits, or not to apply them at all. This is an implicit benefit for the users, and for the traffic controller.

The aim of the BBP model is to reduce bus bunching by maintaining congruent headways. To graphically show that this behavior is being improved by our model, we present Figs. 6, 7, 8 and 9 for the scenarios with bounds on the holding times. The x-axes in these graphs correspond to time (in minutes), while the y-axes represent stops. Each line in these graphs represents a bus that departs from the depot and cruises all the bus stops. Recall from Sect. 1 (see Fig. 1) that in the ideal case, we

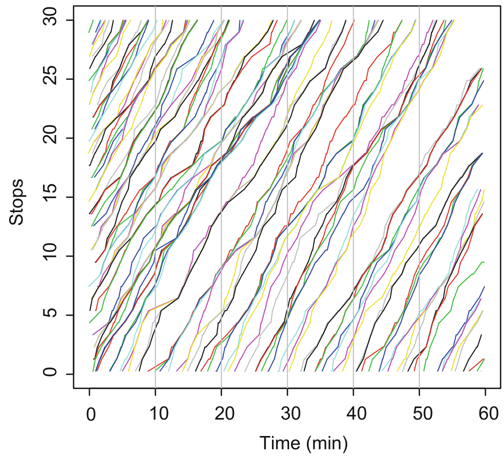



**Fig. 6** Bus transit behavior without control

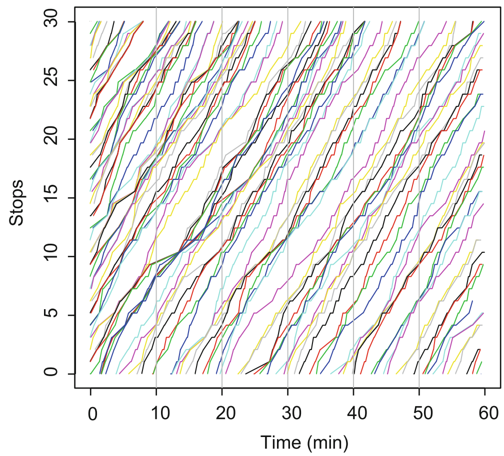



**Fig. 7** Transit with control every 2 min

**Fig. 8** Transit with control every 5 min

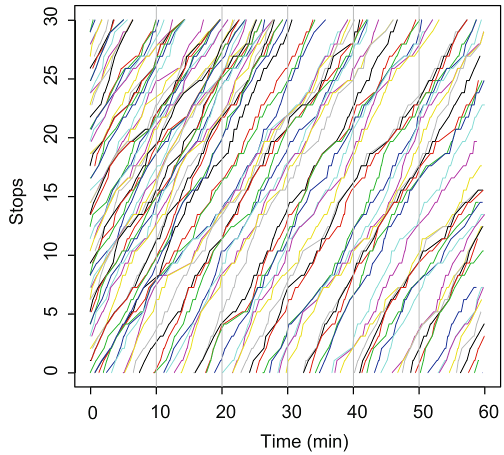
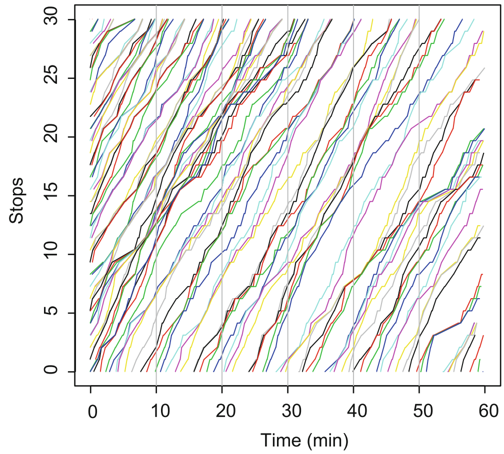



**Fig. 9** Transit with control every 7 min




would have *parallel* lines. Figure 6 displays the case without control and shows that the simulation makes a stochastic scenario. Here the bus bunching problem is notorious, since there are white gaps between the lines. Figures 7, 8 and 9, have time interval controls of 2, 5, and 7 min, respectively. We can observe that with 2 and 5 min controls the BBP is reduced, while for control intervals of 7 min the BBP appears again.

Figure 10 shows two histograms that have in their x-axes the round time of a bus trip. An aspect that we noticed from Table 1 is that the travel times increase with the BBP model. This is obvious because the BBP model introduces holding times for the buses in the corridor. Nevertheless, Fig. 10 shows that the standard deviation when BBP is applied every 5 min (right histogram) is reduced with respect to the case where no controls are used (left-hand side histogram).

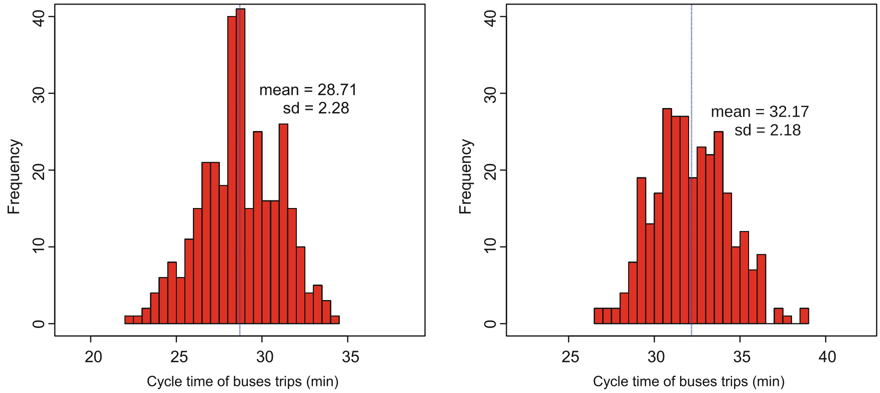



**Fig. 10** Histogram of travel cycle without control (*left*) and with control interval of 5 min (*right*)

## *4.2 Parameter Setting Scenarios*

Our next set of experiments modify the earliness $\psi$ and tardiness $\epsilon$ parameters of the BBP LP objective function to observe the impact they have in the passenger-waiting times and travel time. We can see this set of experiments in Table 2. The first column in the table identifies the scenarios. Ten simulation runs were considered per scenario. The second column represents the values of the earliness parameter, while the third corresponds to the tardiness one. The column "Board" denotes the average time (in seconds) a passenger takes to board a bus, while $maxHold$ stands for the time (in minutes) that the holding times are bounded. This table shows the percentage of reduction in passenger-waiting times ($W_{first}$), and the percentage of increase in the travel times (Travel). Finally, the last column represents the addition of the last two values. Indeed, if there is a reduction in this last column, the percentage would be negative.

An interesting observation from these results is that if we reduce the earliness parameter, we obtain the best results with respect to the passenger-waiting and travel times. Moreover, the BBP LP model yields better results when the holding times are limited by 0.19 min, which is also a quality asset for the user.

A statistical analysis confirms the observations from Table 2. The most influential parameters are the earliness penalty and the $maxHold$ limit. In Table 3, we show a linear regression of the parameters studied in this section. The first column is the parameter, the second corresponds to the "Estimate", the third is the standard error, the fourth stands for the t value, and the fifth one is the significance.

**Table 2** Improvement in the behavior of waiting time and travel time managing parameters

| Scen | $\psi$ | $\epsilon$ | Board (sec) | $maxHold$ (min) | $W_{first}$ % reduction (%) | Travel % increase (%) | $W_{first}$ + Travel % increase (%) |
|---|---|---|---|---|---|---|---|
| $P_1$ | 0 | 1 | 1.25 | 0.19 | 19 | −2 | −4 |
| $P_2$ | 0 | 1 | 1.25 | 0.38 | 22 | 1 | −2 |
| $P_3$ | 0 | 1 | 2.5 | 0.19 | 22 | −4 | −6 |
| $P_4$ | 0 | 1 | 2.5 | 0.38 | 25 | −1 | −4 |
| $P_5$ | 0.5 | 1 | 1.25 | 0.19 | 34 | 10 | 4 |
| $P_6$ | 0.5 | 1 | 1.25 | 0.38 | 55 | 39 | 27 |
| $P_7$ | 0.5 | 1 | 2.5 | 0.19 | 37 | 11 | 5 |
| $P_8$ | 0.5 | 1 | 2.5 | 0.38 | 56 | 41 | 28 |
| $P_9$ | 1 | 0 | 1.25 | 0.19 | 40 | 11 | 5 |
| $P_{10}$ | 1 | 0 | 1.25 | 0.38 | 59 | 42 | 29 |
| $P_{11}$ | 1 | 0 | 2.5 | 0.19 | 44 | 12 | 5 |
| $P_{12}$ | 1 | 0 | 2.5 | 0.38 | 63 | 48 | 34 |
| $P_{13}$ | 1 | 0.5 | 1.25 | 0.19 | 36 | 10 | 4 |
| $P_{14}$ | 1 | 0.5 | 1.25 | 0.38 | 54 | 41 | 28 |
| $P_{15}$ | 1 | 0.5 | 2.5 | 0.19 | 39 | 11 | 5 |
| $P_{16}$ | 1 | 0.5 | 2.5 | 0.38 | 57 | 40 | 27 |
| $P_{17}$ | 1 | 1 | 1.25 | 0.19 | 39 | 11 | 5 |
| $P_{18}$ | 1 | 1 | 1.25 | 0.38 | 48 | 27 | 17 |
| $P_{19}$ | 1 | 1 | 2.5 | 0.19 | 39 | 56 | 43 |
| $P_{20}$ | 1 | 1 | 2.5 | 0.38 | 57 | 50 | 36 |

**Table 3** Linear regression on the main parameters of the BBP model

| | Estimate | Std. error | t value | Pr(> \|t\|) |
|---|---|---|---|---|
| (Intercept) | 1.0314 | 0.0928 | 11.11 | 0.0000 |
| $\psi$ | −0.2234 | 0.0489 | −4.57 | 0.0004 |
| $\epsilon$ | 0.0273 | 0.0489 | 0.56 | 0.5848 |
| Board | −0.0845 | 0.0646 | −1.31 | 0.2106 |
| $maxHold$ | −0.2956 | 0.0646 | −4.57 | 0.0004 |

## 5 Concluding Remarks

In this paper, we presented a methodology based on interleaving optimization and real-time retrieving data to maintain congruent headways in a bus corridor with the aim of solving one of the most annoying problems in public transit networks, the Bus Bunching Problem (BBP).

During the optimization phase of our approach, a linear programming model is built and solved to determine the optimal holding times of the buses at the stops to avoid bus bunching. Our model requires real-time data of the state of the system to operate. Such data is provided by the real-time retrieving phase of our approach, which in our case is supported via simulation. The simulation phase of the system provides data related to positions of the buses, number of passengers in the buses, current bus capacities, and number of passengers waiting at the stops to build our model.

One of the main advantages of considering simulation in our methodology is the evaluation of multiple parameters to assess their impact in our BBP linear programming model. Therefore, we presented a comprehensive evaluation of such parameters, and found that applying holding controls just every 5 min, and bounds on the holding times reduce not only bus bunching frequency but also passenger-waiting times.

We also discussed that most of the works in the literature minimize passenger waiting times, or the variance in the departure times of the buses using quadratic optimization functions, which are more complex to solve. Instead, the linear programming model of our approach makes it suitable for returning optimal solutions efficiently and for interleaving the optimization and real-time retrieving data phases in real-time scenarios.

Although we observe an increase in the travel time of passengers given the introduction of holding times for the buses in the corridor, our approach performs better (i.e., less passenger-waiting time and acceptable travel time) than not introducing any control into the system. A part of our future work will consider the introduction of other actions into our models to reduce the travel time of the passengers in the corridor and lower operational costs. Particularly, we believe that the introduction of bus overtaking actions (i.e., skipping stops) will balance the total time a passenger spends in the system.

**Acknowledgments** L.G. Hernández-Landa and M.L. Morales-Marroquín wish to thank the Mexican National Council of Science and Technology (CONACyT) for graduate scholarship support.

## References

1. Bartholdi III JJ, Eisenstein DD (2012) A self-coördinating bus route to resist bus bunching. Transportation Research Part B: Methodological 46(4):481–491
2. Ceder A (2001) Bus timetables with even passenger loads as opposed to even headways. Transportation Research Record: Journal of the Transportation Research Board 1760(1):3–9
3. Ceder A (2007) Public Transit Planning and Operation: Theory, Modeling and Practice. Elsevier, Butterworth-Heinemann
4. Daganzo CF (2009) A headway-based approach to eliminate bus bunching: Systematic analysis and comparisons. Transportation Research Part B: Methodological 43(10):913–921
5. Daganzo CF, Pilachowski J (2011) Reducing bunching with bus-to-bus cooperation. Transportation Research Part B: Methodological 45(1):267–277

6. Delgado F, Muñoz JC, Giesen R, Cipriano A (2009) Real-time control of buses in a transit corridor based on vehicle holding and boarding limits. Transportation Research Record: Journal of the Transportation Research Board 2090(1):59–67
7. Delgado F, Munoz JC, Giesen R (2012) How much can holding and/or limiting boarding improve transit performance? Transportation Research Part B: Methodological 46(9):1202–1217
8. Desaulniers G, Hickman M (2007) Public transit. Transportation, Handbooks in Operations Research and Management Science pp 69–127
9. Diamond B, Krahl D, Nastasi A, Tag P (2010) Extendsim advanced technology: integrated simulation database. In: Proceedings of the Winter Simulation Conference, Winter Simulation Conference, pp 32–39
10. Ding Y, Chien SI (2001) Improving transit service quality and headway regularity with real-time control. Transportation Research Record: Journal of the Transportation Research Board 1760(1):161–170
11. Eberlein XJ, Wilson NH, Bernstein D (2001) The holding problem with real-time information available. Transportation science 35(1):1–18
12. Furth PG, Muller TH (2007) Service reliability and optimal running time schedules. Transportation Research Record: Journal of the Transportation Research Board 2034(1):55–61
13. Hickman MD (2001) An analytic stochastic model for the transit vehicle holding problem. Transportation Science 35(3):215–237
14. Ibarra-Rojas OJ, Rios-Solis YA (2012) Synchronization of bus timetabling. Transportation Research Part B: Methodological 46(5):599–614
15. Ibarra-Rojas OJ, López-Irarragorri F, Rios-Solis YA (2015) Multiperiod synchronization bus timetabling. Transportation Science
16. Jolliffe J, Hutchinson T (1975) A behavioural explanation of the association between bus and passenger arrivals at a bus stop. Transportation Science 9(3):248–282
17. Krahl D (2009) Extendsim advanced technology: discrete rate simulation. In: Winter Simulation Conference, Winter Simulation Conference, pp 333–338
18. Puong A, Wilson NH (2008) A train holding model for urban rail transit systems. In: Computer-aided Systems in Public Transport, Springer, pp 319–337
19. Ríos-Mercado RZ, Ríos-Solís YA (2012) Just-in-time Systems, vol 60. Springer
20. Rios-Solis YA (2008) Scheduling with earliness-tardiness penalties and parallel machines. 4OR 6(2):191–194
21. Rios-Solis YA, Sourd F (2008) Exponential neighborhood search for a parallel machine scheduling problem. Computers & Operations Research 35(5):1697–1712
22. Sourd F, Kedad-Sidhoum S (2003) The one-machine problem with earliness and tardiness penalties. Journal of Scheduling 6(6):533–549
23. Sun A, Hickman M (2008) The holding problem at multiple holding stations. In: Computer-aided systems in public transport, Springer, pp 339–359
24. Zhao J, Bukkapatnam S, Dessouky MM (2003) Distributed architecture for real-time coordination of bus holding in transit networks. Intelligent Transportation Systems, IEEE Transactions on 4(1):43–51
25. Zolfaghari S, Azizi N, Jaber MY (2004) A model for holding strategy in public transit systems with real-time information. International Journal of Transport Management 2(2):99–110